UNIVERSIDAD DE GRANADA

Departamento de Ciencias de la Computación
e Inteligencia Artificial

Application of Computational Intelligence Techniques in the Development of Intelligent Systems for Supporting Learning

Tesis Doctoral

Manuel Romero Cantal

Granada, Septiembre de 2014

Editor: Editorial de la Universidad de Granada
Autor: Manuel Romero Cantal
D.L.: GR 2319-2014
ISBN: 978-84-9083-366-7

UNIVERSIDAD DE GRANADA

Application of Computational Intelligence Techniques in the Development of Intelligent Systems for Supporting Learning

MEMORIA QUE PRESENTA

Manuel Romero Cantal

PARA OPTAR AL GRADO DE DOCTOR EN INFORMÁTICA

Septiembre de 2014

DIRECTOR

Dr. Juan Luis Castro Peña

Departamento de Ciencias de la Computación
e Inteligencia Artificial

La memoria titulada "*Application of Computational Intelligence Techniques in the Development of Intelligent Systems for Supporting Learning*", que presenta D. Manuel Jesús Romero Cantal para optar al grado de doctor, ha sido realizada dentro del Máster Oficial de Doctorado "*Soft Computing y Sistemas Inteligentes*" del Programa Oficial de Doctorado en "*Tecnología de la información y la Comunicación (D18.56.1)*" del Departamento de Ciencias de la Computación e Inteligencia Artificial de la Universidad de Granada bajo la dirección del doctor D. Juan Luis Castro Peña.

El doctorando Manuel Romero Cantal y el director de la tesis Juan Luis Castro Peña garantizamos, al firmar esta tesis doctoral, que el trabajo ha sido realizado por el doctorando bajo la dirección del director de la tesis y hasta donde nuestro conocimiento alcanza, en la realización del trabajo, se han respetado los derechos de otros autores a ser citados, cuando se han utilizado sus resultados o publicaciones.

El doctorando Manuel Romero Cantal ha realizado una estancia en el *Department of informatics*, departamento de investigación perteneciente a la *De Montfort University* de Leicester (Reino Unido). La estancia, de tres meses de duración, fue supervisada por el investigador Francisco Chiclana en los campos de *Toma de Decisiones y Lógica Difusa.*

De acuerdo a la normativa vigente para el reconocimiento de *Mención Internacional*, tanto el planteamiento de la investigación como las conclusiones se presentan en dos idiomas, español e inglés en nuestro caso. El desarrollo de la investigación se presenta únicamente en inglés, el idioma más extendido en la literatura científica.

Granada, Septiembre de 2014

El Doctorando

Fdo: Manuel Jesús Romero Cantal

El Director

Fdo: Juan Luis Castro Peña

Esta Tesis doctoral ha sido financiada por medio de la beca FPDI (Formación de Personal Investigador correspondiente a los Proyectos de Investigación de Excelencia) referencia P09-TIC-5011 de la Junta de Andalucía.

Agradecimientos

Quiero dedicar esta memoria en primer lugar a mis padres, por la excelente educación que me han dado y por su constante apoyo y cariño durante todo el camino recorrido. También a mi director de Tesis Juan Luís, clave en esta Tesis, por su paciencia, apoyo y disciplina, en mis buenos y malos momentos. A mis familiares más directos, mis tíos, mis primos, mis sobrinos y en especial a mi hermana Clara, mi guía espiritual y amiga, de la cual aprendo cada día. Y a Cristi, por aguantarme (que ya es mucho) y por completarme. Muchas gracias.

También quiero agradecer a todos mis amigos estos cuatro años de experiencias. A mis costras (sois muchos y no cabeis), por nuestras alegrías y nuestras penas, por todo lo vivido. A Juan y Pablo, mi segunda familia. Seguro que la próxima vez paramos el Sol. A Hitos, Castillos y Ernesto, por demostrarme que la amistad no entiende de tiempo ni distancia. También a Alvaro y María, por ser tan buenos conmigo siempre, por su preocupación y su consejo. Y sobre todo a Alex. Sin tí, nada de esto hubiese sido posible. Eres un ejemplo para mí, un modelo a seguir.

Por último, quiero citar a Edu de Virtual Solutions y a María Navarro, antigüo miembro del grupo de investigación, siempre dispuestos a echarme una mano. También a Miguel, mi compañero de estancia, apoyo indispensable en Leicester. Y a Mako, la reina de mi casa.

Estos cuatro años han sido muy intensos en muchos aspectos de mi vida. He aprendido y crecido con todos vosotros. Ya formáis parte de mí.

GRACIAS A TODOS

Resumen

Este proyecto de Tesis tiene por objetivo aportar propuestas significativas al campo de las plataformas e-learning, sistemas de apoyo a la enseñanza-aprendizaje sostenidos por medio de las Tecnologías de la Información y la Comunicación, tratando de encontrar soluciones especificas que favorezcan el proceso de aprendizaje y se cubran las necesidades de cada estudiante de forma personalizada.

En este trabajo proponemos un análisis de varios problemas de gran importancia en el contexto de los sistemas e-learning actuales, con el fin de darle respuesta mediante tecnologías de Inteligencia Computacional. Así, buscaremos soluciones afines a cada problema de forma independiente, respetando las restricciones impuestas por el marco e-learning. Además, presentaremos un sistema completo integrando cada solución individual, capaz de crear dinámicamente y de forma inteligente una secuencia estructurada de actividades de aprendizaje compuestas por contenidos, permitiendo así la adquisición de competencias de un dominio específico.

Nuestra hipótesis de partida es que varios de los problemas clásicos de las plataformas e-learning, mencionados a continuación, se podrían solucionar mediante técnicas especificas de Inteligencia Computacional. Entre dichos problemas identifcamos: (i) la gestión del contenido didáctico es competencia exclusiva del educador, y (ii) los estudiantes están forzados a seguir propuestas curriculares predefinidas y estáticas que no tienen en cuenta sus preferencias, necesidades y/o comportamientos.

Concretamente, se abordan los siguientes temas: (i) mecanismos de extracción y organización del conocimiento existente en el contenido didáctico, para obtener representaciones del dominio contenidas en estructuras de meta-datos, (ii) mecanismos de recuperación de información y visualización de información y conocimiento, que permitan la búsqueda de información, la exploración del dominio y la navegación a través de los recursos didácticos, y (iii) mecanismos de adaptación del entorno virtual, que permitan un cierto grado de flexibilidad curricular en función de las necesidades específicas de cada alumno. De forma complementaria, se concluye con el (iv) diseño y validación de un sistema de apoyo al aprendizaje que reúna todos los mecanismos anteriores.

Este trabajo es fruto de un proceso de ingeniería de carácter científico. Desde el punto de vista científico, hemos logrado interesantes resultados y contribuciones al estado del arte avaladas por diversas publicaciones científicas (Capítulo VII). Desde el punto de vista técnico, hemos desarrollado diferentes aplicaciones informáticas de amplio interés comercial.

Summary

The main goal of this phD Thesis is to offer meaningful proposals to the E-learning field in order to obtain specific solutions that improve the educative process and consider the individualised necessities of the students.

We propose an analysis of Computational Intelligence approaches in order to solve several of the main problems related to Educational Systems. Thus, we could obtain specific solutions to such problems, handling e-learning's restrictions. What is more, gathering the individual solutions we could present a complete system able to design a structured and dynamic sequence of learning activities that allows students to develop a range of domain-specific competences.

Our initial hypothesis is that specific Computational Intelligence techniques could handle a number of the main disadvantages proper of traditional e-learning systems: (i) the teacher is in charge of the whole management of the learning content, and (ii) students are forced to follow predefined and static learning schemes that do not consider their necessities, preferences and/or behaviours.

More concretely, following issues are here considered: (i) which techniques should be employed for extracting and organizing the knowledge underlying from the educational content, in order to obtain the representation of the domain through different meta-data structures; (ii) which techniques should be employed for retrieving information and for visualizing the information and the knowledge of the domain, allowing the search of information, the exploration of the domain and the navigation through the learning resources; and (iii) which techniques should be employed for adapting the content, in order to modify the learning strategies in function of specific students' necessities. Additionally, we consider (iv) the design and validation of a e-learning platform that gathers all the commented techniques.

This work is the result of an engineering process of scientific nature. From the scientific point of view, several valuable results and contributions to the state-of-the-art have been published in different scientific media (Chapter VII). From a technical point of view, different computer applications of commercial interest have been developed.

Contents

I Planteamiento de la Investigación y Revisión del Estado del Arte 1

1 Introducción . . . 1

2 Estado del Arte . . . 5

2.1 Principios teóricos y conceptos pedagógicos . . . 5

2.1.1 Principales Paradigmas Pedagógicos . . . 5

2.1.2 Objetivos del aprendizaje . . . 6

2.1.3 Otros conceptos . . . 6

2.2 Tecnología y Educación: Primeros Pasos . . . 8

2.3 Entornos de Aprendizaje Virtuales . . . 9

2.3.1 Objetos de Aprendizaje y Sistema de Gestión del Aprendizaje . . . 10

2.4 Aprendizaje Colaborativo Mantenido por Computador . . . 10

2.4.1 Sistemas Educativos Adaptativos y Tutores Inteligentes Virtuales . . 11

3 Objetivos . . . 13

4 Metodología . . . 14

I PhD Dissertation and Review of the State of the Art 17

1 Introduction . . . 17

2 State of the Art . . . 21

2.1 Theoretical principles and pedagogical concepts . . . 21

2.1.1 Main Pedagogical Principles . . . 21

2.1.2 Educational Objectives . . . 22

2.1.3 Other concepts . . . 22

2.2 Technology and Education: Firsts Steps . . . 24

2.3 Virtual Learning Environments . . . 25

2.3.1 Learning Objects and Learning Management Systems . . . 25

2.4 Computer-Supported Collaborative Learning . . . 26

2.4.1 Adaptive Educational Systems and Intelligent Tutoring Systems . . 26

3 Objectives . . . 29

4 Methodology 30

II Knowledge Extraction and Organization Methods for the Educational Content 33

1 Introduction 33

1.1 Taxonomy 34

1.2 Folksonomy 34

1.3 Ontology 35

2 ALKEx: Key Term Extraction system for Domain Taxonomy Construction from Learning Resources 37

2.1 Introduction 37

2.2 Related Works 38

2.2.1 General AKE methods 38

2.2.2 AKE methods in e-learning 40

2.3 Wikipedia-based Knowledge Source 40

2.3.1 Wikipedia-based dictionary of concepts 41

2.4 Term Frequency in Language 41

2.4.1 Term Frequency dictionary 42

2.4.2 Google-based simulated frequency 42

2.5 System Overview 43

2.5.1 Preprocessing Module 43

2.5.2 Candidate Extraction Module 45

2.5.3 Multi-domain Candidates' Procedure 45

2.5.4 Specific Candidates' Procedure 50

2.5.5 Taxonomic Indexation of the Learning Resources 50

2.6 Experiences with the System 51

2.6.1 Type of Learning Resources 51

2.6.2 Data Set 52

2.6.3 Design Principles 53

2.6.4 Setting Parameters 53

2.6.5 Experimental Results 54

2.6.6 The Effect of Key Term Division on Learning Resource Dataset . . . 56

2.7 Conclusions and Future Work 57

3 TRLearn System: Creating a Conceptually Extended Folksonomy from Scratch . . . 59

3.1 Introduction 59

3.2 Related Works 61

3.3 System Overview 63

3.3.1 Tag Extraction Module 64

3.3.2 Tag Recommendation Module 69
3.4 Experiences with the System 71
3.4.1 Dataset 73
3.4.2 Design principles 73
3.4.3 Setting parameters 73
3.4.4 Experimental evaluation 74
3.5 Conclusions 76
4 ORLearn: Recommending Concept and Relations for Learning Ontologies from Educational Resources 78
4.1 Introduction 78
4.2 Related Works 80
4.2.1 Concepts Extraction 81
4.2.2 Taxonomic Relations Extraction 81
4.2.3 Non-Taxonomic Relations Extraction 82
4.2.4 Ontology Learning systems 82
4.3 System Overview 83
4.3.1 Relation Extraction Module 85
4.4 Experiences with the System 89
4.4.1 Dataset 89
4.4.2 Design principles 90
4.4.3 Experimental evaluation 90
4.5 Conclusions and Future Work 91
5 Final Discussions and Future Work 93

III Information Retrieval and Visualization Methods for the Educational Domain 95
1 Introduction 95
2 FRLearn: Highly-Precise FAQ Retrieval System for Virtual Learning Environments . 98
2.1 Introduction 98
2.2 WordNet 99
2.3 Related works 99
2.4 System overview 101
2.4.1 FAQ list structure 101
2.4.2 WSSU extraction module 102
2.4.3 Query preprocessor module 104
2.4.4 Retrieval module 105
2.5 Experiences with the System 108
2.5.1 Dataset 108

2.5.2 Design principles 109
2.5.3 Setting parameters 109
2.5.4 FAQ Retrieval experiments 110
2.5.5 FAQ Retrieval results 111
2.6 Conclusions and Future Work 112
3 Integrating Knowledge and Information Visualization Techniques in Virtual Learning Environments 113
3.1 Introduction 113
3.1.1 Tag Clouds 114
3.1.2 Concept Maps 114
3.2 Related Works 116
3.2.1 Tag clouds 116
3.2.2 Concept Maps 118
3.2.3 Other Visualization Techniques 119
3.3 Obtaining Visual Representations from the Educational Domain 119
3.3.1 Tag Cloud Generation Algorithm 119
3.3.2 Concept Map Generation Algorithm 123
3.4 Visualization Techniques Supporting Real-life Applications 124
3.4.1 Integrating Tag Clouds into FRLearn 125
3.4.2 Integrating Tag Clouds into TRLearn 127
3.4.3 Integrating Concept Maps into ORLearn 129
3.5 Experimental Evaluations 130
3.5.1 Tag Cloud Generation algorithm: Testing the Performance of the Method 131
3.5.2 Concept Map Generation Algorithm: Testing the Usability of the Method 133
3.6 Conclusions and Future Work 134
4 Final Discussions and Future Work 136

IV Intelligent Adaptive Methods for Educational Systems 139

1 Introduction 139
2 Improving E-assessment: a Fuzzy Test Generation Framework 143
2.1 Introduction 143
2.2 Related works 145
2.3 Framework Overview 146
2.3.1 Test Items Properties 148
2.3.2 Objective Base 148

2.3.3 Acceptance Threshold Calculation module 152
2.3.4 Item Selection module 156
2.3.5 Test Evaluation Module 157
2.4 Experiences with the System 157
2.5 Conclusions and Future Work 160
3 TTutor: Integrating Fuzzy Logic into the Student Model Diagnosis of a Intelligent Tutoring System based on E-Assessment 161
3.1 Introduction 161
3.2 Related works 163
3.2.1 Overlay Model 163
3.2.2 Stereotype Model 164
3.2.3 Perturbation Model 164
3.2.4 Machine Learning Techniques 165
3.2.5 Cognitive Theories 165
3.2.6 Fuzzy Student Model 166
3.2.7 Bayesian Networks for Student Modelling 167
3.2.8 Ontology-based Student Model 168
3.3 System overview 168
3.3.1 Domain model 169
3.3.2 Student model 170
3.3.3 Tutoring Model: Curriculum planning 172
3.3.4 Tutoring Model: Diagnosis module 174
3.3.5 User Interface 179
3.4 Experiences with the System 183
3.5 Conclusions and Future Work 185
4 Final Discussions and Future Work 188

V Integrating the Intelligent Modules: Influence Factors, Architecture, Methodology and Validation of a Virtual Learning Environment 189
1 Introduction 189
2 Factors Affecting Effectiveness in E-learning 191
2.1 Student Self-regulation 191
2.2 Student Interaction 192
2.2.1 Student-Student Interaction 192
2.2.2 Student-Teacher Interaction 193
2.2.3 Student-Content Interaction 193
2.3 E-Assessment and Self-assessment 194

2.4 Formative/Summative Feedback 195
2.5 Blended Learning Environment 196
3 ivLearn: an System for Learning Supported by Computational Intelligence Mechanisms197
3.1 VLE Features 197
3.2 Design Architecture of ivLearn 199
3.2.1 Course management module 199
3.2.2 User Management Module 199
3.2.3 Resource Management Module 201
3.2.4 FAQ Retrieval Module 202
3.2.5 Taxonomy Module 203
3.2.6 Folksonomy Module 204
3.2.7 Ontology Module 204
3.2.8 Visualization Module 204
3.2.9 Assessment Module 205
3.2.10 Intelligent Tutoring Module 206
3.2.11 Student Information Module 207
3.2.12 Collaborative Module 208
3.3 Using ivLearn: Definition of the Regulatory Methodology 208
3.3.1 Learning stages 209
4 Examining the Effects of ivLearn: Cases of Study 213
4.1 Research approach 213
4.2 Case of study (year 2012/13) 214
4.2.1 Participants 215
4.2.2 Learning context 215
4.2.3 Research results: student satisfaction 216
4.2.4 Research results: student achievement 216
4.3 Case of study (year 2013/14) 218
4.3.1 Participants 219
4.3.2 Learning context 219
4.3.3 Research results: student achievement 220
5 Final Discussions and Future Work 222

VI Conclusions and Future Work 225
1 Conclusions 225
1.1 Knowledge Extraction and Organization Methods for the Educational Content 225
1.2 Information Retrieval and Visualization Methods for the Educational Domain 227

1.3 Intelligent Adaptive Methods for Educational Systems 228

1.4 Integrating the Intelligent Modules into a real Virtual Learning Environment 228

1.5 General Conclusions 229

2 Future Work 230

VI Conclusiones y Trabajos Futuros 233

1 Conclusiones 233

1.1 Métodos de Extracción y Organización del Conocimiento para el Contenido Didáctico 233

1.2 Métodos de Recuperación de Información, y de Visualización de Información y Conocimiento para el Contenido Didáctico 235

1.3 Métodos de Adaptación Inteligente del Entorno Educativo 236

1.4 Integración de los Métodos Inteligentes en un Sistema de Apoyo al Aprendizaje Real 237

1.5 Conclusiones Generales 238

2 Trabajos Futuros 239

VII List of Publications: Submitted, Published, and Accepted Articles 241

Appendices 249

1 ALKEx: Google-based regression analysis process 249

Bibliography 251

List of Figures

i.1 Taxonomía de Bloom: dominio cognitivo 7

i.2 Taxonomía de Bloom: dominio afectivo 7

i.1 Bloom’s taxonomy of educational objectives: cognitive domain 23

i.2 Bloom’s taxonomy of educational objectives: affective domain 23

ii.1 Example of how a Wikipedia article’s structure defines a concept 41

ii.2 Concept extracted from ’Biological neural network’ article 42

ii.3 System architecture 44

ii.4 Related concepts for ’neural network’ term 47

ii.5 Prototipical interface of a index application supported by the taxonomy 51

ii.6 Example of GIFT questions 52

ii.7 Example of tasks 52

ii.8 Performance evaluation extended by ALKEx modules 57

ii.9 Tag recommendation scheme 60

ii.10 TRLearn architecture overview 63

ii.11 “Intensity of relative document frequency” fuzzy variable 67

ii.12 “intensity of relative corpus frequency” fuzzy variable for 1-grams 68

ii.13 Ordered list of candidate tags presented to the user 71

ii.14 Relation Recommendation Interface 84

ii.15 ORLearn architecture overview 84

iii.1 System architecture 102

iii.2 Tag Cloud visualization technique 115

iii.3 Concept Map example 115

iii.4 Tag cloud representation 122

iii.5 Summary model interface 124

iii.6 Conceptual-level concept map interface 125

iii.7 FAQ cloud usability 126

iii.8 Extended architecture of FRLearn 127
iii.9 Output interface for the user query: 'Where can I find information about official masters in the UGR?' 128
iii.10 Extended architecture of TRLearn 129
iii.11 Interface for the individual tag "web ontology language" 129
iii.12 Extended architecture of ORLearn 130

iv.1 The four-component ITS architecture 141
iv.2 Test Generation Framework architecture 147
iv.3 Assessment objectives selection interface 149
iv.4 Type of Item Objective defined using percentages 150
iv.5 "Type of Item" Objective defined using linguistic labels 150
iv.6 Difficulty fuzzy variable 151
iv.7 (Acceptable) Frequency fuzzy variable 151
iv.8 Test Relevance to issue 1 using the "exactly" quantifier 152
iv.9 Test Results interface 158
iv.10 TTutor architecture 169
iv.11 Learning stage definition interface 170
iv.12 Student model attributes 171
iv.13 Fuzzy sets defining the level of knowledge in a concept 175
iv.14 Fuzzy variables: (a) Improvement in the percentage of variation of the DM of c_i, (b) Level of relation between c_i and c_j, given c_i, and (c) Variation of DM of c_j . . . 177
iv.15 Learning subject index interface 180
iv.16 Test results interface 182
iv.17 Student model interface 184
iv.18 Teacher interface showing a summary of the current state of each student in each learning subject 185

v.1 Architecture overview 200
v.2 Collaborative Tasks sequence diagram 203
v.3 Evaluation form of a real student in ivLearn system 207
v.4 Sequence of stages of the ivLearn regulatory methodology 210
v.5 Bar chart of the distribution of grades 219

List of Tables

II.1 Key terms extracted from the 'principle of knowledge' related question-answer pair 38

II.2 Key terms from a computer science question-answer pair 44

II.3 Candidate Extraction algorithm 46

II.4 Term Sense Disambiguation algorithm 48

II.5 Key terms extracted from two AI-related documents 49

II.6 Percentage of key terms by type 53

II.7 Key term extraction results from question-answer learning task collection 56

II.8 Key term extraction results from GIFT quiz question collection 56

II.9 Key term extraction results of each independent module of ALKEx in question-answer task collection 57

II.10 Linguistic patterns used to recognize candidate concepts 64

II.11 Tag weighting algorithm 66

II.12 Example of candidate extraction and weighting process 70

II.13 Weight adaptation algorithm after successful tag selection 72

II.14 Proportion of accepted resources by issue 74

II.15 Proportion of accepted resources by issue 74

II.16 Evaluation of recommended tags for the course resources 75

II.17 Results of the 5 items of the satisfaction questionnaire 76

II.18 Tag selected by the two groups 77

II.19 Superclass Relation Rule 86

II.20 Subtopic linguistic patterns 87

II.21 Subtopic Relation Rule 87

II.22 Superclass Relation Rule 88

II.23 Subordinate Relation Rule 88

II.24 Content-based Relation Rule 89

II.25 Concepts for the evaluation of the relation recommendation process 90

II.26 Evaluation of recommended relations for the course resources 91

III.1 Part-of-speech patterns of Conceptual-WSSUs 103

III.2 WSSU extraction from Linux FAQ entries 104

III.3 Expanded words of the 'Where is the professor Michael's office?' query 105

III.4 Example of expanded words reduction and final candidate selection 107

III.5 Example of output FAQ entries 108

III.6 Details of FAQ list 109

III.7 Ordered rank of each method in Restaurant FAQ 111

III.8 Ordered rank of each method in Linux FAQ 111

III.9 Ordered rank of each method in UGR FAQ 111

III.10 Example of tag selection performance 121

III.11 Ordered rank of each method on Restaurant FAQ list 132

III.12 Ordered rank of each method on Linux FAQ list 132

III.13 Ordered rank of each method on UGR FAQ list 133

III.14 Results of the 5 items of the usability questionnaire 134

IV.1 User test request 144

IV.2 Results of the 5 items of the usability questionnaire 159

IV.3 Candidate Test Item selection algorithm 173

IV.4 Candidate Test Item ordering algorithm 174

IV.5 Fuzzy rules for the expansion of dependencies process. Abbreviations: (LK) Level of Knowledge, (Rel(c_i/c_j)) Level of relation of c_i and c_j, given c_i, ($\Delta DM(c_i)$) Improvement in the percentage of variation of the degree of mastery of c_i, ($\Delta DM(c_j)$) Resulting Variation of DM of c_j 178

IV.6 Relation existent between the concept of the example 178

IV.7 Items and results of the questionnaire 186

V.1 Items of the Prospective Andalucian Centre questionnaire regarding the methodology and the learning resources 214

V.2 Demographic distribution of participants in control and experimental groups 215

V.3 Distribution of proposed resources by experimental group 215

V.4 Results of the 11 items of the Prospective Andalucian Centre questionnaire 217

V.5 Ranges for the classification of student by degree of usage 217

V.6 Distribution of grades obtained in the experimental group by usage classification . 218

V.7 Distribution of grades obtained by students in control group 218

V.8 Demographic distribution of participants in control and experimental groups 219

V.9 Ranges for the classification of student by degree of usage 221

V.10 Distribution of grades obtained in the experimental group by usage classification . 221

Chapter I

Planteamiento de la Investigación y Revisión del Estado del Arte

1 Introducción

Desde la última década del siglo XX, las Tecnologías de la Información y la Comunicación (TIC) han sufrido un desarrollo excepcional. La naturaleza de este progreso ha desencadenado la evolución de los pilares que sustentan nuestra sociedad (sin ir más lejos, vivimos en la "sociedad de la información"). Precisamente, la base de nuestro comportamiento social se basa en la transmisión de información de unos individuos a otros mediante el aprendizaje. Durante siglos, la transferencia de conocimiento se realizaba de forma generacional, sustentada en mayor medida por escuelas y universidades. Ahora, la información está siempre disponible de forma global gracias al crecimiento de Internet, lo cual ha provocado un gran impacto en los paradigmas clásicos de la transmisión de conocimiento, esto es, de la educación. El modelo clásico de instrucción presencial comparte protagonismo con la preparación on-line. Este nuevo modelo de enseñanza sostenido por medio de las TIC se conoce comúnmente como *e-learning* (electronic learning, en inglés), y tiene como principales objetivos facilitar un aprendizaje interactivo, flexible (en tiempo y lugar) y centrado en los estudiantes, mediante entornos virtuales. La ausencia de restricciones para adquirir información modifica el rol del estudiante, que pasa a ser el actor principal del proceso educativo. En consecuencia, el papel del profesor también se ve modificado, que debe ahora estimular al estudiante para que tome control de su aprendizaje de forma activa. El paradigma clásico de enseñanza dirigido exclusivamente por el profesor claudica en favor del paradigma centrado en el alumno.

En el contexto de la economía actual, el mercado laboral impone una competencia voraz entre los trabajadores, que necesitan de una preparación extraordinaria que les permita estar siempre un paso por delante del resto. Así, con la idea de proporcionar la máxima preparación posible para sus alumnos, las organizaciones docentes (inclusive el propio entrenamiento corporativo) han impulsado en gran medida el desarrollo de metodologías e-learning, mediante sistemas de instrucción basados en web para impartir sus cursos a través de Internet. El sector de la educación se ha desarrollado como uno de los mercados más importantes para inversiones de empresas y privadas. Sin ir más lejos, la Comisión Europea ha trabajado durante los últimos 10 años en desarrollar, mejorar e implementar el concepto de e-learning [CotEC01]. Así mismo, el Espacio Europeo de Enseñanza originado por el Proyecto de Bolonia persigue la implantación del paradigma centrado en el alumno, entre otras medidas, con el propósito de obtener un mayor grado de independencia e implicación de los estudiantes [EC09, Hei05, Lou01].

La cada vez mayor transcendencia económica y social del e-learning se refleja en la aparición de nuevas líneas de investigación, que aúnan tecnología y educación con el objetivo de obtener plataformas de apoyo al aprendizaje cada vez más funcionales. En la actualidad existen numerosas aplicaciones modernas diseñadas para enriquecer el aprendizaje. Por ejemplo, las plataformas de aprendizaje actuales incluyen entornos de colaboración siguiendo las pautas de la Web 2.0 ó módulos inteligentes capaces de adaptar el propio entorno de aprendizaje en base a las características de cada usuario. Así, las plataformas de aprendizaje proporcionan a los estudiantes gran libertad para desarrollar sus propios planes de aprendizaje, para construir y compartir conocimiento, para buscar información complementaria .... Desafortunadamente, la tendencia de uso actual de las plataformas e-learning en entornos reales se basa en la gestión convencional de cursos y contenido digitalizado [DK12]. Los sistemas clásicos de este tipo presentan ciertas deficiencias asociadas a la sobrecarga de información. En entornos donde el contenido es extenso, complejo y mal estructurado se pueden dar situaciones de sobrecarga cognitiva y problemas de desorientación por parte de los estudiantes. Esto provoca dificultad adicional para captar y aprender el dominio de conocimiento [BC94]. Por este motivo, surge el interés de contar con herramientas cognitivas que ayuden solventar las limitaciones comentadas [KL97]. De esta forma, sería interesante contar con herramientas capaces de organizar, visualizar y/o navegar a través de los recursos educativos de forma (semi)automática e inteligente. Ésto facilitaría en gran medida la labor del profesor, o administrador del sistema. Concluyendo, el desarrollo de herramientas de explotación del conocimiento aplicadas al contenido didáctico podría potenciar en gran medida la aptitud de los sistemas e-learning. En particular, detectamos tres ámbitos de tratamiento:

- Extracción y organización del conocimiento a partir del contenido didáctico: comprende la extracción, organización y almacenado del conocimiento contenido en el contenido didáctico, en estructuras de meta-datos accesibles, extensibles y que se puedan compartir. En particular, consideramos varios niveles de riqueza semántica. Partiendo de términos clave simples, abordaremos la adquisición inteligente de conocimiento cada vez más complejo.

- Recuperación de información y visualización del dominio didáctico: comprende dos procesos de explotación del conocimiento del dominio. Por una parte, consideramos la recuperación de recursos didácticos en base a criterios del usuario relacionados con el dominio (en concreto, consultas formuladas en lenguaje natural). Ante grandes volúmenes de datos, tanto estudiante como profesores podrán acceder a los recursos educativos de forma eficaz y eficiente. Por otra parte, consideramos la representación visual y navegable del dominio didáctico. En particular, tomamos en consideración técnicas de visualización de conocimiento e información. Así, los estudiantes contarán con herramientas que les permita captar el esquema conceptual de la materia didáctica impartida de forma efectiva. Además, el esquema facilitará la navegación a través de los recursos educativos contenidos en el sistema.

- Adaptación inteligente del entorno didáctico: comprende la adaptación del entorno virtual de aprendizaje en función de las características, necesidades y/o actuación del estudiante. En este trabajo, consideramos los siguientes dos niveles. En primer lugar, nos centramos en la composición automática de recursos de evaluación a partir de una configuración de objetivos dada por el usuario. Así el usuario podrá seleccionar los objetivos de aprendizaje que desee para generar recursos que se ajusten a ellos. En segundo lugar, consideramos un método de adaptación curricular capaz de acondicionar los objetivos y el entorno de aprendizaje en función de cada estudiante. Ahora será el sistema el encargado de obtener automáticamente la configuración de objetivos actuales del estudiante a partir de su nivel de conocimiento sobre la materia, no solo para generar recursos elementales, si no para seleccionar y adaptar estrategias de evaluación más complejas.

En computo, la aplicación de cada solución individual como parte de un sistema completo daría lugar a una herramienta con la habilidad de facilitar la labor tanto de profesores como de los alumnos.

Esta Tesis doctoral tiene por finalidad aportar avances científico-tecnológicos al estado del arte en forma de soluciones inteligentes a los problemas derivados de la extracción y organización del conocimiento, recuperación y representación visual del dominio, y adaptación del entorno en sistemas e-learning. Nuestra *metodología* consistirá en estudiar tales problemas de forma independiente, para así desarrollar estrategias aplicables a cada ámbito individual. Para ello, nos valdremos de técnicas de Inteligencia Computacional (IC). Nuestros *objetivos*, por tanto, son dos: (i) dar solución a los problemas abordados, mediante un proceso tecnológico de carácter científico, y (ii) componer, como valor añadido, un sistema final que albergue cada solución individual y facilite una validación global.

Comenzaremos con un estudio de los métodos de extracción y organización del conocimiento didáctico en sistemas e-learning. Estudiaremos tres configuraciones atendiendo al nivel de riqueza considerada: (i) términos clave, (i) etiquetas conceptuales y (iii) conceptos y relaciones. El estudio de este problema tiene un interés justificado dados los requisitos propios del e-learning. La organización de conocimiento resultante ha de estar bien ajustada en el dominio docente, con el fin de evitar los problemas derivados de la sobrecarga de información. Esto limita el uso de técnicas genéricas de extracción y organización de conocimiento. Este estudio nos proporcionará una base sólida sobre la que desarrollar métodos sofisticados que ofrezcan solución al resto de problemas detectados.

Continuaremos por un análisis de los métodos de recuperación de información, y de visualización del dominio didáctico. Analizaremos primero los sistemas de recuperación de información basados en consultas de usuarios. Si bien su uso es común en el marco de las tecnologías de acceso a la información, no ocurre así en el campo del e-learning. En segundo lugar, estudiaremos distintas técnicas de visualización de información y conocimiento. Más concretamente, examinaremos las técnicas de visualización conocidas como Mapas Conceptuales y Nubes de Etiquetas. El empleo de Mapas Conceptuales es común en el ámbito didáctico dada su capacidad para representar esquemáticamente los conceptos y relaciones más importantes del dominio. Por su parte, las Nubes de Etiquetas cobran especial interés en las comunidades virtuales de Internet, como herramienta de exploración y navegación.

Seguiremos nuestro estudio abordando los mecanismos inteligentes de adaptación del entorno didáctico. Consideraremos en primer lugar la generación de recursos en base a preferencias explicitas de los usuarios. En concreto, propondremos un método de generación automática de tests de evaluación. El campo relacionado con la generación de tests de evaluación adaptables presenta un creciente interés dentro del ámbito e-learning, y responde a la necesidad de contar con instrumentos dinámicos que soporten las actividades de evaluación y auto-evaluación propias y necesarias en el aprendizaje. En segundo lugar, analizaremos la gestión y adaptación inteligente del entorno en base a características extraídas automáticamente por el sistema. En detalle, desarrollaremos un método de adaptación curricular del estudiante, en función de los niveles de conocimiento que presente frente a sus objetivos de aprendizaje actuales. Este sub-problema se enmarca dentro de los campos de los Sistemas Educativos Adaptativos y los Tutores Inteligentes Virtuales, de gran importancia y difusión en la actualidad.

Concluiremos de forma complementaria con la anexión de los métodos anteriores en una plataforma de ayuda al aprendizaje completa. Para ello, diseñaremos la arquitectura y funcionalidad de la plataforma, así como una metodología de regulación de su uso como parte de un curso universitario. De esta manera pretendemos dar valor añadido a esta investigación y podremos, al mismo tiempo, establecer una validación formal de este trabajo de forma conjunta. Para ello, analizaremos y evaluaremos su aplicación en un curso universitario de ingeniería durante dos años académicos consecutivos.

No es objetivo de esta Tesis realizar un análisis exhaustivo de todas las tecnologías y métodos aplicables al campo del e-learning disponibles desde un punto de vista teórico, sino analizar factores reales e influyentes para así proponer una solución concreta en forma de aplicaciones informáticas que se ajusten a las demandas impuestas por la sociedad de la información en el campo de la educación. Este trabajo es, por tanto, el fruto de un proceso de ingeniería de índole científica. Con esto, pretendemos enfatizar el carácter de aplicabilidad y transferencia tecnológica del estudio. Por otro lado, los desarrollos teóricos y métodos propuestos en este trabajo han sido, en su mayoría, publicados en diferentes revistas científicas y expuestos en congresos.

Este proyecto de Tesis ha sido realizado dentro del Máster oficial de Doctorado *Soft Computing y Sistemas Inteligentes* del programa oficial de Doctorado en “Tecnología de la información y la Comunicación” del Departamento de Ciencias de la Computación e Inteligencia Artificial de la Universidad de Granada, bajo la dirección del doctor D. Juan Luis Castro Peña. El proyecto, de cuatro años de duración, ha sido financiado mediante la beca FPDI (Formación de Personal Investigador correspondiente a los Proyectos de Investigación de Excelencia) de la Junta de Andalucía, y ha sido desarrollado en el Centro de Investigación en Tecnologías de la Información y las Comunicaciones (CITIC-UGR).

Dedicaremos este capítulo a revisar el Estado del Arte en lo referente a los sistemas e-learning y a exponer los objetivos de Tesis y la metodología seguida. El resto de la memoria de Tesis se estructura de la siguiente manera. El Capítulo II se dedica a las técnicas de extracción y organización del conocimiento del dominio subyacente al contenido didáctico. En el Capítulo III nos centraremos en técnicas de recuperación de información y de visualización de información y conocimiento del dominio didáctico. El Capítulo IV está dedicado a los mecanismos inteligentes de generación y adaptación del entorno didáctico. A continuación, presentamos la arquitectura y funcionalidad del sistema e-learning completo, la metodología de aprendizaje que regula su uso, y el proceso de validación formal de la propuesta, en el Capítulo V. Finalmente, presentaremos las conclusiones y trabajos futuros en el Capítulo VI .

2 Estado del Arte

En esta sección presentamos una revisión del estado del arte referente a la evolución de los sistemas de apoyo al aprendizaje. Si bien este trabajo está enfocado bajo un punto de vista tecnológico, es imprescindible comenzar con un estudio de los principios pedagógicos que rigen el proceso educativo. A continuación, examinaremos el desarrollo histórico de los sistemas e-learning. Así mismo, se introducirán algunos conceptos básicos utilizados en el desarrollo de este trabajo.

2.1 Principios teóricos y conceptos pedagógicos

Las particularidades propias del proceso formativo dotan a nuestro problema de un carácter interdisciplinar. Aunque este trabajo este enfocado desde el punto de vista tecnológico de las Ciencias de la Computación, es imprescindible conocer los aspectos puramente pedagógicos implicados en el proceso de captación de conocimiento, o proceso de aprendizaje. En esta sección presentamos los modelos teóricos más extendidos, desde el punto de vista de la Pedagogía. Cabe destacar que no se proporciona un estudio exhaustivo, dada la complejidad y extensión del campo. Para obtener un nivel mayor de detalle, el lector puede consultar un estudio extenso de la materia en [Sch12].

2.1.1 Principales Paradigmas Pedagógicos

En primer lugar, comentaremos las características de las principales teorías o paradigmas pedagógicos desarrollados durante el siglo XX. La complejidad del proceso de aprendizaje ha motivado el desarrollo de varios modelos compuestos por una serie de reglas metodológicas. Precisamente, el desarrollo de propuestas tecnológico-educativas requiere de un análisis previo de las posibles estrategias de aprendizaje. Dichas estrategias funcionarán como modelo lógico del sistema. La comunidad científica en el ámbito de la pedagogía distingue tres corrientes principales: conductismo, cognitivismo y constructivismo [EN93, Sch12].

El *conductismo* es una corriente de la psicología inaugurada por John B. Watson (1878-1958) que defiende el empleo de procedimientos estrictamente experimentales para estudiar el comportamiento observable. Aborda las leyes del aprendizaje mediante el condicionamiento estímulo-respuesta. Su fundamento teórico está basado en la premisa de que un estimulo produce una respuesta directa, siendo ésta el resultado de la interacción entre el organismo que recibe el estimulo y el ambiente en que encuentra [Sch12]. Supongamos que un profesor escribe en una pizarra la ecuación "2 + 3 = ?". Esta ecuación se corresponde con el estímulo, que motiva la respuesta directa "6" por parte del estudiante. Así, el aprendizaje se considera como un proceso mecánico en el que se trata de estimular un cambio de comportamiento en los estudiantes, mediante refuerzos positivos y negativos. No tiene en cuenta los procesos mentales implicados (la mente es una caja negra), y censura la participación activa del estudiante.

El *cognitivismo*, como su propio nombre indica, nace a partir de los modelos provenientes de las ciencias cognitivas, durante los años 50. La atención se centra en la promoción de procesos cognitivos más complejos que el simple comportamiento observable (propio del conductismo), como por ejemplo la resolución de problemas, el lenguaje, la formación conceptual y el procesado de la información [Sne83]. Las teorías cognitivas se centran en la conceptualización de los procesos de aprendizaje de los estudiantes: cómo se recibe, organizar, almacena y se devuelve la información en la mente [EN93]. El proceso de adquisición de conocimiento se divide en tres etapas [Sch12]. Primero, se recibe la información a través de los sentidos del individuo. Posteriormente, se almacena en la memoria a corto plazo. Por último, se transfiere a la memoria a largo plazo. El conocimiento, por su parte, se considera como un esquema interno de estados mentales. Este esquema se modifica a medida que el individuo aprende. Durante el proceso, la nueva información se compara con el esquema actual, que puede ser modificado o extendido en función del conocimiento adquirido.

El último modelo pedagógico aquí presentado es el del *constructivismo*. Esta teoría equipara el aprendizaje con la creación de significado a partir de la experiencia [BCDP92]. Al igual que la corriente cognitivista, se considera el aprendizaje como una actividad mental, aunque existen puntos de diferenciación. Mientras la teoría cognitivista considera a la mente como una herramienta de referencia para el mundo real, la teoría constructivista afirma que la mente filtra la información percibida del mundo para desarrollar una realidad propia e individualizada [Jon92]. Así, se considera que no se realiza una transferencia de información del mundo real a la memoria del individuo, si no que él construye una interpretación subjetiva del mundo real, basada en sus propias experiencias e interacciones. Por tanto, la representación interna del conocimiento está en constante evolución.

Como se puede observar, cada teoría presenta implicaciones y limitaciones distintas, en función del tipo de conocimiento a adquirir y de las metas del aprendizaje. En el capítulo V se ofrecen detalles adicionales referentes a la integración del constructivismo como parte del diseño lógico de un sistema de apoyo al aprendizaje.

2.1.2 Objetivos del aprendizaje

En 1956, Bloom y Krathwohl publicaron la llamada Taxonomía de Objetivos Educacionales [BK56]. Muchos de los estudios relacionados con el diseño y evaluación de sistemas de apoyo al aprendizaje utilizan los conceptos comentados en dicha taxonomía. A continuación, presentamos sus aspectos básicos.

La taxonomía identifica tres dominios diferentes: cognitivo, afectivo y psicomotor. Cada dominio comprende un conjunto jerarquizado de comportamientos de aprendizaje que favorece la adquisición de la nueva habilidad, conocimiento y/o actitud.

El nivel cognitivo atiende al funcionamiento del cerebro, y a los procesos cognitivos relacionados con el aprendizaje, como la comprensión o la construcción de conocimiento. Incluye el reconocimiento de los hechos concretos, los patrones procedurales y los conceptos que intervienen en el desarrollo de habilidades y destrezas intelectuales. Hay seis categorías principales de los procesos cognitivos, que se detallan en la Figura i.1.

El nivel afectivo describe los aspectos emocionales del aprendizaje, como la motivación, el compromiso o el comportamiento del estudiante. Existen cinco categorías, descritas en la Figura i.2.

El aspecto psicomotor se centra en los procesos de manipulación de herramientas o instrumentos. Este dominio se aleja del ámbito de los sistemas actuales de apoyo al aprendizaje, por lo que no será aquí comentado. Se puede consultar información al respecto en [Sim70].

2.1.3 Otros conceptos

A continuación, desglosaremos el resto de conceptos pedagógicos mencionados en esta tesis.

- *Adaptación curricular*: decisiones relativas a la organización de los recursos didácticos dirigidos al análisis de los diferentes alumnos. Adecuación de los objetivos educativos, modificación de contenidos, la metodología de aprendizaje a seguir y la modificación de los criterios de evaluación.

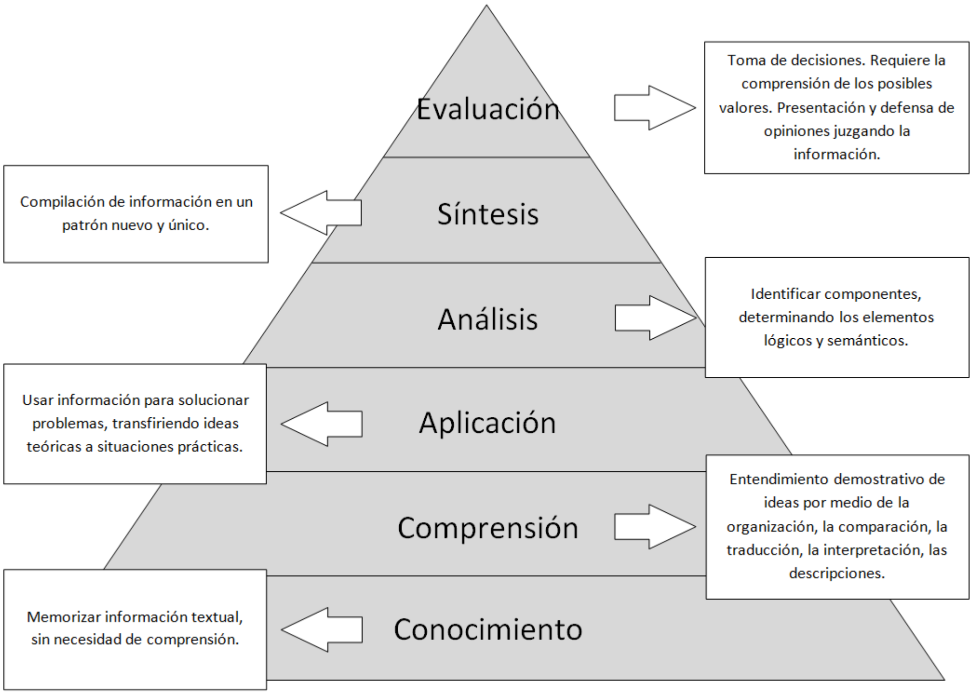

Evaluación
Toma de decisiones. Requiere la comprensión de los posibles valores. Presentación y defensa de opiniones juzgando la información.
Síntesis
Compilación de información en un patrón nuevo y único.
Análisis
Identificar componentes, determinando los elementos lógicos y semánticos.
Aplicación
Usar información para solucionar problemas, transfiriendo ideas teóricas a situaciones prácticas.
Comprensión
Entendimiento demostrativo de ideas por medio de la organización, la comparación, la traducción, la interpretación, las descripciones.
Conocimiento
Memorizar información textual, sin necesidad de comprensión.

Figure i.1: Taxonomía de Bloom: dominio cognitivo

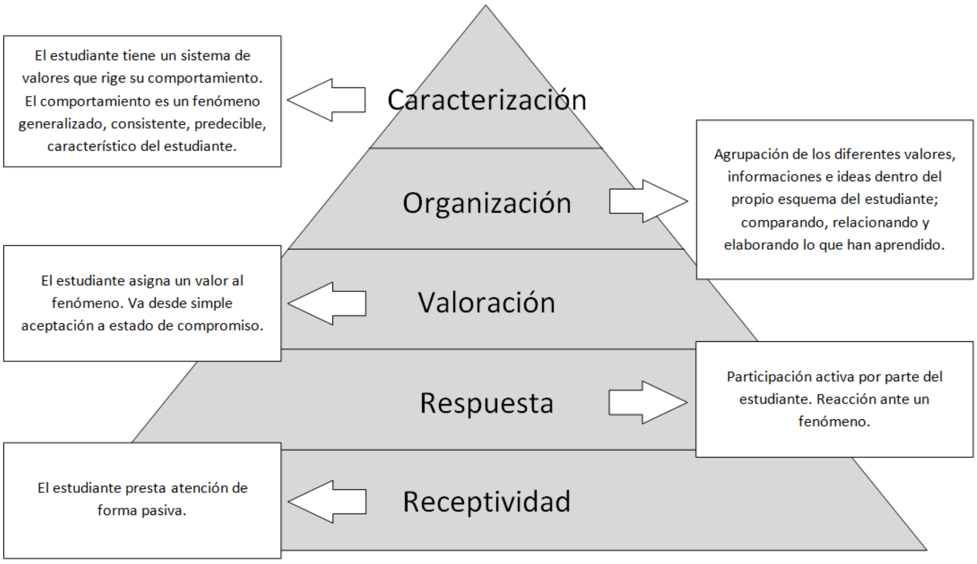

El estudiante tiene un sistema de valores que rige su comportamiento. El comportamiento es un fenómeno generalizado, consistente, predecible, característico del estudiante.
Caracterización
Organización
Agrupación de los diferentes valores, informaciones e ideas dentro del propio esquema del estudiante; comparando, relacionando y elaborando lo que han aprendido.
El estudiante asigna un valor al fenómeno. Va desde simple aceptación a estado de compromiso.
Valoración
Respuesta
Participación activa por parte del estudiante. Reacción ante un fenómeno.
El estudiante presta atención de forma pasiva.
Receptividad

Figure i.2: Taxonomía de Bloom: dominio afectivo

- *Auto-regulación* o *aprendizaje auto-regulado*: grado en el que el estudiante es capaz de centrar de forma sistemática sus pensamientos, sentimientos y acciones en el logro de sus objetivos didácticos, participando de forma activa en el proceso de aprendizaje [Zim89, Zim86].

- *Auto-evaluación*: forma de evaluación donde el estudiante se aplica a si mismo un instrumento que explora el grado de conocimiento que ha adquirido. Su propósito es el de identificar los defectos presentes en el proceso de aprendizaje para corregirlos, estableciendo para ello actividades de solución.

- *Nivel de satisfacción de estudiante*: se refiere a la percepción (positiva/negativa) que tiene el estudiante sobre el proceso de aprendizaje, motivado por la estrategia de aprendizaje dada.
- *Nivel de éxito del estudiante*: se refiere a la calidad del conocimiento adquirido por los estudiantes tras el proceso formativo amparado por la estrategia de aprendizaje correspondiente.
- *Evaluación formativa*: actividad sistemática y continua de evaluación, que tiene por objeto proporcionar la información necesaria sobre el proceso educativo, para reajustar sus objetivos, revisar críticamente los planes, los programas, los métodos y recursos, orientar a los/las estudiantes y retroalimentar el proceso mismo. La evaluación formativa tiene que ver más con los procesos de aprendizaje que con los productos del mismo.
- *Evaluación sumativa*: actividad de evaluación al final de cada proceso de aprendizaje, con el objeto de determinar el grado de consecución de los objetivos de aprendizaje por parte del alumnado. Este grado determina la posición relativa de cada alumno en el grupo y lo sitúa en determinados niveles de eficacia, marcados habitualmente (y establecidos normativamente) por una escala de calificaciones conocida.

2.2 Tecnología y Educación: Primeros Pasos

Aunque el concepto del e-learning está actualmente vinculado a la educación a través de Internet, los primeros modelos que aunaban tecnología y educación datan la segunda mitad del siglo XX.

Los primeros pasos en el campo se produjeron por influencia de la industria militar. Durante la Segunda Guerra Mundial, las características del conflicto potenciaron el desarrollo de métodos eficientes de adiestramiento militar. Se comenzaron a utilizar películas de instrucción como método de entrenamiento de los sistemas armamentísticos. También se desarrollaron unos dispositivos mecánicos de entrenamiento militar llamados "phase check" que permitían abordar cada proceso de entrenamiento (por ejemplo, montaje o desmontaje de algún componente) como una secuencia estructurada de pasos. De esta forma, se establecieron las bases del diseño instruccional. Precisamente, fue Skinner, profesor de la Universidad de Harvard, quien introdujo el concepto de *instrucción programada*, desarrollando la llamada *teaching machine* [Ski60]. La instrucción programada consiste en la división modular y secuencial de una tarea didáctica mediante frames (pequeños módulos en los que se descomponen los materiales de entrenamiento). Por tanto, el desarrollo de la habilidad se realizaba de forma incremental, mediante un proceso continuo de sondeo que potenciaba las respuestas correctas [OHC+12], siguiendo el esquema pedagógico conductista.

Durante los años 60, la instrucción programada evolucionó hacia un nuevo concepto conocido como *aprendizaje basado en computador* o *instrucción asistida por computador*, término empleado para clasificar aquellos proceso de aprendizaje que tienen a los computadores como componente clave. Los dispositivos mecánicos dieron paso a modernas estrategias sostenidas por computadores. Uno de los primeros sistemas generalizados de enseñanza asistida por computador fue PLATO (*Programmed Logic for Automated Teaching Operations*) [BBL62], creado por Don Blitzer en la Universidad de Illinois en 1960. El sistema, considerado como el precursor directo de los entornos e-learning actuales, incluía terminales gráficos de alta resolución e implementaba el lenguaje de programación TUTOR. Dicho lenguaje permitía crear módulos de enseñanza interactivos para alumnos y profesores y establecía mecanismos de comunicación entre usuarios a través de notas electrónicas. Pocos años después, los profesores de la Universidad de Psicología de Stanford Patrick Suppes y Richard C. Atkinson comenzaron a utilizar un sistema de instrucción asistido por computador [Sup66] para enseñar matemáticas y lectura a niños en las escuelas elementales de Palo Alta, en

1966. La instrucción asistida por computador comienza a incorporar estrategias cada vez más complejas de aprendizaje, incluyendo el uso de cuestionarios de evaluación, juegos o monitorización de los alumnos [JJS85].

La evolución de los computadores de sobremesa durante los años 80 impulsó el uso de la tecnología en la las organizaciones docentes, introduciendo aplicaciones pedagógicas a nivel de aula. Sin embargo, su salto definitivo se produjo con la creación de la *World Wide Web*. La posibilidad de contar con entornos distribuidos de aprendizaje y contenido digital supuso el impulso necesario para adquirir una escala global. Surge así el término *aprendizaje basado en web*, que define aquellos marcos en los cuales la interacción entre profesor y estudiante se realiza vía web, dando paso a los *entornos de aprendizaje virtuales*.

2.3 Entornos de Aprendizaje Virtuales

El auge de Internet y de las tecnologías basadas en web favoreció el desarrollo de entornos virtuales dedicados. Un Entorno de Aprendizaje Virtual (EAV) es una plataforma diseñada para gestionar los procesos de aprendizaje a través de recursos educativos. En este trabajo, notamos *recurso educativo* como cualquier elemento educativo mantenido por tecnología que pueda ser utilizado o referenciado durante el proceso de aprendizaje. Los EAV establecen un marco de reparto de información donde los profesores pueden diseñar actividades de aprendizaje, tanto individuales como colaborativas. Las actividades se estructuran en cursos e han de incluir suficientes recursos para ayudar a los estudiantes a alcanzar sus objetivos [DSS$^+$02, Wel07]. Entre los entornos más utilizados se encuentran WebCT[1], Blackboard [2] y Moodle[3].

En primer lugar, Blackboard es una línea de productos comercial desarrollada por Blackboard Inc., utilizada en más de 80 países y por más de 3000 instituciones docentes, incluyendo escuelas, facultades y universidades. La línea de productos comprende dos paquetes comerciales, uno destinado al mantenimiento del contenido didáctico, y otro enfocado al procesamiento de transacciones comerciales.

Por su parte, WebCT (Web Course Tools) nació como aplicación de código abierto en la Universidad de British Columbia, pero fue adquirida por BlackBoard Inc. en el año 2006. Actualmente, presenta dos versiones: WebCT Vista, dirigida a empresas, y WebCT Campus Edition dirigida a organizaciones universitarias. Incluye herramientas de gestión de contenidos, de evaluación y de seguimiento y gestión de alumnos.

Por último, Moodle (Modular Object-Oriented Dynamic Learning Environment) fue desarrollado por Martin Dougiamas a finales de 2002. Aunque su uso y descarga es gratuita, por tratarse de una aplicación con Licencia Pública GNU, existe una línea de soporte comercial a través de la empresa "moodle.com", creada en 2003. Esta aplicación se utiliza por más de 71 millones de usuarios a lo largo de 235 países [Moo14]. Su desarrollo sigue las pautas de la teoría del constructivismo social, e incluye funcionalidad para que los alumnos puedan participar en el proceso de construcción del material didáctico (aunque se puede utilizar como un instrumento de distribución de contenido simple). Como suele ocurrir con las aplicaciones de código abierto, la clave principal de éxito proviene de la gran comunidad on-line encargada de su mantenimiento, y del desarrollo de nuevos módulos.

[1] www.WebCT.com

[2] www.blackboard.com

[3] www.moodle.org

2.3.1 Objetos de Aprendizaje y Sistema de Gestión del Aprendizaje

La creciente popularidad del e-learning motiva la aparición de entornos cada vez mas sofisticados y complejos. La modularidad de los sistemas se antoja un esquema de desarrollo indispensable para elaborar herramientas completas, distribuibles y fáciles de mantener. En este sentido, los objetos de aprendizaje representan un método de modularización para desarrollar contenidos y arquitecturas flexibles para este tipo de aplicaciones.

Una de las definiciones más extendidas de *objeto de aprendizaje* la proporciono Wiley en su trabajo [Wil03]:

> "Un objeto digital es cualquier recurso digital que se pueda reutilizar como soporte del proceso de aprendizaje. Esta definición incluye cualquier elemento que se puede adquirir a través de la red bajo demanda, sea grande o pequeño . . . Ejemplos de recursos pequeños incluyen imágenes digitales, *feeds* de datos, fragmentos de vídeo o audio, pequeños trozos de texto, animaciones, y pequeñas aplicaciones web, como una calculadora Java. Ejemplos de recursos grandes incluyen páginas web enteras que combinen texto, imágenes, y otros materiales o aplicaciones que proporcionen experiencias completas, como un evento instructivo completo".

Actualmente, la idea de objeto de aprendizaje se asocia con la última parte de la definición. Así, un objeto de aprendizaje se considera como un módulo de un sistema que trata de facilitar la interoperabilidad, la flexibilidad y la capacidad de reutilización del contenido de aprendizaje digital. La idea subyacente comparte características con el paradigma de diseño Orientado a Objetos. Existe una relación directa entre el nivel de flexibilidad del sistema y el nivel de granularidad gestionado mediante objetos de aprendizaje [RM03]. El concepto de objeto de aprendizaje lidera un cambio en la estrategia empleada para realizar propuestas tecnológicas educativas. Nace así el concepto de *Sistemas de Gestión del Aprendizaje* (SGA).

Si bien estos sistemas pertenecen al ámbito de la educación, muchos de ellos están más enfocados a la administración de cursos y objetos de aprendizaje que al desarrollo de actividades pedagógicas. Así, este tipo de sistemas toman el papel de *frameworks*. El material pedagógico *per se* se incluye dentro de los objetos de aprendizaje, no en la aplicación que las mantiene. Así, un SGA podría considerarse como un sub-tipo de EAV, encargado de la administración del proceso educativo, en lugar de enfocarse en el propio proceso. Suele incluir funcionalidad para el registro de estudiantes, evaluación on-line y administración de recursos educativos [MJ97, APRS03]. Por ejemplo, la plataforma Moodle se enmarca en este tipo de sistemas.

Con frecuencia, los términos EAV y SGA suelen considerarse como sinónimos. En inglés, es bastante común que se use indistintamente "Learning Management System" y "Virtual Learning Environment", dando lugar a bastante confusión. Con el fin de evitar este problema en la presente Tesis, usaremos el término en español "Sistema de Apoyo al Aprendizaje" para referirnos a cualquier sistema, completo o parcial, de gestión o de aplicación, que de soporte al proceso educativo. Sin embargo, en inglés usaremos el término "Virtual Learning Environment" con el mismo fin, por tratarse del término más extendido en la literatura.

2.4 Aprendizaje Colaborativo Mantenido por Computador

Centrado en aspectos puramente didácticos, surge la rama del *Aprendizaje Colaborativo Mantenido por Computador* (ACMC). Se trata de un modelo pedagógico, sostenido por el uso de computadores, que trata de motivar la colaboración entre estudiantes [SKS06]. El aprendizaje tiene lugar mediante la interacción entre los alumnos, evitando la sensación de aislamiento como individuos.

De esta forma, el diseño del ACMC tiene como objetivos la creación de actividades y entornos que fomenten las prácticas en grupo. Así, los estudiantes pueden buscar cooperativamente la solución a un problema dado, implicarse en el propio proceso de construcción del conocimiento y compartir sus reflexiones, tomando así conciencia de los procedimientos y el conocimiento que necesitan para superar tal problema [DCVOE00]. Koschmann [Kos02] describe este área como el

> “campo de estudio centrado en aquellas prácticas significativas presentes en un contexto de actividad conjunta, y en las formas en las que tales prácticas se desarrollan mediante artefactos”.

En la actualidad existen bastantes trabajos que analizan las bondades de entornos de aprendizaje colaborativo [LR10, OJK14, YZ14, MR13].

2.4.1 Sistemas Educativos Adaptativos y Tutores Inteligentes Virtuales

Durante los años 90, los avances obtenidos en el campo de la Inteligencia Computacional impulsaron el desarrollo de un nuevo sub-tipo entre los Sistemas de Apoyo al Aprendizaje, especializados en la adaptación del propio sistema en base a las características del usuario. Estos sistemas se enmarcan principalmente dentro de los campos de investigación conocidos como: los “Sistemas Adaptativos Educativos” y los “Tutores Inteligentes Virtuales” (el lector puede consultar un análisis detallado de las características de ambos campos en el Capítulo IV Sección 1).

Los *Sistemas Adaptativos Educativos* (subtipo de los Sistemas Adaptativos Hipermedia) incluyen a cualquier sistema con tecnología hipermedia e hipertexto que sea capaz de traducir las características del usuario en un modelo de representación interna, y que utiliza tal modelo para adaptar varios rasgos visuales del sistema al usuario [Bru96]. Su principal requerimiento consiste en determinar qué características de usuario se deben usar para adaptar el material en un dominio específico, cómo mostrar tales características, cómo actualizarlas y qué tipo de adaptación se implementará. Por tanto, este tipo de sistemas incluyen los siguientes componentes: el sistema hipermedia, el componente de adaptación y el modelo de usuario [BM07].

Los *Tutores Inteligentes Virtuales* presentan un modelado y características similares a los sistemas anteriores, aunque la adaptación ahora no está restringida explícitamente al contenido. En su lugar, tratan de obtener instrucción personalizada en base al modelo de usuario, notado ahora como modelo de estudiante. Por ejemplo, se persigue adaptar el material de aprendizaje al usuario, proporcionar consejos pedagógicos de forma inteligente como resultado de acciones erróneas, o guiar al alumno durante el desarrollo de un problema. Además, se suele vincular el concepto de Inteligencia Computacional a este tipo de sistemas, ya que se emplean técnicas de esta rama para proporcionar la adaptación deseada.

Como parte de estos campos de aplicación, podemos encontrar varias propuestas en la literatura. En primer lugar comentaremos algunos ejemplos de sistemas que pertenecen únicamente al campo de los Sistemas Adaptativos Educativos. En [BC98], De Bra y Calvi presentan una arquitectura hipermedia de adaptación de contenido y enlaces realizada a partir de fragmentos condicionales, usando el estándar HTML. La arquitectura está diseñada para generar aplicaciones web. Para ello implementan un modelo de filtrado adaptativo a partir del modelo de usuario. Otro trabajo propuesto por Mitsuhara et al. [MOKY02] presenta MITS (*Multi-hyperlink Tailoring System*), una plataforma de aprendizaje adaptativa basada en web. En este entorno, los estudiantes pueden crear y compartir hiper-enlaces, que son adaptados mediante técnicas de filtrado colaborativo para así reducir la sobrecarga de opciones. La plataforma tiene como objetivo encontrar y solucionar los puntos muertos producidos en el proceso de aprendizaje por el exceso de enlaces inadecuados a los intereses del usuario. Ambas propuestas presentan técnicas eficientes de adaptación, pero que no pueden considerarse "inteligentes".

A continuación, examinamos dos ejemplos de propuestas que ofrecen soluciones inteligentes, pero que no ofrecen una adaptación explicita de contenido. Primero, *German Tutor*, presentado en [HN01], es un sistema web que contempla las componentes gramaticales de un curso integral de alemán. El sistema contiene una gramática y un analizador que examina las sentencias escritas por parte de los estudiantes con el fin de detectar errores de tipo gramatical. Como capacidad de adaptación, presenta un sistema modular de mensajes de *feedback* y actividades ajustadas a cada estudiante. En segundo lugar, Mitrovic presenta SQL-Tutor [Mit03], un tutor inteligente basado en web para el aprendizaje del lenguaje de base de datos SQL. El sistema implementa un modelo de restricciones condicionales. Las soluciones propuestas por el estudiante ante problemas de aprendizaje del lenguaje SQL se casan con tales restricciones con el objetivo de proporcionar mensajes de *feedback*.

Por último, examinamos algunos ejemplos de propuestas híbridas que aúnan adaptación de contenido y técnicas inteligentes. Graesser presenta AutoTutor en [GCHO05], un sistema complejo que simula la interacción con un tutor humano, manteniendo conversaciones en lenguaje natural con el usuario. El sistema ayuda al alumno en la construcción de respuestas ante problemas planteados, y se adapta de forma dinámica. Utiliza el Análisis de la Semántica Latente para este fin. AutoTutor funciona bien cuando se aplica en dominios cualitativos y cuando el conocimiento compartido entre el tutor y el alumno es bajo o moderado (en lugar de alto). Weber y Brusilovsky presentan en [WB$^+$01] ELM-ART, un sistema educacional interactivo e inteligente para el soporte del aprendizaje del lenguaje de programación LISP. Este sistema proporciona recursos de aprendizaje on-line a través de una interfaz que recuerda a un libro interactivo y adaptable. Para ello, utiliza una combinación del modelo de superposición y un modelo de estudiante basado en episodios (para más información sobre estos conceptos puede consultar el capítulo (capítulo IV)). Implementa ayuda a la navegación adaptable, cursos secuenciales, diagnostico individualizado de las soluciones de usuario y ayuda para la resolución de problemas. Otro ejemplo de modelo híbrido puede ser observado en [ACC$^+$03]. En este trabajo se propone un sistema que sugiere material educativo al usuario en función de su comportamiento, mediante un sistema de recomendación multi-agente. El sistema es capaz de filtrar eficazmente los recursos pertenecientes al entorno Web, usando para ello los intereses compartidos entre los usuarios y aunando los beneficios del análisis de contenido.

3 Objetivos

El objetivo de esta Tesis es aportar soluciones específicas a los problemas detectados dentro del marco de los sistemas e-learning, desde un punto de vista puramente práctico. Con esto, pretendemos enfatizar el carácter de *aplicabilidad* de este trabajo. Para conseguir este fin, planteamos los siguientes objetivos transversales:

- Diseñar los procesos de: extracción y organización de conocimiento, recuperación de información, visualización del dominio y adaptación del entorno didáctico. Cada proceso ha de estar gestionado mediante un módulo inteligente independiente, favoreciendo así la cohesión entre su diseño lógico y físico, y garantizando el cuidado de los niveles de granularidad y agregación. Posteriormente, cada propuesta ha de ser validada de forma independiente. También se persigue proponer contribuciones a la literatura científica.
- Como objetivo complementario y metodológico, diseñar una plataforma e-learning completa que incorpore los módulos comentados en el objetivo anterior, así como una metodología de aprendizaje que regule el uso de la plataforma, favoreciendo las actividades de desarrollo y evaluación propias de un curso educativo. Posteriormente, se ha de validar la bondad de la plataforma/metodología de aprendizaje, analizando sus efectos mediante casos de uso reales, concretamente en cursos universitarios de ingeniería.

De forma más específica, pretendemos abordar los siguientes objetivos parciales:

- Dotar a la aplicación de la capacidad de captar y organizar de forma inteligente el conocimiento subyacente en el dominio didáctico, a partir de los recursos educativos que lo componen. Para ello, se estudiarán una serie de técnicas de IC relacionadas con los campos Extracción de Información y Aprendizaje de Ontologías.
- Desarrollar técnicas de recuperación de información y de visualización del dominio didáctico, fomentando la exploración y comprensión del contenido de cada área didáctica. Así, examinaremos técnicas de IC relacionados con los campos Recuperación de Información, y Visualización de Información y Conocimiento.
- Proporcionar un esquema de adaptación curricular automática, es decir, adaptación del entorno didáctico y los criterios de evaluación en función de las características individuales de cada alumno. Observaremos para ello los Sistemas de Evaluación Mantenidos por Computador, los Tutores Inteligentes Virtuales y los Sistemas Educativos Adaptativos.
- Diseñar la arquitectura de un sistema e-learning completo que implemente la gestión de las áreas curriculares, contenido didáctico y usuarios, así como procesos pedagógicos que favorezcan el aprendizaje: construcción conjunta del contenido didáctico, mecanismos de evaluación, y mecanismos de interacción. Además, debe incluir los módulos inteligentes comentados anteriormente.
- Desarrollar y validar una metodología didáctica que regule el uso del sistema e-learning como parte de cursos universitarios de ingeniería.

4 Metodología

En esta sección, describimos la metodología de análisis y desarrollo que se ha seguido a lo largo de esta Tesis.

Basaremos nuestro estudio en el análisis de dos etapas fundamentales: (i) discusión, modelado y validación experimental de técnicas de la Inteligencia Computacional que apoyen los procesos de extracción y organización del conocimiento, recuperación de información, representación visual del dominio, y adaptación del entorno didáctico e (ii) integración de dichas técnicas en un sistema completo, que será validado mediante casos de uso reales a través de la metodología de aprendizaje correspondiente.

La primera etapa se centrará en la aplicación de técnicas de IC para el desarrollo de herramientas de soporte de Sistemas de Apoyo al Aprendizaje. Primero, analizaremos técnicas que permitan obtener con la mínima intervención del usuario varias estructuras de meta-datos a partir del contenido didáctico. Cada estructura presentará un nivel de riqueza semántica distinto, comenzando por términos simples hasta considerar estructuras conceptuales. Segundo, exploraremos métodos de recuperación de información y de visualización que faciliten la comprensión del dominio y la navegación a través del contenido didáctico. Tercero, examinaremos métodos de adaptación del entorno didáctico. Consideraremos dos niveles de adaptación: por petición expresa del usuario y por decisión interna del módulo inteligente. Siguiendo un proceso de ingeniería de carácter científico, utilizaremos las siguientes pautas metodológicas para la resolución de cada sub-problema:

- Análisis de las características del problema abordado.
- Revisión del estado del arte.
- Diseño y desarrollo de una propuesta de solución del problema dado.
- Validación experimental de la propuesta en base al problema abordado. Se utilizarán para ello métricas de evaluación estandarizadas y se realizarán, de ser posible, comparaciones contra propuestas significativas del estado del arte.
- Extracción de conclusiones y propuesta de trabajos futuros.

En detalle, cada sub-problema se abordará mediante técnicas independientes de la Inteligencia Computacional:

- *Extracción y organización del conocimiento del contenido didáctico.* Se estudiarán mecanismos relacionados con los campos Minería de Datos y Extracción de Información para obtener estructuras de meta-datos a partir del contenido didáctico. Abordaremos la tarea de forma incremental, considerando estructuras de cada vez mayor riqueza semántica.

  Extracción automática de términos clave: consiste en la detección automática de los términos más importantes de una fuente de recursos. El proceso comienza con la detección de los términos significativos, y continúa con un proceso de filtrado o clasificación basado en su importancia respecto al dominio. Una vez obtenidos, cada recurso del sistema se puede clasificar mediante sus términos clave en una estructura de indexación sencilla. El problema se puede abordar mediante esquemas estadísticos, clasificadores ML (de sus siglas en inglés *Machine Learning* o modelos basados en diccionarios de términos de dominio.

  Extracción y recomendación de etiquetas: su finalidad es la de recomendar a cada usuario en entornos colaborativos una serie de etiquetas textuales a partir de un recurso. Las etiquetas deben ser validadas por el usuario. El conjunto de etiquetas seleccionado por todos los usuarios de la comunidad se emplean para clasificar los recursos a través de una estructura taxonómica, conocida como folksonomía. Existen tres tipos de estrategias para su resolución: colaborativas, basadas en el contenido y basadas en ontologías.

Aprendizaje de ontologías: comprende los mecanismos de extracción estructura ontológica a partir de una fuente de recursos. Comprende la detección automática de conceptos y relaciones entre conceptos. Existen cuatro técnicas fundamentales para su resolución: técnicas estadísticas, técnicas aprendizaje supervisado, métodos de procesamiento del lenguaje natural y uso de diccionarios de dominio.

- *Recuperación de información y representación visual y navegable del dominio.* Se estudiarán mecanismos relacionados con los campos Recuperación de Información, y Visualización de Información y Conocimiento a partir del dominio didáctico. En concreto, examinaremos los siguientes mecanismos:

  Recuperación de información: su finalidad es la de recuperar de aquellos fragmentos de información que guardan mayor relación con consultas realizadas por usuarios, expresadas en el lenguaje natural. En la literatura, este campo es comúnmente conocido como *Question Answering*. Su campo de aplicación más generalizado es el de los documentos FAQ (de sus siglas en ingles *Frequently Asked Questions*). Para su resolución, se pueden identificar dos tipos de estrategias: métodos basados en conocimiento y métodos estadísticos.

  Nubes de etiquetas: método de visualización que dispone las etiquetas más populares de una comunidad virtual en una representación de nube. El problema correspondiente a la elección de las etiquetas a visualizar se conoce como “selección de etiquetas”. Para su resolución se utilizan estrategias estadísticas.

  Mapas conceptuales: técnica usada para organizar y representar el conocimiento de un dominio como una red de conceptos. En la red, los nodos representan los conceptos, y los enlaces representan las relaciones entre los conceptos. Como método de visualización simple, no presenta mayor dificultad que la disposición de los elementos en un entorno visual. Es la etapa de captación de conocimiento previa a la visualización la que suscita mayor interés en la literatura. Existen cuatro técnicas principales para su resolución: técnicas estadísticas, técnicas aprendizaje supervisado, métodos de procesamiento del lenguaje natural y uso de diccionarios de dominio.

- *Adaptación del Entorno.* Investigaremos los métodos de adaptación automática del entorno de aprendizaje, mediante dos técnicas especificas del campo de aplicación e-learning.

  Generación automática de tests: se refiere a los métodos de generación de tests en base a criterios de evaluación seleccionados por el usuario. Dicha tarea se suele identificar dentro del campo conocido como “Tests Adaptables mantenidos por Computador”. Se suelen emplear estrategias basadas en plantillas y estrategias estadísticas para su resolución.

  Tutor inteligente virtual: área específica del e-learning. Comprende el desarrollo de sistemas educativos que simulen la labor del profesor a la hora de proveer al alumno con contenidos didácticos, estrategias de aprendizaje y criterios de evaluación seleccionados en función de sus necesidades. La dificultad de la tarea recae en el proceso de identificación de las características del estudiante, conocido como modelado del estudiante. Como métodos de resolución, podemos encontrar técnicas basadas en: el modelo de revestimiento, el modelo de estereotipos, el modelo de perturbación, el aprendizaje supervisado, teorías cognitivas, lógica difusa, redes bayesianas y ontologías.

En la segunda etapa de la metodología, se contempla el desarrollo de un sistema e-learning completo que recoja los mecanismos inteligentes desarrollados y permita una evaluación global de la propuesta. El diseño de este tipo de sistemas está lejos de ser trivial. Su éxito depende de ciertos factores que afectan a la consecución de sus objetivos: desde las características intrínsecas cada estudiante, pasando por el grado de interacción entre estudiantes o la tecnología utilizada. Estos factores evolucionan a la par de que las propias TIC, alterando la percepción de idoneidad de algunos sistemas frente a otros. Por este motivo, esta etapa tiene por finalidad identificar y analizar los principales factores que afectan al potencial de las plataformas e-learning, para así estar en disposición de desarrollar una herramienta completa.

En primer lugar, estudiaremos los indicadores cognitivos que se suelen emplear como medidas de efectividad para los sistemas tecnológicos educativos ("nivel de satisfacción" y "nivel de éxito"). A continuación, examinaremos en qué medida afectan ciertos factores pedagógicos a dichos indicadores. Distinguiremos entre factores propios de metodología (auto-regulación del estudiante, interacción estudiante-contenido, evaluación formativa/sumativa, enseñanza semipresencial) y factores propios de aplicación (interacción estudiante-profesor, interacción estudiante-contenido, auto-evaluación, y mecanismos de ayuda pedagógica). Tras ello, presentaremos el diseño de la plataforma, y su metodología de aprendizaje reguladora. Posteriormente, validaremos la propuesta mediante casos de uso reales. La metodología particular de esta fase seguirá las siguientes pautas:

- Discusión de los indicadores de efectividad de plataformas e-learning.
- Discusión de los factores pedagógicos que afectan a los indicadores de efectividad.
- Propuesta de arquitectura y funcionalidad del sistema.
- Propuesta de metodología de aprendizaje.
- Validación de la propuesta mediante casos de uso reales correspondientes a cursos universitarios de ingeniería.

Chapter I

PhD Dissertation and Review of the State of the Art

1 Introduction

From the last decade of the 20th century, the massive growth of the Information and Communication Technologies (ICT in the following) has lead to an evolution of the main foundations of our society (we currently live in the "information age"). In this way, the base of our social behaviour is derived from the sharing and acquisition of information through learning. For centuries, the knowledge was transferred from one generation to other, mainly supported by schools and universities. Notwithstanding, the boost suffered by the Internet has provide a new learning context. In these days the information is always available in global scale. This has provoked a deep change on the classical paradigms of knowledge sharing, i.e. education and learning. The classic face-to-face model coexists with on-line models, which solve the physical and temporal restrictions of the participants. The new learning scheme supported by the ICTs is called *e-learning*. Its main goals consists of facilitating an interactive and flexible (in time and space) learning through virtual environments, that is centred on the student. The absence of restrictions has changed the role of the student, being now the principal actor of the learning process. In consequence, the role of the teacher is also modified. Today, he/she must stimulate students to take an active role in their learning activities. Summarizing, the classic teacher-centered paradigm gives rise to the student-centered learning.

In the context of global economy, the job market suffers from a fierce competence among the workers. They need an extraordinary preparation that allows them to surpass the rest of competitors. Thus, in order to provide the better competences, the teaching organizations (including the corporate training) are intensely involved in the development of e-learning methodologies, mostly based on web environments. The sector of education has become one of the most important markets regarding corporate and private investments. Without going any further, the European Commission has worked on the promotion, improvement, and implementation of the e-learning during the last 10 years [CotEC01]. Thus, the European Higher Education Area, developed under the Bologna Process, pursues along other measures the establishment of student-centered learning with the aim of fostering a better degree of students' implication and independence [EC09, Hei05, Lou01].

The increasing economic and social relevance of e-learning is provoking the development of new research lines centred in the design of functional learning support environments. Today, there exist a number of applications focused on improving the learning processes. For example, it is common to find virtual environments promoting the collaboration among students, following the Web 2.0 scheme. What is more, there is an increasing development of intelligent modules able to adapt the learning environment in function of the user's characteristics. This kind of application provides students with freedom for carrying out their proper learning plans, for constructing and sharing knowledge, for searching complementary material, ... Unfortunately, most of current e-learning system which are actually used in real scenarios are focused on the managements of courses and digitalized content [DK12]. Systems of this kind may suffer from a number of problems derived from the information overload. That is, the presence of large, complex and ill-structured content might lead to student's cognitive overload and disorientation problems due to the difficult of acquiring the domain knowledge [BC94]. Therefore, it would be arguably desirable to count with cognitive tools in order to overcome the above limitations [KL97]. In this sense, the development of tools able to organize and visualize the learning content in an automatic and intelligent manner could be very helpful for the teacher, or the manager of the system. Concluding, the development of knowledge exploitation tools from the educational content may lead to a better support of the learning process. Particularly, we detect three scopes of application:

- Extraction and organization of the knowledge from the educational content: it is referred to the extraction, storage, and organization processes of the knowledge underlying from the educational content. The knowledge should be organized in meta-data structures, since they are reusable, extendible and shareable. More concretely, we consider the extraction of the most representative elements of the domain, taking into account different levels of semantic richness. Starting from simple key terms, we will handle the acquisition of more complex knowledge.

- Information retrieval and visualization of the educational domain: it involves two exploitation processes of the domain knowledge. On the one hand, we consider the retrieving of learning resources in function of user selected criteria (e.g. natural language queries). Thus, both students and teachers could access to the learning resources in a efficient and effective way. On the other hand, we consider the visual and navigable representation of the domain. More concretely, we pay attention to knowledge and information visualization methods. Hence, students would be able to effectively acquire the conceptual scheme of the given learning subject. Moreover, this scheme will aid the navigation through the learning resources in the system.

- Intelligent adaptation of the learning environment: it involves the adaptation of the virtual environment in basis of the individual users' characteristics. In this work, we consider the two following scopes. First, we consider the automatic generation of assessment resources in function of objectives configured by the user. Then the user will be able to set his/her individual learning path and goals, and subsequently receive adjusted assessment resources. Second, we take into account an adaptive method of the student's curriculum able to modify the learning goals and environment in function of the student's knowledge state. Now the system is responsible for automatically inferring the current learning objectives in basis of the student's level of knowledge about the learning subject, not only for providing single assessment elements, but also for selecting more complex assessment strategies.

Summarizing, the integration of each individual solution into a complete system may lead to improved learning process for both teachers and students.

This PhD dissertation is aimed for proposing valuable technological-scientific contributions to the state of the art in form of technological solutions to the problems related with knowledge extraction and organization, information retrieval and visual representation of the domain, and adaptation of the environment in e-learning systems. Our *methodology* consists of analysing each task individually in order to design individual solutions. To that end, we will focus on Computational Intelligence (CI) techniques. Our *goal* is two-fold: (i) to find effective technological solutions to the commented tasks, and (ii) to integrate each individual solution into a complete e-learning system, as added value.

We will start with a study of the knowledge extraction and organization methods considering the e-learning scenario. We will attend to three different levels of semantic richness: (i) key terms, (ii) conceptual tags, and (iii) concepts and relations. Most of related techniques do not consider the specific requirements of e-learning. In addition, this study will provide us with a solid base for developing methods able to handle the rest of detected tasks.

We will continue with a discussion on information retrieval methods, as well as visualization methods for the educational domain. We will analyse firstly the information retrieval systems based on users' queries. Although this kind of tools are commonly considered in the scope of information access systems, that is not the case in the e-learning field. Subsequently, we will afford the investigation of two visualization techniques: concept maps and tag clouds. Concept maps are usually employed in the learning field due to their ability of representing the concepts and relations of the domain. Tag clouds are receiving increasing interest in virtual communities, as exploration and navigational tool.

After that, we will focus on the analysis of intelligent mechanisms for the adaptation of the educational environment. We will firstly consider the generation of resources in function of user-selected preferences. Concretely, we will propose an automatic assessment tests generation method. The field related with Computer Adaptive Tests is broadly spread, and has the aim of providing dynamic assessment instruments as part of the learning process. Then, we will analyse the intelligent adaptation of the environment in basis of user's characteristics. In detail, we will develop a method for adapting the curriculum of the student sequentially, in function of his/her level of knowledge in regard to the current learning goals. This task is framed on the fields of Adaptive Educational Systems and Intelligent Tutoring Systems, very popular today.

We will conclude this research by integrating all the previously commented methods into a complete e-learning system. To that end, we will design the architecture of the platform, as well as a regulatory methodology of usage as part of a higher education course. Therefore, we would not only provide added value to this investigation, but also we would establish a validation framework. To that end, we will analyse and validate the application of the system as part of a higher education engineering course during two consecutive years.

Providing an exhaustive analysis of all technologies and approaches applicable to e-learning field from a theoretical point of view falls beyond the scope of this Thesis. We are rather interested in proposing a concrete solution to the field in form of a computer application. Therefore, this PhD Thesis results from an engineering process of scientific nature. With this, we intend to reinforce the applicability nuance of this work. Most methods and theoretical advances presented in this dissertation have been published in different scientific journals or conferences.

This Thesis project has been developed as part of the official PhD Master degree on *Soft Computing and Intelligent Systems* under the PhD official program *Tecnología de la información y la Comunicación* of the Department of Computer Sciences and Artificial Intelligence of the University of Granada, under the tutelage of Professor Juan Luis Castro Peña. This four-years project has been funded by the FPDI (*Formaci'on de Personal Investigador correspondiente a los Proyectos de Investigaci'on de Excelencia*) scholarship of the *Junta de Andalucía*, and has been developed in the CITIC-UGR (*Centro de Investigación en Tecnologías de la Información y las Comunicaciones*).

We will devote the rest of this chapter to review the state of the art of e-learning field, and to expose our objectives and methodology. The rest of this dissertation is organized as follows. Chapter II is dedicated to those information extraction techniques employed to handle the acquisition of the knowledge from the educational content. Next, chapter III attends to the information retrieval, and knowledge and information visualization techniques used for representing the domain of the educational content. Chapter IV depicts the mechanism involved on the adaptation of the educational environment. After that, we will present the architecture of a complete e-learning system, as well as a usage regulatory methodology, and the validation of the proposal in Chapter V. Finally, main conclusions and future works will be presented in chapter VI.

2 State of the Art

In this section, we present a review of the state-of-the-art in the e-learning field. Although this work is focused on the technological viewpoint, it is necessary to start with a discussion of the pedagogical principles which regulate the learning process. Subsequently, we will examine the historical development of e-learning. Besides we will introduce some basic concepts employed in this work.

2.1 Theoretical principles and pedagogical concepts

The inner particularities of the learning process give our problem of a interdisciplinary nature. Therefore, in this subsection we provide an overview of the most common pedagogical theories. Nevertheless, providing an exhaustive study of the field is not objective of this Thesis, because of its complexity and extension. We refer the reader to the study in [Sch12] for further information.

2.1.1 Main Pedagogical Principles

In first place, we comment the features of the main pedagogical paradigms developed during the 20th century. The complexity of any learning process has fostered the design of some theoretical models composed by methodological rules. Precisely, it is necessary to analyse the possible learning models before facing the development of technological-educational approaches. Those strategies compound the logical design of the system. The research community in the field of pedagogy distinguish between three different trends: behaviourism, cognitivism, and constructivism [EN93, Sch12].

The *behaviourism* is a psychology trend initiated by John B. Watson (1878-1958) that stands up for the use of experimental procedures in order to study the observable behaviour. It handles the learning laws by means of the stimulus-response conditioning. Its theoretical foundation relies on the premise which considers that a stimuli produces a direct response, being this the result of the interaction between the organism receiving the stimuli and the environment in which it is produced [Sch12]. Let suppose that teacher writes into a blackboard the equation "2 + 3 = ?". This equation is the stimuli, that produces the direct response "6" of the student. Therefore, the learning is considered as a mechanic process destined to stimulate a change in the behaviour of students, using to that end positive and negative reinforcements. It does not take into account the involved mental processes (the human mind is considered as a black box), and do censor an active participation of the student.

The *cognitivism* was born from the cognitive science models during the 1950's decade. The idea behind this is centred in the promotion of more complex cognitive processes than the simple observable behaviour (behaviourism), such as the resolution of problems, the language, the conceptual formation, and the information processing [Sne83]. The cognitive theories are focused on the conceptualization of the learning process: how the information is received, organized, stored and retrieved in the human mind [EN93]. The knowledge acquisition process is divided into three stages [Sch12]. First, the information is perceived through the human senses. Then, it is stored in the short-term memory. Lastly, the information is transferred to the long-term memory. Meanwhile, the knowledge is considered as a internal scheme of mental states. This scheme is modified at the same time that the individual learns. During the process of learning, the new information is compared against the current scheme, which could be modified or extended in function of the new acquired knowledge.

The last pedagogical model presented here is the *constructivism.* This theory compares the learning with the creation of meaning from the experience [BCDP92]. As it is held by the cognitivism model, the constructivism considers the learning as a mental activity, although there exists some differences between both models. The cognitivism paradigm perceives the human mind as a reference tool for the real world. In contrast, the constructivism paradigm affirms that the human mind filters information perceived from the world, in order to develop a proper and individualized reality [Jon92]. Thus, the information flow is not realised from the real world to the human memory. On the contrary, the human memory constructs a subjective interpretation of the real world, based on the individual's experiences and interactions. Therefore, the internal representation of the knowledge is continuously changing.

As it could be observed, each theory presents different implications and limitations, in function of the type of knowledge to be acquired and the learning goals. In Chapter V we provide additional details about how to include the constructivism paradigm as part of the design of e-learning mechanisms.

2.1.2 Educational Objectives

In 1956, Bloom and Krathwohl published the so-called Taxonomy of Educational Objectives [BK56]. Several studies related with the design of e-learning models use the taxonomy as design basis. We present its main features below.

The taxonomy identifies three different domains: cognitive, affective and psychomotor. Each domain compresses a set of learning behaviours organized in a hierarchy, which favour the acquisition of a new ability, knowledge and/or attitude.

The cognitive level handles the brain functioning, and the cognitive processes related with learning, such as the comprehension or the knowledge construction. It includes the recognition of concrete facts, procedural patterns, and the concepts involved in the development of intellectual abilities and skills. There are six main categories associated to the cognitive process, which are detailed in Figure i.1.

The affective level describes the emotional features of learning, such as student's motivation, engagement or behaviour. There are five categories, depicted in Figure i.2.

Finally, the psychomotor level is centred on the handling process regarding tools and instruments. This domain falls under the scope of e-learning systems. The reader can find additional information in [Sim70].

2.1.3 Other concepts

In the following, we define the rest of pedagogical concepts that take part in this Thesis.

- *Curriculum adaptation*: decisions relative to the organization of learning resources for the analysis of the different students. It comprehends the adaptation of educational objectives, the learning methodology, and the assessment criteria, as well as the modification of the content.

- *Student self-regulation* o *self-regulated learning*: degree in which the student is able to focus systematically their thoughts, feelings and actions in achieving their educational goals, participating actively in the learning process [Zim89, Zim86].

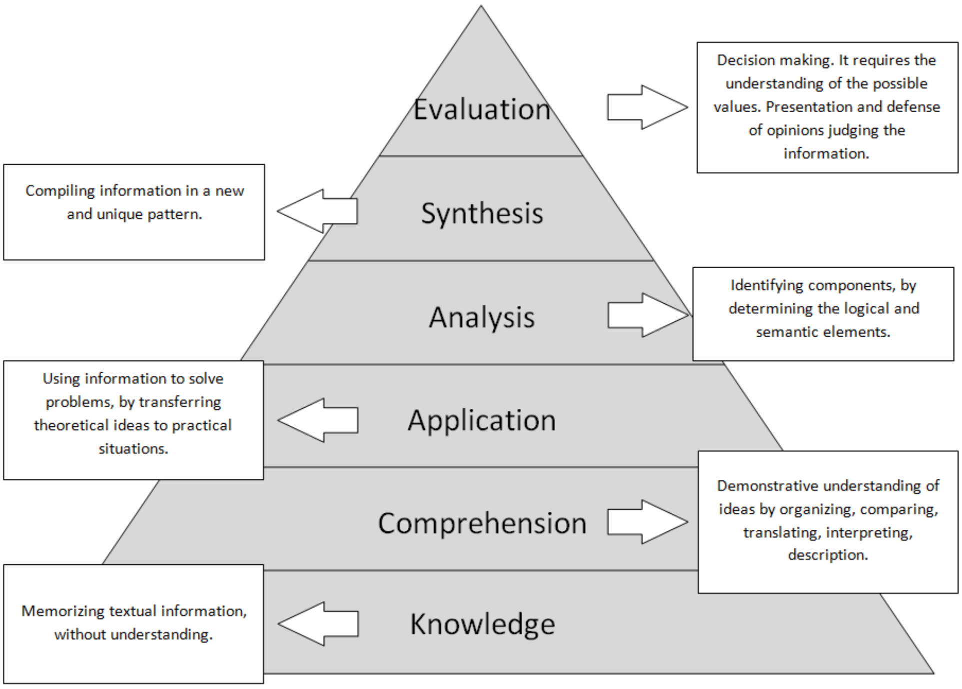

Evaluation
Decision making. It requires the understanding of the possible values. Presentation and defense of opinions judging the information.
Synthesis
Compiling information in a new and unique pattern.
Analysis
Identifying components, by determining the logical and semantic elements.
Application
Using information to solve problems, by transferring theoretical ideas to practical situations.
Comprehension
Demonstrative understanding of ideas by organizing, comparing, translating, interpreting, description.
Knowledge
Memorizing textual information, without understanding.

Figure i.1: Bloom's taxonomy of educational objectives: cognitive domain

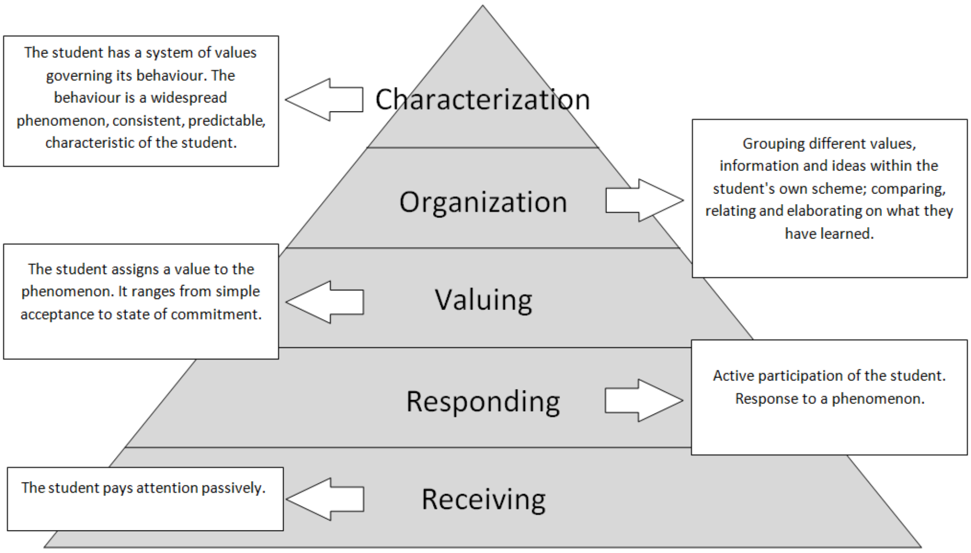

Characterization
The student has a system of values governing its behaviour. The behaviour is a widespread phenomenon, consistent, predictable, characteristic of the student.
Organization
Grouping different values, information and ideas within the student's own scheme; comparing, relating and elaborating on what they have learned.
Valuing
The student assigns a value to the phenomenon. It ranges from simple acceptance to state of commitment.
Responding
Active participation of the student. Response to a phenomenon.
Receiving
The student pays attention passively.

Figure i.2: Bloom's taxonomy of educational objectives: affective domain

- *Student self-assessment*: assessment model where the student applies to himself/herself a tool for measuring his/her level of acquired knowledge. The purpose of this is to identify the failures of learning in order to correct them, by establishing settlement activities.
- *Level of student satisfaction*: it is referred to the (positive/negative) perception of the student about the learning process, motived by a given learning strategy.
- *Level of student achievement*: it is referred to the quality of the knowledge acquired by the students after the learning process carried out with a given learning strategy.

- *Formative assessment*: systematic and continuous assessment activity, which aims to provide the necessary information about the educational process in order to adjust its objectives; to critically review the plans, programs, methods and resources; to orient the students; and to feed back the proper learning process. The formative assessment is more focused of the learning processes than in the results of them.
- *Summative assessment*: assessment activity at the end of each learning process, with the aim of determining the degree of achievement of the learning objectives by students. This level determines the relative position of each student in the group and places him/her at certain levels of efficacy, usually marked (and normatively established) by a known scale of ratings.

2.2 Technology and Education: Firsts Steps

Although the concept of e-learning is currently understood as education via the Internet, the first models combining technology and education come from the second half of the 20th century.

The first steps carried out in the design of learning technologies took place influenced by the military industry. During the Second World War, the inner characteristics of the conflict fostered the development of efficient methods of military instruction. Instructional films were used as training method for the weapon systems. Also mechanical devices of military training, called “phase check”, were developed. These devices addressed any training process (e.g., assembly or disassembly of any component) as a structured sequence of steps. Thus, the foundations of instructional design were established. Indeed, it was Skinner, a professor at Harvard University, who introduced the concept of *programmed instruction*, developing the so-called *teaching machine* [Ski60]. Programmed instruction consisted of a modular and sequential division of an educational task by means of frames (small modules in which training materials are decomposed). Therefore, the development of the ability was performed incrementally by a continuous polling process which enhanced the correct answers [OHC$^+$12], following the behaviourism paradigm.

During the 1960’s decade, the concept of programmed instruction evolved towards the *computer-based learning* or *computer-assisted instruction*, which refer to the learning processes held by a computer as key component. The mechanical devices are replaced by modern computer-supported strategies. One of the first computer-based systems was PLATO (*Programmed Logic for Automated Teaching Operations*) [BBL62], created by Don Blitzer in the University of Illinois in 1960. The system is considered as the direct forerunner of the current e-learning environments. It included high resolution graphics terminals and implemented the TUTOR programming language, allowing to create interactive learning modules for both students and teachers and establishing mechanisms for communication between users through electronic notes. A few years later, Patrick Suppes and Richard C. Atkinson, professors of Stanford University of Psychology, began to use a computer-assisted instructional system [Sup66] for teaching mathematics and reading to children in elementary schools in Palo Alta, in 1966. The computer-assisted instruction begins to include complexer learning strategies, such as the use of assessment questionnaires, games, or student monitoring [JJS85].

The evolution of the desktop computers during the 1980’s decade fostered the use of technology by the learning organizations, introducing pedagogical applications at classroom-level. However, the final leap occurred with the creation of the *World Wide Web*. Thanks to the presence of distributed learning environments and digital content, the technology enhanced learning acquired a global scale. Then the term *web-based learning* replace the previously considered concepts. It is referred to those scenarios in which the interaction between teacher and student is done via web, leading to the *virtual learning environments*.

2.3 Virtual Learning Environments

The rise of the Internet and web-based technologies provided a perfect setting for the development of dedicated virtual environments. A Virtual Learning Environment (VLE) is a platform designed to manage the processes of learning by means of learning resources. In this work, a *learning resource*, or *educational resource*, is referred to any learning element supported by technology which can be employed or referenced during the learning process. Therefore, the VLEs establish a information sharing framework where the teachers are able to design learning activities, both individual and collaborative. The activities are structured into courses and they shall contain resources enough for helping students to reach their learning goals [DSS$^+$02, Wel07]. Possibly the most used VLEs are WebCT[1], Blackboard [2], and Moodle[3].

First, Blackboard is a commercial suite developed by Blackboard Inc. It is used by more than 80 countries and more than 3000 educational organizations, including schools, faculties and universities. The commercial suite is composed by two different packages. The first is destined to the management of the educational content, meanwhile the second is focused on the processing of commercial transactions.

Second, WebCT (Web Course Tools) was born as an open source application in the University of British Columbia, but it was bought by BlackBoard Inc. in 2006. Currently, it presents two versions: WebCT Vista, for companies, and WebCT Campus Edition focused on higher education organizations. It includes tools for content management, assessment, monitoring and student management.

Lastly, Moodle (Modular Object-Oriented Dynamic Learning Environment) was developed by Martin Dougiamas at the end of 2002. Although its use and download are free (GNU Public License), there exists a commercial support line managed by the “moodle.com” company, founded in 2003. The Moodle application is being used by more than 71 millions of users along 235 countries [Moo14]. The design of Moodle follows the social constructivism paradigm: the students can participate in the construction of the educational content. As it is usual with open source platforms, the key of its success comes from the large on-line community responsible for the maintenance and development of new modules.

2.3.1 Learning Objects and Learning Management Systems

The growing popularity of e-learning encourages the development of increasingly sophisticated and complexer environments. The modularity of a learning system is essential for developing comprehensive, distributable, and easy-to-maintain tools. In this sense, the modularization of e-learning systems is usually carried out by means of learning objects.

One of the most common definitions of *learning object* was provided by Wiley in his work [Wil03]:

> “A learning object is any digital resource that can be reused to support learning. This definition includes anything that can be delivered across the network on demand, be it large or small. Examples of smaller reusable digital resources include digital images or photos, live data feeds, live or pre-recorded video or audio snippets, small bits of text, animations, and smaller web-delivered applications, like a Java calculator. Examples of larger reusable digital resources include entire web pages that combine text, images

[1] www.WebCT.com

[2] www.blackboard.com

[3] www.moodle.org

> and other media or applications to deliver complete experiences, such as a complete instructional event".

The learning objects are usually associated with modules of a system which try to facilitate the interoperability, the flexibility and the reuse of the educational content. The underlying idea shares some features with the Object Oriented design paradigm. There exists a direct relation between the level of flexibility of the system and the level of granularity managed by means of learning objects [RM03]. The concept of learning object leads a change in the design procedure of educational systems. The concept of *Learning Management System* (LMS) is established.

Although this kind of systems belongs to the educational scope, most of them are more focused on the managements of courses and learning objects than in the development of pedagogical activities. Therefore, they are viewed as frameworks. The pedagogical material *per se* is included inside the learning objects, not in the LMS. Therefore, a LMS could be considered as a sub-type of the VLEs focused on the management facet of the learning process, rather than in the pedagogical perspective of it. It usually includes functionality for student record, on-line assessment, and management of learning resources [MJ97, APRS03]. For example, the Moodle platform is framed in this kind of applications.

Commonly, the terms VLE and LMS are understood as synonyms, provoking this confusion. In this work, we will use the term "Virtual Learning Environment" for referring any management or application system, complete or partial, which supports the learning process.

2.4 Computer-Supported Collaborative Learning

The field of *Computer-Supported Collaborative Learning* (CSCL) is centred in pure pedagogical facets. It is a computer-supported pedagogical model which tries to promote the collaboration between students [SKS06]. The learning process takes place by means of interaction, trying to avoid the perception of isolation. Therefore, the goal of ACMC is to provide activities and environments fostering cooperative practices. In this kind of environments, the students are able to cooperate between them in order to give a solution to a given problem, to imply in the construction of the knowledge, and to share their reflections. Thus, they become aware of the procedures and knowledge they need to overcome such problem [DCVOE00]. Koschmann [Kos02] defined this area as the

> "field of study centrally concerned with meaning and the practices of meaning-making in the context of joint activity and the ways in which these practices are mediated through designed artefacts".

There exist several works analysing the benefits of collaborative environments for learning [LR10, OJK14, YZ14, MR13].

2.4.1 Adaptive Educational Systems and Intelligent Tutoring Systems

During the 1990's decade, the advances of the Computational Intelligence promoted the development of a new kind of VLE, specialised in the adaptation of the proper environment in base of the student's characteristics. These systems are framed in two research fields: Adaptive Educational Systems and Intelligent Tutoring Systems (we refer the reader to Chapter IV Section 1 for further details).

The *Adaptive Educational Systems* (subtype of Adaptive Hypermedia Systems) include any system with hypertext and hypermedia technology that is able to translate the user's characteristics to a model of intern representation, and that use such model for adapting visual features of the system [Bru96]. The main requirement of an adaptive hypermedia system consists of determining which user's characteristics should be used in order to adapt the material in a specific domain, how to show such characteristics, how to keep them updated, and which kind of adaptation should be implemented. Therefore, an adaptive educational system contains the following components: the hypermedia system, the adaptation component, and the user model [BM07].

The *Intelligent Tutoring Systems* present a similar modelling and features in regard to the previously commented systems. However, the adaptation is not constrained to the content. This kind of systems tries to obtain personalized instruction in base of the user model, that is noted now as the student model. For example, a system of this type could adapt the learning material in function of the student, provide pedagogical feedback as result of incorrect actions, or guide the student during the performance of a problem. In addition, the concept of Computational Intelligence is usually linked to intelligent tutoring systems. This is due to the adaptation is usually supported by means of intelligent techniques.

Several works have been proposed in the literature as part of this two research fields. Firtsly, we comment some examples of pure adaptive educational systems. First, De Bra and Calvi presented a hypermedia architecture for the adaptation of the content and links, using conditional fragments to that end. The architecture has the goal of generating web applications. Authors implemented a adaptive filtering model based on the user model. Mitsuhara et al. [MOKY02] presented MITS (*Multi-hyperlink Tailoring System*), an adaptive web-based learning platform. The platform allows students to create and share hyperlinks, which are subsequently adapted by means of a collaborative filtering approach. This scheme is able to reduce the option overload. The idea behind MITS is to give solution to the deadlocks produced by the excess of inadequate links in the learning process. Although this two proposals present efficient adaptive techniques, they cannot be considered as "intelligent".

Next, we examine two examples of systems which offer intelligent solutions, but not provide a explicit adaptation of the content. First, the German Tutor is a web-based system for learning the German language [HN01]. The system contains a grammar and a parser that examine the written sentences of the student in order to detect grammatical errors. As adaptive feature, the system includes a modular subsystem of feedback messages, and activities adjusted to each student. Second, Mitrovic [Mit03] presented SQL-Tutor, a web-based intelligent tutor designed to support the learning of SQL database language. The system implements a model of conditional restrictions. The solution proposed by the students confronting SQL learning problems are matched with the restrictions. Then, the system is able to provide feedback messages.

Finally, we discuss some examples of hybrid approaches joining content adaptation and intelligent techniques. First, Graesser presented AutoTutor in [GCHO05], a complex system which simulates the student interaction with a human tutor. The system is able to keep conversations with the student in natural language. The system helps the student in the construction of responses to given problems and it is dynamically adapted. It uses the Latent Semantic Analysis technique to that end. The results regarding the validation of AutoTutor showed that the system works well in qualitative domains, and when the knowledge shared by the tutor and the student is low. Next, Weber and Brusilovsky developed ELM-ART, an interactive and intelligent educational system for supporting the learning of LISP programming language [WB$^+$01]. The system provides students with learning resources via on-line through an interface similar to a interactive book. To that end, it combines a overlay model and a student model based on episodes (we refer the reader to Chapter IV Section 3 for further details). It implements adaptive navigation-aids, sequential courses, individualized student diagnosis, and problem-solving help. Another example of hybrid model can be found in

[ACC^{+}03]. In that work, a system able to suggest educative material to the student in function of his/her behaviour is proposed. The proposal implements a multi-agent recommendation system. It can filter the resources of the web environment in an efficient way by using the interests shared between the users, and analysing the content.

3 Objectives

The main objective of this Thesis is to provide a specific solution to the detected problems in the field of e-learning, from a merely practical point of view. With this, we intend to reinforce the applicability nuance of this work. To this end, fix the following transversal objectives:

- To design the processes related to the extraction and organization of the knowledge, information retrieval, visualization of the domain, and adaptation of the learning environment. Each process will be managed by a independent intelligent module, favouring this to the cohesion between the logical and physical design and guaranteeing the care of the levels of granularity and aggregation. Subsequently, each proposal will be validated in an independent manner. Finally, proposing valuable contributions to the state of the art through scientific publications is also pursued.
- To design a complete e-learning system including the modules developed under the prior objective, and a methodology with the aim of regulating the use of the platform as part of an educative course. Then, the effectiveness of the system/methodology will be analysed by testing the effects of them on real cases of use, concretely on higher education engineering courses.

More concretely, we pretend to fulfil with the following partial goals:

- To supply the system with the ability of extracting and organizing the knowledge of the learning domain from the learning resources. To that end, we will study a set of CI techniques related to the Information Extraction and Ontology Learning fields.
- To develop information retrieval and visualization techniques with the aim of fostering the exploration and understand of the content of a given learning subject. Thus, we will examine the IC techniques related with the fields of Information Retrieval, and Information and Knowledge Visualization.
- To provide with an automatic curriculum adaptation scheme, that is, adaptation of the learning environment and the assessment criteria in function of the individual student's characteristics. We will focus on the CI fields related to the Computer-based Test, Intelligent Tutoring Systems, and Adaptive Educational systems.
- To design the architecture of a complete e-learning system including the management of learning subjects, educative content, and users, as well as pure pedagogical processes for supporting the learning: collaborative construction of the educational content, assessment mechanisms, and interaction mechanisms. In addition, it should integrate the previously commented modules.
- To develop and validate a learning methodology that regulates the use of the system as part of a real engineering course.

4 Methodology

This section is to present the methodology we have followed in this Thesis.

We will base our analysis on two main stages: (i) discussion, modelling and experimental validation of the CI techniques involved in the processes of extraction and organization of the knowledge, information retrieval, visual representation of the domain, and adaptation of the learning environment; and (ii) integration of such techniques into a complete system, which will be validated through real cases of use by means of the corresponding learning methodology.

The first stage is concerned with the application of CI techniques for developing support tools for VLEs. First, we will analyse those techniques able to extract meta-data structures from the educational content with minimum human intervention. Each structure will present a different level of semantic richness. Second, we will exploit information retrieval and visualization techniques with the aim of fostering the comprehension and exploration of the educational content. Third, we will examine methods of adaptation of the learning environment. We consider two levels of adaptation: by explicit user request, and by automatic decision of the module. Following a engineering process of scientific nature, we will follow the next guidelines in order to solve each sub-problem:

- Analysis of the main particularities of each problem.
- Review of the state-of-the-art.
- Design and development of a proposal able to solve the given problem.
- Experimental validation of the proposal. We will employ standardized evaluation metrics, as well as suited comparisons against significant methods in the literature (if it is possible).
- Conclusions drawing and proposal for future research.

In detail, each sub-problem will be addressed by independent CI techniques:

- *Extraction and organization of the knowledge from the educational content.* We will study mechanisms related with the Data Mining and Information Extraction fields in order to obtain meta-data structures from the educational content. The task will be addressed incrementally, considering different levels of semantic richness.

  Automatic key term extraction: is referred to the automatic detection of the most important terms from a resource collection. The process starts with the detection of the significant terms. Then the terms are filtered or classified in function of their importance in regard to the domain. The set of extracted terms could be subsequently used for establishing a classification of the resources. This field comprises statistical schemes, Machine Learning classifiers, or dictionary-based models.

  Tag extraction and recommendation: is referred to the process of recommendation of tags in collaborative spaces. Each tag is extracted and recommended for single resource, and is should be validated by the user. The set of tags selected by all users of the community are used to classify resources through a taxonomic structure, known as folksonomy. There are three types of approaches: collaborative, content-based and ontology-based.

  Ontology learning: is referred to the process of extraction of ontological structures from resource collections. It comprehends the automatic detection of concepts and relations between concepts. The solution approaches could be classified into four main types: statistical techniques, supervised machine learning techniques, natural language processing techniques, and dictionary-based techniques.

- *Information retrieval and visualization of the domain.* We will analyse some mechanisms related with the fields of Information Retrieval, and Knowledge and Information Visualization. Concretely, we will pay special attention to:

  Information retrieval: is referred to the retrieval of information fragments which present a high degree of relation with natural language user queries. In the literature, the field is usually known as Question Answering. In addition, the most common application framework relies on the FAQ (Frequently Asked Questions) documents. The approaches can be grouped into two classes: knowledge based methods, and statistical methods.

  Tags clouds: is a visualization method that presents the most popular tags of a virtual community into a cloud representation. The problem considering the selection of tags to be visualized is known as tag selection. The statistical approaches are mostly employed to address this task.

  Concept maps: is referred to the techniques employed for organizing and representing the knowledge of a domain by means of a network of concepts. The nodes represent the concepts, and the links represent the relations existent between the concepts. Although the visual representation does not carry special difficulties, the knowledge acquisition stage is considered a complex task. The solution approaches could be classified into four main types: statistical techniques, supervised machine learning techniques, natural language processing techniques, and dictionary-based techniques.

- *Adaptation of the environment.* We will investigate the adaptation of the environment paying attention to two e-learning related fields.

  Automatic test generation: is referred to automatic generation of assessment tests in function of assessment objectives selected by the user. The field of Computerized Adaptive Tests addresses this task. It is common to find template-based strategies and statistical techniques as main approaches.

  Intelligent virtual tutor: is referred to a specific area of e-learning. It is focused on the development of educative systems having the ability of simulating a human tutor that should provide with specific learning material and different assessment criteria in function of the individualized student's characteristics. The difficulty of the process relies on the identification of the student's characteristics, known as student modelling. As main solution approaches we can find techniques based on: the overlay model, the stereotype Model, the perturbation model, Machine Learning, cognitive theories, fuzzy logic, Bayesian networks, and ontologies.

The second stage of this methodology concerns the design of a complete e-learning system including the previously developed modules. The design of e-learning systems is far from being trivial. The effectiveness of the system depends on some factors: the intrinsic characteristics of the students, the level of interaction between the students, ... These factors evolve at the same time that the proper ICTs. Therefore, this stage has the aim of identifying and analysing the main factors affecting the effectiveness of e-learning systems.

Firstly, we will study two cognitive indicator usually employed to measure the effectiveness of e-learning systems (student satisfaction and student achievement). Then, we will be able to examine the factors that positively affect to such indicators. We will distinguish between methodological factors (student self-regulation, student interaction, summative/formative evaluation, blended learning) and application factors (student interaction, self-assessment, and feedback mechanisms). Then, we will present the design of the platform, and the corresponding regulatory methodology. Finally, we will validate the proposal by means of real cases of use. The particular methodology followed in this stage is summarized as follows:

- Discussion of the main indicators of effectiveness of e-learning platforms.
- Discussion of the factors which affect the indicators of effectiveness.
- Proposal of architecture and functionality.
- Proposal of the learning methodology.
- Introduction in real cases of use corresponding to higher education engineering courses.
- Validation of the proposal.

Chapter II

Knowledge Extraction and Organization Methods for the Educational Content

1 Introduction

The evolution of ICT and the development of the Internet has furthered the generation of content in every domain. During the information age, the information industry has allowed individuals to share information in interoperable and collaborative spaces. Nevertheless, the enormous amount of digital content entails evident issues from both accessing and organization perspectives. Some of these issues are the information overload, the user's disorientation or the cognitive overhead [PDB^{+}10]. Moreover, either organize, index or classify the digital content typically falls on the manager of ICT applications, which are not simple or fast tasks.

Focusing on educational environments, there are increasingly resources for supporting the corresponding learning activities. For example, Moodle users have available more than 214,000,000 quiz questions for their purposes [Moo14]. In a state of information overload, it could be possible that learners feel anxiety, stress and alienation [EM00]. What is more, learners can experience learning disorientation and find themselves being unable to acquire the domain knowledge in current learning environments with large volume of information [SCD00, DS97]. Given this situation, instructors and learners should count with elaborated strategies that facilitate the cognitive processes involved in the knowledge acquisition process. Regardless of the chosen strategy, the organization of the content determines the first logical step. Nonetheless, it is not usual to find mechanism to help managers (usually the teachers) to organize the digital resources present in their learning environments. Even if the environment presents the structural design for organizing the content, the manual classification of resources is very time-consuming and error prone. Above mentioned motivations suggest the study of computer-supported methods for extracting and organizing the knowledge from the educational content. The complexity of the task requires the use of intelligent and efficient techniques, making the Computational Intelligence the most fitted framework. It should be pointed out that we will focus on textual sources that are still the most common form of learning resources.

Attending to the nature of the task, there are different levels of knowledge organization. For example, we could organize the educational content in basis of simple labels representing the main entities in the domain. With this level of organization, learners could use a taxonomic structure for searching specific resources. Obtaining different synonyms of such labels could be very valuable as well. The set of possible terms referred to a same element could be considered as a concept.

Therefore, the detection of concepts instead of simple terms could benefit more complex user search, classifying now the resources by the main concepts of the domain. The higher level of semantic richness has the knowledge organization, the more complex functionally can be implemented. With the purpose of covering different levels of semantic richness, we will consider three different dimensions of semantic for meta-data structures: taxonomies, folksonomies and ontologies. In consequence, we will address the organization of the educational content in basis of the distinct characteristics of each structure.

1.1 Taxonomy

In first place, a *taxonomy* is hierarchical classification of entities. It is mostly created by the designer of the system or a knowledge engineer with domain knowledge. Authors (or categorizers) must find a good place in this hierarchy to position their content [Chr06]. A taxonomy should be clear and consistent, flexible, exhaustive, and practical [PAG09]. In the context of computer science, a taxonomy could be considered as a thesauri or controlled vocabulary reflecting the context and the content of the domain, with the aim of helping users search and handle the information in web scenarios [RM02]. According to the definitions found, it can be stated that taxonomies do not require that their elements should be connected by a specific type of relationship, i.e. it simply requires a correct organization. Taxonomies present the following advantages. First, they are portable and reusable. Second, their simplicity entails easy maintenance. Third, they improve the use of the supported applications because it significantly reduces the needs of cognitive load, memory and learning. Finally, they facilitate the interaction within the application, creating a consistent image of the content organized in them.

A common method to automatically construct a taxonomy from a document set consist of extracting the main terms of each textual source, i.e. the most representative terms in the document [MFH14, LSLW12, ZW08]. Consequently, the set of extracted terms forms the controlled vocabulary, whereas each document is classified and afterwards retrieved in basis of its main terms. This task is usually known as Automatic Keyword Extraction (AKE) or Automatic Key Term Extraction. Although this problem has been widely handled in the literature, we find a lack of methods dealing with the particularities of the educational content. For this reason, we present a specific method for extracting the key terms from learning resources in Section 2.

1.2 Folksonomy

The second organizational structure considered here is the *folksonomy* [Qui05]. A folksonomy is a non-hierarchic labelling methodology consisting of collaboratively generated open-ended labels that categorize web objects. The labels, commonly known as *tags*, are added and managed freely by people in social environments. Each user can freely assign one or more tags in order to classify each element related to a same topic. Thus tags provide meta-data about the topic of each web object. The labelling process is called *tagging*. This practice has become a popular strategy in several well-known social websites (Bibsonomy[1], Delicious[2], or Flickr[3]). Folksonomies deviate from hierarchical structures to approach an organization based on collaboration. Thence, it stands to reason that folksonomies have been well-received in the trend of the Web 2.0.

The absence of any kind of restriction for tagging the resources represents the main advantage

[1] http://www.bibsonomy.org
[2] http://delicious.com
[3] https://www.flickr.com

of folksonomies. According to Noruzi [Nor06], folksonomies are most useful when there is nobody in the “librarian” role or there is simply too much content for a single authority to classify. This kind of knowledge organization scheme is built attending uniquely to the community users' mental model. Folksonomies are easy to create due to the users do not need any special skill to tag [HJSS06]. Additionally, it represents the evolving vocabulary of the whole community of users that participate in the process[GSCAGP12, GH06]. Users tend to stabilize the vocabulary employed to tag similar resources through several iterations. Therefore, the commented classification scheme is able to be rapidly adapted to continuous changes concerning terminology and expansion of the domain.

Unfortunately, the simplicity of the process also entails some important disadvantages due to the lack of semantics and relations among tags (see Section 3). Tag Recommender systems have been developed to assist the user in the task of tagging. These systems usually suggest to a particular user a subset of tags from the complete tag space in order to annotate a particular resource. After the recommendation, the user is able to freely select which tag or tags he prefers to label the resource. Despite the fact that this kind of applications are nowadays very popular, current approaches are hardly suitable for the educational context. Most of them are rather dependent on the domain of social web sites. Those which might be used in our context require predefined meta-knowledge structures, adding further difficulty to the implantation. Thereupon we have developed a tag recommender system able to work in the majority of common educational domains (Section 3).

1.3 Ontology

The structure with higher level of semantic richness presented here is the *ontology*. Gruber [Gru93] defined an ontology as:

> “an explicit specification of a conceptualization (. . .) When the knowledge of a domain is represented in a declarative formalism, the set of objects that can be represented is called the universe of discourse. This set of objects, and the describable relationships among them, are reflected in the representational vocabulary with which a knowledge-based program represents knowledge.”.

In other words, ontologies are formal conceptualizations of the knowledge and relations in a given domain. They provide a shared and a common understanding of a domain, by means of controlled *vocabulary* including *objects*, *properties*, and *relations*, each with an explicitly defined and machine processable semantics. Those concepts are usually hierarchically defined by means of taxonomies of classes. Most common components in ontologies also include individuals, axioms, or formal restrictions. Usually, term *domain-ontology* is preferred to emphasize the dependence on the domain. That is, an ontology models the *part of the world* regarding a specific domain. Thus, among the various meanings of a given term, only that related to the model is specified in the ontology. Ontologies play a major role in supporting the information exchange and sharing by extending syntactic interoperability of the Web to semantic interoperability [HEBR11]. The cheap and fast construction of domain specific ontologies is essential for the success and the proliferation of the Semantic Web [MS01].

The construction of ontologies entails a difficult challenge on its own, entailing problems on time, difficulty and confidence. *Ontology Learning* (OL) is the field in charge of facilitating the construction of ontologies. Because the fully automatic acquisition of knowledge by machines remains in the distant future, the OL process is usually considered as semi-automatic with human intervention

[MS01]. Most of approaches in this field require machine learning techniques, complicating their integration in contexts where the users do not necessarily have technical knowledge. Accordingly, we have designed an OL mechanism in order to help teachers to build an ontology from the learning resources present in the system.

The rest of this chapter is structured as follows. Next, Section 2 offers an study on Automatic Key Term Exaction as reliable method for constructing taxonomies. The collaborative construction of folksonomies guided by a Tag Recommender will be explained in Section 3. We present our study on Ontology Learning in Section 4 and conclude with some final discussions in Section 5.

2 ALKEx: Key Term Extraction system for Domain Taxonomy Construction from Learning Resources

2.1 Introduction

The demand for and adoption of on-line learning system has triggered a need for learning resources that support learning regardless the knowledge domain and the individual students' characteristics. In order to overcome the problems derived from information overload state, the learning resources should be organized into a knowledge structure allowing an effective navigation through the learning content. In fact, the presence of navigation methods is crucial in the scope of continuous learning [ZZZNJ04]. An index of learning resources favours a quickly access to those parts of interest. Nonetheless, creating an index for a steadily growing amount of learning material is a lot of work, especially as there is usually little or no support from the VLE. Flat taxonomies perfectly fit this demand, and are a suited model for indexing of learning resources. In flat taxonomies, all the contained elements belong to the same category. That is, all the elements have equal weight.

In this context, the use of a Information Extraction technique would help to collect meta-data for each learning resource in order to automatically create an index. The manager of a VLE could take advantage of an AKE method for collecting the most meaningful terms from all the learning material elements. Subsequently, the key terms could be arranged together with links to the content. Although AKE is well-studied field, the potential of such method to enhance e-Learning remains to be explored.

The inherent particularities of learning resources make the application of generic AKE methods inappropriate. Classical AKE studies deal with general-domain documents (containing world knowledge from public sources covering high diffusion topics). In addition, each document presents extensive and self-describing information about its main topic. If we consider common learning resources such as quiz questions (true/false answer style, multiple choice style, fill-in-the-blank, and descriptive answer style), problem-solving guides, or support documents (tutorials, Frequently Asked Question lists, walkthroughs, etc.), we can observe substantial differences. Their topics are frequently referred to specific-domains (containing knowledge from specific issues, usually with limited or private diffusion). Moreover, this kind of documents do not contain every detail of the considered topic, but only specific information referred to significant aspects of it.

Attending to these features, we distinct two classes of key terms in e-learning context: "Multi-domain key terms", meaningful in a multi-domain context; and "Specific domain key terms", proper nouns or technical terms closely related to specific aspects of the document topic. We provide an example of such distinction in Table II.1. It contains a question-answer pair (formulated by a teacher and answered by a student) about the topic "principle of knowledge" of an Artificial Intelligence (AI) course. The terms "Artificial Intelligence" or "knowledge" are easily recognizable considering a multi-domain scenario. However, the term "principle of knowledge" is specific of the domain.

The above commented situation has motivated us to design an system to automatically extract key terms from learning resources. There exist a number of approaches in charge of solving this task that are analysed in Section 2.2. From them, we have decided to combine two strategies. As first strategy, a thesaurus-based approach is employed to detect multi-domain key terms. To that end, we found rather appropriate to develop a controlled dictionary of concepts drawn from Wikipedia[4]. On the one hand, Wikipedia is currently one of the most valuable multi-domain sources of information. Attending to the literature, it has been used in a number of approaches with very promising results.

[4] http://en.wikipedia.org/

Question answer pair about a AI related problem
Q: *State the principle of knowledge and discuss its importance in the field of Artificial Intelligence.* ***A:*** *Thanks to this principle, it was realized that the ability of a computer program to solve problems does not lie in the formal expression or logical inference schemes employed, but in the knowledge it possesses.*
Key terms: *principle of knowledge, Artificial Intelligence, knowledge*

Table II.1: Key terms extracted from the 'principle of knowledge' related question-answer pair

On the other hand, Wikipedia contains an evolving vocabulary of concepts highly related with learning. Considering now the second strategy, we focus our attention on frequency in language feature of terms. As it has been commented, specific-domain key terms are hardly contained in multi-domain controlled dictionary due to its technical nature. Specific-domain terms are probably uncommon terms in the language. Thus a term frequency list would facilitate their detection. Bearing this in mind, we have determined to construct a frequency in language dictionary from word frequency lists available in the Corpus of Contemporary American English (COCA) project [IM01]. In addition, we have designed a modular filter with the aim of discarding noisy terms from the learning documents.

The rest of this section is organized as follows. Subsection 2.2 offers an overview of previous work on key term extraction and e-learning. The Wikipedia-based dictionary of concepts is presented in Subsection 2.3. Next, we describe the frequency in language feature of terms and its applications in Subsection 2.4. Subsection 2.5 describes the architecture of our proposed system. The method of analysis and the experimental validation of our method are outlined in Subsection 2.6. Finally Subsection 2.7 concludes with a discussion of results and future research.

2.2 Related Works

In this section, we firstly review the main approaches in Keyword Extraction, analysing the characteristics of each approach considering our application domain. Next, we comment the few existing researches that consider the domain of e-learning.

2.2.1 General AKE methods

First works on this field were based on Machine Learning (ML) algorithms. Supervised learning methods treat this task as a classification problem using lexical, syntactic or statistical features (or a mixture of them) of the training labelled data to extract keywords [ZHXL06, HKSKF07, LICC12, CM08, XL10]. In this way, supervised algorithms can extend any other AKE approaches. The most classical approaches were suggested by [Tur00] and [FPW$^+$99]. In [Hul03], the author performed a rule induction-based method using term frequency, collection frequency, relative position of the first occurrence, and part-of-speech tag of terms as features. Another well-known proposal is KEA [WPF$^+$99]. This tool employs a Naïve Bayes algorithm using the Term Frequency - Inverse Document Frequency (TF*IDF) measure and the first occurrence of a term as features. Machine Learning methods depend on a training data set for and may provide poor results when the training set does not fit well with the processed documents. Collections of learning resources contain documents belonging to specific-domains. Therefore, if new documents are added to a collection, they will probably belong to domains not considered in the training dataset. In addition, taking a ML approach will force the manager of the VLE (usually the teacher) to know deep knowledge about the training process, being this a very strong limitation. Concluding, we consider that ML is not the most suitable strategy to detect neither multi-domain nor specific-domain key terms.

Next, frequency-based approaches employ document frequency of appearance as criterion for weighting terms' importance. Most of frequency-based approaches make use of TF*IDF weighting algorithm [WS98]. The first approximation was proposed by Salton and Buckley in 1988 [SB88]. This model has been extended in other works [Ker03]. Nevertheless, frequency-based approaches do not seem appropriate for detecting key terms from learning resources. This kind of scheme would discard infrequent terms that could actually be meaningful in the domain.

More recent, word association-based approaches attend to the correlations between candidate words. These schemes assume that semantically similar words tend to occur in similar contexts. Matsuo et al. [MI04] implemented an algorithm based on the co-occurrence distribution information of the candidate terms and the frequent terms of the document, i.e. occurrences in the same sentence. In their approach, Wartena et al. [WBS10] assumed that if a word occurs in a number of documents on the same topic, it has more discriminative power than a word occurring in the same number of documents but scattered over different topics. Other statistical key term extraction methods without reference corpora can be consulted in [BRK05, PNPH08]. Considering the particularities of our task, word association-based approaches might be also inappropriate due to we have not found significant co-occurrence distributions between terms in learning resources.

Word position-based approaches attend to the outline of the document, assuming that key terms appear more often in particular positions of the document [Tur00, HKSKF07]. Hence, they are applied to structured documents such as XML documents, scientific papers or Wikipedia articles. From the e-learning perspective, learning resources do not present standardized and well-defined structures, being this an obstacle for word position-based approaches.

Afterwards, linguistic-based approaches are based on linguistic parsing and pattern matching using the part-of-speech tag [MS99] of terms in: filtering procedures, as a feature in classification schemes [KKK08], or as pattern in the candidate extraction process. This strategy is not usually employed to develop a complete system.

Following, graph-based approaches model each document as a graph where the vertices represent the candidate terms. Any possible relation between two vertices is a potentially useful connection that defines the edge between them. Subsequently, graph analysis algorithms[BP98, AS08] are used to rank the candidates. Mihalcea and Tarau introduced the TextRank graph-based ranking model [MT08] that uses the co-occurrence relation of the candidates. Litvak and Last [LL08] compared two approaches: supervised and unsupervised. Meanwhile, Grineva et al. [GGL09] applied a network analysis algorithm on semantic graph representation of the document. In their work, key terms are selected from the densest groups discarding those belonging to sparse groups.

In last few years, controlled vocabularies have been widely used for extending classical approaches, not only in keyword extraction task [WLT09, BP06, MWM08, MC07, XL10, GGL09, CMM08, WHZC09, GWB10]. Thesaurus-based methods extend the previous strategies exploiting the background knowledge drawn from manually constructed thesaurus [WLT09, GWB10], or online dictionaries or encyclopaedias such as Wikipedia [BP06, MWM08, MC07, XL10, GGL09, CMM08]. Mihalcea and Csomai implemented the Wikify! system exploiting the linking relation of their candidate terms [MC07]. In the work of Coursey et al. [CMM08], two previous systems [MC07, MT08] were associated in order to build a hybrid system improving the results of each approach separately. As we previously commented, multi-domain key terms will be detected following this strategy.

Further, there are systems with keyword extraction modules for performing more complex tasks: for finding meaningful keyword sequences [CL11], for finding key phrases for text summarization [LL08], for providing document headlines [XL10], or for supporting computer assisted writing systems [LLYC11, LLD12].

2.2.2 AKE methods in e-learning

The use of key term extraction techniques in the e-learning domain is rather uncommon. Although there are a number of approaches, all of them make use of simple statistical techniques for extracting relevant single keywords.

Yu and Luna [YL13] proposed a text mining technique for discovering and extracting knowledge from a collection of survey comments evaluating an E-learning system. The main purpose of their approach was to assist the user in the task of analysing the feedbacks from students in order to come up with conclusions about the e-learning system evaluated. To that end, they employed a clustering technique on the set of keywords extracted from each survey comment. Keywords were identified using statistical term weighting algorithms: entropy, Term Frequency - Inverse Document Frequency (TF*IDF) and Inverse Document Frequency (IDF). They found that their method captured only important terms which represented the main focuses or concerns of the document collection as a whole. However, some important terms which occurred only in few documents were not included in any specific cluster. Avelãs et al. [ABDGM08] designed a keyword extractor for adding a new functionality to VLEs. They focused on the extraction of keywords from text-based Learning Objects using the TF*IDF algorithm and a term frequency adjusted version of IDF as weighting methods. TF*IDF method obtained the best results although they provided a poor valuation (F-measure = 0.22). Other works based on the application of the TD*IDF algorithm as keyword weighting method for e-learning resources can be consulted in [LVK$^+$07, LH12]. Similarly, Takase et al. [TKT12] proposed a method to extract keywords from descriptive answers of a quiz. They assumed that answers consisted of a few sentences only. Additionally, they considered that answers tended to be similar to each other. Since quizzes are deeply linked to the lecture, students would answer with similar expressions and limited vocabulary. In order to extract suited keywords, they focused on frequency of each word to evaluate importance: words frequently used in a document are important for the document, and words used in many documents are not important. Subsequently, they employed the radial basis function in Gaussian type for extracting the most important words from each answer. Finally, Kuo et al. [KCTL03] presented CanFind, a semantic image indexing and retrieval system. Having the aim of identifying the target images of interest in the database in the conceptual level, their system made use of previously annotated keywords as the input of searching vehicle. Since the system does not detect keywords from textual sources, it falls out the boundaries of this part of the research.

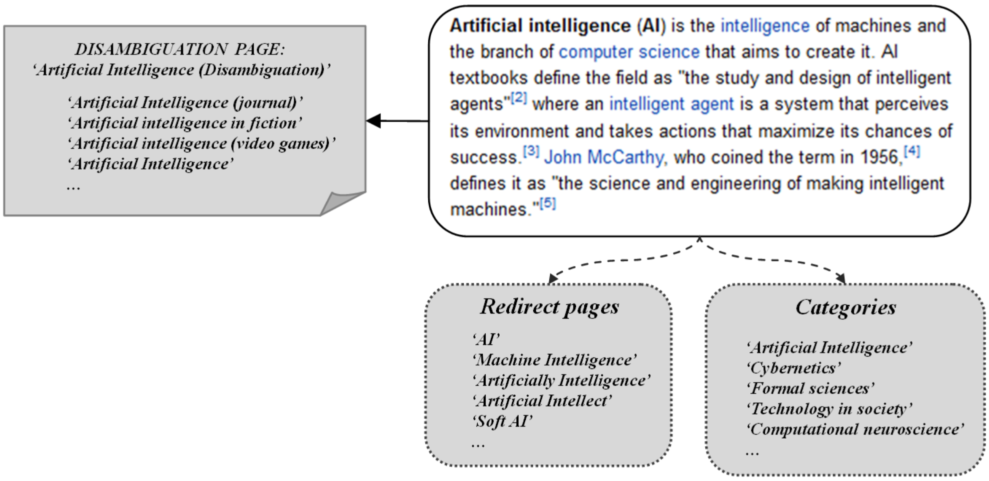
2.3 Wikipedia-based Knowledge Source

In this section, we explain characteristics of the dictionary of concepts extracted from Wikipedia. This dictionary is not only one of the core components of our AKE system, but also it takes place in most of the intelligent solutions designed along this phD Thesis.

2.3.1 Wikipedia-based dictionary of concepts

The free on-line encyclopaedia Wikipedia[5] provides an extensive coverage about a large range of general-domains, and has been successfully used in Natural Language Processing applications (Named Entity Disambiguation [BP06], Topic Indexing [MWM08], or Text Classification [GM06, WHZC09]).

Each Wikipedia article describes a complete entity so-called *concept*, i.e. a complete representation of the information drawn by one Wikipedia article. A concept is compound by an identifier name or title, a set of alternative aliases or synonyms, and the categories (hyponyms) to which it belongs. In addition, the number of articles in which its title appears as a link ($numArtHv$), and the number of articles in which its title appears ($numArt$) are stored as useful information. Figure ii.1 shows an example of how the structure of the Wikipedia 'Artificial Intelligence' article defines a concept.



Artificial intelligence (AI) is the intelligence of machines and the branch of computer science that aims to create it. AI textbooks define the field as "the study and design of intelligent agents"[2] where an intelligent agent is a system that perceives its environment and takes actions that maximize its chances of success.[3] John McCarthy, who coined the term in 1956,[4] defines it as "the science and engineering of making intelligent machines."[5]

DISAMBIGUATION PAGE:
'Artificial Intelligence (Disambiguation)'
'Artificial Intelligence (journal)'
'Artificial intelligence in fiction'
'Artificial intelligence (video games)'
'Artificial Intelligence'
...
Redirect pages
'AI'
'Machine Intelligence'
'Artificially Intelligence'
'Artificial Intellect'
'Soft AI'
...
Categories
'Artificial Intelligence'
'Cybernetics'
'Formal sciences'
'Technology in society'
'Computational neuroscience'
...

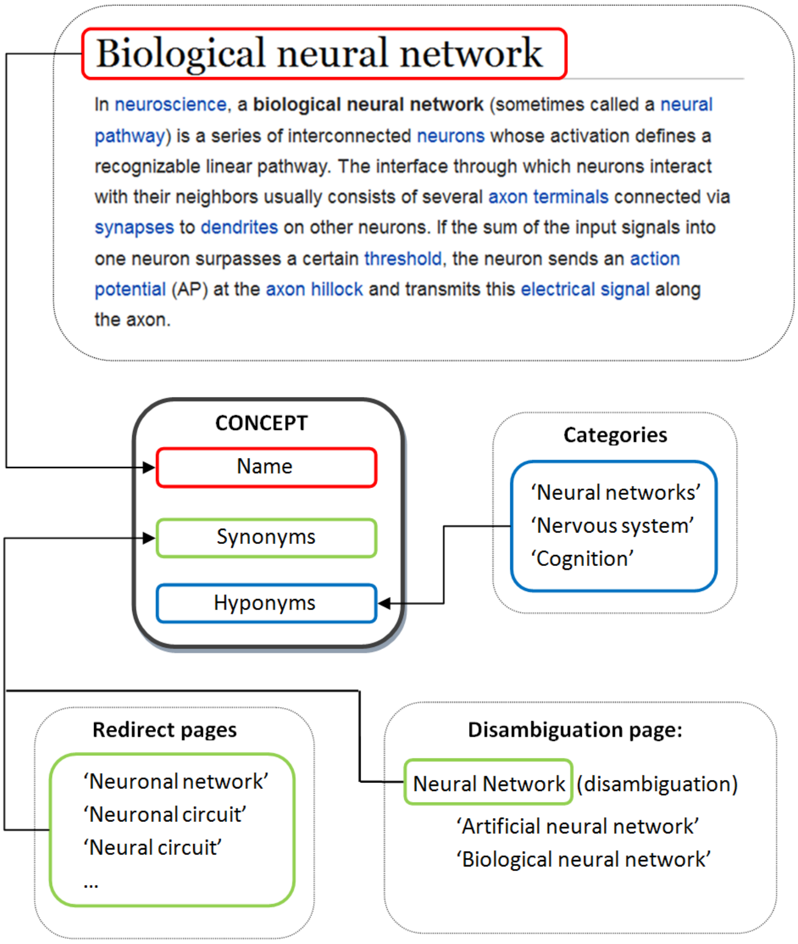
Figure ii.1: Example of how a Wikipedia article's structure defines a concept

Following the approach commented in [WHZC09], we make use of these concepts to generate a dictionary. In that work, authors used the dictionary for Text Classification. Now, we extend it for Keyword Extraction. In order to build the dictionary, we have discarded disambiguation pages; redirect pages; appendixes; articles belonging to chronological or numerical categories (Integers, Perfect numbers, Numbers, Rational numbers, Real numbers...); and articles whose titles are a sequence of stop words only. Then, a concept is created from each valid article and it is mapped in a dictionary structure. The identifier name is extracted from the article's title. The synonyms are extracted from the title of the redirect pages linked to the article, and from the title of the disambiguation page linked to the article. Then the article's categories define the hyponyms. Figure ii.2 summarizes the process.

2.4 Term Frequency in Language

This section describes the term frequency in language dictionary for English. For obtaining the frequency of terms that are not contained in this dictionary, we have designed an algorithm for calculating an approximate simulated frequency based on Google's search engine.

[5] http://en.wikipedia.org



Biological neural network

In neuroscience, a **biological neural network** (sometimes called a neural pathway) is a series of interconnected neurons whose activation defines a recognizable linear pathway. The interface through which neurons interact with their neighbors usually consists of several axon terminals connected via synapses to dendrites on other neurons. If the sum of the input signals into one neuron surpasses a certain threshold, the neuron sends an action potential (AP) at the axon hillock and transmits this electrical signal along the axon.

Figure ii.2: Concept extracted from 'Biological neural network' article

2.4.1 Term Frequency dictionary

Word frequency lists have proven to be a useful tool in practical researches [SFK00]. We have chosen the frequency lists extracted from the Corpus of Contemporary American English[6] (COCA) because it is the largest publicly-available, genre-balanced corpus of English. This text corpus has 425 million words extracted from spoken, fiction, popular magazines, newspapers, and academic texts, ant it has been used in Computational Linguistic tasks [Dav10].

Following the structure of the list, we develop a dictionary of 1-grams, 2-grams and 3-grams that are mapped to their absolute and normalized frequency. It allows us to find which terms in a document are less frequent in the language.

2.4.2 Google-based simulated frequency

It is likely that some terms will not be contained in the frequency in language dictionary. For this reason, we will approximate their frequency heuristically by considering the number of results returned by the Google Search engine.

[6] http://www.wordfrequency.info/intro.asp

Google search engine is one of the most common linguistic resources in Computational Linguistics tasks [CL11, AEARAK$^+$11]. This is not unusual due to Google contains more than 25 billion web pages indexed. We take advantage of it using the search engine for obtaining an approximation of the frequency in language of uncommon terms. The number of results obtained in a search is directly related with the commonness of the query terms. For example, most common terms (such as 'the', 'of', and 'and') return the highest number of results in a Google query (25,270,000,000 results) whereas infrequent terms (such as 'equanimous' or 'succumb') return a number of results much more few. Moreover, the order of the words in a query affects to the number of results. In this context, we think that regression analysis[KK62] could be helpful to predict the frequency in language value of terms. It provides a conceptually straightforward method for investigating functional relationships between one or more factors and an outcome of interest. The relationship is expressed in the form of an equation or a model connecting the response or dependent variable and one or more explanatory or predictor variable.

By considering the frequency in language as the dependant variable, and the number of results in Google as the predictor, we are able to apply the regression analysis strategy (following an a-priori sample size calculator schema [AS65, Coh88, CCWA03]) in order to obtain the regression equation[7]. The details of the complete process and the resulting regression equation can be consulted in Appendix 1.

2.5 System Overview

In this section, we provide the details of the multi-stage architecture of our AKE system: the Automatic Learning-Domain Key Term Extraction (ALKEx). The system is able to detect two classes of key terms from the learning resource collection: multi-domain key terms (MKT), and specific-domain key terms (SKT).

The system is composed by the following modules (Figure ii.3). In an initial stage, the Preprocessing module prepares the content of the documents in the collection. Then, the algorithm performs at document-level. The Candidate Extraction module extracts two sets of candidates. Those terms with a frequency in language lower than a predefined threshold are Specific-domain Candidate Terms (SCT). The rest of terms conform the Multidomain Candidate Terms (MCT) set.

In order to illustrate each set, let us consider the example in Table II.2. It includes a question-answer pair about AI search algorithms. The terms “breadth-first search” and “depth-first search” references a concrete AI-related term. These proper nouns have an insignificant degree of use in English, but are completely meaningful in the specific domain of application. In contrast, the terms “search algorithm” or “collection of items” in the same example are key terms in a multi-domain context.

2.5.1 Preprocessing Module

The goal of this module is to prepare the document collection for the subsequent steps. For each document in the collection several language processing techniques are applied: splitter, part-of-speech (POS) tagging [MS99], stop words removal, and lemmatization. The GPL Library FreeLing

[7] Since the Internet is continuously evolving, the regression curve could not be considered as a static result. Instead, it should be considered as a pre-calculation for the algorithm that may be periodically repeated in order to gain in reliability.

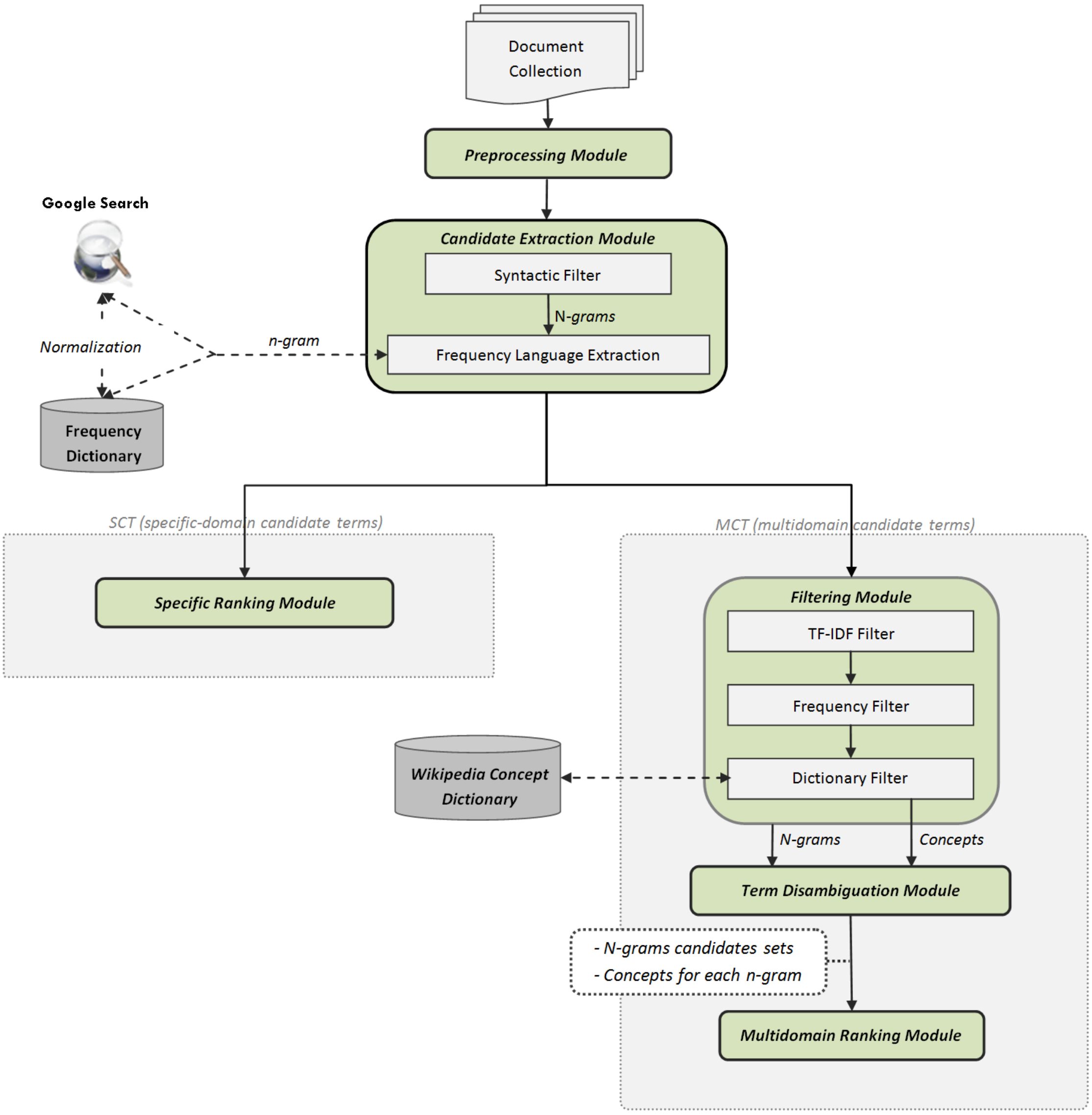

Document Collection
Preprocessing Module
Google Search
Candidate Extraction Module
Syntactic Filter
N-grams
Normalization
n-gram
Frequency Language Extraction
Frequency Dictionary
SCT (specific-domain candidate terms)
MCT (multidomain candidate terms)
Specific Ranking Module
Filtering Module
TF-IDF Filter
Frequency Filter
Wikipedia Concept Dictionary
Dictionary Filter
N-grams
Concepts
Term Disambiguation Module
- N-grams candidates sets
- Concepts for each n-gram
Multidomain Ranking Module

Figure ii.3: System architecture

Learning document	Multi-domain Key terms	Specific domain Key terms
Q: *What is a search algorithm?* ***A:*** *A search algorithm is an algorithm for finding an item with specified properties among a collection of items. There exist different approaches: breadth-first search, depth-first search, (...)*	search algorithms, collection of items	breadth-first search, depth-first search

Table II.2: Key terms from a computer science question-answer pair

3.0[8] was used to implement the preprocessing.

[8] http://nlp.lsi.upc.edu/freeling/

We deal with the different morphological variations of words by means of lemmatization. For example, 'link, 'linked' and 'links' share the lemmatized form 'link'. By lemmatizing each concept title and synonyms we reduce the candidate set of key terms.

2.5.2 Candidate Extraction Module

The Candidate Extraction module obtains a set of MCT and a set of SCT. This module includes a Syntactic Filter and a Frequency Language Extraction module. All the 1-grams and 2-grams of the document are taken as input.

First, the Syntactic Filter restricts the set of input terms composing the initial precandidate set. We have empirically contrasted that relevant key terms are likely nouns or adjectives. Thus, other ones are filtered out in this stage.

Later, the Frequency Language Extraction module assigns the frequency in language value to each precandidate term. This value is obtained from the frequency language dictionary if the term is stored in it, or by means of the Google-based frequency algorithm otherwise. If the frequency in language value exceeds a predefined threshold α, the precandidate is considered as MCT. In the other case, it is considered a SCT. From this point, each set follows a different procedure.

The α-threshold should be regarded as the degree of commonness in language of a term (it establishes how common a term is in order to be considered a specific-domain term). The tune of this threshold only affects on the delimitation of each candidate set. The adjustment of α-threshold is discussed in Subsection 2.6.4.

The complete candidate classification process is shown in Table II.3. In this algorithm, $candidate \cdot POS()$ denotes the POS tag for the given candidate's words, $freqDict(x)$ denotes de frequency in the language for the given term x, $googleSearch(x)$ denotes de number of Google search results for the given term x, and $regFunction(y)$ denotes the approximated frequency resulting of the regression function applied to a given number y.

2.5.3 Multi-domain Candidates' Procedure

As a result of the previous algorithm, a set of MCT was obtained. Then, the Multi-domain candidate procedure refines the MCT set in order to obtain a rank. In this section we explain how the Filtering Module discards potentially irrelevant candidates, and how the Term Disambiguation Module unifies the candidates in order to compose the final ranking.

Up to now, all terms matching the part-of-speech patterns composed the candidate set. Thus, it is very likely that this set will be too large. Consequently, we develop a multi-filtering procedure in order to adjust the MCT set leading a more direct and efficient ranking process. This Filtering module is composed by three main parts: (i) a statistical filter, (ii) a frequency filter, and (iii) a dictionary-based filter.

TF*IDF Filter . The statistical TF*IDF filter discards frequent MCT that are also frequent in the entire corpus. TF*IDF is a widely used measure that weights the importance of words according to their relative frequency of appearance in the document (term frequency, tf), and the number of documents containing them (document frequency, df) (Equation II.1). To access a vaster discussion

```
INPUT:
    $C$: set of 1-grams and 2-grams parsed through the document.
    $P = \{NN, NP, JJ\}$: POS tag subset (NN=common noun, NP=proper noun,
        JJ=adjective).
    $freqDict$: Frequency language dictionary.
    $\alpha$ = threshold of frequency language value.

OUTPUT:
    $multidomainSet$: Multi-domain Candidate Term set.
    $specificSet$: Specific-domain Candidate Term set.

begin
    $multidomainSet \leftarrow \varnothing$
    $specificSet \leftarrow \varnothing$
    foreach $candidate_i \in C$ do
        if $candidate_i \cdot POS() \in P$ then
            if $candidate_i \in freqDict$ then
                $freqValue \leftarrow freqDict(candidate_i)$
            else if
                $numResults \leftarrow googleSearch(candidate_i)$
                $freqValue \leftarrow regFunction(numResults)$
            end if
            if $freqValue < \alpha$ then
                $specificSet \leftarrow specificSet \cup candidate_i$
            else if
                $multidomainSet \leftarrow multidomainSet \cup candidate_i$
            end if
        end if
    end for
end
```

Table II.3: Candidate Extraction algorithm

about this method, the reader can consult the work performed by Khoury et al. [KKK08]. Finally, those MCT whose score is lower than a predefined threshold β are filtered out.

$$score(U, S) = \frac{\sum_{w \in U,S} \lambda_w^2 \cdot tf_U(w) \cdot tf_S(w)}{\sqrt{\sum_{w \in U} tf_U(w)^2 \cdot \sum_{w \in S} tf_S(w)^2}} \tag{II.1}$$

$$\lambda_w \equiv idf(w) = log \left(\frac{|D|}{|\{d \in D : tf_d(w) > 0\}|} \right) \tag{II.2}$$

The β-threshold determines the maximum degree of TF*IDF frequency of a term in the collection for being considered as key term. If a term appears frequently in all the domains of the collection, then it will hardly be considered key term. The tune of β-threshold will be depicted in Subsection 2.6.4.

Frequency Filter . Following the TF*IDF criterion, if a term appears frequently in any document of the language, it would be hardly considered key term. For example, adjectives like 'small' or 'different' appear frequently in texts of any domain. They would be hardly considered as key terms by human annotators because they are too generic to clarify the content of a document. Hence, the Frequency filter removes those MCT that are common in the language, i.e. those whose frequency in language value (computed in the Frequency Language Extraction Module) exceeds a certain threshold γ. Subsection 2.6.4 can be consulted for comprehending how to adjust the γ-threshold.

Dictionary Filter . The last step in the multi-filtering stage tries to relate every candidate with the previously collected concepts. Each MCT is searched through the synonyms of the concepts. The matching concepts are stored as possible concepts for the MCT (Figure ii.4).

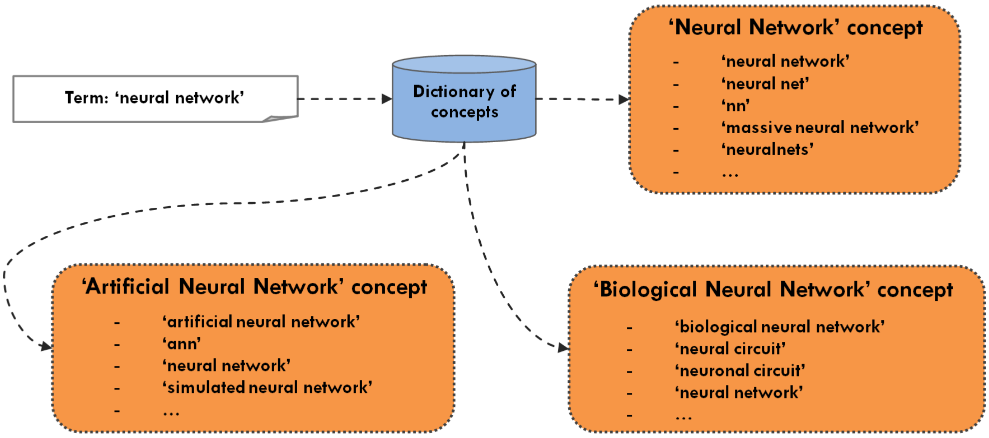


Figure ii.4: Related concepts for 'neural network' term

Hence, the goal of the Dictionary Filter is two-fold. On the one hand, those terms unrelated to any existing concept are discarded. On the other hand, every related concept is discovered. Both the valid MCT and their related concepts are submitted to the Term Disambiguation module that tries to discern the most appropriate candidate-concept pair.

The Term Disambiguation module deals with polysemous candidates, discerning the intended concept from all the possible meanings within the document context. We adopt a methodology based on the first word sense disambiguation strategy explained in [WHZC09].

The Term Disambiguation strategy is the following. Given a term and a set of ambiguous concepts, and considering the document as the disambiguation context, we compute the similarity measure by means of their vectorial representation (Equation II.3). Thus, the most similar concept to the document is taken.

$$cosSim(\vec{U},\vec{V}) = \frac{\vec{U}\cdot\vec{V}}{|\vec{U}|\cdot|\vec{V}|} \tag{II.3}$$

It is possible that concepts with few synonyms tend to present the same cosine similarity mean. In this case, the concept whose hyponyms obtain the higher average of Google's results in Wikipedia site is taken.

The remarking Term Sense Disambiguation process is shown in Table II.4. In it, $x.concepts()$

returns the set of ambiguous concepts related to a given candidate x, $c.synonyms()$ returns the set of synonyms and the title of a given concept c, $c.hyponyms()$ returns the set of hyponyms for a given concept c, $vectorRep(y)$ returns the TF*IDF vector representation for a given string y, $cosSim(\vec{u}, \vec{v})$ denotes the cosine similarity value computed for the given vectors $\vec{u}$ and $\vec{v}$, and $gWSearch(x)$ denotes the number of Google results for the string x on a search in the 'en.wikipedia.org' site.

```
INPUT:
    $\vec{doc}$: vector representation of the document.
    $C$: multi-domain candidate term set

OUTPUT:
    $ccPairSet$: candidate-concept pair set.

begin
    $ccPairSet \leftarrow \varnothing$
    foreach $candidate_i \in C$ do
        $cmaxAvgCos \leftarrow 0$
        foreach $concept_j \in candidate_i \cdot concepts()$ do
            $cosSimSum \leftarrow 0$
            foreach $synonym_k \in concept_j \cdot synonyms()$ do
                $\vec{v} \leftarrow vectorRep(synonym_k)$
                $actualCosSim \leftarrow cosSim(\vec{doc}, \vec{v})$
                $cosSimSum \leftarrow cosSimSum + actualCosSim$
            end for
            $actualAvgCos \leftarrow \frac{cosSimSum}{Numberofsynonymsofconcept_j}$
            if $actualAvgCos > maxAvgCos$ then
                $maxAvgCos \leftarrow actualAvgCos$
                $selectedConcept \leftarrow concept_j$
            end if
            if $actualAvgCos = maxAvgCos$ then
                $gResSum \leftarrow 0$
                $maxAvgRes \leftarrow 0$
                foreach $hyponym_k \in concept_j \cdot hyponyms()$ do
                    $actualRes \leftarrow gWSearch(hyponym_k)$
                    $gResSum \leftarrow gResSum + actualRes$
                end for
                $actualAvgRes \leftarrow \frac{gResSum}{Numberofhyponymsofconcept_j}$
                if $actualAvgRes \geq maxAvgRes$ then
                    $maxAvgRes \leftarrow actualAvgRes$
                    $selectedConcept \leftarrow concept_j$
                end if
            end if
        end for
        $ccPair \leftarrow$ create pair with $candidate_i$ and $selectedConcept$
        $ccPairSet \leftarrow ccPairSet \cup ccPair$
    end for
end
```

Table II.4: Term Sense Disambiguation algorithm

The last step performed by ALKEx system regarding the MCTs is carried out by the Multi-domain Ranking module. It assigns a numeric value to each candidate and establishes a ranking. We compute a score based on one of the most well-known measures used in the literature: *keyphraseness*. This measure computes the probability for a candidate term of being a hyperlink (i.e. an important term in Wikipedia) in any article of Wikipedia. Terms achieving high keyphraseness are considered consequently keywords or keyphrases [MC07]. For each candidate c, the probability is computed as follows:

$$keyphraseness(c) = \frac{count(D_{Link})}{count(D_a)}$$

where $count(D_{Link})$ is the number of Wikipedia articles in which this candidate appears as a link (the pre-stored $numArtHV$ value), and $count(D_a)$ is the number of articles in which it appears (the pre-stored $numArt$ value).

However, the authors of this proposal computed this measure in terms of the document collection as a whole, rather than in terms of individual documents [MWM08]. In this sense, this technique assumes that a candidate term has the same discriminative power in any document of the collection. This consideration may be right for classical datasets tested in Keyword Extraction task but this is not the case in the context of e-learning resources. One term with high discriminative power in the domain of e-learning will probably be inessential in any other document of the collection. Attending to the example in Table II.5 that consider the prior AI-related question-answers, the term 'principle of knowledge' is a key term in the first document but no so for the second one. However, the keyphraseness measure would select (or not) the term as key term in both documents. Although these documents are about different topics, they belong to the same course. For these reasons, we need to extend this score method by adding a different feature.

Question-answer about 'principle of knowledge'	**Question-answer about 'search algorithms'**
Q: *State the principle of knowledge and discuss its importance in the field of Artificial Intelligence.* ***A:*** *Thanks to this principle, it was realized that the ability of a computer program to solve problems does not lie in the formal expression or logical inference schemes employed, but in the knowledge it possesses.*	***Q:*** *What is a search algorithm?* ***A:*** *A search algorithm is an algorithm for finding an item with specified properties among a collection of items. There exists different approaches: breadth-first search, depth-first search, (...)*
Key terms: *principle of knowledge, Artificial Intelligence, knowledge*	***Key terms:*** *search algorithms, collection of items, breadth-first search, depth-first search*

Table II.5: Key terms extracted from two AI-related documents

In order to increase the strength of those terms whose keyphraseness value is low, we follow a frequency-based strategy taking into account the number of appearances (appearance frequency) of the synonyms of the concept (related to the candidate) in the document. Finally, we design a measure to establish an egalitarian relation between both keyphraseness and frequency of appearance values. Hence, the scoring function for a candidate term t and its corresponding concept c, can be defined by as:

$$multidomain\_score(c) = \sqrt{nFreq(c)^2 + nKeyphraseness(c)^2},$$

where $nFreq(c)$ denotes the normalized frequency of appearances on average of the synonyms of the concept c in the document, and $nKeyphraseness(c)$ denotes the normalized keyphraseness value of the concept c. Once all the candidates are measured, the top N candidates in the ranking are considered MKT for the given document.

2.5.4 Specific Candidates' Procedure

Similarly to the MCT set procedure, the system follows a raking algorithm to obtain the most meaningful SKT from the document. The ranking method is performed by the Specific Ranking module.

In this occasion, no filter is needed for the SCT set. The first filtering multi-domain process attends to the TF*IDF value of candidates. This filter is not essential for SKT, due to such set is specific for a single document instead of the entire collection. For the two remaining filtering processes, the language frequency filter is implicit in the selection of these candidates, and the dictionary filter is useless regarding the SKT have not to hold a semantically related concept. Thus, only the ranking procedure is needed.

In this work, we assume that the frequency in language is determinant to detect relevant terms in a bounded context. However, we cannot rank the SCT set using only their frequency in language value. Therefore, the frequency of appearance of each term in the document is taken into account:

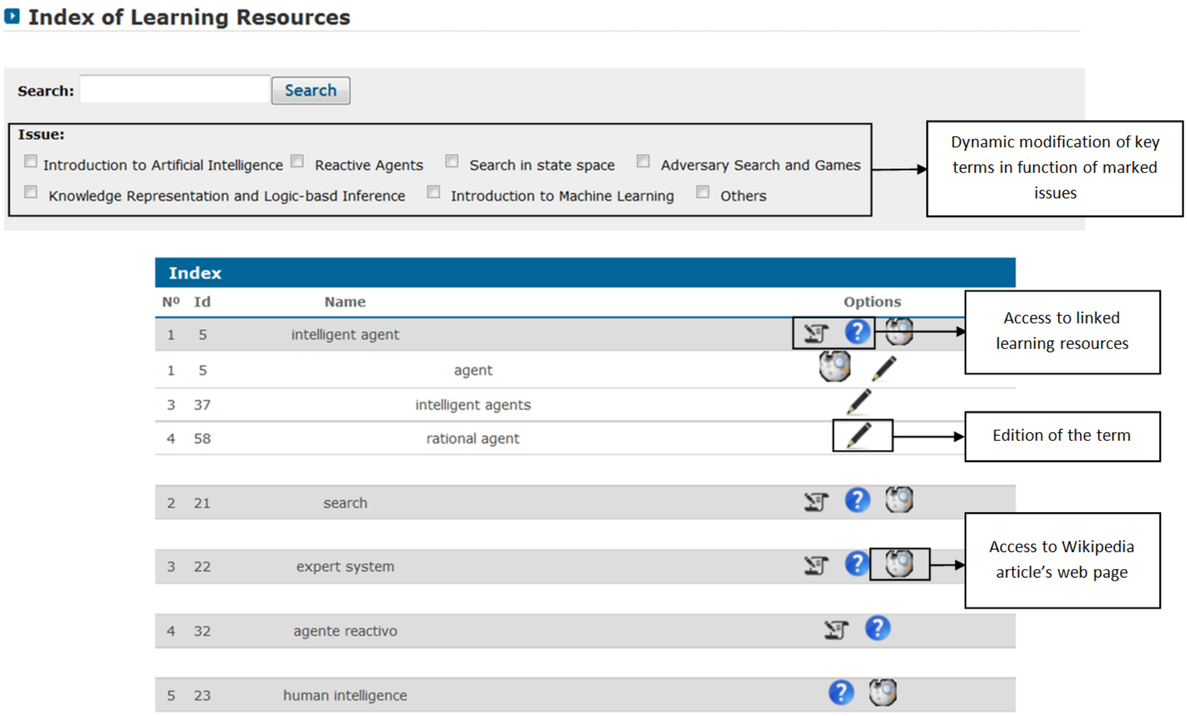
$$specific\_score(t) = \sqrt{nLangFreq(t)^2 + nFreq(t)^2},$$

where $nLangFreq(t)$ is the normalized frequency in language of the term t, and $nFreq(t)$ is the normalized frequency of the term t in the document. Finally, the top N specific candidate terms in the ranking are considered SKT for the given document.

2.5.5 Taxonomic Indexation of the Learning Resources

As result of the proposed system, we obtain a classification scheme of the learning resources mapped to the key terms that are stored in the controlled vocabulary of the domain, i.e. the flat taxonomy. In addition, the weights of each extracted key term can serve us as guide of the importance of the learning resources linked to each key term within the domain. Therefore the development of an indexation application is direct. We present a prototypical application for taxonomic indexation of learning resources in the following (see Figure ii.5).

The index structure contains the key terms contained in the taxonomy, ordered by their corresponding weights. Each key term is additionally enriched by the morphological variations, or synonyms detected in the AKE process. Hence the user can access the related learning resources simply by clicking in the corresponding icons next to each term. What is more, the application includes a search mechanism, facilitating the exploration of the index structure. In addition, the index is dynamically modified in function of the checked issues. Considering that the learning resources are catalogued into specific issues of the course (in case that the corresponding VLE would implement such functionality), the index is adapted to them when the user marks one or more checkboxes. Therefore the indexation application only displays the key terms obtained through the learning resources belonging to marked issues. Next, those key terms which were linked to Wikipedia articles before the multi-domain ranking module automatically adds a link to the Wikipedia article's web page. This is arguably a helpful resource for the student. Finally, the manager can modify each extracted term if it is needed.



Index of Learning Resources

Search: Search

Issue:

□ Introduction to Artificial Intelligence □ Reactive Agents □ Search in state space □ Adversary Search and Games

□ Knowledge Representation and Logic-basd Inference □ Introduction to Machine Learning □ Others

Dynamic modification of key terms in function of marked issues

Index

Nº	Id	Name	Options
1	5	intelligent agent	
1	5	agent	
3	37	intelligent agents	
4	58	rational agent	
2	21	search	
3	22	expert system	
4	32	agente reactivo	
5	23	human intelligence	

Access to linked learning resources

Edition of the term

Access to Wikipedia article's web page

Figure ii.5: Prototipical interface of a index application supported by the taxonomy

2.6 Experiences with the System

This section reports the empirical results obtained in the evaluation of our system. The validity of our system has also been contrasted to other state-of-the-art methods.

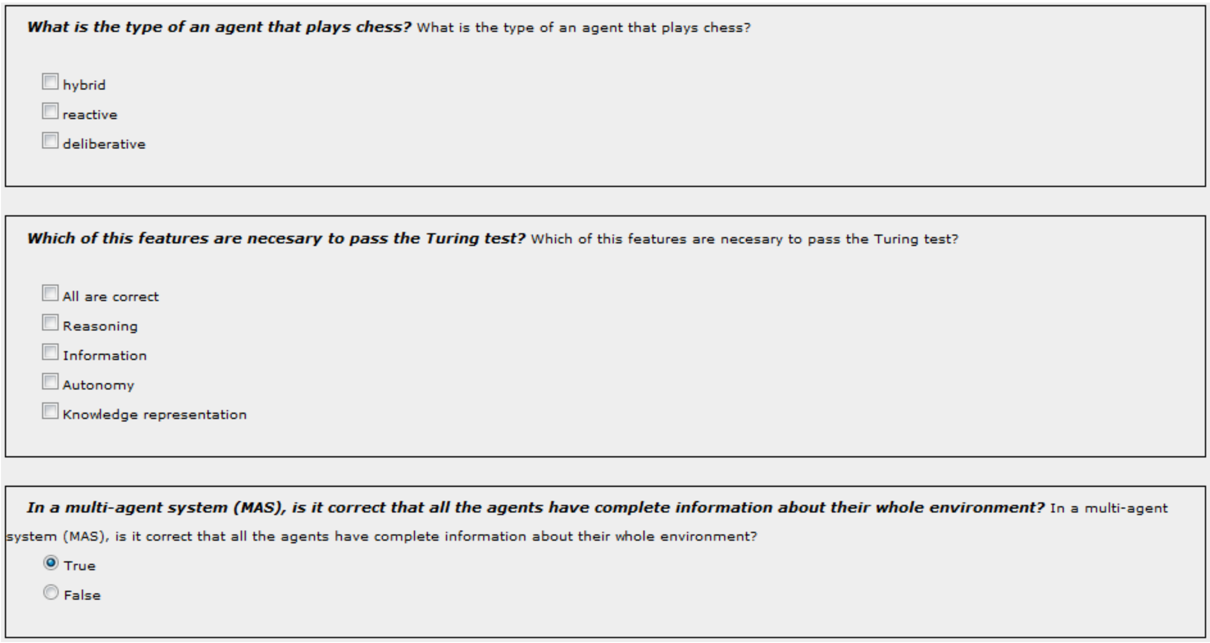
2.6.1 Type of Learning Resources

In order to validate the proposed approach in the e-learning context, we have tested the system's performance by applying it to a collection of learning resources. Unfortunately, there is no standard benchmark at our disposal. What is more, there exist several types of learning resources, each one with particular characteristics.

Therefore we have selected two different collection of learning resources created in the undergraduate course "Artificial Intelligence" in the Computer Engineering degree in the University of Granada. Such course follows a constructivist methodology, gathered in this Thesis (Chapter V Section 3.3), that supports the generation of learning resources by the proper students. Consequently, it could be regarded as reasonable approximation.

As first type of considered learning resources, we have selected quiz questions formulated in GIFT format. It supports Multiple-Choice, True-False, Short Answer, Matching and Numerical questions. Some examples of GIFT questions are presented in Figure ii.6.

The second learning resources involved in this research are a type question-answer documents, so-called collaborative learning tasks. A collaborative learning task is considered in this work as any request of information about a given topic formulated by a teacher. A learning task consists of an information request and the corresponding student's answer, both formulated in HTML. Any



What is the type of an agent that plays chess? What is the type of an agent that plays chess?

□ hybrid
□ reactive
□ deliberative

Which of this features are necesary to pass the Turing test? Which of this features are necesary to pass the Turing test?

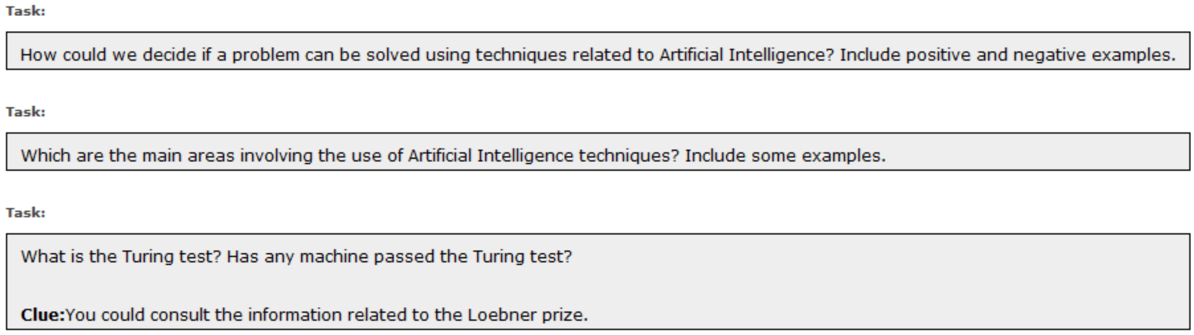
□ All are correct
□ Reasoning
□ Information
□ Autonomy
□ Knowledge representation

In a multi-agent system (MAS), is it correct that all the agents have complete information about their whole environment? In a multi-agent system (MAS), is it correct that all the agents have complete information about their whole environment?

● True
○ False

Figure ii.6: Example of GIFT questions

external resource can be added to the application enhancing the learning knowledge, as well as mathematics formulas, tables, or so on. In Figure ii.7 we show some examples of proposed tasks. It should be pointed out that we take into consideration uniquely the textual sources corresponding to the teacher's request of information and the student's answer, both formulated in HTML.



Task:

How could we decide if a problem can be solved using techniques related to Artificial Intelligence? Include positive and negative examples.

Task:

Which are the main areas involving the use of Artificial Intelligence techniques? Include some examples.

Task:

What is the Turing test? Has any machine passed the Turing test?

Clue: You could consult the information related to the Loebner prize.

Figure ii.7: Example of tasks

Further details of this two kind of learning resources are provided in Chapter V Section 3.2.3.

2.6.2 Data Set

We employ two different datasets of learning resources. One the one hand, the first collection is composed by 100 collaborative learning tasks. On the other hand, our second collection is composed by 100 quiz questions formulated in GIFT format. Key terms on each document were manually annotated by two PhD student of computer science and ten undergraduate students in the field of computer science. Each student was requested to extract a set of 3 to 20 key terms (including both specific and multi-domain key terms) from each document. Each document was analysed by four students. Next, we select as valid those key terms selected by at least two students. Finally, we obtain two data sets of 100 documents and their corresponding key terms. Table II.6 shows the proportion of key terms to each type. This percentage had been calculated *a posteriori*.

	Specific-domain Key terms	**Multi-domain Key terms**
question-answers (learnign tasks)	23,08%	76,92%
GIFT quiz questions	47,97%	52,03%

Table II.6: Percentage of key terms by type

2.6.3 Design Principles

Our system was implemented with the following architectural design principles. Both the dictionary of concepts and the frequency dictionary are stored in a MySQL database (version 14.12). We use the Hibernate[9] library (version 3.3.5), as framework for mapping the dictionaries.

Since Google does not provide the Google search API to obtain the number of results of a given query, we had implemented a specific module by means of the HttpComponents[10] library (4.0.1). This module automatically obtains the Google results page for any given input. Later, the number of results is extracted from the result page.

We use Java SDK 6 (1.6.0_20) to implement the system. Finally, a dedicated machine with 8Gb RAM was used for the evaluation.

2.6.4 Setting Parameters

There are a number of parameters involved in the main modules of our system. We comment in this section how we chose the values for them.

These parameters are related with the tolerance for each candidate set, acting as thresholds in the following situations: (1) a threshold α with a range of values between 0.01 and 1.0 is used to distinguish which 1-grams and 2-grams will be multi-domain candidates, or otherwise specific candidates; (2) a threshold β with a range of possible values between 0.01 and 1.0 is used to discards multi-domain candidates with a TF*IDF score lower than it; and (3) a threshold γ with a range of possible values between 0.01 and 1.0 is used to discards multi-domain candidates whose normalized frequency in the language is greater than it. Hence, these values are directly involved in the candidate extraction process giving to it more or less flexibility. Here we discuss how to tune α, β, and γ.

We selected ten question-answer documents with its corresponding manually annotated key terms of our collection firstly. Those terms of each document were manually grouped in two sets of key terms: multi-domain key term (MKT) set, if they were contained in the synonyms or the name of a concept in our dictionary; and specific-domain key term (SKT) set, in other case. To select the most appropriate thresholds for the three involved modules, we established different values for each one and took the resulting candidate set after passing the corresponding module. Then, we compared the appropriate manually annotated key term set with the systems' output set. Hence, we selected the tuning for each threshold that provided us the closest matching between the key terms

[9] http://www.hibernate.org/

[10] http://hc.apache.org/

sets and the corresponding output sets of the involved module. In more detail, the α-threshold was setted to the lowest value that divide the original set of n-grams in the following way. The MKT set should retain at least 70% of its members contained in the manually annotated MKT set, and the SKT set should retain at least 70% of its members contained in the manually annotated SKT set. The β-threshold was setted to the lowest value that allows the MKT set (after the TF-IDF filter) to retain at least 70% of its members contained in the manually annotated SKT set; the γ-threshold was setted to the greater value that allows the MkT set (after the Frequency filter) to retain at least 70% of its members contained in the manually annotated SKT set. The resulting values are $\alpha = 0.20$, $\beta = 0.30$, and $\gamma = 0.30$.

2.6.5 Experimental Results

In order to evaluate the performance of our system we carried out an experimental comparison of the proposal against conventional keyword extraction methods. Given that conventional methods were not designed to deal with learning resources, we selected the most used methods in the state-of-the-art for ordinary documents. As commented in Introduction, Machine Learning methods were not considered in the comparatives because of the nature of the considered resources. Five algorithms are used in this study:

- **Yahoo! Terms Extractor** is a web service[11]. This system provides a list of significant words or phrases extracted from an input content through a request URL. We use this application as baseline. The number of output key terms is not customizable.

- **TF*IDF** is a well-known unsupervised ranking method. This conventional algorithm establishes a ranking of candidate terms in basis of the TF*IDF measure. For obtaining the frequency values necessaries to apply the TF*IDF measure, the complete collections were used as training dataset. At last step, the top-K terms are selected as valid key terms. We select K equal to the number of key terms extracted by our system to make the comparison more illustrative. We could preselect the candidates for this ranking algorithm extracting it by our controlled dictionary. However, we decide not to do it since specific key term would not be considered.

- **Wikify!** is a knowledge base system that extracts key terms candidate from a Wikipedia based controlled vocabulary. This system ranks the candidates looking at the probability of a term being keyword in a Wikipedia article. In their work, the authors established that the number of key terms extracted by their system is around the 6% of the number of words in a document. Consequently, the system extracts a number of key terms equal to the specified ratio of words in each document.

- **TextRank** is a graph-based ranking algorithm from extracting key terms. This system models the input document as a graph being terms of a certain part-of-speech (nouns and adjectives) the vertices of the graph. An edge is added between those vertices whose respective terms co-occur within a window of N words. In their study the authors obtained the better results with a window of two words. Also, the number of key words is set to a third of the number of vertices in the graph. We follow these indications to apply their algorithm to our document collection. Finally, the system incorporates a post-processing stage. In this stage, compound key terms are obtained from combination of the previously founded key words.

[11] http://developer.yahoo.com/search/content/V1/termExtraction.html

- **Longest Common Substring** is a hybrid system compounds by the last two commented algorithms. For each element in the Wikify! output, the method finds the longest common subsequence in all of the TextRank output for the same document and vice versa. The final output is the union of the two set of substring that represents the longest fragments found in the key term set generated by both algorithms.

We evaluate each method by means of the F-measure metric (II.6). This metric is the weighted harmonic mean of precision (II.4) and recall (II.5), which are the most common metrics used in the literature. Given a set of key terms extracted by the system and a set of key terms manually extracted by humans, Precision is the number of matched key terms divided by the total number of key terms extracted by the system; Recall is the number of matched key terms divided by the total number of human extracted key terms. As could be seen, the results directly depend on the number of key terms extracted by our algorithm.

To choose the number N of key terms from each ranking, we conducted the following evaluation. The maximum number of manually annotated key terms in a given document from the both collections is 20. In the comparison, we have considered the cases of $N = 1$, to $N = 30$ subsequently, for each ranking. Then we have empirically contrasted that $N = 15$ was the best configuration in terms of F-measure for the FAQ collection and $N = 3$ for the quiz question collection. Therefore, we select the top-15 terms from the specific ranking and the top-15 terms from the multi-domain ranking (30 key terms for document) for the FAQ collection, and the top-3 terms from the specific ranking and the top-3 terms from the multi-domain ranking (6 key terms for document) for the quiz question collection.

$$Precission = \frac{|\{manually\ extracted\} \cap \{system\ extracted\}|}{|\{system\ extracted\}|} \tag{II.4}$$

$$Recall = \frac{|\{manually\ extracted\} \cap \{system\ extracted\}|}{|\{manually\ extracted\}|} \tag{II.5}$$

$$F - measure = 2 \cdot \frac{Precission \cdot Recall}{Precission + Recall} \tag{II.6}$$

Regarding the question-answer collection, the results presented in the Table II.7 shown that our ALKEx system outperforms all the comparative methods. TF*IDF obtained the second best scores. This situation confirms our earlier assumption that a great subset of key terms does not present a close relation between then and are hardly contained in multi-domain controlled vocabularies. In turn, Wikify! obtained the worst scores in the comparison. Although the controlled vocabulary obtains promising results handling ordinary documents, is not the case for learning resources. Moreover, the keyphraseness measure does not offer good results in this scenario. TextRank outperforms the controlled vocabulary strategy. This algorithm presents a candidate extraction process much more permissive allowing it to detect both technical terms and proper nouns. In this way, the algorithm is able to extract specific domain candidates, but its ranking method relies only on the co-occurring relation of the candidates whereas SKT are lacking of this kind of relation. Finally, using LCS provides a higher F-score than considering TextRank and Wikify! alone. This confirms that considering an hybrid strategy incurs in the improvement of the quality of the extracted key terms.

Following, Table II.8 shows the experimental results of key term extraction applied to quiz questions in GIFT format. Our system still exhibited the best performance although its behaviour

	Performance		
Method	Precission	Recall	F-measure
*TF*IDF*	19.21	49.40	27.09
Yahoo! Term Extractor	24.38	30.44	17.32
Wikify!	09.89	41.05	15.68
TextRank	13.43	62.45	21.24
LCS	17.68	36.66	22.38
ALKEx	**39.37**	**73.88**	**50.20**

Table II.7: Key term extraction results from question-answer learning task collection

was slightly modified. Given the short size of quiz questions, most of the human selected key terms were always detected by the system, thus improving the recall rate. In contrast, the precission rate was inferior. Regarding the rest of the algorithms, their performances were improved due to the same motive. TF*IDF algorithm showed in this case the worst performance. Wikify! obtained an improvement of 13.23% in it F-measure value. Although the quiz domain is still complex for a multi-domain controlled vocabulary, the system is able to obtain a more fitted candidate set due to the short nature of the document. TextRank results also experiment an important improvement in its performance (over 10.13%), outperforming the Wikify! system again. This is due to the word co-occurrence distribution always is pronounced if the texts present short size. Consequently, the improvement of Wikify! and TextRank systems leads a more successful assessment for the LCS hybrid strategy.

	Performance		
Method	Precission	Recall	F-measure
*TF*IDF*	22.67	41.25	28.78
Yahoo! Term Extractor	38.42	38.75	36.79
Wikify!	20.67	29.16	28.91
TextRank	23.53	48.83	31.37
LCS	**39.62**	26.85	32.00
ALKEx	35.28	**79.25**	**48.82**

Table II.8: Key term extraction results from GIFT quiz question collection

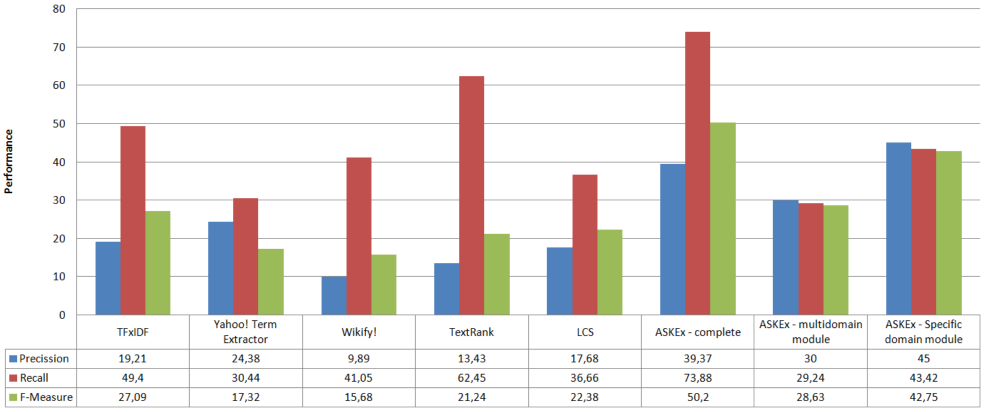
2.6.6 The Effect of Key Term Division on Learning Resource Dataset

In this section we study how the two different evaluation methods affect the performance of ALKEx. One of the specific novelties of our system is to consider two types of key terms besides the classical MKT set. We identify terms with low frequency in language to enrich the output of the system. Also, we consider those terms that are important in a multi-domain context. To evaluate the importance of key term division, we show the results obtained when only one the type of key terms is considered on our question-answer learning task dataset, i.e. when only one of the two modules of the system is executed (Table II.9). We have evaluated this feature only with the question-answer task collection because the short size of the GIFT collection makes the interpretation blurry. Short texts present low rate of key terms and therefore the number of MKT and SKT is not sufficient for analysing their individual effects.

	Performance		
Module	Precission	Recall	F-measure
ALKEx Specific-domain Key Term Module	45.00	43.42	42.75
ALKEx Multi-domain Key Term Module	30.00	29.24	28.63

Table II.9: Key term extraction results of each independent module of ALKEx in question-answer task collection

For each type of candidate, we select the top 15 terms of the ranking. We found that the performance is deteriorated if only one type of key terms is chosen. However, the results evidence that even considering only one candidate set, the system improves the results obtained by the comparison algorithms (Figure ii.8).



	TFxIDF	Yahoo! Term Extractor	Wikify!	TextRank	LCS	ASKEx - complete	ASKEx - multidomain module	ASKEx - Specific domain module
🟦 Precission	19,21	24,38	9,89	13,43	17,68	39,37	30	45
🟥 Recall	49,4	30,44	41,05	62,45	36,66	73,88	29,24	43,42
🟩 F-Measure	27,09	17,32	15,68	21,24	22,38	50,2	28,63	42,75

Figure ii.8: Performance evaluation extended by ALKEx modules

On the one hand, the multi-domain approach outperforms the results obtained by the rest of the systems except TF*IDF algorithm. Attending to the system Wikify!, the results showed that the addition of the redirect information of the Wikipedia articles enriches the controlled vocabulary aiming to improve the candidate detection. Moreover, the adapted keyphraseness works better in specific document context. On the other hand, the module devoted to detect SKT still exhibits the best performance among all other algorithms. Technical terms and proper nouns are crucial in specific document context. Furthermore, it has been proved that combining both strategies incurs in a better performance.

2.7 Conclusions and Future Work

As part of the organization schemes needed to overcome the information overload in scenarios with large volume of documents, we have presented in this section a method able to automatically obtain the taxonomic representation of a collection of learning resources. Thanks to the features of taxonomies, the educational content can be organized in effective way. What is more, taxonomies allow to automatically obtain index structures from the educational content, which favour the search of information. Flat taxonomies perfectly fit this demand, and are a suited model for indexing of learning resources.

Bearing that in mind, we have focused on the AKE field. The key terms extracted from the learning resources by an AKE method could be arranged together with links to the content, making a flat taxonomy. Unfortunately, classical AKE methods do not deal with the inner particularities of the e-learning context. In this kind of documents, we identify two types of key terms: “Multi-domain key terms”, meaningful in a multi-domain context; and “Specific domain key terms”, proper nouns or technical terms closely related to specific aspects of the document topic. Although classical approaches could handle the identification of multi-domain key terms, it is less possible that they could effectively identify specific domain terms. For this reason, we have designed a system able to recognize two different key term sets. Composed by an hybrid strategy, it takes advantage of the strengths of the frequency-based and thesaurus-based criteria, introducing some novelties. On the one hand, it identifies the main multi-domain important terms enclosing general key terms of the collection. In order to identify them, a dictionary of concepts drawn from Wikipedia is used. This dictionary represents a wide-ranging controlled vocabulary taking an active part in most of the main modules of the ALKEx system. Moreover, we extended the so-called keyphraseness method to rank multi-domain candidate terms adding a document-specific feature (following a frequency-based strategy). On the other hand, the system extracts specific key terms highly related with the context of e-learning, since specific-domain key terms are usually technical terms or proper noun. They are uncommon in the language so we use the frequency in language to detect such terms. The system has obtained promising results in experimental validation, being compared with some of the most important algorithms in Keyword Extraction task. Nevertheless, AKE methods present a high rate of imprecision, as it could be observed in the experimental evaluation. This effect is may provoke negative effects if the results are not revised by the teacher.

The system’s architecture is highly modularized and can be easily adapted to other systems or languages. Every module of our method is easily extensible to other systems. Moreover, both the dictionary of *concepts* and the frequency-in-language dictionary can be extended in many languages. In addition, the system could be extended to any kind of learning resource with minor modification.

Notwithstanding, there is work ahead. First of all, the taxonomy extracted by this method does not consider hierarchical classes. Although it is sufficient to index the learning resources, a flat taxonomy is insufficient for other tasks. For example, we could consider the hierarchical organization of the learning resources classified by categories. In this sense, we plan to apply the categories stored in our Wikipedia-based dictionary as hyponyms. This task will require a prior analysis and filtering process of the category structure of Wikipedia. In second place, our disambiguation strategy could be improved by adding further considerations. Although the term disambiguation algorithm is important for the performance of our system, we have adopted a heuristic strategy. Thus, a deeper analysis of this module would be desirable. On a different matter, our dictionary of concepts contains a well-formed hierarchy of categories. The use of such hierarchy could enhance the quality of certain parts of our system, as is the case of the ranking modules or the term disambiguation module.

3 TRLearn System: Creating a Conceptually Extended Folksonomy from Scratch

3.1 Introduction

The emergence of the Web 2.0 has fostered new ways of characterizing digital educational resources, which moves from the expert-based descriptions relying on formal classification systems to a less formal user-based tagging [DHMPP11]. The process of *tagging* or *social tagging* has become a popular strategy to improve the organization, description, browsing and retrieval of web objects, including educational resources. Tagging process could be defined as a text annotation method for labelling web resources with free-from user-defined keywords or keyphrases, the so-called *tags*, that represent the topics of each resource. Users are free to choose any tag in order to label their resources subjectively, allowing an effective categorization of the web objects. The set of tags used by the user community for labelling a set of resources forms a taxonomy of tags, known as *folksonomy*[WPS10, JMH$^+$07].

Social tagging of educational resources has received increasing attention since educational resources are not meant to be used only by their creators, but ideally to be re-used in different contexts. Community users produce tags that are significantly different from formal expert meta-data, contributing to a richer vocabulary [ZS14]. This improves the searchability of educational resources, since users find their searches based on social tags more useful for them compared to their searches with formal meta-data [VO09]. The applicability of social tagging in allowing students to share resources is clear, with recent recommendations on its usage by Franklin and Van Harmelen [FVH$^+$07] and Lapham [Lap07].

Nevertheless, the simplicity of social tagging entails both strengths and drawbacks. On the one hand, folksonomies are easy to create due to the users do not need any special skill to tag [HJSS06]. In addition, folksonomies are able to be rapidly adapted to continuous changes concerning terminology and expansion of the domain. That is, folksonomies evolve at the same time as the vocabulary of the involved community [GSCAGP12, GH06]. On the other hand, the lack of semantics and relations among tags provoke important disadvantages [GSCAGP12, GH06, LDGS$^+$13]. First, it is common that different users employ different morphological variations of a same term (plural, acronyms, misspelling...) for referring to a same entity (e.g. *UK*, or *United Kingdom*). Similarly, a resource might be named using different synonyms of a same term (e.g. *holidays* or *vacations*). Third, the multiple meanings of a same term, i.e. polysemy, may confuse the user about what entity is being referenced (e.g. *wood* could be a piece of a tree or a geographical area with many trees). Finally, it is probably that different users tag a same resource with different level of abstraction (e.g. *celebration* in contrast to *wedding*).

The development of Tag Recommendation (TR) systems has lighten the semantic unification of the tags. TR systems assist the user looking for the subset of tags from the complete tag space which better fits with the current resource. In any case, the final election falls on the user, who can freely select one of them or add a different one. The usual scheme is depicted in Figure ii.9. Once a user post a resource, the system analyses information related to that resource, to the whole set of resources or to the tags already selected by other users in order to produce a list of recommended tags. Then, the user freely chooses the tags to annotate it. The resource and the selected tags are linked in the folksonomy.

Taking into consideration the capabilities of TR systems, it is difficult to understand why their use is not more widespread in e-learning platforms. Even e-learning related works consider the TR

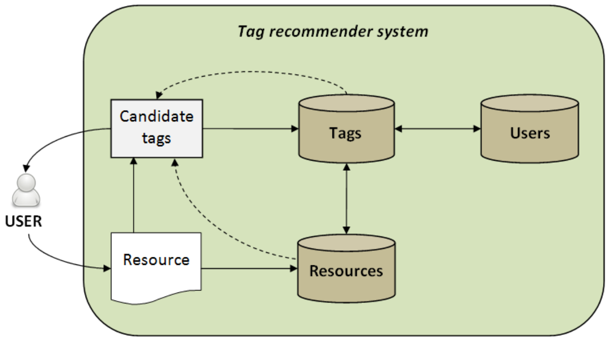


Figure ii.9: Tag recommendation scheme

problem in a generalist manner [ARS+11, SLA13] without focusing in the educational particularities. Attending to the literature, there exist three main approaches for TR (the reader can find a review of the state-of-the-art in [MNI10]). Firstly, most of current systems employ a collaborative approach, using the previously added tag space in order to recommend similar tags to similar users or items [LN13, SVZ08, AM14, ZZT09]. Unfortunately, this is a strong limitation during the initial phases of a new system (or in a new domain) since the initial process of defining the tags of a domain is very time-consuming. Secondly, content-based approaches only consider the textual content of the resource, in order to extract those explicit terms from the content that better represent the resource [PDB+10, MFW09]. However, this scheme reduces the problem to an information extraction problem (e.g. automatic keyword extraction task), without having into account the community actuation. Thirdly, ontological approaches employ external structured ontologies as controlled vocabularies of tags [MW13, SDMVM06, Pas07, PDB+10]. This kind of approaches adds great strength to the process of representing the information. Thanks to the properties of ontologies, the systems are able to detect concepts in the resources instead of simple tags. Unluckily, if there are not predefined ontologies about the domain, the construction process is far from a trivial task (as it will be explained in Section 4). As it can be seen, the common TR approaches present a set of limitations regarding the educational context. Particularly, the main problem comes with the necessity of prior knowledge. The construction of the prior knowledge would be responsibility of the teacher, being this an error-prone and time-consuming task. In addition, the scope of application is limited due to users must present technological knowledge on the field.

The above commented reasons have motivated us to design a TR system able to create a multi-domain conceptually extended folksonomy from scratch, without the necessity of a prior knowledge. The system can be applied in any learning domain. Moreover, the multi-domain conceptually extended folksonomy obtained will represent the knowledge about the domain, which is learnt for the system in a collaborative approach. According to [PFM06], we focus on multi-terms that are more descriptive than single terms in order to identify the content of the sources. In order to avoid the necessity of a prior tag space or the construction of a domain ontology, we employ a combination of a multi-domain dictionary (Section 2.3) and a set of heuristic rules to detect conceptual tags. The dictionary of concepts is used to allow the identification of the semantic of terms, instead of being used as controlled vocabularies. Additionally, the heuristic rules enhance the process allowing to extract meaningful terms that might not be taken into account in the dictionary. Finalizing, the actuation of the community is also taken into account in order to adapt the importance of those candidate tags detected by the system.

More concretely, in this part of the dissertation we present the TRLearn (Tag Recommendation for Learning resources) system. The system uses an hybrid approach to detect an initial set of candidate tags from the content of each learning resource, by means of syntactic, semantic, and frequency features of the terms. After that, each time a candidate tag is selected to label a resource, the system modifies the weights of the rest of candidates in function of syntactic and semantic relations existing among candidate tags. Furthermore, this step is configurable, allowing to choose from a totally automatic construction to a social collaborative construction of the folksonomy. Once the system counts with a set of selected tags, the folksonomy can be used to explore the educational content of the course, for example by means of a tag cloud representation, common in social tagging scenarios (see Chapter III Section 3).

The rest of the section is organized as follows: Subsection 3.2 reviews the state-of-the-art on tagging systems. After that, the architecture of the proposed system is commented in Subsection 3.3. Then, we offer a validation of our methodology in Subsection 3.4. Finally, Subsection 3.5 includes the conclusions.

3.2 Related Works

In this section, we review the most relevant previous works on tagging applications that aim to help in the process of building folksonomies.

One of first work in the field was carried out in [GH06]. The authors studied the information dynamics of Delicious, discussing how users employ same tags over time and how these tags tend to stabilize over time. Additionally, the found two semantic difficulties: tag redundancy, when multiple tags have the same meaning, and tag ambiguity, when a single tag has multiple meanings. Concerning tag recommender systems, we can find the following scheme in literature. Given a resource and a user, the tag recommender analyses the underlying information related to the resource or the previous actuation of the user for retrieving a ordered list of candidate tags. Subsequently, the user freely chooses the tags that are more suitable for labelling the resource. Therefore, we can establish a classification for the different approaches attending to the technique employed to analyse the resources and/or the user previous actuation. We distinguish three main types of approaches: content-based, graph-based approaches and hybrid approaches.

First, content-based tag recommenders detect tags directly from the content of the resource, having the assumption that the resource has at least one textual attribute. Hence, theses systems assign a score to each candidate term in the text, or classify each term as a tag or not based on a set of features of the candidate. As it can be seen, the task is considered as an automatic keyword extraction problem [PDB$^+$10, MFW09, SLP14]. To that end, frequency-based, graph-based, or dictionary-based approaches can be selected, in both supervised and unsupervised methodologies. [ASRB07] developed a tag recommender for documents based on Semantic Web ontologies. They employ three Web 2.0 services (Tagthe.net, Yahoo's Term Extraction and Topicalizer) to extract relevant key phrases from the texts. Finally, the system matches the extracted key phrases to instances of a predefined ontology. In [BM06], authors present a system for recommending tags from blog post. To that end, it extracts a weighted list of candidate terms from the text using the TF*IDF score. The top-3 candidate terms are suggested to the user. In [LC07], artificial neural networks are trained with lexical and statistical features of terms, with the aim to extract and recommend tags from blogs. Next, a supervised learning strategy is considered in [LYCH09]. Features related to the content of the document and to previous available tag information are exploited to train a machine-learning classifier. [LXCH09] evaluates a method that uses a statistical machine translation approach to learn the translation probability from words in document to tags. The proposal recommends candidate tags from the texts using maximum likelihood and statistical machine translation model. The work in [PDB$^+$10] also follows a classification scheme to extract and recommend tags from web resources. Additionally, the presented system matches the obtained tags with a predefined ontology in order to spread the result. The same strategy is developed in [SCP12], where a topic ontology is constructed using Wikipedia and Wordnet. The main idea behind the topic ontology is to detect the common topics that may are interested from blog users. Finalizing, a supervised learning framework for extracting tags from short entries of social media services is depicted in [SLP14]. The proposed system use stream level features of text as training features of a SVM classifier. [MW13] describes an algorithm to frequent tags from clinical trial texts. The algorithm builds all the possible n-grams (from 1-grams to 10-grams), having into account some restrictions. After that, only those candidates containing at least one substring of a UMLS concept are retained (the n-grams are matched against the UMLS Metathesaurus). The remaining tags are used as a controlled vocabulary to index clinical texts. Finally, other work concerning the content-based analysis for images or photos can be consulted in [DJLW08].

Second, graph-based approaches exploit the co-occurrence relations among users, resources, and tags represented in a folksonomy graph. The main idea behind is that users with similar behaviour tend to choose similar tags and similar resources tend to have similar tags as labels. In most cases, graph-based tag recommendation systems are implemented with collaborative filtering methods. One of first works in the field was presented in [Mis06]. Recommendations are based on the tags posted in similar weblogs. For a given resource, the system takes first the similar posts in the collection. Then, the tags linked to those posts are merged, filtered and re-ranked. The system suggests the top-ranked candidate tags. [JMH$^+$07] present FolkRank, an adaptation of the classical PageRank algorithm for a folksnomy graph. The algorithm process the folksnomy graph, and the vertex are mutually reinforced by themselves by spreading their weights. Tags with the highest weights are returned as recommendations. In [GH06], authors propose a user-based collaborative filtering approach and state that users with similar tag vocabularies tend to tag alike. The work in [Sym09] used Latent Semantic Analysis to model the relations among users, resources, and tags by means of 3-dimensional matrices. Their method reveals latent relations among objects of the same type, as well among objects of different types. In addition, they provide user recommendations. The system proposed in [GSRM09] adapt the K-Nearest Neighbor algorithm for tag recommendation. In this work, neighbours are selected attending to previous tagging of the same resources with the same tags. The proposed modifications dramatically reduced the computational costs. Next, authors propose two document-centred approaches designed for working on large-scale data sets in [SZG11]. First, a graph-based method represents the tagged data into two bipartite graphs of document-tag and document-word, and finds document topics by leveraging graph partitioning algorithms. Second, a prototype-based method is designed to find the most representative documents within the data collections and employs a multi-class Gaussian classifier for document classification. Recommendations are performed by classifying a new document into one or more topic classes, and then recommending the most relevant tags from them.

Finally, hybrid approaches are designed to combine the advantages of content-based and graph-based approaches. The system proposed in [TSD08] recommend tags extracted from documents and user models. A number of natural language processing techniques are considered to extract tags from the resources. Then, they are merged with content-based tags. Next, the work presented in [JH09] uses a simple weighting scheme for combining different information sources and a candidate filtering method for tag recommendation. The system matches a set of candidate terms extracted from the resource with previously tagged documents to evaluate the likelihood of a candidate being used as a tag. These candidates are later linearly combined with tags from resource and user profiles. In [GRS$^+$09], a system based uniquely on the information extracted from the folksonomy graph is studied. The system is exploit six recommendation models, including the most frequent tags from resource and user profiles as well as four collaborative filtering methods for comparing the similarity between users and resources. More recent, [LM11] propose a system built with five recommender strategies: tags extracted from a resource title, tags extracted through resource and user profiles, and tags extracted from the title that are later extended by an activation algorithm using title-to-tag or tag-to-tag co-occurrence graphs. The candidate tags are then merged and re-scored. The values of the merging coefficients are learned based on real tags from user feedback. Authors in [LDGS+13] exploit and combine three techniques. First, the textual metadata of the resources is analysed in order to extract content-based tags. Second, the previous tagging activity of individual users is analysed in order to assign higher weights to tags employed in similar resources. Third, this last technique is considered having into account the tagging activity of the whole community.

3.3 System Overview

In this section, we present an overview of TRLearn for extracting and recommending conceptual tags from a set of educational resources. The system can perform from an automatic way (without any supervision) to a social scheme (users are in charge of select which tags are or are not related to the corresponding resources). The set of selected tags are employed to generate a conceptually extended folksonomy that could be subsequently used to improve the navigation through the educational content. Figure ii.10 depicts the proposed system architecture.

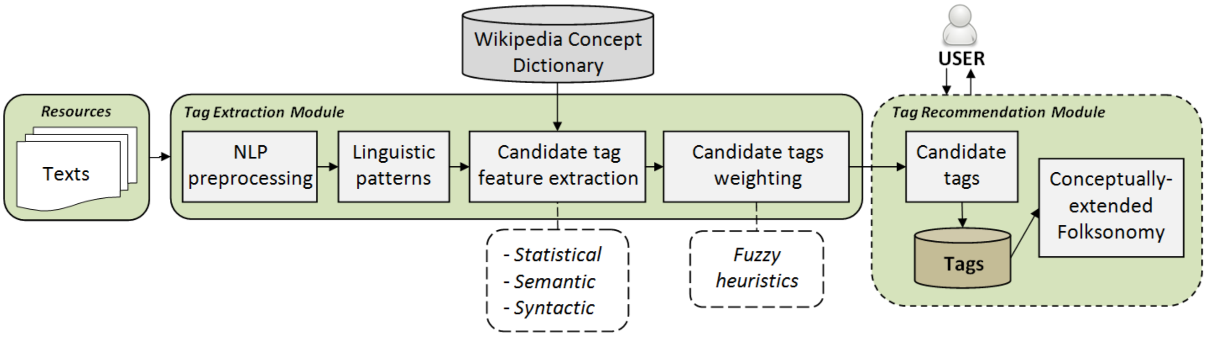


Figure ii.10: TRLearn architecture overview

The information extraction process is the following. First, the most relevant candidate tags are found from the textual fields of the available set of resources, and a weight is assigned to each one in function of statistical and semantic features of candidates. Although some features are extracted considering the whole dataset, the system performs at document level. As part of the process, candidates are automatically linked to the resources, and to an additional knowledge base (Wikipedia articles). By means of the Wikipedia-based dictionary of concepts (Section 2.3) and some natural language processing techniques, we are able to handle the morphological variations and synonymy of the terms. After that, if TRLearn works automatically, the top-k candidates with higher weight are included as valid tags. Despite the fact that user validation is not mandatory, the automatic extraction of noisy tags may provoke the same negative effects that we are trying to prevent with this methodology (lose of focus, disorientation problems...). Therefore, if the social/collaborative scheme is chosen, the candidate tags are presented to the user in order to validate them. In the last step, a feedback weighting process is carried out. When a candidate is validated, the weights of the rest of candidates are adapted in function of some syntactic and semantic features. Summarizing, the quality of the results depends on two different processes (tag extraction, and tag recommendation) that are carried out by two main modules in TRLearn: the tag extraction module, and the tag recommendation module.

3.3.1 Tag Extraction Module

During the first stage, the tag extraction module works to obtain a weighted list of candidate tags, related to the set of available resources.

The candidate tags are extracted (and linked) from the textual fields of documents in the dataset. Considering a given document of the dataset, the tag extraction module employs syntactic patterns to detect the meaningful terms in the sentences. The part-of-speech (POS) [MS99] of each word is obtained, and 1-grams, 2-grams, 3-grams and 4-grams from the texts are identified as candidate tags using the patterns showed in Table II.10. The GPL Library FreeLing 3.0[12] was used to convert sentences to lower case, and to obtain the POS tags of each one. After that, the candidates are checked through LanguageTool[13], an Open Source proof-reading software for analysing and detecting misspellings in texts. It supports several languages and there is a Java API available.

1-gram	**2-gram**	**3-gram**	**4-gram**
N	N + N A + N	N + N + N A + N + N A + A + N N + SP + N	N + 3-gram pattern A + 3-gram pattern N + SP + 2-gram pattern

Table II.10: Linguistic patterns used to recognize candidate concepts

Subsequently, the single candidate terms are expanded to candidate conceptual tags: both morphological variations and synonyms of the same term are joint together in a set of terms representing the tag. We use the Levenshtein distance to detect morphological variations of the same word. Also, a simple heuristic is able to detect abbreviations checking if the first letters of the multi-word terms match with the single-word candidates. Next, candidates are searched in the Wikipedia-based dictionary of concepts. If there is at least one concept that matches with the term, the synonyms are added to the candidate. To that end, the candidate term should coincide with one of the different synonyms of the concept.

If there are two or more concepts matching with the term, the system applies the term disambiguation process explained in Section 2.5.3. That is, considering the document as disambiguation context, the cosine similarity measure is computed using the vector representation of it and the vector representation of the synonyms of each possible concept. The concept with highest value in regard to the document is taken. It should be pointed out that the different synonyms added to a candidate tag are used only to calculate the strength of such tag. That is, only one term defining the tag will be recommended to the user.

Afterwards, a domain-independent and efficient heuristic strategy weights the candidate tags. First of all, the module extracts the following statistical and semantic features for each candidate c_i:

$wikiConcept_{c_i}$ Boolean variable. *true* if c_i exactly matched with the name of a Wikipedia concept or its synonyms in the previous step.

$partialWikiConcept_{c_i}$ Boolean variable. *true* if c_i matches partially with the name of a Wikipedia concept or its synonyms, if any concept was linked to it in the previous step[14].

[12] http://nlp.lsi.upc.edu/freeling/

[13] https://languagetool.org/

[14] The synonyms of the concept are not added to the candidate if $wikiConcept_{c_i}$ is *false*

$docFreq_{c_i}$ Relative frequency of c_i in the document set.

$$docFreq_{c_i} = \frac{numDocs(c_i)}{|N|} \cdot 100 \qquad \text{(II.7)}$$

where $numDocs(c_i)$ obtains the number of documents containing c_i, and $|N|$ is the total number of resources. To calculate the frequency, the algorithm considers all the terms associated to c_i: misspellings, morphological variations, and synonyms if it is the case.

$corpusFreq_{c_i}$ Relative frequency in a multi-domain corpus of c_i. The frequency is calculated considering the number of Wikipedia articles in which the candidate appears. Each term of the candidate is matched against the content of all the Wikipedia articles. Only the higher value is kept for the whole candidate.

$$corpusFreq_{c_i} = \frac{wikiFreq(c_i)}{|W|} \cdot 100 \qquad \text{(II.8)}$$

where $wikiFreq(c_i)$ obtains the number of Wikipedia articles containing c_i, and $|W|$ is the total number of Wikipedia articles.

$numWords_{c_i}$ Number of words of c_i.

To establish a weight w_i for a candidate c_i, a number of heuristic rules are applied (Table II.11). As it can be seem, two of the proposed rules are designed using fuzzy variables. The election of fuzzy sets [Zad65] as tool to measure the degree of importance of the frequency features of the candidates was made attending to the variation about the frequency of a same term in different domains or in different datasets. Fuzzy set theory allows to quantify the non-stochastic uncertainty induced from subjectivity, vagueness and imprecision [ZN09a]. Accordingly, TRLearn performs dynamically in a multi-domain context. The functionality of the proposed algorithm is explained in the following.

Firstly, the presence of the candidate in the dictionary of concepts is assessed. Wikipedia is a proven well-built dictionary, storing a long number of articles of all kind of topics. An exact match between the candidate and any article's title (or its synonyms) guarantees that the term has enough importance to be considered in a multi-domain scenario. Therefore, an increase of $INCR\_W1$ is added to the tag's weight. Additionally, if the whole term is partially contained in any concept, a lower increment ($INCR\_W2$) is added to its weight.

Secondly, the level of specificity of a term is taken into account. As more words contains a single term, more descriptive is it. This is due to additional words tend to add additional information. For example, the term 'network' for itself is generic enough to be a tag in many domains. However, if we consider 'neural network', or moreover 'artificial neural network', the possible domains are greatly bounded. Hence, the weights for multi-word candidates are slightly increased in function of the number of words that it contains ($INCR\_E2$ to $INCR\_E4$).

Thirdly, we handle the frequency in the documents of candidate tags. The module measures the membership degree of c_i with respect to a fuzzy variable that defines the level of intensity of the relative document frequency. The fuzzy variable is modelled with three fuzzy sets, attending to the level in which a term has low frequency (LDF), high frequency (HDF) and very high frequency ($VHDF$) with respect to the document dataset. The fuzzy sets take into account the mean value of the relative document frequencies ($mean_df$), the mean deviation value of the relative document frequencies (md_df) and the maximum value of relative document frequencies (max_f) considering all the candidates (Figure ii.11). More formally, we defined a fuzzy variable named “intensity of

```
INPUT:
    Set of features of $c_i$

OUTPUT:
    $w_i$: weight representing the importance of $c_i$ as tag

begin
    $w_i \leftarrow 0$
    //Rule 1: presence in dictionary
    if $wikiConcept_{c_i} = true$ then
        $w_i \leftarrow w_i + INCR\_W1$
    else if $partialWikiConcept_{c_i} = true$ then
        $w_i \leftarrow w_i + INCR\_W2$
    end if

    //Rule 2: level of specificity
    if $numWords_{c_i} = 2$ then
        $w_i \leftarrow w_i + INCR\_E2$
    else if $numWords_{c_i} = 3$ then
        $w_i \leftarrow w_i + INCR\_E3$
    else if $numWords_{c_i} = 4$ then
        $w_i \leftarrow w_i + INCR\_E4$
    end if

    //Rule 3: document frequency features
    $w_i \leftarrow w_i + \mu_{LDF}(docFreq_{c_i}) \cdot INCR\_DF$
    $w_i \leftarrow w_i + \mu_{HDF}(docFreq_{c_i}) \cdot INCR\_DF$
    $w_i \leftarrow w_i + \mu_{VHDF}(docFreq_{c_i}) \cdot INCR\_VHDF$
    if $docFreq_{c_i} \in top - nFrequencies$ then
        $w_i \leftarrow w_i + INCR\_TOPDF$
    end if

    //Rule 4: corpus frequency features
    if $corpusFreq_{c_i} > 0.0$ then
        if $numWords_{c_i} = 1$ then
            $w_i \leftarrow w_i + \mu_{LCF_1}(corpusFreq_{c_i}) \cdot INCR\_CF$
            $w_i \leftarrow w_i + \mu_{HCF_1}(corpusFreq_{c_i}) \cdot INCR\_CF$
        else if $numWords_{c_i} = 2$ then
            $w_i \leftarrow w_i + \mu_{LCF_2}(corpusFreq_{c_i}) \cdot INCR\_CF$
            $w_i \leftarrow w_i + \mu_{HCF_2}(corpusFreq_{c_i}) \cdot INCR\_CF$
        else if $numWords_{c_i} = 3$ then
            $w_i \leftarrow w_i + \mu_{LCF_3}(corpusFreq_{c_i}) \cdot INCR\_CF$
            $w_i \leftarrow w_i + \mu_{HCF_3}(corpusFreq_{c_i}) \cdot INCR\_CF$
        else if $numWords_{c_i} = 4$ then
            $w_i \leftarrow w_i + \mu_{LCF_4}(corpusFreq_{c_i}) \cdot INCR\_CF$
            $w_i \leftarrow w_i + \mu_{HCF_4}(corpusFreq_{c_i}) \cdot INCR\_CF$
    else if
            $w_i \leftarrow w_i - DECR\_NCF$
    end if
end
```

Table II.11: Tag weighting algorithm

relative document frequency", with three sets of linguistic values *T = {LDF, HDF, VHDF}*, *U = [0,100]* and the following membership functions:

$$\mu_{LDF}(x) = \begin{cases} 1, & 0 \leq x < x_1 \\ \frac{x_2 - x}{x_2 - x_1}, & x_1 \leq x < x_2 \\ 0, & x \geq x_2 \end{cases} \tag{II.9}$$

$$\mu_{HDF}(x) = \begin{cases} 0, & x < x_2 \\ \frac{x - x_2}{x_3 - x_2}, & x_2 \leq x < x_3 \\ 1, & x \geq x_3 \end{cases} \tag{II.10}$$

$$\mu_{VHDF}(x) = \begin{cases} 0, & x < x_3 \\ \frac{x - x_3}{x_4 - x_3}, & x_3 \leq x < x_4 \\ 1, & x \geq x_4 \end{cases} \tag{II.11}$$

, where $x_1 = mean_{df} - md_{df}$, $x_2 = mean_{df}$, $x_3 = mean_{df} + md_{df}$, and $x_4 = (max_{df} - (mean_{df} + md_{df}))/2.0$.

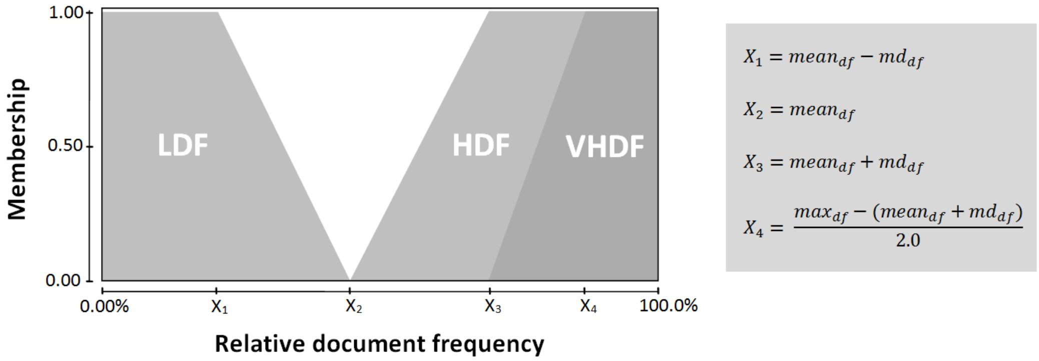


Figure ii.11: "Intensity of relative document frequency" fuzzy variable

Subsequently, the membership functions $\mu_{LDF}(docFreq_{c_i})$, and $\mu_{HDF}(docFreq_{c_i})$ are computed obtaining three values ranged between [0,1]. Each membership value is then multiplied by the factor $INCR\_DF$. Besides, the membership function $\mu_{VHDF}(docFreq_{c_i}$ is computed and the resulting value is multiplied by $INCR\_VHDF$. The increment factor $INCR\_VHDF$ is slightly superior to $INCR\_DF$ (following classical Information Retrieval statements, we consider that terms occurring in the majority of documents tend to be important in the dataset context). In addition, the top-*nFreq* candidates with higher relative document frequency (those whose frequency is included in the set $top - nFrequencies$) suffer an additional increment of $INCR\_TOPDF$.

Finally, the algorithm computes the frequency of the candidate within an external multi-domain corpus (Wikipedia). Similarly to the previous step, a set of fuzzy variables measuring the depth of use of a term in the language (or in a multi-domain corpus) is modelled. Due to we cannot equally compute the frequency of a 1-gram and a 4-gram (since specific terms tend to occur less than generic terms), we model four fuzzy variables reflecting low frequency (LCF_1 to LCF_4) or high frequency

(HCF_1 to HCF_4) in the corpus, taking into account only 1-grams, 2-grams, 3-grams or 4-grams in each case. The fuzzy variable for 1-grams is built considering the mean value of the relative frequencies of the 1-gram candidates ($mean_{cf\_1}$), and the mean deviation value of the relative frequencies of all the 1-grams candidates (md_{cf_1}) (Figure ii.12). More formally, we defined a fuzzy variable named (1-gram)", "intensity of relative corpus frequency with two sets of linguistic values *T = {*LCF_1*,* HCF_1*}*, *U = [0,100]* and the following membership functions:

$$\mu_{LCF_1}(x) = \begin{cases} 1, & 0 \leq x < y_1 \\ \frac{y_2 - x}{y_2 - y_1}, & y_1 \leq x < y_2 \\ 0, & x \geq y_2 \end{cases} \tag{II.12}$$

$$\mu_{HCF_1}(x) = \begin{cases} 0, & x < y_2 \\ \frac{x - y_2}{y_3 - y_2}, & y_2 \leq x < y_3 \\ 1, & x \geq y_3 \end{cases} \tag{II.13}$$

, where $y_1 = mean_{cf_1} - md_{cf_1}$, and $y_2 = mean_{cf_1}$, $y_3 = mean_{cf_1} + md_{cf_1}$.

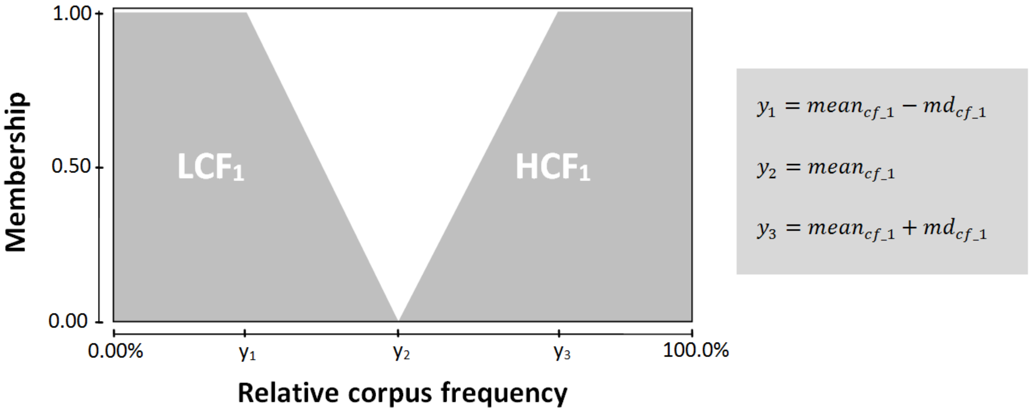


Figure ii.12: "intensity of relative corpus frequency" fuzzy variable for 1-grams

The rest of fuzzy variables for 2-grams, 3-grams and 4-grams follow the same model. The $corpusFreq_{c_i}$ acts as input of the two membership functions for low or high frequency of the corresponding fuzzy variable[15]). The obtained value ranged between [0,1] is multiplied by $INCR\_CF$, and it is added to w_i. In addition, if the candidate does not occur in any source of the external corpus, the weight is decreased in $DECR\_NCF$. Although a less frequent term tends to be more important in specific domains, it is unlikely that it does not occur in the whole content of Wikipedia. In fact, during our work we found that most terms that did not occur in Wikipedia were noisy words that the preprocessor was not able to handle.

Concluding, the system computes a weight for each candidate tag and produces an ordered list. It is sent to the tag recommendation module. If the candidate is linked to a concept, the system chooses the concept's title as representative for such candidate. If it is not the case, the term of the candidate with higher frequency in the given document is selected.

[15] Since a candidate tag can store various synonyms, all terms are evaluated independently, and the maximum resulting value is kept.

We show an example of the detection and weighting of the candidate tags from two short documents related to the semantic web in Table II.12. In the example, the parameters related to the increments of weight are the same that those specified in Section 3.4.3. To clarify the reading only 2 synonyms are added to each candidate tag. The candidate tags are ordered by weight and they present their representative title in bold. Considering the length of the documents, no more than 4 tags would be displayed to the user for each document. As it can be seen, the candidate tags recommended to the user are quite similar. The algorithm is able to detect same concepts although the terms employed to refer to them are different in each document. Besides, other concepts not contained in the dictionary are also detected (i.e., “web ontology” in the example).

Finalizing, when a new document is included in the dataset, the process is the same. If the document contains terms previously included in the tag space, they are detected by comparing the different synonyms of the conceptual tags. Then, only the relative document frequency feature should be calculated. The relative document frequency for the candidate also considers the documents referenced by the valid tag. If this tag is selected as valid by the user, the new document is linked to the valid tag previously accepted in the tag space, and the weight of the previously validated weight is modified now considering the new relative document frequency. That is to say, the weights of the tags always have into account the whole domain. As more documents are added, the weights of the tags already validated that are contained in the new documents are modified. In contrast, if new tags are detected, they are included as new in the tag space.

3.3.2 Tag Recommendation Module

The tag recommendation module has two main goals. First, it presents the selected tags to the user. If TRLearn applies the collaborative scheme, the tags are displayed as candidates, and the user has the responsibility to choose the most appropriate. Figure ii.13 shows the interface containing the list of candidate tags extracted in the previous example. In contrast, if the process is automatic, the top-n candidates are displayed as valid tags. In any case, when a tag is picked as valid, both the tag and the resource linked to it are included in the conceptually extended folksonomy. Additionally, the module adapts the weights of the rest of the candidates in function of syntactic and semantic features.

The algorithm in charge of adapting the weight of a candidate when a tag is selected as valid is depicted in Table II.13. The algorithm operates with some of the features calculated in the previous stage. Given a selected tag st and a candidate tag ct_i, four different rules are checked in order to adapt the weight of the candidate.

First of all, a rule measuring the relation of st and ct_i within the Wikipedia corpus is considered. In the previous stage, each candidate tag was linked to the set of Wikipedia articles whose title exactly matched with it. The candidate ct_i is searched in the content of the articles linked to st. For example, if we consider the candidate tag “semantic web” extracted in the previous example, the terms “RDF” or “OWL” are contained in the Wikipedia article corresponding to it. If the search is successful, the relative corpus frequency of ct_i (calculated in the previous stage) is used as input of the fuzzy linguistic variables “intensity of relative corpus frequency”. Following the prior scheme, the length of the synonyms of ct_i is checked in order to employ the corresponding fuzzy variable for 1-grams, 2-grams, 3-grams or 4-grams. Then, $corpusFreq_{c_i}$ acts as input of the membership function for low frequency of the corresponding fuzzy variable. Since it might be possible that ct_i contains more than one term, the maximum value is kept. The resulting value ranged between [0,1] is afterwards multiplied by $INCR\_REL\_CORPUS$, and it is added to aw_i.

The second rule checks if the candidate tag is a syntactic specification of the selected tag. Given

Document$_1$:
The semantic web was developed to handle the common disadvantages proper of the WWW.
It involves publishing in any language specifically designed for data: RDF, or OWL.

Document$_2$:
The semantic web makes use of languages such as Resource Description Framework, or Web Ontology Language,
in order to improve the interpretation of web documents and perform intelligent processes for
capturing and managing the information.

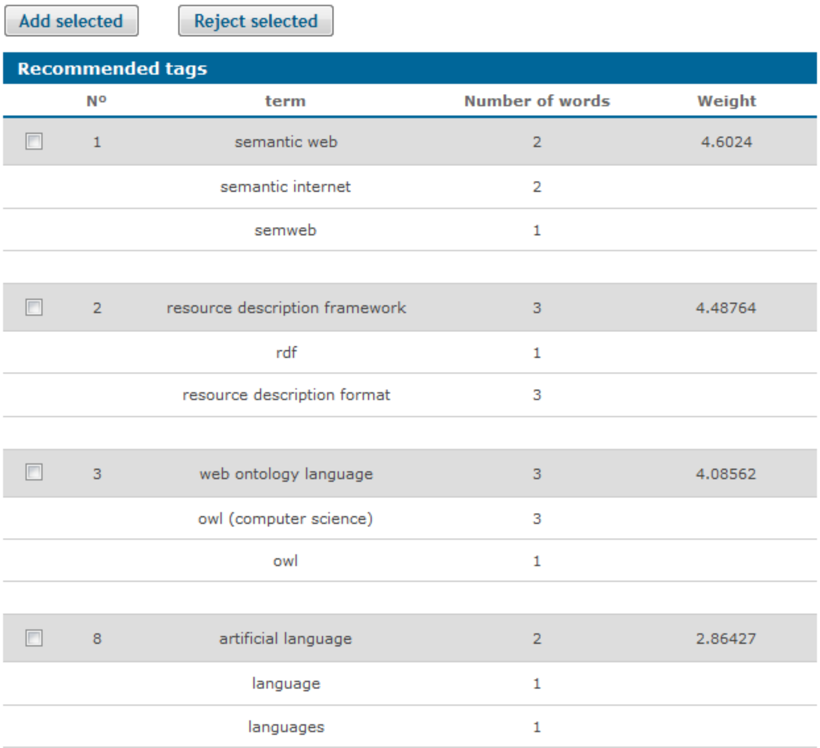
Candidate tags	*wikiConcept*	*partialWikiConcept*	*docFreq*	*corpusFreq*	*numWords*	*weight*
{**semantic web**, semantic internet, semweb}	Y	N	100%	0.04%	2	4.60
{**resource description framework**, rdf, resource description format}	Y	N	100%	0.17%	1	4.48
{**web ontology language**, owl (computer science), owl}	Y	N	100%	0.01%	3	4.08
{**ontology language**, ontology specification language, ontology representation}	Y	N	50%	0.04%	2	3.36
{**web ontology**}	N	Y	50%	0.02%	2	3.06
{**artificial language**, language, languages}	Y	N	100%	16.14%	1	2.86
{**description framework**}	N	Y	50%	0.008%	2	2.05
{**framework**, frameworks}	Y	N	50%	0.004%	1	1.67
{**ontology**, ontological, ontologic}	Y	N	50%	0.70%	1	1.64
{**resource**, resources}	Y	N	50%	1.21%	2	1.62
{**interpretation**, interpretations}	Y	N	50%	4.24%	1	1.50
{**web**, webs}	Y	N	100%	39.12%	1	1.50
{**document**, documents, documenting}	Y	N	50%	4.69%	1	1.48
{**process**, processes, processing}	Y	N	50%	5.48%	1	1.45
{**web document**, web documents}	Y	N	50%	1.19%	2	1.44
{**data (computing)**, data, computer data}	Y	N	50%	6.59%	1	1.41
{**information**, informative subject, informative}	Y	N	50%	8.32%	1	1.34
{**intelligent processes**}	N	N	50%	0.15%	2	0.72
{**world wide web**, www, the web}	Y	N	50%	68.31%	1	0.67
{**description**, descriptions}	Y	N	50%	29.01%	1	0.67
{**disadvantages**}	N	Y	50%	0.36%	1	0.65
{**interpretation of web documents**}	N	N	50%	0%	4	0.57
{**common disadvantages**}	N	N	50%	0.13%	2	0.55
{**resource description**}	N	Y	50%	0.32%	2	0.33

Output:

List for *document*$_1$:
semantic web, resource description framework, web ontology language, artificial language

List for *document*$_2$:
semantic web, resource description framework, web ontology language, web ontology

Table II.12: Example of candidate extraction and weighting process



Recommended tags

	Nº	term	Number of words	Weight
□	1	semantic web	2	4.6024
		semantic internet	2	
		semweb	1	
□	2	resource description framework	3	4.48764
		rdf	1	
		resource description format	3	
□	3	web ontology language	3	4.08562
		owl (computer science)	3	
		owl	1	
□	8	artificial language	2	2.86427
		language	1	
		languages	1	

Figure ii.13: Ordered list of candidate tags presented to the user

the lists of terms of ct_i and st, the algorithm checks if any of the terms of st is a substring of any of the terms of ct_i. For example, given a selected tag “neural network”, the candidate tag “artificial neural network” would satisfy the rule. In that case, the factor $INCR\_REL\_SPC$ is added to aw_i.

Similarly, the third rule tests if any of the terms of ct_i is a syntactic generalization of any of the terms of st. Following the previous example, a tag “neural network” is a generalization of “artificial neural network”. Thence, if any term of ct_i is a substring of a term of st, the factor $INCR\_REL\_GEN$ is added to aw_i.

The final rule also operates with the terms of ct_i and st. The set of single words belonging to each term of ct_i is compared against the set of single words belonging to each term of st. If there exists a term of ct_i and a term of st sharing a same word, both ct_i and st are considered similar (for example, “fuzzy logic” and “fuzzy variable” tags). It should be pointed out that stop words are removed from the terms. If the rule is satisfied, the weight of the candidate is increased by the factor $INCR\_REL\_SIM$. The process is performed for every candidate tag.

3.4 Experiences with the System

In this section, we discuss the quality of our proposal in basis of a set of experimental tests. The performance of the tag extraction module is evaluated by applying our algorithm on a dataset of educational resources. Finally, we examine the impressions of the users involved in this experiment about the proposed TRLearn.

```
INPUT:
    $st$: selected tag
    $ct_i$: $ith$ candidate tag
    Set of features of $ct_i$ obtained in the previous stage
    $w_i$: weight of $ct_i$

OUTPUT:
    $aw_i$: adapted weights of $ct_i$

begin
    $aw_i \leftarrow w_i$
    //Rule 1: relation by corpus
    $wikiArticles_{st} \leftarrow obtainWikipediaArticles(st)$
    if $ct_i$ in $wikiArticles_{st}$ then
        if $numWords_{ct_i} = 1$ then
            $aw_i \leftarrow aw_i + \mu_{LCF_1}(corpusFreq_{ct_i}) \cdot INCR\_REL\_CORPUS$
        else if $numWords_{ct_i} = 2$ then
            $aw_i \leftarrow aw_i + \mu_{LCF_2}(corpusFreq_{ct_i}) \cdot INCR\_REL\_CORPUS$
        else if $numWords_{c_i} = 3$ then
            $aw_i \leftarrow aw_i + \mu_{LCF_3}(corpusFreq_{ct_i}) \cdot INCR\_REL\_CORPUS$
        else if $numWords_{c_i} = 4$ then
            $aw_i \leftarrow aw_i + \mu_{LCF_4}(corpusFreq_{ct_i}) \cdot INCR\_REL\_CORPUS$
        end if
    end if

    //Rule 2: relation by specification
    if $\exists\ term_{st} \in st.termList(), \exists\ term_{ct_i} \in ct_i.termList() \mid term_{st}$ is substring of $term_{ct_i}$ then
        $aw_iaw_i \leftarrow aw_i + INCR\_REL\_SPC$
    end if

    //Rule 3: relation by generalization
    if $\exists\ term_{ct_i} \in ct_i.termList(), \exists\ term_{st} \in st.termList() \mid term_{ct_i}$ is substring of $term_{st}$ then
        $aw_i \leftarrow aw_i + INCR\_REL\_GEN$
    end if

    //Rule 4: relation by similarity
    if $\exists\ term_{ct_i} \in ct_i.termList(), \exists\ term_{st} \in cs.termList() \mid term_{ct_i}.ngramSet() \cap term_{st}.ngramSet() \neq \phi$ then
        $aw_i \leftarrow aw_i + INCR\_REL\_SIM$
    end if
end
```

Table II.13: Weight adaptation algorithm after successful tag selection

3.4.1 Dataset

The evaluation process has been carried out over a dataset of learning resources belonging to the undergraduate course "Artificial Intelligence" in the Computer Engineering degree in the University of Granada. We employ the same type of learning resources used for the AKE method validation: quiz questions and learning tasks (Section 2.6.1). These learning resources are collaboratively generated by the students, following a constructivist approach. Consequently, this is a suitable scenario to test out TR method due to the great amount of created resources and the presence of a user community.

We have randomly selected 100 quiz questions and 100 learning tasks from the complete collection. In order to set a suited evaluation scenario, we have not considered each type of learning resources independently. Current collaborative web environments do not make distinctions between the type of supported resources. This framework differs considerably from common evaluation datasets which focus only in specific types of documents. Thus that decision has allowed us to prove the quality of our approach on a real-life platform.

3.4.2 Design principles

TRLearn was implemented with the following architectural design principles. We use a MySQL database (version 14.12), and the Hibernate[16] library (version 3.3.5), as framework for mapping the dictionary of concepts. We use Java SDK 6 (1.6.0_20) to implement the system. Finally, an Intel(R) Core(TM) i5 processor 430M (2.26Ghz) and 4GB RAM was used for the evaluation.

3.4.3 Setting parameters

There are a number of parameters involved in the main modules of our system. We comment in this section how we chose the values for them.

The parameters involved in the tag extraction module concern the increments added or subtracted to the candidate's weight when the different rules are satisfied. Concretely, we count with 8 increment parameters and one single decrement parameter. To set the correct value for these parameters we conducted some preliminary experiments. We chose different values for the different parameters, all of them in the range [0,1], and we observed its impact in the results. We tested the quality of the results attending to the F-measure metric (see Eq. (II.6)) used in the experimental evaluation of TRLearn. We observed that the increment parameters (and the single decrement parameter) should be balanced in order to allow a good distinction of candidates in function of their weight. Higher values led to an extreme division of the candidates, augmenting the precission (see Eq. (II.4)) but decreasing too much the weights of the rest of candidates. That is, if the recommended tags were rejected, the weights of the rest of candidates were too low to be significant in the domain. In contrast, lower values led to the opposite situation, having too many noisy candidates in the recommended list. Therefore, the values showed in Table II.14 were empirically verified as the most appropriate balance between precission (it varied about 13% while ranging the parameters) and the real usability of the system.

After that, we carried out a similar experiment concerning the parameters involved in the weighting adaptation algorithm. The chosen parameters are depicted in Table II.15.

[16] http://www.hibernate.org/

Rule	Parameter name	Value
Rule 1	$INCR\_W1$	1.0
	$INCR\_W2$	0.2
Rule 2	$INCR\_E2$	0.2
	$INCR\_E3$	0.3
	$INCR\_E4$	0.4
Rule 3	$INCR\_DF$	0.2
	$INCR\_VHDF$	0.3
	$INCR\_TOPDF$	0.5
Rule 4	$INCR\_CF$	0.5
	$DECR\_NCF$	1.0

Table II.14: Proportion of accepted resources by issue

Rule	Parameter name	Value
Rule 1	$INCR\_REL\_CORPUS$	0.4
Rule 2	$INCR\_REL\_SPC$	0.3
Rule 3	$INCR\_REL\_GEN$	0.1
Rule 4	$INCR\_REL\_SIM$	0.1

Table II.15: Proportion of accepted resources by issue

3.4.4 Experimental evaluation

As main experiment, we carried out an evaluation of the procedure in charge of the generation of the conceptually extended folksonomy from scratch. It should be remembered that TRLearn do not operate with a prior tag space or controlled vocabulary. This fact has hindered a comparison with other tag recommendation proposals. In addition, the strength of our system relies on the collaborative path established among the users. For these reasons, we have mainly focused the evaluation on the manual side. Therefore, it is composed by two different phases: an evaluation about the performance and an evaluation about the usability of the system.

Initially, all resources have been tagged by one domain expert (the teacher in charge of the course) in order to define a relevant human baseline. Each resource was tagged with 4 to 7 tags. Then, the manually-selected tags for each resource were compared against the tags recommended by the two comparison methods. Since the list of tags recommended by our system is dynamically modified, we consider the first element of the list at each iteration. That is, given a document the system proposes an ordered list of tags and the first one is selected as valid. Then, the system adapts the weights of the rest of tags and a new list is shown to the user. At that point, the first tag is again selected. The process iterates as many times as the number of desired tags.

The quality of the results is tested by means of the F-measure metric (II.6) (Section 2.6). Remember that this metric is computed as the weighted harmonic mean of precission (II.4) and recall (II.5). Given a set of tags extracted by the compared methods and a set of tags manually extracted by humans, precission is the number of matched tags divided by the total number of tags extracted by the system; and recall is the number of matched tags divided by the total number of human extracted tags.

As could be seen, the results directly depend on the number of tags extracted by our system. To choose the number N of tags, we conducted the following evaluation. The average number of manually annotated tags for a document from the collection is 5.8. In the comparison, we have considered the cases of $N = 5$, to $N = 10$. Then we have empirically contrasted that $N = 5$ was the best configuration in terms of F-measure. In consequence, in this experiment we consider the top-5 tags for each resource. We consider the top-5 terms extracted by the TF*IDF method as well. It should be taken into consideration that this decision provoked that the maximum recall rate obtained by the two methods was always equal or lower than 0.86.

In the first phase of the evaluation, the performance of the tag recommendation algorithm is compared against the unsupervised ranking method TF*IDF [WS98] as baseline. In Table II.16, the average values for the commented metrics are shown. As it could be seen, our method reaches a good level of performance (F-measure = 62.07%), with a relatively high precission (73.34%) and a lower recall (54.00%). This is a expected results since the number of recommended tags is lower than the average number of manually annotated tags. Nevertheless, the tag recommendation algorithm gives priority to obtaining few suitable tags instead of obtaining a large number of tags. The results are good enough considering that the tags are detected without any prior controlled vocabulary. The method allows to detect common concepts in any domain, but also to detect specific concepts that would probably be discarded attending only to a controlled vocabulary. In contrast, the baseline method obtained a worse performance (F-measure = 35.68%). The main problem of this algorithm is that it extracts single terms. Even if the TF*IDF method extracted all the words of a manually annotated multi-term (we considered that tag as valid in that case), there were a lot of single words that do not carry any meaning by themselves. For example, we found that the term 'learning' was mostly extracted in resources related to 'automatic learning'. The problem is that 'automatic' term was not recommended in some cases and the term 'learning' by itself was not accurate enough to be considered as valid. Notwithstanding, most of tag recommender systems operates in this manner, losing specificity in complex scenarios.

	Performance		
Method	%Precission	%Recall[17]	%F-measure
TRLearn	73.34	54.00	62.07
TF*IDF	43.33	30.33	35.68

Table II.16: Evaluation of recommended tags for the course resources

In the second phase, the usability of our system was contrasted by the students of the "Artificial Intelligence" course from 2012/2013 year. Two different groups were selected from them. 25 volunteer students composed the first group. Their objective was to utilize our tool to extract up to 5 tags for a subset of 10 documents from the dataset. In this regard, the system was stored in an university server under Apache Tomcat 6 (6.0.32) to manage the data storage and was accessible via web application. During the progress of the experiment, a technical assistant was available to assure the correct operation of the system and to manage the technical issues reported by the users. All participants were instructed with a one-hour specific lesson about the use of TRLearn. After that, we applied a common model of information technology in order to measure the level of satisfaction about TRLearn system. Hence, they could perform an usability questionnaire to check their opinions about our method. In contrast, 23 volunteer students in the second group were requested to manually detect up to 5 tags for the same documents. The results and duration of the two processes were then compared.

Students in first group accepted 38 tags on average for the ten documents (that is, 3.8 tags per resource). This fact supposes an acceptance rate of 76% of the recommended tags. 11 different tags were selected considering all the documents (Table II.18). Students spent 2 minutes and 16 seconds on average to complete the activity. After completing the tag selection task, they were requested to respond a questionnaire composed by 5 items based on a five-point Likert scale regarding to the usability of the system in terms of recommended tags. The scale ranges from strongly agree (5 points) to strongly disagree (1 point).

We summarize the mean and standard deviation of the scores assigned by students to the 5 items in Table II.17. As it could be observed, the rates are significantly high, being near to 4 points in all cases. Students were satisfied with the system, and they thought that it was intuitive and easy-to-use. The second item concerning the quality of the recommended tags obtained a rate of 3.68, showing good behaviour of the tag recommendation module. The rates for the third and fourth items also indicate that the documents of the course were well represented and referenced in the folksonomy. Concluding, the last item points out that the students considered the tag cloud a useful tool to explore the content of the domain.

Item	Students (N = 25)	
	M	*SD*
1. The system is intuitive and easy-to-use	4.08	1.11
2. The recommended tags are related enough to the content of the documents of course	3.68	0.98
3. The folksonomy representation summarizes the content of the domain of the course	3.84	0.85
4. The resources of the course are correctly referenced by the tags in the folksonomy	3.48	1.15
5. The domain folksonomy would help me for searching information about the domain	4.08	0.81

Table II.17: Results of the 5 items of the satisfaction questionnaire

In regard to the tags manually obtained by the students in the second group, they generated 35 tags on average for the 10 documents. Therefore, each document was labelled by 3.5 tags on average. 13 different tags were selected considering all the documents (Table II.18). Students expended 5 minutes and 44 seconds on average for completing the activity. Attending to the manually extracted tags, we could found some of the common problems commented in the introduction. First, some of the terms present a lower level of abstraction than those selected trough TRLearn (for example, the tag 'agent' was selected instead of 'intelligent agent'). Second, different students chose different synonyms to refer the same object (e.g., 'ai' and 'artificial intelligence'). Finally, we have detected that students tended to use only the explicit proper nouns present in the texts, even if they are not the most related concepts with the document (for example, the tag 'sensor' was selected by 10 students).

3.5 Conclusions

In this work, we have presented TRLearn, a tag recommender system able to create a conceptually extended folksonomy from scratch. As second meta-data structure viewed in this chapter, folksonomies provide with a organization of the resources following a collaborative approach. The resources are tagged by the user of a community, which can assign any tag in a free form to a given resource. In this context, TR systems help the users in such task and could be arguably valuable in a e-learning platform with large volume of resources.

Group 1	Group 2
	reactive agents
reactive agent	deliberative agents
deliberative agent	agent
intelligent agent	artificial intelligence
artificial intelligence	ai
game theory	iconic representation
agent architecture	agent architecture
models of intelligence	path
subsumption architecture	gps
applications of artificial intelligence	subsumption architecture
agent based model	sensor
global positioning system	games
	models of knowledge

Table II.18: Tag selected by the two groups

Unfortunately, the use of TR systems is not widespread in e-learning platforms. Attending to classical approaches, they present two major limitations regarding the e-learning scenario. Most of them are multi-domain and/or depend on prior knowledge for its application. It may limit the scope of application of the system due to users must present technological knowledge on the field. In addition, the construction of the prior knowledge would be responsibility of the teacher, being this an error-prone and time-consuming task.

The above commented reasons have motivated us to design a TR system operating in any domain and without the necessity of prior knowledge. Simultaneously to the tag recommender functionality, the system obtains the knowledge about the domain in a collaborative approach. The conceptually extended folksonomy can be seen used to obtain a “conceptual” cloud representation of the domain. The extracted tags are a set of multi-terms representing concepts of the knowledge. To that end, we designed an hybrid approach to detect an initial set of candidate tags from the content of each resource, by means of syntactic, semantic, and frequency features of the terms. Additionally, the system adapts the weights of the rest of candidates after a tag is selected, in function of syntactic and semantic relations existing among them.

Experiments evidence that our system obtain a good performance, being tested on a set of learning resources which belongs to a higher education course. Additionally, a satisfaction questionnaire was fulfilled by a set of users, revealing that they felt comfortable with the system, and that it was useful to extract tags and to represent the content of the domain.

4 ORLearn: Recommending Concept and Relations for Learning Ontologies from Educational Resources

4.1 Introduction

So far, we have faced two different information extraction methods for extracting and organizing the knowledge from the educational content: AKE for building taxonomies and TR for building folksonomies. The next natural step consists of analysing intelligent methods for building domain ontologies.

The appearance of the Web 2.0 paradigm has provoked an important change of educational focus. Today all the participants in the learning process can easily create and share resources. The educational content is produced not only by teachers, but also by students that should participate actively in the creation of content as important part of the learning process. Hence the adoption of VLEs has encouraged a new vision of the learning process that is now considered a combination of interaction and content consumption. The teacher takes the role of supervisor, becoming more a facilitator of the learning process than a producer of content. Therefore, content management clearly becomes one of the most important services that the virtual learning environment must deal with [MSL11].

Within this context, ontologies appear to be one of the most adopted solutions to organize the educational content, since they foster interoperability between human and computer, and enhance the knowledge representation, sharing and reusing [LM04]. In this regard, ontologies have been proved to be a valuable tool for personalization of learning environments [KKD14], for adaptive multimedia presentation [CDNR09], for the adaptive management and reuse of e-learning resources in distributed environments [Siv14], for content search in known repositories of learning materials [DRMLOA08], for representing the data managed by e-assessment environments [RGC$^+$13], and so on.

We shall remember that an *ontology* refers to a formal specification of conceptualization [Gru93]. In this sense, a *domain ontology* is a type of ontology which is used to represent to represent the knowledge for a particular application domain [DBM04]. It can be described be defining a set of objects and describable relationship among them that are reflected in the vocabulary that represents knowledge. Additionally, we can distinguish between lightweight ontologies and heavy weight ontologies [Chi07]. A *lightweight ontology* takes the simple form of a taxonomy of concepts. In contrast, a *heavy weight ontology* comprises a taxonomy as well as the axioms and constraints which characterize some prominent features of the real world.

Nonetheless, the manual construction of ontologies is a tedious and cumbersome task, since it consumes both time and resources [RMVGFB$^+$11]. Maedche and Staad stated that manual building of an ontology results in long and tedious development stages and becomes a knowledge acquisition bottleneck, even if it is performed by knowledge engineers and domain experts [MS01]. What is more, this task is expensive, error-prone, biased towards its developer, inflexible and specific to the purpose that motivated its construction [MS01, SAB03, HEBR11].

In order to solve these issues, researches have focused on Ontology Learning (OL) mechanisms by using semi-automatic of automatic methods for building ontologies. OL consists of a set of methods that allow building from scratch, enriching or adapting an existing ontology in a semi-automatic fashion, using heterogeneous information sources [Sán09]. Cakula and Salem [CS13] explained the main benefits of OL: (1) to facilitate the sharing the common understanding of the structure of information among people or software agents; (2) to enable the reuse of domain knowledge; (3) to make domain assumptions explicit; (4) to separate domain knowledge from operational knowledge; and (5) to analyse domain knowledge.

OL from text sources usually involves three major construction processes as minimum: document preprocess, concepts extraction, and concept relations explorations [GPMM04]. Additionally, there are other construction steps that augment the complexity of ontologies such as attributes extraction, instances extraction, and axioms extraction [ZN10]. In this work, we mainly focus on the three first steps. Document processing consists of filtering out noisy terms in documents; concept extraction consists of extracting domain concepts out of vocabulary; and concept relations exploration consists of mining relations between concepts and organizing them to finish the ontology construction process. In accordance with Shih et al. [SCCC11], concept relations exploration is the most important process of ontology learning because relations between concepts and the ways concepts are organized by their relations influence the ontology structure, winch in turn affects the accuracy of the domain knowledge.

Most of approaches on OL considers taxonomic relations between concepts, i.e. generalization/specification relationships, and non-taxonomic relations including part-of relations, synonymy relations, possession relations, or causality relations [SCCC11, BR12]. However, although these approaches are quite useful, they do not consider specific educational-related relationships between concepts which might help students to acquire an schematic view of the importance and dependences of each topic along the course content. For example, which concept should be first studied in order to understand another, or which concepts are a subtopics of others. The reader may think that this could be performed only with taxonomic or part-of relationships but a concept does not have to be a subclass of another so that it should be studied first. To understand it, let see the following example. The concept "intelligent agent" is a main topic of an AI course, having the subtopics "types of intelligent agents" or "examples of intelligent agents". These two last concepts are not subclasses of 'intelligent agent" nor present a part-of relation with it.

The above reasons have encouraged us to develop an semi-automatic method for building domain ontologies for the educational field. We focus on lightweight ontologies for two main reasons. In first place, the method will be included as part of a VLE where the teacher is usually the manager of the system. Teachers do not necessarily have to be knowledge engineers nor posses AI knowledge (one purpose of a VLE is to be accessible for different domains), and therefore they are typically unaware of ontology existence and its relevancy altogether [HGS+12]. Thus a heavyweight ontology may fall outside their scope. In second place, the extraction of lightweight ontologies presents a computational complexity much lower than the extraction of the heavyweight ones. In this sense, the extraction of the conceptual structure of the educational content can be achieved with a lightweight domain ontology.

Our construction method follows the common OL scheme considering three construction steps: document preprocess, concepts extraction, and concept relations explorations. The concept extraction and concept relations extraction steps are tackled by means of recommendation more than automatic extraction. To that end, we take advantage of the tag recommendation method commented in the former section where each tag represents a complete concept instead of a single term. However, we remove now the collaborative path. The teacher should be now the only responsible for validating each element of the ontology, since he/she is the expert about the field. Students will be novice in the field, and they will require accurate structures which facilitate their learning processes. Regarding the concept relations exploration, we employ a similar recommendation strategy by presenting the automatically identified relations between concepts to the teacher. More concretely, our system detects taxonomic relations between concepts (superclass relation, and subclass relation) and educational specific relations (subtopic relation, subordinate relation, and relation through the course content).

To that end, we have designed a hybrid system, so-called ORLearn (Ontology-based Recommendation from Learning resources), based on heuristics that combines the strength of Natural Language Processing (NLP) techniques with statistical methods. Our contribution here is twofold. On the one hand, it alleviates the bottleneck provoked by the manual knowledge acquisition process. On the other hand, it provides a meta-data knowledge structure that focuses on the educational domain. In addition, the method allows an easy and intuitive interaction with the teacher in charge of the VLE. What is more, this procedure will make easier the production of visual representations of the domain, e.g. concept maps, which favour the navigation through the learning resources and support the acquisition of the competencies of the course.

The rest of the section is organized as follows: Subsection 4.2 reviews the state-of-the-art on OL mechanisms. After that, the architecture of the proposed approach is commented in Subsection 4.3. Then, we offer a validation of our method in Subsection 4.4. Finally, Subsection 4.5 includes the conclusions and future work.

4.2 Related Works

In this section, we review research on ontology learning from textual sources. We will first emphasize in the type of sources used by the learning process. After that, we will discuss the ontology learning approaches. Finally, we will comment a number of complete systems that have been proposed for OL from texts. It is important to remark that is not a goal of this Thesis to provide a detailed overview of the field due to its enormous extension. Here we proceed to offer a rough outline of OL. We refer the reader to a number of surveys presented in the literature [ZN10, DG08, HEBR11, LHC11, BR12] for a more detailed specification of the different approaches and tools. We have relied on these surveys in order to provide this overview.

Ontologies can be learnt from unstructured, semi-structured or structured datasets. Obviously, more structured datasets lead to richer results. This is due to structured data provides more semantics, so in response better inference and deduction can be performed. First, structured information sources are database schema, existing ontologies and knowledge bases. The central problem in learning from structured data is to determine which pieces of structural information can provide relevant knowledge [DG08]. Second, semi-structured data are WordNet [Mil95b], Wikipedia, HTML and XML documents, etc. Third, unstructured data are the most available source of information. This type of data includes natural language texts, i.e. text documents. Learning resources are mainly classified into this last type. This is the reason why this review focuses particularly on ontology learning from texts.

The OL process includes a number of sub-tasks that have to be performed in order to learn lightweight ontologies: concepts extraction, taxonomical relations extraction, and non-taxonomic relations extraction. It should be pointed out that the following additional steps must be performed in order to learn heavyweight ontologies: attributes extraction, instances extraction and axioms extraction. Nevertheless, these sub-tasks fall outside the scope of this work. Further information about these steps can be consulted in the survey proposed by Zouaq [ZN10]. Thus we highlight state-of-the-art knowledge extraction techniques for each sub-task of ontology learning.

4.2.1 Concepts Extraction

This task involves the identification of the main concepts, or domain classes, from the textual sources. Therefore the main challenge is to differentiate domain concepts from non domain concepts.

First, NLP techniques utilize linguistic information to extract the meaningful terms from text. This kind of approaches considers terms as candidate concepts, using parsers and taggers in order to determine the syntactic role of terms or to discover linguistic patterns. For example, some works apply a POS tagger over the corpus in order to identify manually defined patterns [Sab05, ZN09c]. Thus the syntactic analysis allows identifying the nominal phrases that might be important for the domain. In this sense, the most common approach is to use lexico-syntactic patterns (LSP) matching, introduced by Hearst [Hea92]. In this case, a set of seed concepts or patterns is used to extract new concepts or patterns, initiating a cycle of discovery and extraction. An important problem here is to control the quality of the extraction, using some discriminative performance metric.

Second, statistical approaches consider all terms in the sources as potential concepts and employ quantitative metrics to measure the importance of each candidate. For example, the TF*IDF [SB88] measure or the C-Value/NC-Value [FAT98] measure have been used to that end. Usually statistical methods are applied as filters after the detection of concepts with a NLP technique. Additionally, clustering techniques facilitate the elicitation of clusters of similar terms that can benefit the identification of concepts and their synonyms. For example, in the work proposed by Lin and Pantel [LP01] each word is represented by a feature vector that corresponds to a context in which the word occurs. The obtained vectors are then used to cluster the similar terms. In addition, statistical approaches can also be used on top of NLP-based approaches to identify only relevant domain concepts by comparing the distribution of terms between corpora [NVCN04].

Third, some approaches identify concepts using a domain thesaurus or topic map, based on lexical relations and groups of synonyms between terms. Kietz et al. [KMV00] employed the lexical-semantic net for the German language, called GermaNet [HF97], as base ontology. Tan et al [THE00] used WordNet for analysing semantic relations between word forms in order to reduce the raw data extracted from texts to candidate concepts that are useful for ontology development. Velardi and Navigli [VNCN05] analysed WordNet from a linguistic perspective focusing on the textual description of the concepts. Their aim was of extracting relevant information about a given concept and enrich its properties. This analysis can help detect synonyms and related words and can contribute to concept definition. Nevertheless, if the vocabulary in a thesaurus is insufficient or unable to cover all domain concepts, it is possible that the resulting ontology do not adequately convey the domain knowledge.

4.2.2 Taxonomic Relations Extraction

Extraction of taxonomic relationships has been extensively studied, using both NLP and statistical methods.

One of the earliest attempts to derive relations from texts was described by Hearst, who used lexico-syntactic patterns for semantic knowledge extraction [Hea92]. The main idea behind this technique is that linguistic patterns within a corpus can permit identification of the syntactic relationship between terms of interest, and therefore can be used for semantic knowledge acquisition. In Pattern-based techniques, instances of distinct lexico-syntactic patterns are searched in the textual source as indication of taxonomical links. Patterns are usually expressed as regular expressions [CV05b] but they can also be represented by dependency relationships [ZN09b, LP01]. The LSP method can be further improved by using machine learning methods to learn LSP patterns. For example, Snow et al. [SJN04] represented the Heart's patterns using a dependency parse tree in order to train a classifier, and several new patterns were identified. Another linguistic technique for concept relation extraction uses compound noun information [BCM05, VNCN05, CPSTS05]. By means of internal structure of multiple-word terms (nouns phrases), taxonomical relationships can be deduced.

Regarding statistical methods, clustering algorithms are used in order to extract taxonomical relations from text and produce hierarchies of clusters, albeit lees frequently and with less success rate than in concepts extraction [LHC11]. For example, Maedche et al. [MPS03] described two main approaches for hierarchical clustering: the bottom-up approach which starts with individual objects and groups the most similar ones, and the top-down approach, where all the objects are divided into groups. Alfonseca and Manandhar [AM02] propose a top-down search. Starting with the most general concept in the hierarchy, the new concept was added to the existing concept if their topic signature were the closest.

4.2.3 Non-Taxonomic Relations Extraction

Again, the main two methods for extraction of non-taxonomic relations between concepts are part of NLP and statistical techniques.

Regarding linguistic strategies, most of proposal work with linguistic patterns to unveil ontological relations from text. Following Hearst's work [Hea92] on taxonomic relations, different researchers created patterns for non-hierarchical relationships [IMK00], for part-of relations (meronymic) [BC99, Sun02] or causal relations [GBM03]. In fact, many works consider that ontological relationships are mostly represented by verbs and their arguments [ZN10].

Most of the work on relation extraction combines statistical analysis with linguistic analysis. For example, Kavalec [KS05] used a statistical approach, supported with some linguistic information. The linguistic feature used was based on the assumption that relational information is typically conveyed by verbs at the sentence level. Another work, proposed by Zouaq and Nkambou [ZN09c], used typed dependencies to learn relationships. In addition, they employed statistical measurements in order to determine whether or not the relationships should be included in the ontology. An alternative statistical approach uses association rule mining methods to extract relationships between concepts [GBK09, BAB05], where association rules are created from the co-occurrence of elements in the corpus. This technique has been adopted by the Text-to-Onto [MS01]. However, these relationships should be manually labelled later and this task is not always easy for the ontology engineer [ZN10].

4.2.4 Ontology Learning systems

In addition to the commented OL methods, a number of systems for supporting ontology construction have been proposed. A few examples are as follows.

Text2Onto [CV05a] is a tool and framework based on the previous system Text-to-Onto [MS01]. It represents the knowledge at a meta-level in the form of of instantiated modelling primitives. It combines machine learning approaches with basic linguistic processing, such as tokenization, lemmatizing and shallow parsing.

OntoLancs [GSR08] is a framework designed as a cyclical process to experiment with different combination of techniques. The ontology engineer is responsible for deciding which techniques will be used to extract the ontology. In addition, the framework considers methods for evaluating the usefulness and accuracy of the combination of techniques.

OntoCase [Blo09] is a semi-automatic system based on the principles of case-based reasoning. The case base is defined by means of a pattern base, containing both ontology design patterns and architecture patterns. First, the system performs case retrieval and analyse the input matching it to the pattern base. Second, the system reuses the retrieved patterns and constructs a first version of the ontology. Next, the ontology is revised for improving the ontology quality. Finally, the system implements the discovery of new patterns as well as storing pattern feedback.

Finally, [ZGH11] OntoCmaps is a domain-independent tool that extracts deep semantic representations from the input. It uses the extracted representation for building concept maps. The system is based on the inner structure of graphs with the aim of identifying important elements without using any other knowledge source.

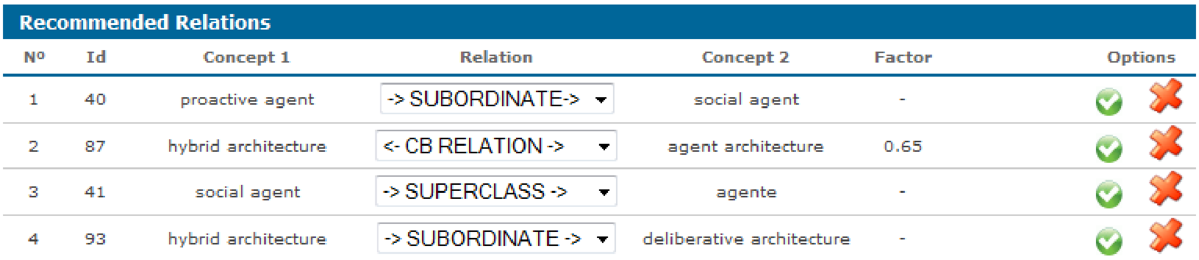
4.3 System Overview

In this section, we present an overview of ORLearn for extracting and recommending concepts and relations between concepts from a set of educational resources. The system performs semi-automatically in order to generate a specific lightweight ontology for the educational domain.

Regarding the concept extraction and recommendation process, it follows the same strategy used to recommend tags in our previous TRLearn system (Section 3). Remember that TRLearn dealt with the different morphological variations and synonyms of the candidates in order to provide conceptually extended tags. Although only one representative term was presented to the user, the system stored the set of terms regarding each tag as a unique concept. Therefore we have performed a minimum adaptation of the method in order to provide recommendation of concepts to the teacher. Nevertheless, the spirit of the methods stays the same. The method is summarized next.

The data flow begins in the Concept Extraction module. First, the most relevant terms are found from the learning resources by means of predefined syntactic patterns. The patterns are defined using the POS tags of each word in the textual source. Next, each candidate is extended with the set of morphological variations and synonyms, using our Wikipedia-based dictionary of concepts (Section 2.3) and some NLP heuristics. Thus this extended set of terms for each candidate defines a concept. Afterwards, a domain-independent and efficient heuristic strategy weights each candidate concept. First of all, the module extracts a set of statistical and semantic features for each candidate. Subsequently, a number of fuzzy heuristic rules are applied. Concluding, the system computes a weight for each candidate concept and produces an ordered list, which is sent to the Recommendation module that acts as interface. Once the teacher selects a concept as valid, the Recommendation module adapts the weights of the rest of the candidate concepts in function of syntactic and semantic features.

The focus of this part of the dissertation lies on the recommendation of taxonomic and non-taxonomic relations between concepts. The system employs a similar strategy, although no weight is provided to the candidate relations. For each pair of validated concept, the Relation Extraction module applies five heuristic rules mixing linguistic patterns with information of the learning resources. Each rule has the goal of extracting a type of relationship. The five relationships are classified into two classes: taxonomic relations (superclass relation, and subclass relation) and educational specific relations (subtopic relation, subordinate relation, and relation through the course content). When a pair of concepts matches with a rule, no other rules are checked for such pair. That is, the rules are exclusive. Finally, the extracted relations are subsequently presented to the teacher trough the Recommendation module (Figure ii.14), who must validate them in order to be include in the ontology. Logically, the teacher can modify the type of relation existent between pairs of concepts.



Recommended Relations

Nº	Id	Concept 1	Relation	Concept 2	Factor	Options
1	40	proactive agent	-> SUBORDINATE-> ▾	social agent	-	✓ ✗
2	87	hybrid architecture	<- CB RELATION -> ▾	agent architecture	0.65	✓ ✗
3	41	social agent	-> SUPERCLASS -> ▾	agente	-	✓ ✗
4	93	hybrid architecture	-> SUBORDINATE -> ▾	deliberative architecture	-	✓ ✗

Figure ii.14: Relation Recommendation Interface

Figure ii.15 depicts the proposed system architecture. In the following subsection, the functionality and implementation of the Relation Extraction module is detailed.

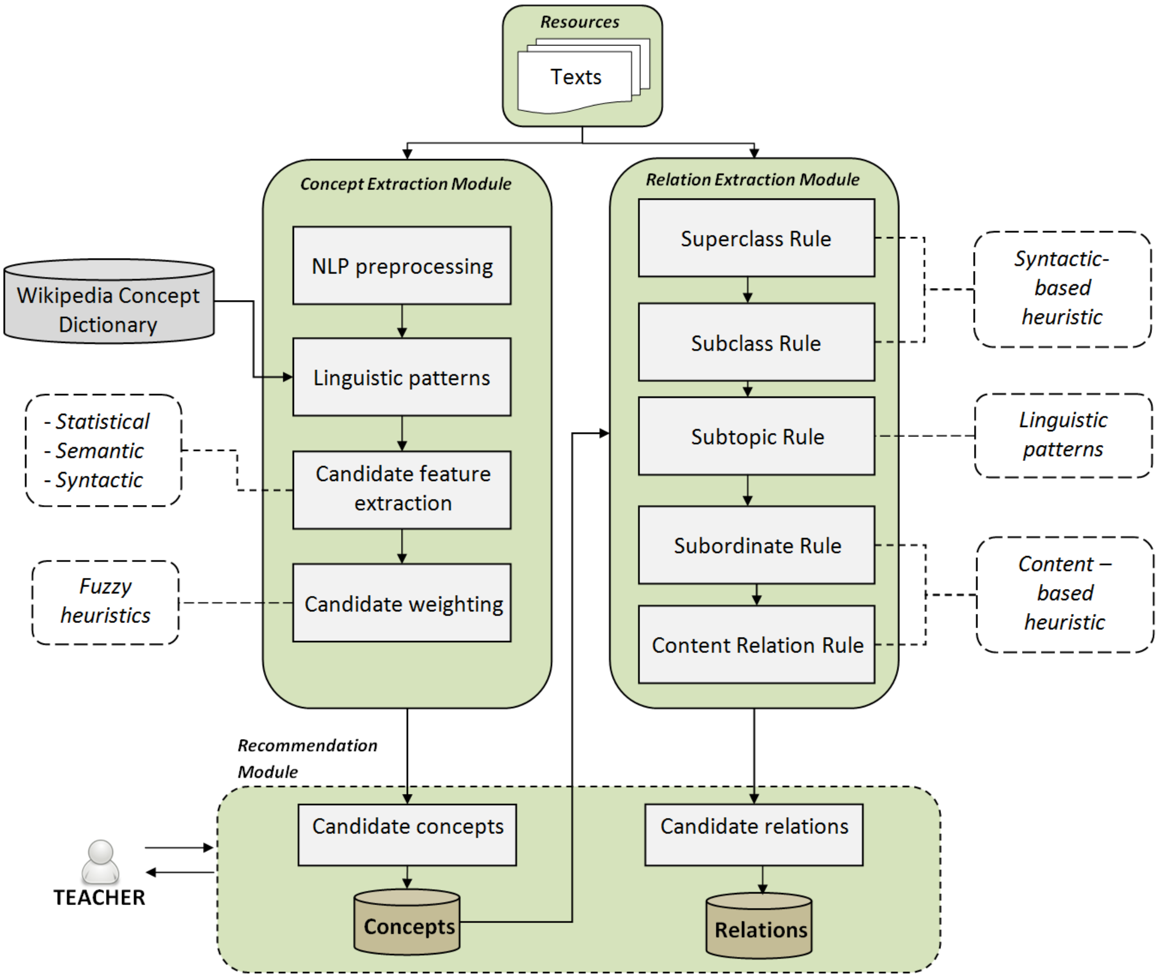


Figure ii.15: ORLearn architecture overview

4.3.1 Relation Extraction Module

The Relation Recommendation module receives the set of concepts and the corresponding linked resources as input, and extracts a list of relations. To that end, five rules are applied iteratively. Each rule is exclusive and looks for a different type of relation.

First, we define relation r_{ij} between the concept c_i and the concept c_j as the triple $r_{ij} = (c_i, c_j, tr) | tr \in \{r_{supc}, r_{subc}, r_{st}, r_{sb}, r_{cb}\}$, where tr defines the type of relation, and r_{supc}, r_{st}, r_{subc}, r_{sb}, r_{cb} refers to superclass relation, subtopic relation, subclass relation, subordinate relation and content-based relation respectively.

Given a pair of concepts $p_{ij} = (c_i, c_j)$, the module checks if any of the relation extraction rules can be satisfied following a order: superclass relation rule, subtopic relation rule, subclass relation rule, subordinate relation rule, and content-based relation rule. If the current rule is satisfied, the corresponding r_{ij} is added to the relation candidate set R, and no other rule is checked for p_{ij}. The algorithm iterates through all possible pairs of concepts. The five rules designed to identify the relations are described next.

(i) Superclass Relation Rule The first rule is in charge of detecting taxonomical superclass relationships between concepts. The main idea behind this rule is that if two or more concepts of the concept base share a common substring, it is rather possible that such substring would be superclass of the concepts containing it. Let illustrate this with an example. Let us assume that we are evaluating the concepts $c_i = \{RAM, RAMmemory\}$ and $c_j = \{ROM, ROMmemory\}$. Both concepts share the term "memory". If such shared term belongs to a concept c_k in the concept base CS, we assume that $c_k = \{memory, \ldots\}$ is superclass of $c_i = \{RAM, RAMmemory\}$ and $c_j = \{ROM, ROMmemory\}$.

Table II.19 describes the process. Given two concepts c_i and c_j, the terms of them having the same number of words are considered. Such terms $term_i$ and $term_j$ belongs to the synonym set of the c_i and c_j respectively. Next, the algorithm look for a shared substring t_{aux} among $term_i$ and $term_j$. The substring must encompass complete words starting from the end of each term. If such substring presents a number of words equal to the number of words of $term_i$ and $term_j$ minus 1, and it belongs to the synonym set of another concept c_k in CS, then the relations $r_{ki} \leftarrow (c_k, c_i, r_{supc})$ and $r_{kj} \leftarrow (c_k, c_j, r_{supc})$ are created.

(ii) Subtopic Relation Rule The second rule is in charge of detecting education-related subtopic relationships between concepts. We employ a set of linguistic patterns to that end (Table II.20). Given two concepts c_i and c_j, the algorithm firstly replaces the variable t_v at the beginning or the end of each pattern with the terms of c_j. Therefore, if a term of c_i matches with linguistic patterns joint to a term of c_j, we consider that c_i would be subtopic of c_j.

For example, let us assume that we are considering the concepts $c_i = \{intelligentagent, agent\}$ and $c_j = \{architecureofagents, agentarchitecture\}$. Then c_j matches with the linguistic patterns "$agentarchitecture(s)$" and "$architecture(s)(in|of|within|on)(a|the)?agent$". Therefore we assume that $c_j = \{architecureofagents, agentarchitecture\}$ is subtopic of $c_i = \{intelligentagent, agent\}$.

Table II.21 shows the algorithm. The algorithm receives the set of linguistic patterns LPS. In addition, the function $regExp.add(t)$ returns the corresponding regular expression obtained after replace the variable t_v of the linguistic pattern with the given term t.

```
INPUT:
    $c_i$: $ith$ concept
    $c_j$: $jth$ concept
    $CS$: concept set

OUTPUT:
    $R_{supc}$: resulting superclass relation candidate set

begin
    $R_{supc} \leftarrow \varnothing$
    foreach $term_i \in c_i.synonymSet$ do
        foreach $term_j \in c_j.synonymSet$ do
            if $term_i.numWords = term_j.numWords \wedge term_i.numWords > 1$
            $\wedge term_i.lastSubstring = term_j.lastSubstring$ then
                $t_{aux} \leftarrow term_i.lastSubstring$
                if $t_{aux}.numWords = (term_i.numWords - 1)$ then
                    if $\exists c_k \in CS | t_{aux} \in c_k.synonymSet, c_k \notin \{c_i, c_j\}$ then
                        $r_{ki} \leftarrow (c_k, c_i, r_{supc})$
                        $r_{kj} \leftarrow (c_k, c_j, r_{supc})$
                        $R_{supc} \leftarrow R_{supc} \cup r_{ki}$
                        $R_{supc} \leftarrow R_{supc} \cup r_{kj}$
                    end if
                end if
            end if
        end for
    end for
end
```

Table II.19: Superclass Relation Rule

(iii) Subclass Relation Rule The third rule is in charge of detecting taxonomical subclass relationships between concepts. The main idea behind this rule is the following. If a term of the concept c_i is completely contained in a term of other concept c_j, it is rather possible that c_j would be subclass of c_i. Let illustrate this with an example. Let us assume that we are evaluating the concepts $c_i = \{intelligentagent, agent\}$ and $c_j = \{reactiveagent\}$. The term "agent" is contained as substring in "reactive agent". Therefore we assume that $c_j = \{reactiveagent\}$ is subclass of $c_i = \{intelligentagent, agent\}$. Table II.22 shows the algorithm. It should be pointed out that this rule is checked after the subtopic relation rule because if two concepts satisfy the former rule, also satisfy this. Thus we prioritize the subtopic detection.

(iv) Subordinate Relation Rule The fourth rule is in charge of detecting education-related subordinate relationships between concepts. The idea of subordination is related to the educational content from which the concepts where extracted, i.e. content-based subordination. That is, a concept c_i is subordinate of other one c_j if the set of all learning resources linked to c_i is contained in the set of the resources linked to c_j. The rule is shown in Table II.23.

(v) Content-based Relation Rule Finally, the fifth rule is in charge of detecting education-related content-based relationships between concepts. The idea behind the rule is similar to previous

Linguistic Pattern	Examples
$t_v\ architecture(s)?$	system architecture, building architecture
$architecture(s)?\ (in\mid of\mid within\mid on)?\ (a\mid the)?\ t_v$	architecture of agents
$type(s)?\ (in\mid of\mid within\mid on)?\ (a\mid the)?\ t_v$	type of approach
$t_v\ definition(s)?$	algorithm definition
$definition(s)?\ (in\mid of\mid within\mid on)?\ (a\mid the)?\ t_v$	definitions on e-learning
$t_v\ concept(s)?$	learning concepts
$concept(s)?\ (in\mid of\mid within\mid on)?\ (a\mid the)?\ t_v$	concept of artificial intelligence
$t_v\ interaction(s)?$	student-student interaction
$interaction(s)?\ (in\mid of\mid within\mid on\mid among\mid between)?\ (a\mid the)?\ t_v$	interaction among students
$t_v\ model(ling\mid s)?$	user modelling
$model(ling\mid s)?\ (in\mid of\mid within\mid on\mid among\mid between)?\ (a\mid the)?\ t_v$	models of intelligence
$t_v\ strcuture(s)?$	algorithm structure
$strcuture(s)?\ (in\mid of\mid within\mid on\mid among\mid between)?\ (a\mid the)?\ t_v$	structure of the platform
$t_v\ classification(s)?$	animal classification
$classification(s)?\ (in\mid of\mid within\mid on\mid among\mid between)?\ (a\mid the)?\ t_v$	classification of elements
$t_v\ application(s)?$	virtual learning application
$application(s)?\ (in\mid of\mid within\mid on\mid among\mid between)?\ (a\mid the)?\ t_v$	applications of e-learning tools
$t_v\ taxonom(y\mid ies)?$	computer taxonomy
$taxonom(y\mid ies)?\ (in\mid of\mid within\mid on\mid among\mid between)?\ (a\mid the)?t_v$	taxonomy of animals species

Table II.20: Subtopic linguistic patterns

one, although less restrictive. The content-based relation considers again the number of resources linked to each concept. Nevertheless, now not the complete resource set linked to c_i must be contained in the resource set linked to c_j. Therefore, a concept c_i is content-based related of other one c_j if the resource set linked to c_j contains at least one resource linked to c_i. The rule is shown in table II.24.

```
INPUT:
    $c_i$: $ith$ concept
    $c_j$: $jth$ concept
    $LPS$: linguistic pattern set

OUTPUT:
    $R_{st}$: resulting subtopic relation candidate set

begin
    $R_{st} \leftarrow \varnothing$
    foreach $term_i \in c_i.synonymSet$ do
        foreach $term_j \in c_j.synonymSet$ do
            if $\exists regExp \in LPS | term_i matcheswithregExp.add(term_j)$ then
                $r_{ij} \leftarrow (c_i, c_j, r_{st})$
                $R_{st} \leftarrow R_{st} \cup r_{ij}$
            end if
        end for
    end for
end
```

Table II.21: Subtopic Relation Rule

```
INPUT:
    $c_i$: $ith$ concept
    $c_j$: $jth$ concept

OUTPUT:
    $R_{subc}$: resulting subclass relation candidate set

begin
    $R_{subc} \leftarrow \varnothing$
    foreach $term_i \in c_i.synonymSet$ do
        foreach $term_j \in c_j.synonymSet$ do
            if $term_i \subseteq term_j$ then
                $r_{ji} \leftarrow (c_j, c_i, r_{subc})$
                $R_{subc} \leftarrow R_{subc} \cup r_{ji}$
            end if
        end for
    end for
end
```

Table II.22: Superclass Relation Rule

```
INPUT:
    $c_i$: $ith$ concept
    $c_j$: $jth$ concept

OUTPUT:
    $R_{sb}$: resulting subordinate relation candidate set

begin
    $R_{sb} \leftarrow \varnothing$
    $resSet_i \leftarrow c_i.linkedResources$
    $resSet_j \leftarrow c_j.linkedResources$
    if $\forall r_i \in resSet_i | r_i \in resSet_j$ then
        $r_{ij} \leftarrow (c_i, c_j, r_{sb})$
        $R_{sb} \leftarrow R_{sb} \cup r_{ij}$
    end if
end
```

Table II.23: Subordinate Relation Rule

Additionally, we have defined a measure that checks the strength of the content-based relation between two concepts (II.14). If the weight is not significantly high (higher than 0.65), the relation is not recommended to the teacher. In that case the relation is not discarded, but it is kept out from the recommendation module. It remains in a waiting state until more resources containing both concepts are added to the system. When a user inserts a new document, the system automatically runs the concept extraction module. The new concepts are extracted, weighted and sent to the recommendation module. In contrast, if the concepts in the document are currently in the concept base, the new document is linked to them. Next, the system recalculates the content-based relation

```
INPUT:
    $c_i$: $ith$ concept
    $c_j$: $jth$ concept

OUTPUT:
    $R_{sb}$: resulting subordinate relation candidate set

begin
    $R_{sb} \leftarrow \varnothing$
    $resSet_i \leftarrow c_i.linkedResources$
    $resSet_j \leftarrow c_j.linkedResources$
    if $\exists r_i \in resSet_i | r_i \in resSet_j$ then
        $r_{ij} \leftarrow (c_i, c_j, r_{cb})$
        $R_{cb} \leftarrow R_{cb} \cup r_{ij}$
    end if
end
```

Table II.24: Content-based Relation Rule

strength measure for the accepted content-based relations, taking now into consideration the new document.

$$wcb_{(cj/ci)} = \frac{|\{c_i.linkedResources\} \cup \{c_j.linkedResources\}|}{|\{c_i.linkedResources\}|} \tag{II.14}$$

4.4 Experiences with the System

In this section, we discuss the quality of the concept relations exploration stage of our OL approach. The concept extraction module was previously evaluated as part of the experiments carried out for the TRLearn system (Section 3.4). Hence we provide a set of experimental tests considering the concept relation recommendation process. The performance of the relation extraction module is evaluated by applying our algorithm on a dataset of educational resources. Finally, we examine the impressions of the teacher involved in this experiment.

4.4.1 Dataset

The relation recommendation process depends on the previously accepted concepts. For this motive we have used the same dataset of learning resources employed during the evaluation of the concept recommendation process explained in Section 3.4. The evaluation dataset is composed by 100 quiz question and 100 learning tasks randomly selected from a VLE, which was being used to support the undergraduate course “Artificial Intelligence” in the Computer Engineering degree in the University of Granada.

In addition, we have collected the concepts that were manually selected by the three domain experts that took part in the tag recommendation experiment. Subsequently, we have discarded those concepts that were not extracted by our algorithm. A set of 55 concepts remains for this experiment. A few examples are shown in Table II.25 (plural and misspellings are not displayed in the examples).

Concept name	Synonyms
intelligent agent	agent, artificial agent, rational agent
search	search algorithm
expert system	-
reactive agent	-
intelligence	human intelligence
depth-first search	dfs algorithm
artificial intelligence	ai, machine intelligence
deliberative agent	-
multi-agent	multi-agent system, mas
alan turing	turing

Table II.25: Concepts for the evaluation of the relation recommendation process

4.4.2 Design principles

TRLearn was implemented with the following architectural design principles. We use a MySQL database (version 14.12), and the Hibernate[18] library (version 3.3.5), as framework for mapping the dictionary of concepts. We use Java SDK 6 (1.6.0_20) to implement the system. Finally, an Intel(R) Core(TM) i5 processor 430M (2.26Ghz) and 4GB RAM was used for the evaluation.

4.4.3 Experimental evaluation

With the aim of testing the relation recommendation procedure, we have mainly focused the evaluation on the manual side. It is composed by two different phases: an evaluation about the performance and an evaluation about the usability of the system.

Initially, we ask a volunteer teacher in the field of "Artificial Intelligence" for defining the relations existing in the datasets for the given set of concepts. Then, the manually-selected relations were compared against the relations recommended by our proposed method. Manually extraction identified 37 relations (7 superclass relations, 7 subclass relations, 14 subtopic relations, 8 subordinate relations, and 1 content-based relation).

Following prior criteria, the quality of the results is tested by means of the F-measure metric (II.6) (Section 2.6). Remember that this metric is computed as the weighted harmonic mean of precission (II.4) and recall (II.5). Given a set of relations extracted by the method and the set of relations manually extracted by humans, precission is the number of matched relations divided by the total number of relations extracted by the system; and recall is the number of matched relations divided by the total number of human extracted relations.

The performance of our method regarding each type of recommended relation is summarized in Table II.26. During the process, 58 relations were extracted and recommended. From them 43 were selected as valid, rising to 74.13% of overall precission. The recall rate is less pronounced (67.56%). This fact is mainly due to some of the accepted relations were manually classified into other type of relation, provoking a lower level of recall. For example, in the initial stage the teacher identified the relation existent between "type of agents" and "deliberative agent" as subclass. In contrast, the system extracted this relation as subordinate. Although the teacher selected this subordinate relation as valid during the recommendation, this relation is not considered for the recall rate. The results show a robust performance of our approach, although some considerations need to be taken

[18] http://www.hibernate.org/

into account.

	Relations extracted		Performance		
Relation	Correct	Wrong	%Precission	%Recall	%F-measure
Superclass relation (Taxonomic)	3	2	60.00	42.85	49.99
Subclass relation (Taxonomic)	9	4	69.23	57.14	62.60
Subtopic relation (Education-related)	9	0	100.0	64.28	78.25
Subordinate relation (Education-related)	19	9	67.85	62.50	65.06
Content-based relation (Education-related)	3	0	100.0	100.0	100.0
Overall	43	15	74.13	67.56	70.69

Table II.26: Evaluation of recommended relations for the course resources

The system offers poor results taking into consideration the recommended taxonomic relations. Although heuristics employed are simple, efficient and domain-independent, it is necessary to perform a deeper analysis of the task. For example, it would be desirable to define a set of linguistic patterns to that end. Precisely, the linguistic pattern approach designed for detecting subtopic relations have resulted extremely successful. 100% of recommended relations were subsequently accepted. Additionally, all the recommended subtopic relations were included in the set of human annotated relations. This approach seems to be indeed the most adequate for this task. Finally, the precission and recall rates obtained for the content-based relation have increased significantly the overall performance. This kind of relations is rather easy to detect for both the system and the human annotator. This is due to the relation is defines in basis of the shared resources linked to the related concepts. Here human subjectivity does not affect the results.

After analysing the empirical results, we ask the teacher for his opinion about the usability of the system. He expressed a great satisfaction regarding the educational-based relations. Considering the taxonomic relations, he found the performance slightly worst:

> “The system is easy-to-use and accurate for the educational domain. Although the recommended taxonomic relations were slightly insufficient, the recommended educational-based relations perfectly fit with the organizational needs of a high education course. In overall, the system seems adequate and helpful for organizing the course content in an ontology structure. It requires minimum intervention for correcting the recommended results.”

4.5 Conclusions and Future Work

In this work, we have presented ORLearn, a ontology recommender system able to create a lightweight domain ontology from scratch, without the necessity of a prior knowledge. The computer-supported construction of ontologies is a wide branch of the field of Information Extraction, and it is usually known as Ontology Learning. There exists several approaches, each one with their proper benefits and disadvantages, depending on the context and/or the domain. Nevertheless, although most of approaches on OL considers taxonomic and non-taxonomic relations between concepts, they do not take into account educational-related relationships between concepts, that might help students to acquire an schematic view of the importance and dependences of each topic along the course content.

The above reasons have encouraged us to develop an semi-automatic method for building domain ontologies for the educational field. We focus on lightweight ontologies for two main reasons. In first place, users may not have enough expertise in the field for managing complex ontology constructions. In second place, the computational complexity is substantially reduced. The system operates in a multi-domain scheme. Apart form the identification of concepts, the system is able to detect five types of relations: two taxonomic relations (superclass and subclass relations) and three educational-based relations (subtopic, subordinate and content-based relations).

Experiments evidence that our system obtain a good performance, being tested on a set of learning resources which belongs to a higher education course. Particularly, the system exhibited an excellent performance regarding the detection of educational-based relations. In contrast, the performance was slightly worst considering the taxonomic relations. Additionally, we asked the teacher of the course for his opinion about the usability of the system. He thought that the system is suited tools for helping the VLE manager to organize the course content in an ontology meta-data structure.

Regarding future work, we plan to adjust the algorithms in charge of detecting the taxonomic relations, which obtained the poorest results. In order to perform such task, we will develop a further investigation about the use of linguistic patterns for extracting taxonomic relations. In addition, we plan to design new extraction methods considering now other types of relationships between concepts: partonomic relations, causal relations. . . .

5 Final Discussions and Future Work

The increasing volume of learning resources in e-learning scenarios may lead to information overload states. Students may feel learning disorientation and find themselves being unable to acquire the knowledge. The first step towards the development of cognitive strategies which may alleviate this situation consists of organizing the knowledge of the domain. In this part of the dissertation we have focused on the development of three strategies able to extract and organize the knowledge from the educational content with the minimum intervention of the teacher. We have selected three well-known meta-data structures as containers of the knowledge: taxonomies, folksonomies, and ontologies. Each structure requires a different level of semantic richness, and therefore a different procedure in order to be constructed.

First, we have developed an Automatic Key term Extraction method able to deal with the particularities of the educational content. In general, classical AKE methods work with general-domain documents (containing world knowledge from public sources covering high diffusion topics). However, common learning resources are frequently related to specific-domains (containing knowledge from specific issues, usually with limited or private diffusion). Moreover, this kind of documents do not contain every detail of the considered topic, but only specific information referred to significant aspects of it. For that reason, our method present a two-fold scheme. On the one hand, it detects multi-domain key terms taking advantage of Wikipedia, a well-known educative resource. On the other hand, it employs a frequency in language dictionary for identifying key terms of specific domains. The key terms extracted are linked to the resources from which they were extracted. This allows to create a taxonomic structure of the educational content. In addition, we have proposed a prototypical application for taxonomic indexation of learning resources. The counterpart of AKE methods relies on the relatively high rate of imprecision detected in the results. The balance between the complexity of the proposal and the quality of the results is rather delicate. Teachers may not have enough technical expertise for managing more complex approaches, such as ML strategies. Therefore a validation of the results would be desirable.

Second, we have designed a Tag Recommender method able to create a conceptually extended folksonomy from scratch. Although TR methods present a high diffusion in social applications, it has not been sufficiently explored yet in the field of e-learning. Attending to literature, we found that the major approaches in TR require from prior domain knowledge. This is a strong requirement in initial phases of the system, limiting this the scope of application. Therefore, we have designed a TR system which operates without prior knowledge, using Wikipedia as multi-domain knowledge base. It assists the members of a community looking for a subset of conceptual tags which better fits with the current learning resource. The set of conceptual tags selected by the user community for labelling the set of learning resources forms a folksonomy. The method perfectly fits with the educational domain, since the users are the final responsible of accepting the results. Experiments evidence that our system obtain a good performance, being tested on a set of learning resources which belongs to a higher education course. Additionally, a satisfaction questionnaire was fulfilled by a set of users, revealing that they felt comfortable with the system, and that it was useful to extract tags and to represent the content of the domain.

Third, we have focused on Ontology Learning problem in order to develop a semi-automatic method that assists the teacher in the task of obtaining lightweight domain ontologies from the educational content. Although today OL is widely tackled, we found a lack of methods taking into account educational-related relationships between concepts, that might help students to acquire an schematic view of the importance and dependences of each topic along the course content. Therefore, our method extracts and recommends concepts and relationships between concepts to the teacher from the learning resources. The extracted relations are classified into taxonomic relations and educational-based relations. We have paid special attention to the educational-based relations, and therefore additional efforts on the refine of the extraction procedure of taxonomic relation are necessary. This fact was confirmed by the teacher in charge of validating the usefulness of our method, although he stated that the system was adequate and helpful for organizing the course content in an ontology structure.

The obtained meta-data structures will serve as base for the development of cognitive strategies able to support the learning process in e-learning. Thus, the next point to deal with in this dissertation concerns with retrieval and visualization techniques for the educational domain (Chapter III).

Chapter III

Information Retrieval and Visualization Methods for the Educational Domain

1 Introduction

Current Virtual Learning Environments provide features for creating and organizing learning resources of multiple formats through web-based interfaces. Attending to their design characteristics, they are easily recognizable as hypermedia environments that consist of network-like information structures, where fragments of information are stored in nodes that are interconnected and can be accessed by electronic hyperlinks [Rou96, CON87]. Hypermedia systems provide quick access to a large amount of information under multiple formats. The Internet is indeed a important collection of sources of information. With more and more information put on the websites, the scale and complexity of modern websites are growing rapidly. Nevertheless, it is often difficult to find information relevant to the interests of individual users when confronted by the huge amount of possible content. This leads to delicate matters that are been addressed by the scientific community.

The problem of *information overload* is widely recognised today. It is referred to the stress that can be experiences from a feeling of lack of control [EM00]. We can find several definitions in early literature. For example, Butcher [But98] stated that it can mean several things, such as having more relevant information that one can assimilate or it might be burdened with a large supply of unsolicited information, some of which might be relevant. Klapp [Kla86] declared that a large amount and high rate of information act like noise when they reach overload: a rate too high for the receiver to process efficiently without distraction, stress, increasing errors and other costs making information poorer.

The commented situation also leads to the issue known as *disorientation problem.* This is not a contemporary matter, but it was widely reviewed in early years in the literature [HA89, HE89, Fos89]. As brief introduction, and according to Stanton et al. [SCD00], we identify two facets in the disorientation problem that might be considered. First, the disorientation is rooted in the intrinsic nature of hypertext, constituting mainly a browsing problem that occurs as a result of the environment's complexity. The second facet results from the cognitive demands that hypermedia systems impose on their users and is called *cognitive overhead*, i.e., the additional effort and concentration necessary to maintain several tasks or trails at one time [CON87].

The matters concerned with information overload and disorientation are not solved yet. Most websites today do not provide navigation guides tailored to individual user needs. These circumstances are not dissimilar for educational systems that suffer from the same drawbacks. The presence of advanced features in educational environments has provided access to a richer and more complex variety of learning resources. This proliferation of information has imposed information overload on students [SH13, Kal11, SVMP98]. Shrivastav et al. [SH13] identify the causes of information overload based on Bloom's taxonomy domains. Additionally, the characteristics of e-learning platforms may provoke disorientation in students. While resource-abundant learning environments allow students a great deal of freedom and flexibility in searching for, selecting, and assembling information, students may suffer from cognitive overload and conceptual and navigational disorientation when faced with massive information on-line [WPC+11, Ter05, MM99]. In fact, students' disorientation problems may affect their learning performance [SHC10]. Wang et. al [WPC+11] stated

> "(...) The challenge is even greater when learning contents are scattered under disparate topics and complex knowledge structures. When faced with this problem, many students are unable to figure out features and meaningful patterns of various kinds of information, and are easily hampered by limited working memory. This is mainly because novices lack sufficient knowledge and a deep understanding of the subject domain, which is crucial to organizing information and knowledge for retention in long-term memory. Also, traditional education breaks wholes into parts, and focuses separately on each part, and students are often unable to create the big picture before all the parts are presented."

The development of solutions to handle the disadvantages of hypermedia systems is still overriding. In the context of e-learning, VLEs should incorporate mechanisms for helping students to carry out the learning process more efficiently and effectively [CLC05, Ham01, WPC+11]. Most of resolution methods are related with the implementation of adequate aids for navigating through the content space. The concept of navigation is a meaningful one in a hypermedia system in the sense that we can understand users' actions as a movement through electronic space [MDR90]. Therefore, there exists a demand for intelligent tools able to make website navigation easier. We found the following three categories of application:

- *Search engines* are usual in-site tools for retrieving resources from web sites associated to user queries. The search engine usually looks for those documents that are closely related to the input keywords and present them directly to the user.
- *Visualization* provide effective methods for representing the underlying knowledge of the content. Visualization tools for knowledge management make use of the human cognitive processing system in order to create and convey content more efficiently. We can distinguish between information and knowledge visualization. Both mechanisms employ similar techniques. Based on specific mapping rules, they translate resource objects into visual objects, offering easy and comprehensive access to the underlying content [JLH05]. A deeper analysis of this two subject can be consulted in section 3.
- *Adaptive mechanisms* in hypermedia systems build a model of the goals, preferences and/or knowledge of each individual user, and use this model throughout the interaction with the user, in order to adapt the content to their needs [Bru00]. The user model is constructed from various sources that can include implicitly observing user interaction and explicitly requesting direct input from the user. It is used to provide adaptation effect, i.e., tailor interaction to different user in the same context [BM02].

Although thus methods and tools are gradually being incorporated in hypermedia systems, their usage is less frequent in VLEs. Therefore, the design of intelligent tools in e-learning context seems mandatory. For that matter, we study in this chapter the application of search mechanisms, and knowledge and information visualization techniques. In order to accomplish such task, we will deal with CI techniques related to the fields Question Answering for searching, and Tag Clouds and Concept Maps for visualization. First, Question Answering is a field that aims to handle the interpretation of Natural Language (NL) user queries. It has the purpose of retrieving the resources with high rate of relation with each query. Thanks to an instrument of this kind, both students and teachers would be able to perform more complex searches than those based on simple keywords. More concretely, we will focus on FAQ retrieval, practical field of application of Question Answering. Frequently Asked Question lists are receiving great attention for their capacity to collect and organize user questions and expert answers about specific topics, and are being increasingly used in e-learning [dBC02, Zha04]. Second, Tag Clouds are one of the most usual visualization techniques employed tu summarize the domain knowledge of textual sources. Although they present a high diffusion in social environments, its use in educational context is rather unusual. Third, we focus on Concept Maps since they probably are the most common form of visual representation in e-learning.

It should be pointed out that in this part of the dissertation we do not address adaptive approaches, postponing its discussion to chapter IV. Although adaptive hypermedia mechanism mainly consider the automatic adaptation of the content, the concept of "adaptation" in the e-learning context encompasses rather more elements. In fact, it involves two different fields of computer science: Adaptive (Hypermedia) Educational Systems and Intelligent Tutoring Systems. Thus such fields are quite extensive by themselves to be taken into consideration separately. Additionally, we establish the distinction between searching/visualization and adaptive methods attending to their design scheme. Meanwhile searching and navigation methods require from a direct interaction with the structural representation of the educational content, educational adaptive mechanisms pay more attention to the representation of the knowledge and goals of students.

The rest of this Chapter is structured as follows. Section 2 offers an study on FAQ Retrieval problem. The both knowledge and information visualization methods will be explained in Section 3. Finally we conclude with some final discussions in Section 4.

2 FRLearn: Highly-Precise FAQ Retrieval System for Virtual Learning Environments

2.1 Introduction

Current researches on e-learning are exploring new ways to promote interaction between students, emphasizing the information share among the course members. The asynchronous framework provided by VLEs helps students to obtain immediate solutions for their needs without the teachers staying on-line.

In this sense, models based on Frequently Asked Question (FAQ) are being increasingly adopted as efficient and asynchronous tools for knowledge sharing. FAQ lists are very valuable for helping non-experts users to learn the main concepts of any topic. FAQ model within VLEs allows students to reuse resources, helps students learn by themselves, and reduces teacher workloads [LYO$^+$11]. Students and teachers can collaborate with each other in real time without having to schedule office hours or wait for email responses, along with a way for teachers to post previously answered questions allowing students to get solutions to their problems when the teacher is unavailable [FK04]. These benefits have fostered the implantation of FAQ managing mechanisms in e-learning frameworks [MKSG06, LYO09, ZYY07, JZZL10].

Nevertheless, manual searches on FAQ lists present a number of drawbacks. According to Sneiders [Sne99], the search for relevant information in large FAQ lists may result tedious for the user. What is more, the information might be mixed up along the document or even not be contained in it. In this scenario, an automatic Information Retrieval (IR) method could improve the model, allowing automatic search through the knowledge content. In particular, FAQ retrieval is the sub-problem of the field Question Answering (QA) that considers the automatic retrieval of Question/Answer pairs (from the FAQ collection) relevant to users queries expressed in Natural Language (NL). Those pairs are usually displayed as ranked lists, according to their relevance to the user query. FAQ retrieval system has two main stages: prior knowledge representation of the FAQ collection, and semantic search of the information.

Attending to above considerations, FAQ model on large-data e-learning scenarios arguably depends on suited FAQ retrieval methods able to (i) manage large volumes of information efficiently, (ii) automatically capture the expert knowledge in an interpretable and extendible form, and (iii) retrieve high-precise answers. Nevertheless, traditional FAQ retrieval systems present two important limitations with dependence on the approach. Simple knowledge representations of the dataset lead to efficient responses, but present a lack of semantic analysis. Statistical representations are indeed hardly interpretable and extendible. On the contrary, complex knowledge representations are suitable for NL scenarios, but they require support from domain-dependant resources (keywords, linguistic rules, lexicons, domain ontologies, or question templates). Regardless the manager's expertise on knowledge engineering, manual construction of knowledge bases from large datasets becomes a complex and time-consuming task. In addition, FAQ collections should not be statistic. The flow of information changes as students' needs grow. Hence the performance of the system must not be dependent of manual knowledge construction or validations.

In this context, we have designed a new system so-called FRLearn (FAQ Retrieval for Learning) that simplifies both approaches but taking the main strengths of them. On the one hand, the representation of the knowledge is extracted using a shallow parsing NLP method. By means of linguistic patterns we detect meaningful semantic information units on corpora. The proposed scheme is easily interpretable and portable, fulfilling the requirements of the problem. In addition, we establish the importance of the extracted information units using a statistical approach, without needing of human interaction. On the other hand, we apply two predefined multi-domain knowledge sources in order to improve the semantic search. We employ WordNet and Wikipedia as knowledge sources, which do not need from additional maintenance. Effectiveness of our retrieval system is contrasted with state-of-the-art algorithms for FAQ retrieval.

The rest of this section is organized as follows. First, we provide the features of WordNet as knowledge source in Subsection 2.2. Next, Subsection 2.3 offers an overview of FAQ Retrieval field. In Subsection 2.4, we comment the system architecture and functionality. FAQ retrieval algorithm is explained in Subsection 2.4.4. The method of analysis and the experimental validation of our modules are outlined in Subsection 2.5. Finally, Subsection 2.6 concludes with a discussion of results and future research.

2.2 WordNet

WordNet is a publicly available[1] lexical resource broadly use by a huge number of NLP systems (such was the case that citing here just a few would be thus uninformative). Albeit it was defined by the English language, it has been adapted to many languages in the so-called EuroWordNet project[2].

WordNet collects nouns, verbs, adjectives, and adverbs under the concept of *synsets* —numeric codes that univocally identify different synonymous sets of words. These synsets are linked according to different semantic relations conforming thus a network of meaningfully related work. This network allows NLP techniques to implement rich procedures exploiting distances between words.

WordNet contains 117000 synsets that are linked to other synsets through conceptual relations. Additionally, a brief definition of each synstet, called *gloss* serves to briefly describe the synonym set. Most important relations among synsets include *hyperonymy* and *hyponymy* (also called IS-A relations). Moreover, common nouns and specific instances are explicitly differentiated in WordNet. In addition to IS-A relations, PART-OF relations are also represented in the net. Finally, adjectives are also linked through antonymy relations, that is, opposite polarity in the semantic of adjectives is also reflected in the net.

2.3 Related works

In this section, we discuss the main FAQ retrieval approaches related to our work. To that end, we briefly introduce the earlier systems. Later, current approaches are divided into two categories: methods requiring complex knowledge bases and methods that do not. We first comment FAQ retrieval mechanisms in e-learning platforms. Subsequently, we comment generic approaches.

One of the first works in FAQ retrieval task was FAQ Finder [HBML95]. This system employs a NLP strategy involving a syntactic parser to identify nouns and verbs, and performs concept matching using semantic knowledge through WordNet. It uses a vector-space model (VSM) in order to calculate the similarity degree between questions. Later, Whitehead proposed the Auto-FAQ system [Whi95]. It follows a keyword comparison criterion to implement the question matching in a shallow language understanding perspective. The system proposed by Sneiders [Sne99] works in a similar way, mixing a shallow language understanding strategy with a keyword comparison technique called Prioritized Keyword Matching strategies. SPIRE [DR97] is a hybrid Case-Based

[1] http://wordnet.princeton.edu/

[2] http://www.illc.uva.nl/EuroWordNet/

Reasoning (CBR) and IR system. The system first follows a CBR approach to reduce the number of candidate documents. Then, its INQUERY retrieval engine module processed the documents employing IR techniques. This method present a main handicap: text passages have to be manually labelled. Finally, FallQ system also employs a CBR approach [LB97]. The system represents the domain knowledge by means of manually crafted keywords in order to build Information Entries (IEs). It depends on, therefore, expert knowledge to define the closed domain.

Recent researches could be classified into two categories: those approaches that require much knowledge modelling, and those that do not. (1) The first category normally involves (a) NLP systems and (b) template-based systems. NLP systems aim to obtain a formal representation of NL to give back a concise answer. Meanwhile, template-based systems use set of linguistic templates for the matching process. (2) The second category usually involves (c) statistical systems, which match the user queries to FAQ questions by establishing semantic distance measure between them. They are the most usual to deal with large collections. Their main challenge entails defining syntactic links between linguistic structures with the same semantic (i.e. to find which words can be perceived as synonyms in a given context).

Often, the domain knowledge is modelled by domain ontologies. Wang et at. [WWY05] proposed a semantic search mechanism as part of the e-Learning Message Communication framework. The method is supported by a domain ontology and a set of predefined linguistic patterns. This methodology can detect question patterns and find the positions of the keywords in the ontology. This allows a rapid access to the documents in the knowledge base. Regarding non-educational approaches, the system proposed by Yang et al. [YCH07] combined a domain ontology with a probabilistic keyword comparison measure. Yang also presented a mixed approach combining templates with a domain ontological model based on keywords for catching the user's intention [Yan09b]. Another ontology expansion method is proposed by Liu et al. [LLL10]. This system added new manually annotated questions when the obtained similarity score does not exceed the threshold. A different approach involving domain ontologies is analysed by Wu et al. [WYC05]. The method performs an initial classification of the user's questions into ten question types. The answers in the FAQ collections are then clustered using Latent Semantic Analysis (LSA) and K-means algorithm. The system employs an ontology based on WordNet and HowNet to obtain the semantic representation of the aspects. Finally, the maximum likelihood estimation in a probabilistic mixture model is used as the retrieval process. Finalizing, the system proposed by Guo and Zang [GZ09] uses a domain ontology representation of the knowledge for providing personalized services based on users' profiles. Apart from the ontology modelling, there are a number of FAQ retrieval systems that make use of a set of linguistic templates to cover the knowledge. The Sneider's template-based systems [Sne99, Sne02] are examples of this kind. They use matching with both regular expressions and keywords in the retrieve process. We refer the reader to a number of studies involving knowledge modelling [RAM03, GGSB05, Win00, CCV$^+$08].

The main strength of knowledge-based methods is that they provide precise answers in general. However, they imply many knowledge modelling. To overcome this disadvantage, statistical methods have been proposed [JCL05, SB04, BCC$^+$00]. These approaches perform without complex knowledge bases. Yang [Yan09a] proposed a FAQ system for supporting learning in a on-line community. The system is considered as a knowledge share platform. FAQ are collaboratively created by students and teachers, and added to the knowledge base. The framework contains a FAQ retrieval module that works in two stages. During the first stage, the module automatically extracts an index of the documents in the knowledge base. Each document is indexed by a set of keywords extracted using the log-entropy weighting scheme for detecting meaningful sentences and TF*IDF as filter of important terms. The second stage relies on the retrieval of documents. This is performed using the cosine similarity measurement for retrieving semantically related question-answer pairs to the user's query. Considering now non-educational systems, FRACT system [KLS07] performs automatic clustering on a set of previously introduced questions (query logs) to expand them. Then, the system matches the user query not only with the initial set but also with the expanded set of questions. Those query logs are easy to collect and they cover a large language. Other methods follow a hybrid statistical and NLP strategy. As an example we can observe the proposal of Kwok et al. [KEW01]. The NLP module uses a syntactic parser to classify the type of each question. After that, the statistical module takes part performing a comparison between keywords. This work was developed within the international Text REtrieval Conference (TREC) [Voo01] that promotes the design of FAQ retrieval projects. Next, the system proposed by Xue et al. [XJC08] calculates the probability of translation between Question/Answer pairs. OPTRANDOC system [AAOE11] implements a self-learning algorithm that automatically modifies the set of keyword terms using TF*IDF considerations by taking advantage of query logs. The Information Retrieval engine is implemented with a genetic algorithm. Finalizing, Minimal Differentiator Expressions (MDE) algorithm [MNCZ12] is a domain-independent CBR system. It represent each FAQ entry as a case composed of linguistic reformulations collected from users. In the training stage, the smallest multi-word expressions allowing the complete differentiation among cases are obtained. Later, those expressions are used to obtain the similarity score.

2.4 System overview

The main goal of FAQ retrieval systems entails retrieving precise answers. These answers are considered to be relevant to the user question and are usually presented as a ranked list. In addition, the knowledge representation of the domain should be interpretable and extendible.

Bearing this in mind, we have designed a FAQ Retrieval system which works in two stages. Initially, in an automatic learning stage, the system extracts information units from each element of the FAQ list. Hence, each FAQ entry of the collection is associated with a set of information units. These units contain the semantic information of each element. This process has to be carried out only when the FAQ list is modified. Once the system is initialized, the user can query it by means of NL questions. The preprocessor prepares each query to the subsequent steps. Then, the FAQ retrieval algorithm searches the most relevant FAQ entries in the collection, and presents them in the output interface. Figure iii.1 displays the modular architecture of our system.

In the following, we first discuss how is composed the FAQ lists and then we describe the main modules of our approach in detail. Since FAQ retrieval algorithm represents the key issue of this part of the research, it will be explained independently in Section 2.4.4.

2.4.1 FAQ list structure

Before explaining further details of the system, let us define some notation that will lead us formally define the structure of our FAQ list and its elements as well.

A FAQ list F is defined as a set of n FAQ entries. Each FAQ entry is composed by a set of questions (reformulations) and an answer, and is represented as $E_i = (Q_i, A_i)$, where $Q_i = S_i^0, \ldots, S_i^k$ is the the non-empty reformulations set of the ith entry. S_i^0 denotes the original question of the entry, and $S_i^j, j = 1, \ldots, k$ are the k linguistic reformulations collected from users question during the life of the system. Finally, A_i denotes the answer associated to the ith FAQ entry.

The number of reformulations in each FAQ entry depends on the life time of the system. The higher the quantity of reformulations, the better its syntactic variety is. However, the amount of

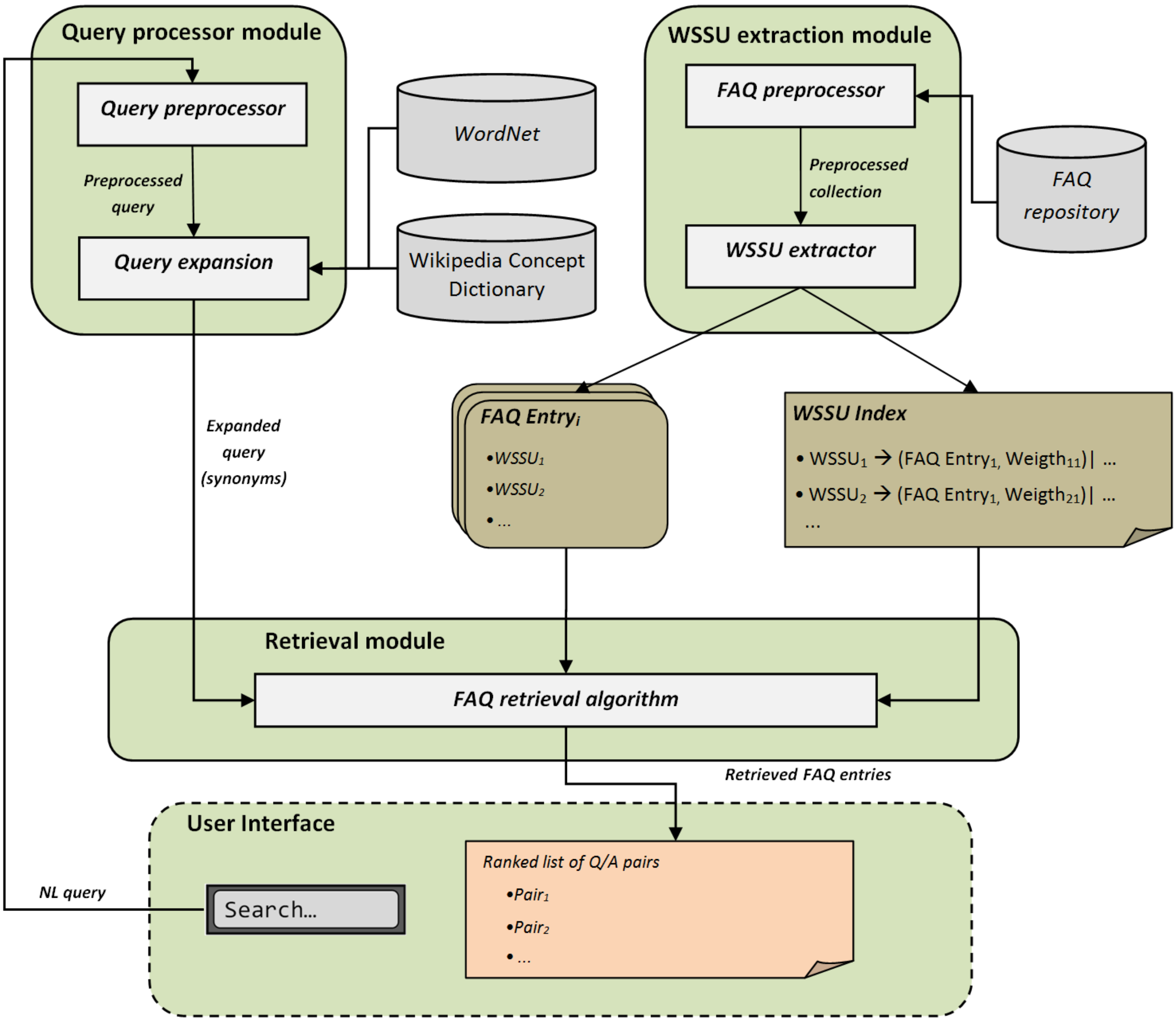


Figure iii.1: System architecture

reformulations could vary significantly from one entry to another (the amount of reformulations depends on the number of times in which an entry is queried).

2.4.2 WSSU extraction module

Domain modelling is an intensive time-consuming task. Also, any lack of information in the model will affect negatively the retrieval performance. In this regard, we have designed an automatic process to extract weighted information units, so-called Weighted Significant Semantic Units (WSSU). These units constitute the base of our system and allow us to design a quick and precise retrieval process. In this section, we describe in detail the role of the module in charge of extracting the information units.

Weighted Significant Semantic Units A $WSSU_i = (t_i, w_i)$ is a term-weight pair where each term t_i is extracted from one question, and it is an n-gram of the question holding semantic significance. The associated weight w_i represents the strength of the WSSU in the context of the FAQ list. To delimit which terms in a question have semantic significance, we have studied a FAQ list provided by the University of Granada. The university web page implements a Virtual Assistant that stores 3 FAQ collections containing over 5000 questions. FAQ entries in such lists have less than 9.45 words per question (and reformulation) in average. This means that the majority of the users include short queries to search the information, and therefore the most part of each question holds semantic significance. In this regard, we will apply a shallow parsing strategy to extract the meaningful fragment of a FAQ. To automatically measure the weight of each WSSU, a frequency-based strategy is followed (the more frequent a WSSU is in a set of questions, the less discriminative power it is with respect to a particular question of the set).

Extracting the WSSUs The first stage of the WSSU Extraction Module deals with the mining of each FAQ entry in the list to obtain their corresponding WSSUs. First, the FAQ Preprocessor performs a Part-Of-Speech (POS) tagging of each question of the list and then converts each word to lower case. The GPL Library FreeLing 3.0 was used to implement it.

Then, the WSSU Extractor performs a POS pattern matching to obtain two types of WSSU: (1) Action-WSSU (A-WSSU), composed by the 1-grams[3] obtained from the main verbs in the questions, and (2) Conceptual-WSSU (C-WSSU), composed by 1-grams, 2-grams, and 3-grams that match one of the patterns in Table III.1. The 1-gram of the A-WSSUs is lemmatized for reducing verbs to their common base form.

1-gram	2-gram	3-gram
Noun *Adjective*	*Noun + Noun* *Adjective + Noun*	*Noun + Noun + Noun* *Adjective + Noun + Noun* *Adjective + Adjective + Noun* *Noun + Preposition + Noun*

Table III.1: Part-of-speech patterns of Conceptual-WSSUs

Once the WSSU Extractor has captured all the WSSUs, it assigns their weights in order to measure its quality as classifiers. The weight of the unit λ_t is calculated using the $idf(t)$ term-weight criteria, where *idf* stands for 'inverse document frequency'. That is, the most frequent a term is in the collection, the less effective its discriminative power. We employ the *idf* criterion instead of other statistical ones because of the variability in the number of reformulations of each question in the FAQ list. For example, other criteria such as TF*IDF were originally designed keeping in mind large documents. For this reason we think that frequency criterion will not be appropriate in this context, where a sentence (about 9.45 words) represents a document. The weight of WSSU is obtained by means of the following equation:

$$\lambda_t = \frac{idf(t)}{log|C|} = \frac{log(\frac{|C|}{|\{Q \in C | f_Q(t) > 0\}|})}{log|C|} \tag{III.1}$$

where $|\cdot|$ computes the number of elements of a set, C is the set of FAQs in the collection, Q is a question in C, and f_Q counts the number of occurrences of term t in question Q. Eq. (III.1) measures the normalized inverse frequency of the number of questions that contain the term t.

At the ending step of this phase, two different resources are acquired: an dictionary and an index. The dictionary, called WSSU-dictionary, includes the set of all extracted WSSUs grouped

[3] The preprocessor detects phrasal verbs as chunks. In this way, we store each phrasal verb as a single 1-gram joining the verb with the particle by means of a underscore.

by each FAQ entry (from which were obtained). The index, called WSSU-index, stores a mapping between each n-gram of the extracted WSSU and the list of FAQ entry/weight pairs related to that WSSU. Table III.2 shows a simple example where the extracted WSSUs from two FAQ entries are disposed in the index structure (for the shake of simplicity, normalized *idf* weights were omitted). These resources will be used in subsequent stages of our system. This process is only carried out after each modification in the FAQ list.

FAQ Entry	**Questions**	**WSSUs**
pc_shut_down	Can I shut down Linux from a remote console? How can I shut down my computer?	'shut down' 'Linux' 'remote' 'console' 'remote console' 'computer'
boot_configuration	Can I configure the Linux boot? Can I modify the boot file in Linux?	'configure' 'Linux' 'boot' 'Linux boot' 'modify' 'boot file'

Index:

'shut down' → (*pc_shut_down*)
'Linux' → (*pc_shut_down*, *boot_configuration*)
'remote' → (*pc_shut_down*)
'console' → (*pc_shut_down*)
'remote console' → (*pc_shut_down*)
'computer' → (*pc_shut_down*)
'configure' → (*boot_configuration*)
'boot' → (*boot_configuration*)
'Linux boot' → (*boot_configuration*)
'modify' → (*boot_configuration*)
'boot file' → (*boot_configuration*)

Table III.2: WSSU extraction from Linux FAQ entries

2.4.3 Query preprocessor module

This module is responsible for preprocessing and expanding the user query to guarantee a correct coverage of the information requested by the user.

At the beginning of this phase, the module takes the user question as input. The Query preprocessor applies language-dependant NL steps to normalize the query terms. All words are converted to lower case, stop words are removed, and POS tagging is conducted. These steps are implemented by the GPL Library FreeLing 3.0.

After that, the query expansion process tries to obtain synonyms for the query terms. For each preprocessed query word (and its corresponding POS tag), the following procedure is carried out. If the word is a main verb, the term is replaced by its lemma. Otherwise, the system searches synsets (set of synonyms) containing the word in WordNet [Mil95a]. If that is the case, the words of these synsets are preprocessed and stored along with the query word. If no WordNet synsets are found, the same process is carried out employing our Wikipedia-based dictionary of concepts (Chapter II Section 2.3). If no synonyms are found in any of the two dictionaries, the query word is preprocessed and stored alone.

As a result, the query is composed by m expanded query words, $q = (ew_1, \dots, ew_m)$. Each expanded query word states as follows: $ew_i = (\{w_i\}, s_i^1, \dots, s_i^l)$, where $\{w_i\}$ is a set containing uniquely the ith word of the user query, and s_i^1 to s_i^l are the l synonym sets in which w_i is included. Table III.3 shows the expanded words for the query ‘Where is the professor Michael’s office?’ as an example. In this example, both the WordNet synonyms and Wikipedia synonyms are showed.

Word	Valid	POS tag	Expanded word set
where	N	wh-adverb	{-}
is	Y	verb	{{be}}
the	N	determiner	{-}
professor	Y	noun	{{professor}, $\{\text{prof}\}_{WN}$, {professors, university professor, …$\}_{WP}$ }
Michael	Y	proper noun	{{Michael}}
’s	N	possessive ending	{-}
office	Y	noun	{{office}, $\{\text{business office}\}_{WN_1}$, {agency, federal agency, bureau, …$\}_{WN_2}$, {function, part, role$\}_{WN_3}$, …, {offices, work office, …$\}_{WP}$ }

Table III.3: Expanded words of the ’Where is the professor Michael’s office?’ query

2.4.4 Retrieval module

The main role of the system is carried out by the Retrieval module. At the end of this phase, the user obtains a ranked list of Q/A pairs and a cloud of tags related to that list.

The FAQ retrieval algorithm receives the expanded query obtained in the previous stage, and the WSSUs extracted from the FAQ list. The algorithm tries to match each query word (and their synonyms) with WSSUs, and then it calculates a score for each FAQ entry in function of their corresponding matched WSSUs’ weights. Finally, a ranked list of FAQ entries (disposed as Q/A pairs) is obtained. The top-ranked elements of the list are suppose to be the best related to the user query.

The algorithm work-flow presents two main phases: (a) query reduction and candidate extraction, and (b) candidate weighting process. It takes three parameters as input: (1) the WSSU-dictionary, (2) the WSSU-index, and (3) an expanded query q. It is important to remember that a query contains a list of m expanded query words ($q = (ew_1, \dots, ew_m)$). In turn, an expanded query word is composed by a set of l' sets of synonyms ($ew_i = (s_i^1, \ldots, s_i^{l'})$). Such sets are composed by single words.

At the first phase, the expanded words of q will be reduced to only one set of synonyms, discarding the rest. In the same process, a subset of the FAQ list will be selected as candidate FAQ entries. Given the ith expanded query word of q, the algorithm iterates over each s_i^j, where $j \in 1, \ldots, l'$. Then, an exact matching between each synonym of s_i^j and all C-WSSUs included in the WSSU-index is performed. The matched C-WSSUs of s_i^j are employed to obtain all FAQ entries indexed by them. This FAQ entries are stored in a set $FE_{s_i^j}$. Once all the sets of synonyms of ew_i are explored, the algorithm looks for the $s_i^{j'}$ whose FAQ entry set $FE_{s_i^{j'}}$ presents the higher number of elements. Then the rest are discarded. Hence, ew_i will contain only $s_i^{j'}$, and it is associated uniquely to $FE_{s_i^{j'}}$. This procedure is carried out for all the expanded query words in q. Concluding, at the end of this phase, q is redefined as a list of synonym sets: $q' = (ew'_1, \ldots, ew'_m) = (\{s'_1\}, \ldots, \{s'_m\})$. In addition, the union of all FAQ entry sets associated to $(ew'_1, \ldots, ew'_m)$ conforms the final candidate FAQ entry set $CFE = \{FE_1, \ldots, FE_p\}$.

Table III.4 illustrates how an initial expanded query is reduced to a single synonym set, and the final candidate FAQ entry set obtained within this process.

In the second phase, the candidate FAQ entries are weighted by means of their WSSUs. Given a FAQ entry FE_i of CFE, the algorithm takes all its C-WSSU and A-WSSU from the WSSU-dictionary and stores them in a set $WSSUS_{FE_i} = \{wssu_1, \ldots, wssu_q\}$. Now, coincidences between q' and each WSSU of $WSSUS_{FE_i}$ are looked for in the following manner. Each unigram of $wssu_i$ should be in one synonym set of q', keeping the order. For example, if $wssu_i = [w_i^1, w_i^2, w_i^3]$ is formed by a 3-gram, and w_i^1 is contained in the jth synonym set of q', then w_i^2 has to be in the $j+1$th synonym set, and w_i^3 has to be in the $j+2$th synonym set. The process acts always in the same mode, even if a synonym is composed by more than one word. Let consider a synonym ('Granada university') contained in the ith synonym set, and the wssu 'Granada university students'. The two first words of the wssu ('Granada university') matches with the synonym. Therefore, the last word 'students' has to be in the $i+1$th synonym set. If any word of $wssu_i$ are not contained in the synonym sets of q' or the order relation is not satisfied, no match is produced and $wssu_i$ is removed from $WSSUS_{FE_i}$. This matching scheme is applied for all FAQ entries in CFE. Finally, using their WSSUs, a relation measure is computed for each one. We consider the relation function shown in (III.2), where FE is a FAQ entry of CFE, ω_{wssu_i} is a modification of the weight of $wssu_i$ (see (III.3)), and NW_{wssu_i} is the number of words in $wssu_i$ n-gram.

$$Relation(FE)_{wrtq'} = \sum_{wssu_i \in WSSUS_{FE}} \omega_{wssu_i} \tag{III.2}$$

$$\omega_{wssu_i} = \sqrt[NW_{wssu_i}]{weight_{wssu_i}} \tag{III.3}$$

The modification on the WSSUs' weight aims to increment the importance of a WSSU in function of its specificity. In more detail, if a given WSSU including two words is in $WSSUS_{FE_i}$, then another two WSSUs also including each of these words are also in $WSSUS_{FE_i}$ (see Section 2.4.2). Consequently, the more number of words, the more likely WSSU is 'specific' with respect to its corresponding FAQ entry. Then, the measure is slightly increased when the WSSU is 'specific'.

Once all FAQ entries of CFE are measured, those whose relation score is lower than a predefined threshold α are discarded. Finally, the valid entries are returned as a list sorted by weight. The α-threshold defines the minimum relation score of a FAQ entry to be considered sufficiently relevant to the user query. Moreover, results of our system are directly affected by this threshold. The higher

Query q: Where can I *find information* about the *UGR*?

Expanded words extracted from q:
$ew_1 = \{\{\text{find}\}\}$, $ew_2 = \{\{\text{information}\}, \{\text{information, data}\}, \{\text{information, warning}\}\}$,
$ew_3 = \{\{\text{UGR}\}, \{\text{UGR, Granada university}\}\}$

Candidate FAQ entries and C-WSSUs:
FE_1: {information, master, scholarship, requirement, Granada university, university, Granada}
FE_2: {information point, information, point, professor, contact data, contact, data, phone, UGR}
FE_3: {internet, registration process, registration, process, website, virtual assistant, virtual, assistant}

Synonym set	Synonyms	Coincident FE+WSSUs	Cardinal	Extracted FE candidates
ew_1	**{find}**	-	-	-
ew_2	{information}	FE_1:information FE_2:information point FE_2:information	3	$\{FE_1, FE_2\}$
	{information, data}	FE_1:information FE_2:information point FE_2:information FE_2:contact data FE_2:data	**5**	
	{information, warning}	FE_1:information FE_2:information point FE_2:information	3	
ew_3	{UGR}	FE_2:UGR	1	$\{FE_1, FE_2\}$
	{UGR, Granada university}	FE_2:UGR FE_1:Granada university	**2**	

Output:

$q' = (\{\text{find}\}, \{\text{information, data}\}, \{\text{UGR, Granada university}\})$
$CFE = \{FE_1, FE_2\}$

Table III.4: Example of expanded words reduction and final candidate selection

the threshold, the more precise the answers are. In turn, lower values, will allow users to express their questions in a more flexible manner. To choose an appropriate value for this parameter, an independent empirical study will be discussed in Section 2.5.3.

Table III.5 continues the previous example, showing the output FAQ entry list obtained in the complete process. It has to be pointed out that A-WSSUs are included at this step. As could be observed, the weights of the WSSUs are not calculated considering only the specified FAQ entries. Weights correspond to real weights of each WSSU in our UGR dataset.

Expanded query $q' = (\{\text{find}\}, \{\text{information, data}\}, \{\text{UGR, Granada university}\})$

Candidate FAQ entries and WSSUs:

FE_1: {find:0.758, exchange:0.758, information:0.155, master:0.66, scholarship:0.477, requirement:0.722, Granada university:0.317, university:0.212, Granada:0.31}

FE_2: {find:0.758, search:0.719, look_for:0.424, information point:0.879, information:0.155, point:0.808, professor:0.616, contact data:0.619, contact:0.786, data:0.758, phone:0.394, UGR:0.218}

FAQ entry	WSSU	Match w.r.t. q'	ω_{wssu_i}	$Relation(FE)_{w.r.t.q'}$
FE_1	find	Y	0.758	1.476
	exchange	N	-	
	information	Y	0.155	
	master	N	-	
	scholarship	N	-	
	requirement	N	-	
	Granada university	Y	0.563	
	Granada	N	-	
	university	N	-	
FE_2	find	Y	0.758	**1.889**
	search	N	-	
	look_for	N	-	
	information point	N	-	
	information	Y	0.155	
	point	N	-	
	professor	N	-	
	contact data	N	-	
	contact	N	-	
	data	Y	0.758	
	phone	N	-	
	UGR	Y	0.218	

Output:

Top FAQ Entry = FE_2

Table III.5: Example of output FAQ entries

2.5 Experiences with the System

This section reports the empirical results obtained in the evaluation of our system. The validity of our FAQ retrieval algorithm has been contrasted to other state-of-the-art methods. To this end, we will first evaluate the performance of our FAQ retrieval approach. The datasets and the metrics we use for this purpose will be described below.

2.5.1 Dataset

We test our methods with three FAQ lists from different domains. These FAQ lists were already employed while evaluating other related systems [MNCZ12].

- **Restaurant FAQ list**: This FAQ list is extracted from the domain of a Restaurant and it contains entries about reservations, prizes, menus, etc.
- **Linux V.2.0.2 FAQ list**: This FAQ list is a public (and available in[4]) list of entries about the Linux operating system.
- **UGR FAQ list**: This FAQ list belongs to the Virtual Assistant on the University of Granada web page [5]. It stores more than 5000 questions. Consequently, we randomly selected 5000 questions to perform the experiments.

For each FAQ list, a subset of equal proportion and size to perform 10-fold cross-validation was selected. The number of FAQ entries, the number of reformulations, the number of training and testing set, and the average of reformulations for each entry are displayed in Table III.6.

	FAQ entries	Reform.	Density (ref/entries)	Training sets	Testing sets
Restaurant FAQ rep.	39	400	10.26	360	40
Linux V.2.0.2 FAQ rep.	59	450	7.63	400	45
UGR FAQ rep.	310	5000	16.13	4500	500

Table III.6: Details of FAQ list

2.5.2 Design principles

Our system and the comparison algorithms were implemented with the following architectural design principles. We use a MySQL database (version 14.12), and the Hibernate[6] library (version 3.3.5), as framework for mapping the dictionaries. We use Java SDK 6 (1.6.0_20) to implement the system and the comparison algorithms. Finally, an Intel(R) Core(TM) i5 processor 430M (2.26Ghz) and 4GB RAM was used for the evaluation.

2.5.3 Setting parameters

The α-threshold determines the minimum degree of weight for a FAQ entry to be considered properly related to the user query (see Section 2.4.4).

To select the parameter α we carried out some preliminary experiments. We set different values for α (between 1.0 and 2.0) and we observed its impact in the results. We have evaluated the statistical significance of these experiments by means of 10-fold cross-validation for each FAQ list and α setting. $\alpha = 1.3$ was empirically verified as the most appropriate balance between precision (precision varied about 3% while ranging α between 1 and 2) and frequency of response (with the maximum value, the system retrieve approximately 20% less answers with respect to the maximum) in our datasets.

Regarding the comparison algorithms in FAQ retrieval task, we took the same parameters reported by their authors in their corresponding papers.

[4] http://www.linux-es.org/Faq/Html/

[5] http://tueris.ugr.es/elvira/

[6] http://www.hibernate.org/

2.5.4 FAQ Retrieval experiments

In order to evaluate the performance of our FAQ retrieval method we carried out 10-fold cross-validation for each FAQ list. We complete ten validations in order to obtain significant results to confirm the results with a t-test. In each validation, the data set in consideration is divided into ten mutually exclusive subsets of the same size. Each fold is used to test the performance of the FAQ retrieval method, using the combined data of the remaining nine folds to extract the WSSUs.

Then, we employ two well-known metrics in the literature: Precision and Mean Reciprocal Rank (MRR). Precision computes the proportion of correctly returned FAQ entries (answers) with respect to all the retrieved entries (see (III.4)).

$$Precision = \frac{|\{correct\ answers\} \cap \{retrieved\ answers\}|}{|\{retrieved\ answers\}|} \tag{III.4}$$

MRR is a measure for evaluating any process that produces a list of possible responses to a query, ordered by probability of correctness. The reciprocal rank of a query result is the multiplicative inverse of the rank of the first correct answer. The MRR is the average of the reciprocal ranks of results for a sample of queries (see (III.5)):

$$MRR = \frac{1}{T} \cdot \sum_{t \in T} \frac{1}{rank(t)}\ , \tag{III.5}$$

where T is the entire testing set and $rank(t)$ computes the rank of the top-ranked FAQ entry given by the user query t.

The whole process has been repeated 10 times in order in order to diminish the dependence of the results with respect to a concrete partition of the reformulation set. We also compare our results with the following state-of-the-art methods:

- **TF*IDF** [SM86] measures the similarity between two word frequency vectors $\overrightarrow{U}$ and $\overrightarrow{S}$. This scoring function weights each word in terms of its frequency in a document (*TF*) and its inverse document frequency (*IDF*).

- **Adaptive TF*IDF** [SM86] is an improved version of the TF*IDF method. The main difference is that this method performs a hillclimbing for each word weight to bring a question and its corresponding answer “closer”, raising the score function.

- **Query Expansion** [BCC$^+$00] aims to bridge the lexical chasm between questions and answers. The key point of this method consists of adding some words to the query which are likely synonyms of (or at least related to) words in the original query. These words are added by calculating the mutual information between query terms and answer terms in the training set.

- **FRACT** [KLS07] is a cluster-based system. It clusters the so-called *query logs* into predefined FAQ entries and extracts weight scores of potentially occurring words from the clusters by using a centroid finding method based on LSA techniques. It represents FAQs and query logs in latent semantic space and uses the vector-similarity function to compute the closeness scores between FAQs and query logs. Then, the clusters are used as a form of document smoothing during retrieval.

- **Co-occurrence Model** [Jua10]: This method takes advantage of word co-occurrence corpus (semantic model) to improve its ability to match questions and answers through a question similarity measurement. Similarity is based on the number of relative terms and the length of the query sentences.

- **Rough Set Theory** [CPC08]: This algorithm combines hierarchical agglomerative clustering method with rough set theory to address the problem of FAQ retrieval. The lower/upper approximations to a given cluster are used to classify users queries.

- **EMD algorithm** [CPC08] performs a search on the space of words combination, keeping the smallest sets of words that differentiate one question from the rest of questions in the collection. Those sets conforms the expressions that are weighted reflecting their potential as classifiers. The EMD are employed to perform the similarity measure in the retrieval process.

2.5.5 FAQ Retrieval results

We present the experimental results described in the previous subsection. Tables III.7-III.9 summarizes the results for the different FAQ retrieval algorithms with respects to Precision and MRR measurements for each collection.

Measure	1	2	3	4	5	6	7	8
Precision	WSSU	MDE	Ad. tf·idf	tf·idf	Q. Expansions	Co-Model	FRACT	RoughSet
	0.9310	0.8236	0.7481	0.7452	0.7364	0.6843	0.6795	0.6745
MRR	MDE	Ad. tf·idf	tf·idf	WSSU	Q. Expansions	Co-Model	FRACT	RoughSet
	0.9138	0.8493	0.8470	0.8401	0.8328	0.7820	0.7526	0.7454

Table III.7: Ordered rank of each method in Restaurant FAQ

Measure	1	2	3	4	5	6	7	8
Precision	WSSU	MDE	Ad. tf·idf	tf·idf	FRACT	RoughSet	Q. Expansions	Co-Model
	0.9289	0.7226	0.7087	0.7060	0.6707	0.6633	0.6515	0.6390
MRR	MDE	WSSU	tf·idf	Ad. tf·idf	FRACT	Co-Model	Q. Expansions	RoughSet
	0.8661	0.8331	0.8081	0.8076	0.7549	0.7548	0.7513	0.7417

Table III.8: Ordered rank of each method in Linux FAQ

Measure	1	2	3	4	5	6	7	8
Precision	WSSU	MDE	FRACT	Co-Model	Ad. tf·idf	tf·idf	Q. Expansions	RoughSet
	0.8089	0.8059	0.7547	0.5969	0.5475	0.5466	0.5371	0.5520
MRR	MDE	FRACT	WSSU	Ad. tf·idf	tf·idf	Q. Expansions	Co-Model	RoughSet
	0.8853	0.8109	0.8095	0.7211	0.7205	0.7087	0.6852	0.6325

Table III.9: Ordered rank of each method in UGR FAQ

In this study, we have focused on high-precision. In this regard, results corroborate our expectations – WSSU outperformed all comparison methods in terms of precision. However, this is not the case for MRR, where MDE has proven to be the most reliable. In any case, it is worthwhile remarking that, even in MRR, our algorithm showed comparable results. In this regard, we have also measured our system in terms or MRR_α. This measure is a simple variation of the classical MRR measure that considers only the cases when the system produces an answer. We obtained a MRR_α value of 0.9178 on the Linux dataset, 0.9537 on the restaurant dataset, and 0.8754 on the UGR dataset. The MRR_α results demonstrate that every time the algorithm produces an answer, the output is likely to be correct. Next, MDE shows the highest global average in MRR and it has good performance in terms of precision too. However, even if MDEs algorithm was proven to behaves quasi-linearly in terms of time complexity (see [MNCZ12] and [MRCZ12] for more details) it takes about 60 second while training the UGR dataset. In this regard, it should be remarked that our algorithm takes only 15 seconds for the same dataset. Furthermore, Query Expansion algorithm obtained low precision because of over-training. In turn, the Co-occurrence model performs better when the set of reformulations is increased, as well as FRACT algorithm. Finally, the RoughSet model shows poor results due to the fact that the size of the output clusters is usually too big.

In addition, we have performed a t-test with a confidence level of 95% for each pair of algorithms to assure a significant improvement of the results, obtaining a p-value < 0.0001 in all cases.

2.6 Conclusions and Future Work

In this paper, a FAQ retrieval system is presented under e-learning scenario. FAQ collections could be too large to be organized by the manager of the VLE (usually the teacher). Therefore efficiency in terms of time complexity becomes a paramount issue to this research. In addition, traditional FAQ retrieval systems need for knowledge modelling and expert support, being this a strong limitation in e-learning.

These requirements motivated us to design a system focusing in two main aspects. On the one hand, answers should be precise. Experiments evidence that our method outperformed the comparison algorithms in terms of precision, in small and large collections. In addition, our system is comparable in terms of MRR measure, even for large datasets. The functionality of our system relies on the set of semantics units extracted from the collection. We have implemented an automatic method to extract and organize weighted significant semantic units (WSSU) from each FAQ entry of the collection. Hence, since our system does not require any knowledge modelling or expert support, it can be easily adapted to new domains.

The knowledge is represented in such a way that its interpretation becomes easy. Frequency-based techniques were here applied to automatically weight each semantic unit. In addition, following a NLP methodology, our system incorporates a query expansion module that takes advantage of WordNet and Wikipedia to enhance the NL query after the retrieval process. It aims to improve the language coverage of the possible wordings users could employ when querying the system.

However, even if results are encouraging, there is still much work ahead. Since the technique described here is not tailored to the FAQ retrieval problem, we believe that our proposal could be extended to new resources with minimum effort. We plan to apply the model with a set of different learning resources. More concretely, we will apply our WSSU algorithm on two collections of learning resources including GIFT quiz questions and collaborative tasks (Chapter II Section 2.6.1).

3 Integrating Knowledge and Information Visualization Techniques in Virtual Learning Environments

3.1 Introduction

In today's Information Age, we are witnessing an explosion of information and knowledge on the Internet. Although information technologies enable users to access large amount of resources across geographical boundaries, this proliferation has led to the commented information overload issue — having more information available than one can efficiently process. Tackling this problem has thus become particularly interesting for the research community.

An adopted approach of how to cope with information overload is to provide with appropriate visualization techniques. Visualization of knowledge and information are widely applied in the fields of education and knowledge management to help users in processing, getting access, and dealing effectively with complex knowledge and large amounts of information [KT05]. As main reason, visualization techniques deal with the human cognitive processing system. According to Mayer [May02], visualization involves cognitive processing in many subsystems of the human working memory and therefore supports processes of learning. In addition, visualization techniques can enhance our processing abilities by visualizing abstract relationships between visualized elements and may serve as basis for externalized cognition [SR96, Cox99].

Attending to literature we can find that knowledge visualization and information visualization have been treated as two distinct fields of research [JLH05, KT05, CCH^{+}05]. *Knowledge Visualization*(KV) has its origins in the social sciences, particularly in the field of learning and instructional science. It deals with techniques for external representations of individual knowledge in a visual-spatial format. Normally the knowledge is represented as concepts and relationships between concepts. A commonly accepted definition of knowledge visualization was proposed by Buckhard and Meier [BM04]. They defined KV as the use of visual representations to transfer knowledge between at least two persons. In contrast, *Information Visualization* (IV) is a more recent field that emerges from computer science. Regarding the scope of this work, IV is considered as a technology for visualizing abstract data structures and their relations. The type of information visualized depends on both the underlying data type and the users' needs [Shn96].

Nonetheless KV and IV share a common core: based on mapping rules, resource objects are translated into visual objects as meaningful representations, offering easy and comprehensive access to the subject matter presented [JLH05]. Therefore KV and IV should be able to collaborate between each other in order to achieve a common objective: mapping tools could display conceptual knowledge and content knowledge within one and the same visual environment. Visual representations of concepts representing the domain could serve as navigational tool that provides knowledge-based access to information.

We base on the above idea for developing two visualization techniques for representing the underlying knowledge of the domain, and for linking the knowledge representation with the educational content. To that end, we focus on the two well-know visualization techniques: tag clouds and concept maps. So far, we have faced the automatic acquisition of the underlying knowledge from the educational content. In Chapter II we discussed three different methods for obtaining three well-known meta-data structures: taxonomies, folksonomies, and ontologies. Here we take advantage of folksonomies and ontologies in order to design such visualization methods. First, tag cloud is a visualization technique commonly used on tagging systems. The most popular tags stored in the folksonomy are displayed in a cloud representation. In this kind of visualization technique, the tags are links to the resources of the system, allowing a easy and intuitive navigation through them. Second, concept maps are graphical tools for representing concepts and relations between them, having the aim of organizing a student's cognitive structure to encourage a deep level of integrated knowledge. Since this technique involves the representation of concepts and relations, it could be supported by ontologies. In this way, concept maps can be directly translated from the main concepts and relations of ontologies. We extend these two techniques in order to facilitate both the domain's overview and the navigation through the educational content.

The rest of this section is organized as follows. We first comment the main features of both techniques in the following two subsections. Subsection 3.2 offers a review of the state-of-the-art in the fields. Next, we present the methods for obtaining tag clouds and concept maps in Subsection 3.3. Subsequently, we offer concrete examples of applicability of both techniques in Subsection 3.4. We discuss the validity of the corresponding methods in Subsection 3.5. Finally, Subsection 3.6 concludes with a discussion of results and future research.

3.1.1 Tag Clouds

The concept of *tag cloud* became popular on Web 2.0 due to their ability to provide a fast overview of a given domain. Tag clouds are visual displays of descriptive terms, also called keywords or *tags*. These descriptive terms are annotations over digital resources, such as bookmarks[7], pictures[8], or products[9].

In this kind of visualization technique, the tags are links to the resources of the system, allowing a easy and intuitive navigation through them. In a tag cloud, textual attributes such as size, color or font weight are used to represent the association strengths among different tags or between tags and resources (Figure iii.2. According to Rivadeneira et al. [RGMM07], the following functions from a tag cloud model of visualization are expected:

- Search and retrieval of a specific term or concept of the domain.
- Sallow exploration of the whole domain.
- Understanding about the levels of importance of each concepts throughout the domain.

Concluding, tag clouds represented a suited tool for bringing together the strengths of knowledge and information visualization. Their capacity to display the overview of the domain might result in a better understand not only of the meaningful concepts of the domain, but also of the importance of each one within the educational content. In addition, students can easily navigate through the visual structure getting direct access to the related learning resources.

3.1.2 Concept Maps

A concept map is a visual representation of different concepts and their relationship. Precisely, their creators Novak and Gowin [Nov84] defined concepts maps as

> "graphical representations of knowledge that are comprised of concepts and the relationships between them".

[7] http://delicious.com

[8] http://www.flickr.com/

[9] http://www.amazon.com/

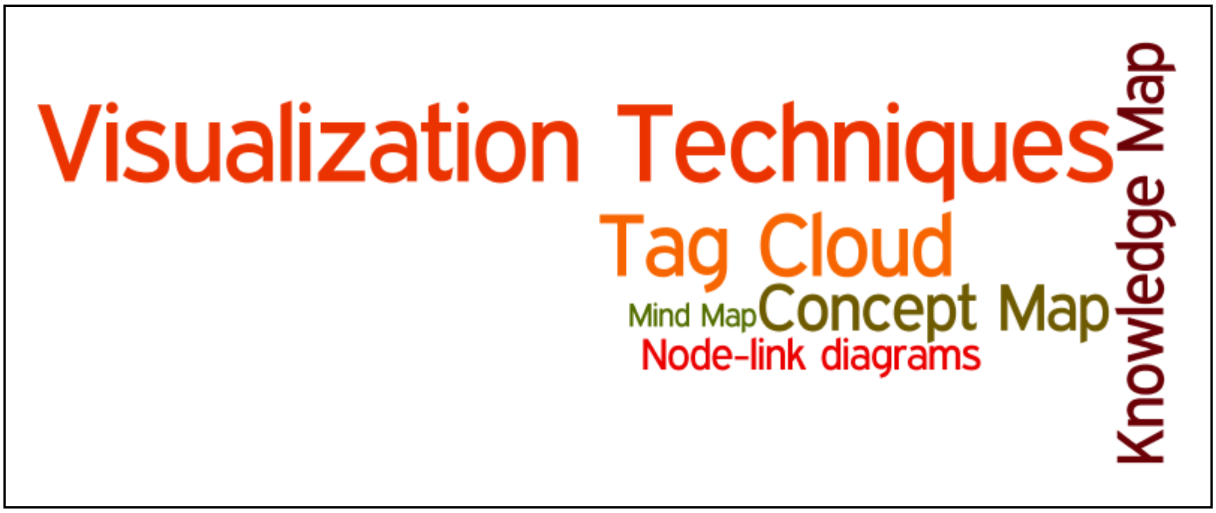


Figure iii.2: Tag Cloud visualization technique

This kind of visualization technique organizes the student's cognitive structure to foster a deep level of integrated knowledge. That is, concept maps provide a visual representation of the knowledge that student should have about a specific topic. In fact, concept maps are graphical tools that enable anybody to express their knowledge in a form that is easily understood by others. The visual representation can be used to illustrate thoughts, ideas, or planned actions that arise from a group of stakeholders on a particular issue. It consists of nodes, containing a concept or item, and links connecting two nodes to each other and describing their relationship, where each node-link relation makes a proposition.

In this scheme concepts are usually represented within boxes or circles. Concepts are connected by directed arcs encoding brief relationships. Traditionally, the concepts were disposed in a hierarchical manner. The vertical axis expressed a hierarchical framework for the concepts. More general and inclusive concepts were found at the highest level, with progressively more specific and less inclusive concepts arranged below them. Nevertheless, today is accepted that the topology of the a concept map can take a variety of forms, ranging from hierarchical to non-hierarchical and data-driven forms [ZKM12]. Figure iii.3 shows an example of a concept map about animal groups.

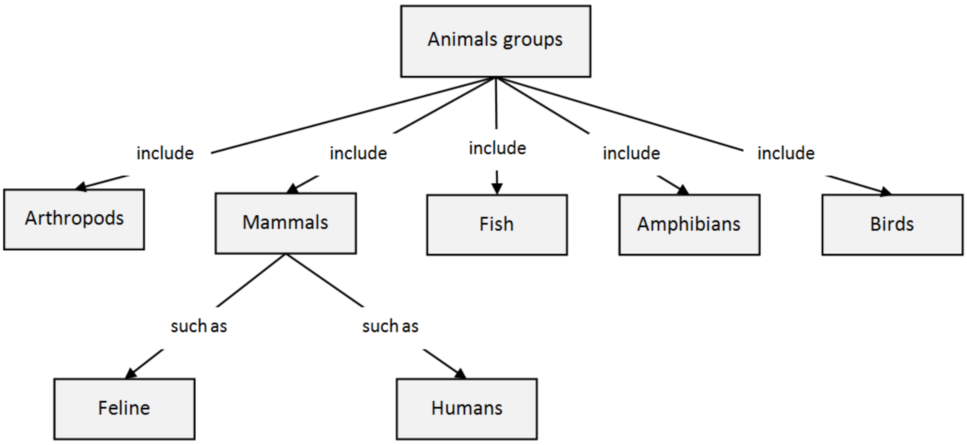


Figure iii.3: Concept Map example

The structure of a concept map depends on its context. In this way, maps having similar concepts can vary from one context to another. This characteristic makes concept maps a great resource for supporting learning. Besides the benefits of using concept maps from a educational perspective (commented in the subsection below), we view this scheme as a excellent tool for organizing and facilitating the access of the educational content. If the concepts would be linked to the learning resources related to them, students could learn the knowledge and understand the role of each concept within the course at the same time.

Fortunately, in this work such prerequisite is already fulfilled. In Chapter II Section 4 we proposed a method for acquiring with the minimum intervention of the teacher a lightweight domain ontology from the educational content. The resulting ontology contains the main concepts and relations underlying form the educational content. What is more, some of those relations are specific of the educational context. As it can be seen, there exists a direct relation between ontologies and concept maps [CLT11]. Although the motivation of each technology is different, in our work both of them shares common context and goals. Therefore our lightweight educational-domain ontology model is able to represent and store concept map information related to the same context. In consequence, we have the necessary elements to perform a direct translation of the acquired ontology into a concept map representation. Summarizing, this strategy allow the system to automatically obtain a concept map representation of the domain. On the one hand, this might foster the students' comprehension of the knowledge. On the other hand, the learning resources associated to each concept of the map are directly accessible, i.e. the navigation through the educational content might be improved.

3.2 Related Works

We discuss, in this section, the different researches carried out on the fields of tag clouds and concept maps. In addition, we comment other kind of techniques used for knowledge and information visualization.

3.2.1 Tag clouds

The tag cloud model has the main goal of visualizing the overview of the main elements within a context. For example, Xexeo et al. [XMF09] proposed the use of tag clouds to represent a conceptualization of a given text. In this sense, they define the idea of *summary tag cloud* to summarize the complete content of a whole document collection. A tool designed to visualize textual sources in form of clouds was presented in [VWF09]. The tool so-called Wordle, extracts words directly from texts, and it is currently very popular. Another example of this kind of technique can be consulted in [Zub09]. Zubiaga designed a system for extracting tag clouds from the articles of Wikipedia, and concluded that it was useful to improve the navigation and search through this social encyclopaedia. The use of tag clouds has been also employed even in a political scenario [KM10]. In this work, transcriptions of political meetings were summarized by means of tag clouds. Also the model has been used for refining the search results, for example, from web queries [KHGW07], or databases [KZGM09]. In this approach, the system offers a tag cloud representation of the results obtained through the search query, helping the user to better understand the scope of the query.

More concretely, state-of-the-art researches on tag cloud are easily classifiable into three different fields: tag cloud selection, tag cloud visualization and tag cloud evaluation. We proceed to discuss each field below.

In first place, the *tag selection* (or *tag ranking*) task address the problem of selecting tags to summarize query results. Tag clouds are been used recently for supporting search systems. Their main purpose is to provide the user with visual approximations of the results obtained by user queries. It is plausible that in some circumstances the user does not have enough knowledge of the domain to input a well-defined query that exactly retrieves the needed information. It might also occur that the user does not even know which concrete information is looking for. In those cases, the user could take advantage of an overview of the most related tags to the initial query. Moreover, the proper tag cloud could facilitate the user to refine the initial query [MHSGB10]. A number of different approaches have been performed to deal with tag selection task. As pre-initial study, Halvey and Keane [HK07a] tested a random and alphabetical selection method for ordering a set of tags in a tag cloud. More recent, Venetis et al. [VKGM11a] presented an algorithm combining frequency and diversity to increase the coverage of the query results. Furthermore, a set of metrics to measure the structural properties of tag clouds was discussed. As part of these metrics, authors defined the concepts of *coverage* and *overlap* of tag clouds (that we will employ to perform our experiments). The work in [SA11] proposed a set of tag ranking strategies and a set of metrics to evaluate them. Authors concluded that diversification methods achieve the best performance for all the proposed metrics. Finally, a combination of clustering methods for cloud generation was commented in [LDLD12]. The study showed that extending the cloud generation based on tag popularity with clustering slightly improve coverage. In addition, authors proved that a cloud generated by clustering independently of the tag popularity baseline minimize overlap and increase coverage.

In second place, there exists a line of research that address the different *tag cloud visualization* schemes used to represent tag clouds, and the impact of the possible layout attributes on the user experience. In [KHGW07], both cloud and list layouts were compared. The authors established two measures to weight the correctness of response and response time. They concluded that clouds led to longer response times than lists, but improved correctness. Also authors found that users gave better answer to overall question when using tag clouds. Another comparative study between cloud and lists layouts was discussed by Halvey and Keane [HK07b]. Here, a group of participant tried to find a item within a visualization of ten items. They found that the item was found in more time in clouds than in lists. Furthermore, the conclusions revealed that items with larger font sizes and in the upper-left corner of the tag clouds were recognized faster than the other items. In this way, Rivadeneira et al. [RGMM07] deduced that tags with larger fonts were recalled better and recognized more quickly by comparing different tag cloud layouts. Next, the study commented in [SCH08] performed a comparison of web searching with and without tags. As main results, the researchers observed that users preferred tag clouds when browsing for general and unspecific information, but they preferred traditional search interfaces when searching for specific information. Finalizing, a comparative analysis among a number of layout attributes (font size, tag area, number of characters, tag width, font weight, colour, intensity, number of pixels) was carried out in [BGN08]. To that end, authors designed a measure to test their effect on link selection. The stronger effects in link selection were produced by font size and font weight.

In third place, a number of works have focused on *tag cloud evaluation* for measuring the effectiveness of the results. Venetis et al. [VKGM11b] defined some metrics for measuring the structural properties of tag clouds: extent, coverage, overlap, cohesiveness, relevance, independence, and balance. The metrics were defined in order to test the utility of tag clouds when they are employed for summarizing query results. In the same way, Durao et al. [DDLL12] stated that coverage, overlap and relevance are the most important metrics. Authors in [SA11] additionally proposed the "selectivity" metric measuring the number of filtered objects on a tag cloud when a tag is selected. Finally, Venetis et al. conducted an additional research in [MKWS12]. They argued that some of the commented metrics present conflicts among them. For example, if tag cloud generation process tries to increase the coverage, the overlap is also increased. This fact is negative due to overlap should be as minimum as possible.

3.2.2 Concept Maps

Most of state-of-the-art studies on concept maps rely on the educational perspective. Here we discuss a number of considerations.

Initially, concepts maps were invented by Novak and Gowin [Nov84], based on the ideas of Ausubel [Aus63]. Ausubel thought that the individual's subject-matter knowledge is mentally represented in a hierarchy of concepts. Novak and Gowin defined concept maps as spatial arrays that represent elements of knowledge by means of nodes and directionally labelled or named links, the nodes representing ideas, concepts and beliefs and the links relations between them. Students who use this knowledge representation acquire a meaningful and interconnected learning. Students "learn how to learn" more effectively [Nov84].

More recently, the benefits attached to concept have continued to grow. Kommers and Lanzing [KL97] suggested that concept maps take advantage of the capabilities of the human visual perception system and the benefits of visual information representation. These benefits include ease of recognition, the possibility to quickly scan a picture and find differences or keywords, compactness of representation, and the observation that it seems to be easier to keep an overview. In the same way, Cox and Brna [CB95] commented that our processing ability can be enhanced by visualizing abstract relationships of knowledge. There is a great deal of evidence indicating that the use of concept maps is a valuable strategy for supporting cognitive processing in a variety of learning settings, e.g. [Ter05, Nov90, BB00]. According to Kinchin [Kin11], the use of concept maps makes it possible to relate the structure of the curriculum to the structure of the discipline, in order to support the development of robust student knowledge structures in ways that reflect the professional practices of subject experts. In addition, this method provides a visual and holistic way for sharing ideas in an accessible and concise form, thereby turning knowledge-sharing into a promising model [WCLK08]. Finally, Chiou et al. [CLL12] verified that concept map technique significantly improves students' short-term learning achievements.

Considering now the organization of the educational content, concept maps still present benefits. Chang et al. [CSC03] found that using the maps can reduce students' problems of disorientation when browsing linear-structured material. Their study results indicated that internet material using graphic displays fashions can reduce problems of learning disorientation and improve learners' learning outcomes. More recent studies evidence the same results [HCC$^+$12]. Therefore applying concept maps to structure teaching material enables knowledge to be structured and integrated in a hierarchical order [Cof07].

It should be pointed out that there exists a line of research directly related with the automatic creation of concept maps from documents. The study of this field would be very helpful for us in case that we would not count with a ontology learning method. Nevertheless, our objective here is not to automatically mine concept maps from text. In contrast, we focus on the translation of a ontology into a concept map structure. There exist four main approaches for extracting concept maps from documents. (1) Statistical methods try to analyse the frequency of terms and their co-occurrence in a document. They tend to be efficient and portable but imprecise, because the semantics of terms are not considered. These methods are commonly employed combined with other approaches such as machine learning or NLP. Some examples are [CHB$^+$06, CK04, VC09]. (2) Machine learning methods are used for extraction of concepts and relationships from unknown data. Classification, association rules and clustering are the techniques most commonly used in this process [LLL09, RT02]. (3) Dictionary-based methods use ontologies and lists of predefined terms as a seed in order to detect concepts and relationships more precisely [Coo03, CDF$^+$07]. For example, given a term, this approach look for related terms and relationships that most frequently occur with it across the document collection. Another example of application consists of using the dictionary as controlled vocabulary. Also the dictionary could be used for grouping words into clusters. Finally, (4) NLP techniques such as lemmatization or stemming extends basic statistical and data mining approaches [KGRS10, LSL$^+$09].

3.2.3 Other Visualization Techniques

Apart from tag clouds and concept maps, there exist some other techniques for creating concept visualizations. We comment the main characteristics of some of these techniques below.

Node-link diagrams [KL09, GFC04] exhibit multiple connections among shared ideas and uses shapes to enclose concepts [KL09]. This visualization technique allows users to organize concepts sequentially or hierarchically. In this kind of diagrams, nodes are represented by shapes where features (colour, size and position) indicate hierarchical conceptual importance, and lines establish relationship among nodes. Different visualization tools rely on mapping structures [KL09]. Mapping draws upon node-link structures for establishing concept location and guiding navigation in abstract structures. This technique requires visualization features to specify semantic relations among terms and to relate concepts. Mind maps and knowledge maps are the most employed mapping structures to the purpose. Mind maps [KL09, BL11, CBOC11] support collaborative brainstorming with and organized visual structure. In that structure, each key concept is located at the middle of the page. Then, several related main topics in different colours are radiated out in the shape of thick branches. Attached to these main branches, other smaller branches represent related concepts. In this way, related words are associated through curved main and sub-branches. The main difference between mind maps and concept maps is that in mind maps, concepts and ideas are represented, without signifying the particular meaning imposed on the relationships [BB93]. Finally, knowledge maps [Ter05, LWZ$^+$12, LS12] are a two-dimensional graphical display to present information, as well as concept maps.

3.3 Obtaining Visual Representations from the Educational Domain

In this section, the methods employed to automatically construct tag clouds and concept maps are discussed. First, we tackle the generation of tag clouds. Subsequently, our concept map approach is presented.

3.3.1 Tag Cloud Generation Algorithm

In this part of the dissertation, we handle the generation of the tag cloud as result of a user query, following the tendency of the state-of-the-art. This approach receives a set of resources and a set of tags mutually linked between each other, and responses with a tag cloud representation. Each resources in the input set presents a weight that measures the level of relation of the resources against the user query. However, the search method is not considered here[10]. In addition, the method is easily extendible for working with no query. In that case, the algorithm only needs to

[10] Any search model which returns a set of resources weighted by their importance regarding the user query could be applied first. Logically, the returned set of resources should be linked to the tags in the knowledge base

consider a hypothetical query that obtains the whole set of resources. That is, the complete set of resources would act as input.

Additionally, we will apply the algorithm on the FAQ retrieval and social tagging system proposed in this Thesis (section 3.4). It allows us to provide two complete frameworks of application. The first framework will consider the retrieved results related to a user query from the FAQ retrieval engine. The second framework extends the tag selection algorithm without considering any query, i.e. considering the whole set of resources in the system. What is more, by means of theses frameworks we will be able to evaluate our tag generation algorithm.

Tag selection algorithm The tag selection task is defined as follows. Let us assume a set of (learning) resources O, and a set of tags T. There exists a mapping function $m_t : O \times T \rightarrow \{0, 1\}$ mapping tags to resources. $T(o)$ denotes the set of tags assigned to the resource o, and $O(t)$ denotes the set of resources tagged with t. Now, we will assume that a query q belongs to T. Given a query q, the set of results is a subset $O_q \in O$, and the set $T(q)$ of all tags related to it can be defined as the union of the sets of tags assigned to O_q:

$$T(q) = \bigcup_{o \in O_q} T(o) \tag{III.6}$$

Thus, the goal consist of finding a subset $T_q \subseteq T(q)$ of size k sufficiently meaningful to summarize and expand the information related to a particular query q. The tags in T_q should be ranked, since this ranking will be used in the tag cloud visualization task.

Let $f_q(t)$ be a scoring function that assigns an utility score in the interval [0, 1] to each tag in T_q. Finally, given a query q and an integer K, the optimal tag cloud for q is the set of T_q with size K (see Section 3.5) that maximizes the following function:

$$F(T_q) = \frac{\sum_{t \in T_q} f_q(t)}{|T_q|} \tag{III.7}$$

Then, the main concern is on how to define the utility function $f(t)$. Let TOP_q be the top-ranked resource in O_q, and T_{TOP_q} be the set of tags corresponding to TOP_q. The resulting tag cloud will be called T_C. We define the utility function of a tag t as follows:

$$f(t) = \max\{r(t, t_r) | t_r \in T_C\} \cdot \max\{s(o, T_C) | o \in O_q(t)\} \cdot weight_t \ , \tag{III.8}$$

where

$$r(t_i, t_r) = \frac{|O_q(t_i) \cup O_q(t_r)|^2}{|O_q(t_i)| \cdot |O_q(t_r)|} \ , \tag{III.9}$$

$$s(o, T_C) = \begin{cases} 1 & if \ \ \forall t_j \in T_C, o \notin O_q(t_j) \\ 0 & in\ other\ case \end{cases} \ , \tag{III.10}$$

T_C represents the tags already included in the tag cloud output set, and $O_q(t)$ is the set of resources in O_q that contains the given tag. Therefore, $|O_q(t_i) \cup O_q(t_r)|$ computes the number of resources in O_q that contains both t_i and t_r. In addition $weight_t$ represents the weight of the tag, i.e. its importance under the domain. For example, we could employ the weight of the tags of our social tagging system (Chapter II Section 3.3), or the weight of the WSSUs of our FAQ retrieval system (Section 2.4).

As can be seen, the utility score of a tag directly depends on the previously included and selected tags. To initialize the output set T_C, we first select the tag t of $T(q)$ with higher frequency, i.e.,

$$\max_{t \in T_q}\{ \frac{|O_q(t)|}{|T_q|} \} . \tag{III.11}$$

Then, we employ a greedy heuristic to complete the selection process. For the remaining $K-1$ iterations, our algorithm selects the tag which maximizes the utility function (see (III.8)).

A simple example of this process is offered in Table III.10. In this example, only the two firsts iterations are shown —subsequent iterations behave similarly. In the example three resources that are linked to their corresponding tags. Each tag presents a weight that describes its importance under the domain. Column max_1 on the table represents the $\max\{r(t,t_r)|t_r \in T_{cloud}\}$, and column max_2 represents the $\max\{s(o,T_{cloud})|o \in O(t)\}$.

Resources and tags:
$resource_1$: {semantic web:4.75, rdf:4.48, owl:4.08, language:2.86}
$resoruce_2$: {semantic web:4.75, intelligent agent:4.23, xml:3.95, semantic search:2.15}
$resource_3$: {sparql:3.82, dbpedia:3.70, rdf:4.08, information retrieval:1.94}

Iter	**Candidate**	max$_1$	max$_2$	**f(t)**	T_C
1	-	-	-	-	{**semantic web***}
2	rdf	4	0	0	{semantic web, **sparql**}
	owl	2	0	0	
	language	2	0	0	
	intelligent agent	2	0	0	
	xml	2	0	0	
	semantic search	2	0	0	
	sparql	4.5	1	17.19	
	dbpedia	4.5	1	16.65	
	information retrieval	4.5	1	8.73	

*tag with higher frequency

Table III.10: Example of tag selection performance

Tag cloud representation Regarding the representation of the tag cloud, it is defined in basis of the principles depicted in [BGN08]. Those tags with higher utility score will present higher font sizes and font weights. The disposition in the layout is random. We employ JS Graph[11], a JavaScript Graph library for visualizing information in HTML5 web applications and environments. In Figure iii.4, the tag cloud corresponding to the tags extracted for the previous example is shown.

As it can be seen, we do not pay special attention to the visualization layouts and other related aspects of tag clouds. To access a vaster discussion on these issues, we refer the reader to works commented in Section 3.2.

[11] http://www.js-graph.com/

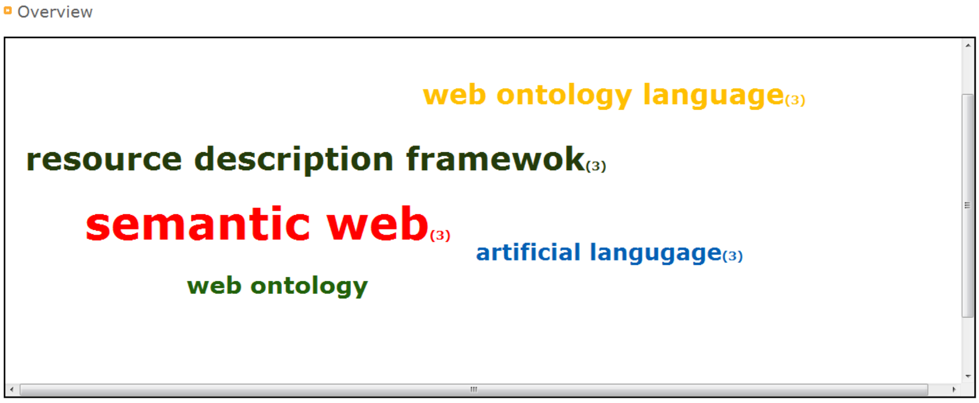


Figure iii.4: Tag cloud representation

Tag expansion To perform the tag selection task, we assume that any user query will return a quantity of resources enough for representing the tag in a tag cloud. However, a very specific user query may retrieves few results, and therefore, O_q may be too small to generate a useful tag cloud. In such cases, we apply a process for tag expansion as added value for our system.

If the number of resources retrieved by the search process is less than 4, our method uses the top-ranked resource as initial source to achieve additional information. It takes the tag linked to the top-ranked resources (T_{TOP_q}), and the rest of the tags in the knowledge base (T_{rest}) as inputs. For each $t \in T_{rest}$, the summation of the similarity score ((III.12)) between t and all elements in T_{TOP_q} is computed. The $K - |T_{TOP_q}|$ elements of T_{rest} with higher summation values are added to the tag cloud as expanded tags. After all tags are included in the tag cloud, their importance is measured accordingly to its corresponding weights. Then, the tag cloud representation principles do not vary.

$$r(t, T_{TOP_q}) = \sum_{t_r \in T_{TOP_q}} \frac{|O_q(t) \cup O_q(t_r)|^2}{|O_q(t)| \cdot |O_q(t_r)|} \ , \tag{III.12}$$

Setting the number of tags in the cloud In order to select the integer K which determines the number of tags to be inserted in the cloud, we first carried out a simple utility studio. We asked ten undergraduate students in the field of computer science for opining about four different configurations of K. Four different tag clouds (including 20, 40, 60, or 80 tags) obtained from a set of 10 queries, were shown to each student. 80% of the students agreed that the comprehensibility of the tag cloud decreases while more than 40 tags per cloud are displayed in the majority of the cases. From this point, the students chose 22 tag clouds including 20 tags, 43 tag clouds including 40 tags, 21 tag clouds including 60 tags, and 15 tag clouds including 80 tags. We then decided to set K dynamically in basis of the number of resources and tags. Therefore, K is dynamically set following (III.13)

$$K = Min \ \{ \sum_{o \in O_Q} |T(o)| \ , \ 40 \ \} \tag{III.13}$$

3.3.2 Concept Map Generation Algorithm

In this section we present our approach for generating concept maps for supporting both the learning process and the navigation through the educational content. The approach is straightforward due to we apply a direct translation of a lightweight domain ontology in order to obtain a set of maps. Therefore the difficulty of the process relies on the ontology learning stage (Chapter II Section 4.1). Our approach is composed by two visual models explained below.

The model is named as *Summary model.* Its purpose is to provide a simple overview of the concepts that present strongest relation with the educational content. It is not properly considered a concept map. The concepts of the ontology are divided in two parts of the visual interface. Those concepts with higher number of relations (high-related concepts) are represented individually, ordered in function of a importance factor. The rest of concepts are grouped into a single box at the bottom of the interface. Additionally, the interface includes the set of issues related to the course, as prototypical functionality. Students can mark one or more issues of the course using the corresponding checkboxes and subsequently the interface is modified including only the concepts related to such issues. The marked issues also affects to the display order of the concepts. The process in charge of selecting and ordering high-related concepts is explained below.

Let define the input set of concepts as $C = \{c_0, \ldots, c_n\}$ and the set of relations $R = \{r_0, \ldots, r_m\}$ where $r_i \in R = (c_s, c_p, tr); s \neq p$ refers to a relation between c_s and c_p with $tr \in r_{supc}, r_{subc}, r_{st}, r_{sb}, r_{cb}$ referring to superclass relation, subtopic relation, subclass relation, subordinate relation and content-based relation respectively. Our method considers a concept c_i as high-related if $\sum_{j=0,\ldots,m} r_j \geqslant \alpha | r_j = (c_i, c_x, tr'); i \neq x; 0 \leqslant x \leqslant n; tr' \in \{r_{subc}, r_{st}\}$. That is, only the concepts having α or more relations of the type subclass or subtopic are individually displayed. The manager of the system will be in charge of setting this parameter.

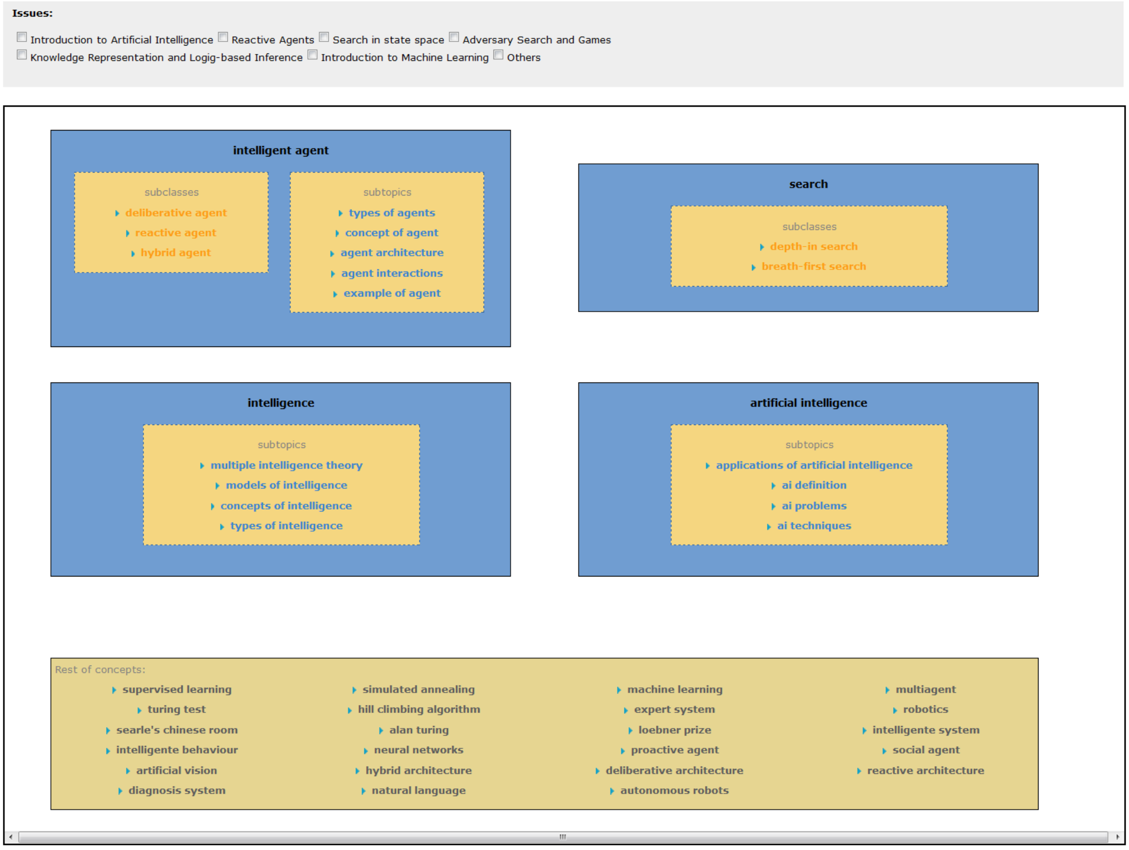
Subsequently, the high-related concepts are presented in the interface following a order based on a importance factor. For each high-related concept c_i, a weight w_i is assigned to it using the following equation:

$$w_i = \beta \cdot \frac{|lr_i^{mi}|}{|lrt^{mi}|} + \gamma \cdot \frac{|lr_i^{mi}|}{|lr_i|} \qquad \text{(III.14)}$$

, where lr_i^{mi} refers to the set of learning resources linked to c_i that also belong to the marked issues, lrt^{mi} refers to the set of learning resources linked to any concepts in the ontology that also belong to the marked issues, and lr_i refers to the set of learning resources linked to c_i in the whole course (regardless of the marked issues). β and γ are two parameters that reflects the importance of the learning resources regarding the marked issues against the importance of those regarding the whole course. The manager of the system will be in charge of setting them.

Figure iii.5 shows the summary model. As it can be seen, the high-related concepts are displayed into a box that contains the representative term and the subclasses and subtopics of the them. The concepts are not linked by arcs between each other. This model serves as a summary, since all the displayed terms are links to individual concepts maps.

The *individual concepts maps* represent the second visual model. When a user clicks into a term belonging to a concept of the summary (Figure iii.5), the visual interface is modified focusing in such concept. It could be considered a concept-level conceptual map. In it, the focused concept is displayed in the middle of the interface. The related concepts are enclosed to it, and linked by means of a labelled arc. First, the subtopics appear just below the focused concept, and a dashed box separate them from the rest of relations. Taxonomic-related concepts are shown at the top and at the bottom of the dashed box. Finally, subordinate and content-related concepts are represented in at the right of the dashed box. Figure iii.6 depicts this model. At it can be seen, the ontology is



Issues:

□ Introduction to Artificial Intelligence □ Reactive Agents □ Search in state space □ Adversary Search and Games
□ Knowledge Representation and Logig-based Inference □ Introduction to Machine Learning □ Others

Figure iii.5: Summary model interface

divided and translated into a set of concept maps, each one focused on a single concept. In addition, the upper-left part of the interface shows links to the learning resources linked to the concepts, and to the related Wikipedia article's web page (automatically obtained during the ontology learning process).

Summarizing, students could first skim trough the overview in order to understand the conceptual organization of the course, and the level of importance of the concepts. Then they could perform a deep exploration into the concepts in order to understand more complex relations. Then students cloud access to the learning resources related to each concept, gathering in this manner the individual knowledge necessary.

Finally, we have used the JS Graph[12], a JavaScript Graph library for visualizing information in HTML5 web applications and environments, in order to generate the two models.

3.4 Visualization Techniques Supporting Real-life Applications

In this section, we apply the previously commented visualization methods into our proposed FAQ retrieval, social tagging and ontology learning systems. On the one hand, we provide real-life frameworks of application. On the other hand, it allows us to test the performance and usefulness of the visualization methods.

[12] http://www.js-graph.com/

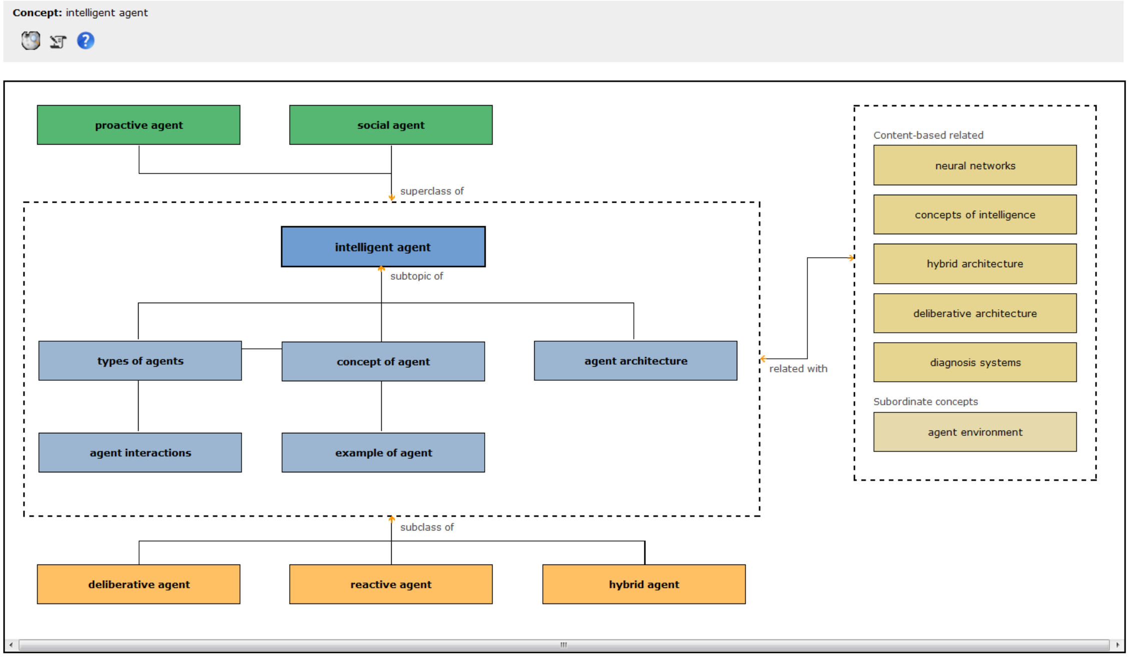


Figure iii.6: Conceptual-level concept map interface

3.4.1 Integrating Tag Clouds into FRLearn

As first framework of application, we extend the FRLearn system with our tag cloud generation algorithm. Our motivation here is two-fold. On the one hand, the trend initiated by the peak of Web 2.0 reveals that classical FAQ retrieval systems do not have into account more users' requirements than the retrieved answer[Chi03]. Current systems should provide the user with more information about the query than just a ranked list of FAQ entries. For example, by presenting an overview of all the resources available about the query to the user. On the other hand, in e-learning context this model of visualization based on the Web 2.0 can enhance the learning activity notoriously [CHK13, Cho10, JBFC10].

The main goal of FRLearn entails retrieving precise answers. These answers are considered to be relevant to the user question and are usually presented as a ranked list. Since the user is probably unaware of the knowledge of the entire domain, certain useful and related topics could lead inaccessible to the user. Moreover, since there is usually a gap between users' expectations on the domain, and the domain knowledge itself, users could be confused while formulating their questions. To solve this situation, we aim to go beyond the above discussed functionality, not only to providing users with answers to their questions but also to expanding the user's knowledge about the initial question (as many times as the user desires). The functioning scheme of our system is depicted in Figure iii.7.

Remember that in an initial learning stage, FRLearn extracts information units (WSSUs) from each element of the FAQ list. Hence, each FAQ entry of the collection is associated with a set of information units. These units contain the semantic information of each element. Then when the user inputs a query, the system retrieves the most relevant FAQ entries in the collection, and presents them in the output interface. In addition, the system runs our Tag Cloud Generation algorithm considering the information units associated to each FAQ entry as tags, and the retrieved FAQ entries as resources.

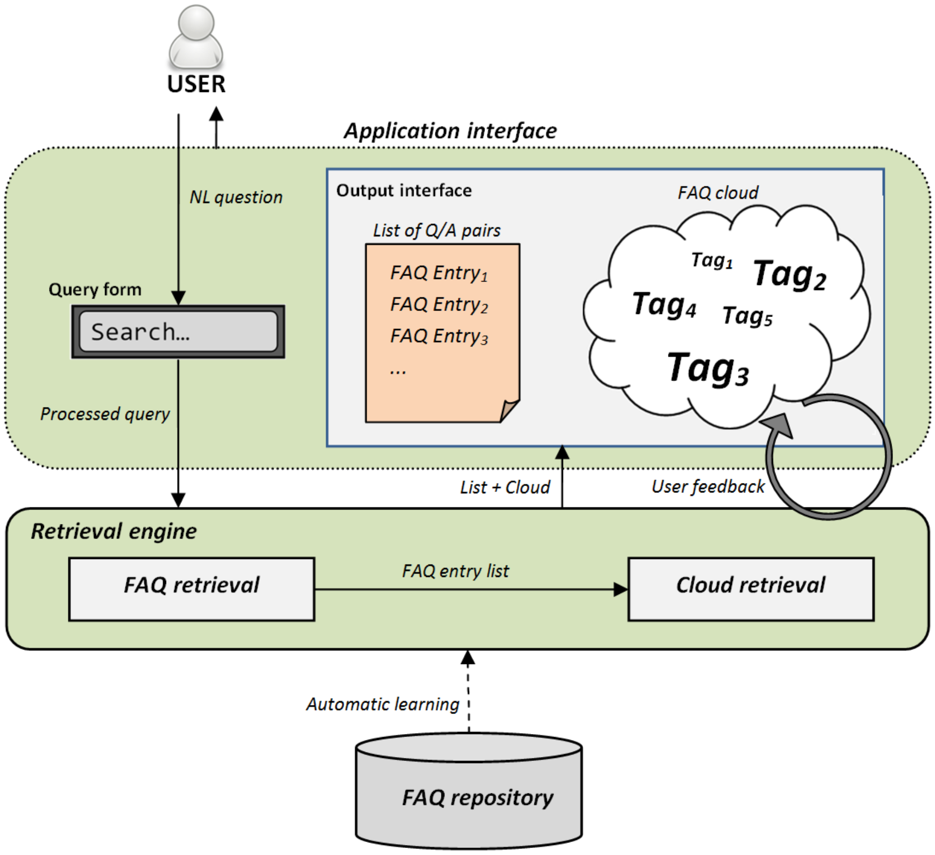


Figure iii.7: FAQ cloud usability

Finally, the tags in the tag cloud representation serve as links to another user queries. That is when the user clicks on a tag, the output interface shows a new ranked list of FAQ entries (now related to the clicked tag), and the cloud is refreshed in the same manner.

Architecture of the extended FRLearn system The modular architecture of FRLearn is extended including now a Cloud Retrieval module, in charge of running the Tag Cloud Generation algorithm (Figure iii.8).

Now the main goal of the retrieval module is to obtain a ranked list of Q/A pairs and a cloud of tags related to that list (Figure iii.9).

Firstly, the FAQ retrieval algorithm receives the expanded user query, and the WSSUs extracted from the FAQ list. The algorithm tries to match each query word (and their synonyms) with WSSUs, and then it calculates a score for each FAQ entry in function of their corresponding matched WSSUs' weights. Finally, a ranked list of FAQ entries (disposed as Q/A pairs) is obtained. The top-ranked elements of the list are suppose to be the best related to the user query.

Secondly, the Cloud retrieval algorithm, i.e. that Tag Cloud Generation algorithm, receives the list of ranked FAQ entries and their corresponding WSSUs (the set of resources O_q, and a set of tags $T(q)$). The WSSU of the top-ranked element compose the initial set of tags. Then, the degree of semantic relatedness between each initial tags and the rest of WSSU is measured. For a given size K, those WSSU with higher semantic relatedness are included as tags, expanding the initial

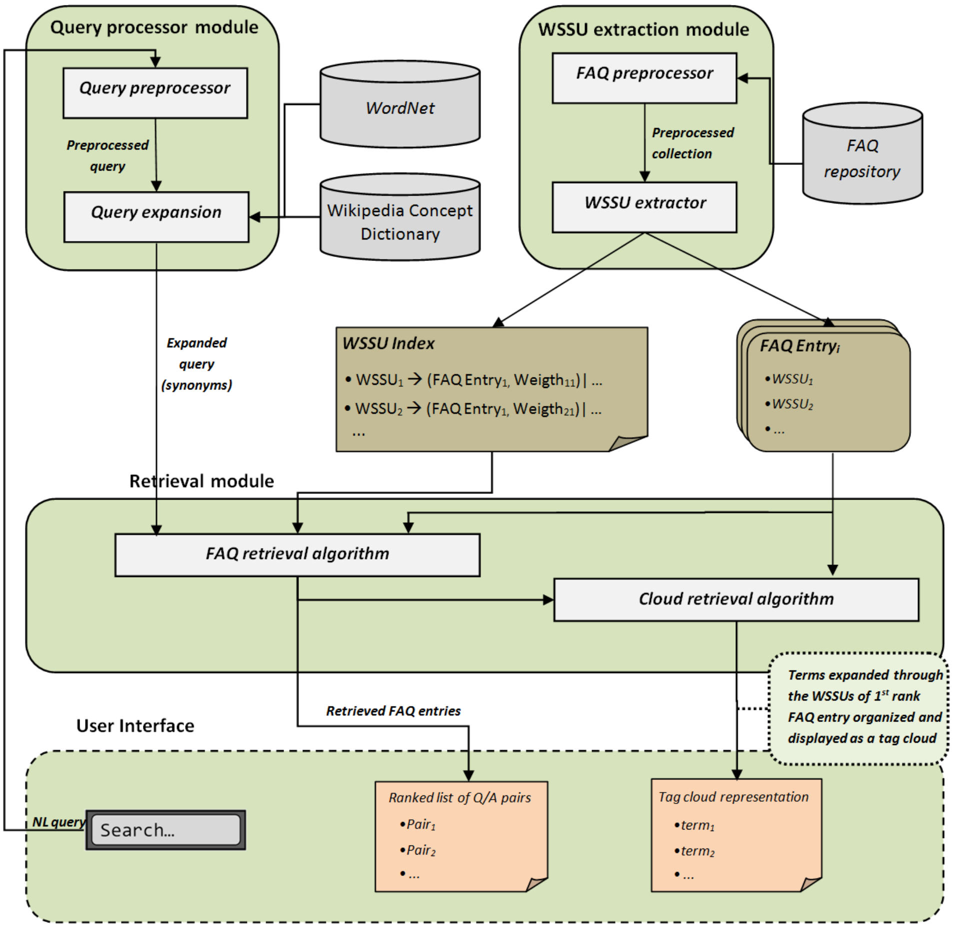


Figure iii.8: Extended architecture of FRLearn

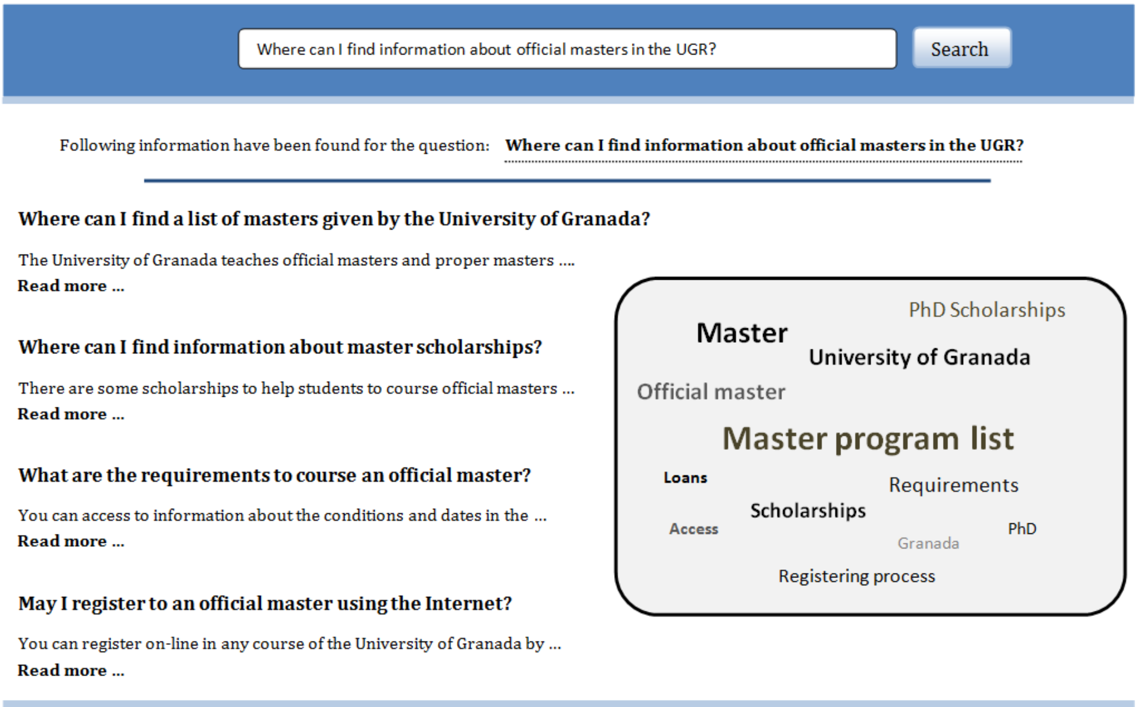
set. Once the initial set is expanded, the tags are organized in a tag cloud.

Finally, each tag of the FAQ cloud represents a links to another search query. When the users clicks in one tag, the system obtains the list of FAQ entries related to the corresponding WSSU, sorted by weight. Once the list is obtained, the Cloud retrieval algorithm is executed again. We call FAQ cloud to this tag cloud representation and its linking structure for searching.

3.4.2 Integrating Tag Clouds into TRLearn

As second framework of application, we extend the TRLearn system with our tag cloud generation algorithm. This application is expected since tag clouds became popular due to its integration in social tagging systems. The goal of TRLearn is to collaboratively obtain a multi-domain conceptually extended folksonomy which represents the knowledge about the domain. Therefore, we extend the system for providing a tag cloud representation, with the aim to facilitate users to explore the conceptually extended folksonomy and to quickly identify which are the most important tags in the domain.

Remember that the information extraction process of TRLearn performs in the following way. First, the most relevant candidate tags are found from the textual fields of the available set of



Where can I find information about official masters in the UGR?

Following information have been found for the question: **Where can I find information about official masters in the UGR?**

Where can I find a list of masters given by the University of Granada?

The University of Granada teaches official masters and proper masters ....
Read more ...

Where can I find information about master scholarships?

There are some scholarships to help students to course official masters ...
Read more ...

What are the requirements to course an official master?

You can access to information about the conditions and dates in the ...
Read more ...

May I register to an official master using the Internet?

You can register on-line in any course of the University of Granada by ...
Read more ...

Figure iii.9: Output interface for the user query: 'Where can I find information about official masters in the UGR?'

resources, and a weight is assigned to each one in function of statistical and semantic features of the candidates. During such process, each candidate is extended with a set of synonyms obtained through our Wikipedia-based dictionary of concepts (Chapter II Section 2.3. Subsequently, the candidate tags are presented to the user in order to validate them. When a tag is picked as valid, both the tag and the resource linked to it are included in the conceptually extended folksonomy.

Therefore we use the tags of the conceptually extended folksonomy in order to obtain a tag cloud representation. It summarizes the content of the documents in the system. Therefore, users are able to explore the main concepts of the domain and to access the documents linked to each tag.

Architecture of the extended TRLearn system The modular architecture of TRLearn is extended including now a Overview Generation module, in charge of running the Tag Cloud Generation algorithm (Figure iii.10).

The Tag Cloud Generation algorithm here considers an hypothetical query which always returns the whole content of the dataset. Therefore the set of resources O_q in the algorithm includes all documents contained in TRLearn, and the set of tags $T(q)$ in the algorithm includes all tags contained in the conceptually extended folksonomy. Once the tag cloud is represented, the user can access the resources referenced by each tag by clinking it.

In addition, the different synonyms of a each conceptually extended tag allow us to add a new functionality to the tag cloud scheme. In addition to the representative term of a tag, our extended system includes the synonyms of such tag in the visual representation. Next to each tag, we can observe a number in brackets referred to the number of synonyms of such tag. If a user clicks in such number, the interface changes to display the different terms of the tag. The terms also present a font size variation, attending to its independent document frequency. As more times a term is

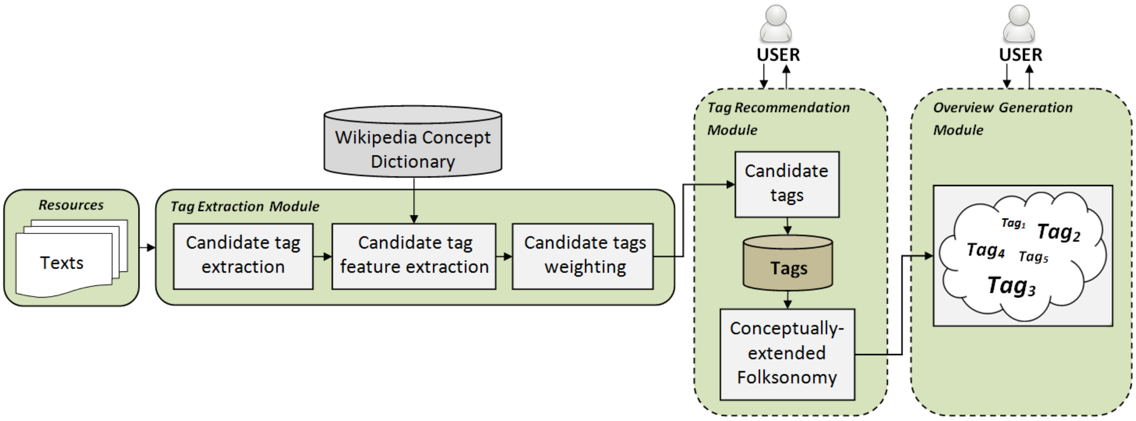
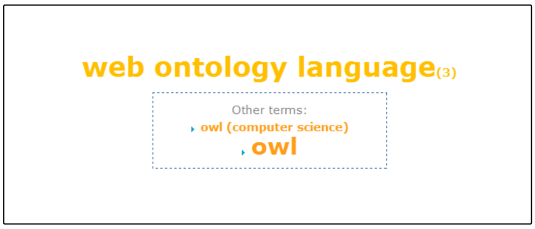


Figure iii.10: Extended architecture of TRLearn

referenced in the resource set, as higher font size it presents. The interface showing the different terms of the conceptual tag “web ontology language” is shown in Figure iii.11. This fact allows users to gain knowledge about the domain under consideration. The user not only will be able to know the different terms regarding to a same concept, but also will see the usual ways to call it in the domain.


web ontology language(3)

Other terms:

- owl (computer science)
- owl

Figure iii.11: Interface for the individual tag “web ontology language”

3.4.3 Integrating Concept Maps into ORLearn

As last framework of application we consider the integration of the Concept Map Generation algorithm into our ontology learning ORLearn system. The initial system was designed for obtaining a lightweight domain ontology from scratch. Using the course’s learning resources as input in the initial phase, the system assists the manager (the teacher) during the ontology learning process. Then the resulting ontology carries two characteristics: it summarizes the knowledge of the course from a educational perspective, and it serves as index of the involved learning resources. Considering now the extended capacity of the system, the automatic generation of concept maps from the inner representation of the ontology complete the framework. Now the students take the leading role of the system. The visual representation of the information/knowledge of the course will guide them through the learning objectives: the representation initially offers the main concepts of the course ordered by its importance with regard to the educational content and the different issues of the course; and afterwards it is focused on a concept-level map representation. Students are able to understand the whole course structure and then acquire a deeper knowledge on individual concepts. Moreover, the representation facilitates the access to the learning resources linked to each concept, and to an additional resource (Wikipedia).

The integration of the Concept Map Generation algorithm on the ORLearn architecture is shown in Figure iii.12. The Map module is now in charge of running the Concept Map Generation Algorithm, receiving the concepts and relations previously validated by the teacher. The summary model highlights the high-related concepts in the interface, and presents the rest of the concepts (non high-related) independently. The name of the concepts serves as link to the individual concept map model. It is centred at concept-level. No complex algorithm is needed here, since the concept map generation process is designed as a straightforward translation of the lightweight domain ontology.

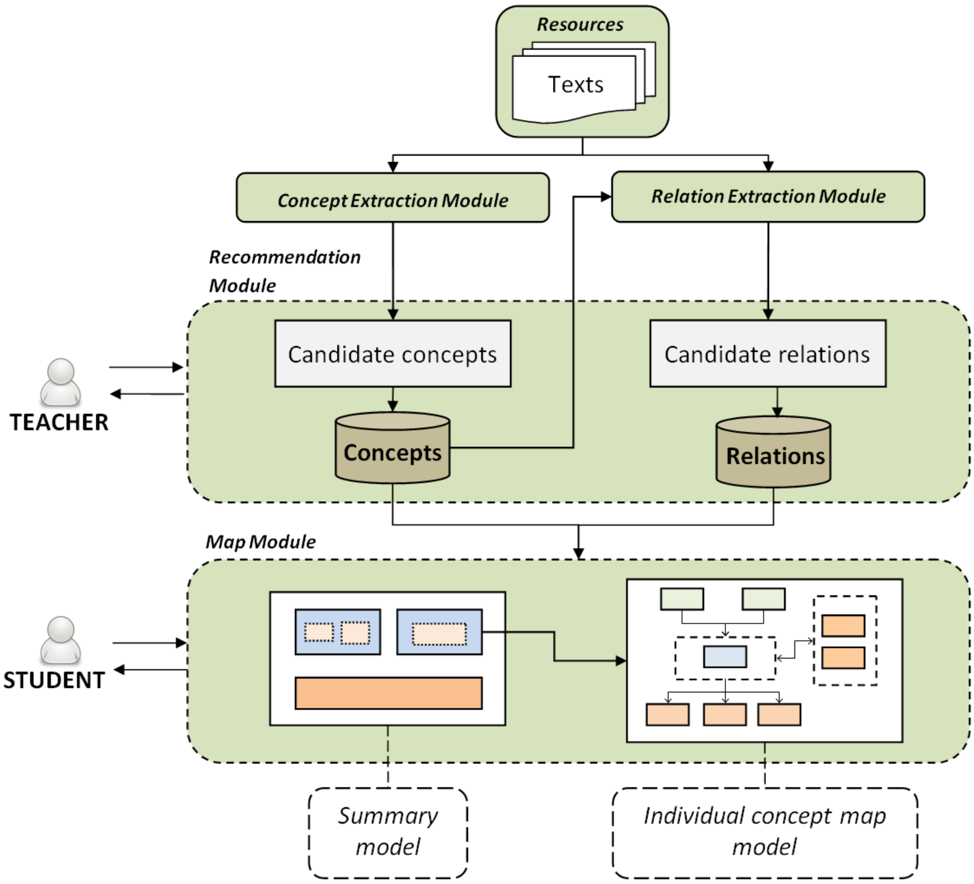


Figure iii.12: Extended architecture of ORLearn

3.5 Experimental Evaluations

This section reports the empirical results obtained in the evaluation of our visualization methods.

First, regarding the tag cloud visualization technique, we test the performance of our tag selection algorithm compared to other state-of-the-art methods. The metrics we used for this purpose will be described below. Due to tag selection is commonly used to summarize the results obtained through user queries, we use the framework provided by our proposed extension FRLearn.

Second, we test the validity of our Concept Map Generation algorithm in terms of usability. As we seen, the algorithm consists of a straightforward translation of the concepts and relations presents in an ontology into a visual representation. Consequently, the strength of the method does not rely in computational aspects. Thus we focus on the opinion of users about the usability of the method as extension of our ORLearn system.

3.5.1 Tag Cloud Generation algorithm: Testing the Performance of the Method

In order to measure the usefulness of our Tag Cloud Generation algorithm, we should attend to how the resulting tag cloud summarize and expand the expected information. Since this task is not trivial, a set of metrics were defined in the literature to evaluate the results of tag selection algorithms [VKGM11a, SA11]. These metrics consider the tag selection task as a method for summarizing the results obtained with a user query. Therefore we consider the proposed extension of FRLearn. We have adopted of the most employed metrics to prove the validity of our tag selection algorithm: coverage, overlap and selectivity.

The *coverage* of a tag cloud TC (set of tags) with respect to the resources returned to the user query O_q represent the portion of the entries in O_q that have at least one tag appearing in the cloud (see (III.15)). The higher the coverage, the more effective the tag cloud.

The *overlap* between to tags t_i and t_j is computed as the portion of resources in O_q tagged with t_j that are also tagged with t_i. Thus, the overlap of a tag cloud TC (see (III.16)) is defined as the average overlap between each pair of tags. The lower the overlap, the more effective the tag cloud is.

Finally, the *selectivity* of a resource o with respect to a tag cloud measures the number of tags belonging to the rest of resoruces which are not included in o. Then, the selectivity of a tag cloud T_C is computed as the average selectivity of the resources in O_q (see (III.17)). The higher the selectivity, the more effective the tag cloud.

$$Coverage(T_C) = \frac{|\{\ o \in O_q\ : T(o) \cap T_C \neq \emptyset\}|}{|O_q|} \tag{III.15}$$

$$Overlap(T_C) = \frac{\sum_{t_i,t_j \in T_C} \frac{|O(t_i) \cap O(t_j)|}{|O(t_j)|}}{|T_C| \cdot (|T_C| - 1)} \tag{III.16}$$

$$Selectivity(T_C) = \frac{\sum_{o \in O_q} \frac{|\{o_i \in O_q\ :\ (T(o) \cap T_C) \not\subseteq T(o_i)\}|}{|O_q|}}{|O_q|} \tag{III.17}$$

We compare our tag selection results with the following methods:

- **Frequency-based algorithm** [SA11]: This algorithm ranks tags based on their frequency, i.e. the number of resources to which a tag is assigned. Therefore, the tag cloud is formed by selecting the top-K most frequent tags.
- **TF*IDF-based algorithm** [VKGM11a, SA11]: This algorithm is based on one of the most-known measures in document retrieval. The utility function is based on the assumption that a tag has lower utility in describing the contents of a group if it also occurs frequently in several other groups. The top-K tags with higher utility score are selected to represent the tag cloud.
- **Maximum coverage algorithm (MCA)** [VKGM11a]: It employs a greedy heuristic to maximize the coverage of the resulting set of tags. For K iterations, this algorithm selects the tag t from T_q that covers the largest number of uncovered entries by that point.
- **Diversity algorithm** [SA11]: The main goal of this algorithm is to select tags that are as dissimilar as possible from each other, in the sense that appear in different set of entries. To that end, the authors design a function to find tags that are not similar to the previously selected tags. The top-K scored tags define the tag cloud.

- **Novelty algorithm** [SA11]: This approach tries to diversify the members of a tag cloud by emphasizing on the novelty of newly selected tags, while the cloud is constructed. It is based on the notion of *information nuggets*. In more detail, this algorithm selects the tags which belong to more entries that are not included yet in the cloud.

For the comparison algorithms, we set the parameters according to details given by the authors in the corresponding works.

Regarding the dataset, we employ the same three collection of FAQ documents employed during the experiments of FRLearn (see Section 2.5): Restaurant FAQ list, Linux V.2.0.2 FAQ list, and UGR FAQ list.

In order to test the performance of each comparison algorithm we have taken the FAQ entries returned during the 10-fold cross-validation of FRLearn from each FAQ list. Remember that in each validation, the data set in consideration was divided into ten mutually exclusive subsets of the same size. Each fold was used to test the performance of the FAQ retrieval method, using the combined data of the remaining nine folds to extract the WSSUs. The whole process was repeated 10 times in order in order to diminish the dependence of the results with respect to a concrete partition of the reformulation set. As result of each validation, a set of FAQ entries was obtained. Therefore we use the resulting FAQ entry set (O_q) and the corresponding WSSUs ($T(q)$) as common input for all the comparison algorithms. Consequently, 10 iterations of the process has been taken into account. Tables III.11 to III.13 display the performance of each algorithm on the three datasets.

Measure	1	2	3	4	5	6
Coverage	FC	Novelty	MCA	tf·idf	Frequency	Diversity
	1	1	1	1	1	1
Overlap	FC	Diversity	Novelty	tf·idf	MCA	Frequency
	0.3067	0.3397	0.3401	0.3449	0.3454	0.3507
Selectivity	FC	MCA	Diversity	Frequency	tf·idf	Novelty
	0.7816	0.7517	0.7516	0.7456	0.7451	0.7442

Table III.11: Ordered rank of each method on Restaurant FAQ list

Measure	1	2	3	4	5	6
Coverage	FC	Novelty	MCA	tf·idf	Frequency	Diversity
	1	0.9500	0.9460	0.9388	0.9375	0.9364
Overlap	FC	Novelty	tf·idf	MCA	Diversity	Frequency
	0.2551	0.2883	0.3189	0.3676	0.3680	0.3940
Selectivity	FC	Novelty	tf·idf	Diversity	MCA	Frequency
	0.8023	0.7723	0.7200	0.6720	0.6666	0.6665

Table III.12: Ordered rank of each method on Linux FAQ list

In light of the results, our tag selection method outperforms the rest in terms of *overlap* and *selectivity* on Linux and Restaurant datasets (all the algorithms obtained the maximum score in terms of *coverage*). With respect to the UGR datasets, our method obtained the highest values on *overlap* and *selectivity*, and the second best value on *coverage*.

Measure	1	2	3	4	5	6
Coverage	Diversity 0.9935	FC 0.9932	Frequency 0.9926	MCA 0.9908	tf·idf 0.9840	Novelty 0.9700
Overlap	FC 0.5252	Novelty 0.5498	tf·idf 0.5622	Frequency 0.5869	MCA 0.5955	Diversity 0.7332
Selectivity	FC 0.8331	Novelty 0.8148	tf·idf 0.8125	Frequency 0.7840	MCA 0.7796	Diversity 0.6500

Table III.13: Ordered rank of each method on UGR FAQ list

The rest of the algorithms perform in a similar way, presenting lower variability among the results. The Novelty algorithm shows good performance in general, obtaining the best results when the number of tags K is high in comparison with the average number of tags. Next, Diversity algorithm obtains low values for *selectivity* in contrast to *coverage*. This may be because to their resulting clouds do not present a smooth distribution of tags for each resource. The MCA algorithm is specifically designed to achieve suitable *coverage* results, as it is the case. However, it exhibits low *overlap* results. Later, tf·idf displays low *selectivity* in comparison to its *overlap* and *coverage*. Finally, the frequency baseline algorithm offers intermediate results in all the cases.

As we did in the FAQ retrieval experiments, we have performed a t-test with a confidence level of 95% for each pair of algorithms, obtaining a p-value < 0.0001 in all cases. Thus, results here exposed are considered to be statistically significant.

3.5.2 Concept Map Generation Algorithm: Testing the Usability of the Method

The validity of our proposed Concept Map model (including both the summary model and the individual concept map model) do not depend on computational aspects. Therefore we have conducted a usability evaluation of the extended functionality of ORLearn. To that end, we took a group 25 volunteers students from the “Artificial Intelligence” course (2012/2013 year). Their objective was to analyse our tool regarding usability aspects. The system contained the lightweight domain ontology of the course generated during the experiments of the initial version of ORLearn (Chapter II Section 4.4). In addition, the extended ORLearn system was stored in a university server under Apache Tomcat 6 (6.0.32) to manage the data storage and was accessible via web application. During the progress of the experiment, a technical assistant was available to assure the correct operation of the system and to manage the technical issues reported by the users. All participants were instructed about the use of the map model of ORLearn. Then they counted with a one-hour session for exploring the content of the course represented by our approach. After that, we applied a usability questionnaire with the aim of measuring their level of satisfaction in regard to the extended functionality of the TRLearn system. The questionnaire was composed by 5 items based on a five-point Likert scale. The scale ranges from strongly agree (5 points) to strongly disagree (1 point).

The description of the items and the mean and standard deviation rates assigned by the students is shown in Table III.14. Students agreed in the good performance of the model. The mean rates for the items were significantly high. This demonstrates the effectiveness of the concept map model. As combination of information and knowledge visualization, our model seems to provide both an effective overview of the domain and a suited organization of the learning resources. Nevertheless this study presents two main limitations. On the one side, the main difficulty of this scheme relies on the generation of the ontology. Once the ontology is created, the complexity of the visual model is scarce. On the other side, volunteer students may tend to have a better opinion of the system than non-volunteer students. Nonetheless, the validity of the concept map model is broadly demonstrated in several studies of the literature. In this sense, we have found similar results along this experiment, confirming the good suitability of concept maps for improving the learning process.

Item	Students (N = 25)	
	M	*SD*
1. The visual interface is intuitive and easy-to-use	4.48	0.58
2. The summary model correctly summarizes the course content	3.72	0.89
3. The individual concept map model correctly represents concept-level knowledge and relations	4.36	0.75
4. The interaction between both models is clear and understandable	4.24	0.89
5. Both models would help me for searching information about the domain and for navigating though the course content	4.24	0.83

Table III.14: Results of the 5 items of the usability questionnaire

3.6 Conclusions and Future Work

In this part of the Thesis, we have designed two main visualization methods with the aim to cope the problems derived on the information overload. The considered models combine the main purposes of information and knowledge visualization fields: they represent the underlying knowledge of the domain, and they serve as link of the knowledge representation with the educational content. We have focus on two well-know visualization techniques: tag clouds and concept maps. The models work with folksonomies and ontologies, completing in this way the knowledge acquisition mechanisms presented in Chapter II. Firstly, the tag cloud model serves as overview and navigation scheme of the educational content. The tags represent the main concepts of the domain. Each tag is displayed with a size and weight corresponding to its importance under the domain. Additionally, the tags are links to the learning resources. The model is mainly designed to work with user queries, extending the results obtained by search algorithms, but it also works without user queries. The complexity of the process relies on the selection of the tags from the tag base that better fit with the retrieved results. Secondly, the concept map model is obtained through a straightforward translation of lightweight domain ontologies. It is composed by two different sub-models. The summary model offers an overview of the highly related concepts in the domain. The individual concept map model focuses at concept-level, and shows a detailed visualization of each concept of the domain and its relations. In addition, the learning resources related to each concept are linked in the interface.

As part of this study, we have defined three frameworks of application of the visualization methods. We have extended our previous FRLearn, TRLearn and ORLearn systems. By means of the extended versions, we have proved the effectiveness of the visualization methods. We tested the performance of our tag selection algorithm against a set of state-of-the-art comparison algorithms. The results obtained by our proposal outperformed the rest of algorithms in terms of specific metrics from the literature: coverage, overlap, and selectivity. Considering now the concept map generation algorithm, we tested the usefulness of the proposal in terms of usability, since the scheme do not present computational complexity. The model was employed by a set of real students on a high education course. They concluded that out concept map model was intuitive and easy-to-use, and it was pertinent for summarizing the knowledge of the course and for organizing the learning resources in a navigable form.

Regarding future work, in this research we choose the layouts aspects of the tag cloud representations according to the main directives followed in state-of-the-art researches. But it is still up to future research to show whether other visualization properties could be more suitable for our purpose. Finally, we plan to develop a user navigation model along the lines of [SA11] to quantify different properties of a tag cloud with respect to search and navigation.

4 Final Discussions and Future Work

The problem of information overload affects e-learning users provoking disorientation problem and cognitive overhead. Most of resolution methods are related with the implementation of adequate aids for navigating through the content space and for helping users to find useful information along the content. In this part of the Thesis we have focused on these matters in order to provide students with tailored methods for improving their searches and navigation through the educational content.

First, we have designed a FAQ Retrieval system able to retrieve precise results from NL user queries. The method processes the FAQ collection obtaining internal representation easily interpretable. The method has been contrasted against state-of-the-art algorithms obtaining promising results. We have selected this specific field of application since FAQ lists are receiving great attention as learning resources. Nonetheless, designed the method is easily extendible to other types of resources. Thus we plan to provide additional validation taking into consideration other types of learning resources.

Second, two methods combining Information and Knowledge Visualization techniques have been proposed. Although Information and Knowledge Visualization are usually treated as two distinct fields, they share several characteristics. For this reason, we have focused our efforts in developing combined strategies with a two-fold goal. On the one hand, we aim to provide users with tools able to represent the knowledge of the domain in a visual form. This facilitates the understanding of the knowledge by promoting human cognitive processes which support learning. On the other hand, the visual representations serve as navigational structures of the educational content. We have selected to well-known visualization techniques: Tag Clouds and Concept Maps.

Tag Clouds are visual displays of tags which are annotations over learning resources. They support a sallow exploration of the whole domain and a organization of the educational content. We have developed a Tag Cloud generation method able to automatically construct this kind of visualization. The method perfectly fits with folksonomies, since is the most popular form of visualization in social tagging systems. In addition, the main task involved in the generation of Tag Clouds relies on the selection of a subset of tags that better fits with NL user queries. Thence we have provided two frameworks of application, using our Tag Generation algorithm as extension of our previous Tag Recommender and FAQ Retrieval systems. We have validated the Tag Cloud generation algorithm against state-of-the-art comparison algorithms. Our method outperforms the rest in terms of coverage, overlap and selectivity.

Concept Maps are visual representations of different concepts and their relationship. This kind of visualization technique organizes the student's cognitive structure to foster a deep level of integrated knowledge. The benefits of Concept Maps regarding the understanding of the domain has been widely documented. In addition, we view this scheme as a excellent tool for organizing and facilitating the access of the educational content. We have taken advantage of our previous Ontology Learning method in order to develop a Concept Map generation algorithm. The concept map model is obtained through a straightforward translation of a lightweight domain ontology. The model has been contrasted by a usability study, carried out by real engineering students. They agreed that the model was useful for understanding the knowledge of the domain and for organizing the educational content.

Finalizing, we do not pay attention to layout aspects of the Tag Cloud and Concept Map representations. Although layout aspect may provoke additional benefits on both models, we have centred here in computational design. Thus we plan to perform additional research about this issue.

As additional method to overcome information overload states, adaptive mechanisms are receiving great attention. Thus the next chapter of this Thesis will be dedicated to explore adaptive methods able to adapt the educational content in function of individual student's characteristics (chapter IV).

Chapter IV

Intelligent Adaptive Methods for Educational Systems

1 Introduction

Nowadays the ICT (particularly the Internet) has provided an ideal setting to make the e-learning a very attractive field from almost every social and economic facet. As a result, the number of studies and systems related to e-learning has grown exponentially in the last 30 years. Nonetheless, most of traditional VLE platforms could be improved taking into consideration several characteristics for improving the learning experiences. Apart from the drawbacks inherited from general hypermedia systems (information overload, disorientation…), educational system have specific limitations associated with the learning process. Compare with real-life classroom teaching, computer-based educational systems presents lacks of contextual and adaptive support, lack of flexible support of the presentation and feedback, and lack of the collaborative support between students and systems [XWS02]. For example, one limitation of traditional VLE applications is that they provide the same static explanation and suggest the same content to students with widely differing goals and knowledge of the subject. Additionally, it may occur that students fail to effectively grasp important information due to having total freedom in browsing the web course. These causes have triggered a growing interest on adaptive e-learning platforms, which is suitable for teaching heterogeneous student populations [SGA08]. Today there exist two well-known fields centred on the design of adaptive e-learning: Adaptive Educational System (AESs) and Intelligent Tutoring Systems (ITSs). They focus on the capacity of automatic adjustment of the system's environment in function of the current state of each user. Such fields are closely related (many actual systems share features of both categories) although they present different bearings. Brusilovsky presented an in-depth study showing the details of both fields [BP03]. We follow that work in order to offer a comprehensive overview to the reader.

One the one hand, AES stress the capacity of offering different content for different students by taking into account information accumulated in the individual student models. In order to achieve this, two major technologies had been explored. First, adaptive presentation technology is concerned with the adaptation of the content presented in each web page to student goals, knowledge, and other information stored in the student model. Therefore the content is not static and it is dynamically generated or assembled for each user. Second, adaptive navigation support technology has the aim to assist the student in hyperspace orientation and navigation by changing the appearance of visible links. For example, the system can adaptively sort, annotate, or partly hide the links of the current page to make it easier to choose where to go next.

On the other hand, ITSs stress the application of CI techniques to provide broader and better support for the users of educational systems. The term ITS was introduced by Sleeman and Brown in their work [SB82]. Major technologies to that end are: curriculum sequencing, intelligent solution analysis and problem solving support. The goal of curriculum sequencing consists of providing the student with the most suitable planned sequence of topics to learn and learning tasks (examples, questions, etc.) to work with. It helps the student find an optimal sequence through the learning resources. Intelligent solution analysis checks the solutions given by the user to educational problems. This technique allows providing extensive error feedback and updating the student model. Finally, the goal of problem solving support technologies is to help the student on each step of problem solving scenarios.

Many systems presented by the scientific community can be classified as both intelligent and adaptive. Even so, there are a number of systems that falls in exactly one of these categories (see Chapter I Section 2). The main goal we set for this part of the Thesis includes how to properly design intelligent techniques to support the learning processes in computer-based educational systems. Thus we will deal uniquely with the ITSs field but bearing in mind the direct connection concerned with both fields.

The leading idea behind ITSs relies on the intelligent presentation of individualized learning schemes for each student. This practice could be easily understood as an interpretation of the pedagogical activity of *human tutoring*. That is, the learning process is performed considering individuality of each student like in traditional one-to-one instructional process. Precisely it has long been recognized that human tutoring is much more effective than group-based learning [DdB12, SRŽG08]. Cohen et al. shown through a meta-analysis of findings from 65 independent evaluations of school tutoring programs that these programs have positive effects on the academic performance and attitudes of those who receive tutoring. In this sense, Bloom [Blo84] quantified that the average student who received one-on-one tutoring from an expert tutor scored two standard deviations higher on standardized achievement tests than an average student who received traditional group-based instruction.

Summarizing, ITSs are a asynchronous sub-type of VLE applications that try to achieve a similar degree of individualization using targeted and appropriate adaptation to students [DdB12]. This involves presenting learning material in a style and order to suit the student (e.g. by presenting learning material matched to poorly understood topics), and also proactively helping students, e.g. by giving intelligent feedback on incomplete or erroneous solutions and guidance to assist students in constructing solutions to problems [SRŽG08]. ITSs show promise in approaching the effectiveness of human tutoring and systems that are accessible, inexpensive, scalable and above all effective would provide one critical component of an overall educational solution [NWM08].

The development of ITSs is therefore related to a number of serious problems, because proper implementation of "human" tutor can be done only in relation to cognitive psychology, artificial intelligence and education. Knowledge is a key to intelligent behaviour and, therefore, ITSs are said to be knowledge-based because they have: domain knowledge; knowledge about teaching principles and about methods for student modelling [SRŽG08]. Nkambou et al. [NBM10] summarized the major components of the classical design architecture of an ITS: the domain model, the student model, the tutoring model and the interface (Figure iv.1).

- The *domain model* (also known as **expert knowledge**, **teacher model** or *domain knowledge*) contains the knowledge representation of the application domain, e.g. the concepts, rules and

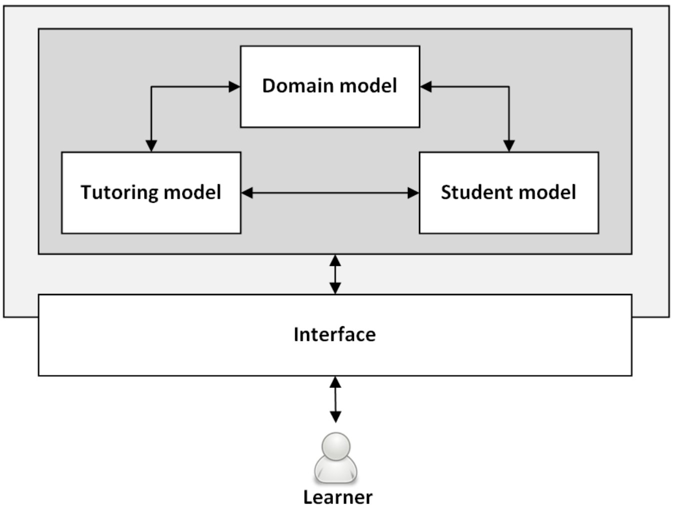


Figure iv.1: The four-component ITS architecture

problem-solving strategies of the domain to be learned. It can fulfil roles such as a source of expert knowledge, a method for evaluating the student's performance, or for detecting errors, etc [NBM10]. It is usually organized into a curriculum sequence which includes all the knowledge elements following a pedagogical flow.

- The *student model* (corresponding to the user model in AESs) is the core component of any ITS for personalization. It is used to reflect the student's cognitive state of the domain model. Based on the student's interactions with the system, it should continuously evaluate the student's progress of the knowledge acquisition. Self [Sel90] defined in 1990 the student modelling as a process devoted to represent several cognitive issues such as analysing the student's performance, isolating the underlying misconceptions, representing the student's goals and plans, identifying prior and acquired knowledge, maintaining episodic memory, and describing personality characteristics. These functions have been both expanded and diversified in the years since then.

- The *tutoring model* (also called *teacher modelling*) receives information from both the domain and student models in order to select the better learning strategies and actions. It could be considered as an inference engine that adapts the elements of the learning domain to be transmitted to the student.

- The *interface component* (also noted as *communication component*) give access to the elements of the learning domain through multiple forms of interaction. This auxiliary module should make the student feel comfortable and confident with the system.

Although this architecture has been employed in early tutoring systems [KTGS93, THP$^+$01], is common to find new proposals that expand or adapt the former model. For example, Yang [Yan10] proposes a variation of it considering now five components: student model, instruction component, domain knowledge, adaptive curriculum planning and user interface. The election (or definition) of an ITS architecture mainly relies on modularity and encapsulation perspectives, since all ITSs shares the common goal.

The construction of an effective student model is the crucial factor for designing ITSs. Indeed personalization of the educational environment heavily depends on the quality of the information that represents system's believes about a student's level of knowledge of a subject to be learned [Bru96]. The information is usually gathered through an diagnosis process, which is about inferring what the student knows from a set of observable facts [JJG12]. The diagnosis process is a key activity on any system which aims to build a dynamic student model [Ohl86]. In ITSs, students' diagnosis requires uncertainty reasoning, since there is no direct interaction between the teacher and the students [GKPM02]. Current approaches handling uncertainty are usually domain dependant: C programming language [CV12], comprehension of history [TGCK03], module of the specific InterMediActor platform [Kav04], mathematics and physics [SMGS05] among others. Thus they are hardly translatable to other domains. Another strategy for inferring the student's knowledge level consists of observing some features related to navigation: number of times that a resource has been consulted, number of clicks on determined links, time spent in each web page, etc. [KDB03, SMGS05]. Nevertheless, this kind of extracted information is likely uncertain in a web environment (the student could be checking his/her Facebook profile at the same time...). In addition, independently of the student modelling approach, the diagnosis is usually performed through a set of quizzes or problems generated by a tutor and solved by students [JJG12]. Although the system automatically chooses the assessment resource, it is rather common that such quizzes or problems are manually designed by a human tutor. This circumstance makes the matter delicate because the tutor should consider all possible students' cognitive states in order to generate fitted assessment resources.

These causes have lead us to investigate a method for automatically assessing the student level of knowledge about a subject handling uncertainty, by means of dynamically generated assessment resources. We will firstly study an intelligent e-assessment mechanism for generating test resources in function of user objectives. This will serve us as engine for designing a domain independent ITS proposal centred on student's assessment.

The rest of this chapter is structured as follows. Section 2 offers an study about the automatic generation of assessment resources. Next we present a proposal of a assessment-based ITS in section 3. Finally, we will conclude with some final discussions in section 4.

2 Improving E-assessment: a Fuzzy Test Generation Framework

2.1 Introduction

Assessment is a crucial process in charge of measuring the cognitive abilities and learning outcomes with the aim of improving student's learning. Through assessment strategies, both teachers and students are able to identify the weakness in the learning process and therefore set methods for improving it. That is, assessment allows for a continuous cycle to improve the student's performance. Considering e-learning field, e-assessment is referred to the use of technology to support the process of testing the student learning status. Current e-assessment systems offer a significant improvement of the process for all its stakeholders, including teachers, students and administrators [McD02]. E-assessment has many advantages over the traditional model: lower long-term costs, instant feedback to students, greater flexibility with respect to location and timing, improved reliability, and enhanced question styles [How10]. As it can be seen, the model is changing from paper and pen to computer-based assessment. Usually, computer-based assessment is carried out using tests [MYRP07]. A *test* is composed by a set of test items (for example, composed by quiz question).

Cheng et al. [CLH09] summarized some features of computer-based tests (CBT). They allow learners to receive the evaluation results immediately, and can save a lot of time for instructors during assessment process. Furthermore, the test record can be saved automatically during examination, and these records can be analysed further. Based on this analysis, the teachers can offer more relevant suggestions to facilitate improved learning. According to Meng et al. [MYRP07], the key issue of computer-based testing systems is related with the efficiency and quality of a test. Besides the quality of each test item available, it also depends on an appropriate algorithm design in charge of selecting the most proper test items. In intelligent tutoring systems, this issue is even more relevant. The information employed by ITSs for representing the student's level of knowledge is usually gathered through an assessment process.

In traditional CBT scenarios, the same predefined set of test items was presented to all students, regardless of their ability. The test item within this fixed set were typically selected covering a broad range of ability levels [Pri99]. In these scenarios, it is rather possible that both high-performance students and low-performance students do not count with suited tests for their level of knowledge. Selecting proper test items is very critical to constitute a test that meets multiple assessment criteria, including the number of test items, the specified distribution of course concepts to be learnt, or the difficulty of the test items [Hwa05]. Since satisfying multiple requirements (or constraints) when selecting test items is difficult, most of existing CBT systems construct the tests by manually or randomly selecting test items from their item banks. These systems may count with too large item banks for setting quality tests suited for all students [HYY06]. Therefore, there is a need of efficient methods for automatically obtain effective tests that are suitable for all students regardless their level of performance.

Computerized adaptive tests (CATs) represent an attempt to automate this difficult task. CAT involves issuing questions of a difficulty level that depends on the students previous responses. If a question is answered correctly, the estimate of his/her knowledge level is raised and a more difficult question is presented, and vice versa, giving the potential to test a wide range of student ability concisely [CW05]. That is, CATs systems select the next test item depending on the estimated knowledge level of the student (obtained from the answers to items previously responded). The system is in charge of deciding when the test should finish. For example, a CAT can finish when a specified target measurement has been achieved, when a fixed number of items have been presented, when the time has finished, etc.

Unfortunately, this kind of systems present some disadvantages. First, test items are modelled with probabilistic functions. The values of these functions are inferred from performances of students that have taken this test non-adaptively. Accordingly, in order to be reliable, a CAT generation system should be able to collect valid student performance information to accomplish this inference. This represents a strong limitation in initial phases of the system. Next, CATs systems usually consider difficulty and concept-related distribution of test items without having into consideration user requirements. This model do not allow students to decide their own learning paths. Self-assessment provides a framework where the students can establish their own learning goals, evaluate them, and adapt their learning behaviour in function of the results obtained in the evaluation. Self-assessment is proved to be a well-suited strategy for the development of students' ability of reflection on their own learning, students' ability to learn how to learn, and students' autonomy [Bou95, Mis11, SZ11]. In addition, these systems do not allow teacher to set common goals for all students. During the course, teachers may require to perform a formative assessment of their students. In such case, it is usual that students share common assessment goals, helping this to mark each individual student in the context of the whole class.

For these reasons, we have decided to design a framework where the user takes control of the constraints needed to generate a test. The framework is able to generate test suited to user requirements from large item banks. A number of requirements have been considered trying to cover real-life needs: number of test items, degree of practical-related items in contrast to theoretical-related items, difficulty of the items, frequency of inclusion of the items, and relation of the items to the course issues. The requirements are not static (e.g. test difficulty is 5). In contrast, they are modelled by handling uncertainty. For example, a user can request "minimum of 10% of easy items and minimum of 90% of hard items". The framework is implemented using Fuzzy Logic (FL), as valuable model to handle with uncertainty. Table IV.1 shows a complete example of a user test request. In addition, the requirements can be weighted, i.e. the user can decide which requirements are more important than others. For example, a user may decide that difficulty of the items is less important than the relation to course issues (but considering both requirements). Additionally, the framework consider other important aspects such as auto-levelling of items' difficulty, number of opportunities for passing a test and grading mode (maximum grade, minimum grade, or average grade), or time-constraints for teacher's formative tests.

User requirements for a desired test	Type of requirement
• 50 items	Number of items in resulting test
• Minimum of 40% of practical-related items	Item type
• Maximum of 20% of theoretical-related items	Item type
• Minimum of 40% of easy items	Item difficulty
• Items with low frequency of inclusion	Item frequency of inclusion
• Minimum of 5 items related to issue 1	Item relation to course's issues
• Maximum of 4 items related to issue 2	Item relation to course's issues

Table IV.1: User test request

The framework is useful for both students and teacher. On the one hand, the model promotes student self-assessment. Students can adjust their assessment goals at each step. Starting from unclear and blurred requirements, they can later focus on more specific requirements which better fit their current needs. On the other hand, teachers can set specific assessment goals simulating traditional summative exams, without having to manually create a test for each student. If an individual test is generated to each student, the possibility of cheating will be decreased.

The rest of this section continues as follows. Section 2.2 will review the state-of-the-art in e-assessment related systems. In Section 2.3 will comment the architecture of our proposed Fuzzy Test Generation framework. Next, the usefulness of our framework is discussed in Section 2.4. Finally, Section 2.5 will offer the final discussion and future work.

2.2 Related works

In this section, we firstly review generic e-assessment methods to provide automatic tests. Then, we will discuss some approaches relates with CATs strategies. Finally, a number of approaches related with the automatic generation of test item will be commented.

A number of generic e-assessment tools have been proposed in the literature. The e-Xaminer system [DA07] proposed a study on how to bring effective e-assessment. This computer-aided summative assessment system was designed to produce tests based on parametrically designed questions. The system receives a skeleton question as input from the course instructor and produces a series of different questions from it. The skeleton takes the form of a static template containing a set of parameters. Then the system produces a question for each student, only modifying the value of each parameter. For example, given the skeleton question "which is the best way of sharing the class B host bits between subnets and computers?", the system generates a question for each student modifying uniquely the parameter B. Perry et al. [PBR07] developed a hybrid formative/summative e-assessment tool for an course in Chemical Engineering. It is supported by Respondous v.3.5 and WebCT 4. The tool contains a set of predefined questions from multiple choice to short answers questions. They did not provide an automatic questionnaire generation tool. In contrast, they focused on the logistic process needed to implement this kind of tool. Answers from a questionnaire completed by tutors and students showed that over 80% of the students found the feedback provided by the e-assessment tool to be very useful and helpful in determining the areas of learning that needed improvement. Other e-assessment tool was proposed by Dib and Adamo-Villani [DAV12]. The application was designed to support the learning of surveying concepts and practices. It consists of 2 components: a VLE that is used by the students to review concepts and procedures and perform surveying exercises; and an evaluation engine that tracks the student's interactions with the program and outputs performance reports.

Attending to CATs strategies, we can find different approaches designed to offer personalized tests to students. Several researches have constructed CATs systems based on item response theory (IRT) [UKMJJ11, HCS12]. In order to create an adaptive test, IRT requires parameters, such as item difficulty, which can be determined using the items for a large number of samples first (as pre-calibration) and then manually derive the question parameters. Guzman and Conejo [GC05] developed SIETTE (System of Intelligent Evaluation Using Tests for Tele-education), a Web-based assessment tool. The system creates test tailor-made for a student according to the specifications stored in the knowledge base, using IRT. When a student begins a test, his or her student model is retrieved from the student model repository. If there is no previous information stored about him or her (from earlier test sessions), his or her student model is initialized as a constant knowledge probability distribution curve. During a test session, the next item to be asked to the student is selected adaptively by one of the following alternatives (decided by the teacher when preparing the test): a Bayesian criterion — starting from the distribution of the estimated student knowledge, the selected item is the one that minimizes the sum of the *a posteriori* variances resulting from a correct or incorrect answer to the item; or a difficulty-based criterion — this method selects the item whose difficulty is closer to the estimated knowledge level of the student. Once an item has been selected, it is removed from the set of available items for this test session. After each answer, the student's knowledge level is computed using a Bayesian method. In the same line, Lilley et al. [LBB04] proposed a software prototype based on ITR. The prototype comprised a database containing 250 objective questions related to the use of English language and grammar. A Graphical User Interface was designed to deliver questions simply and effectively for each candidate. The adaptive algorithm used in the prototype was based on the Three-Parameter Logistic Model (3-PL) from Item Response Theory. Meanwhile, Hwang et al. [Hwa03] proposed a different approach where test generation problem was formulated as a dynamic programming model to minimize the distance between the parameters (e.g., discrimination, difficulty, etc.) of the generated test and the objective values subject to the distribution of concept weights. Anbo et al. [MYRP07] developed an approach based on genetic algorithms. They developed a multi-agent system applied to test generation. Individual chromosome in the genetic algorithm represents a potential solution, i.e. a test. Each test item is identified by its unique number. Consequently, each chromosome consists of some ordering number of items. They selected crossover and mutation as genetic operators. Next, they designed three objective functions in basis of the difficult of each item, the grade of mastery of the item for the current student, and the item type. The functions measure the percentage of such factors in relation to a set of constraints conditions (total difficulty, desired grade of mastery, and desired item type). Such constraints conditions are manually setted by the teacher. Tabu Search has been also applied in order to obtain adapted test [HYY06]. The Tabu search starts with an initial configuration chosen at random and then moves iteratively from one configuration to another until a certain stopping criterion is satisfied. Along each iteration, a set of candidate moves is considered, and thus a neighbourhood of the current configuration is identified. When a move is taken, the direction is recorded in the Tabu list, and the move will not be revisited in the next few iterations. The algorithm was embedded in a computer skill-certification system with large-scale test banks that are accessible to students and instructors through the World Wide Web.

Another related research field concerns with the automatic generation of tests items from textual fields. Although we do not consider this field in this work, we provide some examples in the following. Liu et al. [LWGH05] employed techniques for word sense disambiguation to retrieve sentences from a corpus in which the answer carries a specific sense. Then they applied a collocation-based method for selecting distractors. Brown et al. [BFE05] proposed a system for automatic generation of test items related to vocabulary assessment. In order to test a student's knowledge of a preselected input word, their system produces a non-interrogative stem on the basis of the word's definition or an example of its use in WorNet. Mitkov and Ha [MAHK06] proposed a NLP-based approach for generating test from narrative texts. Their approach uses a simple set of transformational rules, a shallow parser, automatic term extraction, word sense disambiguation, a corpus and WordNet. The method firstly identifies domain specific terms in the corpus which serve as anchors of each question. Document frequency is employed in order to rank the candidate terms, which were formerly extracted using NL patterns. Subsequently, their method employs WordNet to compute concepts semantically close to the correct answer, which are then selected as distractors. Finally, the test questions are generated from declarative sentences using simple transformation rules.

2.3 Framework Overview

In this section we present our test generation framework. We follow the methodology developed by our research group in the work [VLC02]. This framework can generate tailored assessment test from a set a user-selected requirements dynamically. We call *assessment objectives* to the desired assessment requirements. In addition, items in the item bank store a set of *properties* related to the possible objectives. The initial value for each property was manually setted during the creation of the item. We analyse this properties below.

The generation process is carried out by the Test Generation module. It receives the selected objectives and the available items from the item bank as input. The configuration of objective is called *objective base*. Given a item, its properties and the properties of the items previously included in the test are contrasted against the objective base. In this process, the test generation module calculates acceptance thresholds for the item properties. If all the properties exceed the corresponding acceptance thresholds, the item is selected as candidate. Then the best item from the complete set of candidates is included in the test. Subsequently the process is repeated as many times as items are needed.

Both students and teacher can define the configuration of objectives that they prefer in order to generate a test. Logically, tests generated by teachers are assigned to students in their course. In addition, teachers can set exclusive parameters: number of opportunities for passing a test, grading mode (maximum grade, minimum grade, or average grade), and time-constraints. The grading mode is related to the number of opportunities. For example, if a student performs a test three times, the grading mode will determine a global grade considering the maximum, the minimum or the average grade from the three results. In addition, the time-constraints limit the execution of tests to the selected date ranges. It should be pointed out that a new test will be generated for each one of the opportunities. That is, the objective base is stored and a different test is obtained in real-time every time a student performs the corresponding opportunity.

Finally, when a student completes a test, our method returns a interface containing the results. In addition, during the marking process the difficulty of the items is readjusted. Figure iv.2 shows the architecture of our test generation framework. The architecture is flexible and modular. In addition, the framework has been modelled in a easily interpretable form. This allows non-technical users to work with our framework. We detail the modules of the framework below.

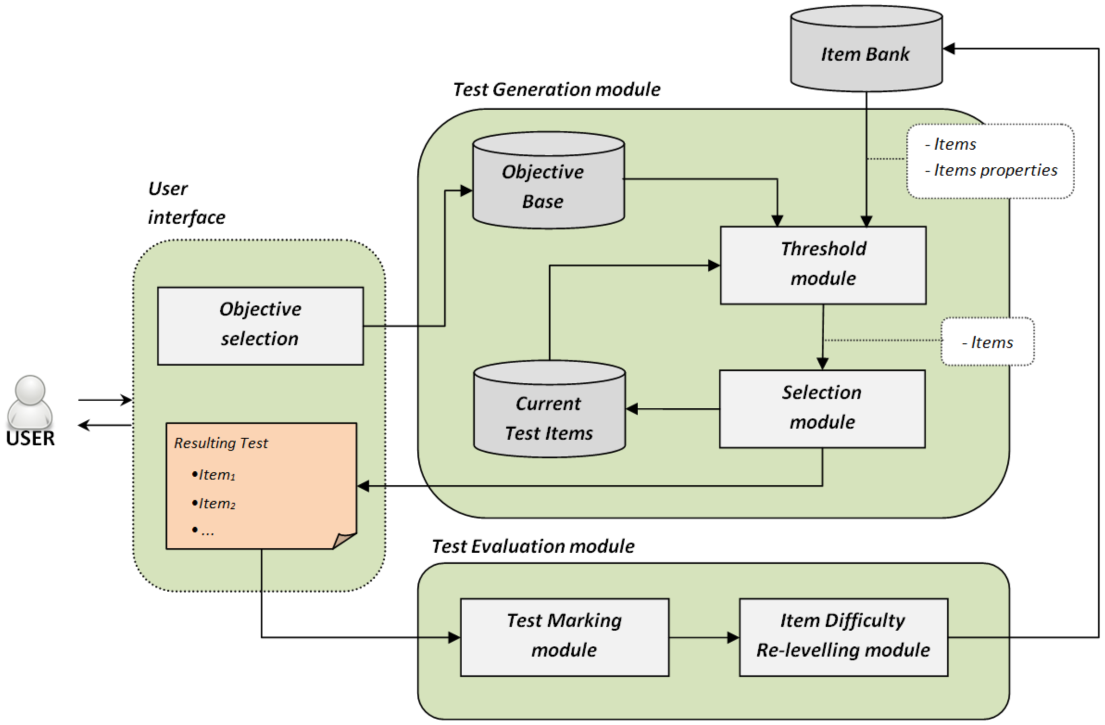


Figure iv.2: Test Generation Framework architecture

It should be pointed out that we have employed GIFT quiz questions (Chapter V Section 3.2.3) as test items. Nevertheless, the framework is not dependant on the type of assessment resource, and therefore any other type could be used.

2.3.1 Test Items Properties

Each item in the item bank is defined by a set of properties. The properties of the items are employed for determining the quality of the item in regard to the pursued assessment objectives. Thus each property will be evaluated through a evaluation function in the domain of the property. Most of the values of the properties have to be manually setted before the items are included in the item bank. For example, the item bank could be created by the students in the course following a constructivist approach (see Chapter V). In that case, the given values should be contrasted with a domain expert (the teacher) in order to assure the correctness of the process. The following properties have been considered:

Type of Item The type of an item is referred to the educational area which the item is related with. The property is composed by two variables: $T \in [0, 1]$ called "theoretical degree"; and $P \in [0, 1]$ called "practical degree". For example, if a item i is totally theoretical, the two variables defining the property will present the values $T(i) = 1$ and $P(i) = 0$. In contrast, if i is totally practical, the variables will take the values $T(i) = 0$ and $P(i) = 1$. In addition, i could be theoretical and practical at the same time with a given grade. That is, the sum of $T(i)$ and $P(i)$ does not have to be equal to 1.

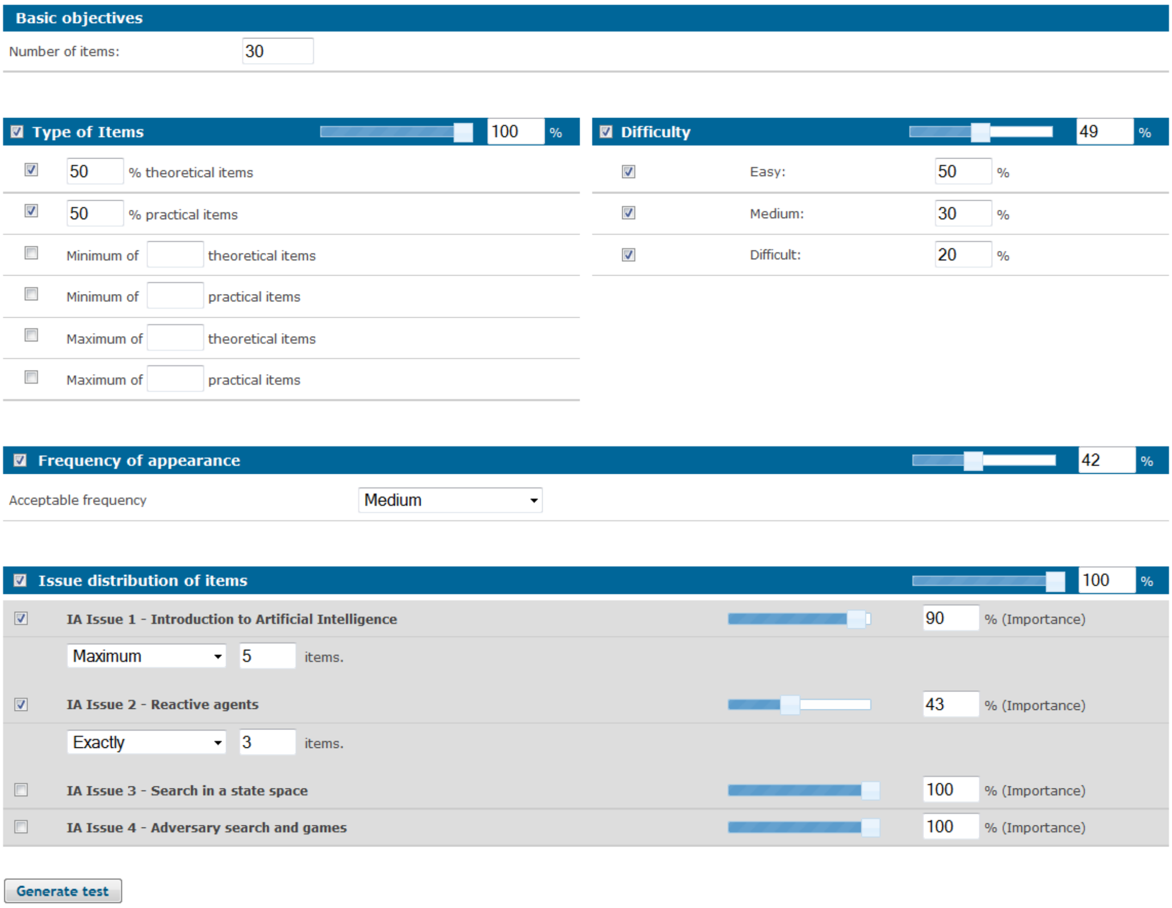
Item Difficulty This property defines the difficulty of a item i, $D(i)$, within the course. It is defined in basis of a variable: $D \in [0, 10]$. The initial value comes from the human expert. Subsequently, the difficulty of the item will be re-levelled in function of the results obtained after the execution of the tests including it.

Item Frequency of Appearance This property is referred to the ratio of frequency in which an item has been included in tests. It is defined in basis of the variable $F \in [0, 1]$. The value is dynamically modified after each test generation process by means of the following function: $F(i) = \dfrac{Fab_i}{A_i}$, where Fab_i is a counter containing the number of times i has been included into a test, and A_i is a counter containing the number of tests generated after i was included in the item bank.

Item Relevance to Course Issues Each item in the bank presents a degree of relevance in regard to the course issues. This property stores as many variables as issues in the course. Given a issue j of the course, the variable $CI^j \in [0, 1]$ defines the degree of relatedness of an item. For example, if a item is strongly related to the issue "Neural networks", the value of the item i for this property is $CI^{neural\ networks}(i) = 1$. A item could be related to one ore more issues of the course.

2.3.2 Objective Base

The objective base defines the minimum requirements pursued in order to generate a test. Users can select which objectives they desire trough the objective interface (Figure iv.3). Objectives present a weight ranged between the interval [0,100], representing the importance of each one. The weights will be used during the selection process. The selected objectives conforms the objective base.



Basic objectives

Number of items: 30

☑ Type of Items 100 %

☑ 50 % theoretical items

☑ 50 % practical items

□ Minimum of theoretical items

□ Minimum of practical items

□ Maximum of theoretical items

□ Maximum of practical items

☑ Difficulty 49 %

☑ Easy: 50 %

☑ Medium: 30 %

☑ Difficult: 20 %

☑ Frequency of appearance 42 %

Acceptable frequency: Medium

☑ Issue distribution of items 100 %

☑ IA Issue 1 - Introduction to Artificial Intelligence 90 % (Importance)

Maximum 5 items.

☑ IA Issue 2 - Reactive agents 43 % (Importance)

Exactly 3 items.

□ IA Issue 3 - Search in a state space 100 % (Importance)

□ IA Issue 4 - Adversary search and games 100 % (Importance)

Figure iv.3: Assessment objectives selection interface

Assessment objectives are related to the items' properties. Objectives are not static. In contrast, users can establish quantifiers (minimum, maximum, around) and weights to each objective defining its importance within the whole set of objectives. For this reason, objectives are modelled by means of fuzzy linguistic labels [Zad65] due to fuzzy set theory allows to quantify the non-stochastic uncertainty induced from subjectivity, vagueness and imprecision [ZN09a]. Once the objectives are fixed, the level of acceptance of each item is calculated using a set of evaluation functions, as it is explained below. It should be remarked that users are not forced to set all the objectives. For example, only the number of items and expected difficulty could be setted. In this work, we have considered the following assessment objectives:

(1) Number of Items This objective defines the total number of items to be included in the test. User can generate tests from 5 to 100 items. The number of items will be employed in subsequent processes of the generation. The setting of this objective is obligatory.

(2) Type of Items This objective is referred to the balance between theoretical and practical items in the desired test. The user can define this objective by two schemes: (1) using percentages or (1) using linguistic quantifiers.

On the one hand, if the percentage of theoretical and practical items is established, the objective is modelled as two fuzzy triangular sets. Each percentage is normalized in the range [0,1] and the resulting value is taken as centre of the fuzzy set, with the following membership function:

$$\mu_o(x) = \begin{cases} 0, & if x \leq (np - 0.1) \\ \frac{x-(np-0.1)}{np-(np-0.1)}, & if\ (np-0.1) < x \leq (np+0.1) \\ \frac{(np+0.1)-x}{(np+0.1)-np}, & if\ np < x \leq (np+0.1) \\ 0, & if\ x > (np+0.1) \end{cases} \tag{IV.1}$$

where np is the normalized percentage.

Let us explain this with an example. Let us assume that a user has selected 40% theoretical items and 30% practical items. This is modelled by means of two fuzzy sets centred in 0.4 and 0.3 respectively (Figure iv.4).

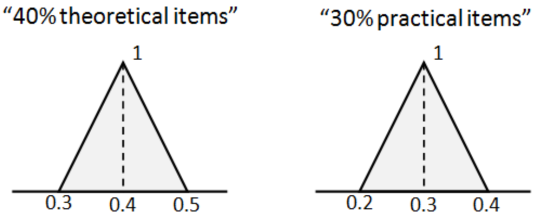


Figure iv.4: Type of Item Objective defined using percentages

On the other hand, the user can employ the quantifiers "minimum" and "maximum" of question of each type. In such case, the objective is modelled as a rectangular fuzzy set. The value associated to the quantifier is converted into percentages normalized in the interval [0,1]. Then the following fuzzy set is modelled:

$$\mu_o(x) = \begin{cases} 0, & if x < np \\ 1, & x \geq np \end{cases} \tag{IV.2}$$

where np is the normalized percentage.

Let assume a user selects the objectives "minimum 5 theoretical items" and "maximum 6 practical items". The values 5 and 6 are converted into percentages normalized in the interval [0,1] $np_{ti} = (5 \cdot 100 \div number\_of\_items) \cdot (1 \div 100)$ and $np_{pi} = (6 \cdot 100 \div number\_of\_items) \cdot (1 \div 100)$. Let assume that the desired number of items is 10. Therefore, $np_{ti} = 0.5$ and $np_{pi} = 0.6$. Then the resulting fuzzy sets are modelled (Figure iv.5).

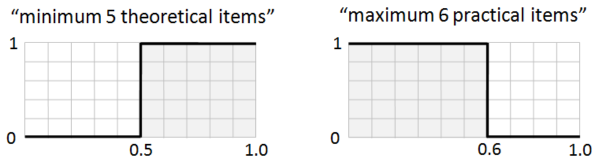


Figure iv.5: "Type of Item" Objective defined using linguistic labels

(3) Test Difficulty The expected difficulty of the test is defined in basis of a fuzzy linguistic variable $Difficulty = \{Easy, Medium, Difficult\}$ (Figure iv.6). In addition, for each linguistic label the user can establish a percentage of desired items. The percentage is modelled following the prior approach: a triangular fuzzy set centred in the percentage (normalized between 0 and 1) is defined.

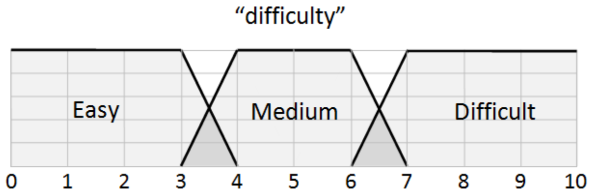


Figure iv.6: Difficulty fuzzy variable

(4) Acceptable Item Frequency This objective represents in which level the repetition of a item is acceptable. The objective is modelled by means of a fuzzy linguistic variable $Frequency = \{Low, Medium, High\}$. The sets corresponding to the linguistic labels are detailed in Figure iv.7. In this occasion, no other value is needed.

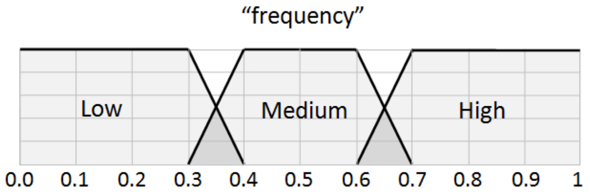


Figure iv.7: (Acceptable) Frequency fuzzy variable

(5) Test Relevance to Course Issues The last objective considered is the test relevance to the course issues. Users can set objectives for one or more issues of the course, and establish a quantifier (minimum, exactly, maximum) related to the number of items for each issue. Also the importance of each issue can be defined by means of a weight range in the interval [0,100]. The issues' weights will act during the candidate selection stage.

The modelling is similar to the type of items objective. For the quantifiers "minimum" and "maximum" the process is identical. The quantifier "exactly" is modelled as a singleton. The value associated to the quantifier is converted into percentages normalized in the interval [0,1]. The singleton is then modelled with the following membership function:

$$\mu_o(x) = \begin{cases} 0, & if\ x < np \\ 1, & if\ x = np \\ 0, & if\ x > np \end{cases} \tag{IV.3}$$

where np is the normalized percentage.

Let us illustrate this with an example. Given the objective "exactly 5 items of the issue 1", the value 5 is converted into a percentage and normalized between [0,1]: $p_{i1} = (5 \cdot 100 \div \mathit{number\_of\_items}) \cdot (1 \div 100)$. Considering a number of items equal to 10, the resulting singleton is establish in 0.5 (Figure iv.8).

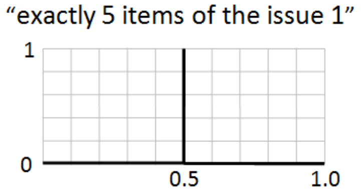


Figure iv.8: Test Relevance to issue 1 using the "exactly" quantifier

2.3.3 Acceptance Threshold Calculation module

The goal of this module is to calculate acceptance thresholds for the items properties according to the objective base OB. Remember that the process takes into consideration the current item and the items previously selected as part of the test. The process is performed as follows.

Let us assume that $i_1, i_2, \ldots, i_k$ is the sequence in items already included in the test. Given an objective $o \in OB$ and the current item i_{k+1}, the module calculates an range of acceptance $R_o^{i_{k+1}} = [\Gamma_o(i_1, \ldots, i_{k+1}),\ \Gamma'_o(i_1, \ldots, i_{k+1})]$ considering that i_{k+1} would be included in the test. The left bound represents the current level of acceptance of o, Γ_o, and the right bound represents the reachable level of acceptance of o, Γ'_o. Then, the threshold $th_o(i_{k+1})$ is calculated taking now into consideration the fuzzy set representing o (explained in above subsection). If the interval $R_o i_{k+1}$ totally contains the base of the fuzzy set, then $th_o(i_{k+1}) = 1$. In any other case, $th_o(i_{k+1}) = MAX(\mu_o(\Gamma_o), \mu_o(\Gamma'_o))$. Finally, if $\forall o \in OB, th_o(i_{k+1}) > 0$, then i_{k+1} is considered candidate item, and it is included in the set of candidates CI.

Γ_o is calculated taking into considerations the value of the corresponding properties of the input items. Each objective[1] presents an evaluation function which are detailed as follows.

Theoretical Type of Items Given the set of k items already included in the test, $numItems$ representing the final number of items to be included in the test, and i_k+1 the current item, the evaluation function for a "theoretical type of item" objective o are calculated as

$$\Gamma_o(i_1, \ldots, i_{k+1}) = \frac{\sum_{j=1}^{k+1} T(i_j)}{numItems} \tag{IV.4}$$

and

$$\Gamma'_o(i_1, \ldots, i_{k+1}) = \frac{\sum_{j=1}^{k+1} T(i_j)}{numItems} + \frac{numItems - (k+1)}{numItems} \tag{IV.5}$$

Therefore, the range of acceptance $R_o(i_{k+1})$ is defined as the interval:

[1] Although the desired number of items is an assessment objective, it is not evaluated as the rest. In contrast, it is used as parameter for the whole process.

$$R_o^{i_{k+1}} = [\frac{\sum_{j=1}^{k+1} T(i_j)}{numItems}, \frac{\sum_{j=1}^{k+1} T(i_j)}{numItems} + \frac{numItems - (k+1)}{numItems}] \tag{IV.6}$$

Let us illustrate this process with an example. Let us assume that a user has selected the following objectives: "number of items equal to 10", and "40% theoretical items", and 3 items has been previously included in the test. The three items present the following theoretical degrees: $T(i_1) = 0.3$, $T(i_2) = 0.7$, $T(i_3) = 0.5$. The second objective, o, is modelled as a triangular fuzzy set with the membership function:

$$\mu_o(x) = \begin{cases} 0, & if\ x \leq 0.3 \\ \frac{x-0.3}{0.4-0.3}, & if\ 0.3 < x \leq 0.4 \\ \frac{0.5-x}{0.5-0.4}, & if\ 0.4 < x \leq 0.5 \\ 0, & if\ x > 0.5 \end{cases} \tag{IV.7}$$

Now a a new item i_4 with $T(i_4) = 0.2$ is checked. The range $R_o^{i_4}$ is calculated as follows:

$$R_o^{i_4} = [0.14, 0.14 + \frac{10-4}{10}] = [0.17, 0.77] \tag{IV.8}$$

As it could be observed, the left bound of $R_o^{i_4}$ is lower than the left bound of the fuzzy set ($0.17 < 0.3$), and the right bound of $R_o^{i_4}$ is higher than the right bound of the fuzzy set ($0.77 > 0.5$). Therefore, $th_o(i_4) = 1$ and i_4 will be considered candidate item if the rest of the threshold are exceeded.

Practical Type of Items Given the set of k items already included in the test, $numItems$ representing the final number of items to be included in the test, and i_k+1 the current item, the evaluation function for a "practical type of item" objective o is calculated as

$$\Gamma_o(i_1, \ldots, i_{k+1}) = \frac{\sum_{j=1}^{k+1} P(i_j)}{numItems} \tag{IV.9}$$

and

$$\Gamma'_o(i_1, \ldots, i_{k+1}) = \frac{\sum_{j=1}^{k+1} P(i_j)}{numItems} + \frac{numItems - (k+1)}{numItems} \tag{IV.10}$$

Therefore, the range of acceptance $R'_o(i_{k+1})$ is defined as the interval

$$R_o(i_{k+1}) = [\frac{\sum_{j=1}^{k+1} P(i_j)}{numItems}, \frac{\sum_{j=1}^{k+1} P(i_j)}{numItems} + \frac{numItems - (k+1)}{numItems}] \tag{IV.11}$$

Let us illustrate this process with an example. Lets assume that a user has selected the following objectives: "number of items equal to 10", and "30% practical items", and 3 items has been previously included in the test. The three items present the following practical degrees: $P(i_1) = 0.8$, $P(i_2) = 0.7$, $P(i_3) = 0.6$. The second objective, o, is modelled as a triangular fuzzy set with the membership function:

$$\mu_o(x) = \begin{cases} 0, & if\ x \leq 0.2 \\ \frac{x-0.2}{0.3-0.2}, & if\ 0.2 < x \leq 0.3 \\ \frac{0.4-x}{0.4-0.3}, & if\ 0.3 < x \leq 0.4 \\ 0, & if\ x > 0.4 \end{cases} \tag{IV.12}$$

Now a a new item i_4 with $P(i_4) = 0.7$ is checked. The range $R_o^{i_4}$ is calculated as follows:

$$R_o^{i_4} = [0.28, 0.28 + \frac{10-4}{10}] = [0.28, 0.88] \tag{IV.13}$$

Then, the threshold $th_o(i_4) = MAX(\mu_o(0.28), \mu_o(0.88)) = MAX(0.8, 0) = 0.88$ is obtained and i_4 will be considered candidate item if the rest of the threshold are exceeded.

Acceptable Item Frequency Given the set of k items already included in the test, *numItems* representing the final number of items to be included in the test, and i_k+1 the current item, the evaluation function for a "acceptable frequency" objective o is calculated as

$$\Gamma_o(i_1, \ldots, i_{k+1}) = \frac{\sum_{j=1}^{k+1} F(i_j)}{numItems} \tag{IV.14}$$

and

$$\Gamma'_o(i_1, \ldots, i_{k+1}) = \frac{\sum_{j=1}^{k+1} F(i_j)}{numItems} + \frac{numItems - (k+1)}{numItems} \tag{IV.15}$$

Let us illustrate this process with an example. Lets assume that a user has selected the following objectives: "number of items equal to 10", and "low acceptable frequency", and 6 items has been previously included in the test. The three items present the following frequencies: $F(i_1) = 0.5$, $F(i_2) = 0.5$, $F(i_3) = 0.6$, $F(i_4) = 0.5$, $F(i_5) = 0.6$, and $F(i_6) = 0.5$. The second objective, o, is directly affected by the fuzzy linguistic variable *Frequency*. Concretely, the linguistic label *Low* with the following membership function is considered:

$$\mu_{Low}(x) = \begin{cases} 1, & if\ 0.0 \geq x < 0.3 \\ \frac{0.3-x}{0.4-0.3}, & if\ 0.3 < x \leq 0.4 \\ 0, & if\ x > 0.4 \end{cases} \tag{IV.16}$$

Now a a new item i_7 with $F(i_7) = 1.0$ is checked. The range $R_o^{i_7}$ is calculated as follows:

$$R_o^{i_7} = [0.41, 0.41 + \frac{10-4}{10}] = [0.41, 1.01] \tag{IV.17}$$

Then, the threshold $th_o(i_7) = MAX(\mu_{Low}(0.41), \mu_{Low}(1.01)) = MAX(0, 0) = 0$ is obtained and i_7 is discarded.

Test Difficulty Given the set of k items already included in the test, $numItems$ representing the final number of items to be included in the test, and i_k+1 the current item, the evaluation function for a "difficulty" objective o is calculated as

$$\Gamma_o(i_1, \ldots, i_{k+1}) = \frac{\sum_{j=1}^{k+1} \mu_{difficulty}(D(i_j))}{numItems} \tag{IV.18}$$

and

$$\Gamma'_o(i_1, \ldots, i_{k+1}) = \frac{\sum_{j=1}^{k+1} \mu_{difficulty}(D(i_j))}{numItems} + \frac{numItems - (k+1)}{numItems} \tag{IV.19}$$

Let us illustrate this process with an example. Lets assume that a user has selected the following objectives: "number of items equal to 10", and "50% difficult items", and 4 items has been previously included in the test. The three items present the following difficulties: $D(i_1) = 5$, $D(i_2) = 6$, $D(i_3) = 8$, and $D(i_4) = 7$. The value of the second objective, o, is modelled as a triangular fuzzy set with the membership function:

$$\mu_o(x) = \begin{cases} 0, & if\ x \leq 0.4 \\ \frac{x-0.4}{0.5-0.4}, & if\ 0.4 < x \leq 0.5 \\ \frac{0.6-x}{0.6-0.5}, & if\ 0.5 < x \leq 0.6 \\ 0, & if\ x > 0.6 \end{cases} \tag{IV.20}$$

Now a a new item i_5 with $D(i_5) = 2$ is checked. The range $R_o^{i_5}$ is calculated by checking the membership degree of the items in regard to the fuzzy linguistic label $Difficult$ belonging to the fuzzy variable $Difficulty$. Remember that it had the following membership function:

$$\mu_{Difficult}(x) = \begin{cases} 0, & if\ x \leq 6 \\ \frac{x-6}{7-6}, & if\ 6 < x \leq 7 \\ 1, & if\ x > 7 \end{cases} \tag{IV.21}$$

The range $R_o^{i_5}$ is calculated as follows:

$$R_o^{i_4} = [0.2, 0.2 + \frac{10-4}{10}] = [0.2, 0.8] \tag{IV.22}$$

Then, the threshold $th_o(i_5) = 1$ is obtained and i_5 will be considered candidate item if the rest of the threshold are exceeded.

Test Relevance to Course Issues Given the set of k items already included in the test, $numItems$ representing the final number of items to be included in the test, and i_k+1 the current item, the evaluation function for a "relevance to a curse issue s" objective o is calculated as

$$\Gamma_o(i_1, \ldots, i_{k+1}) = \frac{\sum_{j=1}^{k+1} CI^s(i_j)}{numItems} \tag{IV.23}$$

and

$$\Gamma'_o(i_1,\ldots,i_{k+1}) = \frac{\sum_{j=1}^{k+1} CI^s(i_j)}{numItems} + \frac{numItems-(k+1)}{numItems} \tag{IV.24}$$

Let us illustrate this process with an example. Lets assume that a user has selected the following objectives: "number of items equal to 10", and "exactly 1 item related to issue 3", and 3 items has been previously included in the test. The three items present the following relevance to issue 3: $CI_3(i_1) = 1$, $CI_3(i_2) = 0$, $CI_3(i_3) = 0$. The second objective, o, is modelled as a singleton with the membership function:

$$\mu_o(x) = \begin{cases} 0, & if\ x < 0.1 \\ 1, & if\ x = 0.1 \\ 0, & if\ x > 0.1 \end{cases} \tag{IV.25}$$

Now a a new item i_4 with $CI_3(i_4) = 1$ is checked. The range $R_o^{i_4}$ is calculated as follows:

$$R_o^{i_4} = [0.2, 0.2 + \frac{10-4}{10}] = [0.2, 0.8] \tag{IV.26}$$

Then, the threshold $th_o(i_4) = MAX(\mu_o(0.2), \mu_o(0.8)) = MAX(0,0) = 0$ is obtained and i_4 is discarded.

2.3.4 Item Selection module

This module is in charge of selecting the next item to be added to the test. The item will be selected from the candidate item set CI, obtained by the threshold module. In this stage, the weights of the objectives $w_o, o \in OB$ are taken into account. The process is straightforward.

To that end, the function $\Psi_i(\{th_o(i) : o \in OB\}, \{w_o : o \in OB\}) = \prod th_o(i) \cdot w_o, \forall i \in CI$ measures the quality of each item in CI, where w_o denotes the weight given by the user to the objective o. If only one item obtains the higher score, it is added to the test. In other case, a random item from the set of higher scored items is added. Subsequently, the acceptance thresholds for the items in the item banks are recalculated, taking also into consideration the new inserted item. The resulting candidate items are measured, and one of them is then inserted in the test. The process iterates until the test is complete.

It is possible that no items surpass the threshold module. In that case, the first item from the current test items is removed and reinserted in the item bank. From here, the process continues normally. If after three iterations, no item has been inserted in the test, the process is stopped and a error message is shown to the user. Nevertheless, if the bank item is high enough this situation would be infrequent.

Finally, it the test is generated by a student, it is directly presented in the interface. In contrast, teacher generated tests are stored and assigned to the students. When a student access to the framework, an information message informs the student that a new teacher test is accessible.

2.3.5 Test Evaluation Module

This module is in charge of marking the tests performed by students, and re-levelling the difficulty of the questions in function of the results.

Since we have employed GIFT quiz questions as items in this work, the marking process is direct. GIFT format includes the right answer for each item, and direct feedback related to each answer. For this reason, we have not made additional efforts in regard to this task. When the module corrects the test, the user interface shows the results by means of two models. If the test was generated by the students, the interface shows the results as well as the right answers for all the items. This allows students to check their exact progresses and perform a self-regulation learning progress. In contrast, if the test was assigned by the teacher, only the mark is offered (it is supposed to be a formative assessment test). We show the result interface in Figure iv.9.

In addition, a re-levelling process for the difficulty of the items taking place in the test is performed. Each time an item is corrected, the difficulty of such item i is readjusted in basis of the following function:

$$Difficulty(i) = D_p(i) \cdot \frac{Fab_i - 1}{Fab_i} + \frac{Fsuc_i \cdot 10}{Fab_i} \cdot \frac{1}{Fab_i} \tag{IV.27}$$

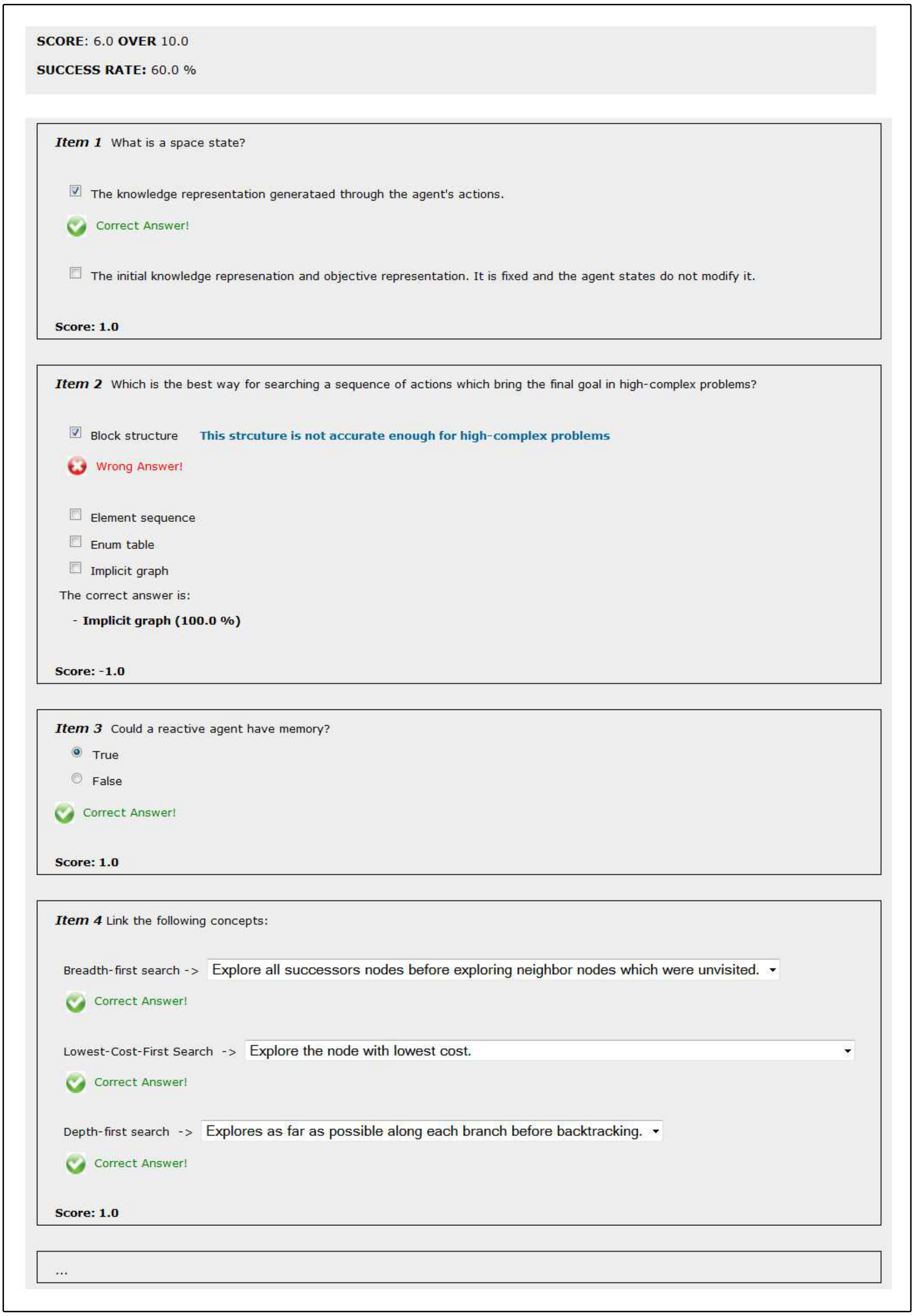
where Fab_i is the counter containing the number of times i has been included into a test, and $Fsuc_i$ is referred to the number of times i has been correctly answered.

2.4 Experiences with the System

In this section, we provide an evaluation of our proposed Fuzzy Test Generation Framework (FTGF). To that end, we conducted an usability study during the application of framework as an addition to the lectures of the undergraduate course “Artificial Intelligence” in the Computer Engineering degree in the University of Granada from September 2011 to July 2012. Concretely, the evaluation approach was made in one of the four groups in which the course is divided (group A). During this year, the framework was accessible as a stand-alone VLE, including uniquely support for the managing of GIFT quiz questions, and the functionality commented in this section. In order to clarify the reading, we will call FTGF to this stand-alone configuration of the VLE. Therefore the traditional face-to-face lectures were then widened by the use of the FTGF, making up a blended learning scenario.

During the evaluation period, FTGF was available without any location/time restriction. In this regard, FTGF was stored in a university server under Apache Tomcat 6 (6.0.32) to manage the data storage and was accessible via web application. During the progress of the courses, a technical assistant was available to assure the correct operation of the framework and to manage the technical issues reported by the users. All participants including the teacher and the students were instructed with specific briefing lessons.

As first part of this evaluation, the teacher in charge of the experimental group was asked about different elements of the FTGF. First, he assessed the quality of the framework’s interface. Regarding his opinion, it was assumed that the interface did not present any major usability problems, being clear and easy-to-use. In addition, the teacher was asked to evaluate the FTGF effectiveness in terms an academic context. He suggested that the framework was suitable for high education courses. Students had accessible a self-assessment method where the generated test were always fitted to the real student needs. The interface was simple enough to be understandable by non-technical students, so engineering students would not experiment major difficulties. Regarding the



SCORE: 6.0 **OVER** 10.0

SUCCESS RATE: 60.0 %

Item 1 What is a space state?

☑ The knowledge representation generataed through the agent's actions.

✓ Correct Answer!

□ The initial knowledge represenation and objective representation. It is fixed and the agent states do not modify it.

Score: 1.0

Item 2 Which is the best way for searching a sequence of actions which bring the final goal in high-complex problems?

☑ Block structure **This strcuture is not accurate enough for high-complex problems**

☒ Wrong Answer!

□ Element sequence

□ Enum table

□ Implicit graph

The correct answer is:

- **Implicit graph (100.0 %)**

Score: -1.0

Item 3 Could a reactive agent have memory?

● True

○ False

✓ Correct Answer!

Score: 1.0

Item 4 Link the following concepts:

Breadth-first search -> Explore all successors nodes before exploring neighbor nodes which were unvisited. ▾

✓ Correct Answer!

Lowest-Cost-First Search -> Explore the node with lowest cost. ▾

✓ Correct Answer!

Depth-first search -> Explores as far as possible along each branch before backtracking. ▾

✓ Correct Answer!

Score: 1.0

...

Figure iv.9: Test Results interface

academic staff, the framework could be useful in terms of efficiency and error reduction, especially when marking a large volume of tests. Additionally, the generation of test for formative assessment (teacher generated tests) was perceived as a perfect model for increasing test security. That is it would become more difficult to “cheat” during assessment periods since the set of questions is not the same for all participants.

After collection the opinions of the teacher, we conducted a study the students of the experimental group. The FTGF was available for students in group A of the course. The use of the VLE was voluntary. At the end of the period, volunteer students were asked about their opinion about the FTGF. The experimental group was then composed by 38 students, 5 were women (13.15%) and 33 men (86.85%). As an addition of the course lectures, students had available the self-assessment tool. A total of 494 self-assessment tests were performed. Each student realised 13 tests on average (M = 13.00, SD = 12.14). Having into account the preliminary nature of this experiment, the teacher did not employ the tool to generate summative assessment exams this year. Therefore, all students performed the traditional final exam, which was common to all students. The decision was taken in order to avoid negative reactions of students in case of they would not be satisfied with the new framework.

After the course period, we collect the students' opinions using a usability questionnaire. The questionnaire was composed by 5 items based on a five-point Likert scale. The scale ranges from strongly agree (5 points) to strongly disagree (1 point). The description of the items and the mean and standard deviation rates assigned by the students is shown in Table IV.2. In addition, each participant gave a brief summary of their attitude to the framework. As it can be seen, student participants reported that they found the framework easy to use, even without prior training. They perceived the user interface as being quite usable and easy to understand. Participants said that they considered the resulting test highly related to the setted assessment objectives. In this line, they felt comfortable since they were able to set the assessment objectives that they considered each time. Thus it was suggested that a test tailored for the needs of an individual student was more likely to improve his or her enthusiasm and motivation during the assessment session than those that are static. Finally, most of them agreed that the framework was a good tool for developing the formative assessment processes. They considered test form much more agile and efficient than traditional paper and pen exams. Nevertheless, some students indicated that each the combination of tests and coursework would be the most suitable option.

Item	Students (N = 38)	
	M	*SD*
1. The framework is intuitive and easy-to-use	4.23	0.81
2. The implemented assessment objectives are sufficient to generate tests that fit my needs	4.13	0.84
3. The framework allows me to self-regulate my learning process	3.81	0.98
4. The frameworks allows me to test efficiently my knowledge in real-time	3.84	1.06
5. I think that the framework is a suited tool for realising formative assessment	4.10	0.86

Table IV.2: Results of the 5 items of the usability questionnaire

Finally, the teacher was asked again about the achievement of the students in the experimental group. He found that volunteer students severely outperformed the rest of students in the group in terms of final grade. What is more, volunteer students felt a greater level of satisfaction among the whole course, which was reflected in the final satisfaction questionnaire carried out by the University of Granada. The university develop systematically a procedure with the aim of collecting the student opinions about the main factors involved in each course since 2008. To that end, an independent organization named the Prospective Andalucian Centre[2] (PAC) manage a questionnaire that is completed by students of each course in the UGR. As summary of the questionnaire, a 5 point Likert-scale item collects the overall opinion about the course. Group A (experimental group) obtained a rate very high rate (M = 4.24, SD = 0.85). The average rate obtained by all the courses and groups of the whole Computer Engineering degree was of 3.81 points (SD = 106), existing a negative difference of 10%.

2.5 Conclusions and Future Work

In this part of the dissertation, we have proposed an intelligent approach for the automatic generation of assessment tests. The method has been proposed in order to overcome the common limitations proper of traditional Computer-Based Test (CBT) systems. In CBT scenarios, all students are usually forces to answer the same set of items. Tests are not tailored for the individual needs of students and they provide very little information about student performance.

We have taken the underlying ideas of Computerized Adaptive Tests (CAT) in order to implement our framework. CAT involves issuing questions of a difficulty level that depends on the students previous responses. If a question is answered correctly, the estimate of his/her knowledge level is raised and a more difficult question is presented. Nevertheless, this kind of systems does not allow students to decide their own learning paths. In addition, these systems do not allow teacher to set common goals for all students. During the course, teachers may require to perform a formative assessment of their students. In such case, it is usual that students share common assessment goals, helping this to mark each individual student in the context of the whole class.

For these reasons, we have designed a framework where the user takes control of the constraints needed to generate a test. The framework is able to generate test suited to user requirements from large item banks. The framework was tested in experimental group obtaining excellent opinions from both students and the teacher. Hence, this guarantees that the correct operand of the framework as engine of the Intelligent Tutoring System proposed in the following section.

Regarding future work, only GIFT quiz question has been included during the experimental evaluation of the framework. Although the architecture in independent of the employed test items, it would be desirable to analyse further students opinions considering other kind of assessment resources.

[2] http://huespedes.cica.es/canp/index.html

3 TTutor: Integrating Fuzzy Logic into the Student Model Diagnosis of a Intelligent Tutoring System based on E-Assessment

3.1 Introduction

As we seen in the Introduction section of this chapter, Intelligent Tutoring Systems (ITSs) are asynchronous systems able to provide advanced learning strategies, by adapting the content and its presentation to individual student's characteristics. This kind of systems have gained popularity for their ability to cope student's cognitive overload or disorientation[CD08].

Possibly the most common methodology employed for providing personalized learning content is the *curriculum sequencing* [BP03, CLC05, HHC07]. The idea of curriculum sequencing is to provide each student with a individualized optimal teaching operation (presentation, examples, questions, ...) at any given moment [HHC07]. That is, the system must generate the operation in the context of available operations which brings the student closest to the learning goal. Usually, the goal will be to learn the knowledge of the domain up to a specific level.

The personalization in ITSs is done through the *student model*, which collects the information that represents the current student's knowledge of the subject domain being studied. The student modelling can be defined as the process of gathering relevant information in order to infer the current cognitive state of the student, and to represent it so as to be accessible and useful to the ITS for offering adaptation [TM09]. Notwithstanding, determining the student model is difficult and is still an open a challenging research task. First, it has to be considered what information about the student's actuation should be gathered, how to obtain a representation of such information, how to keep it updated, and how to use it for providing customization. Ideally, student modelling should consider all the aspects of students' behaviour and knowledge which present relation with their performance and learning [Wen87]. Nonetheless, in practice student modelling depends on the application [SMGS05].

The main approaches for constructing a student model are summarized in a work performed by Chrysafiadi and Virvou [CV13]: (i) the overlay model which represent the student's knowledge lever; (ii) the stereotype model that classifies students into groups according to their frequent characteristics; (iii) the perturbation model which models the student's model and misconceptions; (iv) machine learning techniques for automated observations of students' actions and behaviours, and for automated induction; (v) cognitive theories that attempt to explain the human behaviour; (vi) fuzzy logic modelling techniques or Bayesian networks for dealing with the uncertainty of student diagnosis; and (vii) ontologies for reused student models. Each of above approaches can be used on its own or can be combined with one or more other approaches, building a hybrid student model, according to the system's needs and aim. Further details about the features of each model are offered in Section 3.2.

Independently of the approach, the information is usually gathered through a diagnosis process, which is about inferring what the student knows from a set of observable facts. This *diagnosis process*, or *assessment process*, tries to infer the student's cognitive state from his/her performance. Diagnosis provides detailed information about students' competencies which is used to guide the selection of the next teaching operation. We found two major approaches regarding the diagnosis process. On the one hand, a number of approaches use navigational features of the students (number of times that a resource has been consulted, number of clicks on determined links, time spent in each web page, ...) in order to perform the diagnosis process. In web spaces, we think that this might not be the best strategy. One of the strengths of ITSs relies on their availability without temporal and spatial constraints. Without a direct supervision of a teacher, the system cannot guarantee that the student is not distracted in a concrete session. Therefore a diagnosis based on navigational features could infer incorrect student's knowledge states. On the other hand, most of approaches perform the diagnosis process through a set of quizzes or problems (in problem-solving scenarios) manually created by a tutor and solved by students. The results of such *diagnosis resources* are used to infer the current student's knowledge state. Although the ITS is able to select the diagnosis resources which better fit the current student model, they are predefined by the teacher who must consider all the possible students' cognitive states, being this a time-consumption and error-prone task. Hence, a method which dynamically generates diagnosis resources in function of the current student model is arguably necessary.

The above consideration has encouraged us to develop an ITS supported by automatic tests generation. The tests serve as diagnosis resources as well as teaching operations. On the one hand, the tests will be fitted for any student state, guaranteeing the correctness of the subsequent diagnosis process. Also teachers are released from the tedious task of manually creating each test. On the other hand, students can acquire the knowledge of the course following an e-assessment strategy. They will be able to see in which concepts of the course they are presenting more difficulties, explore the learning resources associated to each concept, and test again their knowledge in order to reach the learning objectives. Our method is based on our previous Fuzzy Test Generation Framework (FTGF) (see Section 2). From a item bank, our framework was able to create tests in function of assessment objectives specified by the user. Now the system itself will automatically define the assessment objectives identifying them from the student model.

In addition, in ITSs there is no direct interaction between the teacher and the students, and therefore the presence of uncertainty in student diagnosis is increased [GKPM02]. For this reason, we set as additional goal of our proposal the handling of uncertainty during the diagnosis process. Attending to literature, two major approaches have been employed to deal with uncertainty: Bayesian Networks (BNs) [CGV02, MM01, Liu06, BC03, SGA08, GSIM11] and Fuzzy Logic (FL) [XWS02, CV12, TGCK03, KDB03, Kav04, JJG12]. In this sense, FL presents advantages with respect to the others alternatives in several facets [HDJG94]. First, FL models reduce reasonably computational complexity. Second, it is considered easy for designers and users to understand and/or to modify. Third, FL can provide human-like descriptions of knowledge and imitate a human style of reasoning with vague concepts. In addition, the Bayesian approach requires the determination of probabilities from experts judgements, whilst fuzzy logic provides a convenient method to elicit the necessary knowledge from domain experts, thus expert teachers in case of student modelling, to implement the system [SMGS05]. It is easier and more reliable to extract knowledge form experts in linguistic form rather than in numbers representing this knowledge since experts feels most comfortable giving the original linguistic data [Kos91]. In this regard, FL techniques can be used to improve the performance of an educational environment. According to Shakouri and Menhaj [SGTN12], an algorithm based on fuzzy decision making helps to select the optimum model considering a set of criteria and model specifications. In this line, Chrysafiadi and Virvou [CV12] stated that the integration of FL into the student model can increase students' satisfaction and performance, improving the system's adaptivity and help the system to make more valid and reliable decisions. Therefore we will extend our student model by using FL techniques.

Summarizing, in this part of the dissertation we present a TTutor, a Test-based Intelligent Tutoring System. The knowledge representation of the domain is defined following an hybrid approach mixing the overlay model with ontological features. That is, the knowledge representation contains a set of concepts and a set of relations between concepts. Then teaching operations of the curriculum sequence will be generated in basis of the concepts. Since the knowledge of any application domain could be gathered through this representation, the system is easily applicable within any course. Next, the student model contains the level of knowledge that students have considering each concept. To that end, the system implements an e-assessment engine that automatically generates tests in function of the current student's knowledge regarding the concepts to be learnt. Those tests serves as teaching operations and the results of such tests serve as diagnosis data for updating the student model. In addition, the system promotes self-assessment by adding game-like features. The curriculum, sequence is divided into game-like stages. The system shows prompts to the student when he/she has passed one stage. Then the student can select a new stage or stay in the same one in order to completely master the current concepts. Finally, although student modelling has been widely studied by the scientific community, few efforts have been performed for opening the model to students. According to Jeremić [JJG12], opening the model to students might help them to better understand their learning and therefore enhance their learning processes. Knowing specific information about their concrete learning states may encourage them to reflect on their knowledge and on the learning process. Thence we have designed an intuitive user interface where the knowledge state of each concept belonging to stage is shown. The concepts are individually represented next next to a progress bar. Additionally, each concept is linked to a set of related learning resources. Thus students can review the concepts they want before performing the next teaching operation. The student model interface is also accessible for teachers who are able to control the performance of the course at any moment.

The rest of this section is organized as follows. Subsection 3.2 offers a discussion on most common student modelling approaches as key component of ITSs. We provide the overview of our proposed ITS in Subsection 3.3. A pre-experimental study carried out in order to validate the system will be commented in Subsection 3.4. Finally, Subsection 3.5 presents the conclusions and future work.

3.2 Related works

In this section, we summarize the main approaches for student modelling. We have considering both AES and ITSs, since the student modelling approach is shareable between this two fields. We base on the review performed by Chrysafiadi and Virvou [CV13] to that end.

3.2.1 Overlay Model

The overlay model was invented by Stansfield et al. [SCG76]. The main idea behind this model relies on the fact that a student may have incomplete but correct knowledge of domain. Therefore the student model is a subset of the domain model, which reflects the expert-level knowledge of the subject. The differences between the student and the domain models are believed to be the student's lack of skills and knowledge, and the instructional objective is to eliminate these differences as much as possible. Consequently, the domain is decomposed into a set of elements and the overlay model is simply a set of masteries over those elements. This model requires that the domain model represents individual topics and concepts. So, the overlay model can represent the user knowledge for each concept independently and this is the reason for its extensive use.

Nevertheless, the overlay model does not allow representing neither the incorrect knowledge that the student acquired or might have acquired. As Rivers [Riv89] pointed out, overlay models are inadequate for sophisticated models because they do not take into account the way users make inferences, how they integrate new knowledge with knowledge they already have or how their own representational structures change with learning. That is the reason why overlay model is usually combined with other student modelling approaches like stereotypes, perturbation and fuzzy techniques.

We provide some examples of overlay based student models in the following. First, Kassim et al. [KKR04] proposed a web-based intelligent learning environment for digital systems (WILEDS). Their system employs the overlay model next to differential and perturbation models for student modelling. Next, Carmona and Conejo [CC04] presented the student model employed by MEDEA, an open system to develop ITSs. Here the student model has two sub-models: the student attitude model, where the static information about the user is stored; and the student knowledge model, where the user's knowledge and performance is stored. The knowledge model is based on the overlay approach with four layers: estimated, assessed, inferred by prerequisite and inferred by granularity. The third and fourth layers are updated by Bayesian inference. Finally, Lu et al. [LWW+05] presented a model for simulating procedural knowledge in the problem solving process with the ontological system, InfoMap. Their method uses an overlay student model in combination with a buggy model for identification of the deficient knowledge.

3.2.2 Stereotype Model

The stereotype model classifies students into groups according to their frequent characteristics. Stereotypes were firstly introduced to user modelling by Rich [Ric79]. The main idea behind stereotypes is to cluster all possible students into several groups according to certain characteristics that they are typically shared. Such groups are called stereotypes. More specifically, a stereotype normally contains the common knowledge about a group of students. A new student will be assigned into a related stereotype if some of his/her characteristics match the ones contained in the stereotype.

The main advantage of this modelling approach is that the knowledge about a particular student will be inferred from the related stereotype(s) as much as possible, without explicitly going through the knowledge elicitation process with each individual student and the information about student stereotypes can be maintained with low redundancy [ZH05]. However, the stereotype approach presents several disadvantages. First, the technique is inflexible, since stereotypes are constructed in a hand-crafted way before real students have interacted with the system [TV02]. Second, in order to use stereotypes, the set of students must be divisible into classes; however, such classes may not exist []. Third, even if it is possible to identify such classes, the manager of the system in charge of building the stereotypes. Regardless the technological level of the manager, this is a process that is both time-consuming and error-prone [Kas91].

Stereotypes have been used in many adaptive or intelligent tutoring systems usually combined with other student modelling methods. For example, Tsiriga and Virvou [TV02] presented Web-PTV, a system focused on teaching the domain of the passive voice of the English language. The student model designed is based on stereotypes and machine learning. Carmona et al. [CCM08] designed a stereotype approach able to classify students in four dimensions according to their learning styles, in combination with Bayesian networks. As final goal, they designed a system able to select the more adequate learning objects for each student. Additionally, the stereotype approach has been used in used in Wayang Outpost, a tutoring system that helps students learn to solve standardized-test type of questions [AMW10].

3.2.3 Perturbation Model

The perturbation student model is an extension of the overlay model. It represents the student's knowledge as including possible misconceptions as well as a subset of the domain knowledge. This extension allows for better solution of student mistakes. The perturbation student model is useful for diagnostic reasoning. According to Martins et al. [MFDCC08], the perturbation student model is obtained by replacing the correct rules with the wrong rules, which when applied they lead to the answers of the student. Since there can be several reasons for a student wrong answer (several wrong values that lead to the student answer) the system proceeds to generate discriminating problems and presents them to the student to know exactly the wrong rules that this student has. The collection of mistakes is usually called bug library and can be built either by empirical analysis of mistakes (enumeration) or by generating mistakes from a set of common misconceptions (generative technique).

Some tutoring systems have included a perturbation student model for reasoning the student's behaviour. For example, Surjono and Maltby [SM03] presented an adaptive system including a perturbation student model. This model is used for performing better corrections of student mistakes.

Faraco et al. [FRG04] proposed a system based on the perturbation model. The misconceptions are represented by the exercises alternatives that are relative to each problem but are incorrect. The closer the weight of the alternative is to -1.0, the more it indicates a student lack of knowledge if he or she chooses this particular alternative as the exercise response. Therefore the system is able to provide personalized feedback and support to the distant students in real time. Next, Baschera and Gross [BG10] presented an inference algorithm for perturbation models based on Poisson regression. The algorithm is designed to handle unclassified input with multiple errors described by independent wrong-rules. The inference algorithm had been employed in a student model for spelling with a detailed set of letter and phoneme based wrong-rules. The model allows for appropriate correction actions to adapt to students' needs.

3.2.4 Machine Learning Techniques

Student modelling involves the inference of the student's behaviour taking into account her/his knowledge level, her/his cognitive abilities, her/his preferences, her/his skills, her/his aptitudes e.t.c. The processes of observation of student's action and behaviour in tutoring system, and of induction, should be made automated by the system. A solution for this is machine learning (ML), which is concerned with the formation of models from observations and has been extensively studied for automated induction [Web98]. Observations of the user's behaviour can provide training examples that a ML system can use to induce a model designed to predict future actions [WPB01].

Examples of ML-based approaches are provided as follows. Tsiriga and Virvou [TV03] used a combination of stereotypes with the machine learning technique of the distance weighted k-nearest neighbor algorithm, in order to initialize the model of a new student. The student is first assigned to a stereotype category concerning her/his knowledge level and then the system initializes all aspects of the student model using the distance weighted k-nearest neighbor algorithm among the students that belong to the same stereotype category with the new student. Wang et al. [WYW09] employed support vector machines in order to provide personalized learning resource recommendation. Later, Al-Hmouz et al. [AHSYAH11] combined two machine learning techniques in order to model the student and all possible contexts related to her/his current situation in an extensible way so that they can be used for personalization.

3.2.5 Cognitive Theories

Some researchers [Sal90, WB00] point out that technology is effective when developers thoughtfully consider the merit and limitations of a particular application while employing effective pedagogical practices to achieve a specific objective. A cognitive theory attempts to explain human behaviour during the learning process by understanding human's processes of thinking and understanding.

Cognitive theories have been used as support for student modelling. First, the Human Plausible Reasoning (HPR) theory [CM89] is a domain-independent theory originally based on a corpus of people's answers to everyday questions, which categorizes plausible inferences in terms of a set of frequently recurring inference patterns and a set of transformations on those patterns [BC88]. A system implementing HPR theory can be found in [VK02]. Second, the Multiple Attribute Decision Making (MADM) involves making preference decisions (such as evaluation, prioritization, and selection) over the available alternatives that are characterized by multiple, usually conflicting attributes [TH11]. An example can be found in [KV04]. Third, the OCC cognitive theory of emotions [Ort90] allows modelling possible emotional states of students. Systems presented in [CZ02, HSAF10] follow this scheme. Finally, the Control-Value theory [PFGP07] is an integrative framework that employs diverse factors, e.g. cognitive, motivational and psychological, to determine the existence of achievement emotions. A system implementing this framework is detailed in [MMKL$^+$10].

3.2.6 Fuzzy Student Model

As it was previously commented, FL has been used to deal with uncertainty in student modelling. Here we discuss most of such attempts.

Xu et al. [XWS02] proposed a prototype of a multi-agent based student profiling system. The system stores the learning activities and interaction history of each individual student into the student profile database. Such profiling data is abstracted into a student model by means of FL. Based on the student model and the content model, dynamic learning plans for individual students are made. Therefore the system is able to provide personalized learning materials, personalized quiz, and personalized advices. This scheme is similar to our approach, since it provides personalized tests. However, authors do not provide details about how test are constructed. This suggests that the test are manually predefined. Tsaganou et al. [TGCK03] presented a Fuzzy-Case Based Reasoning Diagnosis system of Historical Text Comprehension. The system encourages the student to read the historical text and answer to question-pairs selecting from the alternative answers. The student's responses define his observable behaviour. Then the system solves the diagnostic problem in two stages: the Fuzzy inference stage, which infers the arguments' completeness, and the Fuzzy-CBR inference stage, which infers the students cognitive profile and profile descriptor. The model is domain dependant and hardly extendible to other courses. Next, Kosba et al. [KDB03] developed the TADV (Teacher ADVisor) framework, which builds student, group, and class models based on tracking information and uses these models for generating advice to the instructors. The reasoning is based on the assumption that the students' actions provide the main source for inferring roughly about students' knowledge and misconceptions. For example, if a student is found to struggle with a specific concept and the student model indicates that this student has visited little course material about the concept, the instructor will be informed about the problem and will be advised to encourage the student to read the material related to this concept. Kavcic [Kav04] proposed a system able to construct an individualized navigation graph for each student and thus suggests the learning objectives the student is most prepared to attain. The system use fuzzy set theory for dealing with uncertainty in the assessment of students: the marks of assessment tests are transformed into linguistic terms, which are assigned to linguistic variables. Fuzzy IF-THEN rules are applied to obtain the appropriate categories of competences in the navigation graph. The method is rather similar than ours, but again no specification about the assessment tests is provided. We improve this model by automatically generating assessment tests in real time. Later, Salim and Haron [SH06] developed framework for individualizing the learning material structure in adaptive learning system. It aims to utilize the learning characteristics and provide a personalized learning environment that exploit pedagogical model and fuzzy logic techniques. The learning material is predefined and consists of 4 structures; theory, examples, exercises, and activities. The pedagogical model and learning characteristics are based on the student's personality factor, whilst the fuzzy logic techniques are used to classify the structure of learning material which is based on student's personality factors. Chrysafiadi and Virvou [CV10] described the student modelling component of a web-based educational application that teaches the programming language Pascal using fuzzy logic techniques. FL techniques are used to describe student's knowledge level and cognitive abilities. Furthermore, they use a mechanism of rules over the fuzzy sets, which is triggered after any change of the students' knowledge level of a domain concept, and update the students' knowledge level of all related concepts. We perform a similar process in order to update the level of knowledge of the concepts presenting relationship. Finally, Jeremić [JJG12] presented an intelligent tutoring system for learning software design patterns. The student model is a result of combining stereotypes and overlay modelling. The model is domain independent and can be easily applied in other learning domains as well. Having the aim of keeping the student model updated during the learning process, the ITS makes use of a knowledge assessment method based on fuzzy rules (i.e., a combination of production rules and fuzzy logics). Personalization is then provided as well as navigation through the course material by link removal.

3.2.7 Bayesian Networks for Student Modelling

Bayesian networks have been used to relate student's knowledge with student's observable behaviour in a probabilistic form. Bayesian models depend on how accurately they represent the probabilistic dependencies in the domain [CGV02]. A Bayesian network (BN) is a directed acyclic graph in which nodes represent variables and arcs represent probabilistic dependence or causal relationships among variables [Pea88]. The causal information encoded in BN facilitates the analysis of action sequences, observations, consequences and expected utility [Pea96]. In student modelling nodes of a BN can represent the different components/dimensions of a student such as knowledge, misconceptions, emotions, learning styles, motivation, goals, etc. Mayo and Mitrovic [MM01] performed a classification of Bayesian student modelling approaches into three types, according to how the structure of the network and prior, conditional probabilities are elicited: expert-centric models, which use experts to specify the structure of the network and its corresponding initial prior and conditional probabilities; efficiency-centric models that restrict the structure of the network in order to maximize efficiency; and data-centric models, which use data from previous experiment and/or pre-tests to generate the network and its probabilities.

Bayesian networks have attracted a lot of attention due to their sound mathematical foundations and also for a natural way of representing uncertainty using probabilities [Jam95, Liu06]. BNs present high representative power and an intuitive graphical representation. Furthermore, the presence of capable and robust Bayesian libraries (e.g. SMILE), which can be easily integrated into the existing or new student modelling applications, facilitates the adoption of BNs in student modelling [MPDLC02].

Several adaptive and intelligent tutoring systems have employed BNs in order to support the student modelling process. Here we present some examples of them. Conati et al. [CGV02] used BNs in order to devise probabilistic student models for Andes, a tutoring system for Newtonian physics whose philosophy is to maximize student initiative and freedom during the pedagogical interaction. Also, Bunt and Conati [BC03] employed BNs to model the students of an intelligent exploratory learning environment for the domain of mathematics functions, which was named Adaptive Coach for Exploration (ACE). They built a student model capable of detecting when the learner is having difficulty exploring and of providing the types of diagnosis that the environment needs to guide and improve the learner's exploration of the available material. Later, Conati and Maclaren [CM09] developed a probabilistic model of user affect, which recognizes a variety of user emotions by combining information on both the causes and effects of emotional reactions within a Dynamic Bayesian Network.

3.2.8 Ontology-based Student Model

Due to the fact that ontologies are becoming standard for web based knowledge representation, a lot of recent research has been done on the crossroad of student modelling and web ontologies. The benefits of ontology-based student modelling approaches come from the proper strengths of this knowledge representation: ontology supports the representation of abstract machine-readable and machine-understandable knowledge so as to be easily reused and, if necessary, extended in different application contexts. In addition, ontologies defines a format compatible with popular logical inference engines [WBG05].

Several student models have been built based on ontologies. For example, Dolog et al. [DHNS04] presented the Personal Reader, an experimental environment supporting personalized learning based on semantic web technologies. The prototype implements several methods needed for personalization suitable for an environment based on a fixed set of documents (a closed corpus) plus personalized context sensitive information from the semantic web. The prototype provides annotating and recommending learning material suitable for specific courses. To implement the retrieval of appropriate learning resources from the semantic web, authors proposed heuristics and query rewriting rules which allow us to reformulate queries to provide personalized information even when metadata quality is low. As part of the recommendation, the prototype offers quizzes in function of the concepts that student needs to review. However, the quizzes are again predefined. Pramitasari et al. [PHA$^+$09] developed a student model ontology based on student performance as representation of prior knowledge and learning style, in order to create personalization for e-learning system. Finally, Cheung et al. [CWC10] presented an Ontology-based Framework for Personalized Adaptive Learning (OPAL). It was developed for students to learn basic Java programming with adaptive features according to their learning preference and learning performance. In particular, a three-level Java course was developed to enable course content level promotion and demotion to help the student to learn the subject in different ways.

3.3 System overview

In this section, we present the overview of our proposed TTutor, a multi-domain Test-based Intelligent Tutoring System. Our system provides student with learning and assessment resources adjusted to his/her background knowledge and performance in a learning subject. The system includes functionality to create course structures as well as issues for each course. Each course will contain a set of learning resources and a test item bank. These elements should be created in a preliminary step, and therefore we do not consider here how such elements are introduced in the system. Then, TTutor is able to manage as many learning subjects as the teacher decides to create.

A *learning subject* defines an independent learning experience, with their own learning goals. Therefore students can select one or more learning subjects from the list of available subjects. Each learning subject contains a set of learning stages that students must master in order to reach the learning goals. Each stage will be composed by a set of interrelated concepts. Students have accessible information about their level of knowledge regarding the concepts in their current stage. Concepts present a set of learning resources that students can consult. When a student is ready to prove his/her knowledge, a e-assessment module generates a test in function of his/her previous background knowledge and performance. The results of the test are used to infer the new student's knowledge state. Once the learning goals of the current stage are reached with a minimum grade, the student can select the subsequent stage or stay in the current stage in order to completely master the corresponding concepts.

The architecture of the system (Figure iv.10) follows the classical four-component design scheme, including the domain model, the student model, the tutoring model and the interface. The four components are detailed below. It should be pointed out that the architecture is referred to individual learning subjects, since each one defines an independent learning experience.

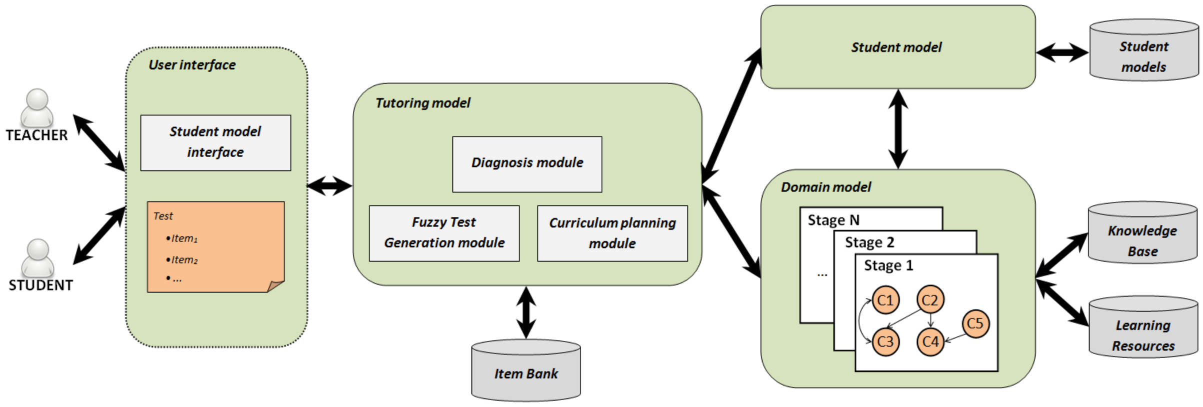


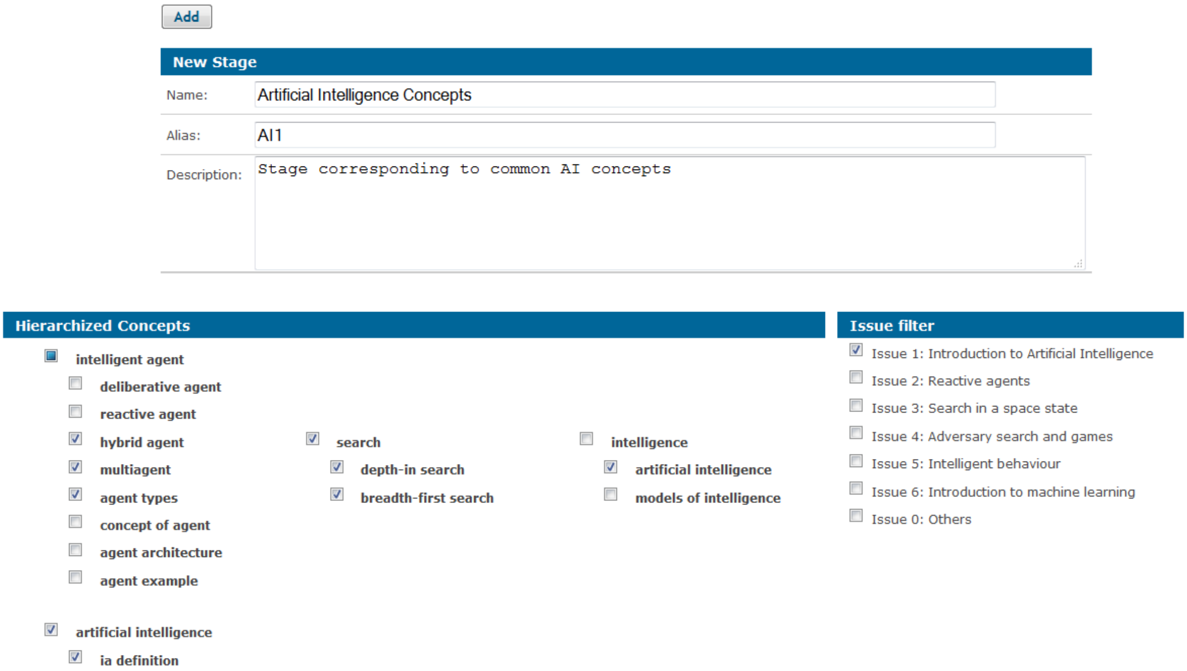
Figure iv.10: TTutor architecture

3.3.1 Domain model

The domain model contains the piece of the learning domain that is relevant for the learning experience. It encompasses an educational knowledge base including concepts, relations and learning resources. Fist, as part of the knowledge base, a lightweight educational-domain ontology contains all the concepts belonging to the course, and relationships between concepts. Among the possible type of relationships considered in a ontology, taxonomic relations (subclass, superclass) and educational-based relations (subtopic, and content-based relation) will be employed in along the ITS functionality. We shall remember that we discussed about these types of relationships and a intelligent approach in charge of detecting them from the learning resources in Chapter II Section 4. Second, the knowledge base contains a set of learning resources that are linked to the concepts which are related with. In addition, each resource belongs to one or more of the issues of the course structure. Two kind of learning resources are considered: assessment resources, i.e. test items; and textual learning resources, e.g. Wikipedia web pages, question-answer pairs, ...Finally, each item in the test item bank contains a set of properties including "type of item", "item difficulty", and "relevance to a course issue". We commented the definition of the test item properties in Section 2.3.

Considering the elements of the knowledge base, we have decided to choose an overlay model extended with ontological features in order to design the domain model. First, the domain model is composed by a set of stages, following a game-like scheme. The teacher is in charge of defining the number of stages and the content of each one (Figure iv.11). The concepts of the ontology which have sub-concepts or subtopics (in the ontology) are displayed hierarchically, meanwhile the rest of concepts of the ontology are displayed separately. The interface also includes the issues of the course structure as filters. Issue filters are optional and will be employed by the curriculum planning module.

Summarizing, the domain model is composed by a set of stage that represents the layers in the overlay model. Each stage is compound by a set of concepts interrelated by means of ontological relationships. Each stage may also contain a set of optional filters related to the course's issues.



Add

New Stage

Name: Artificial Intelligence Concepts

Alias: AI1

Description: Stage corresponding to common AI concepts

Hierarchized Concepts

- ■ intelligent agent
  - □ deliberative agent
  - □ reactive agent
  - ☑ hybrid agent
  - ☑ multiagent
  - ☑ agent types
  - □ concept of agent
  - □ agent architecture
  - □ agent example
- ☑ search
  - ☑ depth-in search
  - ☑ breadth-first search
- □ intelligence
  - ☑ artificial intelligence
  - □ models of intelligence
- ☑ artificial intelligence
  - ☑ ia definition

Issue filter

- ☑ Issue 1: Introduction to Artificial Intelligence
- □ Issue 2: Reactive agents
- □ Issue 3: Search in a space state
- □ Issue 4: Adversary search and games
- □ Issue 5: Intelligent behaviour
- □ Issue 6: Introduction to machine learning
- □ Issue 0: Others

Figure iv.11: Learning stage definition interface

3.3.2 Student model

The student model is the core component of our system, since it contains the student information needed to provide personalization of the learning experience. A student model might contain any kind of information referred to students' characteristics. The election of characteristics to be considered depends on the system requirements. Therefore, we have considered two types of student characteristics (Figure iv.12): personal data and assessment data.

Firstly, *personal data* collects static characteristics of the student, including name, id, username and password, email and phone number. Such information is employed by the system in order to manage the different student accounts, and by the teachers in case that they need to establish a direct communication with the student. Nonetheless, these characteristics are not directly related with the curriculum planning adaptation.

Secondly, *assessment data* encompass the cognitive and individual student's characteristics grouped into four subtypes, each one related with elements of the domain model:

- **Learning Subject data**: it is referred to the student's current level of knowledge of a concrete learning subject. This attribute can take one of the following values: "NOT LEARNED",

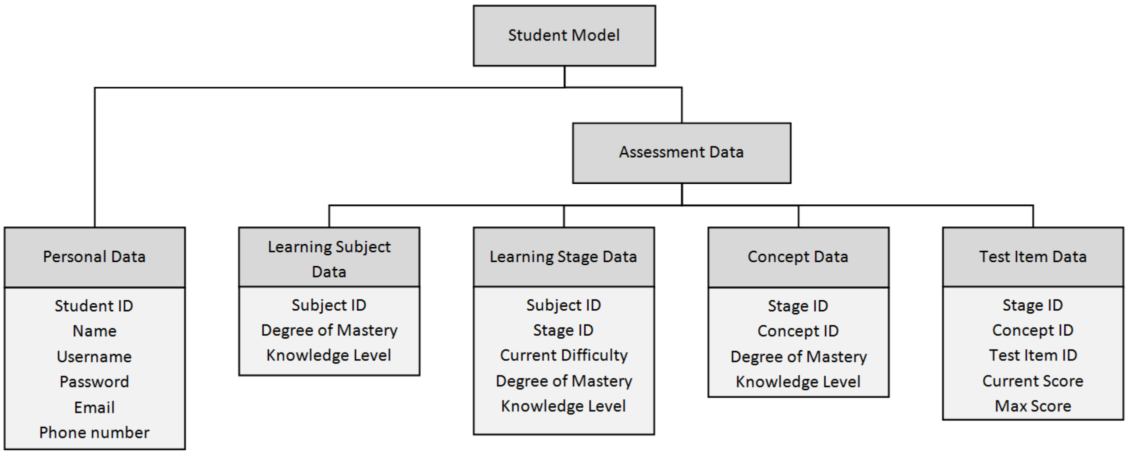


Figure iv.12: Student model attributes

"LEARNED" and "MASTERED". In addition, the degree of mastery attribute represents the level of knowledge by means of a scalar float ranged in the interval [0,100]. The degree of mastery is inferred trough the diagnosis process after the student performs a test. Then, in function of the obtained degree of mastery, the system will set a value for the level of knowledge attribute.

- **Learning Stage data**: it is referred to the student's current level of knowledge and the degree of mastery of a concrete learning stage. Therefore the system will collect these data for all the stages in a learning subject. Both the degree of mastery and the level of knowledge are inferred in the diagnosis process after the student performs a test. These attributes present the same structure commented above. In addition, the level of difficulty for each stage, $LSD \in \{Easy, Medium, Difficult\}$, is also stored here. When a student starts a stage, the level of difficulty is setted to the minimum. The difficulty automatically increases as the student learns concepts.

- **Concept data**: it is referred to the student's current level of knowledge and the degree of mastery of a concrete concept belonging to a learning stage. A concept can be contained in more than one learning stage, hence different data would be collected in that case. Both the degree of mastery and the level of knowledge are inferred in the diagnosis process after the student performs a test. These attributes present the same structure commented above.

- **Test Item data**: it represents the last score obtained by a student for a performed test item as well as the maximum score that the student could have obtained. That is, test items present a maximum reachable score. The score obtained by the student could be the maximum, if he/she correctly responds the item; or it could be decreased or even be negative, if the student gives a partially-correct solution or a wrong solution. Since a test item could be related to one or more concepts, the test item data is grouped within concrete concepts. For example, if a test item i is related to the concept "neural network" and a test is generated in order to evaluate such concept, the data will be stored considering the pair (neural network,i), although i would be also related with "machine learning" concept. This data is the atomic component of the diagnosis process. The degree of mastery of concept, stages and learning subjects are calculated in function of it. The complete diagnosis process will be explained in the subsection below.

3.3.3 Tutoring Model: Curriculum planning

The tutoring model simulates the behaviour of a human tutor in order to provide personalized learning plans to each student. The goal of this model is two-fold. On the one hand, it is in charge of selecting the next teaching operation of the curriculum planning. On the other hand, it performs the diagnosis process after a teaching operation is performed in order to infer the new student's knowledge state. We detail the first functionality in the following.

The curriculum planning represents the complex process of selecting appropriate teaching operations that best fit the current student's characteristics in order to acquire one or more domain concepts from the learning stage. Our system uses tests as bricks to generate learning experiences. A curriculum is composed by a set of concepts, and by a test related to such concepts. If a student wants to perform a test, he/she must indicate the number of items. Then, the test items from the item bank, the related concepts and issue filters from the domain model, and the student's characteristics from the student model are received as input by the curriculum planning module. We define the curriculum planning process in the following.

As first step, the curriculum planning module selects which items from the item bank, IB, are suitable for making up the current test. Let us formally define firstly the environment. For every test item $i_k \in IB$, there exists a mapping function $C(i_k) = \{c_i, \ldots, c_j\}$ that obtains the concepts which are related with i_k. In addition, given a concept c'_k, a mapping function $I(c'_k) = \{i'_i, \ldots, i'_j\}$ obtains the items interrelated with c'_k. Remember that each item in the item bank presents a set of variables defining the degree of relatedness within the course issues. That is, given a issue l of the course, the variable $CI^l(i_k) \in [0,1]$ defines the degree of relatedness of i_k within l. Next, there exists a set of relations between any concept c'_k and the rest of concepts on the domain model. Then $r_{k'l}$ defines a relation between the concept c'_k and the concept c_l as the triple $r_{k'l} = (c'_k, c_l, tr) | tr \in \{r_{supc}, r_{subc}, r_{st}, r_{cb}\}$, where tr defines the type of relation, and r_{supc}, r_{st}, r_{sb}, r_{cb} refers to superclass relation, subtopic relation, subclass relation, and content-based relation respectively. Finally, each concept of the current stage presents a level of knowledge in the student model. Therefore, a function $SM(C) \in \{NOT\ LEARNED, LEARNED, MASTERED\}$ defines the level of knowledge of a concept c in the current stage.

During the selection of the candidate items, the curriculum planning module performs an iterative search looking for the items interrelated with the concepts that satisfy two prerequisite conditions. First, given a concept c_n and a item $i_m \in I(c_n) = \{i_i, \ldots, i_j\}, i \leq m \leq j$, if c_n has subtopics and/or subclasses that are also contained in $C(i_m)$, such subtopic and/or subclass concepts should be already learned in order to consider i_m as candidate. That concepts could be previously learned in the current stage or in previous stages of the learning subject. For example, the student would have to acquire the knowledge of "breath-first search" and "depth-in search" before testing his/her knowledge on "search" concept. Second, if the stage contains a set of issue filters (IF), i_m will be considered candidate if $\forall iss \in IF | CI^{iss}(i_m) > 0$, where iss is a issue of IF. The candidate item selection algorithm is detailed in Table IV.3.

As second stage, the curriculum planning module obtains the test using to that end the FTGF. First, each item in CI is provided with a number of votes. The votes, v_i of a item i determine the necessity of including i in the test, regarding the level of knowledge of the concepts in $C(i)$ and the last score for i obtained by the student (Remember that such information is stored in the test item data of the student model). We note $SCORE(i)$ as the function of the student model that retrieves that last score obtained for i, and $MAX\_SCORE(i)$ as the function that retrieves the maximum score reachable for i. Then, each item i_m in CI is voted in function of the following heuristic steps:

- If $SCORE(i_m) < MAX\_SCORE(i_m)$ then $v_{i_m} = v_{i_m} + (\alpha_1 \cdot |NLC|)$, where NLC a the

```
INPUT:
    $SC$: set of concepts of the current stage.
    $IF$: set of issues included in the filter.
    $SM$: current student model.
OUTPUT:
    $CI$: Candidate test item set.
begin
    $CI \leftarrow \varnothing$
    foreach $concept_i \in SC$ do
        $related_i \leftarrow \varnothing$
        foreach $concept_j | \exists r_{ij} = (concept_i, concept_j, tr), tr \in \{r_{subc}, r_{st}\}$ do
            $related_i \leftarrow related_i \cup concept_j$
        end for
        foreach $item_m \in I(concept_i)$ do
            if $\forall concept_{aux} \in related_i \cap C(item_m) | SM(concept_{aux}) \in \{LEARNED, MASTERED\}$ then
                if $\forall issue_j \in IF | CI^{issue_j}(item_m) > 0$ then
                    $CI \leftarrow CI \cup \{item_m\}$
                end if
            end if
        end for
    end for
end
```

Table IV.3: Candidate Test Item selection algorithm

subset of $C(i_m)$ containing the concepts that have not been learned by the student. That is, $\forall c_k \in NLC|SM(c_k) =' NOT\ LEARNED'$.

- If $SCORE(i_m) < MAX\_SCORE(i_m)$ then $v_{i_m} = v_{i_m} + (\alpha_2 \cdot |LC|)$, where LC a the subset of $C(i_m)$ containing the concepts that have been learned by the student. That is, $\forall c_k \in LC|SM(c_k) \in \{LEARNED, MASTERED\}$.

- If $SCORE(i_m) = MAX\_SCORE(i_m)$ then $v_{i_m} = v_{i_m} - (\alpha_3 \cdot |NLC|)$, where NLC a the subset of $C(i_m)$ containing the concepts that have not been learned by the student. That is, $\forall c_k \in NLC|SM(c_k) =' NOT\ LEARNED'$.

- If $SCORE(i_m) = MAX\_SCORE(i_m)$ then $v_{i_m} = v_{i_m} - (\alpha_4 \cdot |LC|)$, where LC a the subset of $C(i_m)$ containing the concepts that have been learned by the student. That is, $\forall c_k \in LC|SM(c_k) \in \{LEARNED, MASTERED\}$.

The aim of this voting stage is to give higher probability for a item to be selected if the item was previously wrong answered and the related concepts have nor been learned. The values of the parameters α_1 to α_4 are defined by the teacher. The sum of votes (both positives and negatives) for each item is normalized in the range [0,1]. Then, the module obtains a ordered list of candidate items. Nevertheless, we apply a random component for avoiding successive test containing the same items. Table IV.4 shows the algorithm in charge of this task.

As final step, the list of candidate items is then used as input by the FTGF. The level of difficulty stored in the learning stage data of the student model, $LSD \in \{Easy, Medium, Difficult\}$, is used

```
INPUT:
    $CI$: set of candidate test items.
    $VCI$: set of votes assigned to candidate items.
OUTPUT:
    $CIL$: Ordered candidate test item list.
begin
    $CIL \leftarrow \varnothing$
    $L_{aux} \leftarrow randomOrder(CI)$
    while $L_{aux} \neq empty$ do
        $randomFloat \leftarrow obtainRandomFloat()$
        $booleanFlag \leftarrow false$
        foreach $item_i \in L_{aux}$ and $booleanFlag = false$ do
            if $v_{item_i} \geq randomFloat$ then
                $CIL \leftarrow CIL.concat(item_i)$
                $L_{aux} \leftarrow L_{aux}.remove(item_i)$
                $booleanFlag \leftarrow true$
            end if
        end for
        if $booleanFlag = true$ then
            $randomFloat \leftarrow obtainRandomFloat()$
        elsif
            $randomFloat \leftarrow randomFloat - 0.01$
        end if
    end while
end
```

Table IV.4: Candidate Test Item ordering algorithm

to define a Test Difficulty objective in the FTGF with the form “100% of LSD items”. Subsequently, the FTGF checks the acceptance thresholds (see Section 2.3) of the items in the list following the given order. As results, the curriculum planning module obtains a test representing a teaching operation fitted with the performance of the student in the current learning stage.

3.3.4 Tutoring Model: Diagnosis module

The second functionality included as part of the tutoring model involves the process of diagnosing the student’s level of knowledge after a teaching operation is carried out. Therefore the diagnosis module infers the current student’s level of knowledge of concepts, learning stage and learning subjects.

Level of Knowledge of concepts The student knowledge represents the student’s level of understanding of the concepts in a learning stage, and subsequently in a learning subject. The diagnosis process is in charge of inferring the level of knowledge of a student after performing a test. Since the nature of the diagnosis process is imprecise, we deal with this by means of fuzzy set techniques [Zad65, Men95]. We define the following three fuzzy sets for describing student “Level of knowledge” fuzzy variable:

- **Not Learned (NL)**: the degree of mastery in the concept is from 0% to 50%.

- **Learned (L)**: the degree of mastery in the concept is from 45% to 80%.
- **Mastered (M)**: the degree of mastery in the concept is from 70% to 100%.

The membership functions for the three fuzzy sets are shown in Figure iv.13, where the input is the student degree of mastery. The degree of mastery could be referred to a concept, to a stage or to learning subject, since the three of them share the same limited interval [0,100]. Thus, a triplet (μ_{NL},μ_L,μ_M) is used to express the level of knowledge of a concept, learning stage, or learning subject. For example, if the student obtains a degree of mastery of 75 regarding a given concept, then the level of knowledge is described by the triplet (0,0.5,0.5).

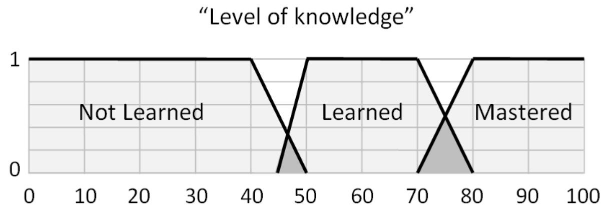


Figure iv.13: Fuzzy sets defining the level of knowledge in a concept

The degree of mastery of all the concepts of a learning stage is calculated after the student performs the test corresponding to the current teaching operation. Initially, the degree of mastery (DM, in the following) of a given concept c is obtained by means of the following function:

$$DM(c) = \frac{1}{min\{\ 1, \dfrac{a_c}{x}\}} \cdot \frac{\sum_{i=1}^{x} score(i_i)}{x} \cdot 100 \tag{IV.28}$$

where a_c denotes a parameter which indicates the minimum number of correctly answered items needed for considering a concept mastered, x represents the number of items linked to the c, and $i_i, i = 1, \dots, x$ represents the items linked to c. As it can be seen, a higher value for the parameter a_c will lead to a lower DM, and therefore more items must be correctly answered in order to reach the mastery of c.

Let us illustrate this process with an example. Let us assume that a concept c is linked to 10 items. A correct answer obtains 1 point, meanwhile an incorrect answer obtains -1 point. In addition, the teacher sets the parameter a_c to 5 items. Before performing the current teaching operation, the scores obtained by the student for items related to c were $SCORE(i_1) = 0$, $SCORE(i_2) = -1$, $SCORE(i_3) = 0$, $SCORE(i_4) = 1$, $SCORE(i_5) = 1$, $SCORE(i_6) = 0$, $SCORE(i_7) = -1$, $SCORE(i_8) = 0$, $SCORE(i_9) = 1$, and $SCORE(i_{10}) = 0$. Let suppose that the current generated test includes the items i_2 and i_8 that are correctly answered by the student. Therefore, the scores of such items change to $SCORE(i_2) = 1$, and $SCORE(i_8) = 1$. Then, the DM of c is calculated:

$$DM(c) = \frac{1}{min\{\ 1, \dfrac{5}{10}\}} \cdot \frac{\sum_{i=1}^{x} score(i_i)}{10} \cdot 100 = \frac{1}{0.5} \cdot \frac{4}{10} \cdot 100 = 80 \tag{IV.29}$$

Finally, the level of knowledge of c is described by the triplet (0,0,1).

A common characteristic of most of learning domain is that the assimilation of some concepts leads to better understanding of other interrelated concepts. For example, if a student has a DM of 0.6 in both "reactive agent" and "deliberative agent" concepts, and after a test he/she increases the DM of "reactive agent", it is rather possible that he/she possess a better understanding of "deliberative agent" although no items related with such concept have been included in the test. Therefore, the DM of "deliberative agent" should be increased. In contrast, if the DM of a concept is decreased and it is strongly interrelated with another concept, the DM of the second concept should be also decreased. We call this process as expansion of dependencies.

The *expansion of dependencies* process occurs between to interrelated concepts c_i and c_j, where c_i represents the concept which has been evaluated in the test, and c_j represents the concept strongly related with c_i which may have been included or not in the test. The idea behind this is that if the DM of c_i suffers a positive variation, then the DM of c_j should suffer a positive variation. In contrast, a negative variation of the DM of c_i should lead to a negative variation of the DM of c_j. We note $\Delta DM(c)$ as the improvement in the percentage of variation suffered by the DM of a concept. For example, if after a test, the DM of a concept c changes from 40 to 80, then $\Delta DM(c) = +40\%$. Furthermore, if the DM of c changes from 80 to 40, then $\Delta DM(c) = -40\%$. Having this into account, we have modelled the expansion of dependencies by means of fuzzy inference.

Fuzzy Inference (FI) is based on FL and has the aim of simulating the human reasoning in order to generate decisions from approximate and uncertain information. It consists of one more more fuzzy rules, a set of facts, and a conclusion [HS98, KY95]. First, fuzzy rules have the form of IF-THEN rules. The IF part corresponds to the antecedents, meanwhile the THEN part corresponds to the conclusion. The antecedent (the rule's premise) describes the conditions that have to be satisfied for the rule to be activated, while the conclusion (the rule's consequent) assigns a fuzzy set to the given input combination. Any rule is allowed to be satisfied, in contrast to crisp rules. We have used three fuzzy linguistic variables representing the antecedents needed to design our fuzzy rules, and another one representing the conclusion. Concretely, we have defined four fuzzy variables including the "Level of knowledge" variable commented above (Figure iv.13). The rest of variables are "Improvement in the percentage of variation of the degree of mastery of c_i", "Level of relation between c_i and c_j, given c_i" (antecedents), and "Variation of DM of c_j" (conclusion). Their membership functions are shown in Figure iv.14. The level of relation of between concepts is stored in the domain model. Remember that such relation is concerned with the learning resources linked to each concept (II.14) (Chapter II Section 4.3). Regarding the "Variation of DM of c_j" variable, the labels are the following: {Strongly decreases (STD), Decreases (D), Slightly Decreases (SLD), Remains (R), Slightly Increases (SLI), Increases (I), Strongly Increases (STI)}.

Using the commented fuzzy variables, we have defined nine fuzzy production rules (Table IV.5). We classify the rules in function of the following facts. First, R_0 is a default rule. This rule serves as regulator of the fuzzy inference process, as it will be commented below. Second, R_1 to R_4 consider the fact that if the DM of c_i increases, then the DM of c_j increases. Third, R_5 to R_8 consider the fact that if the DM of c_i decreases, then the DM of c_j decreases. The logic of the expansion of dependencies follows the four following processes.

First, during the *fuzzification* process the crisp input values are fuzzified into the linguistic values of the fuzzy variables. Second, during the *aggregation* process, the values of the rules' premises are computed. Each condition in the antecedents is assigned with a degree of truth based on the degree of membership of the corresponding linguistic value. The degree of truth of the antecedents is computed as the product of the degree of truth of the conditions. This degree of support for the rule is assigned to the degree of truth of the conclusion. Third, during the *composition* process, the degree of truth of the output values are combined using the sum of the degrees of truth of the rules with the same linguistic terms in their conclusion. Finally, the *defuzzification* process transforms the linguistic values of the output variable into a crisp value. To that end, the Centre of Maximum (CoM) method is employed, where the crisp value is calculated as the best compromise for the most typical values of each linguistic value and their respective degrees of membership. The most typical

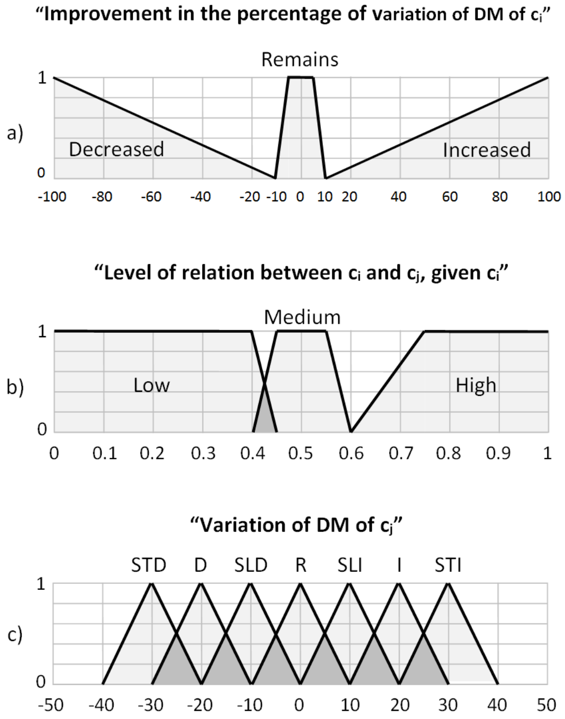


Figure iv.14: Fuzzy variables: (a) Improvement in the percentage of variation of the DM of c_i, (b) Level of relation between c_i and c_j, given c_i, and (c) Variation of DM of c_j

value is the maximum of the respective membership function.

Let illustrate this process with an example. Let consider that the system contains five concepts, c_1 to c_5. The relations between the concepts are depicted in Table IV.6. After a test, including the concepts c_1, c_2, and c_3, the student obtained the following degrees of mastery for the concepts: $DM(c_1) = 65(\Delta DM(c_1) = +30\%)$, $DM(c_2) = 43(\Delta DM(c_2) = +50\%)$, $DM(c_3) = 15(\Delta DM(c_3) = -70\%)$, $DM(c_4) = 85$, and $DM(c_5) = 75$. Then, the following processes are realised.

The first fact ($\Delta DM(c_1) = +30\%$) occurs and the rules R_0 and R_2 are fired, considering c_2 as c_j. The degree of truth (DT) of the antecedent of R_0 is 1, meanwhile the DT of the antecedents of R_2 are computed as the product of the corresponding degrees of membership: $\mu_{Learned}(65) = 1$, $\mu_{NotLearned}(43) = 0.7$, $\mu_{High}(0.65) = 0.33$, and $\mu_{Increased}(+30) = 0.22$. Therefore, DT of R_2 is 0.05. Then, the most typical values of the conclusions ($\Delta DM(c_2)$ Remains, and $\Delta DM(c_2)$ Increases) are 0, and 20, respectively. Therefore, the crisp value regarding the resulting variation of the DM of c_2 is computed as:

$$\Delta DM(c_2) = \frac{1 \cdot 0 + 0.05 \cdot 20}{1 + 0.05} = +0.95\% \tag{IV.30}$$

The second fact ($\Delta DM(c_2) = +50\%$) only fires the default rule, and thence no calculation is carried out.

Finally, the last fact occurs ($\Delta DM(c_3) = -70\%$) and the rules R_0, R_6, and R_8 are fired, considering c_5. The DT of R_1 is 1. The DT of R_6 is 0.33, computed as the product of: $\mu_{NotLearned}(15) = 1$, $\mu_{Learned}(75) = 0.5$, $\mu_{High}(1) = 1$, and $\mu_{Decreased}(-70) = 0.66$. Next, the DT of R_8 is also 0.33, computed as the product of: $\mu_{NotLearned}(15) = 1$, $\mu_{Mastered}(75) = 0.5$, $\mu_{High}(1) = 1$, and

Identifier	Rule antecedents	Rule conclusion
R_0	IF ∅	THEN $\Delta DM(c_j)$ **Remains**
R_1	IF LK(c_i) is *Not Learned* AND LK(c_j) is *Not Learned* AND Rel(c_i/c_j) is *High* AND $\Delta DM(c_i)$ is *Increased*	THEN $\Delta DM(c_j)$ **Slightly Increases**
R_2	IF LK(c_i) is *Learned* AND LK(c_j) is *Not Learned* AND Rel(c_i/c_j) is *High* AND $\Delta DM(c_i)$ is *Increased*	THEN $\Delta DM(c_j)$ **Increases**
R_3	IF LK(c_i) is *Mastered* AND LK(c_j) is *Not Learned* AND Rel(c_i/c_j) is *High* AND $\Delta DM(c_i)$ is *Increased*	THEN $\Delta DM(c_j)$ **Strongly Increases**
R_4	IF LK(c_i) is *Mastered* AND LK(c_j) is *Learned* AND Rel(c_i/c_j) is *High* AND $\Delta DM(c_i)$ is *Increased*	THEN $\Delta DM(c_j)$ **Slightly Increases**
R_5	IF LK(c_i) is equal to LK(c_j) AND Rel(c_i/c_j) is *High* AND $\Delta DM(c_i)$ is *Decreased*	THEN $\Delta DM(c_j)$ **Slightly Decreases**
R_6	IF LK(c_i) is *Not Learned* AND LK(c_j) is *Learned* AND Rel(c_i/c_j) is *High* AND $\Delta DM(c_i)$ is *Decreased*	THEN $\Delta DM(c_j)$ **Decreases**
R_7	IF LK(c_i) is *Learned* AND LK(c_j) is *Mastered* AND Rel(c_i/c_j) is *High* AND $\Delta DM(c_i)$ is *Decreased*	THEN $\Delta DM(c_j)$ **Decreases**
R_8	IF LK(c_i) is *Not Learned* AND LK(c_j) is *Mastered* AND Rel(c_i/c_j) is *High* AND $\Delta DM(c_i)$ is *Decreased*	THEN $\Delta DM(c_j)$ **Strongly Decreases**

Table IV.5: Fuzzy rules for the expansion of dependencies process. Abbreviations: (LK) Level of Knowledge, (Rel(c_i/c_j)) Level of relation of c_i and c_j, given c_i, ($\Delta DM(c_i)$) Improvement in the percentage of variation of the degree of mastery of c_i, ($\Delta DM(c_j)$) Resulting Variation of DM of c_j

$\mu_{Decreased}(-70) = 0.66$. Then, the most typical values of the conclusions ($\Delta DM(c_5)$ Remains, $\Delta DM(c_5)$ Decreases, and $\Delta DM(c_5)$ Strongly Decreases) are 0, -20, and -30, respectively. Therefore, the crisp value regarding the resulting variation of the DM of c_5 is computed as:

$$\Delta DM(c_5) = \frac{1 \cdot 0 + 0.33 \cdot (-20) + 0.33 \cdot (-30)}{1 + 0.33 + 0.33} = -9.93\% \qquad \text{(IV.31)}$$

Relation	c_1	c_2	c_3	c_4	c_5
c_1	0	0.65	0	0.9	0
c_2	0.7	0	0	0.7	0
c_3	0	0	0	0	1
c_4	0.7	0.8	0	0	0
c_5	0	0	1	0	0

Table IV.6: Relation existent between the concept of the example

Attending to the example, we can observe that the default rule performs a regulatory function. The absence of that rule would lead to a excessively aggressive variation of c_j which does not corresponds with a real life scenario. For example, during the first fact of the example, the variation of c_2 would be of $+20\%$ if the default rule would not be considered. The variation is too high, considering that with thee similar facts affecting c_2, such concept would reach the "Learned" level of knowledge even if the concept would not have been included in any test.

Summarizing, the level of knowledge of a concept depends on two main factors: the previous DM of the concept, and the DM obtained after the performance of a test and the subsequent expansion of dependencies process. These two values must be integrated in order to obtain a coherent DM. In this sense, we have designed the following function that represents the final DM of a given concept c after the whole diagnosis process at concept level:

$$DM_{final}(c) = f(DM_i(c), DM_t(c)) = \begin{cases} \dfrac{\beta_1 \cdot DM_t(c) + \beta_2 \cdot DM_i(c)}{\beta_1 + \beta_2}, & if\ DM_t(c) \geq DM_i(c) \\ \dfrac{\gamma_1 \cdot DM_t(c) + \gamma_2 \cdot DM_i(c)}{\gamma_1 + \gamma_2}, & if\ DM_t(c) < DM_i(c) \end{cases} \tag{IV.32}$$

, where the parameters β_1, β_2, γ_1 and γ_2 establishes the degree of modification of the initial DM, after considering the information obtained through the test. These parameters are manually setted by the teacher. After the final degrees of mastery of each concept in the learning stage are calculated, these crisp values are used as input of the fuzzy variable "level of knowledge" and a triplet is associated to each concept.

Level of Knowledge of Learning Stages Once the levels of knowledge of the concepts in the learning stage have been calculated, the diagnosis module proceeds to obtain the level of knowledge of the whole stage. As it occurred before, the learning stage is assigned with a degree of mastery, calculated as the mean value of the DM of all the concepts $(c_1, \ldots, c_n)$ in the learning stage:

$$DM(s) = \frac{\sum_{i=1}^{n} DM(c_i)}{n} \tag{IV.33}$$

The level of knowledge of the stage is represented as the triplet obtained by means of the fuzzy variable "level of knowledge". The DM of the current learning stage is also employed for modifying the current difficulty of the stage. Initially, the difficulty is setted to "Easy". Once the student reaches a DM of the current stage higher than 45 and lower than 75, the difficulty is automatically setted to "Medium". Finally, if the student reaches a DM equal or higher than 75, the difficulty is automatically setted to "Difficult". Remember that the difficulty of the learning stage was employed for the curriculum planing module in order to generate the tests.

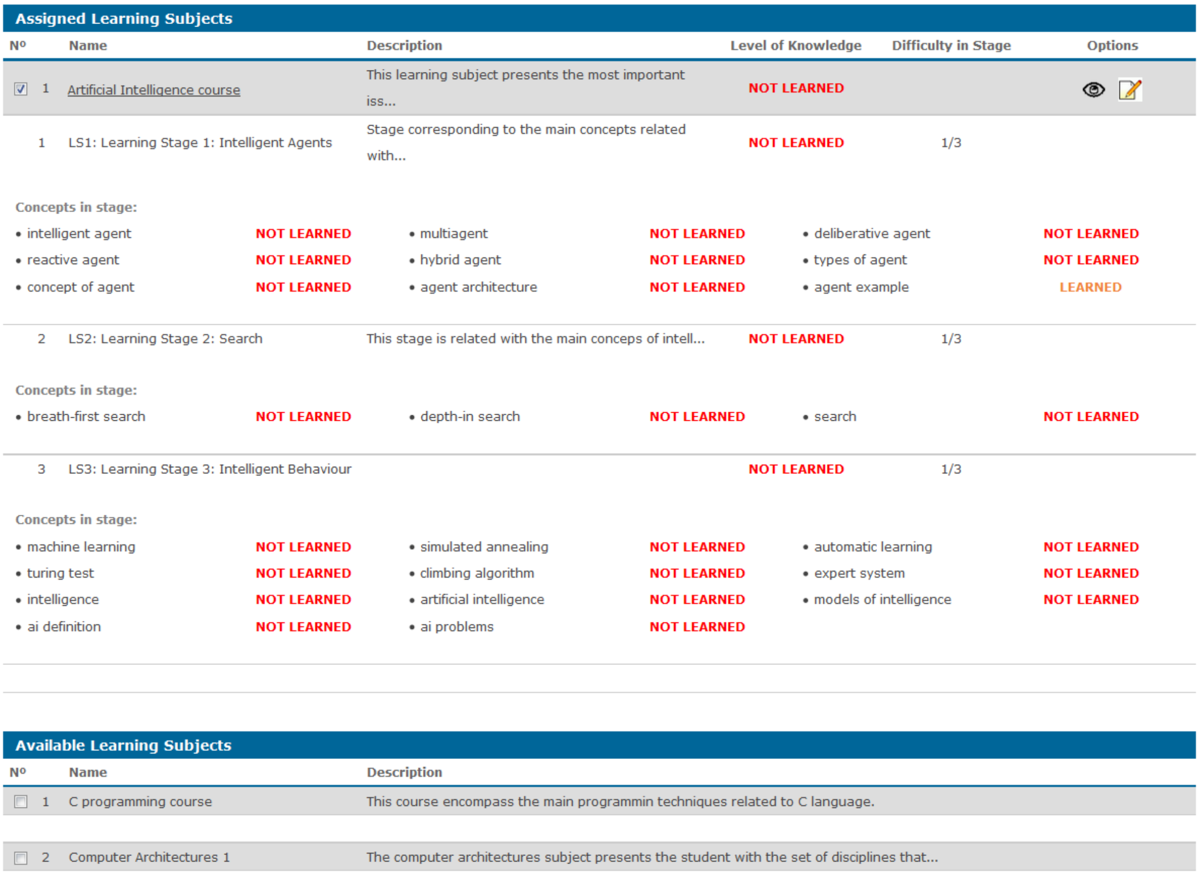
Level of Knowledge of Learning Subjects The final step relies on the calculation of the level of knowledge of the whole learning subject. The subject is assigned with a degree of mastery, calculated as the mean value of the DM of all the learning stage $(s_1, \ldots, s_m)$ in the learning subject:

$$DM_{subject} = \frac{\sum_{i=1}^{m} DM(s_i)}{m} \tag{IV.34}$$

3.3.5 User Interface

Last but not least, the user interface is also a main component in TTutor. It is in charge of: (a) offering the available learning subject to students, (b) presenting the tests (teaching operations) to students, (c) opening the student model to both student and teacher, and (d) offering the learning resources which are associated to the concepts of each learning stage. In order to clarify the functionality of the user interface, we will discuss both the student and the teachers perspectives independently.

Student Perspective When a student creates an account into TTutor, he/she can show the list of available learning subjects in the system (Figure iv.15). Student simply has to mark the check box associated to one available learning subject in order to be enrolled in it. The same index interface serves as summary of the current student's knowledge state of the learning subjects in which he/she is enrolled. To that end, the student only need to click in the name of the learning subject, and subsequently a table summarizing that information is displayed. Initially, the level of knowledge of concepts, stages and learning subjects is "Not Learned". As the user is performing teaching operations, the interface changes to show the labels corresponding to the higher value of the triplet that represents the level of knowledge on each concept, stage, and subject. For example, if the level of knowledge of the "intelligent agent" concept is (0,0.8,0.2), the interface will show the label "Learned". In case that the values for two labels were the same, the interface would show the label corresponding to the lower level of knowledge. From the index interface, the student can perform two operations. First, he can directly perform a test. By clicking in the corresponding button, the system runs the test generation protocol, after asking the user for selecting the number of items that he/she desires to include in the test. Second, the student can navigate to the student model interface where the main information of the student model is displayed.



Assigned Learning Subjects

Nº	Name	Description	Level of Knowledge	Difficulty in Stage	Options
☑ 1	Artificial Intelligence course	This learning subject presents the most important iss...	NOT LEARNED		👁 ✎
1	LS1: Learning Stage 1: Intelligent Agents	Stage corresponding to the main concepts related with...	NOT LEARNED	1/3	

Concepts in stage:

• intelligent agent	NOT LEARNED	• multiagent	NOT LEARNED	• deliberative agent	NOT LEARNED
• reactive agent	NOT LEARNED	• hybrid agent	NOT LEARNED	• types of agent	NOT LEARNED
• concept of agent	NOT LEARNED	• agent architecture	NOT LEARNED	• agent example	LEARNED

2	LS2: Learning Stage 2: Search	This stage is related with the main conceps of intell...	NOT LEARNED	1/3

Concepts in stage:

• breath-first search	NOT LEARNED	• depth-in search	NOT LEARNED	• search	NOT LEARNED

3	LS3: Learning Stage 3: Intelligent Behaviour		NOT LEARNED	1/3

Concepts in stage:

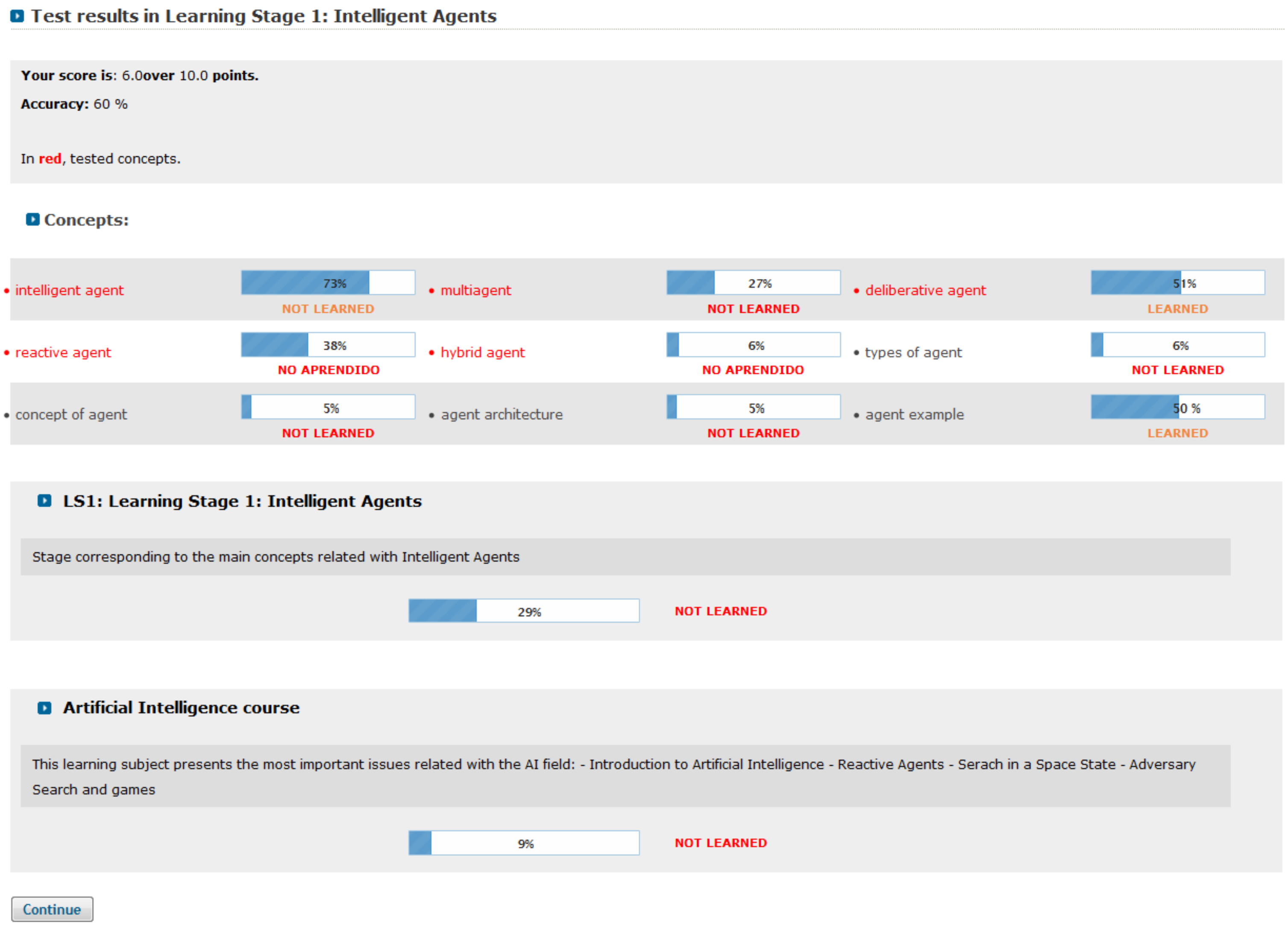
• machine learning	NOT LEARNED	• simulated annealing	NOT LEARNED	• automatic learning	NOT LEARNED
• turing test	NOT LEARNED	• climbing algorithm	NOT LEARNED	• expert system	NOT LEARNED
• intelligence	NOT LEARNED	• artificial intelligence	NOT LEARNED	• models of intelligence	NOT LEARNED
• ai definition	NOT LEARNED	• ai problems	NOT LEARNED		

Available Learning Subjects

Nº	Name	Description
□ 1	C programming course	This course encompass the main programmin techniques related to C language.
□ 2	Computer Architectures 1	The computer architectures subject presents the student with the set of disciplines that...

Figure iv.15: Learning subject index interface

The test result interface is in charge of showing the student's progress in the current learning stage, after a test is performed (Figure iv.16). The concepts of the stage which have been included in the test are highlighted in red. Then, the interface dynamically updates the progress bars corresponding to those concepts, including a programmed delay in order to make it easier for students to focus their attention on the changing of the bars. In addition, the bars corresponding to the learning stage and the learning subject are also dynamically updated. The values of the progress bars correspond to the degree of mastery of each element. If the stage has been accomplished (the value for the "Learned" label in the level of knowledge triplet is 1), the interface displays a prompt congratulating the student, and suggesting that he/she is able to access the next stage. From here, the student can access the student model interface by clicking the "Continue" button.



Test results in Learning Stage 1: Intelligent Agents

Your score is: 6.0**over** 10.0 **points.**

Accuracy: 60 %

In **red**, tested concepts.

Concepts:

• intelligent agent	73% NOT LEARNED	• multiagent	27% NOT LEARNED	• deliberative agent	51% LEARNED
• reactive agent	38% NO APRENDIDO	• hybrid agent	6% NO APRENDIDO	• types of agent	6% NOT LEARNED
• concept of agent	5% NOT LEARNED	• agent architecture	5% NOT LEARNED	• agent example	50 % LEARNED

LS1: Learning Stage 1: Intelligent Agents

Stage corresponding to the main concepts related with Intelligent Agents

29% NOT LEARNED

Artificial Intelligence course

This learning subject presents the most important issues related with the AI field: - Introduction to Artificial Intelligence - Reactive Agents - Serach in a Space State - Adversary Search and games

9% NOT LEARNED

Figure iv.16: Test results interface

The student model interface displays the complete information of the student model regarding the student's current knowledge state, offers a bridge to the learning resources associated to each concept in the domain model, and allows the student to make game-like decisions about his/her learning experience. Figure iv.17 depicts the student model interface. The interface includes a navigational map (Fig. iv.17A) showing the learning stages and their current state by means of colours codes. The student can visit previous stages in order to completely master the corresponding concepts. Next, the interface includes the description of the learning subject, the current level of knowledge considering the whole subject, links to the testing process, and to the next stage if the current has been learned (Fig. iv.17B). The information related to the level of knowledge of each concept, and stage is also available. Finally, the concepts' name in the interface acts as links to the learning resources associated to each one (Fig. iv.17C). As preliminary design, we have linked the TTutor interface with the concept map visualization framework commented in Chapter III Section 3.3. Therefore, the student is able to explore the concept-level conceptual map, examining the relations existing between the focused concept and the rest of concepts in the domain model, as well as to access the learning resources associated to the focused concept.

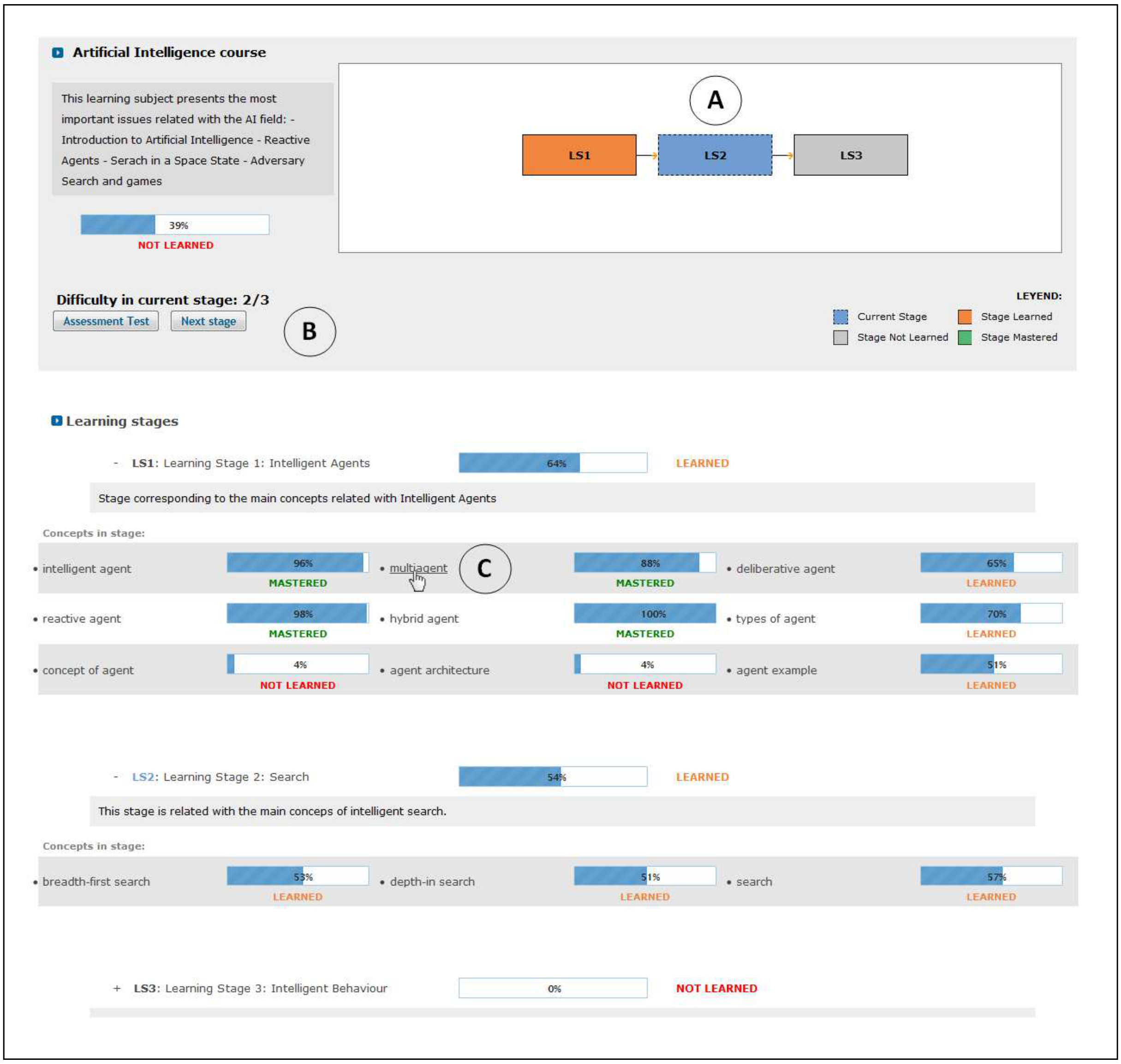
Teacher Perspective The user interface for teachers has the main objectives: to show a simple summary of the students' knowledge states, and to provide extended information in case that the teacher requires it. It should be pointed out that the learning subjects and the learning stages are created also through the user interface, but that is not the scope regarding the exploration of the different student models.

First, all the students enrolled in each available learning subject are organized into tables in the summary interface (Figure iv.18). The level of knowledge of the whole learning subject, the current stage, the current stage difficulty and the level of knowledge of the current stage are shown. This interface allows the teacher to acquire a rapid scheme about the performance of him/her students. Besides, the complete information of each student in each learning subject is accessible. By clicking the "eye" icon, the student model interface is displayed. Such interface is exactly the same that the student interface, but eliminating the buttons corresponding to the e-assessment process and the navigation through stages.

3.4 Experiences with the System

The adaptation developed by our TTutor system has the aim of improving the students' learning experiences. Since the adaptivity scheme employed by our system is complex, it should be evaluated the effectiveness of the adaptation yielded by our system. The most common methods for the evaluation of an ITS are empirical approaches [CV12]. These approaches require from measuring the students' performance by conducting experiments with a experimental group and a control group in a real-life scenario. For example, the Kirkpatrick's model [Kir96], which is one of the most well-known and used models for measuring the effectiveness of training programs, requires form at least a two year evaluation period.

Unfortunately, the development of the TTutor system has been realised during the last period of the complete research work gathered in this Thesis. Due to we do not have had the possibility of performing long term empirical experiments, we provide a pre-experimental evaluation of our ITS attending to the opinions that a group of volunteer students had about TTutor. The participants were students of the 'Artificial Intelligence" 2013/14 course in the Computer Engineering degree in the University of Granada. 10 students (2 were women, and 8 men) took part in this pre-experimental study.



Artificial Intelligence course

This learning subject presents the most important issues related with the AI field: - Introduction to Artificial Intelligence - Reactive Agents - Serach in a Space State - Adversary Search and games

39%
NOT LEARNED

LEYEND:

▨	Current Stage	🟧	Stage Learned
⬜	Stage Not Learned	🟩	Stage Mastered

Difficulty in current stage: 2/3

Assessment Test | Next stage

Learning stages

- **LS1**: Learning Stage 1: Intelligent Agents — 64% — LEARNED

Stage corresponding to the main concepts related with Intelligent Agents

Concepts in stage:

• intelligent agent	96% MASTERED	• multiagent	88% MASTERED	• deliberative agent	65% LEARNED
• reactive agent	98% MASTERED	• hybrid agent	100% MASTERED	• types of agent	70% LEARNED
• concept of agent	4% NOT LEARNED	• agent architecture	4% NOT LEARNED	• agent example	51% LEARNED

- **LS2**: Learning Stage 2: Search — 54% — LEARNED

This stage is related with the main conceps of intelligent search.

Concepts in stage:

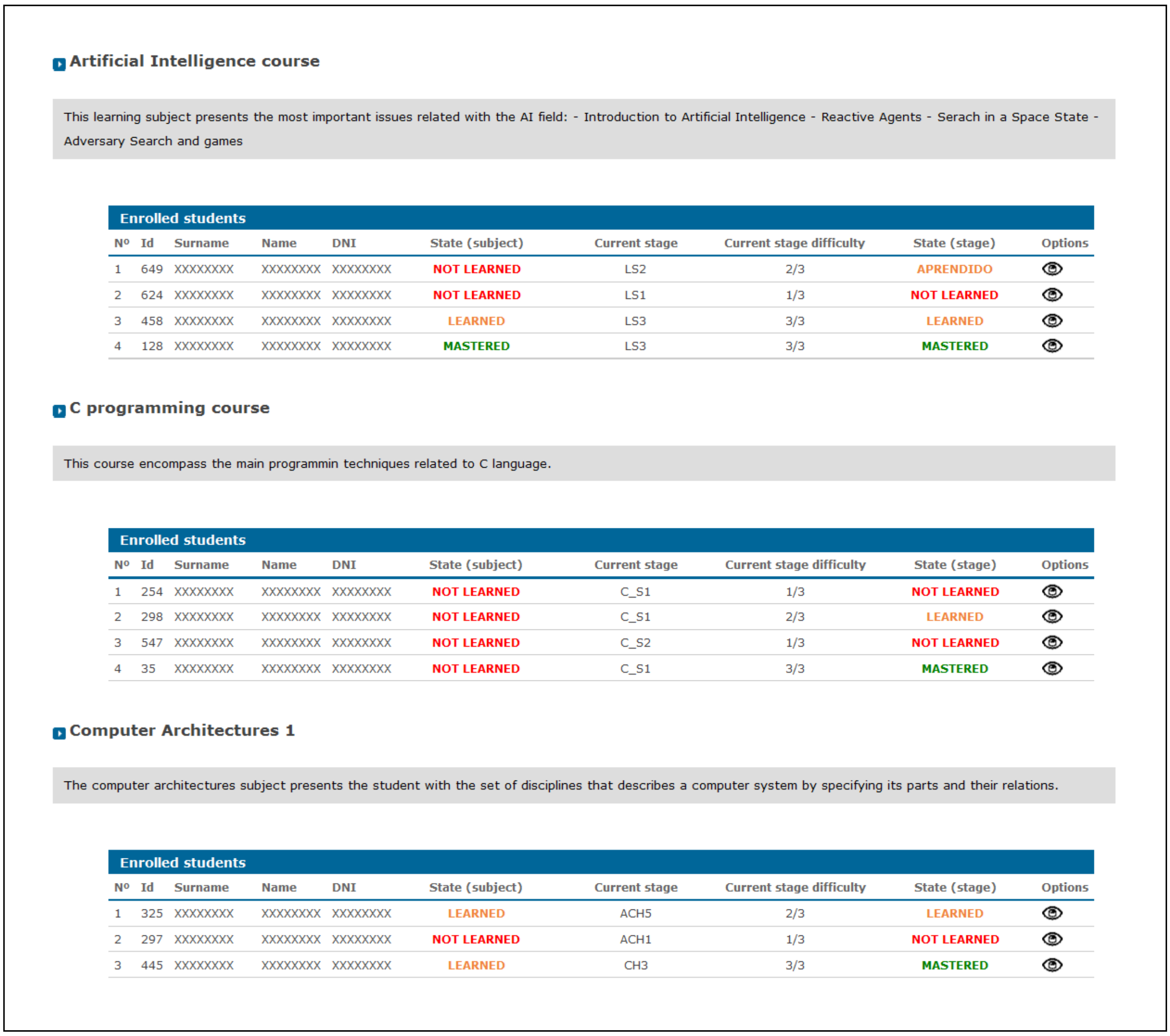
• breadth-first search	53% LEARNED	• depth-in search	51% LEARNED	• search	57% LEARNED

+ **LS3**: Learning Stage 3: Intelligent Behaviour — 0% — NOT LEARNED

Figure iv.17: Student model interface

The TTutor was stored in a university server under Apache Tomcat 6 (6.0.32) to manage the data storage. It was accessible via web application during a in-person two hour session. The participants were initially instructed with a specific briefing lesson of 1 hour. After that, they were able to use the system which included 3 different learning subjects. Subsequently, in order to analyse the students' reactions they completed a questionnaire including 11 items based on a Likert scale with five responses ranging from "very much" (5) to "not at all" (1). We employed a similar questionnaire to that designed by Chrysafiadi and Virvou in [CV12]. They used the questionnaire to test a domain dependant ITS for learning programming, so we extend to any learning domain. The questions are divided into two groups: questions evaluating the effectiveness of the system, and questions evaluating the adaptivity of the system. Both the questions, and the mean and standard deviation values of the students' responses are summarized in Table IV.7. The first block of question corresponds to effectiveness, meanwhile the second block corresponds to adaptivity.

The gathered reactions showed that the students were satisfied with our TTutor system. The



Artificial Intelligence course

This learning subject presents the most important issues related with the AI field: - Introduction to Artificial Intelligence - Reactive Agents - Serach in a Space State - Adversary Search and games

Enrolled students

Nº	Id	Surname	Name	DNI	State (subject)	Current stage	Current stage difficulty	State (stage)	Options
1	649	XXXXXXXX	XXXXXXXX	XXXXXXXX	NOT LEARNED	LS2	2/3	APRENDIDO	👁
2	624	XXXXXXXX	XXXXXXXX	XXXXXXXX	NOT LEARNED	LS1	1/3	NOT LEARNED	👁
3	458	XXXXXXXX	XXXXXXXX	XXXXXXXX	LEARNED	LS3	3/3	LEARNED	👁
4	128	XXXXXXXX	XXXXXXXX	XXXXXXXX	MASTERED	LS3	3/3	MASTERED	👁

C programming course

This course encompass the main programmin techniques related to C language.

Enrolled students

Nº	Id	Surname	Name	DNI	State (subject)	Current stage	Current stage difficulty	State (stage)	Options
1	254	XXXXXXXX	XXXXXXXX	XXXXXXXX	NOT LEARNED	C_S1	1/3	NOT LEARNED	👁
2	298	XXXXXXXX	XXXXXXXX	XXXXXXXX	NOT LEARNED	C_S1	2/3	LEARNED	👁
3	547	XXXXXXXX	XXXXXXXX	XXXXXXXX	NOT LEARNED	C_S2	1/3	NOT LEARNED	👁
4	35	XXXXXXXX	XXXXXXXX	XXXXXXXX	NOT LEARNED	C_S1	3/3	MASTERED	👁

Computer Architectures 1

The computer architectures subject presents the student with the set of disciplines that describes a computer system by specifying its parts and their relations.

Enrolled students

Nº	Id	Surname	Name	DNI	State (subject)	Current stage	Current stage difficulty	State (stage)	Options
1	325	XXXXXXXX	XXXXXXXX	XXXXXXXX	LEARNED	ACH5	2/3	LEARNED	👁
2	297	XXXXXXXX	XXXXXXXX	XXXXXXXX	NOT LEARNED	ACH1	1/3	NOT LEARNED	👁
3	445	XXXXXXXX	XXXXXXXX	XXXXXXXX	LEARNED	CH3	3/3	MASTERED	👁

Figure iv.18: Teacher interface showing a summary of the current state of each student in each learning subject

items were scored with high valuations in general. The item with highest score was the 9th, regarding the adaptivity of the test provided by the ITS. In contrast, the lowest rated item was the 7th, regarding the adaptivity of the system to the educational need. This suggests that other adaptive functionalities, suck navigation adaptation, would be desirable in order to make a improved ITS. Nevertheless, the pre-experimental results were promising.

3.5 Conclusions and Future Work

In this part of the Thesis we have presented an intelligent tutoring system based on e-assessment. Our system arises in the absence of proposals that handle the automatic generation of diagnosis resources as part of the diagnosis process of the student model. Common approaches perform the diagnosis process through a set of tests manually created by a teacher. The diagnosis process has the goal of inferring the student's cognitive state during a learning experience in order to provide adaptation. Therefore, the presence of accurate diagnosis tests is mandatory. The task of manually

Item	Students (N = 10)	
	M	*SD*
1. Does the educational software meet your expectations?	3.4	0.84
2. Does the educational software help you understanding the learning domain?	3.8	0.78
3. Do you think that this educational software is useful as an educational “tool”?	4	0.81
4. Do you think that the use of this educational software is a waste of time?	1.2	0.42
5. After the end of the educational process, do you feel that you have assimilated all the subjects that you are taught?	3.7	0.67
6. Does the program correspond to your knowledge level each time?	3.8	0.63
7. Does the program correspond to your educational needs each time?	3.3	1.05
8. How time do you spend on issues that you already known?	1.7	0.82
9. Do the tests adapt to your educational needs?	4.1	0.73
10. Do you think that each time you go to a next stage, you have known adequately all the concepts of the previous stages?	3.7	0.82
11. Does your return to a previous stage help you to better understand the learning domain?	3.9	0.87

Table IV.7: Items and results of the questionnaire

creating a sufficient set of tests which should fit every student's cognitive state is time-consuming and error prone. Our system is designed in order to accomplish that task automatically. To that end, we have taken advantage of our previous Fuzzy Test Generation Framework (Section 2), which is able to create tests in function of assessment objectives identified from the student model. Our strategy not only solves the problem of manually defining the diagnosis resources, but also allows students to acquire the knowledge of the course following an e-assessment strategy. They will be able to see in which concepts of the course they are presenting more difficulties, explore the learning resources associated to each concept, and test again their knowledge in order to reach the learning objectives.

Since the student modelling process requires the handling of uncertainty, we have designed our diagnosis process using fuzzy set techniques following the tendency of the literature. Although many others approaches has been employed in student modelling, it has be proved that the integration of FL into the student model can increase students' satisfaction and performance, improving the system's adaptivity and help the system to make more valid and reliable decisions. What is more, the linguistic nature of the fuzzy logic makes the extraction of knowledge easier and more reliable than a numeric representation (for example, Bayesian Networks).

Our system employs a hybrid approach mixing the overlay model with ontological features for defining the domain model. The teaching operations of the curriculum sequence are generated in basis of the concepts of the domain model. This scheme allows our system to work with any domain and course. In addition, we have paid special attention on opening the student model for both students and teachers. The user interface clearly shows the current student's levels of knowledge of all the elements which compose the learning experience. Through the user interface, the student is able to access the learning resources from which acquire the corresponding knowledge before performing a test. Furthermore, the system promotes self-assessment by adding game-like features. The domain model is divided into game-like learning stages, and the user is responsible of deciding when he/she moves to the next stage. The level of knowledge is represented with game-like progress bars. Additionally, the teacher is able to explore the current knowledge state of each individual student.

We have provided a pre-experimental evaluation of our system, gathering to that end the opinions that students had after a introductory session. Although the results were promising, we have planned to perform a more solid evaluation of TTutor. We need to compare the performance of students in experimental groups in contrast to the performance of students in control groups. Hence, we will employ TTutor as additional resource of the "Artificial Intelligence" course in the University of Granada during the course 2014/15. The system will be available for students in a experimental group. During the course, student in experimental group will be able to use TTutor as addition of the face-to-face lectures. At the end of the course, the performance of students in control and experimental groups will be compared. Additionally, we have scheduled the validation of the system's navigation efficiency. It is referred to the reliability of the system when it decides that a concept has been learned. We will apply the evaluative methodology depicted in [CV12]. Finally, assuming that the results during the first years would be satisfactory, we plan to apply the TTutor during the 2015/16 year. By collecting the information yielded by the use of the system, we will be able to perform the Kirkpatrick's evaluation model [Kir96], guaranteeing in this way the benefits of TTutor.

4 Final Discussions and Future Work

In this chapter, we have provided a complete adaptive system for supporting the learning process. Adaptation in e-learning is receiving great attention, since it allows to handle the drawbacks inherited from general hypermedia systems and educational systems. On the one hand, the excess of information may lead to information overload, disorientation.... On the other hand, tradition educational systems presents lacks of contextual and adaptive support, lack of flexible support of the presentation and feedback, and lack of the collaborative support between students and systems. We have focused in the field of intelligent tutoring systems in order to adjust the educational environment in function of the current state of each student. In this kind of systems, the diagnosis is usually performed through a set of quizzes or problems generated by a tutor and solved by students. Although some systems automatically choose the diagnosis resources, it is rather common that they will be manually designed by a human tutor. This circumstance makes the matter delicate because the tutor should consider all possible students' cognitive states in order to generate fitted assessment resources.

For that main reason, we have developed a method for automatically providing diagnosis resources in order to test the student's level of knowledge about a learning subject. In order to implement the ITS, we have firstly designed an intelligent approach for the automatic generation of assessment tests. The method has been proposed in order to overcome the common limitations proper of traditional Computer-Based Test (CBT) systems, where all students are usually forces to answer the same set of items. Tests are not tailored for the individual needs of students and they provide very little information about student performance. We have taken the underlying ideas of Computerized Adaptive Tests (CAT) in order to implement our framework, but including self-assessment features. The framework is able to generate test suited to user requirements from large item banks. The framework was tested in experimental group obtaining excellent opinions from both students and the teacher. Hence, this guarantees that the correct operand of the framework as engine of the Intelligent Tutoring Systems proposed in the following section. Subsequently, and taking advantage of that test generation framework, we have presented an intelligent tutoring system based on e-assessment. Our strategy solves the problem of manually defining the diagnosis resources, and also allows students to acquire the knowledge of the course following an e-assessment strategy.

Unfortunately, we are not be able to provide a complete validation of our proposed ITS due to the absence of real-life information about the improvement of the performance of student using the system. Although we have carried out a pre-experimental evaluation of the system obtaining very promising results, we have planned to develop an extensive evaluation by including the system as addition of the face-to-face lectures of the "Artificial Intelligence" course in the computer engineering degree in the University of Granada.

Up to now, we have explained an extensive set of intelligent techniques, methods and systems that supports the learning processes in e-learning scenario. Therefore, the subsequent step consists of integrating all the proposals into a complete virtual learning environment. Thus, the next point to deal with in this dissertation concerns with the development of a complete architecture and a learning methodology that regulates the VLE use (Chapter V).

Chapter V

Integrating the Intelligent Modules: Influence Factors, Architecture, Methodology and Validation of a Virtual Learning Environment

1 Introduction

So far, we have presented a set of intelligent methods designed for supporting the learning process in e-learning environments. Hence, in this chapter we present a complete Virtual Learning Environment (VLE) which integrates the methods as intelligent modules. This will provide us a well-formed framework for validating the complete proposal as a whole, allowing us to test the implications of the methods here studied on real students' performance. Reasonably the application must combine additional features such as course management, learning resources management, user management, etc. Before facing this task, we will investigate common pedagogical dimensions which might benefit the e-learning experience. Further, we will be able to design the architecture of our VLE and to define usage regulatory methodology for the platform. Then, we will be ready to submit it to a high education course of real engineering.

The effectiveness of a VLE depends on several characteristics that must be first analysed. Attending to the social context in which e-learning is currently involved, we can find that traditional teaching methods are evolving towards the *learner-centered paradigm* that has become a key component for on-line distance education [MW97, LYL09]. In this sense, VLEs have been used to facilitate learner-centered instruction [YRK$^+$13, RWW$^+$08]. McCombs and Whisler defined "learner centered" as:

> "the perspective that couples a focus on individual learners — their heredity, experiences, perspectives, backgrounds, talents, interests, capacities, and needs — with a focus on leaning-the best available knowledge about learning and how it occurs and about teaching practices that are most effective in promoting the highest levels of motivation, learning, and achievement for all learners. This dual focus then informs and drives educational decision making" [MW97].

In a recent work, Yildirim et al. [YRK^{+}13] summarize its key features. From the student viewpoint, the knowledge should be constructed by them through gathering and synthesizing information in order to solve real-world problems, in contrast to the teacher-centered paradigm [FH00]. They should also be actively involved in the learning process as well as the teacher [FH00], having as much time as they need to achieve mastery [Sch90]. From the teacher viewpoint, they should be facilitators of the knowledge acquisition process by acting as guides, coaches, and motivators as students become more active in their learning process [MW97]. Furthermore, the teacher should give students some control over the learning processes in order to motivate them to work harder, with initiative and self-direction [Wei13]. Indeed, the education reform imposed by the Bologna process in Europe is focused on engage students to be independent students, promoting lifelong learning and student-centered learning among other priorities [Hei05, Lou01]. The benefits inherited from this educational shift are well-documented. For example, Miller reported that students in learner-centered on-line classrooms produced higher quality course projects and mastered concepts better than those in non-learner centered on-line classrooms [Mil08]. Additionally, Chou [Cho04] conducted a research study analysing the inclusion of leaner-centered on-line activities in the curriculum of an upper level undergraduate course. These activities were found to enhance interpersonal relationships and increase opportunities for students to share information and build knowledge while collaborating with others. Additionally, the research results showed that the incorporation of learner-centered instructional design and constructivist, and cooperative activities into distance education enhanced the learning process by promoting student interaction and active learning. Concluding, how to properly design a VLE in basis of the learner-centered paradigm became a requisite of the utmost importance.

Next, we need to investigate which factors make e-learning strategies effective and satisfactory. Most of studies relies on the following indicators that add value to and influence the effectiveness of learning in VLEs [CG09, GCBSMS09, HJ96, CD96, Bou95, Mis11]: student satisfaction, and student achievement. *Student satisfaction* reflects how positively students perceive their learning experiences. It is particularly important because it is rather probable that students abandon the course it they are not satisfied with the e-learning experience [ABF07]. *Student achievement* (or *student performance*) is referred to the quality of learning that a student reaches after the learning process, that is, student's perception of how much they learned from the course. It is usually concerned with students' grades (assignment, exams, quizzes, i.e. quantitative evaluation of learning) [LLM07, MC11]. Certain strategies improve the students' satisfaction and achievement in VLE. We will analyse the factors and pedagogical strategies involved in order to design an effective VLE and its corresponding usage regulatory methodology. However, it is not a goal for this dissertation to perform a deep analysis on each subject. It would require an extensive knowledge of the pedagogical and education field, which is not a feature of this work. Particularly, we will examine the following pedagogical dimensions: student self-regulation, student interaction (in regard to other students, the teacher and the content), self-assessment, electronic assessment, feedback mechanisms, and blended environment.

Finally, we will discuss the results in terms of effectiveness obtained after conducting two experimental studies based on the application of our proposal as an addition to the lectures of an undergraduate course in the Computer Engineering degree during two consecutive years.

The rest of this chapter continue as follows. Section 2 analyses the influence factors that makes a VLE a suited method for improving learning processes. Next, we comment the architecture of our VLE and the regulatory methodology in Section 3. Subsequently, we present our experimental studies about the use of the VLE in a real engineering course in Section 4. Finally we conclude with some final discussions in Section 5.

2 Factors Affecting Effectiveness in E-learning

As we commented in the introductory section, the effectiveness of e-learning platforms is usually measured through two main learning outcomes: student satisfaction, and student achievement. The objective of this part of the dissertation is to analyse the impact that a number of learning strategies produces in those indicators. To that end, we perform an extensive analysis of the state of the art related to the field. Although some different learning strategies present strong relation between each other, we present the strategies independently in order to clarify the reading. Subsequently, once the usefulness of each strategy is contrasted, we will be able to define the architecture and functionality of the pursued VLE. In this regard, we will conclude each one of the following subsections specifying a set of VLE design principles. Some of them will be directly related to the specific functions of the computer application while others will be related to scheduling policies interrelated with the usage methodology.

2.1 Student Self-regulation

The term *self-regulation* is referred to the use of self-regulatory learning behaviours by students in order to perform a self-regulated learning [GDH07]. Self-regulated learning meant that students must make an intentional effort to manage and direct complicated learning activities [Kau04]. Zimmerman defined self-regulated learning as the student's ability to independently and proactively engage in self-motivation and behavioural processes that increase gaol attainment [Zim00]. More specifically, self-regulated learning can be regarded as a skill where students must know how to set goals, what is needed to achieve those goals, and how to actually attain these goals [DK12].

The self-regulated learning can be understood as a three phase cyclic model, including a planning strategy, a monitoring strategy, and a regulating strategy [Pin99]. Planning strategy means goal setting, in which students set learning goals and make plans to achieve it. Monitoring strategy, the next step after planning, means a monitoring process by evaluating learning goals. This strategy presents strong relationship with self-assessment [Puz08, Wan11]. Students can evaluate their own performance and the results of self-assessment can serve as reference for self-regulation. Finally, regulating strategy means a regulating process based on the evaluation of the goals performed in the monitoring step. Hence students are able to modify their own learning behaviours as well as the learning goals. Self-regulated students engage a cyclic feedback loop until they successfully achieve their goals. Nonetheless, the students are not the only actors. Teachers have the responsibility of facilitating students to perform self-regulated learning by making them use self-regulated learning strategies in order to experience the benefits of self-regulated learning [ZBK96]. That is, if teachers exploit the proper strategies encouraging students to use self-regulatory learning behaviours, it is more likely that student's self-regulated learning ability will be improved and they will be motivated to perform self-regulated learning.

There are many studies showing that self-regulation is important in improving student learning effectiveness [GDH07, LCS08, Sch05, Pin03]. Attending to student satisfaction indicator, it is more likely that self-regulated students experience learning satisfaction than students with low self-regulation [Art08]. Womble [Wom07] found a positive correlation between self-regulation and student satisfaction. In addition, Lin et al. found that self-regulation significantly impacted on-line student satisfaction [LLL08]. Finally, some studies found that self-regulation is one of the most reliable predictors of student achievement [Sch91, Hod08, McG10]. McGhee found a significant, moderate, and positive relationship between self-regulation and academic achievement [McG10]. In the same line, Kaufman [Kau04] believed that a student with good learning achievement is often self-regulated.

The above considerations are sufficient to understand the importance of self-regulation learning strategies. Therefore, the design of our VLE will hold a self-regulation space. Thus, we prioritize the role of the student as main actor in most of learning process in our system. Students will be able to take part on the whole learning cycle, even in the creation of the educational content. In addition, the VLE's usage schedule should consider three different temporal stages, related to the three commented self-regulatory phases.

2.2 Student Interaction

The lack of interaction has been of the biggest concerns related to those e-learning platforms that focus uniquely in the content management. In fact, increasing researches suggest that interaction, whether it is between students, between student and teacher, and/or between student and the content, is one of the strongest predictors of e-learning positive outcomes [ABF07, Flo00, Swa02, MC11, VM99, Arb04]. In traditional teaching, students have no incentive to construct their own knowledge and little motivation to retain information or transfer its use to novel situations [BC95]. In contrast, interaction in on-line learning environments has very positive implications on student satisfaction, and high-order learning [And03, Kha00]. We comment each type of interaction below.

2.2.1 Student-Student Interaction

Interaction between students involves a two-way reciprocal communication among learners, with or without the presence of an instructor [KWSB14]. By interacting with fellow students, they can exchange ideas which improves their achievement [Moo89, And03]. Limniou and Smith [LS10] studied the context of teaching and learning through VLE. Students stated that their difficulties regarding the courses could be facilitated by using collaboration tools and receiving individual feedback.

Regarding student satisfaction, Wiersema [Wie00] states that the use of on-line collaborative techniques increases the participation of the students making the interpersonal communication more effective, and therefore rising their level of satisfaction. Moreover, students perceive a sense of individual autonomy in collaborative scenarios, where they become more involved and engaged with the learning process [CG09]. In this sense, the study carried out by Garzón et al. [GCBSMS09] showed that students (in engineering courses) prefer active methodologies in which they could feel to be a participative agent. This fact may improve their motivation and participation. Harlen and James [HJ96] obtained similar conclusions: collaborative learning emphasizes a more active design, participation and interaction on the perspective of both teachers and students. Additionally, the sharing of knowledge and resources may engage students in higher level thinking skills that promote active and interactive learning from multiple perspectives.

In problem solving scenarios, the sense of collaboration relies on the interactions between students for giving a cooperative solution to a problem. This interpretation does not coincide with our goals, since our pursued VLE should be domain-independent and should not be focused on a single application domain. For this reason, we will strength the interaction between students by providing a space where students will be able to post and solve common questions, share bibliography, obtain information about the VLE... Thus, the proper students will be facilitators of support information about any course maintained by the VLE. Furthermore, they will be in charge of giving their opinions about the educational content collaboratively, being this a great instrument for teachers.

2.2.2 Student-Teacher Interaction

The student-teacher interaction is crucial in e-learning, since it allows an increased interaction and a more equitable distribution of the teachers' attention among the students [H$^+$91]. Furthermore, student-teacher interaction is a key factor considering the learner-centered paradigm. In it, the role of the teacher is to facilitate students to construct their own knowledge. This can take the form of teacher delivering information, encouraging the student, or providing feedback. In addition, this can include the student interacting with the teacher by asking questions, or communicating with the teacher regarding course activities [She09]. This kind of interaction contribute to that objective by establishing an environment that encourages students to understand the content better [SBM$^+$05].

More concretely, we focus on teacher feedback. *Feedback* can be defined as the exchange of information between teacher and student about an action, event, or process that results in enhanced student learning. Timely feedback has been noted as an important variable in student learning [CG87] and distance education courses [Ber02, BCS01, Sci02, TWCF02]. In fact, feedback is critical to assessment and provides students information about their progress in the course [CDBS01]. According to Thurmond and Wambach [TW04], the need for quality feedback in e-learning becomes more paramount because of several factors. First, because a e-learning course lacks face-to-face interaction, receiving written comments from the instructor becomes even more crucial. Second, the geographic separation between student and teacher may limit physical contact and foster a sense of being disconnected from those in the course [AR02]. Third, the flexibility in the pace of Web-based courses allows students to work ahead. Therefore, faculty need to provide timely feedback so that students can maintain their own pace and schedule. Finally, the use of the Web technology for providing feedback may create the need for additional faculty support [CDBS01]. If a large number of students are enrolled in a Web-based course, some faculty may need assistance responding in a timely manner.

Several studies hold that student-teacher interaction is a significant (if not the most important) contributor to student satisfaction [She09, TW04, Bat07]. For example, Jiang and Ting [JT00] examined what variables were predictive of student's perceived learning. Results of multiple stepwise regression analysis indicated that student-instructor interaction was the most significant predictor of perceived learning. Similarly, Fredericksen et al. [FPS$^+$00] reported the most significant variable to learning in an on-line course was students' interaction with the teacher. This relationship was significant because those students who felt they did not have adequate access to their on-line teachers tended to feel that they learned less. Finally, students agreed that timely, prompt feedback from their teacher contributed to positive perceptions of student-teacher interactions [CDBS01, TWCF02].

In consequence, we will consider the development of communication and collaboration paths between teacher and students. Teachers will be participants in the information space of their courses. In addition, we will include the idea of teacher's feedback on the assessment of learning.

2.2.3 Student-Content Interaction

In the context of learner-centered paradigm, student-content interaction refers to a one-way process of elaborating and reflecting on the subject matter or the course content [Moo89]. Conventional e-learning systems were based on instructional packets that were delivered to students using Internet technologies. The role of students consisted in learning from the reading and preparing assignments. By contrast, the new paradigm is built around collaboration, which assumes that knowledge is socially constructed. Learning takes place through conversations about content and grounded interaction about problems and actions.

Some studies show the benefits of promoting this kind of interaction. It provokes that students think to themselves about the information, knowledge, or ideas gained as part of a learning process. Therefore they cognitively elaborate, organize, and reflect on the knowledge they have obtained by integrating prior knowledge [Moo89]. The promotion of student-content interaction matches with the constructivist paradigm, in which the student is considered as an actor that takes part in the information space [CD96]. Additionally, this model implies a direct benefit for teachers that are not forced to construct the educational content by themselves. In this way, Yu et al. [YLC05] found that by enabling students to construct, assess, and review the learning resources (concretely quiz questions in that work), students' confidence and cognitive ability in the applied content domains were promoted.

In the following, we will implement a constructivist scenario of collaboration between students and learning content. One the one hand, it will help students to improve their learning process. On the other hand, the learning content will be enriched not only by the teacher but also by their students.

2.3 E-Assessment and Self-assessment

The purpose of *e-assessment* is to collect evidences to judge the quality of learning and to provide feedback to guide the student trough the learning process [CHB04]. The assessment can analyse several characteristics of students' performance [KS04, KAH$^+$97] or a combination of their performance and invested mental effort [Kal06, SPBVM04]. There exist two common forms of assessment. Formative assessment is essentially feedback to the teacher and the student about present understanding and skill development in order to determine the way forward [HJ96]. In contrast, summative assessment describes learning achieved at a certain time for reporting to parents, other teachers, students themselves, and other interested parties.

Llamas et al. [LNFIGTMF13] found the following benefits about e-assessment. From the teacher viewpoint, it facilitates the classification and management of errors, guaranteeing the coherence of the grading and revision processes. Additionally, it establishes the way forward to create a feedback scenario, providing useful information to improve teaching methods and course material. From the student viewpoint, they can access information about their exams and responses readily available anywhere any time, facilitating rapid feedback. Additionally, students can benefit from incremental support through assessments resources which allows them to complete the learning objectives gradually [CD96].

E-assessment and feedback have potential impact in terms of student achievement if they are correctly implemented [HJ96, OC11]. Considering the influence of assessment in student achievement and according to Cunningham and Duffy [CD96], since e-learning is such a complex process, students can benefit from incremental support through assessments resources which encourage them to build their skills gradually, and provide feedback on their efforts. Moreover, assessment strategies may lead students to the critical points in the course.

Self-assessment provides a framework where the students can establish their own learning goals, evaluate them, and adapt their learning behaviour in function of the results obtained in the evaluation. Boud [Bou95] defined self-assessment as

> "the involvement of students in identifying standards and/or criteria to apply their work and making judgements about the extent to which they have met these criteria and standards".

For self-regulated learning to be equally adaptive and effective, students should be able to accurately monitor and assess their own performance and recognize what an appropriate next task would be [KVGP12]. Therefore, self-assessment is key aspect for defining a self-regulation environment for students [Puz08, Wan11]. It is proved to be a well-suited strategy for the development of students' ability of reflection on their own learning, students' ability to learn how to learn, and students' autonomy [Bou95, Mis11, SZ11]. However, inaccurate self-assessment may negatively affect the selection of an appropriate new learning strategy. For example, if students overestimate their performance, they may choose a task that is too difficult for them [AC04, MA08]. Moreover, even when self-assessment is accurate, novices may still experience problems in selecting appropriate learning strategies.

As conclusion we observe that the implementation of self-assessment mechanisms is highly desirable because it provides an excellent scenario for students to be participative actors in the learning process. Nevertheless, the self-assessment mechanisms must facilitate assessment tasks in function of comprehensive goals in order to avoid inaccurate evaluations. Student should be able to obtain accurate assessment resources even if they have no prior knowledge.

2.4 Formative/Summative Feedback

The sense of educational feedback in VLEs is strongly related to the concept of electronic assessment. A feedback mechanism establishes the path of assessment information between the system and the users. We can identify two types of feedback that rely on the type of assessment paradigm employed. On the one hand, *formative feedback* can be defined as the information communicated to the student and the teacher over the course of instruction to modify their thinking or behaviour, which in turn will improve both learning and teaching [Shu08]. This concept rises from the idea of formative assessment. On the other hand, *summative feedback* refers to the information communicated to the student and the teacher after the course of instruction. It is related to the idea of summative assessment.

The benefits of formative feedback in contrast to summative feedback are clear [Sad89]: students are able to obtain information about the desired knowledge and skills to be acquired in real-time; both teacher and student can compare the student's real knowledge and skills acquired at a given moment with the desired ones; both teacher and student can track previous exams; both teacher and student can tailor learning activities if they found differences in the previous comparison, also in real-time. From the student viewpoint, they become self-regulated learners [NMD06], i.e. they make an active regulation and monitoring of the learning process by setting the learning objectives and strategies, management of learning resources, etc. From the teacher viewpoint, they can obtain feedback during the course in real time about the current students' performance. Using this information, teachers can take decisions about the needed instructional adjustment over the course [LNFIGTMF13].

Concluding, our desired VLE should contain a feedback mechanism based on the idea of formative assessment. That is, the system should not only incorporate a path of detailed information about the students' performance for both teachers and student, but also the corresponding usage schedule should consider a stage of self-regulation and self-assessment to enable the formative feedback process. This could facilitate both teacher and student to make a dynamic adjustment of the learning process.

2.5 Blended Learning Environment

The term *blended learning* is often employed to refer a learning environment that blends asynchronous Internet technology with face-to-face learning. The benefits inherited from this model in contrast to the face-to-face model are experienced by the interaction and the sense of engagement in a community of inquiry and learning, achieved through the effective integration of the information accessible by the Internet [GK04]. Community provides the stabilizing and cohesive influence that allows an open communication and makes the most of the limitless access to information on the Internet. Blended learning has the capabilities to facilitate this scenario. According to Garrison and Hanuka [GK04], the combination of synchronous and asynchronous models offers a distinct advantage in supporting high levels of learning through critical discourse and reflective thinking.

In this sense, the blended learning model not only fits with real education scenarios, but also it presents advantages by itself. Traditional teaching meets on-line education making a rich educational scenario.

3 ivLearn: an System for Learning Supported by Computational Intelligence Mechanisms

Throughout this Thesis we have developed several intelligent methods for enhancing traditional VLEs. In addition, in the first part of this chapter, we have examined a number of design principles that are desirable for improving students' level of satisfaction and achievement. Hence we are finally ready to present a complete e-learning system that combines the intelligent methods without forgetting the pedagogical dimensions of the learning process. More concretely, we present the ivLearn (Intelligent Virtual Learning) system, a VLE designed for support a variety of learning activities in courses of any domain.

3.1 VLE Features

The ivLearn system presents the following features:

- **Multi-domain courses**. The ivLearn system is not constrained to a single application domain. The system do not require from predefined learning resources. In contrast, the creation of learning resources is considered as part of the learning experience itself. This allows to create courses from scratch, regardless the domain of application. Furthermore, the subsystems embedded as modules of ivLearn system work with independence of the domain of the data.

- **Course structure management**. The system implements the management of courses structures from scratch. Each course is composed by issues, i.e. specific lessons. This scheme allows to establish a well-defined organization of the learning resources. When a resource is created within a course, it can be assigned to one or more issues. Additionally, a course can be divided into groups. Therefore, different teachers can supervise different groups of a same course. Thus, students can be registered in one or more groups belonging to available courses.

- **Learning resources management**. We have considered three kind of initial learning resources for ivLearn: GIFT quiz questions, collaborative learning tasks, and frequently asked questions (FAQ) lists. The initial set of learning resources is completed with information and visualization techniques, as commented below.

- **Collaborative construction of the educational content**. Our VLE follows a constructivist approach regarding the construction of the learning knowledge. The system implements a collaborative protocol for the creation of learning resources. Both students and teachers in the system are able to create learning resources. In order to guarantee the quality of the created learning resources, the system provides teachers with alerts after the resources are proposed by students. Then, teacher can assign a grade to each learning resource representing its quality. The benefit of this scheme is threefold. First, the creation of learning resources can be viewed as part of the courses' goals. The grades assigned to the resources can be employed by the teachers as information related to their students' performance. Second, the resources' grades allow to categorize the learning resources in terms of quality. For example, the teacher could decide that only the resources with a grade higher than 7 are available during a course. Third, the teachers are released from the tedious and error-prone task of defining the whole educational content by themselves.

- **Collaborative space for students**. The system includes a collaborative space where students are able to post and solve common questions, share bibliography, obtain information about the VLE. . . Students are encouraged to interact between them in order to make a flexible learning experience.

- **Student-teacher collaboration**. As part of the course activities, we have designed a direct interaction protocol between teachers and students. The interaction is realised by means of the collaborative learning tasks. When a teacher assigns a learning task to a student, a collaboration process is initiated. If the answer given by the student is not correct enough, the teacher could give him/her some feedback or advices in order to improve his/her previous answer. Therefore they are directly communicated until the learning task is completed.

- **Intelligent acquisition of the knowledge from the educational content**. The ivLearn system includes a set of intelligent mechanisms that allow teachers to obtain the knowledge underlying in the educational content, regardless their technological expertise. The knowledge is extracted and represented in three meta-data structures with different levels of semantic richness: taxonomies, folksonomies and ontologies.

- **Intelligent representation of the knowledge from the educational content**. The information gathered in the three meta-data structures commented in the previous item is used by ivLearn for automatically generating visual representations of the knowledge of the learning domain. The system is able to automatically create resources' indexes, tag clouds, and concept maps. Such visual representation allows an effective access, navigation and understanding of the educational content.

- **Self-assessment mechanism**. The system includes a self-assessment space based on the automatic generation of tests. Such tests are automatically generated in function of a set of assessment objectives defined by the student. Therefore, students are able to assess their knowledge at any moment. Then, in function of the results they can adapt their own learning experience. For example, if the results are not well enough, they could review those concepts in which they have failed. They could also increase the difficulty of the tests, whether they obtain good results.

- **Teacher summative assessment**. Similarly to the functionality commented in the previous item, ivLearn includes a assessment space for teacher where they can set assessment objectives in order to generate tests. Such tests could be regarded as summative assessment resources. As addition to the assessment objectives, teachers can define exclusive parameters: number of opportunities for passing a test, grading mode, and time-constraints. Then, the system automatically assigns a different test for each student in the corresponding group (or course).

- **Formative/summative feedback**. The two assessment strategies bring the capability of generating formative and summative feedback to students. The results obtained after the two assessment processes can be consulted by both students and teachers, allowing an active regulation and monitoring of the learning process as well as the comprehension of the real level of knowledge of students.

- **Virtual tutor**. The ivLearn system allows students to register in the virtual learning subjects available in their courses. The learning subjects are directly managed by an intelligent tutoring system, which provides students with adapted material in function of their current knowledge state. This subsystem acts by simulating the behaviour of a human tutor. Along the development of learning subjects, students can acquire the knowledge of concepts from the learning domain. Therefore, the virtual learning subjects can be considered as a complementary material of their corresponding courses.

- **Information about students' performance**. The information related to the whole actuation of any student is stored in the system. The stored information not only contains data about the learning processes (proposed resources, grades, ...) but also log data (access log, action log, usage statistics, ...). The learning related data is available for both students and teachers at any time. Regarding the log data, it is available for the manager of the system.

3.2 Design Architecture of ivLearn

The ivLearn system includes an extensive set of functions involved in the whole life cycle of a learning experience: from the creation of the educational content to the final assessment of the students. The functions are implemented in the system following a modular approach. The different systems and frameworks discussed along this Thesis are included as modules of ivLearn, as well as the general functionality of any VLE. The modular architecture favours the cohesion between the logical and physical design, and guarantees the balance of the levels of granularity and aggregation. Additionally, the modules could be modified or replaced with minimum complexity. Figure v.1 displays the modular architecture of ivLearn and may serve to guide the reading. We analyse each independent module in the following.

3.2.1 Course management module

The course management module maintains all the courses' structure in the system. It is exclusively used by the manager.

A *course* contains a set of courses issues and it is divided into groups. All the elements of the course's organization can be edited or deleted. Logically, if a course is deleted, the dependent elements of the course (courses issues, groups, student's performance information, assessment tests, ...) are also deleted.

First, the *course issues* are primarily used for classifying the learning resources. The organization of learning resources by course issues allows the system to perform more complex operations at issue-level. For example, students can consult the concept map visualization of a single course issue, or generate self-assessment tests related to concrete course issues. The course issues are shared by the groups of a same course.

Second, the *groups* are mainly employed to organize users. A group include at least one teacher and a set of students. Therefore, the system limits the scope of the learning activities at group-level. For example, teachers can consult uniquely the performance of students in their groups, students can only perform summative assessment tests or register in learning subjects which were created in their groups...

As it can be seen, this module brings the system independence regarding any particular course structure or content.

3.2.2 User Management Module

This module manages the users in the system. There are three user profiles in our application:

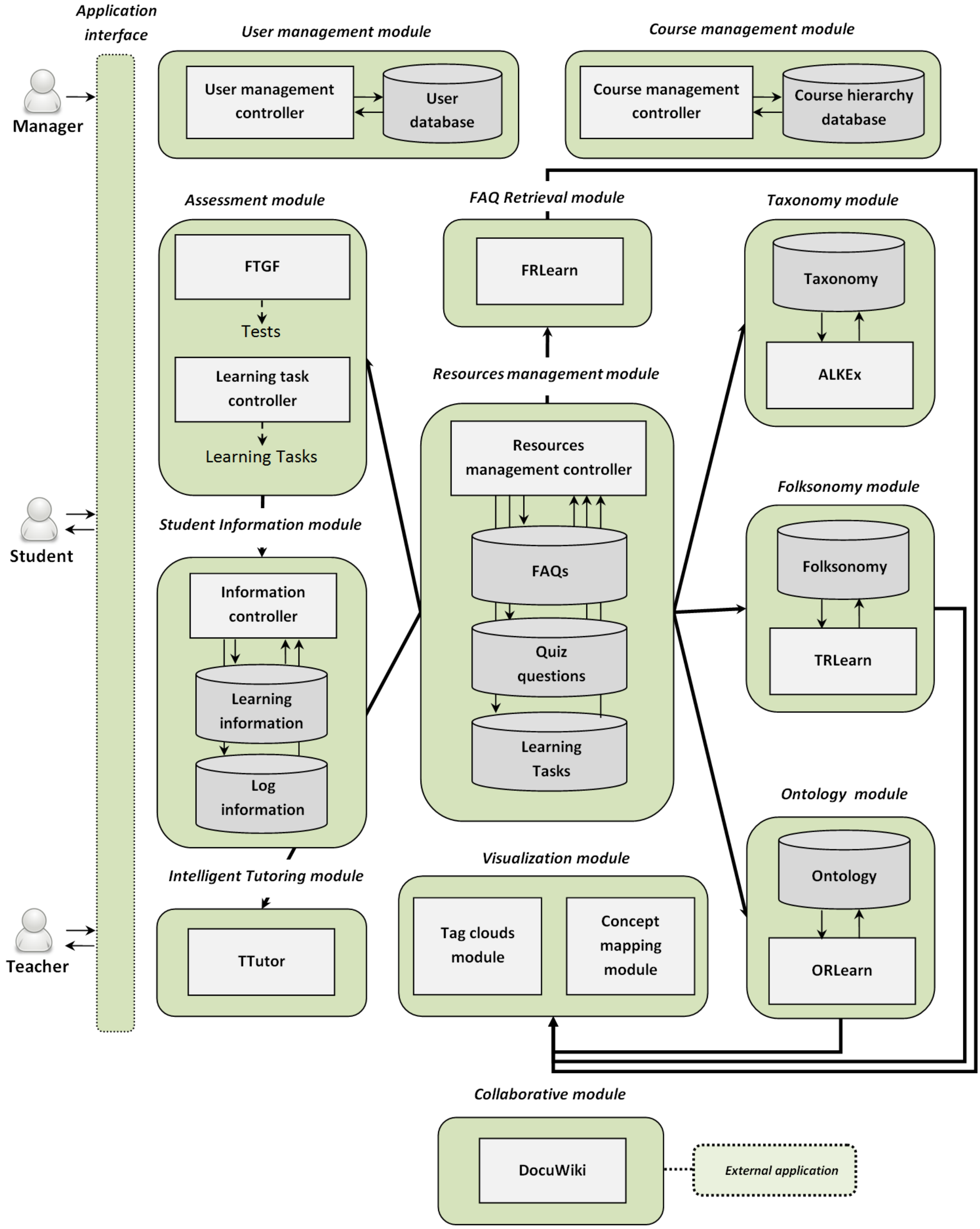


Figure v.1: Architecture overview

First, the *manager* is mainly responsible for creating the course structure including courses and groups. In addition, the manager can add, edit, or delete users of any profile at any moment. Managers can access to the log information of the system: user's access log, and user's action log.

Second, the *teacher* is responsible for the learning experiences which occur in his/her group. The functions of a teacher can be grouped into two classes: student-related functions and organization functions. On the one hand, the student-related functions include the control of the student's performance: management of the students enrolled in the course; review and validation the learning resources proposed by students; summative assessment test generation; direct interaction with the students trough collaborative learning tasks; view of the students' performance information .... On the other hand, the organization functions include: creation of course issues; creation of learning resources; use of the intelligent knowledge acquisition modules; creation of learning subjects for the intelligent tutoring module, etc.

Third, the *student* takes place in the available learning processes: proposal of learning resources; realisation of self-assessment tests, summative assessment tests, and collaborative learning tasks; interaction with other students using the DocuWiki collaborative space; use of the navigational tools (resources index, tag clouds, and concept maps); use of the FAQ semantic search; and enrolment in learning subjects of the course. Besides, students have always available their performance information.

3.2.3 Resource Management Module

The resource management module is the base of the rest of modules in the system. The resource management controller handles the creation, edition, and deletion of learning resources. We have defined three main learning resources as core of the learning activities carried out by ivLearn: FAQ lists, GIFT quiz questions and collaborative learning tasks. The core resources are subsequently complemented with other learning elements which are extracted by the intelligent modules of our application. That is, the navigational tools (resources index, tag clouds, and concept maps), and the Wikipedia's articles linked to the concepts in the concept maps.

Quiz questions and learning tasks proposed by students but not still validated are separately stored from trusted resources — already validated or created by experts. Once a resource of these types is approved, it could be linked to one or more issues of the course. This module allows to list, search, and display resources sorted by issue. Also, this module stores some statistics reflecting the use of each resource in the application. For example, the number of times a given question was chosen into a test, the percentage of correct answers, or the date it was added to the system, are stored by this module.

FAQ lists A FAQ list is composed by a list of questions and answers which frequently appear in a determined context for a concrete matter. A FAQ list is defined by a set of FAQ entries. Each FAQ entry is composed by a set of questions (reformulations) and an answer. The number of reformulations in each FAQ entry depends on the life time of the system. The higher the quantity of reformulations, the better its syntactic variety is.

In this work, we have employed the collaborative space provided by DocuWiki (commented below) for handling the FAQ lists regarding each course's group. Any student can create questions that could be answered by the rest of the students or by the teacher. The question-answer pairs are then collected and stored in ivLearn by the resource management controller, being available for the FAQ retrieval module.

GIFT Quiz Questions The GIFT (General Import Format Technology) format was created by the Moodle community to import and export questions. GIFT quiz questions present an excellent diffusion [LH08, Cos07]. It supports Multiple-Choice, True-False, Short Answer, Matching and Numerical questions. Various question-types can be mixed in a single text file, and the format also supports question names, feedback comments and percentage-weight grades.

GIFT quiz question are defined in plain text, following the criteria imposed by the corresponding GIFT format. In the simple form, the question comes first, then the answers are set in between brackets, with an equal sign (=) indicating the correct answer(s) and tilde ( ) the wrong answers. A hash (#) will insert a response. Questions can be weighted by placing percentage signs (%..%) around the weight. Comments are preceded by double slashes (//) and are not imported. For example, the following example represents the format used for defining a a multiple choice quiz question: “Who’s buried in Grant’s tomb?=Grant no one Napoleon Churchill Mother Teresa ”. A detailed guide about the GIFT format can be found in [Moo13].

Collaborative Learning Tasks The collaborative learning tasks were designed as part of our research in the field of e-learning. It was firstly presented in our work [RMC13]. A collaborative learning task is considered as any request of information about a given topic formulated by a teacher. A learning task consists of an information request and the corresponding student’s answer, both formulated in HTML. Any external resource can be added to the application enhancing the learning knowledge, as well as mathematics formulas, tables, or so on.

This kind of learning resources involves direct collaboration between the teacher and the student. If the answer given by a student for a task is not correct enough, the teacher could give him/her some feedback or advices in order to improve his/her previous answer. The collaboration path is depicted in Figure v.2. First, the system sends an message to the teacher after student answer is proposed. Next, the teacher can mark the answer with a grade, or send the student an advice trough a simple HTML interface. Then the student receives a message informing about the grade or the teacher’s advice. In regard to the second case, the student could improve the prior answer considering now the teacher’s feedback.

Collaborative construction of the educational content As commented before, both teachers and students can add learning resources. First, the FAQ lists are created through the external application DocuWiki, following a collaborative scheme. Second, the quiz questions and the learning tasks are directly controlled by ivLearn. Quiz questions and the learning tasks proposed by teachers are directly added to the knowledge base, while those proposed by students must be first reviewed by teachers. For example, let us suppose that a user propose a quiz question about a given topic. Then, the teacher could approve or discard it. In the former case, the quiz question becomes part of the knowledge base. If so, the teacher assigns a grade reflecting the quality of the resource that will also serve for the future evaluation of the student. This framework encourages students to investigate the topic in depth before reporting any resource. The same scenario goes for learning tasks. The grades assigned to resources are stored in student information module.

3.2.4 FAQ Retrieval Module

The main goal of the FAQ retrieval module entails retrieving FAQ entries from the available FAQ lists that are relevant to a user question formulated in natural language. The FAQ entries are then presented as a ranked list. The FRLearn system (Chapter III Section 2) is in charge of such task.

Initially, in an automatic learning stage, the system extracts information units from each element of the FAQ list. Hence, each FAQ entry of the collection is associated with a set of information

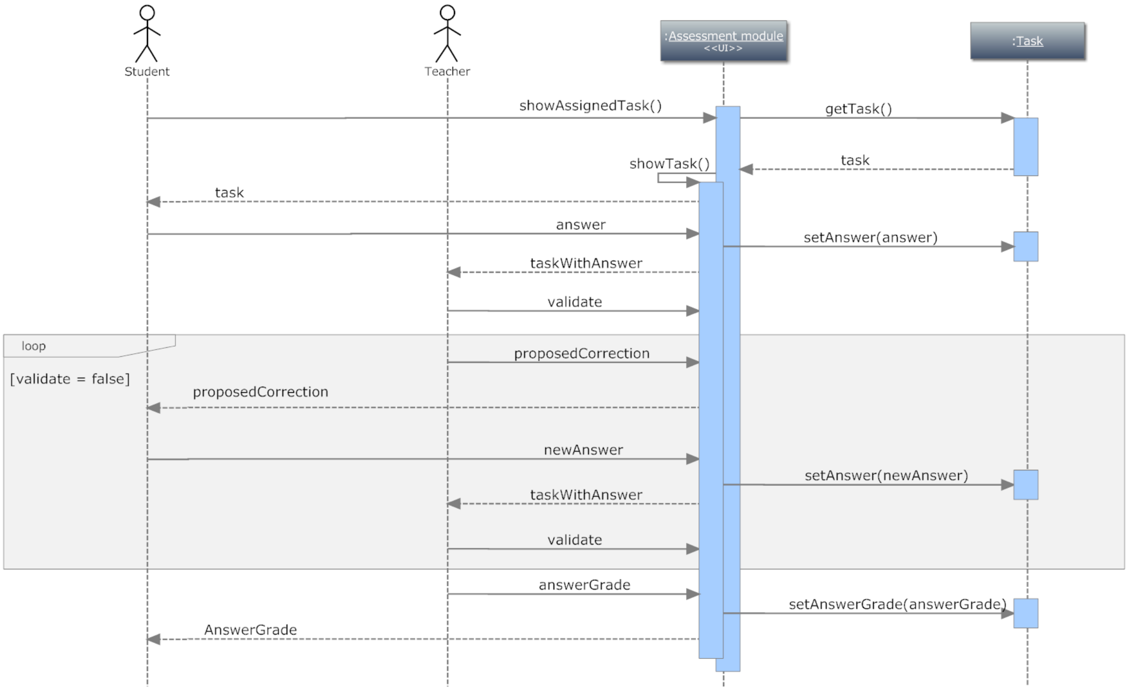


Figure v.2: Collaborative Tasks sequence diagram

units. These units contain the semantic information of each element and are used in subsequent steps of the retrieval process. The teacher is in charge of executing this automatic stage when the FAQ lists are modified.

3.2.5 Taxonomy Module

The taxonomy module is in charge of extracting the taxonomic structure of the available learning resources. The ALKEx system is employed to that end (Chapter II Section 2). This module automatically extracts the meaningful terms from each resource of the resource base and use such terms to make a index structure. During the process, some key terms are linked to Wikipedia articles web pages. In addition, the module provides a simple form of navigation through the course structure. The index structure contains the key terms contained in the taxonomy, ordered by a weights.

Each key term is additionally enriched by a set of morphological variations, or synonyms. Hence the user can access the related learning resources simply by clicking in the corresponding icons next to each term. What is more, the application includes a search mechanism, facilitating the exploration of the index structure.

The index can be dynamically modified in function of the course issues. Considering that the learning resources are catalogued into specific issues of the course, the index is adapted to them when the user marks one or more courses issues. Therefore the indexation application only displays the key terms obtained through the learning resources belonging to marked issues.

Next, the key terms which are linked to Wikipedia articles include an icon to the corresponding article's web page. This is arguably a helpful resource for the student. Finally, the teacher can modify each extracted term if it is needed.

3.2.6 Folksonomy Module

The goal of this module is to categorize the available learning resource by means of tags. To that end, ivLearn includes the TRLearn system commented in Chapter II Section 3. The system can perform from an automatic way (without any supervision) to a social scheme (teachers are in charge of select which tags are or are not related to the corresponding resources). The set of selected tags are employed to generate a conceptually extended folksonomy. In this first version of the ivLearn system, we have considered the teachers as responsible for selecting the tags. This election was made in order to guarantee the coherent representation of the domain knowledge. Nevertheless, the inclusion of students as actors in the tagging process would require minimum changes.

Every time a resource (quiz question or learning task) is created, TRLearn extracts the most relevant candidate conceptual tags from it and assigns a weight for the candidates. Then, the candidate tags are presented to the teachers in the course in order to validate them following a collaborative scheme. In any case, the final election falls on the user, who can freely select one of them or add a different one. Finally, the resource and the selected tags are linked in the folksonomy. All the teachers in the course can tag available resources.

3.2.7 Ontology Module

This module is in charge of extracting the ontological representation of the domain form the set of available learning resources. Our application includes the ORLearn system to that end (Chapter II Section 4).

The method performs in two stages. On the one hand, the concept extraction stage is able to detect main concepts in the learning resource dataset. During the process, the extracted concepts are linked to Wikipedia's articles, if it is possible. On the other hand, the system extracts the main relations existent between the concepts. It is able to detect five types of relations: two taxonomic relations (superclass and subclass relations) and three educational-based relations (subtopic, subordinate and content-based relations). The teachers in charge of the course are responsible for filtering out the noisy concepts, accepting the valid ones, and accepting the valid extracted relationships.

3.2.8 Visualization Module

The visualization module includes two sub-modules which implement two visualization techniques: tag clouds and concept maps.

Tag clouds The tag cloud model serves as overview and navigation scheme of the educational content. The tags represent the main concepts of the domain. Each tag is displayed with a size and weight corresponding to its importance under the domain. Additionally, the tags are links to the learning resources. The tag cloud module is in charge of generating this kind of representation. It works in two different ways. On the one hand, it is able to work with user queries, extending the results obtained by FAQ retrieval module. On the other hand, it is able to represent the main tags contained in the conceptually extended folksonomy.

Firstly, FRLearn is extended with tag cloud representations of the results. The system implements the framework explained in Chapter III Section 3.4.1. In addition to the FAQ entries related to the user query, the system presents a tag cloud representation of the information units also related to the query. The tags in the cloud representation serve as links to another user queries. That is when the user clicks on a tag, the output interface shows a new ranked list of FAQ entries (now related to the clicked tag), and the cloud is refreshed in the same manner. We called FAQ cloud to this tag cloud representation and its linking structure for searching.

Secondly, TRLearn is extended with tag cloud representations of the folksonomy. The system implements the framework explained in Chapter III Section 3.4.2. It summarizes the content of the documents in the system. Therefore, students are able to explore the main concepts of the domain and to access the documents linked to each tag. In addition, the different synonyms of a each conceptually extended tag are included in the visual representation. Next to each tag, the number of synonyms of such tag is displayed in brackets. If a student clicks in such number, the interface changes to display the different terms of the tag. The terms also present a font size variation, attending to its independent document frequency. As more times a term is referenced in the resource set, as higher font size it presents. This fact allows students to gain knowledge about the domain under consideration. Students not only will be able to know the different terms regarding to a same concept, but also will see the usual ways to call it in the domain.

Concept maps The concept map model is obtained through a straightforward translation of lightweight domain ontologies. It is composed by two different sub-models. The summary model offers an overview of the highly related concepts in the domain. The individual concept map model focuses at concept-level, and shows a detailed visualization of each concept of the domain and its relations. The concept mapping module is in charge of this task.

The concept mapping module implements the framework explained in III Section 3.4.3. The visual representation of the ontology of the course can serve as guide for students through the learning process. The representation initially offers the main concepts of the course ordered by its importance with regard to the educational content and the different issues of the course; and afterwards it is focused at concept-level. Thus, students will be able to understand the whole course structure and then acquire a deeper knowledge on individual concepts. Moreover, the representation facilitates the access to the learning resources linked to each concept, and to corresponding Wikipedia article's web page, if it is available.

3.2.9 Assessment Module

The assessment module handles all the process related to the assessment tests and the collaborative learning tasks, managed by the Fuzzy Test Generation Framework (FTGF) and by the learning task controller respectively.

Fuzzy Test Generation Framework First, the FTGF, defined in Chapter IV 2, is implemented in the assessment module. The FTGF is able to dynamically generate tailored assessment test composed by GIFT quiz questions from a set a user-selected requirements. When a user desires to generate a test, the application interface shows a variety of assessment objectives (test requirements) including: number of test items, degree of practical-related items in contrast to theoretical-related items, difficulty of the items, frequency of inclusion of the items, and relation of the items to the course issues. Once the assessment objectives have been setted, the FTGF looks for the quiz questions of the corresponding course which better fits with the user-selected assessment objectives in order to create the test. In function of the role of the user, the system forks in two possible scenarios that are commented in the following.

On the one hand, the tests generated by students have a self-assessment purpose. Students can adjust their assessment goals at each step. Starting from unclear and blurred requirements, they can later focus on more specific requirements which better fit their current needs. Self-assessment tests could be attempted as many times as the students desire in order to auto-evaluate their own progresses. When a student completes a test, the results are stored in the student information module and are displayed through the application interface. The result interface also displays the correct answer for each question in the test, and the question feedback, if it is available. In addition, information related to the individual quiz questions (the number of times that the question was chosen into a test, the percentage of correct answer, . . .) is stored.

On the other hand, the tests generated by teachers have a summative assessment purpose. Additionally to the previously commented objectives, teachers have available other exclusive objectives: number of opportunities for passing a test, grading mode (maximum grade, minimum grade, or average grade), and time-constraints. The assessment objectives are stored in the system, and a notification is sent to the students in the corresponding group. Then, the students can face the teacher-defined assessment process, whether the time-constraint are satisfied (i.e. the current date is between the established starting and ending dates). The FTGF generates a different test in real-time for each student and opportunity. Thus, possible cheats are avoided, even if the students performing the assessment tests are in the same room. The results of the tests are stored in the student information module and are displayed in the application interface as well. The correct answers of the test items are only shown if the student passes the test. In addition, during the marking process the difficulty of the items are readjusted, and statistics about the individual quiz questions are stored.

Learning task controller The learning task controller is devoted to manage the collaborative learning tasks. The interface allows the teacher to assign any task to any student in his/her group. This assignment can be defined manually or automatically. In the second case, the teacher should specify also one or more issues of the course. Then, the system looks for the learning tasks related to the to the input course's issues and assigns one to each student, provided that he/she has not created the resources or he/she has not been previously assigned to the learning task. Further, students can also assign the tasks that they prefer. The are free to establish a direct interaction with the teacher.

3.2.10 Intelligent Tutoring Module

As learning complementary material, students can access to our Intelligent Tutoring module. It contains the TTutor intelligent tutoring system commented in Chapter IV 3. It provides student with learning and assessment resources adjusted to his/her background knowledge and performance within a learning subject.

Each available learning subject defines an independent learning experience, with their own learning goals. Therefore students can select one or more learning subjects from the list of available subjects. Each learning subject contains a set of game-like learning stages that students must master in order to reach the learning goals. Each stage is composed by a set of interrelated concepts. Students have accessible information about their level of knowledge regarding the concepts in their current stage. Concepts presents a set of learning resources that students can always consult.

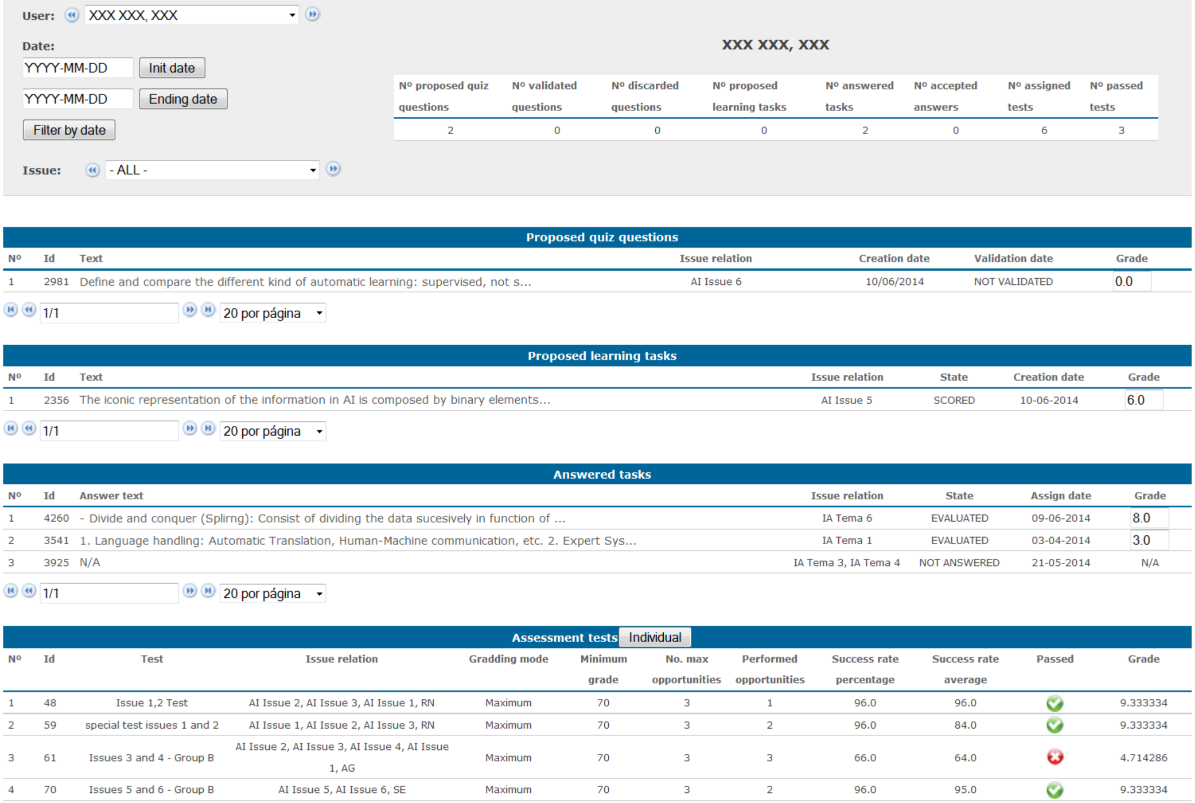
When a student is ready to prove his/her knowledge, a e-assessment module generates a test in function of his/her previous background knowledge and performance. The results of the test are used to infer the new student's knowledge state. Once the learning goals of the current stage are reached with a minimum grade, the student can select the subsequent stage or stay in the current stage in order to completely master the corresponding concepts.

The user interface clearly shows the current student's levels of knowledge of all the elements which compose the learning experience. It can be accessed by both the student and the teacher(s) in charge of the group. The information related to students' performance is kept in this module, separated from the student information module. Besides, through the user interface students are able to access the learning resources from which acquire the corresponding knowledge before performing a test.

3.2.11 Student Information Module

The student information module manages two kind of information: learning-related information and log information.

Learning-related information The student information module stores all grades attached to proposed resources, responses to tasks, self-assessment test results, and summative assessment tests results. Certain usage statistics per student are also stored, such as the number of proposed quiz questions, proposed learning tasks, passed tests, ... The information controller is in charge of collecting such information, and displaying it into the application interface if it is needed. Figure v.3 shows the student information form containing the learning-related data. This information form can be accessed by both students and teachers. The form presents date and course's issue filters. In addition, the complete information can be exported into Excel format or be printed.



User: XXX XXX, XXX

Date:
YYYY-MM-DD Init date
YYYY-MM-DD Ending date
Filter by date

Issue: - ALL -

XXX XXX, XXX

Nº proposed quiz questions	Nº validated questions	Nº discarded questions	Nº proposed learning tasks	Nº answered tasks	Nº accepted answers	Nº assigned tests	Nº passed tests
2	0	0	0	2	0	6	3

Proposed quiz questions

Nº	Id	Text	Issue relation	Creation date	Validation date	Grade
1	2981	Define and compare the different kind of automatic learning: supervised, not s...	AI Issue 6	10/06/2014	NOT VALIDATED	0.0

1/1 20 por página

Proposed learning tasks

Nº	Id	Text	Issue relation	State	Creation date	Grade
1	2356	The iconic representation of the information in AI is composed by binary elements...	AI Issue 5	SCORED	10-06-2014	6.0

1/1 20 por página

Answered tasks

Nº	Id	Answer text	Issue relation	State	Assign date	Grade
1	4260	- Divide and conquer (Splirng): Consist of dividing the data sucesively in function of ...	IA Tema 6	EVALUATED	09-06-2014	8.0
2	3541	1. Language handling: Automatic Translation, Human-Machine communication, etc. 2. Expert Sys...	IA Tema 1	EVALUATED	03-04-2014	3.0
3	3925	N/A	IA Tema 3, IA Tema 4	NOT ANSWERED	21-05-2014	N/A

1/1 20 por página

Assessment tests Individual

Nº	Id	Test	Issue relation	Gradding mode	Minimum grade	No. max opportunities	Performed opportunities	Success rate percentage	Success rate average	Passed	Grade
1	48	Issue 1,2 Test	AI Issue 2, AI Issue 3, AI Issue 1, RN	Maximum	70	3	1	96.0	96.0	✓	9.333334
2	59	special test issues 1 and 2	AI Issue 1, AI Issue 2, AI Issue 3, RN	Maximum	70	3	2	96.0	84.0	✓	9.333334
3	61	Issues 3 and 4 - Group B	AI Issue 2, AI Issue 3, AI Issue 4, AI Issue 1, AG	Maximum	70	3	3	66.0	64.0	✗	4.714286
4	70	Issues 5 and 6 - Group B	AI Issue 5, AI Issue 6, SE	Maximum	70	3	2	96.0	95.0	✓	9.333334

Figure v.3: Evaluation form of a real student in ivLearn system

Log information The student information module also contains log information including access information and action information. Each time a student access into the system, a entry containing the id, the name, the group, and the hour and date is stored. In addition, each action in the system is recorded. For example, every time a student proposes a resource or performs a test, a entry containing the id, the name, the group, and the type of performed action, next to the hour and date is stored. The manager is the only actor who can consult this information.

The log information is very valuable for confirming or refuting complaints of the students. It may occur that some students claim of system malfunction. Sometimes, it is possible that the complaints are false, i.e. it is an excuse for failing to carry out a task. By means of the log information, the manager can consult if the student actually tried to perform that operation, if he/she was actually in the system in the time of the complaint... Logically, it is also possible that the student was right. In that case, the action log can serve to detect which functions of the system present errors.

3.2.12 Collaborative Module

The system includes a collaborative space where students are able to post and solve common questions, share bibliography, obtain information about the VLE....

We have implemented this space by means of an external application: docuWiki[1], a simple open-source wiki software. In this virtual space, students had available information resources, such as important links for the course, frequently asked questions about the course, useful bibliography or a guide about the ivLearn functionalities. Also they can contribute to this space posting new resources. The platform allows users to score every available resource with a five-point Likert scale, following the scheme imposed by the Web 2.0. Students are able to share their opinions about the resources collaboratively, while teachers are able to check if the resources are properly understood.

The frequently asked questions maintained in docuWiki are collected by the FAQ retrieval module, enabling then the user semantic search on the FAQ collection.

3.3 Using ivLearn: Definition of the Regulatory Methodology

The use of a VLE within a real educational scenario should be regulated and scheduled in order to fit with the real organization of a course. For this reason, we have designed a regulatory methodology for the use of ivLearn in a higher education course, concretely in a engineering course. Although the features of university engineering courses are different from other kind of courses (e.g. primary education), the processes involved in learning are similar, and therefore minimum changes should be necessary for applying the methodology to other contexts.

The educational context in which we find ourselves is governed by the Bologna plan[Hei05, Lou01]. Regarding the proper learning facets, the courses organized by the Bologna plan are regulated by European Credit Transfer System (ETCS). A ETCS is equivalent to between 25 and 30 hours of learning, including hours inside and outside the classroom. In addition, this plan aims to establish and develop the learner-centered paradigm, whose features are summarized in the following:

- The student goes from being a passive to an active agent of the learning processes.

[1] https://www.dokuwiki.org/dokuwiki

- The student should also be actively involved in the learning process as well as the teacher, having as much time as they need to achieve mastery.
- The educational knowledge should be constructed by student through gathering and synthesizing information in order to solve real-world problems.
- The teacher should be facilitator of the knowledge acquisition process by acting as guide, coach, and motivator as students become more active in their learning process.
- The teacher should give students some control over the learning processes in order to motivate them to work harder, with initiative and self-direction.

In consequence, we consider the VLE as an complementary tool of the face-to-face lectures, making up a blended educational scenario. The activities designed in the regulatory methodology can be performed at home, contributing in this way to the realization of the hours of learning which should be carried out outside the classroom. Further, the methodology takes into account the features of the learner-centered paradigm as well as the pedagogical dimensions viewed at the beginning of this chapter.

The regulatory methodology is composed by three stages. First, the first stage is related to the identification and acquisition of the main concepts of the domain. Second, the next stage concerns a deeper comprehension of the domain, expanding and increasing the previously obtained knowledge. Third, the last stage is designed to test the acquired knowledge, as part of any learning process. The two first stages promotes student self-regulation, student interaction (with other students, with the teacher and with the content), and student self-assessment as basis to reach the learning goals. In contrast, the last stage pursues to perform a summative assessment of the students, simulating traditional final exams. The stages and objectives of our learning methodology are detailed in the following subsections.

3.3.1 Learning stages

The regulatory methodology of ivLearn for its use in a higher education engineering course is composed by three temporal stages (Figure v.4). The three stages are not overlapped in time. The teacher is in charge of concreting how long is each stage. In this sense, each stage should be long enough in order to ensure the students' self-regulation, especially the first two. At the end of the last stage, the evaluation of the students is considered.

Concept acquisition stage The first stage is focused on the domain concept acquisition. That is, during this stage the students should understand the main topics of the course. To that end, the student interaction is promoted by means of two activities.

First, the students are requested to propose learning resources, including quiz questions, learning tasks, and frequently asked questions about the course. While students design suitable resources (for example, a quiz question), their effort is oriented towards an analysis of the subject (identifying the important topics on it, which relations have the topics among them, or which topics are more difficult to learn, among other responsibilities). Accordingly, the analysis and formulation of learning resources leads to the identification and acquisition of the main concepts of the domain. The benefits of this activity are two-fold. First, the student-content interaction is promoted, since the students are contributors of the information space, following a constructivist approach. The traditional role of students is modified, allowing them to proceed from passive to participative agents. In addition,

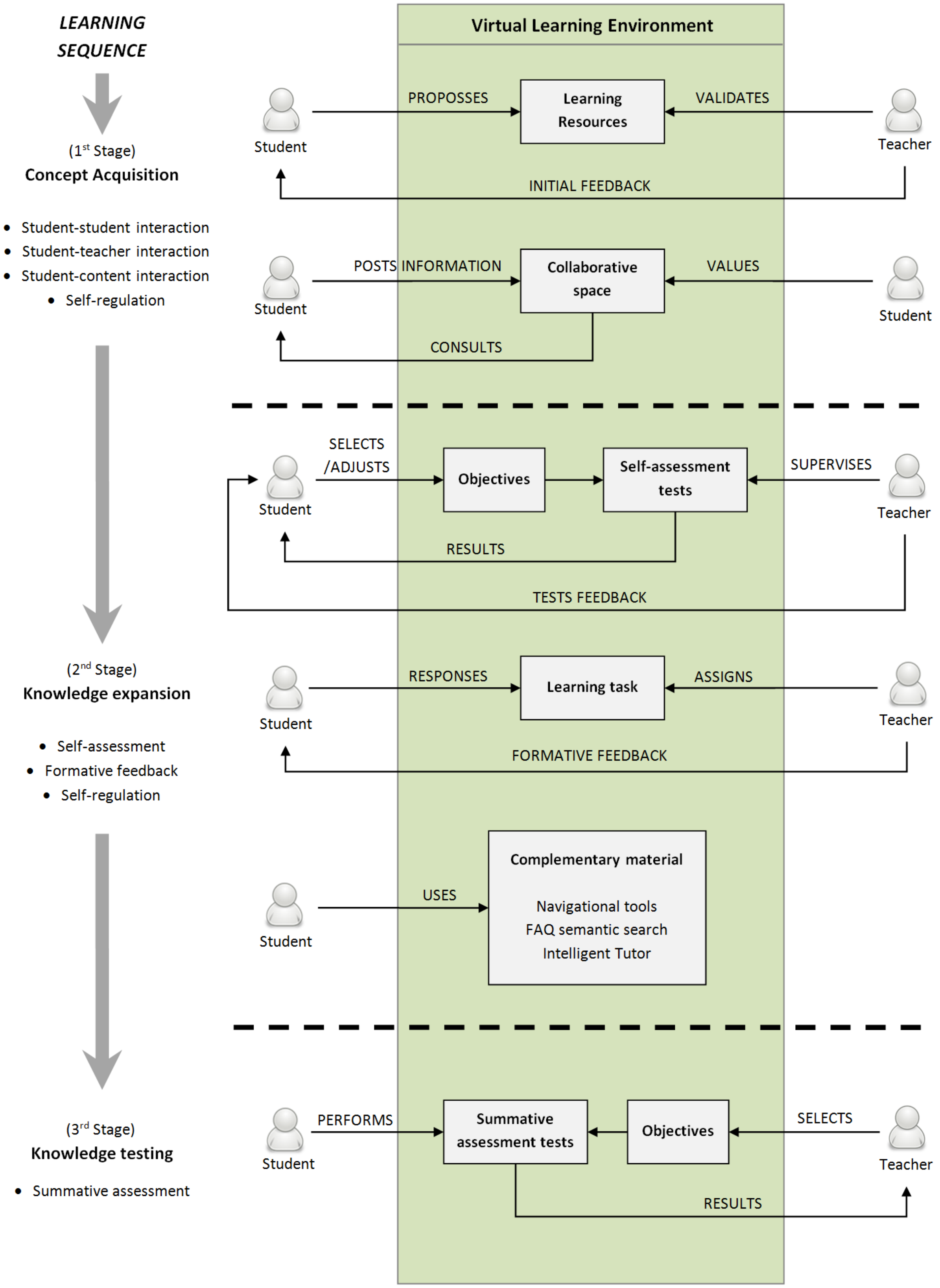


Figure v.4: Sequence of stages of the ivLearn regulatory methodology the teacher is released from the task of creating the educational content by himself/herself. Second, since the teacher is in charge of validating and marking the proposed resources, he/she can observe preliminary indicators about the knowledge level of the students. In case that the students have misconceptions, the teacher can initiate a feedback with the students, making the learning guidance more effective and efficient.

As second activity of the first stage, the students are encouraged to interact between them, using the collaborative space of the VLE. Thence, the students are able to value the resources proposed by other students, consult the resources, and answer the questions that other students had proposed about the course. The knowledge is collaboratively enhanced, and it is supervised by the teacher. As it can be seen, the collaboration scheme fosters the participation and interaction of students.

Knowledge expansion stage The second stage concerns a deeper comprehension of the domain, expanding and increasing the previously obtained knowledge. We have considered two main activities in this stage, which are complemented with the available tools included in the VLE.

First, the students are requested to initiate a self-assessment process. Thanks to the test generation framework included in the VLE, the students can set the assessment objectives that better fit their current knowledge state in order to obtain assessment tests. Therefore, the students can manage their own schedule through a cyclic process. This scenario allows students to regulate their own level, making conclusions every time they perform a test, and adapting the objectives as the same time that they acquire a deeper understanding of the domain. Moreover, students do not have to expend time looking for proper assessment resources, since they are automatically provided with. The assessment cycle should be self-regulated. That is, the students are responsible for deciding the assessment objectives in each moment, how many tests they should perform, etc. In this sense, the activity is not imposed by the teacher. In contrast, he/she takes the role of supervisor. By supervising the results obtained by the students, the teacher can distinguish the difficult topics of the course. For example, if some students usually fail in test items related to a concrete issue of the course, the teacher could focus their lectures on that course's issues, or provide specific feedback for them.

The second activity involves the direct collaboration between the students and the teacher by means of learning tasks. The learning tasks can be assigned to students by the teacher, or by auto-assignment. The process fosters the student-teacher interaction and the idea of formative feedback. The feedback process is stopped when the teacher considers that the response to the learning tasks is correct enough. Once again, students are responsible for their own schedule. Hence, the teacher should provide formative feedback in the shortest possible time.

This two activities are complemented with the available tools of the VLE. The students can employ the navigational tools included in ivLearn (resources' index, tag clouds, concept maps), search for question-answers by means of the FAQ semantic search, or perform learning subjects by means of the intelligent tutor. The use of this material is under the self-regulation of the student.

Knowledge testing stage Finally, the third stage is designed to test the acquired knowledge. First, the interaction through the learning tasks is stopped. The students cannot improve their responses from this moment, and no more teacher's feedback is provided. Therefore, the teacher is responsible for marking the students' responses of the learning tasks. Second, the test generation framework is now employed to generate summative assessment tests. Teachers are in charge of defining the assessment objectives which are common to all the students, and setting the time period during which the tests might be performed. Hence, students who are close to master the knowledge of the domain are encouraged to learn the deepest details about the course.

Final student evaluation Concerning the final student evaluation, we encourage the teacher to follow the formative assessment theory. That is, the grade given to each student should be determined in basis of his/her complete actuation within this methodology: grades of initial proposed resources, implication in regard to the self-assessment process, grades of assigned learning tasks, grades of the summative assessment tests, ...

Bearing that in mind, students should take an active part in the whole course, instead of making final effort few weeks before the final course tests. In addition, the teacher's feedback is not only given at the end of the course, like it occurs in traditional learning. Teachers can supervise and guide the whole student performance. Moreover, this model allows to reduce the time gap between student actuation and teacher feedback.

4 Examining the Effects of ivLearn: Cases of Study

The aim of this section is to provide a validation about the effectiveness of our VLE and the regulatory methodology. To that end, we have focused on two indicators which are usually employed in the literature: student satisfaction, and student achievement. Therefore, we have conducted two experimental studies based on the application of our learning method as an addition to the lectures of the undergraduate course "Artificial Intelligence" in the Computer Engineering degree in the University of Granada (UGR) from September 2012 to July 2014. Specifically, the research questions which guided the validation are:

1. Which are the effects of ivLearn regarding to student satisfaction?

2. Which are the effects of ivLearn regarding to student achievement?

It should be pointed out that we conducted a pre-experimental evaluation of our method during the year 2011/12. During that year, ivLearn was in a preliminary version, including only functionality to manage courses, users, quiz questions and assessment tests (i.e., the version included the fuzzy test generation framework). The results of the pre-experimental evaluation were formerly presented in this Thesis, concretely in Chapter IV Section 2.4.

In the following subsections we will discuss the research approach, and the results obtained by the two experimental cases of study.

4.1 Research approach

In order to test the effectiveness of our VLE and the regulatory methodology, we designed two experiments based on the application of ivLearn in the "Artificial Intelligence" course of the UGR. The course is divided into four groups in the organization chart of the Computer Engineering degree. The traditional face-to-face lectures were widened by the use of ivLearn, making up a blended learning scenario. The experiments were performed during the years 2012/13 and 2013/14.

During the evaluation period, ivLearn was stored in a university server under Apache Tomcat 6 (6.0.32) to manage the data storage and was accessible via web application. During the progress of the courses, a technical assistant was available to assure the correct operation of the VLE and to manage the technical issues reported by the users. At the beginning of each course, all participants including both teachers and students were instructed with specific briefing lessons. In addition, teachers were instructed about the stages of the methodology and their role in each one.

The evaluation of ivLearn was performed following an empirical approach in the two cases of study. The students' performance was measured by conducting experiments with a experimental group and a control group. The experimental group followed the learning methodology explained in Section 3.3, with a little variation. The version of ivLearn employed during the experiments did not have integrated the FAQ retrieval module yet. Although the collaborative space was operational, the teachers of the experimental group did not request student to propose frequently asked questions about the course as part of the "concept acquisition" stage. It must be taken into account in order to provide further experimentation in future works. Next, the teacher in the experimental group marked his students by considering their complete actuation within the ivLearn's regulatory methodology. Meanwhile, the control group followed the traditional the conventional methodology. That is, the hours of learning destined to home of the corresponding ETCS of the course were focused on the performance of practical activities common for all students. In addition, students in control group performed a final exam at the end of the course which was equal for all of them. The grades in this case were established in function of the results of the practical activities plus the grade of the final exam. The analysis of the two indicators was made by attending to different information collected from the students in the experimental group, that is discussed in the following.

First, regarding the level of satisfaction, we observed that several strategies were employed in the literature: interviews [MJSIdBC13], attitudinal surveys [JGF09], the technology acceptance model [MTTMG$^+$08], or the IS Success Model [HLP06], among others. Nonetheless, the university develop systematically a procedure with the aim of collecting the student opinions about the main factors involved in each course since 2008. The procedure is carried out by an independent organization, named the Prospective Andalucian Centre (PAC). The PAC performs the analysis by means of satisfaction questionnaires which are completed by the students of each course in the UGR. The questionnaire is composed by 28 items based on a five-point Likert scale regarding learning factors such as the opinion about the teacher, the methodology, the available resources or the subject of the course. Hence, in order to take advantage of this procedure we asked to the PAC for the information referred to our considered courses. Since the questionnaire contains items non-related to the satisfaction level, we enclosed it in a subset of 11 items (Table V.1). Due to the fact that the survey is gathered for each course in the UGR, we compared the opinion of the students in our courses with respect to the opinions of the students in the rest of courses of the Computer Engineering degree.

Items
1. The learning methodology was correctly organized and the involved objectives were suitable
2. The different stages present in the learning methodology were correctly explained at the beginning of the course
3. The learning methodology was correctly fitted for the course's schedule
4. The learning resources present in the system were valuable
5. There was a sufficient number of learning resources
6. The practice activities present in the methodology were useful
7. The practice activities present in the methodology promote collaboration in class
8. I find suitable the factors involved in my evaluation
9. The practice activities present in the methodology helped me to reach the course's objectives
10. The learning methodology enhanced my motivation towards the course
11. In overall, I find the methodology useful in my learning activity regarding this course

Table V.1: Items of the Prospective Andalucian Centre questionnaire regarding the methodology and the learning resources

Finally, the analysis of the student achievement was made by comparing the final grades obtained by the students in the control and experimental group.

4.2 Case of study (year 2012/13)

During the year 2012/13, a control group and a experimental group were taken from the "Artificial Intelligence" course. The sample of participants, the learning context and the obtained results are detailed in the following subsections.

4.2.1 Participants

The sample of this case of study included a total of 105 of undergraduate students, 12 were women and 93 men, aged between 19-26 years. The experimental group was setted up by students of the group D of the course. 65 students were initially enrolled in the group D. From them 55 students (5 women and 50 men) voluntarily decided to take part of this experiment. Meanwhile, the control group was setted up by 50 students (7 women and 43 men) from the rest of the groups who completed the traditional learning methodology. The distribution of undergraduate students by group can be consulted in Table V.2.

	Control Group		Experimental Group		**Overall**	
Gender	*Sum*	%	*Sum*	%	*Sum*	%
Male	43	86.00	50	90.90	93	88.57
Female	7	14.00	5	09.10	12	11.42
Total	50		55		105	

Table V.2: Demographic distribution of participants in control and experimental groups

4.2.2 Learning context

The experimental group taking part in this study applied all the stages of our leaning methodology. During the first stage, each student proposed near to 10 resources on average (M = 10.72, SD = 3.51) for the course. From them, 5.45 were quiz questions (SD = 2.44) and 5.27 were learning tasks (SD = 1.39) on average. In total, 590 resources were generated in the experimental group, 300 were quiz questions and 290 were learning tasks. From them, the teacher considered 50.33% as valid resources with a grade equal or higher than 7 (40.66% quiz questions and 60.34% tasks). Further, in this stage the teacher marked 82.33% of quiz questions and 89.31% of learning tasks with a grade equal or higher than 5. The information about the mean and standard deviation of proposed resources per student, the total number of proposed resources, and the percentage of validated resources by experimental group is shown in Table V.3.

	Experimental Group				
	M	*SD*	*Sum*	*%val*	$\%grade \geq 5$
Quiz questions	5.45	2.44	300	40.66	82.33
Learning tasks	5.27	1.39	290	60.34	89.31
Both	10.72	3.51	590	50.33	85.76

Table V.3: Distribution of proposed resources by experimental group

In the second stage, the self-assessment tool was available. A total of 780 self-assessment tests were done during the second phase of the method. Student performed near to 15 tests on average (M = 14.89, SD = 08.83). As it can be observed, the standard deviation was relatively high pronounced. This is due to the difference in terms of use of those participative users with respect to those who barely used the system. For example, the most participative user performed 44 self-assessment tests. In contrast, 6 students performed less than 5 tests. Accordingly, this circumstance helped us to divide the students and to analyse their achievement in function of their actuation (subsection 4.2.4). Additionally, during the second stage 256 learning tasks were self-assigned by the students (M = 5.15 , SD = 2.26), 6 learning tasks per student were assigned by the teacher.

Lastly, the teacher generated 6 different summative assessment tests. Each summative test had a limit of 3 attempts per student (only the best grade for exam was kept and considered in the final student assessment). The total number of attempts reached to 577. Students were finally assessed in basis of the following criteria:

- Up to 1 point for the proposed quiz questions.
- Up to 1 point for the proposed learning tasks.
- Up to 3 points for the responses to the tasks assigned by the teacher.
- Up to 5 points for the final exams generated by the teacher.

At the end of the course, VLE-users were asked to answer a questionnaire designed to evaluate their level of satisfaction towards the new methodology.

4.2.3 Research results: student satisfaction

As commented before, the PAC questionnaire was available for the students in the experimental group. The fulfilment of the questionnaire is voluntary, according to the policy of the UGR to that end. In consequence, 21 students from the experimental group voluntarily agreed to complete the survey. A summary of the valuations established by them is shown in Table V.4, including the number of students, and the mean and standard deviation of the opinions classified by item.

As it can be observed, the rates are above 4 points in most cases and very similar. The high degree of acceptance is stressed in the eleventh item, which refers to students' opinion about the methodology in overall (M = 4.40, SD = 0.92). Regarding the whole Computer Engineering degree, the methodology obtained a rate that was 13.40% higher than the rate of the degree.

In contrast, the lowest rated statement was the seventh item, which refers to collaboration in class. Although the collaborative space was available in the VLE, this reflects that students understand the sense of collaboration from a problem-based viewpoint. This fact confirmed us that collaboration should be improved in terms of student-student interaction oriented towards problem-based activities.

4.2.4 Research results: student achievement

The student performance was analysed attending to their implication within the methodology. Since self-assessment and self-regulation are viewed as crucial factors for developing this learning method, we carried out an analysis of the student achievement assembled by their intervention in the two first stages of the approach.

All the involved students were classified considering three different variables: proposed resources (stage 1), performed self-assessment tests (stage 2), and self-assigned tasks (stage 2). These variables define the student's degree of usage in the VLE. Consequently, we could observe the level of

Item	Exp. Group (N = 21)	
	M	SD
1	4.05	0.84
2	4.35	0.79
3	4.19	0.79
4	4.15	0.65
5	4.29	0.82
6	4.25	0.62
7	3.81	0.91
8	4.50	0.76
9	4.29	0.57
10	3.90	0.92
11	4.40	0.58

Table V.4: Results of the 11 items of the Prospective Andalucian Centre questionnaire

dedication that each student had towards the methodology by measuring the level of use that they had with the VLE. The teacher established a weighting model to that end, estimating the importance and time-consumption of the activities in each stage. The usage weight for a given student was calculated as: $w(student) = No.proposedresources + No.self - assingedlearningtasks \cdot 1.5 + No.performedself - assessmenttests \cdot 3$. The student with highest usage weight obtained 153. In contrast, the lowest usage weight was of 3. The mean value for all the weights was of 63.2, with an average deviation of 20.34. After that, in order to perform the student classification, the usage weights were used to separate students in 4 types, covering from very-light users (S1) to heavy users (S4). The classification ranges are showed in Table V.5.

Class	Description	Usage weights ranges
S1	Very-light usage	[minimum weight, mean weight - avg. deviation weight) = [03.00, 42.86)
S2	Light usage	[mean weight - avg. deviation weight, mean weight) = [42.86, 63.20)
S3	Regular usage	[mean weight, mean weight + avg. deviation weight) = [63.20, 83.54)
S4	Heavy usage	[mean weight + avg. deviation weight, maximum weight] = [83.54,153]

Table V.5: Ranges for the classification of student by degree of usage

Then, we were be able to compare the grades of the students classified into the four usage groups. The number of students in each class, the mean value and the standard deviation value of the final grades obtained in the experimental group and are shown in Table V.6.

The results shown that students collated in higher usage-classes obtained higher grades, with less rate of standard deviation. This growth is progressive. The mean grades obtained in the S1-class and S2-class are 7.33 (SD = 1.23) and 7.56 (SD = 0.88), respectively. This represents a difference of 03.05% between both values. The mean values for S3-class is 7.88 (SD = 0.34), having a difference with S2-class value of 4.07%. Considering the S4-class, the mean value is 8.23, (SD = 0.36) representing a difference of 4.26% in regard to S3-class. Summarizing, the degree of usage had a critical impact in the student achievement, rising to a improvement of 10.94% comparing the very-light usage students in contrast to the heavy usage students. Additionally, a higher usage of

	Experimental Group		
	N	*M*	*SD*
Overall	55	7.70	0.73
S1	10	7.33	1.23
S2	25	7.56	0.88
S3	10	7.88	0.34
S4	10	8.23	0.36

Table V.6: Distribution of grades obtained in the experimental group by usage classification

the system led to a descent in the standard deviation rate. This fact confirms that the achievement of those student who implied less within the methodology were more erratic. On the contrary, the students in heavy usage class obtained a much more coherent achievement.

Subsequently, we compared the final grades obtained in the experimental group with those obtained in the control group. The mean value and the standard deviation value of the grades from students in control group are detailed in Table V.7. Students obtained a mean grade of 6.29 (SD = 1.96). As it can be seen, it is significantly lower than the mean grade obtained from the VLE-users in higher usage classes. For example, students in S4-class obtained an improvement of 23.58%. This is a clear sign about the benefits of our methodology, compared with a traditional scenario. The three stages of our methodology lead to a less traumatic learning process, helping students to consolidate the knowledge before deal with the final tests. In addition, the grades are not only based on the final tests but also on the whole student actuation. A visual representation of the mean values of final grades classified by level of usage in the experimental group in contrast to the grades obtained in the control group is provided in Figure v.5.

Control Group		
N	*M*	*SD*
50	6.29	1.96

Table V.7: Distribution of grades obtained by students in control group

4.3 Case of study (year 2013/14)

During the year 2013/14, we continued our experimental study taking new control and experimental groups. The sample of participants, the learning context and the obtained results are detailed in the following subsections. Unfortunately, the information related to the satisfaction questionnaire and the final grades of students in control groups was not available at the closing date of this work. Therefore, the discussion of these results will be performed in future works.

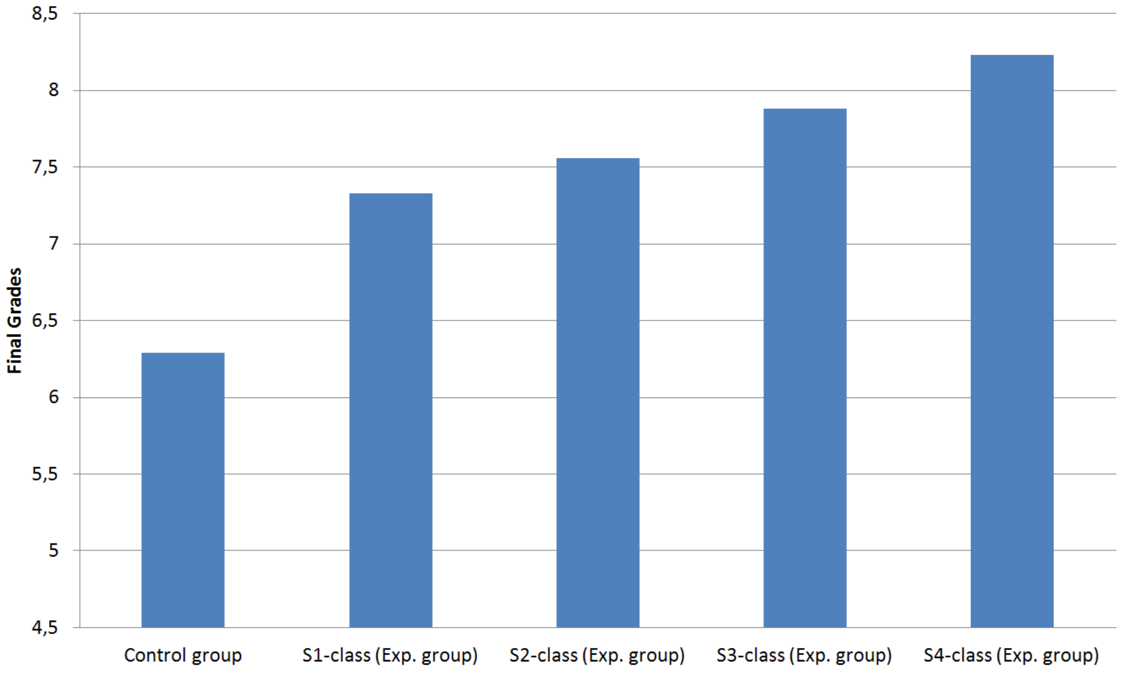


Figure v.5: Bar chart of the distribution of grades

4.3.1 Participants

The sample of this case of study included a total of 103 of undergraduate students, 22 were women and 81 men, aged between 19-26 years. The experimental group was setted up by students of the group C of the course. 65 students were initially enrolled in the group C. From them 46 students (6 women and 40 men) voluntarily decided to take part of this experiment. Meanwhile, the control group was setted up by 50 students (9 women and 41 men) from the rest of the groups who completed the traditional learning methodology. The distribution of undergraduate students by group can be consulted in Table V.8.

	Control Group		Experimental Group		**Overall**	
Gender	*Sum*	%	*Sum*	%	*Sum*	%
Male	41	82.00	40	86.95	81	84.37
Female	9	18.00	6	13.05	15	15.63
Total	50		46		96	

Table V.8: Demographic distribution of participants in control and experimental groups

4.3.2 Learning context

The experimental group taking part in this study applied all the stages of our leaning methodology. Nevertheless, during the first stage, the teacher in charge of the group only requested student to propose learning tasks. The decision was made attending to the volume of quiz questions contained for the course. Since the created learning resources for the “Artificial Intelligence” course from former years are kept in the system, the teacher decided that the resource base was completed enough.

Then, during the first stage, each student proposed 4.00 learning tasks on average (SD = 3.14). In total, 184 resources were generated in the experimental group. From them, the teacher considered 85.16% as valid learning tasks with a grade equal or higher than 7. Further, the teacher marked 99.45% of learning tasks with a grade equal or higher than 5.

In the second stage, the self-assessment tool was available. A total of 976 self-assessment tests were done during the second phase of the method. Student performed near to 21 tests on average (M = 21.21, SD = 19.02). As it occurred during the former experiment, the standard deviation was highly pronounced. In this way, the most participative user performed 95 self-assessment tests in contrast to 4 students who performed less than 5 tests. Additionally, during the second stage 3 learning tasks per student were assigned by the teacher. Only 2 students decided to auto-assign learning tasks, choosing 2 tasks each one.

Lastly, the teacher generated 3 different summative assessment tests. Each summative test had a limit of 3 attempts per student (only the best grade for exam was kept and considered in the final student assessment). The total number of attempts reached to 213. Students were finally assessed in basis of the following criteria:

- Up to 2 point for the proposed learning tasks.
- Up to 3 points for the responses to the tasks assigned by the teacher.
- Up to 5 points for the final exams generated by the teacher.

At the end of the course, VLE-users were asked to answer a questionnaire designed to evaluate their level of satisfaction towards the new methodology. Unfortunately, the results were not available at the closing date of this work.

4.3.3 Research results: student achievement

We conducted the same evaluation scheme carried out in the previous experiment. All the involved students were classified considering the three following variables related to the usage of the VLE: proposed resources (stage 1), performed self-assessment tests (stage 2), and self-assigned tasks (stage 2). Thence, a usage weight was provided for each student in function of the following formula: $w(student) = No.proposedresources + No.self - assingedlearningtasks \cdot 1.5 + No.performedself - assessmenttests \cdot 3$. The student with highest usage weight obtained 294. In contrast, the lowest usage weight was of 6. The mean value for all the weights was of 67.78 with an average deviation of 39.48. After that, in order to perform the student classification, the usage weights were used to separate students in 4 types, covering from very-light users (S1) to heavy users (S4). The classification ranges are showed in Table V.9.

Subsequently, the classes were employed to compare the grades of the students in the experimental group in function of their level of usage of the VLE. The number of students in each class, the mean value and the standard deviation value of the final grades obtained in the experimental group and are shown in Table V.10.

The results are consistent with those obtained in the previous year. The performance of the students in the “very-light usage” class was worse than the performance of students in the same

Class	Description	Usage weights ranges
S1	Very-light usage	[minimum weight, mean weight - avg. deviation weight) = [06.00, 28.30)
S2	Light usage	[mean weight - avg. deviation weight, mean weight) = [28.30, 67.78)
S3	Regular usage	[mean weight, mean weight + avg. deviation weight) = [67.78, 107.26)
S4	Heavy usage	[mean weight + avg. deviation weight, maximum weight] = [107.26,294]

Table V.9: Ranges for the classification of student by degree of usage

	Experimental Group		
	N	*M*	*SD*
Overall	46	7.12	1.50
S1	10	4.96	1.31
S2	21	7.31	1.26
S3	8	8.36	0.72
S4	7	8.19	0.63

Table V.10: Distribution of grades obtained in the experimental group by usage classification

class in the previous year. Nevertheless, students in higher-usage classes obtained the higher grades, with less rate of standard deviation. The difference between the grades obtained in S1-class and S4-class was of 39.44%. As it occurred the previous year, a higher usage of the system led to higher grade. Additionally, the standard deviation rate also decreased with increasing system usage.

It should be pointed out that grades of the students in control groups were not available at the closing date of this work. In consequence, the comparison and the subsequent analysis of the grades in control and experimental groups will be discussed in future works.

5 Final Discussions and Future Work

In this chapter, we have investigated how to develop a complete virtual learning environment and how to integrate it in a real higher education engineering course. To that end, we have firstly studied a number of pedagogical factors that should be taken into account in order to improve the effectiveness of e-learning systems, including student self-regulation, student interaction, the benefits of electronic assessment methods as well as student self-assessment, the effects of formative and summative feedback, and the integration of the VLE in a blended environment. The analysis of such factors have allowed us to design a complete e-learning environment that not only integrates the intelligent modules designed along this work, but also takes into account the real necessities of teachers and students. What is more, we have designed a regulatory methodology which considers the implantation of the VLE in a engineering course.

The architecture of our e-learning system is modular, favouring the cohesion between the logical and physical design, guaranteeing the balance of the levels of granularity and aggregation, and allowing an easy modification of the scheme. It operates with independence of the domain, and supports the creation of any kind of course from scratch. The system includes a variety of intelligent functions which are useful for teachers and students. The system facilitates to work with internal representations of the knowledge of the learning domain, regardless of the users' technological expertise. The knowledge is extracted by the intelligent modules with minimum human intervention. Then the system allows the creation of navigational, search, and indexation tools which improve the understanding of the domain, and facilitate the interaction with the educational content. Next, the system implements a collaborative protocol in charge of the construction of the learning content, which is not exclusive responsibility of the teacher. The VLE includes e-assessment mechanism which can be used by students and teachers, and a intelligent tutor based on e-assessment. On the one hand, this encourages students to carry out a self-assessment process. On the other hand, it allows teachers to create assessment resources automatically, in case of the necessity of summative assessment. The information related to the students' actuation is always available. Finally, the interaction between students is fostered by means of a collaborative space.

The regulatory methodology is defined in basis of the learner-centered paradigm: students should be participative actors, with control over the learning process. It it is focused on three fundamental stages: the domain concept acquisition, where students should understand the main topics of the course; the knowledge expansion stage, where students should expand and increase the previously obtained knowledge; and the knowledge testing stage, where students should test the acquired knowledge. The methodology considers student self-regulation as base of the learning process. That is, the stages are flexible and well-defined. The students are responsible for deciding when they should perform an activity or how many activities they should perform. For example, during the second stage, students are encouraged to perform self-assessment tests. However, since no imposition is made, they could perform 100 tests or none without affecting this to the student final grade. Besides, the methodology is consistent with the constructivist paradigm. It is originally designed to support higher education engineering courses, although it could be applied in other domains with minimum complexity.

We have tested our method on the "Artificial Intelligence" course in the Computer Engineering degree in the University of Granada. Two cases of study carried out in two consecutive years were discussed. Our principal goal was to assess the improvement in student satisfaction and student achievement that our methodology can produce in higher education. To that end, we employed an empirical approach by conducting experiments with experimental groups and control groups. First, the student satisfaction after the experimental periods was measured by means of a satisfaction questionnaire, which is actually part of the policy of the University of Granada. The high rates obtained revealed that users felt comfortable with our model. What is more, the rates obtained were higher than the rates of the rest of the courses that belong to the same degree. Second, we compared the grades of the students in the control group and the experimental group after the experimental periods. The students involved within our methodology were classified attending to their degree of usage regarding the VLE. Those students collated in higher usage-classes obtained the higher grades. Consequently, the degree of usage had a critical impact in the final student achievement. Furthermore, the grades of the students in control group were significantly lower than those obtained from the VLE-users in experimental group.

Nevertheless, we found that students preferred other kind of interaction between them. Although the methodology includes a collaborative space between students, it is no focused on the sense of cooperation. The lowest rate obtained by the item related to this aspect in the satisfaction survey reflected that students had other expectations in terms of collaboration. Therefore, we believe that is important to design a cooperative scenario, attending to the results of problem-solving methodologies that obtained very positive results [LR10, Ste09].

Chapter VI

Conclusions and Future Work

This PhD dissertation was devoted to offer an study on intelligent techniques that have the aim of improve the performance of current e-learning systems. This final chapter is to offer our main conclusions and to outline the avenues of our future research.

1 Conclusions

Following the methodology proposed in Chapter I Section 4, our work has been performed in two main stages. During the first stage, we classified the intelligent techniques into three blocks. First, we addressed the task of acquiring and organizing the knowledge underlying in the educational content. Second, we considered the processes regarding information retrieval and visualization of the learning domain. Third, we focused on the methods able to adapt the learning environment. During the second stage, we designed a virtual learning environment including the prior intelligent modules.

Then, we will first present our main conclusions on each block of the first stage, and on the second stage. After that and for the sake of consistency, the global conclusions of this Thesis will be presented.

1.1 Knowledge Extraction and Organization Methods for the Educational Content

The main goal of this block concerned with the extraction and organization of the knowledge from the educational content. The interest of this problem is justified by the fact that current e-learning systems might present information overload states due to the large volume of available resources, provoking this student's disorientation and cognitive overhead. To our aim, we focused on knowledge acquisition and organization methods able to create meta-data structures from scratch. Concretely, we handled the creation of taxonomies, folksonomies, and ontologies. Each structure presents a different level of semantic richness, and hence a different technique for its composition was required.

First, taxonomies are hierarchical classifications of elements. This kind of structure allows systems to provide resource indexes in a straightforward manner. We handled the creation of flat taxonomies by means of an AKE method. This kind of methods automatically collects the most representative term from a document collection. Unfortunately, traditional AKE methods do not consider the particularities of the educative domain. That is, they operate in a multi-domain manner, but they do not usually detect specific terms of the educative domain. This specific terms do not present significant statistical distributions, nor are included in multi-domain controlled vocabularies. In consequence, we developed a specific method taking advantages of Wikipedia and a frequency-in-language dictionary as knowledge bases. Then, the method links the key terms with the resources from which they were extracted and establishes an organization of the educational content. Our proposed AKE algorithm obtained the best results in term of $F - measure$ against state-of-the-art methods. Nonetheless, the results should be taken carefully. AKE methods produce a high rate of wrong results due to the inner difficulty of dealing with natural language texts. Our method suffers from the same drawback, although it outperforms the rest in the learning domain. Therefore, we recommend a manual revision of the results after the execution of the method.

Second, folksonomies are non-hierarchical organizations of web elements by means of user selected tags. The structure is collaboratively enhanced by all the members of a virtual community. Tag recommendation systems are usually employed in this context, having the aim of helping users to tag the web elements. Nevertheless, TR are not usually used in the e-learning context. Additionally, several of these methods require from prior knowledge modelling. This limit the scope of application in initial phases above all, since the user are forced to have technological expertise. For these reasons, we proposed a TR method for recommending tags from learning resources without knowledge modelling. In addition, it presents a important novelty. Our method recommends conceptual tags instead of simple tags. The tags selected by the users next to the learning resources form a conceptually extended folksonomy. We took advantage of Wikipedia as knowledge base. Our method was contrasted against a baseline method obtaining the best results in terms of $F - measure$. In addition, we presented the method in a real learning environment in order to provide an usability study. Results shown that our method was well received.

Third, ontologies are formal conceptualizations of the knowledge and the relations of a given domain. The (semi)automatic learning of ontologies has received a great deal of attention given the large number of potential applications as knowledge base. There exist several approaches to that end. Unfortunately, most of the proposals do not consider the extraction of educational domain relations. For this reason, we developed an semi-automatic method dealing with the extraction of educational domain lightweight ontologies. The method is able to extract concepts and relations, and recommend them to the teacher/manager. More concretely, the main feature of our proposal consists of the extraction of educational domain relations: subtopic relations, subordinate relations, and content-based relations. Complementary, our method also extracts taxonomic relations: superclass relations and subclass relations. We followed the common approach of the literature in order to validate the method. We established a validation on the manual side, using a manually labelled dataset to that end. Additionally, we conducted an usability study by collecting the opinions of a professor of an engineering course. First, the results of the manual validation shown that the performance of our method is significantly high in terms of $F - measure$, regarding the extraction of educational domain relations. However, it offered poorest results regarding the extraction of taxonomic relations. This fact was also reflected in the usability study. The teacher involved in that study positively appreciated our method. He stated that the application seems adequate and helpful for organizing the course content in an ontology structure. Additionally, he stated that the extraction process regarding the taxonomic relations should be improved.

1.2 Information Retrieval and Visualization Methods for the Educational Domain

We devote our second block to address the problems of information retrieval and information and knowledge visualization in the leaning domain. This problem is of great importance insofar the search of information as well as the visualization of the domain are able to solve the problems derived from the information overload.

First, we designed a FAQ retrieval method able to extract precise information stored in the system from natural language user queries. The FAQ retrieval field is being increasingly considered in e-learning. Examining the approaches in the literature, we can classify them into two major groups: those who need knowledge modelling, and those who do not. The first ones offer fitted results, but they require from prior knowledge modelling. In contrast, the second ones are simpler, but they do not take into account the semantics of words. Moreover, the internal representation of the documents are hardly interpretable, due to they are created by means of statistical analysis. For these reasons, we have developed a method that firstly obtains the semantic intern representation of the document collection in a automatic way. The inter representation is easily interpretable and extendible. Our method obtained the best results in terms of *Accuracy* and *MRR* against state-of-the-art methods in our experiments. It should be pointed out that we focused on the FAQ document context, since this kind of documents is receiving a great deal of attention by the research community as learning resources. In any case, our proposal is domain independent and can be easily used with other kind of datasets.

Second, we explored two different methods which combine the features of Knowledge Visualization and Information Visualization techniques. Therefore, our methods are able to represent the knowledge of the domain in a visual manner, at the same time that such representation serves as navigable structure of the educational content. The two implemented methods are: tag clouds, and concept maps.

Tag clouds are visual representations of tags. The tags are linked to the learning resources in the learning environment. Therefore, this technique allows students to perform a shallow exploration of the domain. Furthermore, it serves as method of organization of the educational content trough the navigational representation. Our method obtained the best results in terms of *Coverage*, *Overlap*, and *Selectiviy* against state-of-the-art methods. Additionally, we presented two application frameworks of this model, by integrating it into the FAQ retrieval system and the Tag Recommender system commented above. It should be remarked that in this work we have not paid attention to the impact of the layout features of tag clouds. This task is programmed to future works.

Concept maps are visual representations of the concepts and relations of a domain. As it stands for tag clouds, concept maps serve as domain comprehension tool, as well as navigational and organization tool for the educational content. Because of the similarities found between concept maps and ontologies, our proposal consisted of a straightforward translation of lightweight educational domain ontologies. The method was validated by means of an usability study. To that end, it was presented to a group of students belonging to a engineering course. Students coincide on the opinion that the method is useful for the understanding of the domain, and for the organization of the educational content. Finally, we proposed the integration of this method into the ontology learning system commented above.

1.3 Intelligent Adaptive Methods for Educational Systems

The aim of the third block concerned with the automatic adaptation of the virtual environment. This kind of methods are receiving a great deal of attention due to their ability to solve some problems associated to classic e-learning systems. On the one hand, adaptive strategies foster the access and comprehension of the information in information overload scenarios. On the other hand, classic e-learning systems are static, i.e. they offer the same content to all the students regardless of their different learning goals or their different knowledge states. Concretely, we have analysed the field of Intelligent Tutoring Systems in order to provide a method able to adjust the environment in function of the individual student's characteristics and knowledge state. Classical ITSs obtains a representation of the student's knowledge state by means of a diagnosis process based on manually created tests in most cases. This provokes that the teacher/manager should create diagnosis tests fitted to all possible cognitive states of the students. Therefore, our method is able to provide fitted diagnosis tests automatically.

In first place, we designed an intelligent method for automatic generation of tests. It is based on Computerized Adaptive Tests. This kind of systems provides students with tests that are adapted in function of the given previous responses. However, these systems do not consider the benefits of self-assessment, a framework in which the students can decide which are their learning objectives. In our aim, our method generates assessment tests in function of assessment objectives explicitly setted by the user. We validated our method by means of a satisfaction study after the implantation of it in a engineering course during a complete academic year. Our proposal obtained a very positive valuation from the students and the teacher of the course.

Subsequently, we employed the test generation method in order to design a intelligent virtual tutor based on e-assessment. The system generates diagnosis tests automatically in basis of the current knowledge state of each student. The assessment tests are related to the concept of the course. Hence, students are able to consult in which concepts they are presenting more difficulties, to explore the learning resources corresponding to such concepts, and to perform a subsequent assessment process until they reach their learning objectives. In addition, the virtual tutor presents game-like features, such as game-like stages, or interactive progress bars. As validation approach, we carried out a usability study by presenting the virtual tutor to a group of students of an engineering course. Although the results were encouraging, the method should be tested more carefully. To that end, we planned to apply the system as an addition of an engineering course during complete academic years.

1.4 Integrating the Intelligent Modules into a real Virtual Learning Environment

During the second stage of our methodology, our goal was two-fold. First, we defined the architecture and functionality of a VLE including the intelligent modules. After that, we defined a methodology that regulates the use of the VLE as part of a higher education engineering course. In order to reach those goals, we firstly analysed a number of pedagogical factors which affect to the performance of e-learning systems: student self-regulation, student-student interaction, student-teacher interaction, student-content interaction, e-assessment, student self-assessment, formative/summativ feedback, and blended environments. Then we established a set of design requirements and defined the architecture of the VLE. Furthermore, we specified a usage regulatory methodology for the VLE, as part of higher education engineering course under the Bologna plan.

Among the design characteristics of the VLE, we proceed to comment the following. Our VLE is modular, helping this to an easy maintaining and updating. The VLE is domain independent, allowing the creation of any course from scratch. Most of intelligent modules operate automatically, and the rest do with minimum human intervention. Thus, users of all kind would be able to use the system regardless of their technological expertise. The VLE implements a model of collaborative construction of the educational content. Next, each intelligent method is embedded as an independent module, which receives and sends the data flow as a black box. That is, both the parameter configuration and the automatic learning stages (if it is necessary) are performed through the user interface. Lastly, the system stores automatically information related to each student: proposed resources, assessment results, access logs, ...

In regard to the usage regulatory methodology of the VLE, is is framed on the learner-centered paradigm (one of the requirements imposed by the Bologna plan). The student should be the main actor of the learning process, keeping control of it, and being actively involved in it. In contrast, the teacher takes the role of guide or supervisor of the process. Our methodology is composed by three main stages: domain concept acquisition, knowledge expansion, and knowledge testing. It has as cornerstone the concept of student self-regulation. That is, students are responsible of deciding how much time and effort they spent at each stage. Finally, the methodology follows the constructivist paradigm in regard to the collaborative construction of the educational content. The methodology is not fixed to any specific domain and it can be applied to any course.

We validated the VLE by means of two empirical experiments realised in a real engineering course. During two consecutive academic years, we took a control group and a experimental group. Then we analysed the impact of the application of the VLE (regulated by the methodology). More concretely, we studied the impact of the VLE in terms of student satisfaction and student achievement, the two cognitive indicators most employed in the literature. First, we measured the student satisfaction by means of a satisfaction questionnaire fulfilled by the students after the end of each course. Results shown that students felt satisfied with the VLE. Nonetheless, we found that students considered the student-student interaction scheme insufficient. This could be solved by including a cooperative problem-solving space. Second, we compared the grades of the students in control and experimental groups in order to deduce the level of student achievement. Grades of students in control group were significantly lower than grades obtained in the experimental group. We also established a classification of the students in the proper experimental group in function of their level of usage of the VLE. Results shown that more implied students within the methodology obtained the best grades. Thus, our proposal has a critical impact in the level of student achievement.

1.5 General Conclusions

The study of CI techniques as support mechanisms of the e-learning systems seems to be extremely beneficial. In such context, our hypothesis is that the e-learning systems could be benefited from the application of specific techniques which allow to obtain, organize and apply the knowledge underlying the educational content, having the purpose of improving the learning processes. Our main goal was to offer specific solutions to those tasks, and to integrate them into a complete VLE. To that end, we have classified those tasks into three blocks: knowledge extraction, retrieval and visualization of information and knowledge, and adaptation of the learning environment. Then, we have integrated the obtained methods into a complete VLE. Our main conclusions in this regard could be summarized as follows:

- The construction of meta-data structures from the educational content becomes more complex as the level of semantic richness increases. Therefore, more human intervention is needed if the level of semantic richness is higher. Additionally, the exploitation capability of the structure is also increased as much semantic richness it holds. Concretely, starting from simple indexations of resources, the VLE offered visual and navigational conceptual representations of the domain.

- The techniques based on information retrieval and on information and knowledge visualization may solve the problems derived from the information overload. This kind of tools foster the direct search of resources, the comprehension of the learning domain, and the interactive exploration of the educational content.
- The techniques related with the adaptation of the virtual environment allow to develop personalized learning plans in function of the individual student's characteristics. This improves the performance of the VLE in comparison with static environments. Nevertheless, counting with mechanisms able to infer the cognitive state of the students is mandatory. The complexity of such inference process depends on the complexity of the domain.
- Wikipedia seems as a excellent knowledge base for supporting information extraction processes in the educative domain. Its diffusion is fairly justified, since with minimum treatment the knowledge base can be exported to common storing systems: databases, XML documents,... Once the knowledge base is processed, it is able to support the majority of the information extraction mechanisms. It takes the form of dictionaries, controlled vocabularies, lexicons, etc. Apart from the benefits of its shareability and easy-maintaining, the main benefit of Wikipedia is that its content is in constant evolution. As main drawback, we found an excessively ambiguous organization of the Wikipedia articles into categories. This complicates the exploration of the category tree of Wikipedia.
- Constructing a VLE requires from a prior study of a number of pedagogical factors. Thus, the effectiveness of the system is favoured in function of the considered pedagogical factors. Concretely, student self-regulation is a key factor considering the application of VLEs in the context of the Bologna plan. Precisely, our system is designed to promote self-regulation, that facilitates the learning processes for teachers and students. The organized and friendly design of the system, as well as the rich and complete educational experiences that are provided, may help teachers to promote the degree of motivation and participation of the students.

This Thesis results from an engineering process of scientific nature. From the point of view of the engineering process, this work has resulted in various computer applications of commercial interest. Considering each intelligent mechanism proposed here independently, they could be integrated in VLEs of all kind. Furthermore, this work has lead to the creation of the ivLearn system, a modular system presenting broad educational functionality.

From the scientific point of view, this work has resulted in various publications in international journals as well as in scientific conferences (Chapter VII).

2 Future Work

We have already outlined our future work and research interests throughout each chapter. We thus will here highlight only the most imminent and interesting research to our eyes.

First, we pay attention to the learning resources supported by ivLearn. In this work we have focused on textual resources, leaving aside multimedia resources such as videos, audio, ... The exploitation of this kind of resources would require a deeper comprehension of the knowledge extraction techniques.

Second, the collaborative scheme between students considered here should be reviewed in basis of two different scopes. On the one hand, it might be necessary to take into account the collaboration scheme imposed by the Web 2.0. Collaborative environments of this type include methods of communication between users (forums, chats, e-mails, etc.), integration with poplar social applications (Twitter, Youtube,. . .). In general, they offer mechanisms which favours the view of the students not only as consumer but also as producer of the educational content [SH08]. Recent researches in the field suggest that the presence of collaborative tools enhances the student's motivation and participation. What is more, this is reflected in a enhancement of their achievement [MJ10, CGPRC+14, Bro10]. On the other hand, it might be necessary to include cooperative spaces for problem-solving. The cooperative resolution of tasks seems to improve the analytical and critical competency of students [LNH+13], as well as their degree of motivation and participation [Pan99]. In addition, the presence of cooperative spaces seems to improve the achievement, productivity, social competence, and self-esteem of students [LG12]. Although our proposal includes a collaborative space for students, it seems to be insufficient in comparison with the above mentioned characteristics. Therefore, we plan a detailed study of the collaboration schemes as future work.

Chapter VI

Conclusiones y Trabajos Futuros

Este trabajo de Tesis ha servido para ofrecer un estudio sobre una serie de técnicas inteligentes que tienen como objetivo mejorar el desempeño de los sistemas e-learning actuales. Dedicaremos esta última sección a exponer y comentar las principales conclusiones que se derivan de este proyecto y a esbozar lo que representarán nuestros futuros objetivos.

1 Conclusiones

De acuerdo a nuestra metodología (Capítulo I Sección 4) nuestro trabajo se ha desarrollado en dos etapas principales. Durante la primera etapa, se propone la clasificación de las técnicas inteligentes en tres bloques diferenciados. En primer lugar, abordamos la adquisición y organización inteligente del conocimiento subyacente en el contenido educativo. En segundo lugar, consideramos la recuperación de información y representación visual del dominio didáctico. Finalmente, examinamos los métodos de adaptación del entorno didáctico. Durante la segunda etapa, nos centramos en el desarrollo de un sistema de apoyo al aprendizaje completo que incluye los mecanismos inteligentes desarrollados.

Por tanto, en esta sección discutiremos las conclusiones extraídas tras la realización de cada bloque correspondiente a la primera etapa, y tras el desarrollo y evaluación correspondiente a la segunda etapa. Por coherencia, expondremos en último lugar las conclusiones generales que se derivan de esta Tesis.

1.1 Métodos de Extracción y Organización del Conocimiento para el Contenido Didáctico

En este bloque, nos marcamos como objetivo desarrollar diferentes estrategias de extracción y organización del conocimiento a partir del contenido didáctico. La necesidad de contar con herramientas de este tipo aparece ante la posible sobrecarga de información existente en los entornos e-learning convencionales, lo cual puede provocar desorientación y desánimo de los estudiantes hacia el proceso de aprendizaje. Para solventarlo, nos centramos en la adquisición inteligente del conocimiento presente en el contenido didáctico, y en su organización en tres estructuras diferentes de meta-datos: taxonomías, folksonomías y ontologías. Cada estructura es capaz de contener información con diferente riqueza semántica, por lo que es necesario abordar su construcción de forma independiente.

En primer lugar, las taxonomías son clasificaciones jerárquicas de elementos. Un sistema que cuente con este tipo de estructura puede proporcionar índices de recursos de forma sencilla. Teniendo este objetivo, en nuestro caso nos centramos en la extracción de taxonomías planas, donde todos los elementos están al mismo nivel. Para su extracción automática, hemos abordado el campo de la extracción automática de palabras clave. Este tipo de métodos recolecta automáticamente los términos más importantes de una colección de documentos. Sin embargo, los métodos tradicionales no están desarrollados para trabajar específicamente en el dominio educativo. Es decir, operan normalmente en multi-dominio, pero no suelen detectar términos específicos del dominio educativo que no presentan características estadísticas o distributivas especiales, y/o no están incluidos en vocabularios controlados multi-dominio. En consecuencia, hemos desarrollado un método específico para el ámbito educativo, empleando Wikipedia y un diccionario de frecuencias del lenguaje como bases de conocimiento. Además de términos clave multi-dominio, nuestro método obtiene términos clave específicos del dominio, sin necesidad de entrenamiento o conocimiento específico previo. Tras la extracción de términos clave, el método enlaza dichos términos con los recursos a partir de los que se han extraído, estableciendo así una organización del contenido didáctico. El algoritmo propuesto obtuvo importantes mejoras en términos de $F-measure$ con respecto a otros algoritmos de extracción de palabras clave en nuestros experimentos. Sin embargo, estos resultados deben ser considerados cuidadosamente. La mayoría de los métodos de este tipo obtienen un porcentaje alto de elementos erróneos en sus salidas. Nuestro método, aun mejorando al resto, también adolece de esta desventaja. Por tanto, se recomienda una revisión manual de los resultados tras su ejecución.

En segundo lugar, las folksonomías son organizaciones no jerárquicas de elementos web mediante etiquetas establecidas por usuarios de forma colaborativa en comunidades virtuales. Los sistemas de recomendación de etiquetas se emplean en este tipo de entornos para facilitar el etiquetado de los recursos. Sin embargo, su difusión no alcanza el ámbito e-learning. Además, muchos de los métodos clásicos requieren conocimiento de dominio previo para su funcionamiento, lo que obliga a los usuarios a tener un conocimiento tecnológico adecuado. Esto limita la operabilidad del sistema, sobre todo en fases iniciales. En nuestro caso, hemos propuesto un método de este tipo capaz de operar sin conocimiento previo y que incorpora una importante novedad. En lugar de etiquetas simples (es decir, términos), nuestro método es capaz de recomendar etiquetas conceptuales. El conjunto de etiquetas conceptuales seleccionados por la comunidad, junto con los recursos educativos a partir de los que se extraen, forman una folksonomía extendida conceptualmente. De nuevo, nos valemos de Wikipedia como base de conocimiento del sistema. El sistema propuesto fue contrastado contra un algoritmo base, obteniendo los mejores resultados en términos de $F-measure$. Además, sometimos nuestra propuesta en un entorno educativo real par así realizar un estudio de utilidad. Los resultados obtenidos demostraron que el método es satisfactorio dentro del ámbito educativo.

Por último, las ontologías son conceptualizaciones formales del conocimiento y de las relaciones de un dominio dado. El aprendizaje (semi)automático de ontologías es un campo de gran difusión dentro de la comunidad investigadora. Existen una gran variedad de propuestas desarrolladas para este fin. Sin embargo, las propuestas de la literatura no se centran en la extracción de relaciones específicas del dominio educativo. Por este motivo, nosotros hemos desarrollado un método semi-automático de extracción de ontologías de dominio educativo y ligeras. El método es capaz de extraer conceptos y relaciones entre conceptos, y recomendarlos al profesor, o administrador del sistema educativo. En concreto, la principal característica de nuestro modelo consiste en la extracción de relaciones de dominio educativo: relación de subtema, relación de subordinación y relación basada en el contenido didáctico. De forma complementaria, también extrae relaciones de tipo taxonómico: relación de superclase y relación de subclase. El sistema propuesto fue validado manualmente, utilizando para ello un conjunto de datos etiquetado a mano como referencia, siguiendo la pauta habitual del estado del arte. Al mismo tiempo, llevamos a cabo un estudio de utilidad del sistema, para lo cual contamos con la ayuda de un profesor de un curso de ingeniería universitario. En primer lugar, los resultados de la experimentación manual demostraron que nuestro método presenta un muy buen comportamiento en términos de $F - measure$ en el proceso de extracción de relaciones de dominio educativo. Sin embargo, el sistema ofreció peores resultados, considerando en este caso el proceso de extracción de relaciones taxonómicas. Este hecho quedó además reflejado tras el estudio de utilidad. Como segundo método de validación, presentamos nuestro sistema a un profesor voluntario perteneciente al campo de la Inteligencia Artificial. Este profesor valoró muy positivamente la aplicación, considerando que es adecuada y provechosa para la organización del contenido educativo. No obstante, también declaró que la extracción de relaciones taxonómicas debe ser mejorada.

1.2 Métodos de Recuperación de Información, y de Visualización de Información y Conocimiento para el Contenido Didáctico

Durante el segundo bloque, nos marcamos como objetivo la realización de métodos inteligentes de recuperación de información, y de visualización de información y conocimiento para el dominio educativo. Este tipo de mecanismos se muestran clave para solventar los problemas derivados de la sobrecarga de información en entornos virtuales de aprendizaje.

Primero, hemos diseñado un método de recuperación de FAQs, capaz de obtener información precisa a partir de preguntas de usuario formuladas en lenguaje natural. El campo del FAQ retrieval está comenzando a tomar importancia en el ámbito del e-learning. Atendiendo a los métodos de la literatura, se pueden dividir en aquellos que necesitan de modelado previo y aquellos que no. Los primeros ofrecen resultados bien ajustados, pero requiere un proceso manual de creación de conocimiento experto previo. En cambio, los segundos son métodos más sencillos, pero no consideran la semántica de las palabras, lo cual limita su desempeño. Además, las representaciones internas en este caso son difícilmente interpretables, ya que se realizan mediante análisis estadísticos. Por estos motivos, nosotros hemos diseñado un método que obtiene automáticamente representaciones internas semánticas de la colección de documentos que son fácilmente interpretables. En concreto, nuestro método obtuvo importantes mejoras en términos de *Accuracy* y *MRR* con respecto a otros algoritmos de recuperación de FAQs en nuestros experimentos. Cabe destacar que elegimos el ámbito de los documentos FAQs dada su creciente popularidad como recursos de aprendizaje. En cualquier caso, nuestro método es independiente del dominio y puede ser fácilmente aplicable a otros tipos de datos.

Segundo, hemos explorado dos métodos diferentes que combinan las características de las técnicas de Visualización de la Información, y de Visualización del Conocimiento. De esta forma, nuestros métodos permiten representar el conocimiento del dominio de forma visual, al tiempo que dicha representación sirve como estructura navegable del contenido didáctico. Las dos técnicas de visualización implementadas son: nubes de etiquetas y mapas conceptuales.

Las nubes de etiquetas son representaciones visuales de etiquetas, que a su vez funcionan como enlaces a los recursos educativos. Por una parte, esta técnica permite una exploración superficial del dominio. Por otra, sirve como método de organización del contenido educativo a través de la representación navegable. El método desarrollado obtuvo mejoras sustanciales en términos de las medidas *Coverage*, *Overlap* y *Selectivity*, respecto a un conjunto de algoritmos de generación de nubes de etiquetas del estado del arte. De forma adicional, presentamos dos marcos de aplicación de nuestro modelo, integrándolo en los sistemas de recuperación de FAQs y de recomendación de etiquetas comentados con anterioridad. Cabe mencionar que en este trabajo no hemos analizado el impacto que tienen las características de diseño gráfico de las nubes de etiquetas. Por tanto, esta tarea queda programada para trabajos futuros.

Los mapas conceptuales son representaciones visuales de los conceptos del dominio y de sus relaciones. Igual que ocurre con las nubes de etiquetas, los mapas conceptuales sirven como herramienta de comprensión del dominio, y como herramienta de organización y navegación para el contenido didáctico. Dadas las similitudes existentes entre los mapas conceptuales y las ontologías, el algoritmo de generación de mapas conceptuales consiste en una traducción directa de una ontología de dominio educativo ligera. El método fue validado mediante un análisis de utilidad, para lo cual fue presentado a un grupo de estudiantes pertenecientes a un curso de ingeniería. Los estudiantes coincidieron en la opinión de que el método era útil para la comprensión del dominio y para la organización del contenido didáctico. Finalmente, propusimos la integración de este método como parte del sistema de aprendizaje de ontologías comentado anteriormente.

1.3 Métodos de Adaptación Inteligente del Entorno Educativo

El tercer bloque de la primera etapa de la metodología se centró en el desarrollo de métodos de adaptación del entorno virtual. Este tipo de métodos presentan una excelente difusión, debido a su habilidad para solventar los problemas de los sistemas e-learning clásicos. Por una parte, las estrategias de adaptación del entorno favorecen un mejor acceso y comprensión de la información en entornos proclives a la sobrecarga de información. Por otra, los sistemas clásicos son estáticos, de forma que ofrecen el mismo contenido a todos los estudiantes sin considerar la posible variedad de objetivos de aprendizaje, o el estado actual de cada uno. En concreto, hemos analizados el campo de los tutores inteligentes virtuales, para así proponer un método capaz de ajustar el entorno en función de las características y del estado de conocimiento actual de cada estudiante. Los Tutores Inteligentes Virtuales clásicos obtienen una aproximación del estado de conocimiento de cada estudiante mediante un proceso de diagnostico, basado en la mayoría de los casos en la realización de tests creados manualmente. En este sentido, el profesor o administrador del sistema debería crear tests diagnósticos que se ajustaran a todos los posibles estados cognitivos de los estudiantes. Por este motivo, hemos desarrollado un método capaz de proporcionar tests diagnósticos de forma automática.

En primer lugar, y teniendo como objetivo final el desarrollo de un tutor inteligente virtual, hemos diseñado un método inteligente de generación automática de tests. Nuestro método está basado en los sistemas de adaptación de tests. Este tipo de sistemas proporciona a los estudiantes con tests de evaluación que se van adaptando en función de las respuestas previas. Sin embargo, los sistemas de adaptación de tests no tienen en consideración los beneficios de la auto-evaluación, marco en el que son los propios estudiantes los que deciden en cada momento cuáles son sus objetivos de aprendizaje. En nuestro caso, nuestro método genera tests de evaluación a partir de objetivos de aprendizaje definidos explícitamente por el usuario. Para validarlo, realizamos un estudio de satisfacción tras la implantación del mismo en un curso de ingeniería durante un año académico completo. Nuestra propuesta obtuvo una valoración muy positiva por parte de los estudiantes y del profesor del curso.

Posteriormente, utilizamos el método de generación automática de tests para desarrollar un tutor inteligente virtual basado en evaluación electrónica. Nuestro sistema genera tests diagnósticos de forma automática, a partir del estado de conocimiento actual de cada estudiante. Los tests generados guardan relación con los conceptos del curso correspondiente. De esta forma, los estudiantes pueden en cada momento consultar que conceptos les están resultado más complicados, explorar los recursos de aprendizaje asociados a cada concepto, y realizar nuevas evaluaciones hasta alcanzar los objetivos de aprendizaje definidos para el curso. De forma complementaria, nuestro tutor inteligente virtual implementa características comunes de vídeo juegos, tales como la inclusión de fases, o barras de progreso interactivas. Como método de validación, realizamos un estudio de utilidad del sistema, presentándolo a un conjunto de estudiantes de un curso de ingeniera. Aunque los resultados obtenidos fueron satisfactorios, sería conveniente realizar una experimentación más completa, integrando el sistema como parte de un curso de ingeniería real.

1.4 Integración de los Métodos Inteligentes en un Sistema de Apoyo al Aprendizaje Real

Durante la segunda etapa de nuestra metodología, nuestro objetivo fue doble. Primero, definimos la arquitectura y funcionalidad de un sistema de apoyo al aprendizaje completo. Este sistema integra los métodos inteligente desarrollados. Segundo, definimos una metodología que regula el uso del sistema como parte de un curso universitario de ingeniería. Para lograr ambos objetivos, primero analizamos una serie de factores pedagógicos que afectan al desempeño de los sistemas e-learning: auto-regulación del estudiante, interacción estudiante-estudiante, interacción estudiante-profesor, interacción estudiante-contenido, evaluación electrónica, auto-evaluación, dialogo formativo/sumativo, y entornos semi-presenciales. A partir de los factores analizados, establecimos una serie de requisitos de diseño a partir de los cuales realizamos el diseño de la arquitectura del sistema. Del mismo modo, también nos permitieron especificar una metodología de regulación de uso del sistema, como parte de un curso universitario de ingeniería regido por el plan Bolonia.

Entre las características de diseño de sistema más destacables cabe mencionar las siguientes. Nuestro sistema es totalmente modular, lo cual facilita su mantenimiento o modificación. El sistema es independiente del dominio, y permite la creación de cualquier curso desde cero. Los módulos inteligentes operan en su mayoría de forma automática, o con mínima intervención humana. Por tanto, no se requiere conocimiento tecnológico previo, lo cual favorece el uso del sistema por todo tipo de usuarios. También implementa un modelo destinado a la generación colaborativa del contenido didáctico. Cada método inteligente desarrollado está embebido como un módulo independiente, que recibe y envía el flujo de datos como una caja negra. Es decir, tanto la configuración de parámetros como las etapas de aprendizaje automático (si es necesario en cada caso) se realiza a través de la interfaz de usuario. Por último, el sistema almacena automáticamente información relacionada con el estudiante: resultados obtenidos, recursos propuestos, log de accesos, ...

Respecto a la metodología de regulación de uso, se enmarca dentro del paradigma centrada en el estudiante (uno de los requisitos impuestos por el plan Bolonia). El estudiante es el actor principal, que tiene control sobre el proceso educativo y debe estar activamente involucrado en el mismo. En cambio, el profesor toma el papel de guía o supervisor del proceso. La metodología está compuesta por tres etapas: adquisición de los conceptos del dominio, expansión de conocimiento, y prueba del conocimiento adquirido. Tiene como pilar fundamental el concepto de auto-regulación del estudiante. Es decir, los estudiantes son responsables de decidir cuánto tiempo y esfuerzo dedican a cada etapa. Por último, promueve la construcción colaborativa del contenido didáctico, de acuerdo al paradigma constructivista. La metodología esta ideada bajo el marco de ingeniería universitaria. Sin embargo, dado que no se rige por ningún dominio específico, se podría utilizar en cualquier otro curso sin grandes modificaciones.

El sistema se validó mediante dos experimentos empíricos realizados en un curso de ingeniería real. Durante dos años académicos consecutivos, se tomaron un grupo de control y un grupo experimental del curso, y se analizó el impacto derivado de la aplicación del sistema (regulado por la metodología) en el desempeño de los estudiantes. Más concretamente, se analizó el impacto en términos de "satisfacción del estudiante" y "logro del estudiante", los dos indicadores cognitivos que miden la eficacia de sistemas de apoyo al aprendizaje más empleados en la literatura. En primer lugar, el nivel de satisfacción de los estudiantes se extrajo mediante cuestionarios de satisfacción rellenados tras completar el curso. Los resultados obtenidos demostraron que los estudiantes se sienten satisfechos con nuestra propuesta. No obstante, también observamos cierta tendencia a considerar el esquema de interacción estudiante-estudiante relativamente insatisfactorio, lo cual podría ser solucionado mediante la implementación de un espacio de resolución cooperativa de problemas. En segundo lugar, para deducir el nivel de logro comparamos las notas de los estudiantes de los grupos de control y de experimentación. Las notas de los estudiantes en el grupo de control fueron significativamente inferiores a las del grupo experimental. También establecimos una clasificación de los estudiantes dentro del propio grupo experimental, en función del nivel de uso del sistema. Los resultados arrojaron que los estudiantes más implicados en la metodología fueron los que mejores notas obtuvieron, demostrando que la propuesta tiene un alto impacto en el nivel de logro de los estudiantes.

1.5 Conclusiones Generales

El estudio de técnicas de Inteligencia Computacional como modelo de sustento del entorno educativo virtual encierra un enorme potencial de cara a los sistemas de apoyo al aprendizaje. En este contexto, nuestra hipótesis de partida es que los entornos de aprendizaje virtual pueden beneficiarse de la aplicación de técnicas específicas que permitan obtener, organizar y utilizar el conocimiento subyacente al contenido educativo, con el propósito de favorecer el proceso de aprendizaje en este tipo de medios. Nuestro objetivo principal consistió precisamente en proponer soluciones específicas para estas tareas, diseñadas de forma independiente, e integradas finalmente en un sistema completo. Para ello, hemos planteado primero una clasificación de estas tareas en tres bloques: extracción de conocimiento, recuperación y visualización de información y conocimiento y adaptación del entorno. Posteriormente, hemos integrado las técnicas obtenidas durante el desarrollo de cada bloque en un sistema completo. Podemos extraer las siguientes conclusiones generales de este trabajo:

- La construcción de estructuras de meta-datos a partir del contenido didáctico se vuelve más compleja a medida que aumenta el grado de riqueza semántica albergado en la estructura. De esta forma, se requiere mayor intervención del usuario a medida que aumenta la riqueza semántica. Del mismo modo, también aumenta la capacidad de explotación de la estructura. Concretamente, partiendo de indexaciones sencillas de elementos mantenidos por taxonomías, se alcanzan representaciones conceptuales del dominio de forma visual y navegable, gracias a la capacidad de las ontologías.

- Las técnicas basadas en recuperación de información y visualización de información y conocimiento pueden ser de ayuda para solventar los problemas derivados de la sobrecarga de información en entornos de aprendizaje virtual. Este tipo de herramientas favorecen la búsqueda directa de recursos, la comprensión del dominio de conocimiento, y la exploración del contenido didáctico de forma interactiva.

- Las técnicas basadas en la adaptación del entorno virtual en función de las características y/o necesidades del estudiante permiten el desarrollo de planes de estudio personalizados a cada estudiante, lo cual favorece al desempeño del sistema frente a entornos estáticos. Sin embargo, es indispensable contar con mecanismos capaces de inferir el estado cognitivo del estudiante de forma ajustada. La complejidad de dicha inferencia aumenta a medida que aumenta la complejidad del dominio.

- Wikipedia se muestra como una base de conocimiento excelente para el manejo del dominio educativo. Su gran difusión de uso está justificada, ya que mediante un proceso de inicial de extracción de información se puede exportar la base de conocimiento a cualquier sistema de almacenamiento de información: bases de datos, documentos XML, ... Una vez procesada, la base de conocimiento puede ser soportar casi cualquier proceso de extracción de información, tomando la función de diccionarios, vocabulario controlados, lexicones, etc. Aparte de los beneficios de su fácil manteniemiento y de su compartibilidad, el gran punto fuerte de Wikipedia es que su contenido evoluciona a la par que la realidad. Como único punto flaco, destacariamos una categorización de artículos demasiado ambigüa, provocado por la libertad de cualquier usuario de añadir nuevas relaciones entre artículos. Esto dificulta el árbol de categorias de Wikipedia.
- Construir un sistema de apoyo al aprendizaje, independientemente de su ámbito de aplicación, requiere de un estudio previo relativo a los factores pedagógicos que rigen este tipo de sistemas. Así, la efectividad del sistema se ve favorecida de forma proporcional al número de elementos pedagógicos considerados. En concreto, la auto-regulación del estudiante se antoja clave en sistemas de apoyo al aprendizaje enmarcados en el contexto universitario regido por el plan Bolonia. Precisamente, nuestro sistema está diseñado para promocionar esta característica, facilitando la tarea tanto de profesores como estudiantes. Su diseño sencillo y amigable, junto con las experiencias educativas ricas y completas que proporciona, puede ayudar al profesor a promocionar el grado de motivación y participación de los estudiantes.

Con respecto a la vertiente de ingeniería, este proyecto de Tesis ha dado como fruto una serie de aplicaciones informáticas de interés comercial. Desde un punto de vista individual, cada mecanismo inteligente puede ser fácilmente integrado en sistemas de apoyo al aprendizaje de todo tipo. Así mismo, el cómputo global de este trabajo ha dado lugar al sistema ivLearn, un sistema modular con amplia funcionalidad educativa.

Desde el punto de vista de la producción científica, esta Tesis ha dado como resultado la publicación de varios artículos en diversas revistas internacionales y comunicación en congresos de índole científica (capítulo VII).

2 Trabajos Futuros

A lo largo de cada capítulo, y en la sección anterior, hemos ido remarcando los que supondrán nuestros trabajos futuros y líneas de interés a abordar en un futuro. Nos centraremos aquí en remarcar las más inmediatas y de mayor interés desde nuestro punto de vista.

En primer lugar, centraremos nuestra atención en proporcionar una mayor riqueza de recursos educativos. En este trabajo, nos hemos centrado en recursos textuales, dejando de lado recursos multimedia como videos, audios, ... La explotación de este tipo de recursos también requiere de un análisis más amplio de las técnicas de extracción de conocimiento.

En segundo lugar, el esquema de colaboración entre estudiantes considerado en este trabajo debe ser ampliado en base a dos vertientes diferentes. Por una parte, es necesario considerar el esquema de colaboración impuesto por la Web 2.0. Los entornos colaborativos creados en base a este esquema incorporan desde sistemas de comunicación entre usuarios (foros, chats, correos electrónicos) hasta integración con las aplicaciones sociales más populares en la actualidad como Twitter, Youtube, etc. En general, incorporan mecanismos que favorecen la visión del estudiante no solo como consumidor si no también como productor del contenido educativo [SH08]. Investigaciones recientes en este terreno parecen indicar que este tipo de herramientas colaborativas aumentan tanto la motivación como la participación del estudiante, lo cual provoca un aumento en sus logros [MJ10, CGPRC+14, Bro10]. Por otra parte, es necesario incluir espacios de cooperación entre estudiantes. La resolución cooperativa de tareas parece mejorar la capacidad analítica y crítica del estudiante [LNH+13], y su grado de motivación y participación [Pan99]. También parece favorecer el nivel de logros y la productividad del estudiante, así como su competencia desde un punto de vista social y su autoestima [LG12]. Si bien es cierto que nuestra propuesta incluye un entorno de colaboración entre estudiantes, se antoja simple en comparación con las características mencionadas. Por tanto, planificamos un estudio pormenorizado de los esquemas de colaboración existentes como trabajo futuro.

Chapter VII

List of Publications: Submitted, Published, and Accepted Articles

Publications		
	Accepted	Submitted
Journal	4	1
Conference	1	0

- M. Romero, A. Moreo, J.L. Castro, J.M. Zurita, Using Wikipedia concepts and frequency in language to extract key terms from support documents, Expert Systems with Applications, Volume 39, Issue 18, 15 December 2012, Pages 13480-13491, ISSN 0957-4174, 10.1016/j.eswa.2012.07.011.
    - Status: **Published**.
    - Type: Journal Article.
    - Impact Factor (JCR 2013): 1.965.
    - Subject Category: Computer Science, Artificial Intelligence. Ranking 30 / 121 (Q1).
    - Subject Category: Engineering, Electrical & Electronic. Ranking 63 / 247 (Q2).
    - Subject Category: Operations Research & Management Science. Ranking 11 / 79 (Q1).
    - Correspondence with: Chapter II, section 2

Abstract: In this paper, we present a new key term extraction system able to handle with the particularities of "support documents". Our system takes advantages of frequency-based and thesaurus-based approaches to recognize two different classes of key terms. On the one hand, it identifies multi-domain key terms of the collection using Wikipedia as knowledge resource. On the other hand, the system extracts specific key terms highly related with the context of a support document. We use the frequency in language as a criterion to detect and rank such terms. To prove the validity of our system we have designed a set of experiment using a Frequently Asked Questions (FAQ) collection of documents. Since our approach is generic, minor modifications should be undertaken to adapt the system to other kind of support documents. The empirical results evidence the validity of our approach.

- M. Romero, A. Moreo, J.L. Castro, A Cloud of FAQ: A Highly-Precise FAQ Retrieval System for the Web 2.0, Knowledge-Based Systems, Volume 49, September 2013, Pages 81-96, ISSN 0950-7051, http://dx.doi.org/10.1016/j.knosys.2013.04.019
  - Status: **Published**.
  - Type: Journal Article.
  - Impact Factor (JCR 2013): 3.058.
  - Subject Category: Computer Science, Artificial Intelligence. Ranking 15 / 121 (Q1).
  - Correspondence with: Chapter III, section 2, and section 3

Abstract: FAQ (Frequency Asked Questions) lists have recently attracted increasing attention for companies and organizations as a way to other a trusted and well-organized source of knowledge. There is thus a need for high-precise and fast methods able to manage large FAQ collections. In this context, we present a new FAQ retrieval system as part of a FAQ exploiting project. Following the growing trends towards Web 2.0, our goal is to provide users with mechanisms to navigate through the domain of knowledge and to facilitate both learning and searching, beyond classic FAQ retrieval algorithms. To this purpose, our system involves two different modules: an efficient and precise FAQ retrieval module, and a tag cloud generation module designed to help users to complete the comprehension of the retrieved information. Empirical results evidence the validity of our approach with respect to a number of state-of-the-art algorithms in terms of the most popular metrics in the field.

- M. Romero, A. Moreo, J.L. Castro, Collaborative System for Learning based on Questionnaires and Tasks, In 4th International Conference on EUropean Transnational Education (ICEUTE'13), September 2013, Pages 631-640, Salamanca, Spain.
  - Status: **Published**.
  - Type: International Conference paper.
  - Conference: ICEUTE 2013, Salamanca, Spain.
  - Reflected in: chapter V, section 3.

Abstract: Virtual Learning Environments allow to improve the learning interactivity in a collaborative scenario where the learning contents are proposed and accessed by both learners and teachers. In this work, we present CSLQT, a new Collaborative System for Learning based on Questionnaires and Tasks. This system is independent of any course structure or content, and provide users with functionalities to create, review, and evaluate new knowledge resources through questions and tasks. The benefits are two-fold: teachers are released from the tedious task of creating all resources, and students are encouraged to gain the necessary knowledge background before creating any new content. Additionally, a Fuzzy controller generates exams satisfying a customized set of objectives, that could be used for evaluation or auto-evaluation purposes. Our experiences with the system in real courses of the University of Granada indicate the tool is actually useful to improve the learning process.

- M. Romero, J.L. Castro, TRCloud system: creating a conceptually extended folksonomy from scratch, International Journal of Intelligent Systems, Submitted.
  - Status: **Submitted**.
  - Type: Journal Article.
  - Impact Factor (JCR 2013): 1.411.
  - Subject Category: Computer Science, Artificial Intelligence. Ranking 51 / 121 (Q2).
  - Correspondence with: chapter II, section 3

Abstract: In this paper, we present TRCloud, a tag recommender system designed to create a conceptually extended folksonomy from scratch. The system operates in multi-domain scale without the necessity of a prior knowledge. Simultaneously, the system obtains the knowledge about the domain in a collaborative approach. The extracted tags are a set of multi-terms representing concepts of the domain. To that end, we designed an hybrid approach to detect an initial set of candidate tags from the content of each resource, by means of syntactic, semantic, and frequency features of the terms. Additionally, the system adapts the weights of the rest of candidates when a user selects a tag, in function of syntactic and semantic relations existing among tags. TRCloud also provides an overview of the content by means of a tag cloud representation, with the aim to facilitate users to explore the conceptually extended folksonomy and to quickly identify which are the most important tags in the domain.

- A. Moreo, M. Romero, J.L. Castro, J.M. Zurita, Lexicon-based Comments-oriented News Sentiment Analyzer system, Expert Systems with Applications, Volume 39, Issue 10, August 2012, Pages 9166-9180, ISSN 0957-4174, 10.1016/j.eswa.2012.02.057.
  - Status: **Published**.
  - Type: Journal Article.
  - Impact Factor (JCR 2013): 1.965.
  - Subject Category: Computer Science, Artificial Intelligence. Ranking 30 / 121 (Q1).
  - Subject Category: Engineering, Electrical & Electronic. Ranking 63 / 247 (Q2).
  - Subject Category: Operations Research & Management Science. Ranking 11 / 79 (Q1).
  - Correspondence with: none.

Abstract: Thanks to the technological revolution that has accompanied the Web 2.0, users are able to interact intensively on the Internet, as reflected in social networks, blogs, forums, etc. In these scenarios, users can speak freely on any relevant topic. However, the high volume of user-generated content makes a manual analysis of this discourse unviable. Consequently, automatic analysis techniques are needed to extract the opinions expressed in users' comments, given that these opinions are an implicit barometer of unquestionable interest for a wide variety of companies, agencies, and organisms. We thus propose a lexicon-based Comments-oriented News Sentiment Analyzer (LCN-SA), which is able to deal with the following: (a) the tendency of many users to express their views in non-standard language; (b) the detection of the target of users' opinions in a multi-domain scenario; (c) the design of a linguistic modularized knowledge model with low-cost adaptability. The system proposed consists of an automatic Focus Detection Module and a Sentiment Analysis Module capable of assessing user opinions of topics in news items. These modules use a taxonomy-lexicon specifically designed for news analysis. Experiments show that the results obtained thus far are extremely promising.

- A. Moreo, M. Romero, J.L. Castro, J.M. Zurita, FAQtory: A framework to provide high-quality FAQ retrieval systems, Expert Systems with Applications, Volume 39, Issue 14, 15 October 2012, Pages 11525-11534, ISSN 0957-4174, 10.1016/j.eswa.2012.02.130.
    - Status: **Published**.
    - Type: Journal Article.
    - Impact Factor (JCR 2013): 1.965.
    - Subject Category: Computer Science, Artificial Intelligence. Ranking 30 / 121 (Q1).
    - Subject Category: Engineering, Electrical & Electronic. Ranking 63 / 247 (Q2).
    - Subject Category: Operations Research & Management Science. Ranking 11 / 79 (Q1).
    - Correspondence with: none.

Abstract: To facilitate access to information, companies usually try to anticipate and answer most typical customer's questions by creating Frequently Asked Questions (FAQs) lists. In this scenario, FAQ retrieval is the area of study concerned with recovering the most relevant Question/Answer pairs contained in FAQ compilations. Despite the amount of effort that has been devoted to investigate FAQ retrieval methods, how to create an maintain high quality FAQs has received less attention. In this article, we propose an entire framework to use, create and maintain intelligent FAQs. Usage mining techniques have been developed to take advantage of usage information in order to provide FAQ managers with meaningful information to improve their FAQs. Usage mining techniques include weaknesses detection and knowledge gaps discovery. In this way, the management of the FAQ is no longer directed only by expert knowledge but also by users requirements.

Appendices

1 ALKEx: Google-based regression analysis process

This appendix details the process carried out in order to obtain the regression equation that relates the number of results in Google for a query (predictor) and the frequency in language value for such term (dependant variable).

First of all, we have collected a subset of N random terms from the dictionary of frequency in language enclosing a range of normalized frequencies between 0.01 and 0.20 (i.e. absolute frequencies between 1 and 500,000). We took this upper limit value equal to 0.20 because this value helps to delimit common and uncommon terms, as can be observed in our experiments (chapter II Section 2.6). Following an a-priori sample size calculator schema [AS65, Coh88, CCWA03], we setted $N \geqslant 54$ the number of samples in the subset. It allowed us to obtain a medium anticipated effect size of 0.15 (by convention), a desired statistical power level of 0.8 (by convention) and a ρ-value of 0.05 (by convention). Therefore, we established $N = 54$. For each term in the subset, the number of results in Google search engine was collected using the term as a query. Once we obtain the samples with their respective values for the two variables, we compute the correlation coefficient and the ρ-value to observe the linear association between the dependant variable (absolute frequency in language) and the predictor (number of results). The correlation coefficient, r, is 0.5934 and the ρ-value is $2.26 \cdot 10^{-6}$. Given that the ρ-value is lower than $\alpha = 0.05$ (level of significance), we can reject the null hypothesis, i.e. both variables are correlated and the model is valid. The resulting regression equation that defines the relation between frequency and result (x variable) in our work is:

$$FrequencyInLanguage(x) = 1.8 \cdot 10^{-5} \cdot x + 13.493$$

The linear correlation coefficient could be more accurate but it allows us to obtain a useful prediction for terms not contained in the dictionary of frequency in language.

Bibliography

[AAOE11] Agbele K., Adetunmbi B., Olajide S., and Ekong D. (2011) Applying a novel query reformulation keywords algorithm in a mobile healthcare retrieval context. *Research Journal of Applied Sciences* 6: 184–193.

[ABDGM08] Avelãs M., Branco A., Del Gaudio R., and Martins P. (2008) Supporting e-learning with language technology for portuguese. In *Computational Processing of the Portuguese Language*, pp. 192–201. Springer.

[ABF07] Arbaugh J. B. and Benbunan-Fich R. (2007) The importance of participant interaction in online environments. *Decision Support Systems* 43(3): 853–865.

[AC04] Azevedo R. and Cromley J. G. (2004) Does training on self-regulated learning facilitate students' learning with hypermedia? *Journal of educational psychology* 96(3): 523.

[ACC$^+$03] Andronico A., Carbonaro A., Casadei G., Colazzo L., Molinari A., and Ronchetti M. (2003) Integrating a multi-agent recommendation system into a mobile learning management system. *Proceedings of Artificial Intelligence in Mobile System* pp. 123–132.

[AEARAK$^+$11] Al-Eroud A. F., Al-Ramahi M. A., Al-Kabi M. N., Alsmadi I. M., and Al-Shawakfa E. M. (2011) Evaluating google queries based on language preferences. *Journal of Information Science* 37(3): 282–292.

[AHSYAH11] Al-Hmouz A., Shen J., Yan J., and Al-Hmouz R. (2011) Modeling mobile learning system using anfis. In *Advanced Learning Technologies (ICALT), 2011 11th IEEE International Conference on*, pp. 378–380. IEEE.

[AM02] Alfonseca E. and Manandhar S. (2002) An unsupervised method for general named entity recognition and automated concept discovery. In *Proceedings of the 1st International Conference on General WordNet, Mysore, India*, pp. 34–43.

[AM14] Anand D. and Mampilli B. S. (2014) Folksonomy-based fuzzy user profiling for improved recommendations. *Expert Systems with Applications* 41(5): 2424–2436.

[AMW10] Arroyo I., Meheranian H., and Woolf B. P. (2010) Effort-based tutoring: An empirical approach to intelligent tutoring. In *EDM*, pp. 1–10. Citeseer.

[And03] Anderson T. (2003) Modes of interaction in distance education: Recent developments and research questions. *Handbook of distance education* pp. 129–144.

[APRS03] Avgeriou P., Papasalouros A., Retalis S., and Skordalakis M. (2003) Towards a pattern language for learning management systems. *Educational Technology & Society* 6(2): 11–24.

[AR02] Atack L. and Rankin J. (2002) A descriptive study of registered nurses' experiences with web-based learning. *Journal of Advanced Nursing* 40(4): 457–465.

[Arb04] Arbaugh J. (2004) Learning to learn online: A study of perceptual changes between multiple online course experiences. *The Internet and Higher Education* 7(3): 169–182.

[ARS$^+$11] Anjorin M., Rensing C., Steinmetz R., *et al.* (2011) Towards ranking in folksonomies for personalized recommender systems in e-learning. In *SPIM*, pp. 22–25.

[Art08] Artino A. R. (2008) Motivational beliefs and perceptions of instructional quality: predicting satisfaction with online training. *Journal of Computer Assisted Learning* 24(3): 260–270.

[AS65] Abramowitz M. and Stegun I. A. (1965) *Handbook of Mathematical Functions.* Dover, New York, NY.

[AS08] Agirre E. and Soroa A. (2008) Using the multilingual central repository for graph-based word sense disambiguation. In *Proceedings of the Sixth International Conference on Language Resources and Evaluation (LREC'08)*, pp. 1388–1392. European Language Resources Association (ELRA), Marrakech, Morocco.

[ASRB07] Adrian B., Sauermann L., and Roth-Berghofer T. (2007) Contag: A semantic tag recommendation system. *Proceedings of I-Semantics* 7: 297–304.

[Aus63] Ausubel D. P. (1963) *The psychology of meaningful verbal learning.* Grune & Stratton.

[BAB05] Bodenreider O., Aubry M., and Burgun A. (2005) Non-lexical approaches to identifying associative relations in the gene ontology. In *Pacific Symposium on Biocomputing. Pacific Symposium on Biocomputing*, page 91. NIH Public Access.

[Bat07] Battalio J. (2007) Interaction online: A reevaluation. *Quarterly Review of Distance Education* 8(4): 339–352.

[BB93] Buzan T. and Buzan B. (1993) *The mind map book.* Rajpal & Sons.

[BB00] Bruillard E. and Baron G.-L. (2000) Computer-based concept mapping a cognitive tool for students: a review. In *Proceedings of Conference on Educational Uses of Information and Communication Technologies (ICEUT 2000), 16th World Computer Congress, IFIP*, pp. 331–338. Citeseer.

[BBL62] Bitzer D. L., Braunfeld P. G., and Lichtenberger W. (1962) Plato ii: A multiple-student, computer-controlled, automatic teaching device. *Programmed learning and computer-based instruction* pp. 205–216.

[BC88] Burstein M. H. and Collins A. M. (1988) Modeling a theory of human plausible reasoning. In *AIMSA*, pp. 21–28.

[BC94] Bleakley A. and Carrigan J. (1994) *Resource-based learning activities: Information literacy for high school students.* American Library Association.

[BC95] Berge Z. L. and Collins M. P. (1995) *Computer mediated communication and the online classroom: distance learning.* Hampton Press Cresskill.

[BC98] Bra P. D. and Calvi L. (1998) Aha! an open adaptive hypermedia architecture. *New Review of Hypermedia and Multimedia* 4(1): 115–139.

[BC99] Berland M. and Charniak E. (1999) Finding parts in very large corpora. In *Proceedings of the 37th annual meeting of the Association for Computational Linguistics on Computational Linguistics*, pp. 57–64. Association for Computational Linguistics.

[BC03] Bunt A. and Conati C. (2003) Probabilistic student modelling to improve exploratory behaviour. *User Modeling and User-Adapted Interaction* 13(3): 269–309.

[BCC$^+$00] Berger A., Caruana R., Cohn D., Freitag D., and Mittal V. (2000) Bridging the lexical chasm: statistical approaches to answer-finding. In *Proceedings of the 23rd annual international ACM SIGIR conference on Research and development in information retrieval*, pp. 192–199.

[BCDP92] Bednar A. K., Cunningham D., Duffy T. M., and Perry J. D. (1992) Theory into practice: How do we link. *Constructivism and the technology of instruction: A conversation* pp. 17–34.

[BCM05] Buitelaar P., Cimiano P., and Magnini B. (2005) *Ontology learning from text: An overview*, volumen 123.

[BCS01] Billings D. M., Connors H. R., and Skiba D. J. (2001) Benchmarking best practices in web-based nursing courses. *Advances in Nursing Science* 23(3): 41–52.

[Ber02] Berge Z. L. (2002) Active, interactive, and reflective elearning. *Quarterly Review of Distance Education* 3(2): 181–190.

[BFE05] Brown J. C., Frishkoff G. A., and Eskenazi M. (2005) Automatic question generation for vocabulary assessment. In *Proceedings of the conference on Human Language Technology and Empirical Methods in Natural Language Processing*, pp. 819–826. Association for Computational Linguistics.

[BG10] Baschera G.-M. and Gross M. (2010) Poisson-based inference for perturbation models in adaptive spelling training. *International Journal of Artificial Intelligence in Education* 20(4): 333–360.

[BGN08] Bateman S., Gutwin C., and Nacenta M. (2008) Seeing things in the clouds: the effect of visual features on tag cloud selections. In *Proceedings of the nineteenth ACM conference on Hypertext and hypermedia*, pp. 193–202. New York, NY, USA.

[BK56] Bloom B. S. and Krathwohl D. R. (1956) *Taxonomy of educational objectives: The classification of educational goals.* UK, Longman Group.

[BL11] Beel J. and Langer S. (2011) An exploratory analysis of mind maps. In *Proceedings of the 11th ACM symposium on Document engineering*, volumen 36, pp. 100–103.

[Blo84] Bloom B. S. (1984) The 2 sigma problem: The search for methods of group instruction as effective as one-to-one tutoring. *Educational researcher* pp. 4–16.

[Blo09] Blomqvist E. (2009) Ontocase-automatic ontology enrichment based on ontology design patterns. In Bernstein A., Karger D., Heath T., Feigenbaum L., Maynard D., Motta E., and Thirunarayan K. (Eds.) *The Semantic Web - ISWC 2009*, volumen 5823 of *Lecture Notes in Computer Science*, pp. 65–80. Springer Berlin Heidelberg.

[BM02] Brusilovsky P. and Maybury M. T. (2002) From adaptive hypermedia to the adaptive web. *Communications of the ACM* 45(5): 30–33.

[BM04] Burkhard R. and Meier M. (2004) Tube map: Evaluation of a visual metaphor for interfunctional communication of complex projects. In *Proceedings of I-Know*, volumen 4, pp. 449–456.

[BM06] Brooks C. H. and Montanez N. (2006) Improved annotation of the blogosphere via autotagging and hierarchical clustering. In *Proceedings of the 15th international conference on World Wide Web*, pp. 625–632. ACM.

[BM07] Brusilovsky P. and Millán E. (2007) User models for adaptive hypermedia and adaptive educational systems. In *The adaptive web*, pp. 3–53. Springer-Verlag.

[Bou95] Boud D. (1995) *Enhancing learning through self assessment*, volumen 1. London [etc.]: Kogan Page.

[BP98] Brin S. and Page L. (1998) The anatomy of a large-scale hypertextual web search engine. *Computer Networks and ISDN Systems* 24(1-7): 107–117. Proceedings of the Seventh International World Wide Web Conference.

[BP03] Brusilovsky P. and Peylo C. (2003) Adaptive and intelligent web-based educational systems. *International Journal of Artificial Intelligence in Education* 13(2): 159–172.

[BP06] Bunescu R. and Paşca M. (2006) Using encyclopedic knowledge for named entity disambiguation. In *Proceedings of the 11th Conference of the European Chapter of the Association for Computational Linguistic. EACL 2006*, pp. 9–16. Trento, Italy.

[BR12] Barforush A. A. and Rahnama A. (2012) Ontology learning: revisted. *Journal of Web Engineering* 11(4): 269–289.

[BRK05] Bracewell D. B., Ren F., and Kuriowa S. (2005) Multilingual single document keyword extraction for information retrieval. In *Proceedings of 2005 IEEE International Conference on Natural Language Processing and Knowledge Engineering (IEEE NLP-KE'05)*. Wuhan, China.

[Bro10] Brown S. (2010) From vles to learning webs: the implications of web 2.0 for learning and teaching. *Interactive Learning Environments* 18(1): 1–10.

[Bru96] Brusilovsky P. (1996) Methods and techniques of adaptive hypermedia. *User modeling and user-adapted interaction* 6(2-3): 87–129.

[Bru00] Brusilovsky P. (2000) Adaptive hypermedia: From intelligent tutoring systems to web-based education. In *Intelligent Tutoring Systems*, pp. 1–7. Springer.

[But98] Butcher H. (1998) *Meeitng managers' information needs.* London: Aslib.

[CB95] Cox R. and Brna P. (1995) Supporting the use of external representations in problem solving: The need for flexible learning environments. *Journal of Artificial Intelligence in Education* 6: 239–302.

[CBOC11] Cristea A. D., Berdie A. D., Osaci M., and Chirtoc D. (2011) The advantages of using mind map for learning web dynpro. *Computer Applications in Engineering Education* 19(1): 201–207.

[CC04] Carmona C. and Conejo R. (2004) A learner model in a distributed environment. In *Adaptive Hypermedia and Adaptive Web-Based Systems*, pp. 353–359. Springer.

[CCH$^+$05] Cañas A. J., Carff R., Hill G., Carvalho M., Arguedas M., Eskridge T. C., Lott J., and Carvajal R. (2005) Concept maps: Integrating knowledge and information visualization. In *Knowledge and information visualization*, pp. 205–219. Springer.

[CCM08] Carmona C., Castillo G., and Millán E. (2008) Designing a dynamic bayesian network for modeling students' learning styles. In *Advanced Learning Technologies, 2008. ICALT'08. Eighth IEEE International Conference on*, pp. 346–350. IEEE.

[CCV$^+$08] Casellas N., Casanovas P., Vallb J., Poblet M., Blazquez M., Contreras J., López-Cobo J., and Benjamins V. (2008) Semantic enhancement for legal information retrieval: Iuriservice performance. In *Proceedings of the 11th international conference on Artificial intelligence and law*, pp. 49–57. Stanford, California.

[CCWA03] Cohen J., Cohen P., West S. G., and Aiken L. S. (2003) *Applied Multiple Regression/Correlation Analysis for the Behavioral Sciences.* Lawrence Earlbaum Associates, Mahwah, NJ, 3 edition.

[CD96] Cunningham D. and Duffy T. (1996) Constructivism: Implications for the design and delivery of instruction. *Handbook of research for educational communications and technology* pp. 170–198.

[CD08] Chen C.-M. and Duh L.-J. (2008) Personalized web-based tutoring system based on fuzzy item response theory. *Expert Systems with Applications* 34(4): 2298–2315.

[CDBS01] Collis B., De Boer W., and Slotman K. (2001) Feedback for web-based assignments. *Journal of Computer Assisted Learning* 17(3): 306–313.

[CDF$^+$07] Cassel L. N., Davies G., Fone W., Hacquebard A., Impagliazzo J., LeBlanc R., Little J. C., McGettrick A., and Pedrona M. (2007) The computing ontology: application in education. In *ACM SIGCSE Bulletin*, volumen 39, pp. 171–183. ACM.

[CDNR09] Cutts S., Davies P., Newell D., and Rowe N. (2009) Requirements for an adaptive multimedia presentation system with contextual supplemental support media. In *Advances in Multimedia, 2009. MMEDIA'09. First International Conference on*, pp. 62–67. IEEE.

[CG87] Chickering A. W. and Gamson Z. F. (1987) Seven principles for good practice in undergraduate education. *AAHE bulletin* 3: 7.

[CG09] Colomar M. P. A. and Guzmán E. G. (2009) Ict-sustour and marketour: Two second language acquisition projects through a virtual learning environment. *Computers & Education* 52(3): 581–587.

[CGPRC$^+$14] Conde M. Á., García-Peñalvo F. J., Rodríguez-Conde M. J., Alier M., Casany M. J., and Piguillem J. (2014) An evolving learning management system for new educational environments using 2.0 tools. *Interactive Learning Environments* 22(2): 188–204.

[CGV02] Conati C., Gertner A., and Vanlehn K. (2002) Using bayesian networks to manage uncertainty in student modeling. *User modeling and user-adapted interaction* 12(4): 371–417.

[CHB04] Clayton B., Hyde P., and Booth R. (2004) Exploring assessment in flexible delivery of vocational education and training programs.

[CHB$^+$06] Cañas A. J., Hill G., Bunch L., Carff R., Eskridge T., and Pérez C. (2006) Kea: A knowledge exchange architecture based on web services, concept maps and cmaptools. In *Concept Maps: Theory, Methodology, Technology. Proceedings of the Second International Conference on Concept Mapping*, volumen 1, pp. 304–310.

[Chi03] Chiu Y. H. (2003) An interface agent with ontology-supported user models. Master's thesis, National Taiwan University of Science and Technology.

[Chi07] Chi Y.-L. (2007) Elicitation synergy of extracting conceptual tags and hierarchies in textual document. *Expert Systems with Applications* 32(2): 349–357.

[CHK13] Cress U., Held C., and Kimmerle J. (2013) The collective knowledge of social tags: Direct and indirect influences on navigation, learning, and information processing. *Computers & Education* 60(1): 59–73.

[Cho04] Chou C. C. (2004) A model of learner-centered computer-mediated interaction for collaborative distance learning. *International Journal on E-learning* 3(1): 11–18.

[Cho10] Chong E. K. M. (2010) Using blogging to enhance the initiation of students into academic research. *Computers & Education* 55(2): 798–807.

[Chr06] Christiaens S. (2006) Metadata mechanisms: From ontology to folksonomy... and back. In *On the Move to Meaningful Internet Systems 2006: OTM 2006 Workshops*, pp. 199–207. Springer.

[CK04] Clariana R. and Koul R. (2004) A computer-based approach for translating text into concept map-like representations. In *Proceedings of the first international conference on concept mapping*, pp. 14–17.

[CL11] Chen P.-I. and Lin S.-J. (2011) Word adhoc network: Using google core distance to extract the most relevant information. *Knowledge-Based Systems* 24(3): 393–405.

[CLC05] Chen C.-M., Lee H.-M., and Chen Y.-H. (2005) Personalized e-learning system using item response theory. *Computers & Education* 44(3): 237–255.

[CLH09] Cheng S.-C., Lin Y.-T., and Huang Y.-M. (2009) Dynamic question generation system for web-based testing using particle swarm optimization. *Expert Systems with Applications* 36(1): 616–624.

[CLL12] Chiou C.-C., Lee L.-T., and Liu Y.-Q. (2012) Effect of novak colorful concept map with digital teaching materials on student academic achievement. *Procedia-social and Behavioral sciences* 64: 192–201.

[CLT11] Chu K.-K., Lee C.-I., and Tsai R.-S. (2011) Ontology technology to assist learners? navigation in the concept map learning system. *Expert Systems with Applications* 38(9): 11293–11299.

[CM89] Collins A. and Michalski R. (1989) The logic of plausible reasoning: A core theory. *cognitive science* 13(1): 1–49.

[CM08] Csomai A. and Mihalcea R. (2008) Linguistically motivated features for enhanced back-of-the-book indexing. In *Proceedings of the 46th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies (ACL-08: HLT)*, pp. 932–940. The Association for Computer Linguistics, Columbus, Ohio, USA.

[CM09] Conati C. and Maclaren H. (2009) Empirically building and evaluating a probabilistic model of user affect. *User Modeling and User-Adapted Interaction* 19(3): 267–303.

[CMM08] Coursey K. H., Mihalcea R., and Moen W. E. (2008) Automatic keyword extraction for learning object repositories. *Proceedings of the American Society for Information Science and Technology* 45(1): 1–10.

[Cof07] Coffey J. W. (2007) A meta-cognitive tool for courseware development, maintenance, and reuse. *Computers & Education* 48(4): 548–566.

[Coh88] Cohen J. (1988) *Statistical Power Analysis for the Behavioral Sciences.* Lawrence Earlbaum Associates, Hillsdale, NJ.

[CON87] CONKLIN J. (1987) Hypertext: An introduction and survey. *IEEE Computer* 20(9): 17–41.

[Coo03] Cooper J. W. (2003) Visualization of relational text information for biomedical knowledge discovery. In *Information Visualization Interfaces for Retrieval and Analysis workshop. ACM SIGIR.* Citeseer.

[Cos07] Costello E. (2007) Reuse through rapid development. In *International Symposium for Engineering Education, ISEE-07.* Dublin City University.

[CotEC01] Commission of the European Communities B. (2001) Communication from the commission to the council and the european parliament. the e-learning action plan. designing tomorrow's education.

[Cox99] Cox R. (1999) Representation construction, externalised cognition and individual differences. *Learning and instruction* 9(4): 343–363.

[CPC08] Chiu D., Pan Y., and Chang W. (2008) Using rough set theory to construct e-learning faq retrieval infrastructure. pp. 547–552.

[CPSTS05] Cimiano P., Pivk A., Schmidt-Thieme L., and Staab S. (2005) Learning taxonomic relations from heterogeneous sources of evidence. *Ontology Learning from Text: Methods, evaluation and applications* .

[CS13] Cakula S. and Salem A.-B. M. (2013) E-learning developing using ontological engineering. *WSEAS Transactions on Information Science and Applications* 1(1): 14–25.

[CSC03] Chang K.-E., Sung Y.-T., and Chiou S.-K. (2003) Use of hierarchical hyper concept map in web-based courses. *Journal of Educational Computing Research* 27(4): 335–353.

[CV05a] Cimiano P. and Völker J. (2005) Text2onto-a framework for ontology learning and data-driven change discovery. In *Natural language processing and information systems: 10th International Conference on Applications of Natural Language to Information Systems, NLDB 2005, Alicante, Spain, June 15-17, 2005; proceedings. Lecture notes in computer science, 3513.*

[CV05b] Cimiano P. and Völker J. (2005) Towards large-scale, open-domain and ontology-based named entity classification. In *Proceedings of the International Conference on Recent Advances in Natural Language Processing (RANLP).*

[CV10] Chrysafiadi K. and Virvou M. (2010) Modeling student's knowledge on programming using fuzzy techniques. In *Intelligent Interactive Multimedia Systems and Services*, pp. 23–32. Springer.

[CV12] Chrysafiadi K. and Virvou M. (2012) Evaluating the integration of fuzzy logic into the student model of a web-based learning environment. *Expert Systems with Applications* 39(18): 13127–13134.

[CV13] Chrysafiadi K. and Virvou M. (2013) Student modeling approaches: A literature review for the last decade. *Expert Systems with Applications* 40(11): 4715–4729.

[CW05] Conole G. and Warburton B. (2005) A review of computer-assisted assessment. *Research in learning technology* 13(1).

[CWC10] Cheung R., Wan C., and Cheng C. (2010) An ontology-based framework for personalized adaptive learning. In Luo X., Spaniol M., Wang L., Li Q., Nejdl W., and Zhang W. (Eds.) *Advances in Web-Based Learning ? ICWL 2010*, volumen 6483 of *Lecture Notes in Computer Science*, pp. 52–61. Springer Berlin Heidelberg.

[CZ02] Conati C. and Zhou X. (2002) Modeling students? emotions from cognitive appraisal in educational games. In *Intelligent tutoring systems*, pp. 944–954. Springer.

[DA07] Doukas N. and Andreatos A. (2007) Advancing electronic assessment. *International Journal of Computers, Communications & Control* 2(1): 56–65.

[Dav10] Davies M. (2010) The corpus of contemporary american english as the first reliable monitor corpus of english. *Literary and Linguistic Computing* 25(4): 447–464.

[DAV12] Dib H. and Adamo-Villani N. (2012) An e-tool for assessing undergraduate students' learning of surveying concepts and practices. In *IT Revolutions*, pp. 189–201. Springer.

[dBC02] de Boer W. and Collis B. (2002) A changing pedagogy in e-learning: From acquisition to contribution. *Journal of Computing in Higher Education* 13(2): 87–101.

[DBM04] Dittenbach M., Berger H., and Merll D. (2004) Improving domain ontologies by mining semantics from text. In *Proceedings of the first Asian-Pacific conference on Conceptual modelling-Volume 31*, pp. 91–100. Australian Computer Society, Inc.

[DCVOE00] De Corte E., Verschaffel L., and Op't Eynde P. (2000) Self-regulation: a characteristic and a goal of mathematics education. *Handbook of Self-Regulation* .

[DdB12] Desmarais M. C. and d Baker R. S. (2012) A review of recent advances in learner and skill modeling in intelligent learning environments. *User Modeling and User-Adapted Interaction* 22(1-2): 9–38.

[DDLL12] Durao F., Dolog P., Leginus M., and Lage R. (2012) Simspectrum: a similarity based spectral clustering approach to generate a tag cloud. In *Current Trends in Web Engineering*, pp. 145–154. Springer.

[DG08] Drumond L. and Girardi R. (2008) A survey of ontology learning procedures. *WONTO* 427.

[DHMPP11] Derntl M., Hampel T., Motschnig-Pitrik R., and Pitner T. (2011) Inclusive social tagging and its support in web 2.0 services. *Computers in Human Behavior* 27(4): 1460–1466.

[DHNS04] Dolog P., Henze N., Nejdl W., and Sintek M. (2004) The personal reader: Personalizing and enriching learning resources using semantic web technologies. In *Adaptive Hypermedia and Adaptive Web-Based Systems*, pp. 85–94. Springer.

[DJLW08] Datta R., Joshi D., Li J., and Wang J. Z. (2008) Image retrieval: Ideas, influences, and trends of the new age. *ACM Computing Surveys (CSUR)* 40(2): 5.

[DK12] Dabbagh N. and Kitsantas A. (2012) Personal learning environments, social media, and self-regulated learning: A natural formula for connecting formal and informal learning. *The Internet and higher education* 15(1): 3–8.

[DR97] Daniels J. and Rissland E. (1997) What you saw is what you want: Using cases to seed information retrieval. In *Case-Based Reasoning Research and Development*, volumen 1266 of *Lecture Notes in Computer Science*, pp. 325–336.

[DRMLOA08] Duez-Rodriguez H., Morales-Luna G., and Olmedo-Aguirre J. O. (2008) Ontology-based knowledge retrieval. In *Artificial Intelligence, 2008. MICAI'08. Seventh Mexican International Conference on*, pp. 23–28. IEEE.

[DS97] Dias P. and Sousa P. (1997) Understanding navigation and disorientation in hypermedia learning environments. *Journal of Educational Multimedia and Hypermedia* 6(2): 173–185.

[DSS$^+$02] Dillenbourg P., Schneider D., Synteta P., *et al.* (2002) Virtual learning environments. In *Proceedings of the 3rd Hellenic Conference'Information & Communication Technologies in Education'*, pp. 3–18.

[EC09] European Comminsion M. R. f. H. E. (2009) The bologna process 2020? the european higher education area in the decade [online]. *Available* http://www.ehea.info/Uploads/Declarations/Leuven_Louvain-la-Neuve_Communiqu%C3%A9_April_2009.pdf.

[EM00] Edmunds A. and Morris A. (2000) The problem of information overload in business organisations: a review of the literature. *International journal of information management* 20(1): 17–28.

[EN93] Ertmer P. A. and Newby T. J. (1993) Behaviorism, cognitivism, constructivism: Comparing critical features from an instructional design perspective. *Performance improvement quarterly* 6(4): 50–72.

[FAT98] Frantzi K. T., Ananiadou S., and Tsujii J. (1998) The c-value/nc-value method of automatic recognition for multi-word terms. In *Research and Advanced Technology for Digital Libraries*, pp. 585–604. Springer.

[FH00] Freed J. E. and Huba M. E. (2000) Learner-centered assessment on college campuses: Shifting the focus from teaching to learning. *Needham Heights, MA: Allyn &Bacon* .

[FK04] Frees S. and Kessler G. D. (2004) Developing collaborative tools to promote communication and active learning in academia. In *Frontiers in Education, 2004. FIE 2004. 34th Annual*, pp. S3B–20. IEEE.

[Flo00] Flottemesch K. (2000) Building effective interaction in distance education: A review of the literature. *Educational Technology* 40(3): 46–51.

[Fos89] Foss C. L. (1989) Tools for reading and browsing hypertext. *Information Processing & Management* 25(4): 407–418.

[FPS$^+$00] Fredericksen E., Pickett A., Shea P., Pelz W., and Swan K. (2000) Student satisfaction and perceived learning with on-line courses: Principles and examples from the suny learning network. *Journal of Asynchronous Learning Networks* 4(2): 7–41.

[FPW$^+$99] Frank E., Paynther G., Witten I. H., Gutwin C., and Nevil-Manning C. G. (1999) Domain-specific keyphrase extraction. In *Proceedings of the Sixteenth International Joint Conference on Artificial Intelligence (IJCAI '99)*, pp. 668–673. Stockholm, Sweden.

[FRG04] Faraco R. A., Rosatelli M. C., and Gauthier F. A. (2004) An approach of student modelling in a learning companion system. In *Advances in Artificial Intelligence–IBERAMIA 2004*, pp. 891–900. Springer.

[FVH$^+$07] Franklin T., Van Harmelen M., *et al.* (2007) Web 2.0 for content for learning and teaching in higher education. *JISC collections* .

[GBK09] Gulla J. A., Brasethvik T., and Kvarv G. S. (2009) Association rules and cosine similarities in ontology relationship learning. In *Enterprise Information Systems*, pp. 201–212. Springer.

[GBM03] Girju R., Badulescu A., and Moldovan D. (2003) Learning semantic constraints for the automatic discovery of part-whole relations. In *Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology-Volume 1*, pp. 1–8. Association for Computational Linguistics.

[GC05] Guzmán E. and Conejo R. (2005) Self-assessment in a feasible, adaptive web-based testing system. *Education, IEEE Transactions on* 48(4): 688–695.

[GCBSMS09] Garzón Castro C. L., Beltrán Sierra L. M., and Martínez Sánchez P. M. T. (2009) Estudio de percepción sobre metodologías de enseñanza de temas de electrónica en programas diferentes a ingeniería electrónica. *Revista Educación en Ingeniería* 4(8): 93–101.

[GCHO05] Graesser A. C., Chipman P., Haynes B. C., and Olney A. (2005) Autotutor: An intelligent tutoring system with mixed-initiative dialogue. *Education, IEEE Transactions on* 48(4): 612–618.

[GDH07] Gordon S. C., Dembo M. H., and Hocevar D. (2007) Do teachers' own learning behaviors influence their classroom goal orientation and control ideology? *Teaching and Teacher Education* 23(1): 36–46.

[GFC04] Ghoniem M., Fekete J., and Castagliola P. (2004) A comparison of the readability of graphs using node-link and matrix-based representations. In *Information Visualization, 2004. INFOVIS 2004. IEEE Symposium on*, pp. 17–24.

[GGL09] Grineva M., Grinev M., and Lizorkin D. (2009) Extracting key terms from noisy and multitheme documents. In *Proceedings of the 18th international conference on World wide web (WWW'09)*, pp. 661–670. Association for Computing Machinery, Barcelona, Spain.

[GGSB05] Galassi U., Giordana A., Saitta L., and Botta M. (2005) Learning profiles based on hierarchical hidden markov model. In *Foundations of Intelligent Systems*, volumen 3488 of *Lecture Notes in Computer Science*, pp. 1–3. Springer Berlin / Heidelberg.

[GH06] Golder S. A. and Huberman B. A. (2006) Usage patterns of collaborative tagging systems. *Journal of information science* 32(2): 198–208.

[GK04] Garrison D. R. and Kanuka H. (2004) Blended learning: Uncovering its transformative potential in higher education. *The internet and higher education* 7(2): 95–105.

[GKPM02] Grigoriadou M., Kornilakis H., Papanikolaou K. A., and Magoulas G. D. (2002) Fuzzy inference for student diagnosis in adaptive educational hypermedia. In *Methods and Applications of Artificial Intelligence*, pp. 191–202. Springer.

[GM06] Gabrilovich E. and Markovitch S. (2006) Overcoming the brittleness bottleneck using wikipedia: Enhancing text categorization with encyclopedic knowledge. In *Proceedings of the Twenty-First National Conference on Artificial Intelligence*, pp. 1301–1306. Boston, MA.

[GPMM04] Gómez-Pérez A. and Manzano-Macho D. (2004) An overview of methods and tools for ontology learning from texts. *The knowledge engineering review* 19(03): 187–212.

[GRS$^+$09] Gemmell J., Ramezani M., Schimoler T., Christiansen L., and Mobasher B. (2009) The impact of ambiguity and redundancy on tag recommendation in folksonomies. In *Proceedings of the third ACM conference on Recommender systems*, pp. 45–52. ACM.

[Gru93] Gruber T. R. (1993) A translation approach to portable ontology specifications. *Knowledge acquisition* 5(2): 199–220.

[GSCAGP12] Garcia-Silva A., Corcho O., Alani H., and Gomez-Perez A. (2012) Review of the state of the art: Discovering and associating semantics to tags in folksonomies. *The Knowledge Engineering Review* 27(01): 57–85.

[GSIM11] Goguadze G., Sosnovsky S., Isotani S., and McLaren B. M. (2011) Towards a bayesian student model for detecting decimal misconceptions. In *Proceedings of the 19th International Conference on Computers in Education, Chiang Mai, Thailand*, pp. 6–8.

[GSR08] Gacitua R., Sawyer P., and Rayson P. (2008) A flexible framework to experiment with ontology learning techniques. *Knowledge-Based Systems* 21(3): 192–199.

[GSRM09] Gemmell J., Schimoler T., Ramezani M., and Mobasher B. (2009) Adapting k-nearest neighbor for tag recommendation in folksonomies. *ITWP* 528.

[GWB10] Gazendam L., Wartena C., and Brussee R. (2010) Thesaurus based term ranking for keyword extraction. In *Workshop on Database and Expert Systems Applications (DEXA'10)*, pp. 49–53. Bilbao, Spain.

[GZ09] Guo Q. and Zhang M. (2009) Question answering based on pervasive agent ontology and semantic web. *Knowledge-Based Systems* 22: 443–448.

[H$^+$91] Hartman K. *et al.* (1991) Patterns of social interaction and learning to write: Some effects of network technologies. *Written Communication* 8(1): 79–113.

[HA89] Hammond N. and Allinson L. (1989) Extending hypertext for learning: an investigation of access and guidance tools. *People and computers V* pp. 293–304.

[Ham01] Hamid A. A. (2001) E-learning: is it the "e" or the learning that matters? *The Internet and Higher Education* 4(3): 311–316.

[HBML95] Hammond K., Burke R., Martin C., and Lytinen S. (1995) Faq finder: a case-based approach to knowledge navigation. In *Artificial Intelligence for Applications, 1995. Proceedings., 11th Conference on*, pp. 80–86.

[HCC$^+$12] Huang H.-S., Chiou C.-C., Chiang H.-K., Lai S.-H., Huang C.-Y., and Chou Y.-Y. (2012) Effects of multidimensional concept maps on fourth graders? learning in web-based computer course. *Computers & Education* 58(3): 863–873.

[HCS12] Huang Y.-T., Chen M. C., and Sun Y. S. (2012) Personalized automatic quiz generation based on proficiency level estimation. In *Proceedings of the 20th International Conference on Computers in Education*.

[HDJG94] Holt P., Dubs S., Jones M., and Greer J. (1994) The state of student modelling. In *Student modelling: The key to individualized knowledge-based instruction*, pp. 3–35. Springer.

[HE89] Hardman D. and Edwards L. (1989) Lost in hyperspace: Cognitive mapping and navigation in a hypertext environment. *Hypertext: Theory into practice* pp. 105–145.

[Hea92] Hearst M. A. (1992) Automatic acquisition of hyponyms from large text corpora. In *Proceedings of the 14th conference on Computational linguistics-Volume 2*, pp. 539–545. Association for Computational Linguistics.

[HEBR11] Hazman M., El-Beltagy S. R., and Rafea A. (2011) Survey of ontology learning approaches. *International Journal of Computer Applications* 22.

[Hei05] Heitmann G. (2005) Challenges of engineering education and curriculum development in the context of the bologna process. *European Journal of Engineering Education* 30(4): 447–458.

[HF97] Hamp B. and Feldweg H. (1997) Germanet-a lexical-semantic net for german. In *Proceedings of ACL workshop Automatic Information Extraction and Building of Lexical Semantic Resources for NLP Applications*, pp. 9–15. Citeseer.

[HGS$^+$12] Hatala M., Gasevic D., Siadaty M., Jovanovic J., and Torniai C. (2012) Ontology extraction tools: An empirical study with educators. *Learning Technologies, IEEE Transactions on* 5(3): 275–289.

[HHC07] Huang M.-J., Huang H.-S., and Chen M.-Y. (2007) Constructing a personalized e-learning system based on genetic algorithm and case-based reasoning approach. *Expert Systems with Applications* 33(3): 551–564.

[HJ96] Harlen W. and James M. (1996) Creating a positive impact of assessment on learning. In *Anual Meting of the American Educational Research Asociation.* ERIC, New York.

[HJSS06] Hotho A., Jäschke R., Schmitz C., and Stumme G. (2006) Information retrieval in folksonomies: Search and ranking. In *The semantic web: research and applications*, pp. 411–426. Springer.

[HK07a] Halvey M. J. and Keane M. T. (2007) An assessment of tag presentation techniques. In *Proceedings of the 16th international conference on World Wide Web*, pp. 1313–1314. New York, NY, USA.

[HK07b] Halvey M. J. and Keane M. T. (2007) An assessment of tag presentation techniques. In *Proceedings of the 16th international conference on World Wide Web*, pp. 1313–1314. New York, NY, USA.

[HKSKF07] HaCohen-Kerner Y., Stern I., Korkus D., and Fredj E. (2007) Automatic machine learning of keyphrase extraction from short html documents written in hebrew. *Cybernetics and Systems* 38(1): 1–21.

[HLP06] Holsapple C. W. and Lee-Post A. (2006) Defining, assessing, and promoting e-learning success: An information systems perspective. *Decision Sciences Journal of Innovative Education* 4(1): 67–85.

[HN01] Heift T. and Nicholson D. (2001) Web delivery of adaptive and interactive language tutoring. *International Journal of Artificial Intelligence in Education* 12(4): 310–325.

[Hod08] Hodges C. B. (2008) Self-efficacy in the context of online learning environments: A review of the literature and directions for research. *Performance Improvement Quarterly* 20(3-4): 7–25.

[How10] Howarth P. (2010) The opportunities and challenges faced in utilising e-based assessment. In *Proc. of Annual Conference of Educational Research Center on Educational Measurement, Beirut.*

[HS98] Harris J. W. and Stöcker H. (1998) *Handbook of mathematics and computational science.* Springer.

[HSAF10] Hernández Y., Sucar L. E., and Arroyo-Figueroa G. (2010) Evaluating an affective student model for intelligent learning environments. In *Advances in Artificial Intelligence–IBERAMIA 2010*, pp. 473–482. Springer.

[Hul03] Hulth A. (2003) Improved automatic keyword extraction given more linguistic knowledge. In *Proceedings of the 2003 Conference on Empirical Methods in Natural Language Processing*, pp. 216–223. Sapporo, Japan.

[Hwa03] Hwang G.-J. (2003) A test-sheet-generating algorithm for multiple assessment requirements. *Education, IEEE Transactions on* 46(3): 329–337.

[Hwa05] Hwang G.-J. (2005) A data mining approach to diagnosing student learning problems in sciences courses. *International Journal of Distance Education Technologies (IJDET)* 3(4): 35–50.

[HYY06] Hwang G.-J., Yin P.-Y., and Yeh S.-H. (2006) A tabu search approach to generating test sheets for multiple assessment criteria. *Education, IEEE Transactions on* 49(1): 88–97.

[IM01] Ide N. and Macleod C. (2001) The american national corpus: A standardized resource of american english. In *Proceedings of Corpus Linguistics 2001*, volumen 3.

[IMK00] Iwanska L., Mata N., and Kruger K. (2000) Fully automatic acquisition of taxonomic knowledge from large corpora of texts: limited-syntax knowledge representation system based on natural language. In *In LM Iwanksa and SC Shapiro, editors, Natural Language Processing and Knowledge Processing*. Citeseer.

[Jam95] Jameson A. (1995) Numerical uncertainty management in user and student modeling: An overview of systems and issues. *User Modeling and User-Adapted Interaction* 5(3-4): 193–251.

[JBFC10] Jones N., Blackey H., Fitzgibbon K., and Chew E. (2010) Get out of myspace! *Computers & Education* 54(3): 776–782.

[JCL05] Jeon J., Croft W. B., and Lee J. H. (2005) Finding similar questions in large question and answer archives. In *Proceedings of the 14th ACM international conference on Information and knowledge management*, pp. 84–90.

[JGF09] Johnson R. D., Gueutal H., and Falbe C. M. (2009) Technology, trainees, metacognitive activity and e-learning effectiveness. *Journal of managerial psychology* 24(6): 545–566.

[JH09] Ju S. and Hwang K.-B. (2009) A weighting scheme for tag recommendation in social bookmarking systems. In *Proc. the ECML/PKDD 2009 Discovery Challenge Workshop*, pp. 109–118.

[JJG12] Jeremić Z., Jovanović J., and Gašević D. (2012) Student modeling and assessment in intelligent tutoring of software patterns. *Expert Systems with Applications* 39(1): 210–222.

[JJS85] Johnson R. T., Johnson D. W., and Stanne M. B. (1985) Effects of cooperative, competitive, and individualistic goal structures on computer-assisted instruction. *Journal of Educational Psychology* 77(6): 668.

[JLH05] Jaeschke G., Leissler M., and Hemmje M. (2005) Modeling interactive, 3-dimensional information visualizations supporting information seeking behaviors. In *Knowledge and Information Visualization*, pp. 119–135. Springer.

[JMH$^+$07] Jäschke R., Marinho L., Hotho A., Schmidt-Thieme L., and Stumme G. (2007) Tag recommendations in folksonomies. In *Knowledge Discovery in Databases: PKDD 2007*, pp. 506–514. Springer.

[Jon92] Jonassen D. H. (1992) Evaluating constructivistic learning. *Constructivism and the technology of instruction: A conversation* pp. 137–148.

[JT00] Jiang M. and Ting E. (2000) A study of factors influencing students? perceived learning in a web-based course environment. *International Journal of Educational Telecommunications* 6(4): 317–338.

[Jua10] Juan Z. M. (2010) An effective similarity measurement for faq question answering system. *Electrical and Control Engineering, International Conference on* 0: 4638–4641.

[JZZL10] Jia B., Zhong S., Zheng T., and Liu Z. (2010) The study and design of adaptive learning system based on fuzzy set theory. In *Transactions on edutainment IV*, pp. 1–11. Springer.

[KAH$^+$97] Koedinger K. R., Anderson J. R., Hadley W. H., Mark M. A., *et al.* (1997) Intelligent tutoring goes to school in the big city. *International Journal of Artificial Intelligence in Education (IJAIED)* 8: 30–43.

[Kal06] Kalyuga S. (2006) Assessment of learners' organised knowledge structures in adaptive learning environments. *Applied cognitive psychology* 20(3): 333–342.

[Kal11] Kalyuga S. (2011) Effects of information transiency in multimedia learning. *Procedia-Social and Behavioral Sciences* 30: 307–311.

[Kas91] Kass R. (1991) Building a user model implicitly from a cooperative advisory dialog. *User Modeling and User-Adapted Interaction* 1(3): 203–258.

[Kau04] Kauffman D. F. (2004) Self-regulated learning in web-based environments: Instructional tools designed to facilitate cognitive strategy use, metacognitive processing, and motivational beliefs. *Journal of educational computing research* 30(1): 139–161.

[Kav04] Kavcic A. (2004) Fuzzy student model in intermediactor platform. In *Information Technology Interfaces, 2004. 26th International Conference on*, pp. 297–302. IEEE.

[KCTL03] Kuo C.-H., Chou T.-C., Tsao N.-L., and Lan Y.-H. (2003) Canfind-a semantic image indexing and retrieval system. In *Circuits and Systems, 2003. ISCAS'03. Proceedings of the 2003 International Symposium on*, volumen 2, pp. II–644. IEEE.

[KDB03] Kosba E., Dimitrova V., and Boyle R. (2003) Using fuzzy techniques to model students in web-based learning environments. In *Knowledge-Based Intelligent Information and Engineering Systems*, pp. 222–229. Springer.

[Ker03] Kerner Y. H. (2003) Automatic extraction of keywords from abstracts. In *Proceedings of the 7th International Conference on Knowledge-Based Intelligent Information and Engineering Systems (KES 2003)*, pp. 843–849. Springer Berlin / Heidelberg, Oxford, UK.

[KEW01] Kwok C., Etzioni O., and Weld D. S. (2001) Scaling question answering to the web. *ACM Trans. Inf. Syst.* 19(3): 242–262.

[KGRS10] Kof L., Gacitua R., Rouncefield M., and Sawyer P. (2010) Concept mapping as a means of requirements tracing. In *Managing Requirements Knowledge (MARK), 2010 Third International Workshop on*, pp. 22–31. IEEE.

[Kha00] Khan B. H. (2000) Discussion of resources and attributes of the web for the creation of meaningful learning environments. *CyberPsychology and Behavior* 3(1): 17–23.

[KHGW07] Kuo B. Y., Hentrich T., Good B. M., and Wilkinson M. D. (2007) Tag clouds for summarizing web search results. In *Proceedings of the 16th international conference on World Wide Web*, pp. 1203–1204. New York, NY, USA.

[Kin11] Kinchin I. M. (2011) Visualising knowledge structures in biology: discipline, curriculum and student understanding. *Journal of Biological Education* 45(4): 183–189.

[Kir96] Kirkpatrick D. L. (1996) Techniques for evaluating training programs. *Classic writings on instructional technology* 1(192): 119.

[KK62] Kenney J. F. and Keeping E. S. (1962) *Linear Regression and Correlation.* Princeton, 3 edition.

[KKD14] Kurilovas E., Kubilinskiene S., and Dagiene V. (2014) Web 3.0–based personalisation of learning objects in virtual learning environments. *Computers in Human Behavior* 30: 654–662.

[KKK08] Khoury R., Karray F., and Kamel M. S. (2008) Keyword extraction rules based on a part-of-speech hierarchy. *International Journal of Advanced Media and Communication* 2(2): 138–153.

[KKR04] Kassim A. A., Kazi S. A., and Ranganath S. (2004) A web-based intelligent learning environment for digital systems. *International Journal of Engineering Education* 20(1): 13–23.

[KL97] Kommers P. and Lanzing J. (1997) Students' concept mapping for hypermedia design: Navigation through world wide web (www) space and self-assessment. *Journal of Interactive Learning Research* 8: 421–55.

[KL09] Koshman S. and Lu C.-J. (2009) Comparing visualization techniques to structure collaborative concepts. In *Collaborative Computing: Networking, Applications and Worksharing, 2009. CollaborateCom 2009. 5th International Conference on*, pp. 1–8.

[Kla86] Klapp O. E. (1986) *Overload and boredom: Essays on the quality of life in the information society.* Greenwood Publishing Group Inc.

[KLS07] Kim H., Lee H., and Seo J. (2007) A reliable faq retrieval system using a query log classification technique based on latent semantic analysis. *Information Processing & Management* 43(2): 420–430.

[KM10] Kaptein R. and Marx M. (2010) Focused retrieval and result aggregation with political data. *Information retrieval* 13(5): 412–433.

[KMV00] Kietz J.-U., Maedche A., and Volz R. (2000) A method for semi-automatic ontology acquisition from a corporate intranet. In *EKAW-2000 Workshop ?Ontologies and Text?, Juan-Les-Pins, France, October 2000.*

[Kos91] Kosko B. (1991) Neural networks and fuzzy systems: a dynamical systems approach to machine intelligence.

[Kos02] Koschmann T. (2002) Dewey's contribution to the foundations of cscl research. In *Proceedings of the Conference on Computer Support for Collaborative Learning: Foundations for a CSCL Community*, pp. 17–22. International Society of the Learning Sciences.

[KS04] Kalyuga S. and Sweller J. (2004) Measuring knowledge to optimize cognitive load factors during instruction. *Journal of educational psychology* 96(3): 558.

[KS05] Kavalec M. and Svaték V. (2005) A study on automated relation labelling in ontology learning. *Ontology Learning from Text: Methods, evaluation and applications* 123: 44.

[KT05] Keller T. and Tergan S.-O. (2005) Visualizing knowledge and information: An introduction. In *Knowledge and information visualization*, pp. 1–23. Springer.

[KTGS93] Kaplan R. M., Trenholm H., Gitomer D., and Steinberg L. (1993) A generalizable architecture for building intelligent tutoring systems. In *Artificial Intelligence for Applications, 1993. Proceedings., Ninth Conference on*, page 458. IEEE.

[KV04] Kabassi K. and Virvou M. (2004) Personalised adult e-training on computer use based on multiple attribute decision making. *Interacting with computers* 16(1): 115–132.

[KVGP12] Kostons D., Van Gog T., and Paas F. (2012) Training self-assessment and task-selection skills: A cognitive approach to improving self-regulated learning. *Learning and Instruction* 22(2): 121–132.

[KWSB14] Kuo Y.-C., Walker A. E., Schroder K. E., and Belland B. R. (2014) Interaction, internet self-efficacy, and self-regulated learning as predictors of student satisfaction in online education courses. *The Internet and Higher Education* 20: 35–50.

[KY95] Klir G. and Yuan B. (1995) *Fuzzy sets and fuzzy logic*, volumen 4. Prentice Hall New Jersey.

[KZGM09] Koutrika G., Zadeh Z. M., and Garcia-Molina H. (2009) Coursecloud: summarizing and refining keyword searches over structured data. In *Proceedings of the 12th International Conference on Extending Database Technology: Advances in Database Technology*, pp. 1132–1135. New York, NY, USA.

[Lap07] Lapham A. (2007) Creativity through e-learning: Engendering collaborative creativity through folksonomy. In *Proceedings of the 6th European Conference on E-Learning*, pp. 379–390.

[LB97] Lenz M. and Burkhard H. (1997) Cbr for document retrieval: The fallq project. In *Case-Based Reasoning Research and Development*, volumen 1266 of *Lecture Notes in Computer Science*, pp. 84–93.

[LBB04] Lilley M., Barker T., and Britton C. (2004) The development and evaluation of a software prototype for computer-adaptive testing. *Computers & Education* 43(1): 109–123.

[LC07] Lee S. O. K. and Chun A. H. W. (2007) Automatic tag recommendation for the web 2.0 blogosphere using collaborative tagging and hybrid ann semantic structures. *ACOS* 7: 88–93.

[LCS08] Law Y.-k., Chan C. K., and Sachs J. (2008) Beliefs about learning, self-regulated strategies and text comprehension among chinese children. *British Journal of Educational Psychology* 78(1): 51–73.

[LDGS$^+$13] Lops P., De Gemmis M., Semeraro G., Musto C., and Narducci F. (2013) Content-based and collaborative techniques for tag recommendation: an empirical evaluation. *Journal of Intelligent Information Systems* 40(1): 41–61.

[LDLD12] Leginus M., Dolog P., Lage R., and Durao F. (2012) Methodologies for improved tag cloud generation with clustering. In *Web Engineering*, volumen 7387 of *Lecture Notes in Computer Science*, pp. 61–75. Springer Berlin Heidelberg.

[LG12] Laal M. and Ghodsi S. M. (2012) Benefits of collaborative learning. *Procedia-Social and Behavioral Sciences* 31: 486–490.

[LH08] Lengyel P. and Herdon M. (2008) E-learning course development in moodle. In *Proceeding of the International Conference BIOATLAS*.

[LH12] Lin J.-L. and Hwang K.-S. (2012) An automatic classification system of online e-learning resources. In *System Science and Engineering (ICSSE), 2012 International Conference on*, pp. 163–166. IEEE.

[LHC11] Liu K., Hogan W. R., and Crowley R. S. (2011) Natural language processing methods and systems for biomedical ontology learning. *Journal of biomedical informatics* 44(1): 163–179.

[LICC12] Lee L. H., Isa D., Choo W. O., and Chue W. Y. (2012) High relevance keyword extraction facility for bayesian text classification on different domains of varying characteristic. *Expert System with Applications* 39(1): 1147–1155.

[Liu06] Liu C.-L. (2006) Using bayesian networks for student modeling. *Cognitively Informed Systems: Utilizing Practical Approaches to Enrich Information Presentation and Transfer* pp. 282–309.

[LL08] Litvak M. and Last M. (2008) Graph-based keyword extraction for single-document summarization. In *Proceedings of the Workshop on Multi-source Multilingual Information Extraction and Summarization (MMIES '08)*, pp. 17–24. Association for Computational Linguistics, Manchester, UK.

[LLD12] Liu C. L., Lee C. H., and Ding B. Y. (2012) Intelligent computer assisted blog writing system. *Expert System with Applications* 39(1): 4496–4504.

[LLL08] Lin Y.-M., Lin G.-Y., and Laffey J. M. (2008) Building a social and motivational framework for understanding satisfaction in online learning. *Journal of Educational Computing Research* 38(1): 1–27.

[LLL09] Lee C.-H., Lee G.-G., and Leu Y. (2009) Application of automatically constructed concept map of learning to conceptual diagnosis of e-learning. *Expert Systems with Applications* 36(2): 1675–1684.

[LLL10] Liu H., Lin X., and Liu C. (2010) Research and implementation of ontological qa system based on faq. *Journal of Convergence Information Technology* 5(3): 79–85.

[LLM07] Limayem M., Laferrière T., and Mantha R. (2007) Integrating ict into higher education: A study of onsite vs online students' perceptions. *Academy of educational leadership journal* 11(2).

[LLYC11] Liu C. L., Lee C. H., Yu S. H., and Chen C. W. (2011) Computer assisted writing system. *Expert System with Applications* 38(1): 804–811.

[LM04] Lammari N. and Métais E. (2004) Building and maintaining ontologies: a set of algorithms. *Data & Knowledge Engineering* 48(2): 155–176.

[LM11] Lipczak M. and Milios E. (2011) Efficient tag recommendation for real-life data. *ACM Transactions on Intelligent Systems and Technology (TIST)* 3(1): 2.

[LN13] Leksin V. A. and Nikolenko S. I. (2013) Semi-supervised tag extraction in a web recommender system. In *Similarity Search and Applications*, pp. 206–212. Springer.

[LNFIGTMF13] Llamas-Nistal M., Fernández-Iglesias M. J., González-Tato J., and Mikic-Fonte F. A. (2013) Blended e-assessment: Migrating classical exams to the digital world. *Computers & Education* 62: 72–87.

[LNH$^+$13] Lotfi Z., Nasaruddin M., Hanum F., Sahran S., and Mukhtar M. (2013) Collaborative e-learning tool for secondary schools. *Journal of Applied Sciences* 13: 22–35.

[Lou01] Lourtie P. (2001) Furthering the bologna process. report to the ministers of education of the signatory countries. *Prague, May* .

[LP01] Lin D. and Pantel P. (2001) Induction of semantic classes from natural language text. In *Proceedings of the seventh ACM SIGKDD international conference on Knowledge discovery and data mining*, pp. 317–322. ACM.

[LR10] Lazakidou G. and Retalis S. (2010) Using computer supported collaborative learning strategies for helping students acquire self-regulated problem-solving skills in mathematics. *Computers & Education* 54(1): 3–13.

[LS10] Limniou M. and Smith M. (2010) Teachers' and students' perspectives on teaching and learning through virtual learning environments. *European Journal of Engineering Education* 35(6): 645–653.

[LS12] Lee J. H. and Segev A. (2012) Knowledge maps for e-learning. *Computers and Education* 59(2): 353–364.

[LSL+09] Lau R. Y., Song D., Li Y., Cheung T. C., and Hao J.-X. (2009) Toward a fuzzy domain ontology extraction method for adaptive e-learning. *Knowledge and Data Engineering, IEEE Transactions on* 21(6): 800–813.

[LSLW12] Liu X., Song Y., Liu S., and Wang H. (2012) Automatic taxonomy construction from keywords. In *Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining*, pp. 1433–1441. ACM.

[LVK+07] Lemnitzer L., Vertan C., Killing A., Simov K., Evans D., Cristea D., and Monachesi P. (2007) Improving the search for learning objects with keywords and ontologies. In *Creating New Learning Experiences on a Global Scale*, pp. 202–216. Springer.

[LWGH05] Liu C.-L., Wang C.-H., Gao Z.-M., and Huang S.-M. (2005) Applications of lexical information for algorithmically composing multiple-choice cloze items. In *Proceedings of the second workshop on Building Educational Applications Using NLP*, pp. 1–8. Association for Computational Linguistics.

[LWW+05] Lu C.-H., Wu C.-W., Wu S.-H., Chiou G.-F., and Hsu W.-L. (2005) Ontological support in modeling learners' problem solving process. *Educational Technology & Society* 8(4): 64–74.

[LWZ+12] Liu J., Wang J., Zheng Q., Zhang W., and Jiang L. (2012) Topological analysis of knowledge maps. *Knowledge-Based Systems* 36(0): 260–267.

[LXCH09] Liao Z., Xie M., Cao H., and Huang Y. (2009) A probabilistic ranking approach for tag recommendation. *ECML PKDD Discovery Challenge 2009 (DC09)* page 143.

[LYCH09] Lu Y.-T., Yu S.-I., Chang T.-C., and Hsu J. Y.-j. (2009) A content-based method to enhance tag recommendation. In *IJCAI*, volumen 9, pp. 2064–2069.

[LYL09] Lee B.-C., Yoon J.-O., and Lee I. (2009) Learners? acceptance of e-learning in south korea: Theories and results. *Computers & Education* 53(4): 1320–1329.

[LYO09] Liu Y., Yin C., and Ogata H. (2009) Supporting q&a in a web-based japanese language learning environment. In *Information Engineering, 2009. ICIE'09. WASE International Conference on*, volumen 2, pp. 422–425. IEEE.

[LYO+11] Liu Y., Yin C., Ogata H., Qiao G., and Yano Y. (2011) A faq-based e-learning environment to support japanese language learning. *International Journal of Distance Education Technologies (IJDET)* 9(3): 45–55.

[MA08] Moos D. C. and Azevedo R. (2008) Exploring the fluctuation of motivation and use of self-regulatory processes during learning with hypermedia. *Instructional Science* 36(3): 203–231.

[MAHK06] Mitkov R., An Ha L., and Karamanis N. (2006) A computer-aided environment for generating multiple-choice test items. *Natural Language Engineering* 12(02): 177–194.

[May02] Mayer R. E. (2002) Multimedia learning. *Psychology of Learning and Motivation* 41: 85–139.

[MC07] Mihalcea R. and Csomai A. (2007) Wikify!: linking documents to encyclopedic knowledge. In *Proceedings of the ACM 16th International Conference on Information and Knowledge Management (CIKM'07)*, pp. 233–242. Lisbon, Portugal.

[MC11] Martínez-Caro E. (2011) Factors affecting effectiveness in e-learning: An analysis in production management courses. *Computer Applications in Engineering Education* 19(3): 572–581.

[McD02] McDonald A. S. (2002) The impact of individual differences on the equivalence of computer-based and paper-and-pencil educational assessments. *Computers & Education* 39(3): 299–312.

[McG10] McGhee R. M. H. (2010) Asynchronous interaction, online technologies self-efficacy and self-regulated learning as predictors of academic achievement in an online class. *ProQuest LLC* .

[MDR90] McKnight C., Dillon A., and Richardson J. (1990) A comparison of linear and hypertext formats in information retrieval.

[Men95] Mendel J. M. (1995) Fuzzy logic systems for engineering: a tutorial. *Proceedings of the IEEE* 83(3): 345–377.

[MFDCC08] Martins C., Faria L., De Carvalho C. V., and Carrapatoso E. (2008) User modeling in adaptive hypermedia educational systems. *Educational Technology & Society* 11(1): 194–207.

[MFH14] Meijer K., Frasincar F., and Hogenboom F. (2014) A semantic approach for extracting domain taxonomies from text. *Decision Support Systems* 62: 78–93.

[MFW09] Medelyan O., Frank E., and Witten I. H. (2009) Human-competitive tagging using automatic keyphrase extraction. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing: Volume 3-Volume 3*, pp. 1318–1327. Association for Computational Linguistics.

[MHSGB10] Montero Y. H., Herrero-Solana V., and Guerrero-Bote V. (2010) Usabilidad de los tag-clouds: estudio mediante eye-tracking. *Scire: representación y organización del conocimiento* 16(1): 15–33.

[MI04] Matsuo Y. and Ishizuka M. (2004) Keyword extraction from a document using word co-occurrence statistical information. *International Journal on Artificial Intelligence Tools* 13(1): 157–170.

[Mil95a] Miller G. A. (1995) Wordnet: a lexical database for english. *Commun. ACM* 38(11): 39–41.

[Mil95b] Miller G. A. (1995) Wordnet: a lexical database for english. *Communications of the ACM* 38(11): 39–41.

[Mil08] Miller C. T. (2008) Enhancing web-based instruction using a person-centered model of instruction. *Quarterly Review of Distance Education: Research That Guides Practice* 8(1): 25.

[Mis06] Mishne G. (2006) Autotag: a collaborative approach to automated tag assignment for weblog posts. In *Proceedings of the 15th international conference on World Wide Web*, pp. 953–954. ACM.

[Mis11] Mistar J. (2011) A study of the validity and reliability of self-assessment. *TEFLIN Journal: A publication on the teaching and learning of English* 22(1).

[Mit03] Mitrovic A. (2003) An intelligent sql tutor on the web. *International Journal of Artificial Intelligence in Education* 13(2): 173–197.

[MJ97] McCormack C. and Jones D. (1997) *Building a web-based education system.* John Wiley & Sons, Inc.

[MJ10] Mendenhall A. and Johnson T. E. (2010) Fostering the development of critical thinking skills, and reading comprehension of undergraduates using a web 2.0 tool coupled with a learning system. *Interactive Learning Environments* 18(3): 263–276.

[MJSIdBC13] Marín Juarros V., Salinas Ibáñez J., and de Benito Crosetti B. (2013) Research results of two personal learning environments experiments in a higher education institution. *Interactive Learning Environments* (ahead-of-print): 1–16.

[MKSG06] Merino P. J. M., Kloos C. D., Seepold R., and García R. M. C. (2006) Rating the importance of different lms functionalities. In *Frontiers in Education Conference, 36th Annual*, pp. 13–18. IEEE.

[MKWS12] Morik K., Kaspari A., Wurst M., and Skirzynski M. (2012) Multi-objective frequent termset clustering. *Knowledge and information systems* 30(3): 715–738.

[MM99] Miller S. M. and Miller K. L. (1999) Using instructional theory to facilitate communication in web-based courses. *Educational Technology & Society* 2(3): 106–114.

[MM01] Mayo M. and Mitrovic A. (2001) Optimising its behaviour with bayesian networks and decision theory. *International Journal of Artificial Intelligence in Education* 12: 124–153.

[MMKL$^+$10] Muñoz K., Mc Kevitt P., Lunney T., Noguez J., and Neri L. (2010) Playphysics: an emotional games learning environment for teaching physics. In *Knowledge Science, Engineering and Management*, pp. 400–411. Springer.

[MNCZ12] Moreo A., Navarro M., Castro J., and Zurita J. (2012) A high-performance faq retrieval method using minimal differentiator expressions. *Knowledge-Based Systems* 36: 9–20.

[MNI10] Milicevic A. K., Nanopoulos A., and Ivanovic M. (2010) Social tagging in recommender systems: a survey of the state-of-the-art and possible extensions. *Artificial Intelligence Review* 33(3): 187–209.

[MOKY02] Mitsuhara H., Ochi Y., Kanenishi K., and Yano Y. (May 2002) An adaptive web-based learning system with a free-hyperlink environment. In *Proceedings of Workshop on Adaptive Systems for Web-Based Education at the 2nd International Conference on Adaptive Hypermedia and Adaptive Web-Based Systems, AH'2002*, pp. 81–91. Málaga, Spain.

[Moo89] Moore M. G. (1989) Editorial: Three types of interaction. *American Journal of Distance Education* 3(2): 1–7.

[Moo13] Moodle C. (2013) Gift format [online]. *Available* `http://docs.moodle.org/27/en/GIFT_format`.

[Moo14] Moodle (2014) Moodle statistics [online]. *Available* `https://moodle.org/stats/`.

[MPDLC02] Millán E. and Pérez-De-La-Cruz J. L. (2002) A bayesian diagnostic algorithm for student modeling and its evaluation. *User Modeling and User-Adapted Interaction* 12(2-3): 281–330.

[MPS03] Maedche A., Pekar V., and Staab S. (2003) Ontology learning part one?on discovering taxonomic relations from the web. In *Web Intelligence*, pp. 301–319. Springer.

[MR13] Matcha W. and Rambli D. R. A. (2013) Exploratory study on collaborative interaction through the use of augmented reality in science learning. *Procedia Computer Science* 25: 144–153.

[MRCZ12] Moreo A., Romero M., Castro J. L., and Zurita J. M. (2012) Faqtory: A framework to provide high-quality faq retrieval systems. *Expert Systems with Applications* 39(14): 15–29.

[MS99] Manning C. D. and Schütze H. (1999) *Foundations of statistical natural language processing*. MIT press.

[MS01] Maedche A. and Staab S. (2001) Ontology learning for the semantic web. *IEEE Intelligent systems* 16(2): 72–79.

[MSL11] Minguillón J., Sicilia M.-A., and Lamb B. (2011) From content management to e-learning content repositories. In *Content Management for E-Learning*, pp. 27–41. Springer.

[MT08] Mihalcea R. and Tarau P. (2008) Textrank: Bringing order into texts. In *Proceedings of the 2004 Conference on Empirical Methods in Natural Language Processing (EMNLP 2004)*, pp. 404–411. Association for Computational Linguistics, Barcelona, Spain.

[MTTMG$^+$08] Martínez-Torres M. R., Toral Marín S., Garcia F. B., Vazquez S. G., Oliva M. A., and Torres T. (2008) A technological acceptance of e-learning tools used in practical and laboratory teaching, according to the european higher education area 1. *Behaviour & Information Technology* 27(6): 495–505.

[MW97] McCombs B. L. and Whisler J. S. (1997) *The Learner-Centered Classroom and School: Strategies for Increasing Student Motivation and Achievement. The Jossey-Bass Education Series.* ERIC.

[MW13] Miotto R. and Weng C. (2013) Unsupervised mining of frequent tags for clinical eligibility text indexing. *Journal of biomedical informatics* 46(6): 1145–1151.

[MWM08] Medelyan O., Witten I. H., and Milne D. (2008) Topic indexing with wikipedia. In *Proceedings of the first AAAI Workshop on Wikipedia and Artificial Intelligence (WIKIAI'08)*, volumen 1, pp. 19–24. Chicago, I.L.

[MYRP07] Meng A., Ye L., Roy D., and Padilla P. (2007) Genetic algorithm based multi-agent system applied to test generation. *Computers & Education* 49(4): 1205–1223.

[NBM10] Nkambou R., Bourdeau J., and Mizoguchi R. (2010) *Advances in intelligent tutoring systems*, volumen 308, chapter 1. Springer.

[NMD06] Nicol D. J. and Macfarlane-Dick D. (2006) Formative assessment and self-regulated learning: a model and seven principles of good feedback practice. *Studies in higher education* 31(2): 199–218.

[Nor06] Noruzi A. (2006) Folksonomies:(un) controlled vocabulary? *Knowledge Organization* 33(4): 199–203.

[Nov84] Novak J. D. (1984) *Learning how to learn.* Cambridge University Press.

[Nov90] Novak J. D. (1990) Concept mapping: A useful tool for science education. *Journal of research in science teaching* 27(10): 937–949.

[NVCN04] Navigli R., Velardi P., Cucchiarelli A., and Neri F. (2004) Quantitative and qualitative evaluation of the ontolearn ontology learning system. In *Proceedings of the 20th international conference on Computational Linguistics*, page 1043. Association for Computational Linguistics.

[NWM08] Nielsen R. D., Ward W., and Martin J. H. (2008) Soft computing in intelligent tutoring systems and educational assessment. In *Soft Computing Applications in Business*, pp. 201–230. Springer.

[OC11] Oncu S. and Cakir H. (2011) Research in online learning environments: Priorities and methodologies. *Computers & Education* 57(1): 1098–1108.

[OHC$^+$12] Orlich D., Harder R., Callahan R., Trevisan M., and Brown A. (2012) *Teaching strategies: A guide to effective instruction.* Cengage Learning.

[Ohl86] Ohlsson S. (1986) Some principles of intelligent tutoring. *Instructional science* 14(3-4): 293–326.

[OJK14] Oikarinen J. K., Järvelä S., and Kaasila R. (2014) Finnish upper secondary students? collaborative processes in learning statistics in a cscl environment. *International Journal of Mathematical Education in Science and Technology* 45(3): 325–348.

[Ort90] Ortony A. (1990) *The cognitive structure of emotions.* Cambridge university press.

[PAG09] Piraquive F. N. D., Aguilar L. J., and García V. H. M. (2009) Taxonomía, ontología y folksonomía,¿ qué son y qué benefi cios u oportunidades presentan para los usuarios de la web?. *Revista Universidad & Empresa* 11(16).

[Pan99] Panitz T. (1999) The motivational benefits of cooperative learning. *New directions for teaching and learning* 1999(78): 59–67.

[Pas07] Passant A. (2007) Using ontologies to strengthen folksonomies and enrich information retrieval in weblogs. In *Proceedings of International Conference on Weblogs and Social Media.*

[PBR07] Perry S., Bulatov I., and Roberts E. (2007) The use of e-assessment in chemical engineering education. *CHEMICAL ENGINEERING* 12.

[PDB+10] Pudota N., Dattolo A., Baruzzo A., Ferrara F., and Tasso C. (2010) Automatic keyphrase extraction and ontology mining for content-based tag recommendation. *International Journal of Intelligent Systems* 25(12): 1158–1186.

[Pea88] Pearl J. (1988) *Probabilistic reasoning in intelligent systems: networks of plausible inference*. Morgan Kaufmann.

[Pea96] Pearl J. (1996) Decision making under uncertainty. *ACM Computing Surveys (CSUR)* 28(1): 89–92.

[PFGP07] Pekrun R., Frenzel A. C., Goetz T., and Perry R. P. (2007) *The control-value theory of achievement emotions: An integrative approach to emotions in education.* Bibliothek der Universität Konstanz.

[PFM06] Panunzi A., Fabbri M., and Moneglia M. (2006) Integrating methods and lrs for automatic keyword extraction from open domain texts. *Proceedings of the 5th international language resources and evaluation (LREC)* pp. 1917–1920.

[PHA+09] Pramitasari L., Hidayanto A. N., Aminah S., Krisnadhi A. A., and Ramadhanie A. (2009) Development of student model ontology for personalization in an e-learning system based on semantic web. In *International Conference on Advanced Computer Science and Information Systems (ICACSIS09), Indonesia, December*, pp. 7–8. Citeseer.

[Pin99] Pintrich P. R. (1999) The role of motivation in promoting and sustaining self-regulated learning. *International journal of educational research* 31(6): 459–470.

[Pin03] Pintrich P. R. (2003) A motivational science perspective on the role of student motivation in learning and teaching contexts. *Journal of educational Psychology* 95(4): 667.

[PNPH08] Paukkeri M.-S., Nieminen I., Polla M., and Honkela T. (2008) A language-independent approach to keyphrase extraction and evaluation. In *Proceedings of The 22nd International Conference on Computational Linguistics (COLING'08)*, pp. 83–86. Manchester, UK.

[Pri99] Pritchett N. (1999) Effective question design. *Brown S, Race P and Bull J (eds.), Computer-Assisted Assessment in Higher Education, London, Kogan Page* .

[Puz08] Puzziferro M. (2008) Online technologies self-efficacy and self-regulated learning as predictors of final grade and satisfaction in college-level online courses. *The Amer. Jrnl. of Distance Education* 22(2): 72–89.

[Qui05] Quintarelli E. (2005) Folksonomies: power to the people. In *ISKO Italy-UniMIB meeting*.

[RAM03] Razmerita L., Angehrn A., and Maedche A. (2003) Ontology-based user modeling for knowledge management systems. In *User Modeling 2003*, volumen 2702 of *Lecture Notes in Computer Science*, pp. 148–148. Springer Berlin / Heidelberg.

[RGC+13] Romero L., Gutierrez M., Caliusco M., *et al.* (2013) A conceptualization of e-assessment domain. In *Information Systems and Technologies (CISTI), 2013 8th Iberian Conference on*, pp. 1–6. IEEE.

[RGMM07] Rivadeneira A. W., Gruen D. M., Muller M. J., and Millen D. R. (2007) Getting our head in the clouds: toward evaluation studies of tagclouds. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pp. 995–998. New York, NY, USA.

[Ric79] Rich E. (1979) User modeling via stereotypes. *Cognitive science* 3(4): 329–354.

[Riv89] Rivers R. (1989) Embedded user models —where next? *Interacting with Computers* 1(1): 13–30.

[RM02] Rosenfeld L. and Morville P. (2002) *Information architecture for the world wide web*. O'Reilly Media, Inc.

[RM03] Rehak D. and Mason R. (2003) Keeping the learning in learning objects. *Reusing online resources: A sustainable approach to e-learning* pp. 20–34.

[RMC13] Romero M., Moreo A., and Castro J. L. (September 2013) Collaborative system for learning based on questionnaires and tasks. In *4th International Conference on European Transnational Education, (ICEUTE'13)*, pp. 631–640. Springer International Publishing, Salamanca, Spain.

[RMVGFB+11] Ruiz-Martínez J. M., Valencia-García R., Fernández-Breis J. T., García-Sánchez F., and Martínez-Béjar R. (2011) Ontology learning from biomedical natural language documents using umls. *Expert Systems with Applications* 38(10): 12365–12378.

[Rou96] Rouet J. (1996) *Hypertext and cognition*. Psychology Press.

[RT02] Rajaraman K. and Tan A.-H. (2002) Knowledge discovery from texts: a concept frame graph approach. In *Proceedings of the eleventh international conference on Information and knowledge management*, pp. 669–671. ACM.

[RWW+08] Reigeluth C. M., Watson W. R., Watson S. L., Dutta P., Chen Z., and Powell N. D. (2008) Roles for technology in the information-age paradigm of education: Learning management systems. *Educational Technology* 48(6): 32.

[SA11] Skoutas D. and Alrifai M. (2011) Tag clouds revisited. In *Proceedings of the 20th ACM international conference on Information and knowledge management*, CIKM '11, pp. 221–230.

[SAB03] Shamsfard M. and Abdollahzadeh Barforoush A. (2003) The state of the art in ontology learning: a framework for comparison. *The Knowledge Engineering Review* 18(04): 293–316.

[Sab05] Sabou M. (2005) Learning web service ontologies: an automatic extraction method and its evaluation. *Ontology learning from text: methods, evaluation and applications* 123.

[Sad89] Sadler D. R. (1989) Formative assessment and the design of instructional systems. *Instructional science* 18(2): 119–144.

[Sal90] Salomon G. (1990) Studying the flute and the orchestra: controlled vs. classroom research on computers. *International Journal of Educational Research* 14(6): 521–531.

[Sán09] Sánchez D. (2009) Domain ontology learning from the web. *The Knowledge Engineering Review* 24(04): 413–413.

[SB82] Sleeman D. and Brown J. (1982) Intelligent tutoring systems. *Computer* .

[SB88] Salton G. and Buckley C. (1988) Term weighting approaches in automatic text retrieval. *Information Processing and Management* 24(5): 513–523.

[SB04] Soricut R. and Brill E. (2004) Automatic question answering: beyond the factoid. In *Proceedings of Human Language Technology conference/North American chapter of the Association for Computational Linguistics annual meeting*. Boston.

[SBM$^+$05] Su B., Bonk C. J., Magjuka R. J., Liu X., and Lee S.-h. (2005) The importance of interaction in web-based education: A program-level case study of online mba courses. *Journal of Interactive Online Learning* 4(1): 1–19.

[SCCC11] Shih C.-W., Chen M.-Y., Chu H.-C., and Chen Y.-M. (2011) Enhancement of domain ontology construction using a crystallizing approach. *Expert Systems with Applications* 38(6): 7544–7557.

[SCD00] Stanton N., Correia A. P., and Dias P. (2000) Efficacy of a map on search, orientation and access behaviour in a hypermedia system. *Computers & Education* 35(4): 263–279.

[SCG76] Stansfield J. L., Carr B. P., and Goldstein I. P. (1976) Wumpus advisor 1. a first implementation of a program that tutors logical and probabilistic reasoning skills. Technical report, DTIC Document.

[Sch90] Schlechty P. (1990) *Schools for the 21st century*. San Francisco: Jossey-Bass.

[Sch91] Schunk D. H. (1991) Self-efficacy and academic motivation. *Educational psychologist* 26(3-4): 207–231.

[Sch05] Schunk D. H. (2005) Commentary on self-regulation in school contexts. *Learning and Instruction* 15(2): 173–177.

[SCH08] Sinclair J. and Cardew-Hall M. (2008) The folksonomy tag cloud: when is it useful? *Journal of Information Science* 34(1): 15–29.

[Sch12] Schunk D. H. (2012) *Learning Theories: An Educational Perspective (6th Edition)*. Pearson Education, Inc.

[Sci02] Sciuto G. T. (2002) 10. setting students up for success: The instructor?s role in creating a positive, asynchronous, distance education experience. *Motivating & Retaining Adult Learners Online* page 108.

[SCP12] Subramaniyaswamy V. and Chenthur Pandian S. (2012) Effective tag recommendation system based on topic ontology using wikipedia and wordnet. *International Journal of Intelligent Systems* 27(12): 1034–1048.

[SDMVM06] Spyns P., De Moor A., Vandenbussche J., and Meersman R. (2006) From folksologies to ontologies: How the twain meet. In *On the move to meaningful internet systems 2006: CoopIS, DOA, GADA, and ODBASE*, pp. 738–755. Springer.

[Sel90] Self J. A. (1990) Bypassing the intractable problem of student modelling. *Intelligent tutoring systems: At the crossroads of artificial intelligence and education* pp. 107–123.

[SFK00] Stamatatos E., Fakotakis N., and Kokkinakis G. (2000) Text genre detection using common word frequencies. In *Proceedings of the 18th Conference on Computational Linguistics (COLING '00)*, pp. 808–814. Association for Computational Linguistics, Saarbrücken, Germany.

[SGA08] Schiaffino S., Garcia P., and Amandi A. (2008) eteacher: Providing personalized assistance to e-learning students. *Computers & Education* 51(4): 1744–1754.

[SGTN12] Shakouri G H. and Tavassoli N Y. (2012) Implementation of a hybrid fuzzy system as a decision support process: A fahp–fmcdm–fis composition. *Expert Systems with Applications* 39(3): 3682–3691.

[SH06] Salim N. and Haron N. (2006) The construction of fuzzy set and fuzzy rule for mixed approach in adaptive hypermedia learning system. In *Technologies for E-Learning and Digital Entertainment*, pp. 183–187. Springer.

[SH08] Schaffert S. and Hilzensauer W. (2008) On the way towards personal learning environments: Seven crucial aspects. *Elearning papers* (9): 2.

[SH13] Shrivastav H. and Hiltz S. R. (2013) Information overload in technology-based education: a meta-analysis.

[SHC10] SHIH Y.-C., HSUa Y.-C., and CHENab S. Y. (2010) Mining learners? disorientation in web-based learning. In *Workshop Proceedings of the 18th International Conference on Computers in Education: ICCE2010*, page 135.

[She09] Sher A. (2009) Assessing the relationship of student-instructor and student-student interaction to student learning and satisfaction in web-based online learning environment. *Journal of Interactive Online Learning* 8(2): 102–120.

[Shn96] Shneiderman B. (1996) The eyes have it: A task by data type taxonomy for information visualizations. In *Visual Languages, 1996. Proceedings., IEEE Symposium on*, pp. 336–343. IEEE.

[Shu08] Shute V. J. (2008) Focus on formative feedback. *Review of educational research* 78(1): 153–189.

[Sim70] Simpson E. J. (1970) *The classification of educational objectives, psychomotor domain.* Department of Health, Education, and Welfare, Office of Edcn.

[Siv14] Sivaraman K. (2014) Effective web based e-learning. *Middle-East Journal of Scientific Research* 19(8): 1024–1027.

[SJN04] Snow R., Jurafsky D., and Ng A. Y. (2004) Learning syntactic patterns for automatic hypernym discovery. *Advances in Neural Information Processing Systems 17* .

[Ski60] Skinner B. F. (1960) Teaching machines. *The Review of Economics and Statistics* pp. 189–191.

[SKS06] Stahl G., Koschmann T., and Suthers D. (2006) Computer-supported collaborative learning: An historical perspective. *Cambridge handbook of the learning sciences* 2006.

[SLA13] Seitlinger P., Ley T., and Albert D. (2013) An implicit-semantic tag recommendation mechanism for socio-semantic learning systems. In *Open and Social Technologies for Networked Learning*, pp. 41–46. Springer.

[SLP14] Shin Y., Lee S.-J., and Park J. (2014) Composition pattern oriented tag extraction from short documents using a structural learning method. *Knowledge and Information Systems* 38(2): 447–468.

[SM86] Salton G. and Mcgill M. J. (1986) *Introduction to Modern Information Retrieval.* McGraw-Hill, Inc.

[SM03] Surjono H. D. and Maltby J. R. (2003) Adaptive educational hypermedia based on multiple student characteristics. In *Advances in Web-Based Learning-ICWL 2003*, pp. 442–449. Springer.

[SMGS05] Stathacopoulou R., Magoulas G. D., Grigoriadou M., and Samarakou M. (2005) Neuro-fuzzy knowledge processing in intelligent learning environments for improved student diagnosis. *Information Sciences* 170(2): 273–307.

[Sne83] Snelbecker G. (1983) *Learning theory, instructional theory, and psychoeducational design.* New York: McGraw-Hill.

[Sne99] Sneiders E. (1999) Automated faq answering: Continued experience with shallow language understanding. Technical report, AAAI Technical Report FS-99-02.

[Sne02] Sneiders E. (2002) Automated question answering using question templates that cover the conceptual model of the database. In *Natural Language Processing and Information Systems*, volumen 2553 of *Lecture Notes in Computer Science*, pp. 235–239.

[SPBVM04] Salden R. J., Paas F., Broers N. J., and Van Merriënboer J. J. (2004) Mental effort and performance as determinants for the dynamic selection of learning tasks in air traffic control training. *Instructional science* 32(1-2): 153–172.

[SR96] Scaife M. and Rogers Y. (1996) External cognition: how do graphical representations work? *International journal of human-computer studies* 45(2): 185–213.

[SRŽG08] Stankov S., Rosić M., Žitko B., and Grubišić A. (2008) Tex-sys model for building intelligent tutoring systems. *Computers & Education* 51(3): 1017–1036.

[Ste09] Sternberg R. J. (2009) *Cognitive psychology.* Cengage Learning.

[Sun02] Sundblad H. (2002) Automatic acquisition of hyponyms and meronyms from question corpora. *OLT2002, France* .

[Sup66] Suppes P. (1966) *The uses of computers in education.* Freeman.

[SVMP98] Sweller J., Van Merrienboer J. J., and Paas F. G. (1998) Cognitive architecture and instructional design. *Educational psychology review* 10(3): 251–296.

[SVZ08] Sigurbjörnsson B. and Van Zwol R. (2008) Flickr tag recommendation based on collective knowledge. In *Proceedings of the 17th international conference on World Wide Web*, pp. 327–336. ACM.

[Swa02] Swan K. (2002) Building learning communities in online courses: The importance of interaction. *Education, Communication & Information* 2(1): 23–49.

[Sym09] Symeonidis P. (2009) User recommendations based on tensor dimensionality reduction. In *Artificial Intelligence Applications and Innovations III*, pp. 331–340. Springer.

[SZ11] Stoeger H. and Ziegler A. (2011) Self-regulatory training through elementary-school students' homework completion. *Handbook of self-regulation of learning and performance* pp. 87–101.

[SZG11] Song Y., Zhang L., and Giles C. L. (2011) Automatic tag recommendation algorithms for social recommender systems. *ACM Transactions on the Web (TWEB)* 5(1): 4.

[Ter05] Tergan S.-O. (2005) Digital concept maps for managing knowledge and information. In *Knowledge and information visualization*, pp. 185–204. Springer.

[TGCK03] Tsaganou G., Grigoriadou M., Cavoura T., and Koutra D. (2003) Evaluating an intelligent diagnosis system of historical text comprehension. *Expert Systems with Applications* 25(4): 493–502.

[TH11] Tzeng G.-H. and Huang J.-J. (2011) *Multiple attribute decision making: methods and applications.* CRC Press.

[THE00] Tan K.-W., Han H., and Elmasri R. (2000) Web data cleansing and preparation for ontology extraction using wordnet. In *Web Information Systems Engineering, 2000. Proceedings of the First International Conference on*, volumen 2, pp. 11–18. IEEE.

[THP$^+$01] Tretiakov A., Hong H., Patel A., *et al.* (2001) Human teacher in intelligent tutoring system: a forgotten entity. In *Advanced Learning Technologies, 2001. Proceedings. IEEE International Conference on*, pp. 227–230. IEEE.

[TKT12] Takase H., Kawanaka H., and Tsuruoka S. (2012) Real time keyword extraction for e-learning system supporting quiz. In *Soft Computing and Intelligent Systems (SCIS) and 13th International Symposium on Advanced Intelligent Systems (ISIS), 2012 Joint 6th International Conference on*, pp. 222–225. IEEE.

[TM09] Thomson D. and Mitrovic A. (2009) Towards a negotiable student model for constraint-based itss. In *17th International on Conference Computers in Education (ICCE 2009).* University of Canterbury. Computer Science and Software Engineering, Hong Kong.

[TSD08] Tatu M., Srikanth M., and D'Silva T. (2008) Tag recommendations using bookmark content. *ECML PKDD discovery challenge* 2008: 96–107.

[Tur00] Turney P. D. (2000) Learning algorithms for keyphrase extraction. *Information Retrieval* 2(4): 303–336.

[TV02] Tsiriga V. and Virvou M. (2002) Initializing the student model using stereotypes and machine learning. In *IEEE International Conference on Systems, Man, and Cybernetics*. Citeseer.

[TV03] Tsiriga V. and Virvou M. (2003) Initializing student models in web-based itss: a generic approach. In *Advanced Learning Technologies, 2003. Proceedings. The 3rd IEEE International Conference on*, pp. 42–46. IEEE.

[TW04] Thurmond V. and Wambach K. (2004) Understanding interactions in distance education: A review of the literature. *International Journal of Instructional Technology and Distance Learning* 1(1).

[TWCF02] Thurmond V. A., Wambach K., Connors H. R., and Frey B. B. (2002) Evaluation of student satisfaction: Determining the impact of a web-based environment by controlling for student characteristics. *The American Journal of Distance Education* 16(3): 169–190.

[UKMJJ11] Uday Kumar M., Mamatha J., Jain S., and Jain D. (2011) Intelligent online assessment methodology. In *Next Generation Web Services Practices (NWeSP), 2011 7th International Conference on*, pp. 215–220. IEEE.

[VC09] Villalon J. and Calvo R. A. (2009) Concept extraction from student essays, towards concept map mining. In *Advanced Learning Technologies, 2009. ICALT 2009. Ninth IEEE International Conference on*, pp. 221–225. IEEE.

[VK02] Virvou M. and Kabassi K. (2002) F-smile: an intelligent multi-agent learning environment. In *Proceedings of 2002 IEEE International Conference on Advanced Learning Technologies-ICALT*. Citeseer.

[VKGM11a] Venetis P., Koutrika G., and Garcia-Molina H. (2011) On the selection of tags for tag clouds. In *Proceedings of the fourth ACM international conference on Web search and data mining*, WSDM '11, pp. 835–844.

[VKGM11b] Venetis P., Koutrika G., and Garcia-Molina H. (2011) On the selection of tags for tag clouds. In *Proceedings of the fourth ACM international conference on Web search and data mining*, WSDM '11, pp. 835–844.

[VLC02] Verdegay-López J. and Castro J. (2002) Gsadq: Incorporando información difusa o con incertidumbre al diseño automático de cuestionarios. In *Conferencia de la Asociación Española para la Inteligencia Artificial*.

[VM99] Vrasidas C. and McIsaac M. S. (1999) Factors influencing interaction in an online course. *American Journal of Distance Education* 13(3): 22–36.

[VNCN05] Velardi P., Navigli R., Cucchiarelli A., and Neri F. (2005) Evaluation of ontolearn, a methodology for automatic learning of domain ontologies. *Ontology Learning and Population* .

[VO09] Vuorikari R. and Ochoa X. (2009) Exploratory analysis of the main characteristics of tags and tagging of educational resources in a multi-lingual context. *Journal of Digital Information* 10(2).

[Voo01] Voorhees E. M. (2001) The trec question answering track. *Natural Language Engineering* 7: 361–378.

[VWF09] Viegas F. B., Wattenberg M., and Feinberg J. (2009) Participatory visualization with wordle. *Visualization and Computer Graphics, IEEE Transactions on* 15(6): 1137–1144.

[Wan11] Wang T.-H. (2011) Developing web-based assessment strategies for facilitating junior high school students to perform self-regulated learning in an e-learning environment. *Computers & Education* 57(2): 1801–1812.

[WB00] Welch M. and Brownell K. (2000) The development and evaluation of a multimedia course on educational collaboration. *Journal of Educational Multimedia and Hypermedia* 9(3): 169–94.

[WB$^+$01] Weber G., Brusilovsky P., *et al.* (2001) Elm-art: An adaptive versatile system for web-based instruction. *International Journal of Artificial Intelligence in Education (IJAIED)* 12: 351–384.

[WBG05] Winter M., Brooks C., and Greer J. (2005) Towards best practices for semantic web student modelling. In *Proceedings of the 2005 Conference on Artificial Intelligence in Education: Supporting Learning Through Intelligent and Socially Informed Technology*, pp. 694–701. IOS Press, Amsterdam, The Netherlands, The Netherlands.

[WBS10] Wartena C., Brussee R., and Slakhorst W. (2010) Keyword extraction using word co-occurrence. In *Workshop on Database and Expert Systems Applications (DEXA'10)*, pp. 54–58. Bilbao, Spain.

[WCLK08] Wang W., Cheung C. F., Lee W., and Kwok S. (2008) Mining knowledge from natural language texts using fuzzy associated concept mapping. *Information Processing & Management* 44(5): 1707–1719.

[Web98] Webb G. (1998) Preface to umuai special issue on machine learning for user modeling. *User Modeling and User-Adapted Interaction* 8(1): 1–3.

[Wei13] Weimer M. (2013) *Learner-centered teaching: Five key changes to practice.* John Wiley & Sons.

[Wel07] Weller M. (2007) *Virtual learning environments: Using, choosing and developing your VLE.* Routledge.

[Wen87] Wenger E. (1987) Artificial intelligence and tutoring systems.

[Whi95] Whitehead S. D. (1995) Auto-faq: an experiment in cyberspace leveraging. *Computer Networks and ISDN Systems* 28(18): 137–146.

[WHZC09] Wang P., Hu J., Zeng H.-J., and Chen Z. (2009) Using wikipedia knowledge to improve text classification. *Knowledge and Information Systems* 19(3): 265–281.

[Wie00] Wiersema N. (2000) How does collaborative learning actually work in a classroom and how do students react to it? a brief reflection. *Deliberations* 14: 2005.

[Wil03] Wiley D. A. (2003) *Connecting learning objects to instructional design theory: A definition, a metaphor, and a taxonomy.* D. A. Wiley (Ed.), The Instructional Use of Learning Objects. Bloominton: Agency for Instructional Technology and Association for Educational Communications & Technology.

[Win00] Winiwarter W. (2000) Adaptive natural language interfaces to faq knowledge bases. *Data & Knowledge Engineering* 35(2): 181–199.

[WLT09] Wenchao M., Lianchen L., and Ting D. (2009) A modified approach to keyword extraction based on word-similarity. In *Intelligent Computing and Intelligent Systems, 2009. ICIS 2009. IEEE International Conference on*, pp. 388–392. Shanghai.

[Wom07] Womble J. C. (2007) *E-learning: the relationship among learner satisfaction, self-efficacy, and usefulness.* ProQuest.

[WPB01] Webb G. I., Pazzani M. J., and Billsus D. (2001) Machine learning for user modeling. *User modeling and user-adapted interaction* 11(1-2): 19–29.

[WPC+11] Wang M., Peng J., Cheng B., Zhou H., and Liu J. (2011) Knowledge visualization for self-regulated learning. *Educational Technology & Society* 14(3): 28–42.

[WPF+99] Witten I. H., Paynte G. W., Frank E., Gutwin C., and Nevill-Manning C. G. (1999) Kea: Practical automatic keyphrase extraction. In *Proceedings of the 4th ACM Conference on Digital Library (DL'99)*, pp. 254–255. ACM, Berkeley, CA, USA.

[WPS10] Weller K., Peters I., and Stock W. G. (2010) Folksonomy. the collaborative knowledge organization system. *Handbook of research on social interaction technologies and collaborative software: Concepts and trends* pp. 132–146.

[WS98] Watts D. J. and Strogatz S. H. (1998) Collective dynamics of 'small-world' networks. *Nature* 393(6684): 440–442.

[WWY05] Wang Y.-H., Wang W.-N., and Yen Y.-H. (2005) An intelligent semantic agent for e-learning message communication. In *Advanced Information Networking and Applications, 2005. AINA 2005. 19th International Conference on*, volumen 2, pp. 105–108. IEEE.

[WYC05] Wu C., Yeh J., and Chen M. (2005) Domain-specific faq retrieval using independent aspects. *ACM Transactions on Asian Language Information Processing* 4(1): 1–17.

[WYW09] Wang X., Yang Y., and Wen X. (2009) Study on blended learning approach for english teaching. In *Systems, Man and Cybernetics, 2009. SMC 2009. IEEE International Conference on*, pp. 4641–4644. IEEE.

[XJC08] Xue X., Jeon J., and Croft W. B. (2008) Retrieval models for question and answer archives. In *Proceedings of the 31st annual international ACM SIGIR conference on Research and development in information retrieval*, pp. 475–482. Singapore, Singapore.

[XL10] Xu S. and Lau S. Y. F. C. M. (2010) Keyword extraction and headline generation using novel word features. In *Proceedings of the 24th AAAI Conference on Artificial Intelligence (AAAI'10)*, pp. 1461–1466. Atlanta, Georgia, USA.

[XMF09] Xexeo G., Morgado F., and Fiuza P. (2009) Differential tag clouds: Highlighting particular features in documents. In *Proceedings of the 2009 IEEE/WIC/ACM International Joint Conference on Web Intelligence and Intelligent Agent Technology - Volume 03*, pp. 129–132. Washington, DC, USA.

[XWS02] Xu D., Wang H., and Su K. (2002) Intelligent student profiling with fuzzy models. In *System Sciences, 2002. HICSS. Proceedings of the 35th Annual Hawaii International Conference on*, pp. 8–pp. IEEE.

[Yan09a] Yang C.-Y. (2009) A semantic faq system for online community learning. *Journal of Software* 4(2): 153–158.

[Yan09b] Yang S. (2009) Developing of an ontological interface agent with template-based linguistic processing technique for faq services. *Expert Systems with Applications* 36(2): 4049–4060.

[Yan10] Yang F.-J. (2010) The ideology of intelligent tutoring systems. *ACM Inroads* 1(4): 63–65.

[YCH07] Yang S., Chuang F., and Ho C. (2007) Ontology-supported faq processing and ranking techniques. *Journal of Intelligent Information Systems* 28: 233–251.

[YL13] Yu W.-B. and Luna R. (2013) Exploring user feedback of a e-learning system: a text mining approach. In *Human Interface and the Management of Information. Information and Interaction for Learning, Culture, Collaboration and Business,*, pp. 182–191. Springer.

[YLC05] Yu F.-Y., Liu Y.-H., and Chan T.-W. (2005) A web-based learning system for question-posing and peer assessment. *Innovations in Education and Teaching International* 42(4): 337–348.

[YRK$^+$13] Yildirim Z., Reigeluth C. M., Kwon S., Kageto Y., and Shao Z. (2013) A comparison of learning management systems in a school district: searching for the ideal personalized integrated educational system (pies). *Interactive Learning Environments* (ahead-of-print): 1–16.

[YZ14] Yusoff S. R. M. and Zin N. A. M. (2014) Design and evaluation of collaborative learning management system (clms) framework for teaching technical subject. In *New Horizons in Web Based Learning*, pp. 79–89. Springer.

[Zad65] Zadeh L. A. (1965) Fuzzy sets. *Information and control* 8(3): 338–353.

[ZBK96] Zimmerman B. J., Bonner S., and Kovach R. (1996) *Developing self-regulated learners: Beyond achievement to self-efficacy.* American Psychological Association.

[ZGH11] Zouaq A., Gasevic D., and Hatala M. (2011) Towards open ontology learning and filtering. *Information Systems* 36(7): 1064–1081.

[ZH05] Zhang X. and Han H. (2005) An empirical testing of user stereotypes of information retrieval systems. *Information processing & management* 41(3): 651–664.

[Zha04] Zhang D. (2004) Virtual mentor and the lab system-toward building an interactive, personalized, and intelligent e-learning environment. *Journal of Computer Information Systems* 44(3): 35–43.

[ZHXL06] Zhang K., H. Xu J. T., and Li J.-Z. (2006) Keyword extraction using support vector machine. In *Advances in Web-Age Information Management*, volumen 4016 of *Lecture Notes in Computer Science*, pp. 85–96. Springer Berlin / Heidelberg.

[Zim86] Zimmerman B. J. (1986) Development of self-regulated learning: Which are the key subprocesses. *Contemporary Educational Psychology* 16(3): 307–313.

[Zim89] Zimmerman B. J. (1989) A social cognitive view of self-regulated academic learning. *Journal of educational psychology* 81(3): 329.

[Zim00] Zimmerman B. J. (2000) Attaining self-regulation: a social cognitive perspective. *Handbook of self-regulation* pp. 13–39.

[ZKM12] Zubrinic K., Kalpic D., and Milicevic M. (2012) The automatic creation of concept maps from documents written using morphologically rich languages. *Expert Systems with Applications* 39(16): 12709–12718.

[ZN09a] Zenebe A. and Norcio A. F. (2009) Representation, similarity measures and aggregation methods using fuzzy sets for content-based recommender systems. *Fuzzy Sets and Systems* 160(1): 76–94.

[ZN09b] Zouaq A. and Nkambou R. (2009) Enhancing learning objects with an ontology-based memory. *Knowledge and Data Engineering, IEEE Transactions on* 21(6): 881–893.

[ZN09c] Zouaq A. and Nkambou R. (2009) Evaluating the generation of domain ontologies in the knowledge puzzle project. *Knowledge and Data Engineering, IEEE Transactions on* 21(11): 1559–1572.

[ZN10] Zouaq A. and Nkambou R. (2010) A survey of domain ontology engineering: methods and tools. In *Advances in intelligent tutoring systems*, pp. 103–119. Springer.

[ZS14] Zervas P. and Sampson D. G. (2014) The effect of users' tagging motivation on the enlargement of digital educational resources metadata. *Computers in Human Behavior* 32: 292–300.

[Zub09] Zubiaga A. (2009) Enhancing navigation on wikipedia with social tags. In *Wikimania 2009: 4th Annual Conference of the Wikimedia Community.*

[ZW08] Zhang C. and Wu D. (2008) Concept extraction and clustering for topic digital library construction. In *Proceedings of the 2008 IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology-Volume 03*, pp. 299–302. IEEE Computer Society.

[ZYY07] Zhou W., Yasuda T., and Yokoi S. (2007) E-namosupport: A web-based helpdesk support environment for senior citizens. In *Web Information Systems and Technologies*, pp. 295–306. Springer.

[ZZT09] Zhang Y., Zhang N., and Tang J. (2009) A collaborative filtering tag recommendation system based on graph. *ECML PKDD discovery challenge* pp. 297–306.

[ZZZNJ04] Zhang D., Zhao J. L., Zhou L., and Nunamaker Jr J. F. (2004) Can e-learning replace classroom learning? *Communications of the ACM* 47(5): 75–79.