

UNIVERSIDAD DE GRANADA

Escuela Técnica Superior de Ingeniería Informática
Departamento de Ciencias de la Computación
e Inteligencia Artificial



SISTEMAS DE ACCESO A LA INFORMACIÓN
BASADOS EN INFORMACIÓN LINGÜÍSTICA
DIFUSA Y TÉCNICAS DE FILTRADO

MEMORIA DE TESIS PRESENTADA POR

Carlos Porcel Gallego

COMO REQUISITO PARA
OPTAR AL GRADO DE DOCTOR
EN INFORMÁTICA

Granada

Diciembre de 2005

UNIVERSIDAD DE GRANADA

Escuela Técnica Superior de Ingeniería Informática

Departamento de Ciencias de la Computación
e Inteligencia Artificial



SISTEMAS DE ACCESO A LA INFORMACIÓN
BASADOS EN INFORMACIÓN LINGÜÍSTICA
DIFUSA Y TÉCNICAS DE FILTRADO

MEMORIA DE TESIS PRESENTADA POR

Carlos Porcel Gallego

PARA OPTAR AL GRADO DE DOCTOR EN INFORMÁTICA

DIRECTOR

Dr. Enrique Herrera Viedma

FDO. CARLOS PORCEL GALLEGO

FDO. ENRIQUE HERRERA VIEDMA

Granada

Diciembre de 2005

A la memoria de mi padre

Agradecimientos

Desde aquí, quería agradecer a Enrique Herrera, mi director de tesis, por todo su esfuerzo y dedicación durante todo este tiempo, y no sólo en la investigación, sino por todo en general. A Francisco Herrera y todos los miembros del grupo de investigación *Soft Computing y Sistemas de Información Inteligentes*, por la ayuda prestada.

A toda la gente de la Oficina de Transferencia de Resultados de Investigación (OTRI) de la Universidad de Granada, por todo el tiempo que hemos pasado trabajando juntos, que para mi ha significado mucho.

También a Eva, porque siempre ha estado de guardia cuando la he necesitado. A nuestro Chucky, por lo relajante que era su *ronroneo* cuando se venía junto a mí, mientras yo trabajaba. Y como no, a mi familia y en especial a mi madre, sobre todo por su gran paciencia, cariño y comprensión, y todas las horas que hemos pasado juntos...a ver si a partir de ahora empiezas a salir un poco más.

Porque sin vosotros no lo hubiera conseguido.

Índice general

1. Planteamiento, Objetivos y Estructura de la Memoria	3
1.1. Planteamiento	3
1.1.1. Técnicas de Acceso a la Información: Sistemas de Filtrado de Información	6
1.1.2. Uso del Modelado Lingüístico Difuso en el Acceso a la Información	9
1.1.3. Contextos de Acceso a la Información	11
1.2. Objetivos	13
1.3. Estructura de la Memoria	14
2. Sistemas de Filtrado de Información y Perfiles de Usuario	19
2.1. Introducción	19
2.2. Sistemas de Filtrado de Información	21
2.2.1. Definición	21
2.2.2. Uso de los Sistemas de Filtrado de Información	24
2.2.3. Estructura de los Sistemas de Filtrado de Información	26
2.2.4. Métodos de Generación de Recomendaciones: Basados en Contenidos y Colaborativos	31

2.2.5.	Aspectos a Considerar en el Diseño de Sistemas de Filtrado de Información	42
2.2.6.	Problemas Asociados a los Sistemas de Filtrado de Información	44
2.2.7.	Evaluación de los Sistemas de Filtrado de Información. Métricas	47
2.3.	Perfiles de Usuario	51
2.3.1.	Generación de Perfiles de Usuario	53
2.3.2.	Aprendizaje de Perfiles de Usuario	58
2.4.	Sistemas de Filtrado de Información Difusos	61
2.5.	Ejemplos de Sistemas de Filtrado de Información	65
3.	Modelado Lingüístico Difuso de la Información	73
3.1.	Introducción	73
3.2.	Conceptos Básicos de Información Lingüística	76
3.2.1.	Conjuntos Difusos y Funciones de Pertenencia	77
3.2.2.	Definiciones Básicas	79
3.2.3.	Operaciones con Conjuntos Difusos	80
3.2.4.	Modelado Lingüístico Difuso	81
3.2.5.	Pasos para la Aplicación del Enfoque Lingüístico Difuso	84
3.3.	Modelado Lingüístico Difuso Clásico	86
3.4.	Modelado Lingüístico Difuso Ordinal	87
3.4.1.	Modelo de Representación en el Enfoque Lingüístico Ordinal	88
3.4.2.	Modelo Computacional en el Enfoque Lingüístico Ordinal	89
3.5.	Modelado Lingüístico Difuso Basado en 2-tuplas	95
3.5.1.	Modelo de Representación Lingüística Basada en 2-tuplas	95

3.5.2. Modelo Computacional Lingüístico de las 2-tuplas	98
3.6. Modelado Lingüístico Difuso Multi-granular	100
3.7. Modelado Lingüístico Difuso no Balanceado	105
4. Un Sistema de Acceso a la Información en la Web Basado en Información Lingüística Multi-granular y en Técnicas de Filtrado	111
4.1. Introducción	111
4.2. Preliminares	116
4.2.1. Sistemas Multi-agente	116
4.2.2. Un Modelo Multi-agente Basado en Información Lingüística y Técnicas de Filtrado de Información	119
4.3. Un Sistema Multi-agente de Acceso a la Información en la Web Basado en Información Lingüística Multi-granular y en Técnicas de Filtrado	126
4.3.1. Arquitectura del Modelo	130
4.3.2. Funcionamiento del Modelo	133
4.3.3. Ejemplo de Funcionamiento	143
5. Un Sistema de Recomendaciones sobre Recursos de Investigación: SIRE2IN	157
5.1. Introducción	157
5.2. Arquitectura del Sistema	162
5.3. Estructuras de Datos	164
5.4. Actividad del Sistema	172
5.4.1. Proceso de Inserción de Usuarios	173
5.4.2. Proceso de Inserción de Recursos	176
5.4.3. Proceso de Filtrado	178

5.4.4. Proceso de Realimentación de los Usuarios	184
5.5. Desarrollo del Sistema	185
5.5.1. Implementación	185
5.5.2. Descripción de la Aplicación	187
5.6. Discusión	190
6. Comentarios Finales	195
6.1. Conclusiones	195
6.2. Trabajos Futuros	197
Bibliografía	199

Capítulo 1

Planteamiento, Objetivos y Estructura de la Memoria

1.1. Planteamiento

Desde hace unos años, la revolución digital está provocando un descomunal crecimiento de la cantidad de información que se crea y distribuye en formato electrónico. En cualquier ámbito en el que nos encontremos, podemos tener disponible un gran volumen de información de todo tipo, lo cual dificulta el acceso de una manera rápida y sencilla a la información que realmente nos interesa o necesitamos [66, 67]. Por ejemplo, los usuarios suscritos a una lista de distribución pierden gran parte del tiempo ojeando, leyendo o simplemente eliminando mensajes de correo irrelevantes para ellos. Este hecho provoca que el acceso a la información en cualquier ámbito sea una tarea compleja, por lo que los usuarios cada vez más necesitan herramientas automáticas que les ayuden a encontrar la información que mejor se adapte a sus requerimientos. Estar actualizado desde el punto de vista de la información disponible en cualquier ámbito, permite tanto a los usuarios individuales como a empresas ser más competitivos y adoptar mejores decisiones.

Como consecuencia, son numerosos los estudios realizados y sistemas propuestos para dar solución al problema, ya sea en la Web o en cualquier otro ámbito de aplicación [3, 4, 17, 19, 21, 62]. Todas estas investigaciones están basadas en diferentes técnicas o filosofías de trabajo, pero se pueden englobar bajo un mismo concepto, el de *Acceso a la Información* (en inglés, *Information Seeking* [72]), término que describe cualquier proceso que hace posible filtrar la gran cantidad de información disponible, y que el usuario únicamente acceda a información relevante para él. Sin embargo, el gran volumen de información al que nos enfrentamos actualmente limita el rendimiento de estos sistemas, por lo que se hace preciso la aplicación de técnicas de Inteligencia Artificial (IA) para ayudar a los usuarios en sus procesos de acceso a la información, ya que permiten mejorar los resultados obtenidos [103]. Por ello, el estudio de los procesos de acceso a la información, así como la aplicación de mejoras en ellos con el fin de obtener una mayor eficiencia, se muestra como una línea de investigación muy activa. En concreto, destacamos dos tipos de sistemas de acceso a la información [6]:

- Los sistemas de acceso a la información basados en los métodos tradicionales de ***Recuperación de Información (RI)*** que se encargan de dar respuesta a necesidades de información puntuales que puedan tener los usuarios. Estas necesidades quedan representadas como consultas que los usuarios introducen en el sistema y automáticamente obtienen una respuesta, de modo que los resultados que se van obteniendo dependen en gran medida de la habilidad que los usuarios tengan de expresar mediante consultas sus necesidades de información. Son los más extendidos y se conocen con el nombre de buscadores [4], centrados en obtener información relevante para los usua-
-

rios. Su actividad se desarrolla on-line, por lo que el sistema no dispone de ningún tipo de conocimiento a priori sobre los usuarios.

- Sistemas de acceso a la información basados en técnicas de ***Filtrado de Información (FI)***. El Filtrado de Información es un término usado para describir toda una variedad de procesos involucrados en la entrega de información exclusivamente a quiénes la necesitan. Por tanto, estos sistemas evalúan y filtran la gran cantidad de información disponible para los usuarios y así ayudarles en sus procesos de acceso a dicha información. En este caso, el sistema intenta dar respuesta a necesidades de los usuarios más persistentes en el tiempo, y en lugar de representar dichas necesidades mediante consultas puntuales, éstas son deducidas a partir de *Perfiles de Usuario*. Observamos que este tipo de sistemas sí tienen un conocimiento sobre los usuarios, almacenando mediante perfiles las preferencias o características de los mismos, por lo que en este caso la forma de trabajo es off-line. Los sistemas anteriores trabajan buscando información relevante, mientras que los sistemas de FI persiguen satisfacer las necesidades de los usuarios recomendando información personalizada, de ahí que se hayan popularizado bastante con el nombre de *Sistemas de Recomendaciones (SR)* [91].

En cualquier caso, ambos tienen el objetivo de ayudar al usuario a satisfacer sus necesidades de información. En este sentido, Belkin y Croft [6] determinaron que el FI y la RI constituyen las dos caras de una misma moneda que, trabajando en estrecha relación, consiguen ayudar a los usuarios en la obtención de la información que necesitan para lograr sus objetivos. De hecho, usando sistemas de FI, podemos depurar la información seleccionada por los sistemas de RI, de manera que la información mostrada finalmente a los usuarios se adapte lo mejor posible a sus

necesidades.

Por otro lado, nos enfrentamos al problema de disponer de una gran variedad de posibilidades a la hora de representar y evaluar la información [4, 62]. El problema se agrava aún más en los procesos en los que intervienen los usuarios, que muchas veces no son capaces de representar sus necesidades o preferencias de información de una forma adecuada, sino más bien de forma subjetiva, imprecisa o vaga [84, 116]. Se hace, pues, necesario el uso de técnicas para el manejo de información subjetiva, imprecisa y cualitativa como son las técnicas de *Modelado Lingüístico Difuso* para crear un entorno de trabajo flexible [7, 22, 55, 56, 116].

1.1.1. Técnicas de Acceso a la Información: Sistemas de Filtrado de Información

Los *Sistemas de FI* son herramientas que se usan para evaluar y filtrar la gran cantidad de información disponible y así asistir a los usuarios en sus procesos de búsqueda y acceso a la información, a partir de las preferencias y características conocidas sobre los mismos. El uso de estos sistemas está cada vez más extendido debido a su utilidad en el ámbito del comercio electrónico [93, 96, 97], para ayudar a los usuarios a obtener la información o encontrar el producto y/o servicio que están buscando de acuerdo con sus preferencias, necesidades o gustos, ocultando la información no útil existente en la Web. Por tanto, estos sistemas presentan una gran funcionalidad en cualquier ámbito de aplicación como pueden ser empresas, organizaciones de cualquier tipo, centros de I+D, etc., en los que se configuran como herramientas útiles en la distribución del conocimiento entre

todos sus integrantes.

Los sistemas de FI se caracterizan por los siguientes aspectos que los diferencian de los más tradicionales sistemas de RI y bases de datos [6, 34, 69, 91]:

- Un sistema de FI es aplicable en dominios en los que se trabaja con información no estructurada o semi-estructurada, como por ejemplo una página Web o un correo electrónico. Esto los diferencia de una aplicación típica de base de datos en la que la información está perfectamente estructurada en tablas y registros; además, los campos de los registros constan de tipos de datos simples con un significado perfectamente definido.
 - Están basados en perfiles de usuario. El proceso de filtrado está basado en descripciones de intereses o preferencias por parte de usuarios individuales o de grupos de usuarios, que permiten definir los perfiles de los usuarios. Estos perfiles representan intereses a largo plazo, es decir, no suelen responder a necesidades concretas y temporales.
 - Gestionan grandes cantidades de información.
 - Trabajan habitualmente con información en modo texto.
 - Suelen actuar sobre un flujo de información entrante procedente de fuentes remotas. Este escenario es apropiado para diseñadores de agentes inteligentes que asisten a los usuarios de Internet en la búsqueda de información apropiada según sus necesidades. En muchos casos, el filtrado implica eliminar información irrelevante del flujo de entrada, más que encontrar una determinada información en dicho flujo.
-

Tradicionalmente, estos sistemas de FI pueden ser de dos clases [34, 91]:

- *Sistemas basados en contenidos* que filtran y recomiendan ítems realizando un proceso de cálculo de similaridad (matching) entre las características que definen el perfil del usuario y las características usadas en la representación de los ítems, ignorando cualquier información de otros usuarios. En un contexto documental, estas características serían los términos introducidos por los usuarios en sus consultas y los términos índice usados en la representación de los documentos, y se asemeja a un sistema de RI.
- *Sistemas colaborativos* que para filtrar y recomendar ítems a un usuario dado, usan información tanto implícita como explícita sobre las preferencias de un grupo de usuarios, ignorando en el proceso la representación de los ítems. La información disponible sobre las preferencias o características de los usuarios, nos permite definir *perfiles de usuario*. La idea es que si el sistema recomienda a un usuario un determinado ítem, y dicho ítem satisface al usuario, entonces tendríamos que recomendarlo también a aquellos usuarios que tengan un perfil similar porque es probable que también les satisfaga. Este tipo de sistemas de FI no funcionan bien cuando disponemos de poca información sobre los usuarios, o cuando los disponibles tienen intereses diferentes. La construcción o definición de perfiles precisos es un aspecto clave y el funcionamiento correcto del sistema dependerá en gran medida de la disponibilidad de perfiles aprendidos para representar correctamente las preferencias de los usuarios [34].

La elección de un tipo de sistema u otro dependerá de las características del sistema que estemos diseñando, aunque en muchos casos resulta de gran utilidad

adoptar un *enfoque híbrido* en el que se aprovechan las ventajas de ambos y se reducen los inconvenientes que cada uno de ellos presenta por separado.

1.1.2. Uso del Modelado Lingüístico Difuso en el Acceso a la Información

Como hemos comentado, en los últimos años se ha experimentado un interés creciente en la aplicación de técnicas basadas en IA al campo del acceso a la información, con el propósito de resolver los problemas de rendimiento provocados por la expansión del volumen de información disponible. Las técnicas de Soft Computing son una de las técnicas de IA que más se están usando y con buenos resultados [18, 19, 76, 119]. El concepto de Soft Computing fue introducido por Zadeh [117] como una sinergia de metodologías (lógica difusa, computación evolutiva, redes neuronales, razonamiento probabilístico, etc.) que proporcionan los fundamentos para la concepción, diseño, construcción y utilización de sistemas de información inteligentes. El principio básico del Soft Computing [118] es la tolerancia a la imprecisión, incertidumbre y aproximación. El hecho de que la subjetividad y la incertidumbre sean propiedades típicas de cualquier proceso de acceso a la información, sobre todo si intervienen los usuarios, ha dado lugar a que las técnicas de Soft Computing se hayan revelado como una excelente herramienta para el manejo de la subjetividad y la imprecisión en la definición de los sistemas de acceso a la información. El uso de técnicas de Soft Computing puede aportar una mayor flexibilidad a estos sistemas [18, 19, 76]. Existe una gran cantidad de contribuciones que afrontan el uso de las técnicas de Soft Computing en el campo del acceso a la información. En particular, la Lógica Difusa [115, 117] está siendo utilizada

para modelar la subjetividad y la incertidumbre existentes en la actividad de la RI [17, 18, 19, 75].

Hay numerosas situaciones o áreas en las que la información no puede ser evaluada precisamente de forma cuantitativa, pero puede que sí sea factible y a la vez útil hacerlo de forma cualitativa. Así, cuando intentamos cuantificar algún fenómeno relacionado con percepciones humanas, a menudo usamos palabras o descripciones en lenguaje natural, en lugar de valores numéricos, como por ejemplo cuando evaluamos el confort o el diseño de un determinado coche, solemos usar términos como *bueno*, *medio* o *malo*. En otros casos, trabajar con información precisa de forma cuantitativa no es posible, o bien porque no está disponible o bien porque el coste computacional es demasiado alto y nos basta con la aplicación de un "valor aproximado". Por ejemplo, cuando evaluamos la velocidad de un coche, en lugar de usar valores numéricos, solemos usar términos tales como *rápido*, *muy rápido* o *lento*.

En este sentido, el uso de la Teoría de Conjuntos Difusos ha dado muy buenos resultados para el tratamiento de información de forma cualitativa [116]. El *modelo lingüístico difuso* es una herramienta basada en el concepto de *variable lingüística* [116] para tratar las valoraciones cualitativas. Los valores que se asignan a estas variables no son números, sino palabras o sentencias expresadas en lenguaje natural [116]. Cada valor lingüístico se caracteriza por un valor sintáctico o *etiqueta* y un valor semántico o *significado*. Se ha demostrado que es una herramienta muy útil en numerosos problemas, como por ejemplo en la toma de decisiones [55, 106, 109], evaluación de la calidad informativa de documentos Web [46], modelos de recuperación de información [11, 38, 39], diagnósticos clínicos

[22], análisis político [2], etc.

Para aplicar un modelado lingüístico difuso, podemos considerar diferentes enfoques para representar la información lingüística:

1. *Modelado lingüístico difuso clásico* [116].
2. *Modelado lingüístico difuso ordinal* [51, 55], definido para eliminar la excesiva complejidad del modelado lingüístico difuso clásico. Se usan conjuntos de etiquetas simétricos y uniformemente distribuidos.
3. *Modelado lingüístico difuso basado en las 2-tuplas* [56, 58], desarrollado para mejorar el rendimiento del enfoque lingüístico difuso ordinal.
4. *Modelado lingüístico difuso multi-granular* [52, 57], definido para afrontar situaciones en las que la información lingüística puede ser evaluada sobre diferentes conjuntos de etiquetas.
5. *Modelado lingüístico difuso no balanceado* [53, 54], desarrollado para tratar situaciones en las que la información lingüística tiene que ser evaluada sobre un conjunto de etiquetas no balanceado, es decir, un conjunto de etiquetas asimétrico y no uniforme.

1.1.3. Contextos de Acceso a la Información

El desarrollo de sistemas de acceso a la información se plantea en dos ámbitos de acción, uno más general que es el ámbito de la Web, y otro más particular, el

ámbito de una organización concreta:

- *Acceso a la información en la Web.* Consiste en asistir a los usuarios de Internet en sus procesos de búsqueda de información. Los sistemas más tradicionales son conocidos buscadores, en los que los usuarios introducen sus consultas sobre la información a la que desean acceder y el sistema muestra la información recuperada. Posteriormente aparecieron los SR y en los últimos años observamos la aparición de una nueva técnica basada en sistemas multi-agente. Un *sistema multi-agente* es aquel en el que cierto número de agentes individuales cooperan e interactúan entre sí en un entorno distribuido para conseguir un objetivo global. En la Web, este objetivo global consiste en asistir a los usuarios de Internet en sus procesos de búsqueda de información, mediante la participación de agentes inteligentes distribuidos encargados de encontrar la información que mejor se ajusta a las necesidades de información de los usuarios.

 - *Acceso a la información en el ámbito de una determinada organización.* Consiste en centrarnos en un ámbito concreto dentro de una determinada organización, en nuestro caso la OTRI de la Universidad de Granada. La OTRI pertenece al Vicerrectorado de Investigación y Tercer Ciclo de la Universidad de Granada, y su principal objetivo es fomentar y ayudar a la generación de conocimiento desde la universidad, así como a su difusión y transferencia a la sociedad, con el propósito de identificar las demandas y necesidades del entorno productivo. Para llevar a cabo su misión fundamental, la OTRI cuenta con técnicos en Transferencia de Tecnología, y una de las labores que deben realizar los técnicos es la difusión de convocatorias
-

sobre recursos de investigación. Ello implica la selección por parte de los técnicos, de los investigadores y empresas del entorno a los que más les podría interesar cada una de las convocatorias que vayan surgiendo, liberando al resto de usuarios de acceder a esa información que será irrelevante para ellos. Sin embargo, dada la gran cantidad de información y recursos a los que podemos acceder, cada vez se hace más necesaria la existencia de una herramienta automática de difusión personalizada, que facilite esta labor de los técnicos de OTRI. Por ello, proponemos una solución basada en herramientas de filtrado de información. Además, para aportar mayor flexibilidad en la representación y tratamiento de la información, aplicamos el modelado lingüístico difuso multi-granular.

1.2. Objetivos

En esta memoria nos proponemos profundizar en el diseño de sistemas de acceso a la información, realizando algunas propuestas de mejora basadas en hibridaciones de sistemas de acceso a la información basados en RI y FI, junto con la aplicación de técnicas de Inteligencia Artificial como el modelado lingüístico difuso.

En concreto, este objetivo se desglosa en los siguientes sub-objetivos:

- Estudiar y analizar el concepto y características de los sistemas de FI como herramientas útiles para ayudar a los usuarios en sus procesos de acceso a la información, destacando también la funcionalidad de la inclusión de estos
-

sistemas en distintos ámbitos de aplicación, como herramientas efectivas de distribución de conocimiento entre sus integrantes.

- Estudiar y analizar las distintas técnicas de manejo de información lingüística difusa, así como sus aplicaciones.
- Presentar un sistema para la búsqueda y acceso a la información en la Web, diseñado incorporando técnicas de filtrado de información y un tipo particular de modelado lingüístico difuso, el denominado modelado lingüístico difuso multi-granular, útil cuando tenemos distintos conjuntos de etiquetas para valorar la información.
- Diseño e implementación de un sistema de acceso a la información basado en herramientas de filtrado e información lingüística difusa multi-granular y dirigido a investigadores de la Universidad de Granada y empresas del entorno, para que puedan acceder a información diaria y personalizada sobre recursos de investigación (convocatorias, proyectos, eventos, congresos, noticias, etc.) que les puedan ser de interés.

1.3. Estructura de la Memoria

Esta memoria está compuesta por este capítulo de introducción, cuatro capítulos en los que se desarrolla la investigación realizada y un capítulo de comentarios finales en el que se incluyen las conclusiones obtenidas y los trabajos futuros.

En el Segundo Capítulo vamos a revisar las características y aspectos fundamen-

tales relacionados con el diseño e implementación de los Sistemas de FI. Estudiaremos los dos principales tipos de sistemas, los basados en contenidos y los colaborativos, así como la combinación de ambas técnicas para obtener sistemas híbridos. A continuación nos centraremos en los sistemas difusos que nos permitirán desarrollar mejores técnicas para el acceso a la información. Concluiremos viendo algunos ejemplos.

Dedicamos el Tercer Capítulo al estudio de los diferentes enfoques de Modelado Lingüístico Difuso para la representación de información, como son el modelado lingüístico difuso clásico, el modelado lingüístico difuso ordinal, el modelado lingüístico difuso basado en las 2-tuplas, el modelado lingüístico difuso multi-granular y el modelado lingüístico difuso no balanceado.

En el Capítulo Cuarto, presentamos un modelo de sistema multi-agente para el acceso a la información a través de la Web, basado en el uso de técnicas de filtrado mediante perfiles y en el modelado lingüístico difuso multi-granular.

En el Capítulo Quinto continuamos el estudio de los sistemas de acceso a la información, pero nos centramos en el ámbito de una organización determinada. Concretamente, realizamos el diseño e implementación de SIRE2IN, un Sistema de REcomendaciones sobre REcursos de INvestigación en el ámbito de la OTRI de la Universidad de Granada. Este sistema combina los resultados de los capítulos anteriores, por lo que es el resultado de una hibridación de las técnicas estudiadas aplicándolas al contexto de una organización específica, la OTRI.

Por último, finalizamos la memoria con el Capítulo 6 en el que incluimos las

conclusiones obtenidas, así como futuras líneas de trabajo.

Capítulo 2

Sistemas de Filtrado de Información y Perfiles de Usuario

Los Sistemas de Filtrado de Información son sistemas de acceso a la información, que buscan fundamentalmente predecir necesidades de información para recomendar ítems e información a partir de las preferencias y opiniones dadas por los usuarios. El uso de estos sistemas se está poniendo cada vez más de moda en Internet debido a que son muy útiles para evaluar y filtrar la gran cantidad de información disponible en la Web para asistir a los usuarios en sus procesos de búsqueda y acceso a la información. En este capítulo realizaremos una revisión de las características y aspectos fundamentales relacionados con el diseño e implementación de los sistemas de FI analizando distintas propuestas que han ido apareciendo en la literatura al respecto. Finalmente, nos centraremos en el análisis de aquellos sistemas que se basan en el uso de técnicas de Inteligencia Artificial, tales como la Teoría de Conjuntos Difusos.

2.1. Introducción

A menudo es necesario seleccionar una entre varias alternativas sin tener un

conocimiento exacto de cada una de ellas. En estas situaciones, la decisión final suele depender de las recomendaciones de otras personas [91], como ocurre cuando vamos a comprar algún producto y para elegir entre una marca u otra nos basamos en la recomendación de alguien que previamente lo haya adquirido o que tenga un conocimiento más preciso al respecto. En los procesos de acceso a la información, los sistemas de FI son herramientas cuyo objetivo es asistir a los usuarios en sus procesos de búsqueda de información, ayudando a filtrar los ítems de información recuperados, usando para ello recomendaciones pasadas dadas por otros usuarios sobre esos ítems. La recomendación sobre un ítem se genera a partir de las opiniones proporcionadas por otros usuarios sobre el ítem en búsquedas previas o bien a partir de las preferencias del usuario objeto de la recomendación (*usuario activo*) y la representación interna del ítem, dando lugar a los dos grandes grupos de sistemas de FI, los colaborativos y los basados en contenidos [34, 91].

Algunos ejemplos clásicos de sistema de FI en Internet son PHOAKS, Referral-Web, Fab, Siteseer, GroupLens [5, 91], o los más recientes MusicSurfer [82] y MusicStrands [81]. En todos ellos se manifiesta un claro problema para representar la subjetividad e imprecisión asociadas típicamente a las opiniones o recomendaciones de los usuarios. Una técnica de Inteligencia Artificial que ha dado muy buenos resultados para el manejo de información imprecisa, es la Teoría de Conjuntos Difusos [115].

La idea principal de este capítulo es presentar un estudio sobre los sistemas de FI para el acceso a la información, describiendo los aspectos más significativos de su diseño y problemas fundamentales con los que nos encontramos a la hora de construir un sistema de este tipo. Como hemos comentado, existen dos grandes

tipos de sistemas de FI, los colaborativos y los basados en contenidos, que no tienen porqué ser mutuamente exclusivos, y de hecho en muchas ocasiones aparecen juntos dando lugar a los sistemas híbridos.

El capítulo se estructura en cinco secciones. En la Sección 2 realizamos un estudio de los sistemas de FI en general, presentando su estructura, métodos de generación de recomendaciones, aspectos de diseño, problemas que nos podemos encontrar, así como algunos conceptos sobre la evaluación de este tipo de sistemas. Dedicamos la Sección 3 para estudiar con detalle todo lo relacionado con la generación y mantenimiento de los perfiles de usuario. En la Sección 4 analizamos algunos sistemas de FI Difusos, en los que, como ya hemos comentado, se usa la Teoría de Conjuntos Difusos para representar la imprecisión asociada a las recomendaciones en lenguaje natural. Por último, en la Sección 5 estudiamos una serie de ejemplos de sistemas de FI.

2.2. Sistemas de Filtrado de Información

En esta sección analizaremos los aspectos, características, estructura y problemas generales que se deben tener en cuenta a la hora de diseñar un sistema de FI.

2.2.1. Definición

El crecimiento de Internet en la última década ha puesto a disposición de los usuarios gran cantidad de información, servicios y productos a través de la red. Este hecho que a priori es positivo, implica también algunos problemas desde el punto de vista del usuario, tales como, la dificultad de gestionar la excesiva

cantidad de información a la que diariamente nos enfrentamos, complicando el acceso a la información que deseamos, o incapacidad para decidir de entre los ítems encontrados los que mejor se adecúan a nuestras necesidades, etc. Por lo tanto, se hace necesario contar con sistemas automatizados que soporten de forma eficiente y sencilla un acceso a la información relevante según las preferencias o características de los usuarios. El análisis de las características de la Web nos ayudará a comprender mejor el problema del acceso a la información en la Web y el motivo por el que ofrece numerosas oportunidades para su investigación:

- La Web es una fuente de información extremadamente dinámica porque la información que mantiene recibe constantes actualizaciones y además está creciendo continuamente, aunque de forma descontrolada y desorganizada.
- La Web contiene información en múltiples formatos, tales como texto, tablas estructuradas, información multimedia (música, imágenes y películas), etc.
- La Web presenta mucha información redundante y que no aporta nada, y sólo una pequeña parte de la Web contiene información realmente útil y relevante.

Estas son las razones por las que en la Web es muy difícil acceder a información de calidad, relevante según necesidades de información específicas. Los usuarios tienen distintas formas de acceder a la gran cantidad de información disponible a través de la Web, relacionadas con el propósito de su búsqueda. La forma de búsqueda más inmediata es directamente navegar a través de los sitios Web usando los enlaces que se van encontrando en las páginas visitadas; en este caso, no se

necesita ninguna expresión formal acerca de las necesidades de información. Sin embargo, cuando se busca alguna información específica, el paradigma anterior no es práctico y la eficiencia de los resultados que obtengamos depende fuertemente de la página de inicio. Otra forma de búsqueda consiste en usar motores de búsqueda basados en palabras clave, tales como *Google* [33] o *Yahoo* [114]. Consecuentemente, el rendimiento de estos sistemas sufre diversos problemas tales como:

- Problema del Web crawling: los motores de búsqueda pueden cubrir sólo una pequeña porción de la totalidad de la Web, debido a su tamaño, estructura y rápido crecimiento [66, 67].
- Problema del spamming: los algoritmos para la ordenación de resultados usados por los motores de búsqueda, pueden ser fácilmente manipulados para promover ciertas páginas subiéndolas a la parte más alta del conjunto de resultados.
- Problema de sobrecarga de información: usualmente, cuando un usuario busca información a través de la Web, recibe cientos de miles de documentos.

Filtrar la gran cantidad de información disponible permite mejorar el acceso a la información. Los sistemas de FI se han ido consolidando como potentes herramientas para ayudar a los usuarios a reducir la sobrecarga de información a la que nos enfrentamos en los procesos de acceso a la información. Ayudan a filtrar los ítems de información recuperados, usando distintas técnicas para identificar aquellos ítems que mejor casan con las preferencias o necesidades de los usuarios.

Un sistema de FI está asociado con un conjunto de ítems $I = \{i_1, \dots, i_n\}$ y su objetivo es recomendar a los usuarios ítems de I que les puedan ser de interés [113]. Por ejemplo, se podría diseñar un SR para la recomendación de películas; de hecho, a lo largo de la literatura hemos encontrado numerosos ejemplos de SR de películas, tales como Film-Conseil [29, 86], MovieFinder [80, 96], Reel.com [90, 96] o MetaLens [95, 98] entre otros. En un contexto documental los ítems serían los documentos almacenados en las distintas fuentes.

La implementación de técnicas para el desarrollo de los sistemas de FI está íntimamente relacionada con el tipo de información que se vaya a utilizar. Una primera fuente de información a tener en cuenta es el tipo de ítems con los que vamos a trabajar. Habrá situaciones en las que únicamente conozcamos un identificador de cada ítem. Por ejemplo, en el caso de la recomendación de películas solemos conocer únicamente el título. En otras situaciones, dispondremos de más información sobre los ítems, a través de una serie de atributos. En el caso de la recomendación de películas, podrían ser el año en que se hizo la película, el género, el director, protagonistas, etc. En el caso de recuperación de documentos, la información con la que contamos serían los índices usados en su representación. En general, cuanto más sofisticada es la representación de los ítems mejor se puede desarrollar la actividad de los sistemas de FI.

2.2.2. Uso de los Sistemas de Filtrado de Información

Uno de los usos más extendidos de los sistemas de FI es en los portales Web de co-

mercado electrónico donde se suelen usar estas herramientas para sugerir productos y proporcionar a los clientes servicios de valor añadido para de esta forma ayudarles en sus decisiones de compra. Las recomendaciones podrían estar basadas en los ítems más vendidos o preferidos, en decisiones demográficas o en el análisis del comportamiento anterior de los clientes, de cara a predecir comportamientos futuros [78]. La forma de las recomendaciones varía según los casos, pudiendo tratarse de sugerencias de productos, de información personalizada de productos, o bien de resúmenes o críticas del resto de usuarios del sistema. En cierto sentido esta variedad en las técnicas de recomendación, es parte de la personalización del sistema, permitiendo su adaptación a cada uno de los usuarios.

Los sistemas de FI son herramientas muy útiles para cualquier empresa ya que permiten ayudarles a tomar decisiones sobre a quién realizar una oferta o a quién dirigir las promociones publicitarias. Podrían sugerir a motores de búsqueda y compañías publicitarias qué anuncios u ofertas visualizar en función del comportamiento del cliente, ofreciendo de este modo un alto grado de personalización [34, 35].

Teniendo en cuenta esos aspectos, los sistemas de FI ayudan a mejorar las ventas de los portales Web de comercio electrónico de tres formas distintas [96, 97]:

- Convirtiendo los navegadores en tiendas de venta: a menudo los visitantes de un sitio Web se dedican a ojearlo sin llegar a comprar nada. Los sistemas de FI ayudan a los clientes a encontrar ítems interesantes que en muchos casos podrían comprar.
-

- Incrementando las ventas cruzadas: sugiriendo productos adicionales para que los clientes compren, por ejemplo recomendándole productos en función de los productos que vayan teniendo en la cesta; es algo parecido a lo que ocurre cuando vamos a un gran supermercado, que llaman nuestra atención con grandes ofertas, para luego ofrecer otro tipo de productos adicionales que en la mayoría de los casos ya no están de oferta y no teníamos pensado comprar. Si las recomendaciones son buenas, las ventas aumentarán considerablemente.
- Creando fidelidad: en este ámbito del comercio electrónico donde un cliente se puede ir a la competencia con sólo un movimiento de ratón, ganarse la fidelidad de los clientes es una gran estrategia de negocio. Los sistemas de FI mejoran la fidelidad creando relaciones de valor añadido entre los portales Web y los clientes, y eso lo consiguen investigando el comportamiento de dichos clientes y personalizando el portal según sus necesidades.

2.2.3. Estructura de los Sistemas de Filtrado de Información

A continuación presentamos los elementos fundamentales que intervienen en el esquema de funcionamiento de un sistema de FI. Dichos elementos los podemos usar como criterios de clasificación y son los siguientes [34, 35, 95]:

- *las entradas / salidas del proceso de generación de recomendaciones,*
 - *el método usado para generar las recomendaciones, y*
-

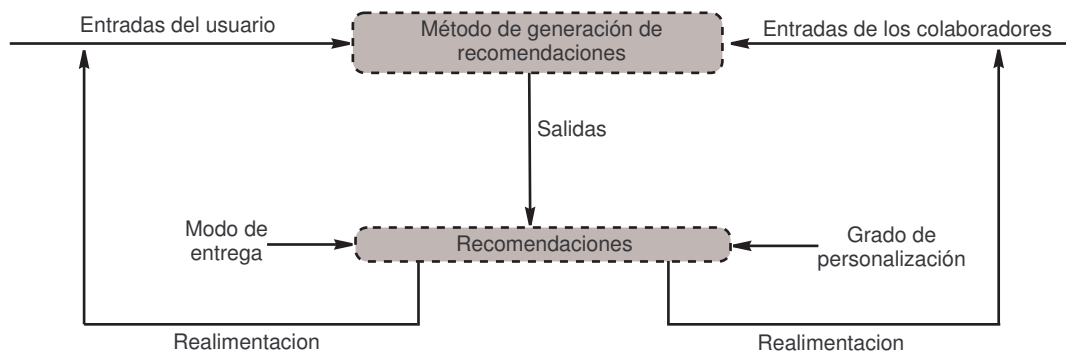


Figura 2.1: Esquema del proceso de recomendación.

- *el grado de personalización.*

Los sistemas de FI usan las entradas del usuario en cuestión, pero también información sobre los ítems o información del resto de usuarios del sistema, que actúan como colaboradores. La realimentación por parte de los usuarios (feedback) es muy importante de cara a albergar una información más completa ante futuros procesos de recomendación. La figura 2.1 refleja el proceso de recomendación.

2.2.3.1. Entradas / Salidas

Para poder realizar una recomendación a un usuario, es necesario conocer algún tipo de información sobre sus preferencias. Además, dependiendo del tipo de sistema también necesitaremos información sobre los ítems a recomendar o bien información reunida sobre el resto de usuarios del sistema (comunidad de usuarios o colaboradores). Esta información que necesitamos para realizar las recomen-

ciones constituye la entrada o entradas del sistema. Hay dos tipos de entradas:

- ***Entradas del usuario activo.*** Un sistema en el que no se tiene en cuenta información sobre el usuario objeto de las recomendaciones, es decir, que no tiene entradas correspondientes al usuario activo, no producirá recomendaciones personalizadas. Añadiendo algún tipo de información sobre el usuario, se permite personalizar las recomendaciones basándose en el comportamiento del usuario, en sus preferencias a largo plazo o en ambos. Esta información puede venir dada de dos formas que no tienen porqué ser mutuamente exclusivas [113]:
 - **Por extensión:** se refiere a información que se tenga sobre las experiencias pasadas del usuario con respecto a los ítems encontrados o siguiendo el rastro del comportamiento del usuario, qué ítems le interesan, en cuáles se entretiene más o cuáles ha añadido a su cesta de la compra. Es lo que también conocemos como *navegación implícita* pues el usuario no es consciente de estos seguimientos. Por ejemplo, en *Amazon* [1] se recomiendan una serie de libros que se consideran similares al libro que el usuario activo esté visualizando en cada momento.
 - **Por información expresada intencionalmente** se entiende alguna especificación de los ítems deseados por los usuarios. También se le llama *navegación explícita* y consiste en que el usuario alimenta al sistema de forma intencionada con información sobre sus preferencias. De esta forma, el usuario recibirá recomendaciones según haya complementado dicha información sobre sus preferencias. Generalmente, en este caso se suelen usar especificaciones similares a las usadas para representar
-

los ítems.

- ***Entradas de los colaboradores.*** Se trata de entradas al sistema que reflejan las opiniones generales de la comunidad de usuarios. Incluyen sentencias sobre los atributos de los ítems que nos permiten clasificar dichos ítems en distintas categorías. La idea es que en muchos casos, atributos tales como el género de una película refleja el consenso de un gran número de usuarios. En este caso también se podrían tener en cuenta las experiencias pasadas de los colaboradores con respecto a un ítem concreto y en función de ello, realizar o no una recomendación. Otros sistemas fomentan que los usuarios introduzcan comentarios de texto a modo de realimentación. De esta forma, cuando un usuario requiere una recomendación, el sistema reúne los comentarios sobre el ítem en cuestión y los presenta como ayuda al usuario a tomar su decisión. El inconveniente es que requiere procesamiento por parte del usuario que tendrá que leer los comentarios proporcionados.

Por otro lado, las ***salidas del sistema*** son las recomendaciones que se suministran a los usuarios. Dichas recomendaciones varían dependiendo del tipo, cantidad y formato de la información proporcionada al usuario. Algunas de las formas más comunes de representar la salida de un sistema de FI son las siguientes:

- Mediante una sugerencia al usuario de que pruebe o estudie el ítem que se le recomienda. En este sentido, habitualmente se adopta la idea de una lista de sugerencias. Algunos diseñadores prefieren mantener la lista de forma desordenada para no dar la impresión de que una recomendación particular sea mejor que otras. Otros sistemas, en cambio, ordenan los ítems recomendados
-

según un ranking, para de esta forma, aportar información extra sobre las recomendaciones generadas.

- Presentar a los usuarios una predicción del grado de satisfacción que se asignará al ítem concreto. Estas predicciones o estimaciones pueden ser presentadas como personalizadas al usuario o como estimaciones generales del conjunto de colaboradores.
- Cuando la comunidad de usuarios es pequeña o se conocen bien los miembros de dicha comunidad, podría ser útil visualizar las valoraciones individuales de los miembros que permitiría al usuario objeto de la recomendación obtener sus propias conclusiones sobre la efectividad de una recomendación.

Independientemente de estos formatos de salida, puede resultar muy interesante incluir una breve **descripción o explicación** sobre el ítem recomendado a modo de justificación del porqué de dicha recomendación.

2.2.3.2. Método de generación de recomendaciones

Dada la importancia que el método de generación de recomendaciones que se utilice, tiene de cara a clasificar un sistema de FI, hemos considerado oportuno dedicarles un apartado específico que veremos más adelante (sección 2.2.4).

2.2.3.3. Grado de personalización

El grado de personalización de un sistema de FI abarca aspectos tales como la precisión y la utilidad de las recomendaciones [35, 95]. La **precisión** mide lo

correcto que es el sistema, en el sentido de la fiabilidad de las recomendaciones generadas, mientras que la **utilidad** incluye factores como si el sistema proporciona recomendaciones válidas aunque no esperadas, o si el sistema proporciona recomendaciones distintas a usuarios distintos (*individualización*). En función de su grado de personalización, podemos clasificar los sistemas de FI en tres grupos:

1. Cuando los sistemas proporcionan las mismas recomendaciones a todos los usuarios, son clasificados como **no personalizados**. Dichas recomendaciones estarán basadas en selecciones manuales, resúmenes estadísticos u otras técnicas similares.
2. Los sistemas de FI que tienen en cuenta la información actual del usuario objeto de las recomendaciones, proporcionan **personalización efímera**, puesto que las recomendaciones son respuesta al comportamiento y acciones del usuario en su sesión actual de navegación.
3. Los sistemas de FI que ofrecen el mayor grado de personalización son los que usan **personalización persistente** ofreciendo recomendaciones distintas para distintos usuarios, incluso cuando estén buscando el mismo ítem. Estos sistemas están basados en el perfil de los usuarios, por lo que hacen uso de métodos de filtrado colaborativo, filtrado basado en contenidos o correlaciones entre ítems.

2.2.4. Métodos de Generación de Recomendaciones: Basados en Contenidos y Colaborativos

En esta sección describimos una serie de métodos de generación de recomendaciones que se usan habitualmente en los sistemas de FI, pero debemos tener en cuenta que no son mutuamente exclusivos entre sí, sino complementarios, es decir, que en un mismo sistema podríamos usar uno o varios de estos métodos, adoptando así sistemas híbridos.

En primer lugar vamos a enunciar los tres métodos más simples:

- **Recuperación pura** o **recomendación nula**, en la que el sistema ofrece a los usuarios una interfaz de búsqueda a través de la cual pueden realizar consultas a una base de datos de ítems. Se trata, pues, de un sistema de búsqueda por lo que técnicamente no es un método de generación de recomendaciones, aunque ante los usuarios aparece como tal.
 - Otros sistemas usan **recomendaciones seleccionadas manualmente** por expertos, como por ejemplo editores, artistas o críticos en el caso de recomendaciones de películas o cd's de música. Los expertos identifican ítems basándose en sus propias preferencias, intereses u objetivos, y crean una lista de ítems que esté disponible para todos los usuarios del sistema. A menudo acompañan estas recomendaciones de comentarios de texto que puedan ayudar a los usuarios a evaluar y entender las recomendaciones.
 - En otros casos, los sistemas ofrecen **resúmenes estadísticos** calculados en función de las opiniones del conjunto de usuarios, por lo que tampoco son personalizados. Por ejemplo, se podrían tener en cuenta el porcentaje de usuarios a los que ha satisfecho o han comprado un artículo, número de usuarios que recomiendan un ítem, o una evaluación media de todos los
-

usuarios con respecto al ítem.

Los tres métodos descritos se usan, pero por su simplicidad no son exactamente considerados como métodos de generación de recomendaciones. Los métodos de generación de recomendaciones que son considerados como tales son aquellos basados en el concepto de personalización, pues usan los perfiles de los usuarios para realizar las recomendaciones. Nos permiten clasificar los sistemas de FI en tres grandes grupos: los sistemas de FI colaborativos, los sistemas de FI no colaborativos o basados en contenidos y los sistemas híbridos.

1. *Sistemas de FI Basados en Contenidos.*

Los sistemas de FI basados en contenidos [34, 95] realizan las recomendaciones basándose en los atributos de los ítems y en las preferencias de los usuarios con respecto a dichos atributos. Las preferencias e intereses de los usuarios son proporcionadas o bien por los propios usuarios, por ejemplo a través de una consulta, o bien adquiridas por la observación de los ítems en los que el usuario se interesa. Los motores de búsqueda de texto son un primer ejemplo de filtrado basado en contenidos, que usan una técnica llamada *indexación*, que como sabemos es un concepto de RI consistente en calcular la frecuencia en que van apareciendo los términos en el texto. Para ello, los documentos y las necesidades de información del usuario se representan mediante vectores de una dimensión con una entrada para cada palabra que aparece en el texto. Cada componente del vector es la frecuencia con que la correspondiente palabra aparece en el documento o en la consulta de

usuario. Los vectores de documentos que son considerados más parecidos a los vectores de consulta, son los correspondientes a los documentos que más pueden interesar al usuario. Otros ejemplos de filtrado basado en contenidos son los *índices de búsqueda Booleanos* donde la consulta es un conjunto de palabras clave combinadas a través de operadores booleanos, *sistemas de recuperación probabilísticos* donde se usa el razonamiento probabilístico para determinar la probabilidad de que un documento satisfaga las necesidades del usuario, o *interfaces de consulta en lenguaje natural*, donde las consultas se introducen en lenguaje natural.

Por otro lado, están los sistemas que usan **correlaciones entre ítems** para identificar ítems asociados frecuentemente a un ítem por el que el usuario ha mostrado interés y por tanto recomendarle dichos ítems al usuario. Como ejemplo de esta idea, supongamos que tenemos un sistema de recomendaciones sobre libros y tenemos dos libros *La clave está en Rebeca* y *La isla de las tormentas*, ambos del mismo autor Ken Follett, pero que además ambos son de intriga y están ambientados en la 2ª Guerra Mundial; por tanto, se podrían considerar en cierto sentido similares, es decir, existe una alta correlación entre ambos. Por esa razón, si un usuario de nuestro sistema está interesado en el libro *La clave está en Rebeca*, podemos recomendarle también la lectura de *La isla de las tormentas*, que tiene una alta correlación con el primero.

En un enfoque basado en contenidos, se dispone de una representación \mathcal{R}_i para cada ítem que el usuario activo haya experimentado, así como de las

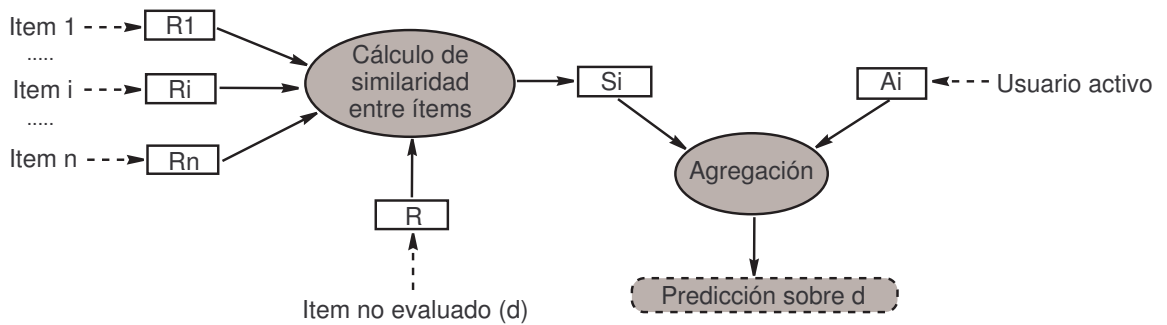


Figura 2.2: Esquema de filtrado basado en contenidos.

evaluaciones a_i de los ítems ya experimentados por el usuario, contenidas en \mathcal{A} . Además, para un ítem d aún no experimentado, que por tanto se está intentando evaluar, únicamente contamos con una representación \mathcal{R} . En este caso, el procedimiento de obtención del grado de recomendación se puede resumir en los dos pasos siguientes (ver figura 2.2):

- *Paso 1.* Combinar las representaciones \mathcal{R} y \mathcal{R}_i para obtener \mathcal{S}_i , que mide el grado de similitud del ítem d con respecto a los ítems ya experimentados.
- *Paso 2.* La predicción de la evaluación del ítem d se calcula realizando una agregación pesada de las tuplas de los vectores \mathcal{S}_i y a_i .

2. *Sistemas de FI Colaborativos.*

Por otro lado, la mayoría de sistemas de FI existentes son colaborativos [5, 31, 32, 35, 36, 77, 85]. Se dice que un sistema de FI es colaborativo si usa la información conocida sobre las preferencias de otros usuarios para realizar la recomendación al usuario que la precise. Estos sistemas colaborativos identifican usuarios cuyas preferencias sean similares a las de otros usuarios

datos y recomiendan a los primeros los elementos que hayan satisfecho a los otros. Por ello, la definición de medidas de similitud entre preferencias es un aspecto clave en estos sistemas.

En la vida cotidiana, a menudo es necesario seleccionar una entre varias alternativas posibles sin tener un conocimiento exacto de cada una de ellas. En estas situaciones, la decisión final puede depender de las recomendaciones de otras personas o amigos que tengan unas preferencias similares a las nuestras. De esta forma, si dos usuarios U_1 y U_2 , comparten el mismo sistema de valores (tienen las mismas preferencias) y al usuario U_1 le ha satisfecho un ítem i , probablemente este ítem también satisfaga al usuario U_2 por lo que deberíamos recomendárselo. Los algoritmos de filtrado colaborativos lo que hacen es automatizar esta idea de recomendación, para que pueda ser procesada por un ordenador y así ser generada automáticamente por el sistema.

A diferencia de los más tradicionales sistemas de filtrado basados en contenidos, desarrollados a partir de técnicas de RI o de inteligencia artificial, las decisiones de filtrado en los sistemas de FI colaborativos están basadas en análisis humanos y no en análisis automatizados. Los usuarios de los sistemas colaborativos realizan valoraciones sobre los ítems que han experimentado y en función de esas valoraciones se establecen sus perfiles de intereses. Con esos perfiles ya definidos, el sistema agrupa a cada usuario con otros usuarios de preferencias o intereses similares, de forma que para generar recomendaciones a un usuario se usan las valoraciones de esos usuarios afines con su

perfil.

El problema es, pues, predecir cómo valoraría un usuario un ítem que aún no haya evaluado a partir de preferencias y opiniones de un grupo de usuarios (comunidad de usuarios). Las preferencias pueden ser sentencias explícitamente expresadas por el usuario o evaluaciones implícitas calculadas a partir de datos disponibles sobre el comportamiento del usuario [83]. Las evaluaciones explícitas suelen tratarse de puntuaciones asignadas por los usuarios a los ítems que ya conozcan, donde puntuaciones altas reflejan un fuerte interés del usuario en ese ítem en concreto, mientras que las puntuaciones bajas, reflejan desinterés. Las evaluaciones implícitas suelen derivar de fuentes de datos tales como registros de compras o Web logs. Esto puede ser representado como una matriz de usuarios e ítems, donde cada celda representa la valoración de un usuario con respecto a un ítem concreto. Así visto, el problema consiste en predecir valores para las celdas que estén vacías. En el filtrado colaborativo, la matriz es por regla general muy dispersa, puesto que cada usuario únicamente habrá valorado un pequeño porcentaje del total de ítems. En la tabla 2.1 podemos ver un ejemplo de matriz que representa valoraciones de usuarios con respecto a una serie de películas.

	Gladiator	Leyendas de pasión	Spiderman	Braveheart
Daniel	5	2	4	5
María	2	5		3
Luis	3	2	4	2
Susana	?	1	4	5

Tabla 2.1: Ejemplo de matriz de valoraciones.

Los algoritmos que se suelen usar para implementar las técnicas de filtrado colaborativo se llaman **métodos basados en vecindad** [37]. Estos métodos consisten en que se selecciona un conjunto apropiado de usuarios, según la similitud de los mismos con respecto al usuario activo, y se usan las valoraciones de dichos usuarios para generar las valoraciones que se hagan al usuario activo. Como ejemplo, consideremos de nuevo la tabla 2.1. Podemos predecir que a Susana le gustará la película *Gladiator*. Observamos que Daniel es el vecino más cercano a Susana, puesto que ambos tienen unas valoraciones muy similares de las películas que ya han visto. Por tanto, la valoración de Daniel sobre la película *Gladiator* tendrá gran influencia en la predicción que hagamos a Susana sobre dicha película. Por el contrario, María y Luis tienen opiniones más dispares con respecto a Susana, por lo que tendrán una influencia mucho menor en las recomendaciones que se hagan a dicho usuario.

Los métodos basados en la vecindad utilizados para generar predicciones, funcionan en tres pasos:

- *Paso 1.* Medir la similitud de todos los usuarios con respecto al usuario activo.
 - *Paso 2.* Seleccionar un subconjunto de usuarios cuyas valoraciones se van a usar y por tanto, tendrán influencia en la generación de la predicción para el usuario activo.
 - *Paso 3.* Normalizar las puntuaciones de los distintos usuarios y calcular una predicción a partir de algún tipo de combinación ponderada de las
-

puntuaciones asignadas al ítem por los usuarios seleccionados en el paso anterior.

Si estudiamos a alto nivel este enfoque colaborativo y el anterior basado en contenidos, podemos observar que existe cierto grado de simetría entre ambos. En efecto, en ambos casos se hace uso de un vector de evaluaciones de los ítems ya experimentados por parte del usuario activo, que denotamos como \mathcal{A} . En el enfoque de *filtrado colaborativo*, se cuenta además con un vector \mathcal{A}_j para cada uno de los colaboradores indicando sus evaluaciones de los ítems correspondientes. Para cualquier ítem d que el usuario activo aún no haya experimentado, se dispone de un vector \mathcal{R} cuyos componentes, r_j son las evaluaciones de dicho ítem por parte de los colaboradores. El proceso de obtención del grado de recomendación, básicamente se puede dividir en los dos pasos siguientes (ver figura 2.3):

- *Paso 1.* Combinar \mathcal{A} y \mathcal{A}_j para obtener \mathcal{S}_j , que mide el grado de similitud entre nuestro usuario y cada uno de los colaboradores.
- *Paso 2.* La predicción de la evaluación del ítem d se calcula mediante una agregación ponderada de las tuplas de los vectores \mathcal{S}_j y r_j .

Ventajas de los sistemas de FI Colaborativos. El filtrado colaborativo presenta tres importantes ventajas con respecto al filtrado basado en contenidos:

- a) *Soporte para el filtrado de ítems cuyo contenido no es fácil de analizar por procesos automatizados.* En el filtrado colaborativo los usuarios
-

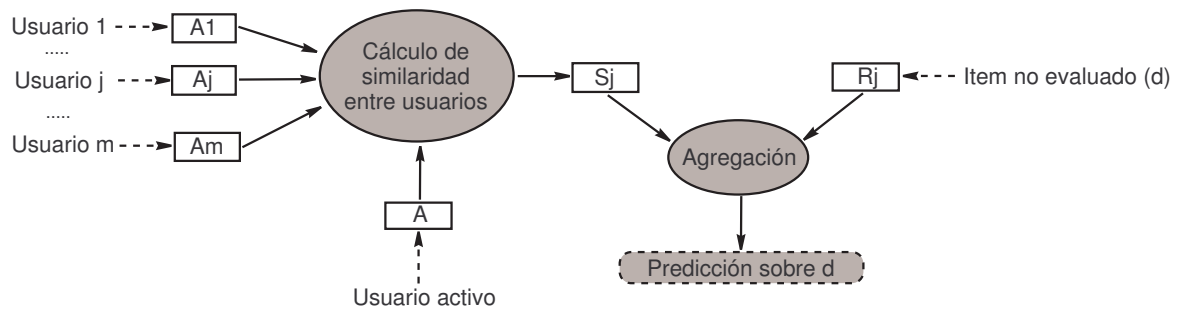


Figura 2.3: Esquema de filtrado colaborativo.

determinan la relevancia, calidad e intereses de un ítem, por lo que el filtrado se puede realizar sobre ítems que son difíciles de analizar por un ordenador, tales como películas, libros, documentos, ideas o sentimientos.

- b) *Posibilidad de filtrar ítems basándose en su calidad o preferencias.* Estos sistemas permiten el filtrado basándose en indicativos que van más allá que el contenido del ítem, como son las necesidades o intereses de los usuarios. Los humanos somos capaces de analizar aspectos como necesidades o intereses, muy difíciles de automatizar para que sean realizados por un ordenador.
- c) *Posibilidad de realizar recomendaciones no esperadas.* Estos sistemas son capaces de recomendar ítems de interés para el usuario, pero que no contienen un contenido esperado para él, es decir, recomendaciones válidas, pero que no esperábamos, lo cual puede resultar de gran utilidad.

3. *Sistemas de FI híbridos.*

Es importante observar que los enfoques de filtrado basados en contenidos y colaborativos no son mutuamente exclusivos, sino que pueden ser integrados en un mismo sistema para proporcionar sistemas híbridos más potentes.

En efecto, los sistemas de FI colaborativos son herramientas muy potentes para realizar el filtrado de información. Sin embargo, para que sean completamente potentes es necesario combinarlos con técnicas de filtrado basadas en contenidos. Los sistemas colaborativos realizan buenas predicciones de ítems que casan con las preferencias e intereses de los usuarios, pero no trabajan tan bien a la hora de filtrar información para necesidades de contenido específicas.

Otro aspecto a tener en cuenta es que los sistemas colaborativos, como ya hemos visto, generan recomendaciones a partir de opiniones y preferencias de otros usuarios por lo que para un buen funcionamiento del sistema, se hace necesario contar con un cierto número de usuarios. Cuando un sistema ya se ha diseñado e implantado y empieza a funcionar, normalmente el número de usuarios con los que se cuenta es muy bajo, por lo que en este caso, se podría empezar trabajando con un filtrado basado en contenidos y cuando se llegue a un número de usuarios ya aceptable, pasar al filtrado colaborativo.

Es por esto que en muchas ocasiones la mejor opción es adoptar un enfoque híbrido entre colaborativo y basado en contenidos y de esta forma disfrutar de las ventajas de ambos. En [14] se propone un sistema híbrido para un periódico on-line.

2.2.5. Aspectos a Considerar en el Diseño de Sistemas de Filtrado de Información

Algunos aspectos que debemos considerar sobre las recomendaciones en el diseño de sistemas de FI son [89, 91]:

- **Representación de las recomendaciones.** Los contenidos de una evaluación o recomendación pueden venir dados por un único bit (recomendado o no) o por comentarios de texto sin estructurar.
 - **Expresión de las recomendaciones.** Las recomendaciones pueden ser introducidas de forma explícita o bien de forma implícita.
 - **Aspectos de identificación de la fuente.** Las recomendaciones pueden realizarse de forma anónima, identificando la fuente, o bien usando un pseudónimo.
 - **Forma de agregar las evaluaciones.** Se refiere a cómo vamos a ir agregando las evaluaciones disponibles sobre los ítems de cara a generar las recomendaciones. Algunos sistemas van realizando una suma ponderada de la importancia (evaluación) concedida por parte de los usuarios con respecto a los ítems.
 - **Presentación o contenido de las recomendaciones.** Las recomendaciones se pueden mostrar de distintas formas. Por ejemplo, se podrían mostrar los ítems en forma de lista ordenada según las recomendaciones de cada uno, o a la hora de visualizar los ítems que se muestre también su recomendación.
-

A continuación describimos algunas sugerencias que nos pueden ayudar a la hora de plantearnos el diseño de un sistema de FI [77, 89, 91]:

- Los diseñadores de sistemas de FI a menudo se encuentran con el problema de elegir entre facilidad de uso (recogiendo poca información sobre las preferencias de los usuarios con respecto a los ítems) o precisión de los algoritmos de filtrado (que requiere una mayor participación por parte de los usuarios). Podemos sugerir que merece la pena recoger más información sobre los usuarios aunque se pierda en facilidad de uso, si ello va a significar un aumento considerable de la precisión.
 - La satisfacción de los usuarios, así como su disposición a valorar los ítems, serán más altas en aquellos sistemas que incluyen algún tipo de información en la página de valoración. Por ejemplo, en un sistema de recomendación de películas, sería útil incluir la portada, pero mejor si se añade algo más de información como puede ser la sinopsis.
 - Definir la escala de las valoraciones es algo complicado, por lo que se aconseja en la medida de lo posible estudiarlo conjuntamente con los futuros usuarios del sistema.
 - Algo realmente atractivo de un sistema de filtrado es que proporcione recomendaciones de ítems que no se esperaban, por ejemplo ítems nuevos o ítems poco conocidos, pero en muchos casos igualmente válidos.
 - Otro aspecto a tener en cuenta es detallar claramente la información característica de los ítems a la hora de recomendarlos, y considerar incluso
-

la posibilidad de incluir opiniones que sobre los ítems tengan el resto de usuarios.

- Sin embargo, la mejor receta para que un sistema de FI sea efectivo es utilizar diferentes estrategias para diferentes personas, es decir, adaptar las técnicas y métodos de diseño según el tipo de usuarios (y de ítems) con los que vaya a trabajar el sistema.

2.2.6. Problemas Asociados a los Sistemas de Filtrado de Información

Los sistemas de FI introducen una serie de problemas que habrá que considerar en su diseño [5, 91].

- En primer lugar, una vez que se ha establecido un perfil de intereses, es fácil considerar libremente las evaluaciones suministradas por otros. Sin embargo, en algunos casos se hace necesario recurrir a incentivos para la provisión de recomendaciones, puesto que los usuarios no suelen estar muy dispuestos a colaborar proporcionando información personal sobre sus preferencias y de esta forma definir su perfil. Estos incentivos podrían consistir en que a cambio de recibir recomendaciones, el usuario debe introducir obligatoriamente datos sobre sus preferencias, o bien asignarle otro tipo de compensaciones.
 - Un segundo problema a solucionar es que si cualquiera puede realizar recomendaciones, los propietarios de determinados productos podrían generar recomendaciones positivas de los mismos y negativas de otros.
-

- En los sistemas de FI también habrá que tener en cuenta aspectos de privacidad y debido a que algunas personas no quieren que se conozcan sus hábitos o preferencias, habría que considerar la participación anónima o bajo un pseudónimo.
- Otro problema es que el mantenimiento de un sistema de FI es costoso, por lo que se hace necesario pensar en modelos de negocio que se podrían usar para generar ingresos suficientes para cubrir dichos costes. Un modelo posible es que los receptores de las recomendaciones paguen una suscripción, es decir, pagar por usar. Un segundo modelo podría ser la inclusión de publicidad, que proporcionaría a los clientes información de mercado detallada. Un tercer modelo es cobrar una cuota a los propietarios de los elementos que deseen sean evaluados. Sin embargo, los dos últimos modelos pueden presentar problemas de corrupción, en el caso de importantes empresas con un gran peso publicitario.
- Aunque se ha demostrado que los sistemas de FI pueden llegar a ser suficientemente precisos en determinados dominios, hay otros dominios en los que existe un gran riesgo en aceptar las recomendaciones proporcionadas por el sistema.

Hay dos razones fundamentales por las que no se tiene confianza en estos sistemas en dominios de alto riesgo. Primera, estos sistemas son procesos que calculan predicciones basándose en modelos humanos que son aproximaciones heurísticas de procesos humanos. Segunda, y quizá más importante, la mayoría de las veces basan sus cálculos en datos incompletos y dispersos. Estas razones hacen que estos sistemas ocasionalmente generen

recomendaciones incorrectas, o sencillamente recomendaciones que no sean suficientemente precisas. Sin embargo los usuarios no reciben ningún tipo de indicador que les permita determinar la confianza de una recomendación cuando estén dudosos. En este sentido, una solución consistiría en incluir **explicaciones** con las recomendaciones, que en cualquier caso nos ayudaría a mejorar el rendimiento del sistema. Estas explicaciones añadirían transparencia al proceso y los usuarios estarían más dispuestos a confiar en las recomendaciones cuando conocieran las razones que hay detrás de una recomendación [34, 77, 78, 95].

Consideremos cómo actuamos nosotros cuando otras personas nos hacen alguna sugerencia. Cuando un amigo nos hace una sugerencia, consideramos cómo han funcionado anteriormente las recomendaciones de dicho amigo o comparamos los intereses generales o preferencias de nuestro amigo con las nuestras en el ámbito concreto en que se realiza la recomendación. Sin embargo, si aún tuviéramos dudas, preguntaríamos para que nuestro amigo nos explicara las razones que le han llevado a realizar esa sugerencia. Entonces, podríamos analizar la lógica de la sugerencia y determinar por nosotros mismos si es suficiente o no según nuestras expectativas. Viendo estos beneficios, parece lógico considerar automatizar esta idea de acompañar las recomendaciones de las explicaciones que han dado lugar a dicha recomendación.

Por tanto, incluir las explicaciones en un sistema de FI tiene las siguientes ventajas:

1. El usuario comprende el razonamiento que hay detrás de una recomendación, de forma que puede decidir qué credibilidad conceder a dicha recomendación.
2. El usuario se involucra en el proceso de recomendación, permitiéndole el uso de su conocimiento para completar el proceso de decisión.
3. El usuario se va familiarizando con el proceso de generación de recomendaciones de forma que puede ir reconociendo las fortalezas y debilidades del sistema.

En definitiva, las explicaciones nos aportan una mayor aceptación del sistema como ayuda de confianza en los procesos de toma de decisiones, puesto que sus limitaciones y potencialidades son visibles, y sus sugerencias se nos muestran justificadas.

- Un último problema es dotar a los sistemas de FI de mejores técnicas de representación de las preferencias o recomendaciones de los usuarios que nos permitan captar verdaderamente su concepto del objeto recomendado y así mejorar la interacción entre el sistema y los usuarios.

2.2.7. Evaluación de los Sistemas de Filtrado de Información. Métricas

A pesar de que este tipo de sistemas están tan extendidos, aún no hay estándares definidos para su evaluación empírica o teórica, sino que cada grupo de investigación ha ido aplicando diferentes técnicas y métricas de evaluación, sin que en

muchos casos existan relaciones entre ellas. Esta diversidad de métricas conlleva tres problemas [35]:

1. Si dos investigadores distintos evalúan sus sistemas con distintas métricas, los resultados no son comparables.
2. Si la métrica usada no está estandarizada, podemos pensar que los investigadores han elegido la métrica más adecuada de cara a obtener los resultados deseados.
3. Sin una métrica estandarizada, cada investigador debe realizar un esfuerzo extra en identificar o desarrollar una métrica apropiada.

En cualquier caso, para medir el rendimiento de un sistema de FI, debemos seguir tres pasos fundamentales:

1. Identificar a alto nivel los objetivos del sistema. Debemos determinar exactamente los objetivos del sistema así como las tareas que el usuario realizará con el sistema.
 2. Identificar las tareas específicas que permiten alcanzar esos objetivos. Estas tareas describirán explícitamente la naturaleza de la interacción entre el usuario y el sistema. La elección de una métrica apropiada para usar en la evaluación de un sistema dependerá de las tareas específicas que sean identificadas en esta fase. Como ejemplos de estas tareas podríamos mencionar el caso de un usuario que quiere localizar un ítem cuyo valor no exceda de un límite, o quiere conocer la mejor opción cuando se le plantean varias alterna-
-

tivas y debe tomar una decisión, o quiere examinar un flujo de información en un determinado orden de importancia, etc.

3. Identificar las métricas a nivel de sistema y realizar la evaluación. La evaluación a nivel del sistema se realizará en casos en los que los investigadores identifiquen indicadores que se puedan medir y que evidencien correlación con la efectividad del sistema independientemente de la interacción de los usuarios. Este tipo de evaluación es la más usada en FI, porque ofrece un análisis sencillo, es fácil de repetir y además los datos de los usuarios son recogidos una sola vez.

Cleverdon [15] identifica cinco medidas a tener en cuenta y que afectan a los usuarios de un sistema de FI:

- **Retardo.** Intervalo de tiempo transcurrido desde que se hace la demanda de información hasta que se da la respuesta. Esta medida es aplicada en los tradicionales sistemas de RI, en los que el usuario introduce consultas con sus necesidades de información.
 - **Presentación.** El formato físico de la salida del sistema.
 - **Esfuerzo del usuario.** El esfuerzo, intelectual o físico que se demanda del usuario.
 - **Exhaustividad (Recall).** Capacidad del sistema de recomendar todos los ítems relevantes. Formalmente se define como el porcentaje de ítems relevantes que son recomendados.
-

- **Precisión.** Capacidad del sistema de ocultar ítems que no sean relevantes. Se define como el porcentaje de ítems recomendados que son relevantes.

Las medidas más populares para evaluar sistemas de FI son la precisión y la exhaustividad. Se calculan a partir de una tabla de contingencia que categoriza los ítems con respecto a las necesidades de información. El conjunto de ítems debe ser clasificado en dos grupos: *relevantes* o *irrelevantes*. Además, también clasificamos los ítems según se hayan recomendado al usuario (*seleccionados*) o no (*no seleccionados*). Con estas cuatro categorías, construimos la tabla de contingencia (tabla 2.2):

	Seleccionados	No Seleccionados	Total
Relevantes	N_{rs}	N_{m}	N_r
Irrelevantes	N_{is}	N_{in}	N_i
Total	N_s	N_n	N

Tabla 2.2: Tabla de contingencia.

La *Precisión* se define como la proporción de ítems relevantes seleccionados con respecto al total de ítems seleccionados, es decir, mide la probabilidad de que un ítem seleccionado sea relevante:

$$P = \frac{N_{rs}}{N_s}$$

Por otro lado, la *Exhaustividad* se define como la proporción de ítems relevantes

seleccionados con respecto al total de ítems relevantes, es decir, representa la probabilidad de que un ítem relevante sea seleccionado.

$$R = \frac{N_{rs}}{N_r}$$

Ambas métricas dependen de la clasificación que se haga de ítems relevantes y no relevantes. Por tanto son métricas apropiadas para aquellos casos en los que hay un claro límite entre qué ítems satisfacen las necesidades de los usuarios y cuáles no.

Otro aspecto a considerar a la hora de evaluar un sistema de FI es su **Cobertura**. Se refiere únicamente al porcentaje de ítems para los que el sistema podría generar una recomendación, puesto que en ocasiones los sistemas de FI no son capaces de generar recomendaciones para determinados ítems debido a la ausencia de datos u otro tipo de restricciones. A la hora de evaluar un sistema, es muy importante determinar la cobertura que se alcanza con el mismo.

2.3. Perfiles de Usuario

Como hemos visto, el FI es un área de investigación que ofrece herramientas para discriminar entre información relevante e irrelevante, proporcionando asistencia personalizada a los usuarios en sus continuos procesos de acceso a la información [34, 91]. Tanto los sistemas de FI basados en contenidos como los colaborativos, comparten el hecho de basar las recomendaciones en las preferencias de los usuarios, representadas mediante perfiles de usuario, por lo que definir perfiles de los

usuarios es un aspecto clave en los sistemas de filtrado de información. Tal y como se indica en [64], un uso inadecuado de los perfiles de los usuarios provoca un rendimiento muy pobre a la hora de filtrar la información por lo que el usuario podría estar sobrecargado de información irrelevante, o bien podría no acceder a información relevante porque ésta haya sido rechazada.

Podemos distinguir dos tipos fundamentales de perfiles de usuario [64]:

1. **perfiles basados en contenidos**, que están representados por un vector de las áreas de interés de cada usuario, y
2. **perfiles colaborativos**, que están basados en las valoraciones de usuarios considerados similares, por lo que se pueden expresar como una lista de usuarios similares. Se basan en la idea de que a usuarios con sistemas de valores similares, probablemente les va a satisfacer el mismo tipo de información.

Además, hay dos propiedades deseables que se deben tener en cuenta a la hora de definir perfiles y son las siguientes:

1. Los perfiles de usuario deben ser adaptables a distintas situaciones, es decir, deben ser dinámicos, debido a que los intereses de los usuarios van cambiando continuamente. Esto implica la necesidad de incluir un módulo de aprendizaje en el sistema de filtrado de información para adaptar los perfiles de los usuarios según la realimentación que introduzcan en el sistema a partir de sus reacciones ante la información que les haya sido entregada.
-

2. La generación y actualización de los perfiles de usuario debe ser llevada a cabo con la menor implicación posible por parte de los usuarios, es decir, minimizando el grado de intervención de los mismos para reducir el esfuerzo que tengan que realizar y así facilitar la interacción sistema-usuario.

2.3.1. Generación de Perfiles de Usuario

Los perfiles de usuario representan, pues, las necesidades o intereses de los usuarios a largo plazo. Por tanto, debemos establecer cómo vamos a reunir la información sobre los usuarios, sus preferencias o necesidades, hábitos, etc. Hay tres enfoques fundamentales a los que pertenecen los métodos que podemos usar para llevar a cabo este proceso de recopilación de información sobre el usuario [34, 64, 88]:

- **Enfoque explícito:** los sistemas que adoptan este enfoque interaccionan directamente con los usuarios a través de un proceso de realimentación. Por tanto, en este enfoque los usuarios expresan ciertas especificaciones sobre la información a la que desean acceder. Este enfoque está bastante extendido debido a su sencillez.
 - **Enfoque implícito:** estos métodos realizan inferencias a partir de algún tipo de observación. Esta observación se puede realizar directamente sobre el comportamiento del usuario o bien sobre determinados entornos, como por ejemplo las URLs visitadas. Las preferencias del usuario son actualizadas cuando se detectan cambios a partir de dichas observaciones.
 - **Enfoque mixto:** que adopta características conjuntas de los otros dos en-
-

foques.

A partir de estos enfoques, se pueden desarrollar diversos métodos para la captación de información sobre los usuarios, como los que se especifican a continuación.

Interrogación a los usuarios

Esta es la técnica más simple y extendida de las que adoptan el enfoque explícito. Consiste en que los usuarios especifican sus preferencias, necesidades, áreas de interés o cualquier otro dato relevante a través de un *formulario* que tienen que rellenar. Otros sistemas proporcionan a los usuarios un *conjunto de términos* que representan el dominio de aplicación concreto del sistema, para que con dichos términos los usuarios generen su propio perfil personal. La idea es evitar confusiones semánticas que se observan en otros sistemas en los que los usuarios disponen de más libertad para elegir los términos, lo que puede derivar en ambigüedades. Otros sistemas más sofisticados permiten que los usuarios puedan asignar *pesos de importancia* a los términos que seleccionen para generar su perfil. También hay *sistemas de filtrado basados en reglas* que utilizan la interacción con los usuarios para definir las reglas de FI. Estos sistemas, habitualmente hacen uso de un editor de reglas que guía a los usuarios en las tareas de definición de reglas.

Analizar el comportamiento de los usuarios

Se trata de un enfoque implícito que no requiere ningún tipo de implicación de los usuarios en el proceso de adquisición de información sobre los mismos. En lugar de ello, esta técnica consiste en analizar las reacciones de los usuarios ante los ítems que se les presentan, para así realizar inferencias sobre la relevancia que

los ítems tendrán sobre los usuarios. Por ejemplo, un aspecto a tener en cuenta a la hora de analizar el comportamiento de los usuarios, podría ser el tiempo que invierten en leer un determinado ítem. Esta técnica es aplicada en GroupLens [91, 94], un sistema de filtrado colaborativo que usa el tiempo de lectura de un ítem como indicador de su relevancia para el usuario. Otros comportamientos de los usuarios que se pueden analizar son, por ejemplo, si ha salvado un documento, si directamente lo ha rechazado, si lo ha impreso o si lo ha reenviado.

Habitualmente, en sistemas reales este enfoque es acompañado de algún tipo de implicación por parte del usuario, tal y como ocurre en GroupLens [91, 94], que incluye realimentación de relevancia además de analizar el tiempo que invierten los usuarios en los ítems recuperados.

Espacio de documentos

Este método adopta un enfoque mixto de adquisición de conocimiento sobre los usuarios. El sistema crea un conjunto de documentos que previamente el usuario ha juzgado y evaluado como relevantes. Ante un nuevo documento, se calcula su similitud con los documentos existentes en el espacio de documentos relevantes. Si la similitud del nuevo documento está por encima de un determinado umbral, es considerado relevante. Es un método mixto, porque el usuario no define su perfil, pero sí se implica en el proceso de evaluar la relevancia de los documentos. Un inconveniente de este método es que puede funcionar bajo ciertos prejuicios que pudieran tener los usuarios, en aquellos casos en que ciertas áreas de interés no estén cubiertas por el espacio inicial de documentos.

Estereotipos

Se trata de otro método mixto de generación y mantenimiento de perfiles, mediante el cual el sistema carece de suficientes detalles o hechos sobre un usuario específico, de forma que el sistema adquiere información más detallada o verifica la información disponible sobre un usuario, basándose en su pertenencia a uno o varios estereotipos. Este método es recomendado cuando no se dispone de suficiente información sobre los usuarios o cuando la información disponible proviene de fuentes poco fiables (por ejemplo si procede de un análisis de la interacción en lenguaje natural entre el usuario y un sistema) [101]. Los usuarios deben proporcionar información explícita sobre sí mismos para permitir al sistema relacionarlos con algún estereotipo. Este conocimiento explícito es complementado por un proceso de inferencia implícita basado en la pertenencia de los usuarios a los distintos estereotipos.

Se distinguen dos métodos principales a la hora de incorporar estereotipos en el modelado de usuarios [101, 102]:

- El método *herramienta complementaria*, donde el estereotipo actúa como herramienta complementaria útil en la construcción de un modelo individual, cuando no se dispone de cierta información sobre el usuario. Se usan, pues, para añadir conocimiento nuevo a un perfil ya existente según el estereotipo o estereotipos a los que el usuario pertenezca.
 - El método *modelo completo*, donde se usan los estereotipos para construir un modelo íntegro sobre el usuario, basándose únicamente en la información contenida en los estereotipos a los que el usuario pertenece.
-

Entonces, un estereotipo va a contener información básica sobre grupos de usuarios relacionados entre sí [99, 101]. En este sentido se distinguen dos enfoques para clasificar a toda la población de usuarios en estereotipos y determinar los atributos y hechos que caracterizan a cada grupo:

- El *enfoque conductista* en el que la determinación de los grupos se realiza basándose en cuestionarios o en un conocimiento en profundidad del conjunto de la población y consultando con expertos en ciencias de la conducta. La mayoría de los sistemas basados en estereotipos adoptan este enfoque.
- El *enfoque matemático* en el que la determinación de los grupos estereotípicos se realiza a partir de algunas formas de cálculo, tales como clustering o teoría de grafos. Este enfoque es más preciso, pero la obtención de los datos necesarios para realizar los cálculos requiere un proceso de aprendizaje prolongado sobre las prácticas personalizadas de los usuarios, registrando sus actividades, los resultados conseguidos y sus grados de satisfacción ante distintas acciones.

El conjunto de datos que forman un estereotipo puede ser representado u organizado de varias formas, tales como en estructura jerárquica, en forma tabular, o bien en forma de reglas.

Una vez que están definidos los estereotipos, la asignación de los usuarios a los mismos se puede realizar mediante disparadores o mediante afinidad total:

- El método basado en *disparadores* consiste en que para ciertos atributos de cada estereotipo se determinan valores concretos que actúan como disparadores para hacer que el usuario pertenezca a dicho estereotipo.
- El método de *afinidad total* realiza una búsqueda del estereotipo más similar a todos los valores incluidos en el perfil del usuario.

A partir de este punto debemos tener en mente la necesidad de un proceso de actualización de estereotipos para el correcto funcionamiento del sistema. Ello incluye la actualización de cada uno de los componentes que hemos ido describiendo, es decir, la actualización de la clasificación en grupos esterotípicos, la actualización de los hechos y datos incluidos en cada estereotipo, así como la actualización de la asignación de los usuarios a estereotipos. La actualización es el resultado de un proceso de aprendizaje basado en la realimentación de los usuarios, así como en la modificación o aparición de nuevos datos sobre los mismos. Este proceso de actualización es muy importante para mantener la efectividad del sistema basado en estereotipos, ya que la escasez inicial de información provoca una mayor dificultad para el establecimiento preliminar de los estereotipos y sus atributos.

En [99, 100] se propone un modelo para aplicar el uso de estereotipos en un sistema de filtrado de información. En la figura 2.4 representamos la descripción funcional del modelo, que vemos que incluye cuatro bases de datos (D1 a D4) y tres procesos fundamentales (F1 a F3).

2.3.2. Aprendizaje de Perfiles de Usuario

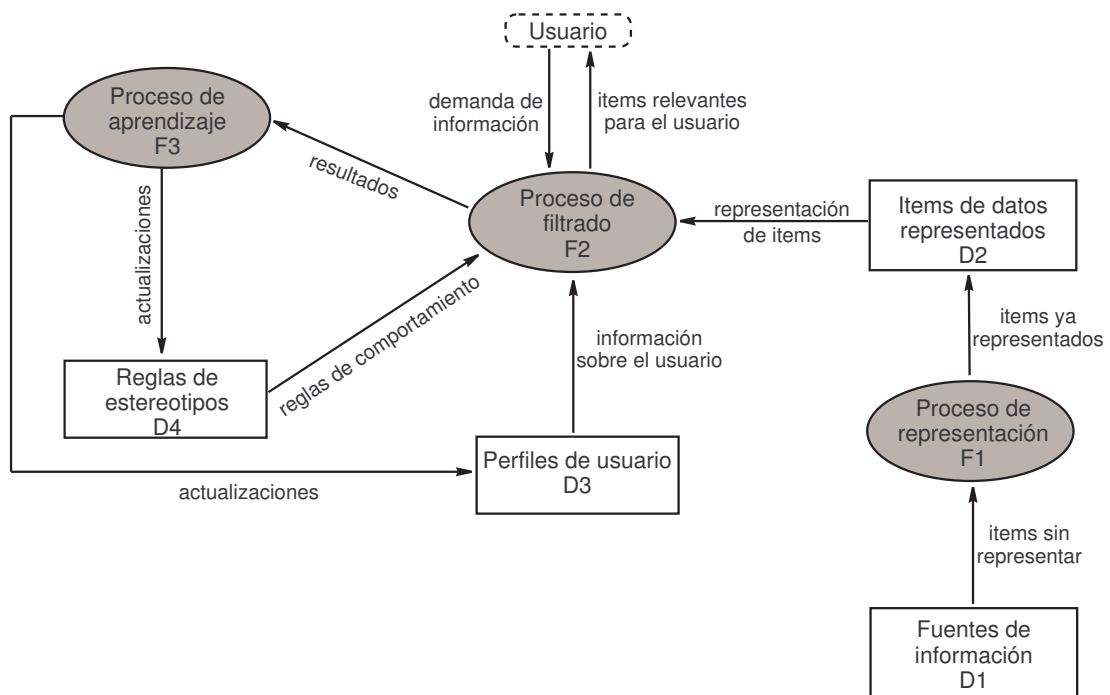


Figura 2.4: Modelo de filtrado de información con estereotipos.

Comentábamos antes que una propiedad que deben poseer los perfiles de usuario, es su capacidad de adaptación ante nuevas situaciones, es decir, que los perfiles de usuario deben ser dinámicos para lo que es necesario incluir en el sistema un módulo de aprendizaje. Este proceso de aprendizaje va a depender por un lado del método que se adopte para obtener información sobre los usuarios, y por otro lado de la frecuencia con la que se realice dicho proceso [34].

Método de aprendizaje

Podemos distinguir tres métodos de aprendizaje, que son los siguientes:

- **Aprendizaje por observación.** Con este método, las situaciones que provocan una determinada acción son memorizadas. Así, cuando se pro-

duce una nueva situación, es comparada con situaciones ya conocidas, y en función de ello se decide el curso de la acción o se sugiere una determinada acción. En el proceso de filtrado, ante un nuevo ítem se compara con alguno ya conocido para ver el comportamiento que tuvo el usuario.

- ***Aprendizaje por realimentación.*** En este caso es el usuario quien proporciona la realimentación, bien directamente indicando al sistema cómo actuar en una situación similar, o bien indirectamente, proporcionando algún tipo de información como por ejemplo la relevancia que para él ha tenido algún ítem.
- ***Aprendizaje por entrenamiento de los usuarios.*** Consiste en que el usuario introduce situaciones hipotéticas y las acciones deseadas del sistema, construyéndose así una base de datos de posibles escenarios. El sistema usará estos escenarios cuando tenga que decidir en posteriores situaciones. En sistemas de filtrado, los usuarios pueden proporcionar al sistema las evaluaciones de los ítems previamente recuperados para así crear o actualizar sus perfiles y tener en cuenta dichas evaluaciones ante nuevos ítems.

Frecuencia de aprendizaje

Esta propiedad se refiere a cuándo se va a llevar a cabo el proceso de aprendizaje.

Hay dos posibilidades:

- ***Aprendizaje crítico,*** que es aplicado cuando se detecta alguna contradicción entre la información disponible y una nueva información. Un sistema que implemente este método, debe chequear posibles contradicciones después de cada sesión de obtención de información.
-

- *Aprendizaje periódico*, consistente en recopilar la realimentación producida durante un período de tiempo y luego compararla con la información ya existente sobre los usuarios.

2.4. Sistemas de Filtrado de Información Difusos

En esta sección nos centramos en estudiar cómo la Teoría de Conjuntos Difusos ha contribuido a solucionar el problema de la representación de las preferencias o recomendaciones aportando mejores estructuras de representación y técnicas de manejo de la información. En concreto, podemos aplicar la lógica difusa en dos ámbitos:

1. Para modelar las preferencias y opiniones de los usuarios.
2. Para modelar los operadores de agregación, que están relacionados con el problema de combinar diferentes valoraciones (puntuaciones, preferencias o graduaciones) de varias fuentes de información. Se proponen una serie de operadores de agregación basados en el operador OWA (Ordered Weighted Averaging) [109].

Vamos a describir brevemente el papel que ambos juegan en los sistemas de FI. Recordemos que, tradicionalmente, estos sistemas se clasifican en dos categorías, los sistemas de filtrado basado en contenidos y los sistemas de filtrado colaborativo. Pues bien, en un sistema de filtrado basado en contenidos, es el propio sistema

quién realiza las recomendaciones, es decir, que es tarea del sistema proporcionar recomendaciones a los usuarios como por ejemplo sobre el orden en que deben ser considerados los documentos de una colección. En este sentido, el sistema de FI es casi como un sistema de RI [30]. Por otro lado, en un sistema de filtrado colaborativo, las recomendaciones son hechas por los usuarios del sistema, es decir, que la tarea del sistema es sintetizar múltiples recomendaciones de los usuarios en una única recomendación para un usuario individual [30].

En ambos casos de sistemas de FI, el método de generación de recomendaciones abarca los dos siguientes pasos [113]:

1. Cálculo de grados de similaridad. En el caso de un sistema basado en contenidos, los grados de similaridad son calculados entre un nuevo ítem sin evaluar y otros ítems que el usuario ha experimentado y evaluado positivamente. En el caso de un sistema colaborativo, los grados de similaridad son calculados entre perfiles de usuario, sin considerar en dicho proceso la representación de los ítems.
2. Agregación de evaluaciones. En el caso de los sistemas basados en contenidos las evaluaciones son proporcionadas por el usuario que recibe la recomendación, mientras que en el caso de sistemas colaborativos las evaluaciones son proporcionadas por otros usuarios.

Centrándose en el ámbito del modelado de las preferencias de los usuarios, en [85, 86] se desarrolla un método de filtrado basado en relaciones de preferencia

difusas y se aplica al campo de la toma de decisiones. La situación concreta es que determinados individuos actúan como asesores de un individuo que busca recomendaciones para sus elecciones personales. Se supone que los distintos individuos han expresado sus preferencias, por ejemplo usando grados de satisfacción sobre los ítems. El objetivo es proporcionar a cada individuo recomendaciones relevantes de ítems de interés que aún no haya evaluado.

Para ello se propone un sistema relacional difuso que permite representar la similitud entre ítems, similitud entre individuos y las preferencias de los individuos sobre los ítems. Para calcular la similitud entre ítems, se construye una relación de similitud basada en la importancia relativa de múltiples atributos tenidos en cuenta en la representación de los ítems. Las relaciones de preferencia difusas son usadas para expresar opiniones difusas positivas o negativas de los usuarios con respecto a los ítems. A partir del perfil de preferencias personal de cada usuario, se construyen dos relaciones de preferencias difusas, la que representa la parte positiva de las preferencias y la que representa su parte negativa. Al igual que se hace con la comparación de objetos, la similitud entre individuos podría estar basada en un análisis multi-atributo de los perfiles característicos de cada usuario. Sin embargo, recordemos que una de las características de un sistema de FI, es la posibilidad de participación anónima o bajo un pseudónimo, lo cual nos impide disponer de un perfil detallado del usuario. Por ello, la información usada en la generación de recomendaciones es la que se tenga sobre las evaluaciones o preferencias de los usuarios con respecto a los ítems, es decir, su sistema de valores. Por tanto, el sistema de valores es la información que define el perfil de preferencias difusas de cada usuario y la información con la que se trabaja para calcular la similitud entre usuarios.

Para que los métodos de filtrado colaborativos puedan generar recomendaciones de interés, requieren de un cierto número de usuarios. En el caso de que el número de usuarios disponibles en el sistema no sea suficiente, se debería optar por un método de filtrado basado en contenidos, por lo que en [85, 86] se propone un enfoque híbrido integrando los dos mecanismos de filtrado, el colaborativo y el basado en contenidos, de forma que a cada usuario se le proporciona un subconjunto difuso de posibles ítems de interés. Con un parámetro se controla la influencia del resto de usuarios en el proceso de recomendación al usuario activo, de forma que al principio a este parámetro se le da un valor para que el filtrado sea puramente basado en contenidos y conforme el número de usuarios va creciendo, progresivamente se va modificando el parámetro para que el filtrado sea cada vez más colaborativo.

Para experimentar este enfoque, se construyó un SR de películas denominado *Film Conseil* [29, 86]. En dicho sistema los ítems a considerar son películas, cada una representada por una serie de atributos (título, género, origen, duración, director, año de producción, protagonistas, etc.), de forma que en la base de datos se van almacenando tuplas con los valores correspondientes a cada uno de los atributos. Cada individuo se representa por su perfil de preferencias que viene dado por las valoraciones asignadas a cada una de las películas que haya visto, según su grado de satisfacción. La similitud difusa entre películas y las relaciones de influencia difusas entre usuarios, son calculadas periódicamente y almacenadas en bases de datos independientes. Al usuario se le proporciona una lista ordenada de recomendaciones, con la posibilidad de acompañarlas de una explicación de dichas recomendaciones generada automáticamente por el sistema.

Por tanto, el rendimiento de un sistema de FI depende fuertemente de la disponibilidad del sistema de permitir al usuario expresar eficientemente sus preferencias. Esta capacidad, a su vez, depende tanto de las afirmaciones y atributos usados para representar los ítems, como de la sofisticación del lenguaje con que el usuario expresa sus preferencias. Para ello se propone un lenguaje basado en el operador de agregación OWA [109], los cuantificadores lingüísticos difusos, y sistemas de reglas difusos, que proporcionan métodos muy adecuados para representar expresiones del lenguaje natural. En el capítulo siguiente estudiaremos el modelado lingüístico difuso, para el manejo eficiente de información lingüística.

2.5. Ejemplos de Sistemas de Filtrado de Información

En esta sección, vamos a analizar algunos ejemplos de sistemas de FI existentes en Internet. Destacar los sistemas presentados en [91], ampliamente conocidos y que han servido de base para numerosos estudios posteriores:

- *PHOAKS*: Se trata de un sistema experimental para solucionar el problema de encontrar información relevante y de alta calidad en la Web, usando el enfoque colaborativo en el que los usuarios recomiendan determinados ítems a otros usuarios. PHOAKS trabaja reconociendo, concordando y redistribuyendo automáticamente recomendaciones de recursos Web extraídos de mensajes de noticias.

- *Referral Web*: Numerosos estudios muestran que una de las formas más efectivas de divulgar información y conocimiento dentro de una determinada organización es a través de una red informal de colaboradores o amigos. Referral Web se basa en la idea de combinar *redes sociales* con el filtrado colaborativo, entendiendo por redes sociales grupos de personas vinculadas por determinadas actividades profesionales.
- *FAB*: Sistema orientado a la recomendación de URLs que combina el uso de información por extensión con el enfoque colaborativo.
- *Siteseer*: Recomienda páginas Web relevantes y usa las listas de favoritos y la organización de registros como una declaración implícita de intereses respecto al contenido subyacente, y se va midiendo el grado de solapamiento con las de otros usuarios.
- *GroupLens*: El proyecto GroupLens diseña, implementa y evalúa un sistema de filtrado colaborativo para Usenet, un servicio de listas de discusión con un alto volumen de negocio en Internet.

Anteriormente, en el apartado **2.2.5** definíamos cinco aspectos o dimensiones a considerar a la hora de diseñar un sistema de FI. Pues bien, en la tabla 2.3 mostramos cómo encuadran en dichas dimensiones estos sistemas de ejemplo.

Más recientes son los dos sistemas que presentamos a continuación, ambos aplicados en el ámbito de las recomendaciones musicales:

- *MusicStrands* [81]. Se trata de una empresa surgida del Instituto de Investigación de Inteligencia Artificial del Consejo Superior de Investigaciones
-

	Contenidos de la recomendación	Tipo de entrada	Identificación de la fuente	Modo de agregación	Uso de las recomendaciones
GroupLens	Númérico: 1-5	Explícita	Seudónima	Suma pesada basada en acuerdos anteriores de los recomendadores	Visualización junto a los artículos en las vistas de resúmenes
Fab	Númérico: 1-7	Explícita	Seudónima	Suma pesada junto con análisis de contenidos	Selección / Filtrado
ReferralWeb	Recomendación de una persona o documento	Extraída de datos públicos	Atribuida	Reunir cadenas referidas a la persona deseada	Visualización
PHOAKS	Recomendación de una URL	Extraída de envíos Usenet	Atribuida	Un voto por persona (por URL)	Visualización ordenada
Siteseer	Recomendación de una URL	Extraída de listas de favoritos	Anónima	Frecuencia de mención de la URL	Visualización

Tabla 2.3: Clasificación de los sistemas de FI de ejemplo.

Científicas (CSIC) y la Universidad de Oregón, que ha desarrollado un sistema que automatiza las recomendaciones musicales mediante el análisis de las pautas de consumo que ofrecen las listas de reproducción que crean los usuarios. El sistema sincroniza las bibliotecas musicales del usuario en lectores como iTunes o Windows Media y aparatos portátiles MP3 y analiza mediante algoritmos patentados cuándo, cuánto y en qué orden escucha las piezas el aficionado, aunque provengan de descargas ilegales. A continuación establece conexiones con los datos del resto de usuarios. El sistema, que cuenta ya con una base de datos de 5 millones de canciones, permite buscar canciones, escucharlas durante 30 segundos sin coste alguno y comprarlas a través de las firmas con las que tiene acuerdos. Para acceder al sistema es necesario registrarse, proceso que completamos de una forma sencilla y gratuita. Una vez registrados, cada vez que accedemos al sistema se nos muestra



Figura 2.5: Menú de cabecera de MusicStrands.

una pantalla con lo más destacado de cada categoría, como son canciones destacadas de la semana, top canciones, top artistas, canciones más recomendadas. En cualquier caso, en la parte superior aparece un menú general desde el que podemos acceder a las distintas opciones (ver figura 2.5)

Para establecer nuestro perfil, tendremos que ir buscando canciones o grupos e ir añadiéndolos a nuestras preferencias. Por ejemplo, cuando buscamos un determinado álbum de algún grupo en el que estemos interesados, se nos muestra la información de dicho álbum (figura 2.6), pero también se nos muestran recomendaciones sobre otros álbumes completos (ver figura 2.7), canciones concretas (ver figura 2.8) o sobre otros artistas (ver figura 2.9).

- *MusicSurfer* [82]. Se trata de una tecnología desarrollada por la Universitat Pompeu Fabra que permite que el ordenador sea capaz de imitar el comportamiento humano a la hora de escuchar, entender y recomendar música. Esta tecnología posibilita nuevos métodos de acceso a las librerías de música personales o a grandes bases de datos. Está basada en el análisis del contenido musical de la canción usando técnicas de procesamiento de señal a bajo nivel



The screenshot shows a music album page for "Un Metro Cuadrado 1m2" by Jarabe De Palo. On the left is the album cover featuring a man in a red shirt and blue jeans sitting on a yellow wall with the number "12" written on it. To the right of the cover, the album title "Un Metro Cuadrado 1m2" is displayed in a large, bold font. Below the title, it says "por **Jarabe De Palo**", "publicado: 2005", and "en la discográfica: WEA Latina". The genre is listed as "Género: International". On the far right, there are three buttons: "Obtener", "Añadir a Mis Deseos", and "Añadir a Favoritos". At the bottom of the page, there are four tabs: "Resumen", "Artículos de este álbum en las revistas", "Recomendaciones", and "Obtener".

Figura 2.6: Resultado de la búsqueda de un álbum.



The screenshot shows a "Recomendaciones" (Recommendations) section. At the top, there is a blue header with the word "Recomendaciones" and a row of white stars. Below this is a grey bar with the text "Álbumes recomendados". The main content area displays six recommended albums in a two-column grid. Each album is represented by its cover art, the album title, the artist's name, and the release year.

Álbum	Artista	Año
El Kilo	Orishas	2005
Monster	R.E.M.	1994
Sabina Y Cia: Nos Sobran Los Motivos	Joaquin Sabina	2001
You Are The Quarry	Morrissey	2004
Todo Esto Es Muy Extranõ	Hombres G	2005
Pafuera Telaranas	Bebe	2004

Figura 2.7: Álbumes recomendados.

Canciones recomendadas

	<u>Gatas, A</u> en <u>Mas Turbada Que Nunca</u> por Gloria Trevi		<u>La Tortura</u> en <u>Fijacion Oral</u> por Shakira
	<u>Naci Orishas</u> en <u>El Kilo</u> por Orishas		<u>Grita</u> en <u>La Flaca</u> por Jarabe De Palo
	<u>Baila Casanova</u> en <u>Border Girl</u> por Paulina Rubio		<u>Quiero Saber</u> en <u>Volare! The Very Best Of The Gipsy Kings</u> por Gipsy Kings

Figura 2.8: Canciones recomendadas.

y, mediante técnicas de inteligencia artificial, la extracción de parámetros semánticos de alto nivel. Permite un análisis perceptual y musicológico de la música a partir del audio. El ordenador es capaz de extraer descriptores de armonía, instrumentación, ritmo, tipos de voz, etc. Una vez almacenados en una base de datos y vinculados a los ficheros de música, el sistema permite hacer recomendaciones de canciones basándose en las similitudes existentes entre las mismas. El criterio de estas similitudes puede ser personalizado por el usuario y especificar, por ejemplo, si se buscan canciones del mismo estilo musical, el mismo tipo de instrumentación, la misma energía, etc.

Artistas recomendados

<p><u>Shakira</u></p> <p>This photogenic Colombian singer began her recording career at the tender age of 13. It wasn't long before her eclectic brand of Latin pop became a sensation in numerous Spanish-speaking countries. In 1998, without making the concession of singing in...</p>	<p><u>Cafe Tacuba</u></p> <p>Mexican rockers Cafe Tacuba are generally regarded as being at the forefront of the Rock En Espanol movement, alongside Mana, Aterciopelados, et al. However, there's considerably less of the pan-Latin rhythmic sensibility (and a greater eclecticism) ...</p>
<p><u>Julieta Venegas</u></p>	<p><u>U2</u> </p>
<p><u>Alejandro Sanz</u></p> <p>b. 18 December 1968, Madrid, Spain. Balladeer and pop singer Alejandro Sanz began recording for WEA Latina in 1991 with <i>Viviendo Deprisa</i>. His fourth long player, 1997's <i>Más</i>, emerged as the success story of 1997 in Spain. Indeed, the album is widely a...</p>	<p><u>Mana</u></p> <p>Probably the biggest of the Rock en Espanol bands, Mexican group Mana has a long history. The core members started recording together under another name in the early 1980s, finally releasing the debut Mana album in 1988. Their sound mixes traditional...</p>

Figura 2.9: Artistas recomendados.

Capítulo 3

Modelado Lingüístico Difuso de la Información

En este capítulo vamos a estudiar las distintas técnicas de modelado lingüístico difuso para el manejo de información lingüística, que nos van a proporcionar una mayor flexibilidad en el tratamiento de la información, especialmente en los casos en que se produce una interacción con los usuarios.

3.1. Introducción

La Lógica Difusa se plantea como alternativa a la lógica tradicional, con el objetivo de introducir grados de incertidumbre en las sentencias que califica [119]. Hay numerosas situaciones en las que la lógica tradicional funciona perfectamente. Por ejemplo, supongamos que partimos de las calificaciones obtenidas en una clase y queremos agrupar a los aprobados (aquellos que hayan obtenido una calificación igual o superior a 5). El proceso de razonamiento que se seguiría mediante la lógica tradicional sería ir comparando cada calificación con 5 hasta obtener cuáles están aprobados y cuáles no:

Es cierto que $7 \geq 5$? SI: Aprobado

Es cierto que $4 \geq 5$? NO: No aprobado

Es cierto que $5 \geq 5$? SI: Aprobado

Sin embargo, el inconveniente de esta lógica es que en la vida real no nos encontramos frecuentemente con criterios de clasificación tan tajantes como en el ejemplo. En efecto, hay numerosas situaciones en las que la información no puede ser evaluada cuantitativamente de forma precisa, pero puede que sí sea posible hacerlo cualitativamente, y en estos casos hemos de hacer uso de un *enfoque lingüístico*. Por ejemplo, cuando intentamos cualificar algún fenómeno relacionado con percepciones humanas, a menudo usamos palabras o descripciones en lenguaje natural, en lugar de valores numéricos. Supongamos que dado un conjunto de personas, las intentamos agrupar según su altura. Las personas no son sólo *altas* o *bajas* sino que la mayoría pertenecen a grupos de altura intermedia. La gente suele ser *más bien alta* o *de altura media*. Casi nunca las calificamos con rotundidad, porque el lenguaje que usamos nos permite introducir modificadores que añaden imprecisión: *un poco*, *mucho*, *algo...*

Como la lógica tradicional es bivaluada (sólo admite dos valores: o el elemento pertenece al conjunto o no pertenece), se ve maniatada para agrupar según su altura al anterior conjunto de personas, puesto que su solución sería definir un umbral de pertenencia (por ejemplo, un valor que todo el mundo considera que, de ser alcanzado o superado, la persona en cuestión puede llamarse *alta*). Si dicho umbral es 1.80, todas las personas que midan 1.80 o más serán *altas*, mientras que el resto serán *bajas*. Según esta manera de pensar, alguien que mida 1.79 será tratado igual que otro que mida 1.60, ya que ambos han merecido el calificativo de personas *bajas*.

Si dispusiéramos de una herramienta para caracterizar las alturas de forma que las transiciones entre las que son altas y las que no lo son fueran suaves, estaríamos reproduciendo la realidad mucho más fielmente. En la realidad hay unos puntos de cruce donde las personas dejan de ser *altas* para ser consideradas *medianas*, de forma que el concepto de *alto* decrece linealmente con la altura. Asignando una función lineal para caracterizar el concepto *alto* en lugar de definir un sólo umbral de separación estamos dando mucha más información acerca de los elementos. Esta función, como veremos, se llamará función de pertenencia.

En este sentido, el uso de la Teoría de Conjuntos Difusos ha dado muy buenos resultados para el tratamiento de información de forma cualitativa [116]. El *modelado lingüístico difuso* es una herramienta que permite representar aspectos cualitativos y que está basada en el concepto de *variables lingüísticas*, es decir, variables cuyo valores no son números, sino palabras o sentencias expresadas en lenguaje natural o artificial [116]. Cada valor lingüístico se caracteriza por un valor sintáctico o *etiqueta* y un valor semántico o *significado*. La etiqueta es una palabra o sentencia perteneciente a un conjunto de términos lingüísticos y el significado es un subconjunto difuso en un universo de discurso.

Se ha demostrado que es una herramienta muy útil en numerosos problemas, como por ejemplo en la toma de decisiones [55, 106, 109], evaluación de la calidad informativa de documentos Web [46], modelos de recuperación de información [11, 38, 39], diagnósticos clínicos [22], análisis político [2], etc.

En este capítulo, vamos a revisar los principales enfoques de modelado lingüístico

difuso que podemos usar para el manejo de información lingüística. En la Sección 2 vamos a revisar los conceptos básicos para el manejo de información lingüística. En la Sección 3 vamos a tratar el modelo tradicional, el modelado lingüístico difuso clásico. En la Sección 4 veremos el modelado lingüístico difuso ordinal definido para eliminar la excesiva complejidad del enfoque lingüístico tradicional. En la Sección 5 nos centraremos en el enfoque de las 2-tuplas, definido como una mejora del anterior. En la Sección 6 estudiaremos el enfoque lingüístico difuso multi-granular que al permitir trabajar con distintos conjuntos de etiquetas nos será muy útil en aquellos casos en los que no sea eficiente valorar la información usando un mismo sistema de valores. Para finalizar, en la Sección 7 veremos el enfoque lingüístico difuso no balanceado para aplicar en aquellas situaciones en las que la información necesite ser valorada sobre un conjunto de etiquetas no uniforme, es decir, asimétrico.

3.2. Conceptos Básicos de Información Lingüística

Vamos a comenzar presentando una revisión de los conceptos básicos de la Teoría de Conjuntos Difusos que van a ser utilizados en el resto de modelados lingüísticos.

El interés de la Teoría de Conjuntos Difusos se centra esencialmente en modelar aquellos problemas donde los enfoques clásicos de la Teoría de Conjuntos y la Teoría de la Probabilidad resultan insuficientes o no operativos. Por ello, generaliza la noción clásica de conjunto e introduce el concepto de *ambigüedad*, de manera

que los conjuntos difusos nos proporcionan una nueva forma de representar la imprecisión e incertidumbre presentes en determinados problemas.

3.2.1. Conjuntos Difusos y Funciones de Pertenencia

La noción de conjunto refleja la tendencia a organizar, generalizar y clasificar el conocimiento sobre los objetos del mundo real. El encapsulamiento de los objetos es una colección cuyos miembros comparten una serie de características o propiedades que implican la noción de conjunto. Los conjuntos introducen una noción de dicotomía, que en esencia es una clasificación binaria: o se acepta o se rechaza la pertenencia de un objeto a una categoría determinada. Habitualmente la decisión de *aceptar* se denota como 1 y la de *rechazar* como 0. Esta decisión de aceptar o rechazar se expresa mediante una función característica, según las propiedades que posean los objetos del conjunto.

La Lógica Difusa se fundamenta en el concepto de *conjunto difuso* [115] que suaviza el requerimiento anterior y admite valores intermedios en la función característica, que se denomina *función de pertenencia*. Esto permite una interpretación más realista de la información, puesto que la mayoría de las categorías que describen los objetos del mundo real, no tienen unos límites claros y bien definidos.

Un conjunto difuso puede definirse como una colección de objetos con valores de pertenencia entre 0 (exclusión total) y 1 (pertenencia total). Los valores de pertenencia expresan los grados con los que cada objeto es compatible con las propiedades o características distintivas de la colección. Formalmente podemos

definir un conjunto difuso como sigue.

Definición 3.1. Un *conjunto difuso* \tilde{A} sobre un dominio o universo de discurso U está caracterizado por una función de pertenencia que asocia a cada elemento del conjunto el grado con que pertenece a dicho conjunto, asignándole un valor en el intervalo $[0,1]$:

$$\mu_{\tilde{A}} : U \rightarrow [0, 1]$$

Así, un conjunto difuso \tilde{A} sobre U puede representarse como un conjunto de pares ordenados de un elemento perteneciente a U y su grado de pertenencia, $\tilde{A} = \{(x, \mu_{\tilde{A}}(x)) / x \in U, \mu_{\tilde{A}}(x) \in [0, 1]\}$. Por ejemplo, consideremos el concepto *persona alta*, en un contexto donde la estatura oscila entre 1 y 2 m. Como es de suponer, alguien que mida 1,30m. no se puede considerar como *persona alta* por lo que su grado de pertenencia al conjunto de personas altas será de 0. Por el contrario, una persona que mida 1,90m. sí la consideramos alta por lo que su grado de pertenencia al conjunto es de 1.

Las gráficas que representan una función de pertenencia pueden adoptar cualquier forma, cumpliendo propiedades específicas, pero es el contexto de la aplicación lo que determina la representación más adecuada en cada caso. Puesto que las valoraciones lingüísticas dadas por los usuarios son únicamente aproximaciones, algunos autores consideran que las funciones de pertenencia trapezoidales lineales son suficientemente buenas para capturar la imprecisión de tales valoraciones lingüísticas. La representación paramétrica es obtenida a partir de una 4-tupla (a, b, α, β) , donde a y b indican el intervalo en que el valor de pertenencia es 1, con α y β indicando los límites izquierdo y derecho del dominio de definición

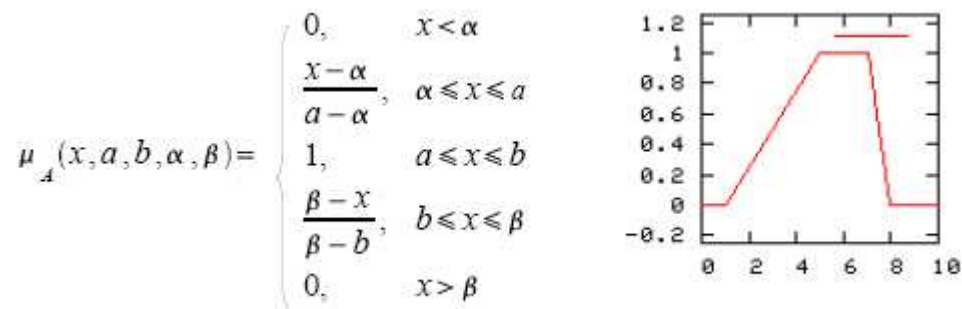


Figura 3.1: Ejemplo de función de pertenencia.

de la función de pertenencia trapezoidal. Un caso particular de este tipo de representación son las valoraciones lingüísticas cuyas funciones de pertenencia son triangulares, es decir, $a = b$, por lo que se representan por medio de una 3-tupla (a, α, β) . La figura 3.1 muestra la descripción y la representación gráfica de un ejemplo de función de pertenencia trapezoidal.

3.2.2. Definiciones Básicas

Definición 3.2. Se define el **soporte** de un conjunto difuso \tilde{A} en el universo U , como el conjunto formado por todos los elementos de U cuyo grado de pertenencia a \tilde{A} sea mayor que 0:

$$\text{supp}(\tilde{A}) = \{x \in U / \mu_{\tilde{A}}(x) > 0\}$$

Definición 3.3. La **altura** de un conjunto difuso \tilde{A} se define como el mayor grado de pertenencia de todos los elementos de dicho conjunto:

$$h(\tilde{A}) = \max\{\mu_{\tilde{A}}(x) / x \in U\}$$

Definición 3.4. El α -*corte* de un conjunto difuso \tilde{A} es el conjunto formado por todos los elementos del universo U cuyos grados de pertenencia en \tilde{A} son mayores o iguales que el valor de corte $\alpha \in [0, 1]$:

$$\alpha_{\tilde{A}} = \{x \in U / \mu_{\tilde{A}}(x) \geq \alpha\}$$

Definición 3.5. Se denomina *conjunto de niveles* de un conjunto difuso \tilde{A} , al conjunto de grados de pertenencia de sus elementos:

$$L(\tilde{A}) = \{a / \mu_{\tilde{A}}(x) = a, x \in U\}$$

3.2.3. Operaciones con Conjuntos Difusos

Al igual que en la lógica tradicional, las operaciones lógicas que se pueden establecer entre conjuntos difusos son la intersección, la unión y el complemento. Mientras que el resultado de operar dos conjuntos clásicos es un nuevo conjunto clásico, las mismas operaciones con conjuntos difusos nos darán como resultado otros conjuntos también difusos.

Hay muchas formas de definir estas operaciones. Cualquier operación que cumpla las propiedades de una t-norma puede ser usada para hacer la intersección, de igual manera que cualquier operación que cumpla las propiedades de una t-conorma

	Propiedades	Ejemplos
T-Normas $T: [0,1] \times [0,1] \rightarrow [0,1]$ $\mu_{A \cap B}(x) = T[\mu_A(x), \mu_B(x)]$	Conmutativa: $T(a,b) = T(b,a)$ Asociativa: $T(a, T(b,c)) = T(T(a,b), c)$ Monotonía: $T(a,b) \geq T(c,d)$ si $a \geq c$, y $b \geq d$ Condiciones frontera: $T(a,1) = a$	Intersección estándar $T(a,b) = \min(a,b)$ Producto algebraico $T(a,b) = a \cdot b$ Intersección drástica $T(a,b) = \begin{cases} a, & \text{si } b=1 \\ b, & \text{si } a=1 \\ 0, & \text{en otro caso} \end{cases}$
T-Conormas $S: [0,1] \times [0,1] \rightarrow [0,1]$ $\mu_{A \cup B}(x) = S[\mu_A(x), \mu_B(x)]$	Conmutativa: $S(a,b) = S(b,a)$ Asociativa: $S(a, S(b,c)) = S(S(a,b), c)$ Monotonía: $S(a,b) \geq S(c,d)$ si $a \geq c$, y $b \geq d$ Condiciones frontera: $S(a,0) = a$	Unión estándar $S(a,b) = \max(a,b)$ Suma algebraica $S(a,b) = a + b - a \cdot b$ Unión drástica $S(a,b) = \begin{cases} a, & \text{si } b=0 \\ b, & \text{si } a=0 \\ 1, & \text{en otro caso} \end{cases}$

Tabla 3.1: T-normas y T-conormas.

puede ser empleada para la unión. La tabla 3.1 muestra las propiedades que deben cumplir las dos familias de funciones y algunos ejemplos.

Las operaciones se definen de la siguiente manera:

- Intersección: $\tilde{A} \cap \tilde{B} = \{(x, \mu_{\tilde{A} \cap \tilde{B}}) / \mu_{\tilde{A} \cap \tilde{B}}(x) = T[\mu_{\tilde{A}}(x), \mu_{\tilde{B}}(x)]\}$
- Unión: $\tilde{A} \cup \tilde{B} = \{(x, \mu_{\tilde{A} \cup \tilde{B}}) / \mu_{\tilde{A} \cup \tilde{B}}(x) = S[\mu_{\tilde{A}}(x), \mu_{\tilde{B}}(x)]\}$
- Complemento: $\mu_{\sim \tilde{A}}(x) = 1 - \mu_{\tilde{A}}(x)$

En la figura 3.2 podemos ver una representación gráfica de dichas operaciones.

3.2.4. Modelado Lingüístico Difuso

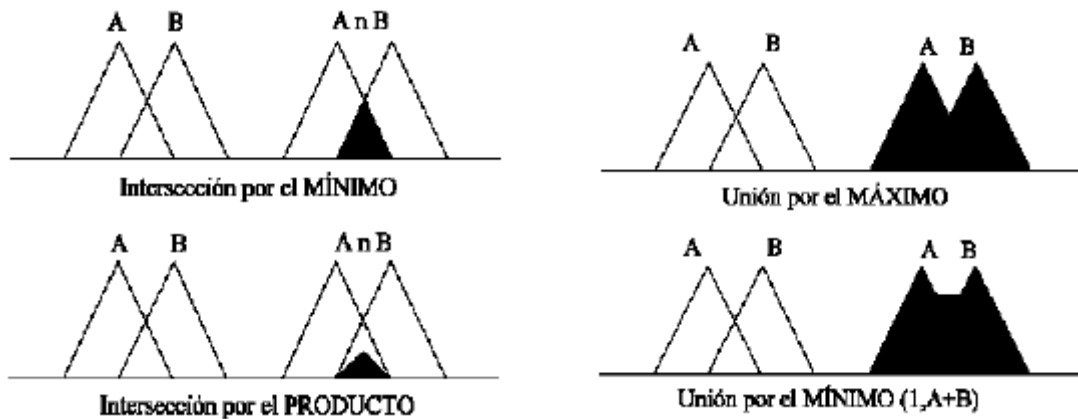


Figura 3.2: Intersección y Unión en conjuntos difusos.

La información que manejamos en el mundo real puede tener diferentes rangos de valoración y los valores pueden tener distinta naturaleza. En ocasiones, puede que no sea fácil valorarla de forma precisa mediante un valor cuantitativo, sin embargo puede que sí sea factible hacerlo de forma cualitativa. En este caso, adoptar un enfoque lingüístico suele ofrecer mejores resultados que si aplicamos uno numérico. Por ejemplo, cuando evaluamos determinados aspectos relacionados con la percepción subjetiva (*diseño, gusto, diversión, etc.*), solemos utilizar palabras en lenguaje natural en lugar de valores numéricos (*bonito, feo, dulce, salado, mucha, poca, etc.*). Esto hecho se puede deber a diversas causas:

- Hay situaciones en las que la información, por su propia naturaleza, no puede ser cuantificada y por tanto únicamente puede ser valorada mediante el uso de términos lingüísticos, como sucede cuando realizamos una valoración sobre un libro que hayamos leído, que solemos usar términos como *bueno, regular o malo*.
- En otros casos, trabajar con información precisa de forma cuantitativa no

es posible, o bien porque no están disponibles los elementos necesarios para llevar a cabo una medición exacta de esa información, o bien porque el coste computacional es demasiado alto y nos basta con la aplicación de un valor aproximado. Por ejemplo, cuando evaluamos la velocidad de una motocicleta, en lugar de usar valores numéricos, solemos usar términos tales como *rápida*, *muy rápida* o *lenta*.

Variables lingüísticas

El modelado lingüístico difuso es, pues, un enfoque aproximado basado en la Teoría de Conjuntos Difusos. Este modelo representa los aspectos cualitativos como valores lingüísticos mediante lo que se conoce como *variables lingüísticas* [116]. Una variable lingüística se caracteriza por un *valor sintáctico* o *etiqueta* que es una palabra o frase perteneciente a un conjunto de términos lingüísticos, y por un *valor semántico* o *significado* de dicha etiqueta que viene dado por un subconjunto difuso en un universo de discurso. Formalmente se define de la siguiente manera.

Definición 3.6. [116] Una *variable lingüística* está caracterizada por una 5-tupla $(H, T(H), U, G, M)$, donde:

- H es el nombre de la variable;
 - $T(H)$ (o sólo T) simboliza el conjunto de términos lingüísticos de H , es decir, el conjunto de nombres de valores lingüísticos de H , donde cada valor es una variable difusa denotada genéricamente como X que toma valores en el universo de discurso;
-

- U el universo de discurso que está asociado con una variable base denominada u ;
- G es una regla sintáctica (que normalmente toma forma de gramática) para generar los nombre de los valores de H ;
- M es una regla semántica para asociar significado a cada elemento de H , que será un subconjunto difuso de U .

Por ejemplo, consideremos la variable lingüística $H = velocidad$, con $U = [0, 125]$ y la variable base $u \in U$. El conjunto de términos asociados con la velocidad podría ser $H(L) = \{baja, media, alta\}$ donde cada término en $H(velocidad)$ es el nombre de un valor lingüístico de *velocidad*. El significado $M(X)$ de una etiqueta $H \in H(velocidad)$ se define como la restricción $H(u)$ sobre la variable base u impuesta según el nombre de H . Por lo tanto $M(X)$ es un conjunto difuso de U cuya función de pertenencia $H(u)$ representa la semántica del nombre H . En la figura 3.3 podemos ver una representación gráfica del ejemplo.

3.2.5. Pasos para la Aplicación del Enfoque Lingüístico Difuso

En cualquier ámbito en el que deseemos aplicar un enfoque lingüístico para la resolución de algún problema, debemos tomar dos decisiones:

- Modelo de representación: elección del conjunto de términos lingüísticos junto con su semántica y así proporcionar a una fuente de información un número reducido de términos con los que poder expresarla.
-

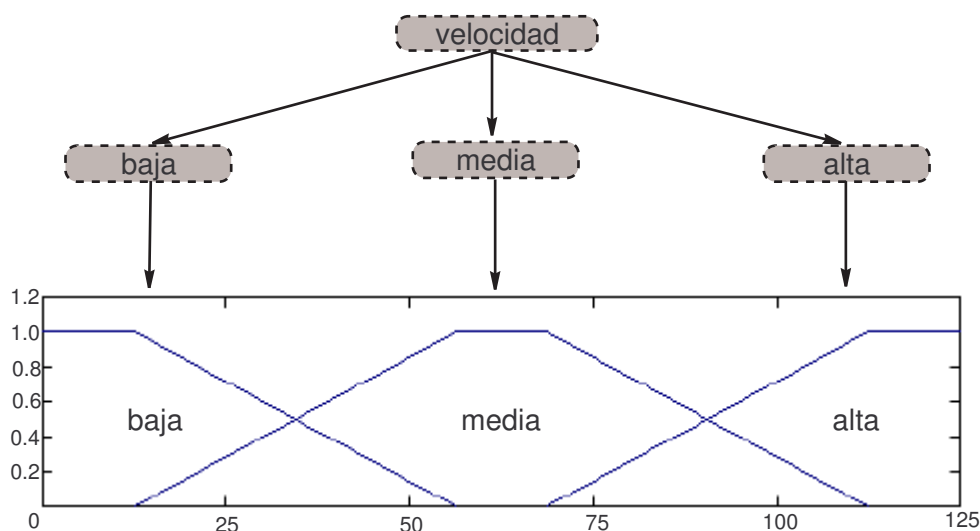


Figura 3.3: Ejemplo de una variable lingüística.

- Modelo computacional: definir el modelo computacional seleccionando los correspondientes operadores de comparación y de agregación.

Un aspecto importante que es necesario analizar con el fin de establecer la descripción de una variable lingüística es la *granularidad de la incertidumbre* [7], es decir, la cardinalidad del conjunto de términos lingüísticos usado para expresar y representar la información. La cardinalidad debe ser suficientemente baja como para no imponer una precisión excesiva en la información que se quiera expresar y suficientemente alta como para conseguir una discriminación de las valoraciones en un número limitado de grados. Habitualmente la cardinalidad usada en los modelos lingüísticos suele ser un valor impar, como 7 o 9, no superando las 11 o 13 etiquetas. El término medio representa una valoración de *aproximadamente 0.5*, y el resto de términos se sitúan simétricamente alrededor de este punto medio [7]. Estos valores clásicos de cardinalidad están basados en la línea de observación de

Miller sobre la capacidad humana [74], en la que se indica que se pueden manejar razonablemente y recordar alrededor de 7 o 9 términos.

Una vez establecida la cardinalidad del conjunto de términos lingüísticos, hay que definir dicho conjunto, es decir, cuáles van a ser las etiquetas lingüísticas y su semántica asociada.

3.3. Modelado Lingüístico Difuso Clásico

El modelado lingüístico difuso clásico adopta un *enfoque basado en una gramática libre de contexto* [7, 10, 116]. Consiste en utilizar una gramática libre de contexto G , donde el conjunto de términos pertenece al lenguaje generado por G . Una gramática generadora G , es una 4-tupla (V_N, V_T, I, P) siendo V_N el conjunto de símbolos no terminales, V_T el conjunto de símbolos terminales, I el símbolo inicial y P el conjunto de reglas de producción. La elección de estos cuatro elementos determinará la cardinalidad y forma del conjunto de términos lingüísticos. Entre los símbolos terminales y no terminales de G podemos encontrar términos primarios (por ejemplo *alto*, *medio*, *bajo*), modificadores (por ejemplo *no*, *mucho*, *muy*, *más o menos*), relaciones (por ejemplo *mayor que*, *menor que*) y conectivos (por ejemplo *y*, *o*, *pero*). Siendo I cualquier término primario y usando P , construimos el conjunto de términos lingüísticos $H = \{muy\ alto, alto, medio, \dots\}$. La semántica del conjunto de términos lingüísticos se define utilizando números difusos en el intervalo $[0,1]$, donde cada número difuso es descrito por una función de pertenencia basada en ciertos parámetros o reglas semánticas.

Con respecto a la definición de operadores de agregación de información lingüística, el modelo clásico lo que hace es extender las operaciones de la lógica tradicional para aplicarlas sobre las funciones de pertenencia. El inconveniente es que como resultado obtendremos otro conjunto difuso que no se corresponde con ninguna etiqueta del conjunto de términos originalmente considerado. Si finalmente deseamos obtener una etiqueta de dicho conjunto, es necesario realizar un proceso de aproximación lingüística consistente en encontrar una etiqueta cuyo significado sea el mismo o lo más parecido posible (de acuerdo a alguna métrica) al significado del conjunto difuso no etiquetado obtenido como resultado de alguna operación.

3.4. Modelado Lingüístico Difuso Ordinal

El modelado lingüístico difuso ordinal [23, 51, 55] es un tipo muy útil de enfoque lingüístico difuso, propuesto como una herramienta alternativa al modelado lingüístico difuso clásico que simplifica la computación con palabras eliminando la complejidad de tener que definir una gramática.

Además, el modelado lingüístico difuso clásico al trabajar con números difusos presenta el inconveniente de que no suelen coincidir con etiquetas del conjunto de términos lingüísticos, por lo que si se desea obtener una etiqueta se hace necesaria una aproximación lingüística. El modelado lingüístico difuso ordinal trabaja directamente con las etiquetas previamente definidas por lo que evita tener que recurrir a aproximaciones lingüísticas complejas.

3.4.1. Modelo de Representación en el Enfoque Lingüístico Ordinal

Un enfoque lingüístico difuso ordinal se define considerando un conjunto de etiquetas finito y totalmente ordenado $\mathcal{S} = \{s_i\}, i \in \{0, \dots, g\}$ con $s_i \geq s_j$ si $i \geq j$, y con una cardinalidad impar (la cardinalidad de \mathcal{S} es $g + 1$). La semántica del conjunto de etiquetas es establecida según la estructura ordenada del conjunto de etiquetas [9], considerando que cada etiqueta del par (s_i, s_{g-i}) es igualmente informativa. Por ejemplo, podríamos usar el siguiente conjunto de 9 etiquetas para representar la información lingüística:

$$\mathcal{S} = \{N, VL, L, M, H, VH, P\}$$

$$\begin{aligned} s_0 &= Nulo = N & s_1 &= Muy bajo = VL \\ s_2 &= Bajo = L & s_3 &= Medio = M \\ s_4 &= Alto = H & s_5 &= Muy alto = VH \\ s_6 &= Perfecto = P. \end{aligned}$$

donde $s_a < s_b$ si y sólo si $a < b$.

A continuación, tenemos que dar significado al conjunto de etiquetas lingüísticas asociando con cada término lingüístico un conjunto difuso definido en el intervalo $[0, 1]$. Para ello, podemos hacer uso de una representación trapezoidal de la función de pertenencia, o de su caso más particular, una representación triangular por medio de una 3-tupla (a, α, β) , donde recordemos que a indica el punto donde el valor de pertenencia vale 1 y α y β los límites izquierdo y derecho respectivamente. Como ejemplo, podemos considerar el anterior conjunto de etiquetas con las siguientes funciones de pertenencia (ver figura 3.4):

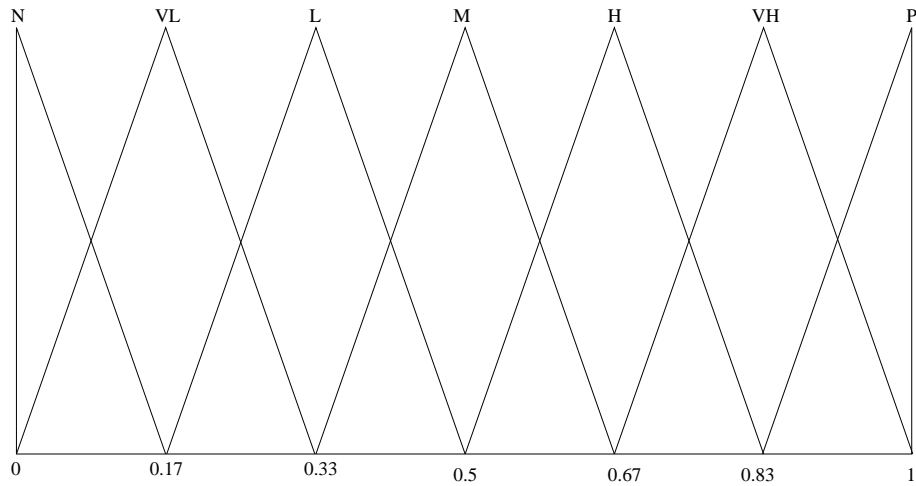


Figura 3.4: Un conjunto de 7 términos lingüísticos y su semántica.

$$\begin{aligned}
 s_0 = Nulo(N) &= (0, 0, 0.17) & s_1 = Muy\ bajo(VL) &= (0.17, 0, 0.33) \\
 s_2 = Bajo(L) &= (0.33, 0.17, 0.5) & s_3 = Medio(M) &= (0.5, 0.33, 0.67) \\
 s_4 = Alto(H) &= (0.67, 0.5, 0.83) & s_5 = Muy\ alto(VH) &= (0.83, 0.67, 1) \\
 s_6 = Perfecto(P) &= (1, 0.83, 1).
 \end{aligned}$$

3.4.2. Modelo Computacional en el Enfoque Lingüístico Ordinal

En cualquier enfoque lingüístico necesitamos operadores para el manejo de la información lingüística. Una ventaja del enfoque lingüístico difuso ordinal es la simplicidad y agilidad de su modelo computacional. Está basado en el cálculo simbólico [51, 55] y actúa operando directamente sobre las etiquetas, teniendo en cuenta el orden de las valoraciones lingüísticas en la estructura ordenada de las etiquetas. Habitualmente, el modelo lingüístico difuso ordinal para la computación con palabras se define estableciendo:

1. un operador de negación,
2. operadores de comparación basados en la estructura ordenada de los términos lingüísticos, y
3. operadores apropiados para la agregación de información lingüística difusa ordinal.

En la mayoría de los enfoques lingüísticos difusos ordinales, a partir de la semántica asociada a los términos lingüísticos el operador de negación se define como:

$$NEG(s_i) = s_j / j = g - i;$$

También podemos definir dos operadores de comparación de términos lingüísticos:

1. *Operador de maximización*: $MAX(s_i, s_j) = s_i$ si $s_i \geq s_j$.
2. *Operador de minimización*: $MIN(s_i, s_j) = s_i$ si $s_i \leq s_j$.

A partir de estos operadores es posible definir operadores automáticos y simbólicos de agregación de información lingüística, como por ejemplo el operador de agregación de información lingüística no ponderada LOWA (*Linguistic Ordered Weighted Averaging*) [55] y el operador de información lingüística ponderada LWA (*Linguistic Weighted Averaging*) [51], que están basados en el operador OWA (*Ordered Weighted Averaging*) definido en [109]. El OWA es un operador de agregación de información numérica que tiene en cuenta el orden de las valoraciones que van a ser agregadas.

Definición 3.7. Operador OWA. Sea $A = \{a_1, \dots, a_n\}$ con $a_i \in [0, 1]$ el conjunto de valoraciones que se quieren agregar y $W = (w_1, \dots, w_n)$ su vector de pesos asociado, tal que (i) $w_i \in [0, 1]$ y (ii) $\sum_{i=1}^n w_i = 1$. El operador OWA, f , se define como:

$$f(a_1, \dots, a_n) = \sum_{j=1}^n w_j \cdot b_j$$

donde b_j es el j -ésimo mayor valor del conjunto A . Por tanto, a partir de los elementos de A podemos obtener un conjunto B ordenando dichos elementos en orden decreciente, es decir,

$$B = \{b_1, \dots, b_n\} / b_i \geq b_j \text{ si } i < j$$

y definir el operador OWA de la siguiente forma:

$$f(a_1, \dots, a_n) = W \cdot B$$

Ejemplo 3.1. Aplicación del operador OWA.

Supongamos que tenemos el siguiente conjunto de valoraciones $A = \{0.6, 1.0, 0.3, 0.5\}$ con el siguiente vector de pesos $W = (0.2, 0.3, 0.1, 0.4)$.

En este caso, el vector ordenado B es

$$B = \begin{bmatrix} 1.0 \\ 0.6 \\ 0.5 \\ 0.3 \end{bmatrix},$$

por lo que:

$$\begin{aligned}
 f(0.6, 1.0, 0.3, 0.5) &= W \cdot B = [0.2, 0.3, 0.1, 0.4] \begin{bmatrix} 1.0 \\ 0.6 \\ 0.5 \\ 0.3 \end{bmatrix} \\
 &= (0.2 \cdot 1.0) + (0.3 \cdot 0.6) + (0.1 \cdot 0.5) + (0.4 \cdot 0.3) = 0.55
 \end{aligned}$$

Definición 3.8. Operador LOWA. Sea $A = \{a_1, \dots, a_m\}$ un conjunto de etiquetas a agregar, $a_i \in \mathcal{S}$, entonces el operador LOWA, ϕ , se define como:

$$\begin{aligned}
 \phi(a_1, \dots, a_m) &= W \cdot B = \mathcal{C}^m\{w_k, b_k, k = 1, \dots, m\} = \\
 &= w_1 \odot b_1 \oplus (1 - w_1) \odot \mathcal{C}^{m-1}\{\beta_h, b_h, h = 2, \dots, m\}
 \end{aligned}$$

donde $W = [w_1, \dots, w_m]$, es un vector de ponderación, tal que,

1. $w_i \in [0, 1]$,
2. $\sum_{i=1}^n w_i = 1$,

y $\beta_h = w_h / \sum_2^m w_k, h = 2, \dots, m$, siendo $B = (b_1, \dots, b_m)$ un vector asociado a A , tal que,

$$B = \sigma(A) = (a_{\sigma(1)}, \dots, a_{\sigma(n)})$$

donde, $a_{\sigma(j)} \leq a_{\sigma(i)} \forall i \leq j$, siendo σ una permutación definida sobre el conjunto de etiquetas A . \mathcal{C}^m es el operador de combinación convexa de m etiquetas [26], de modo que si $m = 2$, entonces se define como

$$\mathcal{C}^2\{w_i, b_i, i = 1, 2\} = w_1 \odot s_j \oplus (1 - w_1) \odot s_i = s_k, \quad s_j, s_i \in \mathcal{S}, (j \geq i),$$

$$k = \text{MIN}\{g, i + \text{round}(w_1 \cdot (j - i))\},$$

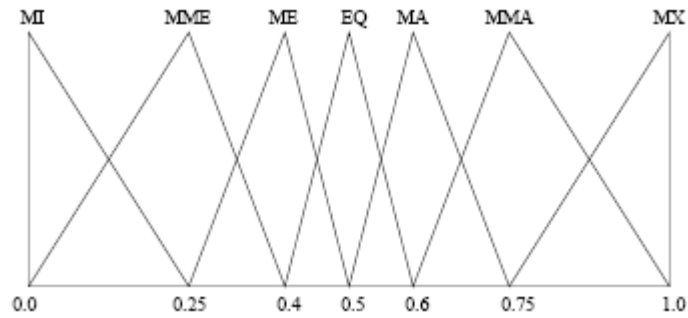


Figura 3.5: Semántica asociada al conjunto de términos lingüísticos.

”round” simboliza el operador de redondeo usual, y $b_1 = s_j$, $b_2 = s_i$. Por otro lado, si $w_j = 1$ y $w_i = 0$ con $i \neq j \forall i$, entonces el operador de combinación se define como:

$$\mathcal{C}^m\{w_i, b_i, i = 1, \dots, m\} = b_j.$$

Ejemplo 3.2. Aplicación del operador LOWA.

Supongamos $m = 2$, $W = [0.4, 0.6]$ y que usamos el siguiente conjunto de siete etiquetas:

$$S = \{s_0 = MI, s_1 = MME, s_2 = ME, s_3 = EQ, s_4 = MA, s_5 = MMA, s_6 = MX\},$$

donde

$$\begin{aligned} MI &= \text{Mínimo} & MME &= \text{Mucho_Menor} & ME &= \text{Menor} \\ EQ &= \text{Equivalente} & MA &= \text{Mayor} & MMA &= \text{Mucho_Mayor} \\ MX &= \text{Máximo} \end{aligned}$$

con los siguiente valores de representación (ver figura 3.5):

		1 - w1 = 0.6			
		MME	MMA	MX	MME
w1 = 0.4	MX	EQ	MMA	MX	EQ
	MI	MME	EQ	MA	MME
	ME	MME	EQ	MA	MME
	EQ	ME	MA	MMA	ME

Tabla 3.2: Tabla del LOWA con $m = 2$.

$$\begin{aligned}
 MI &= (0, 0, 0, 0.25) & MME &= (0.25, 0.25, 0, 0.4) & ME &= (0.4, 0.4, 0.25, 0.5) \\
 EQ &= (0.5, 0.5, 0.4, 0.6) & MA &= (0.6, 0.6, 0.5, 0.75) & MMA &= (0.75, 0.75, 0.6, 1) \\
 MX &= (1, 1, 0.75, 1)
 \end{aligned}$$

Los resultados se muestran en la tabla 3.2, donde por ejemplo:

$$\begin{aligned}
 k_{11} &= \text{MIN}\{6, 1 + \text{round}(0.4 * (6 - 1))\} = 3 \Rightarrow l_{k_{11}} = \text{EQ} \\
 k_{21} &= \text{MIN}\{6, 0 + \text{round}(0.6 * (1 - 0))\} = 1 \Rightarrow l_{k_{21}} = \text{MME}
 \end{aligned}$$

Para concluir, indicar que existen otras opciones de modelado lingüístico difuso ordinal, como generar la semántica de las etiquetas lingüísticas utilizando funciones de negación que inducen una semántica para cada etiqueta [105], estando éstas definidas como intervalos en $[0,1]$.

3.5. Modelado Lingüístico Difuso Basado en 2-tuplas

El *modelado lingüístico difuso basado en 2-tuplas* [56, 58] es un tipo de modelado lingüístico difuso que nos permite reducir la pérdida de información que habitualmente se produce en el modelado lingüístico difuso ordinal. Esta pérdida de información, que provoca una falta de precisión en los resultados, se debe al propio modelo de representación puesto que opera con valores discretos sobre un universo de discurso continuo. La principal ventaja del modelo computacional lingüístico basado en 2-tuplas, es que permite realizar procesos de cálculo con palabras de forma más sencilla y por tanto, sin pérdida de información puesto que utiliza un modelo continuo de representación de la información. Para definirlo, tenemos que establecer el modelo de representación y el modelo computacional de las 2-tuplas para representar y agregar la información lingüística respectivamente.

3.5.1. Modelo de Representación Lingüística Basada en 2-tuplas

Consideremos que $\mathcal{S} = \{s_0, \dots, s_g\}$ es un conjunto de términos lingüísticos con cardinalidad impar, donde el término intermedio representa una valoración de aproximadamente 0.5 y con el resto de términos del conjunto distribuidos simétricamente alrededor de ese punto intermedio. Asumimos que la semántica asociada con cada una de las etiquetas viene dada por medio de funciones de pertenencia triangulares, representadas por 3-tuplas (a, α, β) y consideramos todos los términos distribuidos sobre una escala sobre la que hay establecida una relación de orden total, es decir, $s_i \leq s_j \iff i \leq j$. En este contexto lingüístico difuso, si

mediante un método simbólico de agregación de información lingüística [51, 55] obtenemos un valor $\beta \in [0, g]$, y $\beta \notin \{0, \dots, g\}$, podemos usar una función de aproximación para expresar el resultado obtenido como un valor de \mathcal{S} .

Definición 3.9. [56] Sea β el resultado de una agregación de los índices de un conjunto de etiquetas valoradas sobre un conjunto de términos lingüísticos \mathcal{S} , es decir, el resultado de una operación de agregación simbólica, $\beta \in [0, g]$. Dados $i = \text{round}(\beta)$ y $\alpha = \beta - i$ dos valores, tales que, $i \in [0, g]$ y $\alpha \in [-0.5, 0.5)$ entonces α es lo que denominamos **Traslación Simbólica**, que expresa la diferencia de información entre la información expresada por β y la etiqueta lingüística s_i más cercana a \mathcal{S} .

El enfoque lingüístico difuso basado en 2-tuplas es desarrollado a partir del concepto de translación simbólica, representando la información lingüística por medio de 2-tuplas (s_i, α_i) , $s_i \in \mathcal{S}$ y $\alpha_i \in [-0.5, 0.5)$:

- s_i representa la etiqueta lingüística, y
- α_i es un valor numérico que expresa la translación de β al índice de la etiqueta más cercana, i , en el conjunto de términos lingüísticos ($s_i \in \mathcal{S}$).

Este modelo define un conjunto de funciones de transformación entre valores numéricos y 2-tuplas.

Definición 3.10. Sea $s_i \in \mathcal{S}$ un término lingüístico, su representación mediante

una 2-tupla se obtiene mediante la función θ :

$$\theta : [0, g] \longrightarrow \mathcal{S} \times [-0.5, 0.5)$$

$$\theta(s_i) = (s_i, 0) / s_i \in \mathcal{S}$$

Definición 3.11. [56] Siendo $\mathcal{S} = \{s_0, \dots, s_g\}$ un conjunto de términos lingüísticos y $\beta \in [0, g]$ un valor que representa el resultado de una operación de agregación simbólica, la 2-tupla que expresa la información equivalente a β se obtiene mediante la siguiente función:

$$\Delta : [0, g] \longrightarrow \mathcal{S} \times [-0.5, 0.5)$$

$$\Delta(\beta) = (s_i, \alpha), \text{ with } \begin{cases} s_i & i = \text{round}(\beta) \\ \alpha = \beta - i & \alpha \in [-0.5, 0.5) \end{cases}$$

donde $\text{round}(\cdot)$ es el típico operador de redondeo, s_i es la etiqueta cuyo índice es el más cercano a β y α es el valor de la traslación simbólica.

Ejemplo 3.3. Representación 2-tupla.

Supongamos que trabajamos con el siguiente conjunto de términos lingüísticos $\mathcal{S} = \{s_0, s_1, s_2, s_3, s_4, s_5, s_6\}$ y que como resultado de una operación de agregación simbólica se obtiene el valor $\beta = 2.8$. La representación de este valor mediante una 2-tupla lingüística, sería:

$$\Delta(\beta) = (s_3, -0.2)$$

Definición 3.12. [56] Sea $\mathcal{S} = \{s_0, \dots, s_g\}$ un conjunto de términos lingüísticos y

(s_i, α) una 2-tupla. Se define la función Δ^{-1} , tal que aplicada sobre una 2-tupla (s_i, α) devuelve su valor numérico $\beta \in [0, g]$.

$$\Delta^{-1} : \mathcal{S} \times [-0.5, 0.5) \longrightarrow [0, g]$$

$$\Delta^{-1}(s_i, \alpha) = i + \alpha = \beta$$

3.5.2. Modelo Computacional Lingüístico de las 2-tuplas

A continuación presentamos el modelo computacional que nos permite operar sobre la representación lingüística de las 2-tuplas, basándonos en los operadores de comparación, negación y agregación de 2-tuplas:

1. *Operador de comparación de 2-tuplas.* La comparación de información lingüística representada por medio de 2-tuplas se realiza de acuerdo a un orden lexicográfico normal y corriente. Consideremos dos 2-tuplas (s_k, α_1) y (s_l, α_2) que representan cálculos de información:

- si $k < l$ entonces (s_k, α_1) es menor que (s_l, α_2) .
- si $k = l$ entonces
 - a) si $\alpha_1 = \alpha_2$ entonces (s_k, α_1) y (s_l, α_2) representan la misma información,
 - b) si $\alpha_1 < \alpha_2$ entonces (s_k, α_1) es menor que (s_l, α_2) ,
 - c) si $\alpha_1 > \alpha_2$ entonces (s_k, α_1) es mayor que (s_l, α_2) .

2. *Operador de negación de 2-tuplas.* El operador de negación sobre una 2-tupla se define como:

$$Neg((s_i, \alpha)) = \Delta(g - (\Delta^{-1}(s_i, \alpha))).$$

siendo $g + 1$ la cardinalidad del conjunto de etiquetas \mathcal{S} .

3. *Operador de agregación de 2-tuplas.* La agregación de información consiste en obtener un valor que resuma un conjunto de valores, por lo que el resultado de la agregación de un conjunto de 2-tuplas debe ser una 2-tupla. A lo largo de la literatura podemos encontrar numerosos operadores de agregación que nos permiten combinar la información de acuerdo a distintos criterios. Cualquiera de estos operadores ya existentes puede ser fácilmente extendido para trabajar con 2-tuplas, usando funciones Δ y Δ^{-1} que transforman valores numéricos en 2-tuplas y viceversa sin pérdida de información. Algunos ejemplos de estos operadores son los siguientes:

Definición 3.13. Media aritmética. Siendo $x = \{(r_1, \alpha_1), \dots, (r_n, \alpha_n)\}$ un conjunto de 2-tuplas lingüísticas, la 2-tupla que simboliza la media aritmética, \bar{x}^e , se calcula de la siguiente forma:

$$\bar{x}^e[(r_1, \alpha_1), \dots, (r_n, \alpha_n)] = \Delta\left(\sum_{i=1}^n \frac{1}{n} \Delta^{-1}(r_i, \alpha_i)\right) = \Delta\left(\frac{1}{n} \sum_{i=1}^n \beta_i\right).$$

Definición 3.14. Operador de media ponderada. Siendo $x = \{(r_1, \alpha_1), \dots, (r_n, \alpha_n)\}$ un conjunto de 2-tuplas lingüísticas y $W = \{w_1, \dots, w_n\}$ un vector numérico con sus pesos asociados, la 2-tupla que simboliza la media ponderada, \bar{x}^w , es:

$$\bar{x}^w[(r_1, \alpha_1), \dots, (r_n, \alpha_n)] = \Delta\left(\frac{\sum_{i=1}^n \Delta^{-1}(r_i, \alpha_i) \cdot w_i}{\sum_{i=1}^n w_i}\right) = \Delta\left(\frac{\sum_{i=1}^n \beta_i \cdot w_i}{\sum_{i=1}^n w_i}\right).$$

Definición 3.15. Operador de media ponderada lingüística. Siendo $x = \{(r_1, \alpha_1), \dots, (r_n, \alpha_n)\}$ un conjunto de 2-tuplas y $W = \{(w_1, \alpha_1^w), \dots, (w_n, \alpha_n^w)\}$ sus pesos asociados representados mediante 2-tuplas lingüísticas, la 2-tupla que representa la media ponderada lingüística, \bar{x}_l^w , se calcula de la siguiente manera:

$$\bar{x}_l^w[(r_1, \alpha_1), (w_1, \alpha_1^w) \dots (r_n, \alpha_n), (w_n, \alpha_n^w)] = \Delta\left(\frac{\sum_{i=1}^n \beta_i \cdot \beta_{W_i}}{\sum_{i=1}^n \beta_{W_i}}\right),$$

con $\beta_i = \Delta^{-1}(r_i, \alpha_i)$ y $\beta_{W_i} = \Delta^{-1}(w_i, \alpha_i^w)$.

3.6. Modelado Lingüístico Difuso Multi-granular

Con anterioridad hemos comentado que en cualquier enfoque lingüístico difuso, uno de los parámetros más importantes que hay que determinar es la *granularidad de la incertidumbre*, es decir, la cardinalidad del conjunto de términos lingüísticos \mathcal{S} usado para expresar la información lingüística. En función del grado de incertidumbre que un experto encargado de cualificar un fenómeno tenga sobre el mismo, el conjunto de términos lingüísticos elegido para proporcionar ese conocimiento tendrá más o menos términos. Por lo tanto, cuando distintos expertos tienen diferentes grados de incertidumbre sobre el fenómeno, es conveniente que cada uno trabaje con conjuntos de términos lingüísticos de diferente granularidad de incertidumbre (es decir, trabajar con información lingüística multi-granular) [45, 52, 57]. El uso de diferentes conjuntos de etiquetas es también necesario cuando un experto tiene que valorar conceptos diferentes, como por ejemplo ocurre en los problemas de recuperación de información, al evaluar la

importancia de los términos de la consulta y la relevancia de los documentos recuperados [40], que son conceptos distintos. En ese tipo de situaciones necesitamos herramientas que nos permitan gestionar la información lingüística multi-granular, es decir, necesitamos definir un *modelado lingüístico difuso multi-granular*. Para ello vamos a seguir el modelo propuesto en [57] que hace uso del concepto de jerarquías lingüísticas.

Una ***Jerarquía Lingüística*** es un conjunto de niveles, donde cada nivel es un conjunto de términos lingüísticos con una granularidad diferente del resto de niveles de la jerarquía [16]. A cada uno de los niveles de una jerarquía lingüística los vamos a denotar como $l(t, n(t))$, siendo t un número que indica el nivel de la jerarquía y $n(t)$ la granularidad del conjunto de términos lingüísticos del nivel t .

Normalmente, las jerarquías lingüísticas trabajan con términos lingüísticos cuyas funciones de pertenencia son de forma triangular, simétricas y uniformemente distribuidas en el intervalo $[0,1]$. Además, los conjuntos de términos lingüísticos tienen una granularidad impar, con la etiqueta central indicando un valor de indiferencia.

Los niveles de una jerarquía lingüística están ordenados en función de su granularidad, es decir, que para dos niveles consecutivos t y $t + 1$, $n(t + 1) > n(t)$. Por lo tanto, cada nivel $t + 1$ proporciona un refinamiento lingüístico con respecto al nivel anterior t .

Vamos a definir una jerarquía lingüística, LH , como la unión de todos los niveles

t que la conforman:

$$LH = \bigcup_t l(t, n(t))$$

Para la construcción de LH debemos tener en mente que el orden jerárquico nos viene dado por el incremento de granularidad de los conjuntos de términos lingüísticos de cada nivel.

Partiendo de que $\mathcal{S}^{n(t)} = \{s_0^{n(t)}, \dots, s_{n(t)-1}^{n(t)}\}$ sea el conjunto de términos lingüísticos definido para el nivel t con $n(t)$ términos, la construcción de una jerarquía lingüística debe satisfacer las siguientes reglas básicas [57]:

1. Preservar todos los puntos modales previos de las funciones de pertenencia de cada uno de los términos lingüísticos de cada nivel con respecto a los del nivel siguiente.
2. Hacer que las transacciones entre dos niveles consecutivos sean suaves. El propósito es construir un nuevo conjunto de términos lingüísticos, $\mathcal{S}^{n(t+1)}$, de forma que añadiremos un nuevo término lingüístico entre cada pareja de términos pertenecientes al conjunto de términos del nivel anterior t . Para realizar esta inserción de nuevos términos, reduciremos el soporte de las etiquetas lingüísticas para dejar hueco entre ellas para la nueva etiqueta.

De forma genérica, podemos establecer que el conjunto de términos lingüísticos de nivel $t + 1$, $\mathcal{S}^{n(t+1)}$, puede ser obtenido a partir del nivel anterior t , $\mathcal{S}^{n(t)}$, de la siguiente manera:

$$l(t, n(t)) \rightarrow l(t + 1, 2 \cdot n(t) - 1)$$

En la tabla 3.3 mostramos la granularidad necesaria en cada conjunto de términos lingüísticos de nivel t , dependiendo del valor $n(t)$ definido en el primer nivel (para valores de 3 y 7 respectivamente).

	Nivel 1	Nivel 2	Nivel 3
$l(t, n(t))$	$l(1, 3)$	$l(2, 5)$	$l(3, 9)$
$l(t, n(t))$	$l(1, 7)$	$l(2, 13)$	

Tabla 3.3: Granularidad en distintos niveles de una jerarquía.

En la figura 3.6 se muestra un ejemplo gráfico de jerarquías lingüísticas. Se representa una jerarquía compuesta de 3 niveles, de 3, 5 y 9 etiquetas cada uno de ellos.

En [57] se demostró que las jerarquías lingüísticas son útiles para representar información lingüística multi-granular y por tanto permiten trabajar con información lingüística sin pérdida de información. Para conseguirlo, fue definida una familia de funciones de transformación entre etiquetas de diferentes niveles.

Definición 3.16. Sea $LH = \bigcup_t l(t, n(t))$ una jerarquía lingüística cuyos conjuntos de términos lingüísticos son denotados como $\mathcal{S}^{n(t)} = \{s_0^{n(t)}, \dots, s_{n(t)-1}^{n(t)}\}$. La **función de transformación** de una etiqueta lingüística (representada mediante una 2-tupla) de un nivel t a una etiqueta de un nivel consecutivo $t + c$, con $c \in -1, 1$, se define como:

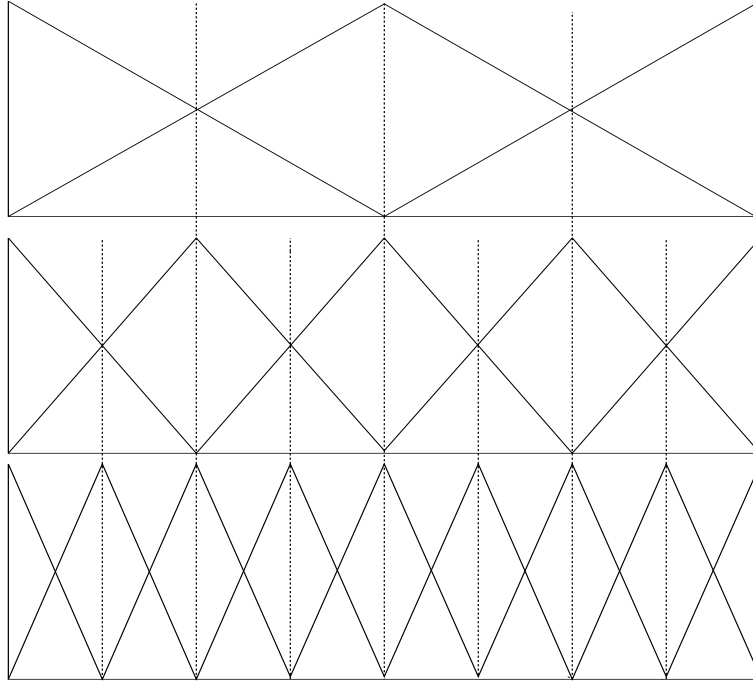


Figura 3.6: Jerarquía lingüística de 3, 5 y 9 etiquetas.

$$TF_{t+c}^t : l(t, n(t)) \longrightarrow l(t+c, n(t+c))$$

$$TF_{t+c}^t(s_i^{n(t)}, \alpha^{n(t)}) = \Delta\left(\frac{\Delta^{-1}(s_i^{n(t)}, \alpha^{n(t)}) \cdot (n(t+c) - 1)}{n(t) - 1}\right)$$

Esta función de transformación fue generalizada para transformar términos lingüísticos entre cualquier nivel dentro de la jerarquía lingüística.

Definición 3.17. Sea $LH = \bigcup_t l(t, n(t))$ una jerarquía lingüística cuyos conjuntos de términos lingüísticos son denotados como $\mathcal{S}^{n(t)} = \{s_0^{n(t)}, \dots, s_{n(t)-1}^{n(t)}\}$. La **función de transformación recursiva** entre una etiqueta lingüística (representada mediante una 2-tupla) perteneciente a un nivel t y una etiqueta perteneciente al nivel $t' = t + a$, con $a \in \mathbb{Z}$, se define como:

$$TF_{t'}^t : l(t, n(t)) \longrightarrow l(t', n(t'))$$

Si $|a| > 1$ entonces

$$TF_{t'}^t(s_i^{n(t)}, \alpha^{n(t)}) = TF_{t'}^{t+\frac{t-t'}{|t-t'|}}(TF_{t+\frac{t-t'}{|t-t'|}}^t(s_i^{n(t)}, \alpha^{n(t)}))$$

Si $|a| = 1$ entonces

$$TF_{t'}^t(s_i^{n(t)}, \alpha^{n(t)}) = TF_{t+\frac{t-t'}{|t-t'|}}^t(s_i^{n(t)}, \alpha^{n(t)})$$

Esta función de transformación recursiva, puede ser definida fácilmente de una forma no recursiva de la siguiente manera:

$$TF_{t'}^t : l(t, n(t)) \longrightarrow l(t', n(t'))$$

$$TF_{t'}^t(s_i^{n(t)}, \alpha^{n(t)}) = \Delta\left(\frac{\Delta^{-1}(s_i^{n(t)}, \alpha^{n(t)}) \cdot (n(t') - 1)}{n(t) - 1}\right)$$

Proposición 1 [57]. Esta familia de funciones de transformación entre etiquetas lingüísticas de distintos niveles de una jerarquía lingüística es biyectiva:

$$TF_t^{t'}(TF_{t'}^t(s_i^{n(t)}, \alpha^{n(t)})) = (s_i^{n(t)}, \alpha^{n(t)})$$

3.7. Modelado Lingüístico Difuso no Balanceado

Según hemos estado viendo, ante cualquier problema que hace uso de información lingüística, el primer objetivo que hay que satisfacer es la elección de los términos

lingüísticos con sus correspondientes semánticas, para así establecer el conjunto de etiquetas que se va a usar. A lo largo de la literatura podemos encontrar dos posibilidades distintas para la elección de los términos lingüísticos y sus semánticas:

- Por un lado, podemos asumir que todos los términos del conjunto de etiquetas son igualmente informativos, es decir, están distribuidos simétricamente tal y como sucede en los modelados lingüísticos difusos que hemos estado viendo hasta ahora.
- Por otro lado, podemos asumir que no todos los términos del conjunto de etiquetas son igualmente informativos, es decir, las etiquetas no están distribuidas simétricamente. En este caso, necesitamos un *enfoque lingüístico difuso no balanceado* [53, 54] para gestionar los conjuntos de términos lingüísticos con distintos niveles de discriminación a ambos lados del término medio. Por ejemplo supongamos el siguiente conjunto de etiquetas, distribuidas tal y como se muestra en la figura 3.7:

$$\mathcal{S} = \{N, L, M, H, QH, VH, T\}$$

$N = Nulo$

$L = Bajo$

$M = Medio$

$H = Alto$

$QH = Bastante alto$

$VH = Muy alto$

$T = Total$

Como se puede ver en [53], en sistemas de RI parece más apropiado el uso de conjuntos de términos lingüísticos no balanceados que el uso de conjuntos

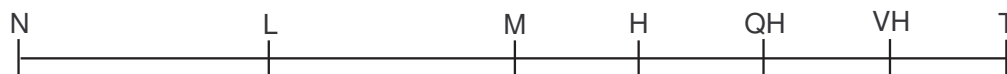


Figura 3.7: Conjunto de términos lingüísticos de 7 etiquetas no balanceado.

de términos lingüísticos simétricos, tanto para expresar los pesos de importancia en las consultas como para representar los grados de relevancia de los documentos.

Para gestionar los conjuntos de términos lingüísticos no balanceados se puede hacer uso del modelado lingüístico difuso basado en 2-tuplas [53]. Básicamente, el método consiste en representar los términos lingüísticos no balanceados usando distintos niveles de una jerarquía lingüística LH , llevando a cabo todas las operaciones mediante el uso del modelo computacional definido para la representación de 2-tuplas. El método consiste, pues, en la realización de los siguientes pasos [53]:

1. Representar el conjunto de términos lingüísticos no balanceados \mathcal{S} mediante una jerarquía lingüística, LH .
 - a) Seleccionar un nivel t^- con una granularidad apropiada para representar, usando el modelo de representación de las 2-tuplas, el subconjunto de términos lingüísticos de \mathcal{S} que hay a la izquierda del término medio.
 - b) Seleccionar un nivel t^+ con una granularidad apropiada para representar, usando el modelo de representación de las 2-tuplas, el subconjunto de términos lingüísticos de \mathcal{S} que hay a la derecha del término medio.
 2. Definir un modelo computacional para trabajar con la información lingüística no balanceada.
-

- a) Seleccionar el nivel $t' \in \{t^-, t^+\}$, de tal forma que $n(t') = \max\{n(t^-), n(t^+)\}$, es decir, el de mayor granularidad.
- b) Definir la operación de comparación entre dos 2-tuplas $(s_k^{n(t)}, \alpha_1)$, $t \in \{t^-, t^+\}$ y $(s_l^{n(t)}, \alpha_2)$, $t \in \{t^-, t^+\}$, cada una representando un cálculo de información no balanceada. Su expresión es similar a la comparación de dos 2-tuplas, pero actuando sobre los valores $TF_{t'}^t(s_k^{n(t)}, \alpha_1)$ y $TF_{t'}^t(s_l^{n(t)}, \alpha_2)$. Una vez definida esta operación de comparación de dos 2-tuplas, fácilmente podemos definir otros operadores como *Max* o *Min*.
- c) Definir el operador de negación de información lingüística no balanceada. Siendo $(s_k^{n(t)}, \alpha)$, $t \in \{t^-, t^+\}$ una 2-tupla que representa información lingüística no balanceada, su negación se define como:

$$\mathcal{NEG}(s_k^{n(t)}, \alpha) = Neg(TF_{t''}^t(s_k^{n(t)}, \alpha)), \quad t \neq t'', \quad t'' \in \{t^-, t^+\}.$$

- d) Definir operadores de agregación de información lingüística no balanceada. Para ello se usan los procesos de agregación definidos en el modelo computacional de las 2-tuplas, pero actuando sobre valores lingüísticos no balanceados previamente transformados mediante la función de transformación $TF_{t'}^t$. Entonces, una vez que se obtiene un resultado, éste es transformado al correspondiente nivel t por medio de $TF_t^{t'}$, para expresar el resultado obtenido en el conjunto de términos lingüísticos no balanceado.

Para ilustrar gráficamente este proceso, a partir del conjunto de términos lingüísticos mostrado en la figura 3.7 y la jerarquía lingüística mostrada en la figura 3.6,

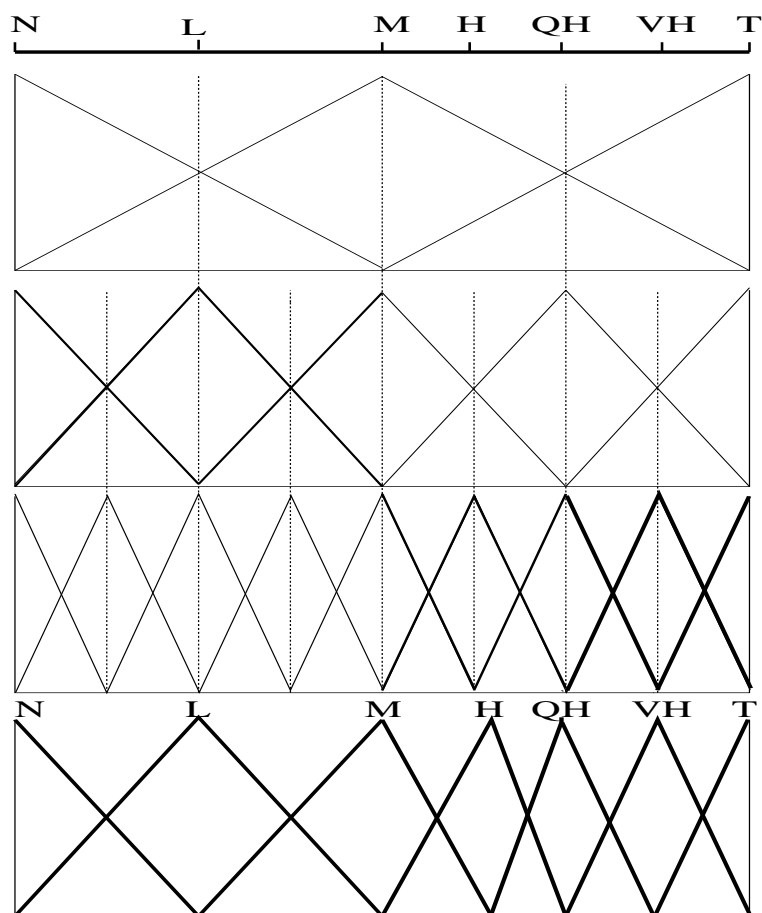


Figura 3.8: Representación de un conjunto de términos lingüísticos no balanceado.

en la figura 3.8 mostramos cómo seleccionar los diferentes niveles para representar el conjunto de términos lingüísticos no balanceado.

Capítulo 4

Un Sistema de Acceso a la Información en la Web Basado en Información Lingüística Multi-granular y en Técnicas de Filtrado

En este capítulo presentamos un modelo de sistema de acceso a información a la Web, en el que para mejorar los procesos de acceso a la información, combinamos las técnicas que hemos estado estudiando en los capítulos previos, es decir, ha sido diseñado basándonos en técnicas de filtrado de información, tanto basadas en contenidos como colaborativas, y adoptando un tipo particular de modelado lingüístico difuso, el denominado modelado lingüístico difuso multi-granular, que nos permite disponer de distintos conjuntos de etiquetas para valorar la información.

4.1. Introducción

El acceso a la información en la Web es un aspecto muy importante, ampliamente

estudiado y debatido. Uno de los principales problemas de Internet es el crecimiento de la cantidad de información a la que los usuarios pueden acceder. El incremento exponencial de sitios y documentos Web, está provocando que los usuarios de Internet no sean capaces de encontrar la información que buscan de una forma rápida y sencilla. Por ello, los usuarios necesitan sistemas que les ayuden con la gran cantidad de información disponible en la Web [12, 60, 67]. Ejemplos de dichos sistemas son los motores de búsqueda, meta-buscadores, sistemas multi-agente y sistemas de filtrado de información [4, 62].

Un sistema multi-agente es aquel en el que cierto número de agentes cooperan e interactúan unos con otros en un entorno distribuido. En la Web, la actividad de un sistema multi-agente consiste en asistir a los usuarios de Internet en sus procesos de acceso a la información por medio de agentes inteligentes distribuidos, para encontrar la información que mejor se ajusta a sus necesidades de información. Los sistemas multi-agente han sido ampliamente usados en aplicaciones Web [20, 68, 79, 97]. Un aspecto básico en la actividad de un sistema multi-agente es una comunicación eficiente entre agentes. El principal obstáculo para esta comunicación es la gran variedad de representaciones y evaluaciones de la información en Internet, y el problema es más acusado cuando los usuarios forman parte del proceso. Esto revela la necesidad de una mayor flexibilidad en la comunicación entre agentes y entre agentes y usuarios [24, 110, 111, 112]. Para solucionar este problema, se ha aplicado satisfactoriamente el modelado lingüístico difuso [55, 56, 116] en el desarrollo de diferentes modelos de sistemas multi-agente distribuidos [24, 25]. En estos modelos, los procesos de comunicación son mejorados representando la información por medio de etiquetas lingüísticas.

Para mejorar el rendimiento de estos modelos multi-agente, en [41] se propone un nuevo modelo multi-agente lingüístico difuso que incorpora en su actividad herramientas de filtrado de información [34, 91]. Usando estas técnicas de filtrado conseguimos filtrar la gran cantidad de información que reciben los usuarios de Internet, para que únicamente reciban aquella información que sea relevante para ellos.

En una sesión mantenida sobre el modelo multi-agente propuesto en [41], un usuario proporciona sus necesidades de información por medio de una consulta lingüística multi-ponderada y un tópico de interés. Entonces, en una primera fase el sistema desarrolla la recuperación documental usando la consulta del usuario, en una segunda fase desarrolla el filtrado documental usando el tópico de interés del usuario y por último, en una tercera fase, recibe la realimentación del usuario, es decir, recomendaciones del usuario sobre los documentos accedidos. Este modelo presenta dos problemas:

1. En el modelado lingüístico difuso: el inconveniente es que tanto las consultas de los usuarios como los grados de relevancia de los documentos recuperados son valorados usando el mismo conjunto de etiquetas, con la misma semántica. Sin embargo, ambos conceptos son diferentes y tienen una interpretación diferente, por lo que parece razonable y necesario valorarlos con distintos conjuntos de etiquetas lingüísticas, es decir, usando valoraciones lingüísticas multi-granulares [40, 57].
 2. En el filtrado de información: el filtrado está basado en perfiles de usuario colaborativos, pero estáticos. El perfil colaborativo está representado por
-

el tópico de interés expresado inicialmente por el usuario. Entonces, para generar recomendaciones usadas para filtrar los documentos recuperados, el sistema agrega todas las recomendaciones individuales existentes sobre los documentos recuperados almacenadas para el tópico de interés correspondiente. Por tanto, para cualquier consulta en el mismo tópico de interés, se incluyen en el perfil colaborativo del usuario todos los usuarios que en búsquedas previas expresaron valoraciones sobre los documentos recuperados. El sistema no usa la realimentación de los usuarios para actualizar sus perfiles, es decir, que carece de un módulo de aprendizaje y no es adaptativo.

En este capítulo presentamos un nuevo modelo de sistema multi-agente lingüístico difuso para acceder y recuperar información en la Web, solucionando los defectos encontrados en [41]. Para ello, basándonos en trabajos previos, incorporamos el uso del modelado lingüístico difuso multi-granular y de técnicas de filtrado con perfiles dinámicos. De esta forma, diseñamos un modelo multi-agente que representa la información lingüística involucrada en el proceso de recuperación de una forma más realista, y mejora los resultados de los procesos de recuperación usando técnicas de filtrado basadas en perfiles de usuario dinámicos. La comunicación entre agentes de diferentes niveles y entre agentes y usuarios, es llevada a cabo usando información lingüística multi-granular, es decir, los diferentes tipos de información que participan en la actividad del sistema multi-agente (pesos de la consulta, grados de satisfacción de los usuarios, grados de relevancia, recomendaciones) son valorados con diferentes grados de incertidumbre, usando varios conjuntos de etiquetas con diferente granularidad de incertidumbre. Como en [41], usamos la representación lingüística difusa basada en 2-tuplas [56] para modelar la información lingüística.

Para procesar la información lingüística multi-granular, proponemos un método basado en contextos lingüísticos jerárquicos [57] como representación básica de la información lingüística multi-granular.

Por otra parte, el modelo multi-agente incorpora técnicas de filtrado colaborativo para crear perfiles de usuario dinámicos, establecer comunidades de usuarios y proporcionar recomendaciones para filtrar los documentos recuperados. Al igual que en [41], después de cada sesión de búsqueda se requiere que los usuarios valoren los documentos accedidos por su relevancia. Las recomendaciones o valoraciones permitirán más adelante la realimentación a un componente de aprendizaje. Estas recomendaciones junto con los tópicos de interés expresados por un usuario en una sesión de búsqueda, representan su perfil en el sistema. En cada sesión, el componente de aprendizaje usa las recomendaciones para generar el perfil colaborativo del usuario, es decir, su comunidad de usuarios similares. Para ello, el componente de aprendizaje plantea una medida de distancia para clasificar en comunidades los perfiles de usuarios. Cada comunidad representa un grupo cuyos miembros comparten intereses comunes, basándose en las recomendaciones que previamente han proporcionado sobre los documentos a los que han accedido. Las recomendaciones previas proporcionadas por un usuario pueden modificar directamente su correspondiente perfil colaborativo (en cada sesión, la comunidad de usuarios similares puede cambiar) e indirectamente los del resto de usuarios. Por lo tanto, en este modelo multi-agente los perfiles de los usuarios son dinámicos y adaptables, de acuerdo con la realimentación proporcionada por los usuarios según su reacción ante la información recibida. Debemos comentar que esta propuesta de filtrado proporciona resultados más satisfactorios que los obtenidos con la propuesta en [41], y sin necesidad de un mayor esfuerzo por parte de los usuarios.

Hemos dividido el capítulo en tres secciones. En la Sección 2 estudiamos los sistemas multi-agente y hacemos un repaso del modelo previo que nos han servido de base para proponer el nuevo modelo. En la Sección 3 estudiamos el nuevo modelo, presentando en primer lugar su arquitectura, analizando cada uno de los niveles que lo componen y a continuación analizamos su funcionamiento, es decir, cómo desarrolla su actividad.

4.2. Preliminares

4.2.1. Sistemas Multi-agente

Los agentes software inteligentes ya han sido definidos previamente en varias ocasiones en la literatura al respecto [13, 71, 87, 108], por lo que no vamos a dar una nueva definición de este concepto, ni a revisar las ya existentes puesto que no es el tema que nos ocupa. En lugar de ello, nos vamos a centrar únicamente en aquellos aspectos relacionados con nuestro propósito específico, comenzando por el concepto de *agente* o *agente autónomo*. Según indica Maes en [71], un agente es un sistema que intenta alcanzar una serie de objetivos predefinidos en un entorno complejo y dinámico. Según sea dicho entorno, podemos establecer una distinción del concepto de agente entre los llamados habitualmente *robots* cuyo entorno es fundamentalmente físico y los llamados *agentes software* cuyo entorno consiste en computadores y redes. Ambos conceptos comparten una característica fundamental y es que son autónomos, es decir, capaces de operar y decidir por sí mismos la

forma de conseguir los objetivos a alcanzar. Sin embargo, como esta característica es de por sí inherente al concepto de agente, un agente autónomo suele denominarse simplemente agente. Con respecto al término *inteligente*, podemos encontrar diserciones acerca de considerar si un agente es o no inteligente por naturaleza [87]. Nosotros los consideraremos como inteligentes, debido a que presentan, en cierto sentido, un comportamiento humano reduciendo así el trabajo a realizar por los usuarios de Internet. Por tanto, los agentes con los que vamos a trabajar, son considerados *agentes inteligentes*.

Una vez definido lo que es un agente, vamos a definir el concepto de sistema multi-agente. Un **sistema multi-agente** es aquel en el que cierto número de agentes individuales cooperan e interactúan entre sí en un entorno distribuido para conseguir un objetivo global. En la Web, este objetivo global consiste en asistir a los usuarios de Internet en sus procesos de búsqueda de información, mediante la participación de agentes inteligentes distribuidos encargados de encontrar la información que mejor se ajuste a las necesidades de información de los usuarios.

Los sistemas multi-agente han sido ampliamente estudiados y usados en aplicaciones Web [20, 27, 68, 79, 97]. En [28, 59] podemos encontrar estudios detallados al respecto. Como los agentes inteligentes trabajan de forma autónoma y pueden aprender de las acciones de los usuarios, la tecnología de agentes también ha sido aplicada en el ámbito del FI para reducir la sobrecarga de información en la Web [70]. Por ejemplo, trabajando en el dominio de las listas de distribución de noticias, NewT [70] usa espacio vectorial basado en algoritmos genéticos para aprender qué artículos deben ser seleccionados y cuáles no, NewsWeeder [65] ayuda a los usuarios filtrando noticias aprendiendo a partir de ejemplos, y GroupLens [61] que

mide la similaridad entre usuarios de acuerdo al consenso que hayan tenido previamente sobre la relevancia de los ítems recomendados. Por otra parte, RE:Agent [8] usa técnicas de aprendizaje basadas en las acciones previas del usuario para clasificar el correo electrónico, y Amalthea [79] es un sistema multi-agente que recomienda sitios Web.

Los agentes no sólo recuperan y filtran información en el sentido de documentos Web [73], sino que también gestionan correos electrónicos, listas de noticias, listas FAQ, etc. [68, 71, 108]. Debido a que están más cercanos al usuario, son conocidos como *agentes de interface* [71]. Sin embargo, toda la información que estos agentes obtienen y gestionan, proviene de alguna parte. En efecto, hay servidores a través de Internet que proporcionan esos servicios de información, mail, noticias y FAQs. A los agentes más cercanos a estas fuentes de información se les conoce como *agentes de información* [104]. Puesto que los usuarios de Internet pueden acceder a los agentes de interface así como a los agentes de información generales, debido a la gran cantidad de dichos agentes, los usuarios se sienten perdidos y sobrecargados de información. Este hecho nos revela la necesidad de una organización entre los agentes, que implica tanto una jerarquía como una arquitectura. Debido a que los elementos que forman parte del proceso de recuperación de información se disponen de forma distribuida, parece sensato considerar una arquitectura distribuida. Para estos modelos multi-agente distribuidos se han propuesto y analizado diversas arquitecturas, pero para nuestro estudio concreto destacamos la propuesta en [104]. En esta arquitectura, además de los antes mencionados agentes de interface y de información, los autores consideran un tercer tipo de agentes, los *agentes de tareas*. Estos agentes gestionan los procesos de toma de decisión y el intercambio de información con los agentes de información, resolviendo posibles

conflictos que pudieran surgir y realizando procesos de agregación de información, con el objetivo de liberar a los agentes de interface de algunas tareas que los hacen menos eficientes. Por lo tanto, este modelo desarrolla la actividad de recuperación contando con cinco niveles: *usuarios de internet*, *agentes de interface*, *agentes de tareas*, *agentes de información* y *fuentes de información*.

En la figura 4.1 podemos observar la arquitectura de un modelo multi-agente de cinco niveles, en un escenario con un único usuario [25].

Sin embargo, la ausencia de conexión y comunicación entre los agentes ha provocado un descenso en la calidad y conveniencia de la información recuperada, además de en la eficiencia del sistema en las tareas de recuperación y filtrado. Por lo tanto, uno de los aspectos fundamentales a considerar es el diseño de protocolos apropiados de comunicación entre los agentes implicados. La gran variedad de representaciones y evaluaciones de la información en Internet constituye el principal obstáculo para esta comunicación entre agentes, y el problema es más acusado en los casos en los que los usuarios forman parte del proceso. Este aspecto acentúa la necesidad de una mayor flexibilidad en la comunicación entre los agentes y entre éstos y los usuarios [111, 112]. Para solucionar este problema, se han propuesto algunos enfoques para gestionar información flexible mediante el uso información lingüística tanto a nivel de agentes como de usuarios [24, 25, 110].

4.2.2. Un Modelo Multi-agente Basado en Información Lingüística y Técnicas de Filtrado de Información

Como hemos visto en el capítulo 2, una técnica que permite mejorar la eficiencia de

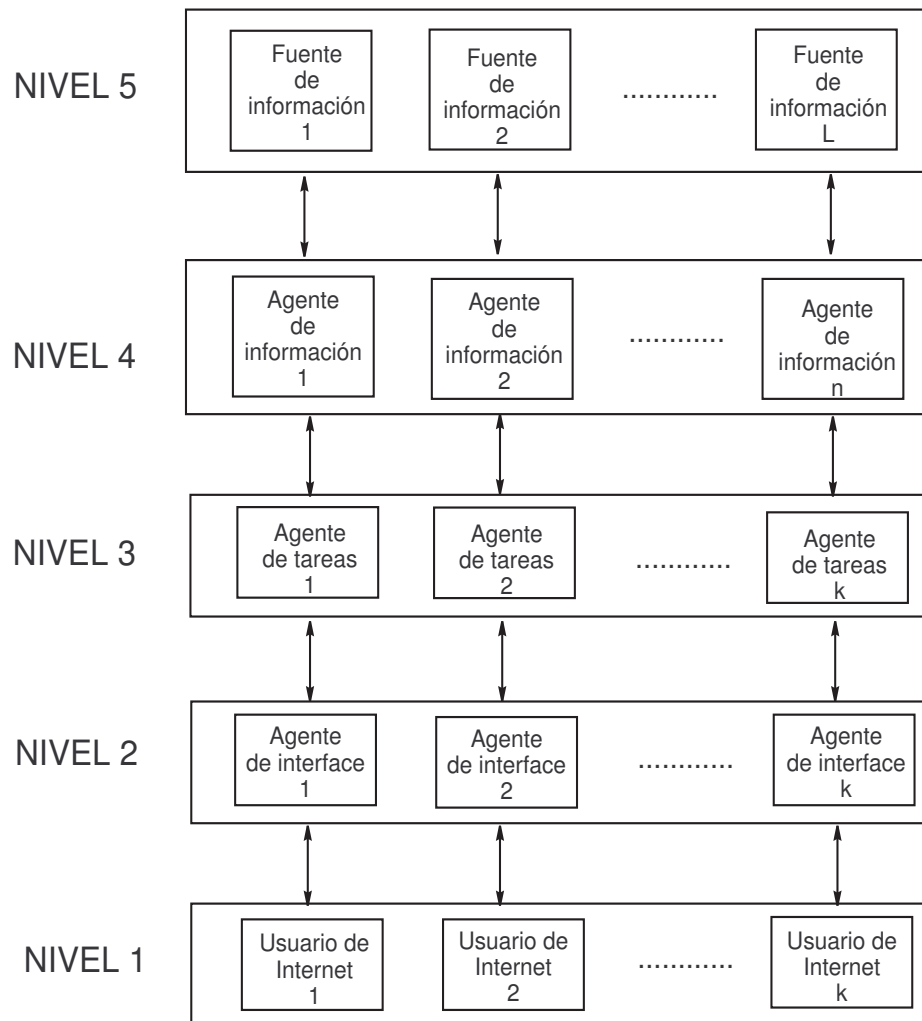


Figura 4.1: Arquitectura de un modelo multi-agente de cinco niveles.

los motores de búsqueda es la de filtrar la gran cantidad de información disponible habitualmente. En este sentido, los sistemas de FI son herramientas útiles para realizar las actividades de evaluación y filtrado, y concretamente el uso combinado de sistemas de FI junto con sistemas de búsqueda multi-agente ha dado muy buenos resultados en la Web.

Ello justifica la idea seguida en el modelo propuesto en [41] consistente en incorporar el uso de sistemas de FI al modelo multi-agente presentado en [25]. Mediante dicha incorporación podemos incrementar las posibilidades de filtrado de información en la Web. Para conseguirlo, se propone un nuevo modelo multi-agente lingüístico difuso que combina en su funcionamiento las dos técnicas posibles de filtrado, que son el filtrado basado en contenidos y el filtrado colaborativo. Por lo tanto, supone la incorporación de dos nuevos niveles a la arquitectura de 5 niveles que hemos visto previamente: por un lado está el nivel correspondiente a los *agentes de filtrado basado en contenidos* y por otro el nivel correspondiente al *agente de filtrado colaborativo*. Además, se incrementan las posibilidades de expresión de los usuarios, que especifican sus necesidades de información mediante una consulta lingüística con múltiples pesos y seleccionan una categoría de información a la que pertenece su petición. Los lenguajes de consulta multi-ponderados permiten a los usuarios expresar mejor sus ideas sobre los conceptos de relevancia y de esta manera, los sistemas de acceso a la información tienen mayores posibilidades de encontrar los documentos deseados [38, 39]. Cada uno de los términos de una consulta de usuario puede ser ponderado simultáneamente por dos pesos lingüísticos. El primero de los pesos está asociado con una semántica de umbral clásica, mientras que el segundo se asocia con una semántica de importancia relativa:

- Asociando *pesos de umbral* a los términos de una consulta, el usuario está pidiendo recuperar todos los documentos suficientemente relacionados con los tópicos representados por tales términos. Estos pesos de umbral son usados por los agentes de filtrado basado en contenidos para realizar un primer filtrado de los documentos a recuperar.
- Asociando *pesos de importancia relativa* a los términos de una consulta, el usuario está pidiendo recuperar todos los documentos cuyo contenido representa el concepto que está más asociado con el término más importante, y menos asociado con el término menos importante. Los pesos de importancia relativa son usados por los agentes de tareas para determinar el número de documentos que serán recuperados de cada uno de los agentes de filtrado basado en contenidos.

Por otro lado, la categoría de información representa el tópico de interés de la información necesaria para el usuario, como por ejemplo, *recuperación de información, medicina, toma de decisiones*, etc. Previamente, el diseñador del sistema ha tenido que establecer el listado de categorías de información que se vayan a presentar al usuario cuando va a realizar sus consultas. Esta categoría de información es usada por el agente de filtrado colaborativo para llevar a cabo una segunda fase de filtrado de los documentos que serán recuperados y mostrados definitivamente al usuario.

Este nuevo model multi-agente presenta una arquitectura jerárquica de siete niveles (ver figura 4.2):

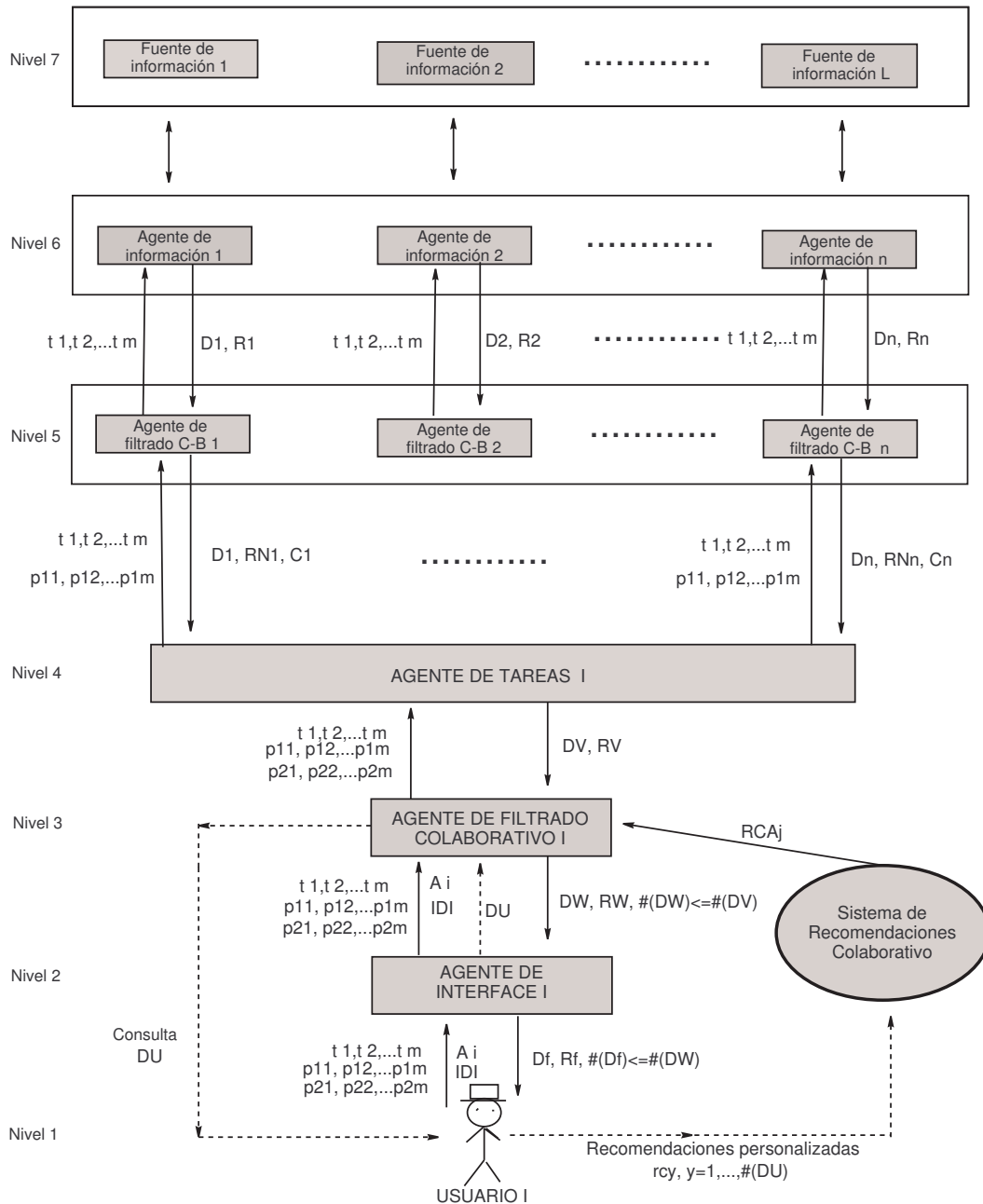


Figura 4.2: Arquitectura de un modelo multi-agente basado en agentes de filtrado.

Nivel 1. *Usuarios de Internet*, que expresan sus necesidades de información por medio de una consulta lingüística con múltiples pesos $\{(t_1, p_1^1, p_1^2), (t_2, p_2^1, p_2^2), \dots, (t_m, p_m^1, p_m^2)\}$, $p_i^1, p_i^2 \in S$ y una categoría de información $\mathcal{A}_i \in \{\mathcal{A}_1, \dots, \mathcal{A}_l\}$. También introducen su identificación \mathcal{ID} , usando por ejemplo su email o un nombre de usuario.

Nivel 2. *Agente de interface* (uno por usuario), que comunica al agente de filtrado colaborativo la consulta expresada por el usuario, la categoría de información y la identificación del usuario, filtra los documentos recuperados por el agente de filtrado colaborativo para mostrar al usuario aquellos que mejor satisfacen sus necesidades, y por último, informa al agente de filtrado colaborativo sobre el conjunto de documentos usados por el usuario para satisfacer sus necesidades de información DU .

Nivel 3. *Agente de filtrado colaborativo* (uno por agente de interface), que comunica la consulta multi-ponderada del usuario al agente de tareas, recibe los documentos más relevantes elegidos por el agente de tareas, y recupera del SR colaborativo las recomendaciones existentes sobre tales documentos usando para ello la categoría de información expresada por el usuario $RC^{\mathcal{A}_i} = \{RC_1^{\mathcal{A}_i}, \dots, RC_v^{\mathcal{A}_i}\}$ $RC_j^{\mathcal{A}_i} \in Sx[-0.5, 0.5)$. Usando dichas recomendaciones filtra los documentos reajustando su relevancia, y comunica al agente de interface estos documentos junto con sus nuevos grados de relevancia. Posteriormente, este agente efectúa sobre el SR colaborativo la actualización de las recomendaciones sobre los documentos usados por el usuario, es decir, invita al usuario a suministrar una recomendación rc_y sobre cada documento seleccionado $d_y^U \in DU$ y esta recomendación es almacenada en el SR colaborativo junto con las recomendaciones proporcionadas por otros

usuarios que también hayan usado d_y^U .

Nivel 4. *Agente de tareas* (uno por agente de filtrado colaborativo), que comunica a los agentes de filtrado basado en contenidos los términos de la consulta del usuario junto con sus respectivos pesos de umbral y filtra los documentos suministrados por los agentes de filtrado basado en contenidos, obteniendo de cada uno de ellos aquellos documentos que mejor satisfacen la consulta ponderada, fusionándolos y resolviendo los posibles conflictos que pudieran surgir entre dichos agentes.

Nivel 5. *Agentes de filtrado basado en contenidos* (uno por agente de información). Cada agente de filtrado basado en contenidos comunica a su respectivo agente de información los términos de la consulta del usuario, y filtra los documentos relevantes suministrados por su agente de información volviendo a calcular su relevancia, usando para ello los pesos de umbral. Entonces, el agente de tareas recibe de cada agente de filtrado basado en contenidos h un conjunto de documentos y su relevancia (D^h, RN^h) , donde cada documento d_j^h tiene asociado un grado de relevancia lingüístico expresado mediante la representación de las 2-tuplas $rn_j^h \in S \times [-0.5, 0.5)$ ($j = 1, \dots, \#(D^h)$). También recibe un conjunto de grados de satisfacción lingüísticos $C^h = \{c_1^h, c_2^h, \dots, c_m^h\}$, $c_i^h \in S \times [-0.5, 0.5)$ de este conjunto de documentos D^h con respecto a cada término de la consulta t_i .

Nivel 6. *Agentes de información*, que reciben de los agentes de filtrado basado en contenidos los términos de la consulta del usuario y realizan las búsquedas de los documentos en las fuentes de información. Entonces, cada agente de filtrado basado en contenidos h recibe de su correspondiente agente de información h el conjunto de documentos relevantes que se han encontrado

a través de las fuentes de información, D^h , y su correspondiente relevancia R^h , donde cada documento d_j^h tiene asociado un grado de relevancia $r_j^h \in S \times [-0.5, 0.5)$ ($j = 1, \dots, \#(D^h)$).

Nivel 7. *Fuentes de información*, consistentes en las fuentes de datos encontradas en Internet, tales como bases de datos o repositorios de información.

4.3. Un Sistema Multi-agente de Acceso a la Información en la Web Basado en Información Lingüística Multi-granular y en Técnicas de Filtrado

En esta sección, proponemos un modelo multi-agente que puede ser considerado como un refinamiento del anterior. Presentamos un modelo de sistema multi-agente lingüístico para el acceso a la información en Internet, donde la comunicación entre agentes de diferentes niveles y entre agentes y usuarios, es llevada a cabo usando distintos conjuntos de etiquetas, es decir, trabajando con información lingüística multi-granular y así conseguir una mayor flexibilidad en los procesos de comunicación del sistema. La relevancia es conceptualizada como la decisión del usuario de aceptar o rechazar la información recuperada por el sistema. La realimentación de relevancia es un proceso cíclico en el que los usuarios realimentan al sistema con decisiones sobre la relevancia de los documentos recuperados y entonces, el sistema usa estas evaluaciones para automáticamente modificar el proceso de filtrado.

En los procesos de acceso a la información vistos en el modelo anterior, podemos observar dos problemas:

1. La forma de representar la información lingüística. En dicho modelo siempre se usa el mismo conjunto de etiquetas para expresar las distintas valoraciones que aparecen en los procesos de comunicación. Sin embargo, cuando distintos expertos tienen distintos grados de incertidumbre sobre el fenómeno, son necesarios varios conjuntos de términos lingüísticos con diferente granularidad.
2. La forma de realizar el filtrado colaborativo. Como hemos visto, el sistema es colaborativo, es decir, que diferentes usuarios colaboran en las recomendaciones generadas para un usuario dado. Sin embargo, estas recomendaciones están basadas en todos los usuarios, y pensamos que en dicho proceso únicamente deberían intervenir aquellos usuarios que tengan preferencias similares.

Con el modelo que presentamos en este capítulo solucionamos esos dos problemas, permitiendo por un lado que las distintas valoraciones de los procesos de comunicación puedan ser valoradas sobre conjuntos de etiquetas distintos, es decir, usando información lingüística multi-granular, y por otro lado trabajando con las preferencias de los usuarios, es decir, manteniendo perfiles de usuarios. Asumimos que en el sistema multi-agente, los pesos de umbral y los pesos de importancia relativa asociados con los términos de las consultas de los usuarios, las recomendaciones proporcionadas por los usuarios sobre los documentos a los que acceden,

el grado de relevancia de los documentos recuperados y los grados de satisfacción de las consultas ponderadas de los usuarios, son expresados por medio de valores lingüísticos valorados sobre conjuntos de términos lingüísticos de diferente granularidad, S_1 , S_2 , S_3 , S_4 y S_5 respectivamente. Estos conjuntos de etiquetas S_i son seleccionados de una jerarquía lingüística LH , es decir, $S_i \in LH$. Por ejemplo, asumiendo la jerarquía lingüística definida en la figura 3.6 del capítulo anterior, que incluye 3 niveles de 3, 5 y 9 etiquetas cada uno, los usuarios pueden valorar los pesos de umbral en el segundo nivel ($S_1 = S^5$), los pesos de importancia relativa asociados con los términos de la consulta en el primero ($S_2 = S^3$) y las recomendaciones en el tercero ($S_3 = S^9$) y los agentes pueden valorar los grados de relevancia de los documentos recuperados en el tercer nivel ($S_4 = S^9$) y los grados de satisfacción de una consulta en el segundo ($S_5 = S^5$). Un aspecto a tener en cuenta es que el número de conjuntos de etiquetas distintos que podemos usar está limitado por el número de niveles de la jerarquía LH , por lo que en determinadas ocasiones distintos conjuntos de etiquetas S_i y S_j , pueden estar asociados al mismo conjunto de etiquetas de LH pero con diferentes interpretaciones, dependiendo del concepto que vaya a ser representado.

Los perfiles son las representaciones de las preferencias de información de los usuarios, y por tanto, pueden ser usados para filtrar los flujos de documentos recuperados por el sistema. El rendimiento del sistema dependerá en gran medida de la posibilidad de aprender perfiles para representar los intereses actuales de los usuarios. Para crear y actualizar los perfiles, usamos realimentación de relevancia por parte de los usuarios. La realimentación de relevancia es un proceso cíclico en el que los usuarios realimentan al sistema con decisiones sobre la relevancia de los documentos recuperados y entonces, el sistema usa estas evaluaciones para au-

tomáticamente modificar el proceso de recuperación [62]. Podemos discernir entre tres maneras de adquisición de perfiles: explícita, implícita y una combinación de ambas. En el enfoque explícito, los usuarios directamente expresan sus preferencias u opiniones en cada tópico. En el enfoque implícito, el sistema captura el perfil del usuario indirectamente, basándose en las evaluaciones que haya realizado el usuario sobre los documentos a los que accede. Después de que el usuario acceda a un documento, se le da la opción de valorar el documento, y dicha valoración es tratada como realimentación al sistema y, por tanto, usada para capturar y actualizar el perfil del usuario. Por último, en el enfoque híbrido, la primera vez que un usuario accede al sistema se le da la opción de que establezca su perfil, y en las siguientes sesiones se le da la opción de proporcionar realimentación de relevancia, que será usada en la actualización continua de su perfil.

En el modelo propuesto adoptamos el enfoque híbrido. En la primera sesión de cada usuario se le muestra un formulario en el que podrá seleccionar su interés en cada tópico, seleccionando un valor de relevancia según el rango definido por las etiquetas de S_4 . Después de cada sesión de búsqueda, se le pide a los usuarios que valoren los documentos accedidos por su relevancia. Estas recomendaciones junto con los tópicos de interés expresados previamente, representan su perfil en el sistema. En cada sesión, el componente de aprendizaje usa las recomendaciones para generar el perfil colaborativo del usuario, es decir, que mediante una medida de similaridad se determina su comunidad de usuarios similares. Cada comunidad representa un grupo cuyos miembros comparten intereses comunes, basándose en las recomendaciones que previamente han proporcionado sobre los documentos a los que han accedido. Las recomendaciones previas proporcionadas por un usuario pueden modificar directamente su correspondiente perfil colaborativo e indirecta-

mente los del resto de usuarios, por lo que en cada sesión, la comunidad de usuarios similares puede cambiar.

A continuación, vamos a presentar la arquitectura de este nuevo modelo multi-agente y posteriormente veremos su funcionamiento.

4.3.1. Arquitectura del Modelo

Al igual que el modelo visto en la sección anterior [41], el nuevo modelo multi-agente que proponemos también presenta una arquitectura jerárquica que contiene siete niveles de actividad, pero en todos ellos trabajando con información lingüística multi-granular:

Nivel 1. *Usuarios de Internet*, que expresan sus necesidades de información por medio de una consulta lingüística multi-ponderada. Cada término de la consulta del usuario debe ser valorado simultáneamente por dos pesos lingüísticos. El primero de los pesos está asociado con una semántica de umbral clásica mientras que el segundo se asocia con una semántica de importancia relativa. Entonces, el usuario realiza una consulta para buscar aquellos documentos relacionados con los términos $\{t_1, t_2, \dots, t_m\}$, que son ponderados por un grado lingüístico de umbral $\{p_1^1, p_2^1, \dots, p_m^1\}$ con $p_i^1 \in S_1$, y por un grado lingüístico de importancia relativa $\{p_1^2, p_2^2, \dots, p_m^2\}$ con $p_i^2 \in S_2$. Si es la primera vez que el usuario accede al sistema, deberá definir su perfil \mathcal{P}_i identificando sus intereses en cada tópico, seleccionando un valor de relevancia según el rango definido por las etiquetas de S_4 . En las siguientes

sesiones el sistema automáticamente mantendrá el perfil del usuario. Toda esta información es introducida por el usuario en el *agente de interface*.

Nivel 2. *Agente de interface* (uno por usuario), que comunica al agente de filtrado colaborativo la consulta ponderada del usuario, y filtra los documentos recuperados por el agente de filtrado colaborativo para mostrar al usuario aquellos que mejor satisfacen sus necesidades. Por último, informa al agente de filtrado colaborativo sobre el conjunto de documentos usados por el usuario para satisfacer sus necesidades de información DU .

Nivel 3. *Agente de filtrado colaborativo* (uno por agente de interface), que comunica al agente de tareas la consulta multi-ponderada del usuario, recibe los documentos más relevantes seleccionados por el agente de tareas, recupera del SR colaborativo las recomendaciones existentes sobre tales documentos usando únicamente las recomendaciones de los usuarios que tengan un perfil similar al usuario que inserta la consulta $RC^{P_i} = \{RC_1^{P_i}, \dots, RC_v^{P_i}\}$ $RC_j^{P_i} \in S_3 \times [-0.5, 0.5)$, filtra los documentos reajustando su relevancia usando dichas recomendaciones, y comunica al agente de interface estos documentos junto con sus nuevos grados de relevancia. Observemos que en este caso, para representar las recomendaciones, se usa otro conjunto de etiquetas S_3 distinto de los usados en el nivel 1 para expresar los pesos de umbral y de importancia relativa. Posteriormente, este agente efectúa sobre el SR colaborativo la actualización de las recomendaciones sobre los documentos usados por el usuario, es decir, invita al usuario a suministrar una recomendación rc_y sobre cada documento seleccionado $d_y^U \in DU$ y esta recomendación es almacenada en el SR colaborativo junto con las recomendaciones proporcionadas por otros usuarios que también hayan accedido a d_y^U .

Nivel 4. *Agente de tareas* (uno por agente de filtrado colaborativo), que comunica a los agentes de filtrado basado en contenidos los términos de la consulta del usuario junto con sus respectivos pesos de umbral, y filtra aquellos documentos de cada agente de filtrado basado en contenidos que mejor satisfacen la consulta, fusionándolos y resolviendo los posibles conflictos que pudieran surgir.

Nivel 5. *Agentes de filtrado basado en contenidos* (uno por agente de información). Cada agente de filtrado basado en contenidos comunica a su respectivo agente de información los términos de la consulta del usuario, y filtra los documentos relevantes suministrados por su agente de información, volviendo a calcular su relevancia usando para ello los pesos de umbral. Entonces, el agente de tareas recibe de cada agente de filtrado basado en contenidos h un conjunto de documentos y su relevancia (D^h, RN^h) , donde cada documento d_j^h tiene asociado un grado de relevancia lingüístico expresado mediante la representación de las 2-tuplas $rn_j^h \in S_4 \times [-0.5, 0.5)$ ($j = 1, \dots, \#(D^h)$). También recibe un conjunto de grados de satisfacción lingüísticos $C^h = \{c_1^h, c_2^h, \dots, c_m^h\}$, $c_i^h \in S_5 \times [-0.5, 0.5)$ de este conjunto de documentos D^h con respecto a cada término de la consulta t_i . Vemos que aparecen dos nuevos conjuntos de etiquetas lingüísticas, S_4 y S_5 , para distinguirlos de los términos valorados en otros niveles.

Nivel 6. *Agentes de información*, que reciben de los agentes de filtrado basado en contenidos los términos de la consulta del usuario y realizan las búsquedas de los documentos en las fuentes de información. Entonces, cada agente de filtrado basado en contenidos h recibe de su correspondiente agente de información h el conjunto de documentos relevantes que se han encontrado

a través de las fuentes de información, D^h , y su correspondiente relevancia R^h , donde cada documento d_j^h tiene asociado un grado de relevancia $r_j^h \in S_4 \times [-0.5, 0.5)$ ($j = 1, \dots, \#(D^h)$).

Nivel 7. *Fuentes de información*, consistentes en las fuentes de datos encontradas en Internet, tales como bases de datos o repositorios de información.

4.3.2. Funcionamiento del Modelo

La actividad de este nuevo modelo multi-agente se divide en dos fases principales:

1. *Fase de recuperación.* Esta primera fase coincide con el proceso de acceso a la información propiamente dicho, es decir, esta fase comienza cuando un usuario especifica una consulta y finaliza cuando selecciona sus documentos deseados de entre los documentos relevantes recuperados y suministrados por el sistema.
2. *Fase de realimentación.* Esta segunda fase consiste en el proceso de actualización, por parte del SR colaborativo, de las recomendaciones existentes sobre los documentos seleccionados, es decir, esta fase comienza cuando el *agente de interface* informa al *agente de filtrado colaborativo* sobre los documentos seleccionados por el usuario, y finaliza cuando el SR agrega y actualiza las recomendaciones sobre dichos documentos.

A continuación vamos a explicar con detalle cada una de las dos fases.

4.3.2.1. Fase de recuperación

El procedimiento de acceso a la información seguido por el sistema multi-agente es descrito a través de los siguientes pasos:

- **Paso 1.** Un *usuario de Internet* expresa sus requerimientos de información por medio de una consulta lingüística multi-ponderada $\{(t_1, p_1^1, p_1^2), (t_2, p_2^1, p_2^2), \dots, (t_m, p_m^1, p_m^2)\}$, con $p_i^1 \in S_1$ y $p_i^2 \in S_2$. Si es la primera vez que el usuario accede al sistema, tendrá que definir su perfil \mathcal{P}_i identificando sus intereses en cada tópico y valorándolos según las etiquetas de S_4 . El sistema también requiere la identificación del usuario \mathcal{ID} . Toda esta información es dada por el usuario al *agente de interface*.
 - **Paso 2.** El *agente de interface* pasa la consulta junto con el perfil del usuario \mathcal{P}_i (sólo la primera vez) al *agente de filtrado colaborativo*.
 - **Paso 3.** El *agente de filtrado colaborativo* envía al *agente de tareas* los términos de la consulta y sus pesos de importancia.
 - **Paso 4.** El *agente de tareas* comunica los términos de la consulta y sus pesos de importancia a todos aquellos *agentes de filtrado basado en contenidos* a los que esté conectado.
 - **Paso 5.** Cada *agente de filtrado basado en contenidos* h realiza la consulta a su correspondiente *agente de información* h y le proporciona los términos de la consulta $\{t_1, t_2, \dots, t_m\}$.
 - **Paso 6.** Todos los *agentes de información* que han recibido la consulta, buscan la información que mejor la satisface en las *fuentes de información*
-

y recuperan de dichas fuentes los documentos seleccionados. Asumimos que en las *fuentes de información* los documentos están representados usando una representación basada en términos índice, como en RI [38, 39, 92]. Por tanto, existe un conjunto finito de términos índice $T = \{t_1, \dots, t_l\}$ usado para representar los documentos, y cada documento d_j es representado como un subconjunto difuso:

$$d_j = \{(t_1, F(d_j, t_1)), \dots, (t_l, F(d_j, t_l))\}, F(d_j, t_i) \in [0, 1],$$

donde F es cualquier función de indexación numérica que pesa los términos índice de acuerdo con su trascendencia para describir el contenido del documento. $F(d_j, t_i) = 0$ implica que el documento d_j no está para nada relacionado con el concepto representado por el término t_i , mientras que $F(d_j, t_i) = 1$ implica que el documento d_j está perfectamente representado por el concepto indicado por t_i .

- **Paso 7.** Cada *agente de filtrado basado en contenidos h* recibe de su correspondiente *agente de información h* un conjunto de documentos y sus relevancias (D^h, R^h) ordenados decrecientemente según su relevancia. Cada documento d_j^h tiene asociado un grado lingüístico de relevancia $r_j^h \in S_4 \times [-0.5, 0.5)$ que ha sido calculado de la siguiente forma:

$$r_j^h = \bar{x}^e[\Delta(g \cdot F(d_j^h, t_1)), \dots, \Delta(g \cdot F(d_j^h, t_m))] = \Delta(g \cdot \sum_{i=1}^m \frac{1}{m} F(d_j^h, t_i)),$$

siendo $g + 1$ la cardinalidad de S_4 . Cada *agente de filtrado basado en contenidos h* filtra los documentos recibidos de su respectivo *agente de información h*, volviendo a calcular su relevancia por medio de una función de

similaridad lingüística:

$$e_h : (S_4 \times [-0.5, 0.5]) \times S_1 \rightarrow S_4 \times [-0.5, 0.5],$$

definida para modelizar la semántica de pesos de umbral asociados con los términos de la consulta. Esta función de similaridad lingüística requiere una transformación previa, puesto que para hacer uniforme la información lingüística multi-granular elegimos el conjunto de términos lingüísticos usado para expresar los grados de relevancia (S_4). Como los pesos de umbral están expresados como etiquetas de S_1 , debemos transformarlos en etiquetas de S_4 . Para realizar esta transformación, hacemos uso de la función que vimos en la definición 3.17 ($TF_{S_4}^t$) del capítulo 3, para transformar las etiquetas lingüísticas del nivel $S_1(t)$ en etiquetas del nivel $S_4(t')$:

$$TF_{S_4}^{S_1}(s_i^{n(S_1)}, \alpha^{n(S_1)}) = \Delta\left(\frac{\Delta^{-1}(s_i^{n(S_1)}, \alpha^{n(S_1)}) \cdot (n(S_4) - 1)}{n(S_1) - 1}\right)$$

obteniendo los nuevos pesos de umbral lingüísticos $\{p_1^{1'}, p_2^{1'}, \dots, p_m^{1'}\}$, $p_i^{1'} \in S_4$ para los términos $\{t_1, t_2, \dots, t_m\}$. Como podríamos suponer, diferentes *agentes de filtrado basado en contenidos* pueden tener distintas funciones de similaridad lingüísticas. Por ejemplo, algunas de las funciones de similaridad que podemos usar son las siguientes:

1. $e^1(\Delta(g \cdot F(d_j, t_i)), p_i^{1'}) = \begin{cases} (s_g, 0) & \text{si } \Delta(g \cdot F(d_j, t_i)) \geq (p_i^{1'}, 0) \\ (s_0, 0) & \text{en otro caso.} \end{cases}$
2. $e^2(\Delta(g \cdot F(d_j, t_i)), p_i^{1'}) = \begin{cases} \Delta(g \cdot F(d_j, t_i)) & \text{si } \Delta(g \cdot F(d_j, t_i)) \geq (p_i^{1'}, 0) \\ (s_0, 0) & \text{en otro caso.} \end{cases}$

$$3. e^3(\Delta(g \cdot F(d_j, t_i)), p_i^{1'}) = \begin{cases} \Delta(\min\{g, 0.5 + g \cdot F(d_j, t_i)\}) & \text{si } \Delta(g \cdot F(d_j, t_i)) \geq (p_i^{1'}, 0) \\ \Delta(\max\{0, g \cdot F(d_j, t_i) - 0.5\}) & \text{en otro caso.} \end{cases}$$

Entonces, cada *agente de filtrado basado en contenidos h* calcula un nuevo conjunto de grados de relevancia $RN^h = \{rn_j^h, j = 1, \dots, \#(D^h)\}$ para los documentos de D^h , de la siguiente manera:

$$rn_j^h = \bar{x}^e[e_h(\Delta(g \cdot F(d_j^h, t_1)), p_1^{1'}), \dots, e_h(\Delta(g \cdot F(d_j^h, t_m)), p_m^{1'})] = \Delta\left(\sum_{i=1}^m \frac{1}{m} \Delta^{-1}(e_h(\Delta(g \cdot F(d_j^h, t_i)), p_i^{1'}))\right).$$

- **Paso 8.** El *agente de tareas* recibe de cada *agente de filtrado basado en contenidos* un conjunto de documentos junto con su nueva relevancia (D^h, RN^h) . También recibe un conjunto de grados de satisfacción lingüísticos $C^h = \{c_1^h, c_2^h, \dots, c_m^h\}$, $c_i^h \in S_5 \times [-0.5, 0.5)$ de D^h con respecto a cada término de la consulta calculado como

$$c_i^h = \bar{x}^e[e_h(\Delta(g \cdot F(d_1^h, t_i)), p_i^{1''}), \dots, e_h(\Delta(g \cdot F(d_{\#(D^h)}^h, t_i)), p_i^{1''})] = \Delta\left(\sum_{j=1}^{\#(D^h)} \frac{1}{\#(D^h)} \Delta^{-1}(e_h(\Delta(g \cdot F(d_j^h, t_i)), p_i^{1''}))\right).$$

donde los $p_i^{1''}$ son los pesos p_i^1 expresados en el conjunto S_5 , usando para ello la función de transformación $TF_{S_5}^{S_1}$.

A continuación, el *agente de tareas* calcula el número de documentos que va a recuperar de cada *agente de filtrado basado en contenidos h*. Para hacerlo, se aplican los tres pasos siguientes:

- **Paso 8.1:** el *agente de tareas* ordena D^h según la nueva relevancia que se ha calculado.
- **Paso 8.2:** el *agente de tareas* agrega los pesos de información, tanto la satisfacción de los términos de la consulta de cada *agente de información* $(c_i^h, \alpha_i^w), c_i^h \in S_5$, como los pesos de importancia que el usuario asignó a dichos términos $(p_i^2, \alpha_i), p_i^2 \in S_2$. Para ello se usa el proceso de agregación de información lingüística multi-granular que se presentó en [57]:

1. *Fase de normalización:* para hacer uniforme la información lingüística multi-granular, seleccionamos el conjunto de términos lingüísticos usado para expresar la relevancia. Entonces, expresamos toda la información usando dicho conjunto de términos lingüísticos, por medio de la representación de las 2-tuplas.
2. *Fase de agregación:* agregamos la información usando un operador de agregación de las 2-tuplas. En este modelo, nosotros hemos seleccionado el operador de media ponderada lingüística, \bar{x}_i^w , para combinar las satisfacciones de los términos de la consulta y sus pesos de importancia.

Sean $\{[(p_1^2, \alpha_1), (c_1^h, \alpha_1^w)], \dots, [(p_m^2, \alpha_m), (c_m^h, \alpha_m^w)]\}$, $p_i^2 \in S_2$ y $c_i^h \in S_5$ un conjunto de parejas de 2-tuplas lingüísticas de importancia y satisfacción, que van a ser agregadas por el *agente de tareas* para cada *agente de información h*. Entonces, para combinarlas primeramente hay que transformar los valores $(p_i^2, \alpha_i), p_i^2 \in S_2$ y $(c_i^h, \alpha_i^w), c_i^h \in S_5$ para que estén expresados en el conjunto de términos lingüísticos usado

para expresar los grados de relevancia, en este caso S_4 , obteniéndose sus correspondientes valores $(p_i^{2'}, \alpha_i')$, $p_i^{2'} \in S_4$ y $(c_i^{h'}, \alpha_i^{w'})$, $c_i^{h'} \in S_4$. Una vez que la información lingüística multi-granular ha sido unificada mediante el operador de media ponderada lingüística, la agregación de las parejas asociadas con cada término se obtiene de la siguiente forma:

$$\lambda^h = \bar{x}_l^w([(p_1^{2'}, \alpha_1'), (c_1^{h'}, \alpha_1^{w'})], \dots, [(p_m^{2'}, \alpha_m'), (c_m^{h'}, \alpha_m^{w'})])$$

- **Paso 8.3:** para obtener los mejores documentos de los *agentes de filtrado basado en contenidos*, el *agente de tareas* selecciona un número de documentos $k(D^h)$ de cada *agente de filtrado basado en contenidos* h proporcional a su correspondiente grado de satisfacción λ^h :

$$k(D^h) = \text{round}\left(\frac{\sum_{i=1}^n \#(D^i)}{n} \cdot P_s^h\right),$$

donde $P_s^h = \frac{\Delta^{-1}(\lambda^h)}{\sum_{i=1}^n \Delta^{-1}(\lambda^i)}$ es la probabilidad de seleccionar documentos del *agente de filtrado basado en contenidos* h .

- **Paso 9.** El *agente de filtrado colaborativo* recibe del *agente de tareas* una lista de documentos $DV = \{d_1^V, \dots, d_v^V\}$ ordenada según la relevancia RV de los mismos, de tal forma que:

1. $r_j^V \geq r_{j+1}^V$,
 2. para un documento dado $d_j^V \in DV$ existe un h tal que $d_j^V \in D^h$ y $r_j^V \in RN^h$, y
 3. $\#(DV) = v \leq \sum_{i=1}^n k(D^i)$.
-

Entonces, el *agente de filtrado colaborativo* filtra los documentos proporcionados por el *agente de tareas* usando las recomendaciones que sobre esos documentos han proporcionado previamente otros usuarios con perfiles similares. Los usuarios con perfiles similares se mantienen agrupados en comunidades que se van calculando tras la fase de realimentación, por lo que detallaremos este proceso cuando estudiemos dicha fase. Estas recomendaciones se mantienen almacenadas junto con los perfiles de los usuarios en un SR colaborativo. Este proceso se divide en dos pasos:

- **Paso 9.1:** el *agente de filtrado colaborativo* consulta al SR colaborativo las recomendaciones existentes sobre DV que han proporcionado los usuarios con un perfil similar al del usuario activo \mathcal{P}_i (de la misma comunidad de usuarios), y las recupera:

$$RC^{\mathcal{P}_i} = \{RC_1^{\mathcal{P}_i}, \dots, RC_v^{\mathcal{P}_i}\}, RC_j^{\mathcal{P}_i} \in S_3 \times [-0.5, 0.5).$$

- **Paso 9.2:** usando estas recomendaciones $RC^{\mathcal{P}_i}$, el *agente de filtrado colaborativo* vuelve a calcular la relevancia de los documentos y los filtra. Entonces, para cada documento $d_j^V \in DV$ se calcula un nuevo grado de relevancia lingüística r_j^{NV} a partir de r_j^V y $RC^{\mathcal{P}_i}$, usando el operador \bar{x}^w , definido en la definición 3.14 del capítulo anterior:

$$r_j^{NV} = \bar{x}^w(r_j^V, TF_{S_4}^{S_3}(RC_j^{\mathcal{P}_i})),$$

usando por ejemplo el vector de pesos $W = [0.6, 0.4]$.

- **Paso 10.** El *agente de interface* recibe del *agente de filtrado colaborativo*
-

una lista de documentos $DW = \{d_1^W, \dots, d_w^W\}$ ordenada según su relevancia RW , de tal forma que:

1. $r_j^W \geq r_{j+1}^W$,
2. para un documento dado $d_j^W \in DW$ existe un i tal que $d_j^W = d_i^V$ y $r_j^W = r_i^{NV}$, y
3. $\#(DW) = w \leq v = \#(DV)$.

Entonces, el *agente de interface* filtra estos documentos con el objetivo de proporcionar al usuario únicamente aquellos documentos que mejor satisfacen sus necesidades, que simbolizamos como D_f . Por ejemplo, se puede seleccionar un número fijo de documentos K y mostrar los K mejores documentos.

4.3.2.2. Fase de realimentación

Esta fase abarca toda la actividad desarrollada por el SR colaborativo una vez que son entregados al usuario los documentos recuperados por el sistema multi-agente. Como hemos visto, en los sistemas de FI colaborativos los usuarios colaboran para ayudarse unos a otros a filtrar la información, registrando sus reacciones con respecto a los documentos a los que van accediendo [46, 91]. En nuestro modelo multi-agente, la actividad de realimentación es desarrollada a través de los siguientes pasos:

- **Paso 1.** El *agente de interface* envía al *agente de filtrado colaborativo* la identificación del usuario \mathcal{ID} (habitualmente será su email) junto con el conjunto de documentos $DU = \{d_1^U, \dots, d_u^U\}$, $u \leq \#(D_f)$ a los que el usuario ya ha accedido previamente.
- **Paso 2.** El *agente de filtrado colaborativo* consulta al usuario, por ejemplo por medio de un email, su opinión o juicios de evaluación sobre DU .
- **Paso 3.** El *usuario de Internet* comunica al SR colaborativo sus juicios de evaluación lingüísticos, rc_y , $y = 1, \dots, \#(DU)$, $rc_y \in S_3$.
- **Paso 4.** El módulo de aprendizaje usa estas recomendaciones para actualizar el perfil colaborativo del usuario, agregando las recomendaciones con los tópicos de interés que tenga seleccionados. Como es posible que el perfil haya cambiado, habrá que calcular de nuevo la comunidad de usuarios, midiendo la similaridad entre los perfiles (Sim). Como los perfiles los definen por un lado los tópicos de interés y por otro las recomendaciones, hay que tener en cuenta una parte estática (Se) correspondiente a los tópicos, y una parte dinámica (Sd) correspondiente a las recomendaciones. Tanto los tópicos como las recomendaciones se representan mediante vectores, por lo que para calcular ambas partes podemos usar una medida angular, tal como la Distancia Euclídea o la Medida del Coseno. Controlamos el peso de cada una de las partes mediante un factor $\alpha \in [0, 1]$. Por tanto, la similaridad entre dos perfiles p_1 y p_2 , se calcula de la siguiente manera:

$$Sim(p_1, p_2) = \alpha \cdot S_e(p_1, p_2) + (1 - \alpha) \cdot S_d(p_1, p_2)$$

Por último, si $Sim(p_1, p_2)$ supera cierto valor de corte definido previamente,

consideraremos que ambos perfiles son similares, por lo que los usuarios pertenecerán a la misma comunidad.

- **Paso 5.** El SR colaborativo vuelve a calcular las recomendaciones lingüísticas del conjunto de documentos DU , agregando las opiniones proporcionadas por la comunidad de usuarios calculada en el paso anterior. Para ello usamos el operador de agregación de 2-tuplas \bar{x}^e que vimos en la definición 3.13 del capítulo anterior. Por lo tanto, dado un documento $d_y^U \in DU$ para el que el usuario da una recomendación o juicio de evaluación $rc_y \in S_3$, y suponiendo que en el SR colaborativo ya se almacenan una serie de recomendaciones lingüísticas, $\{rc_1, \dots, rc_M\}$, $rc_i \in S_3$ asociadas con d_y^U , que fueron suministradas en búsquedas previas por M usuarios de la comunidad a la que pertenece el usuario, entonces el nuevo valor de la recomendación de d_y^U se obtiene de la siguiente manera:

$$RC_y^{\mathcal{P}_i} = \bar{x}^e[(rc_1, 0), \dots, (rc_M, 0), (rc_y, 0)].$$

4.3.3. Ejemplo de Funcionamiento

Por último vamos a ver un ejemplo sobre el funcionamiento del nuevo modelo multi-agente de acceso a información en la Web. Para el tratamiento de la información lingüística hemos adoptado el enfoque multi-granular, para lo que debemos definir una jerarquía lingüística; nosotros usaremos la definida en el apartado anterior. Concretamente las etiquetas de cada nivel son las siguientes:

- 1^{er} nivel: $S^3 = \{a_0 = Nulo = N, a_1 = Medio = M, a_2 = Total = T\}$.
-

- 2º nivel: $S^5 = \{b_0 = Nulo = N, b_1 = Bajo = L, b_2 = Medio = M, b_3 = Alto = H, b_4 = Total = T\}$
- 3er nivel: $S^9 = \{c_0 = Nulo = N, c_1 = Muy_Bajo = VL, c_2 = Bajo = L, c_3 = Algo_Bajo = MLL, c_4 = Medio = M, c_5 = Algo_Alto = MLH, c_6 = Alto = H, c_7 = Muy_Alto = VH, c_8 = Total = T\}$

Con esta jerarquía, para valorar los distintos conceptos usamos los conjuntos de etiquetas que indicábamos antes, es decir, los usuarios pueden valorar los pesos de umbral en el segundo nivel ($S_1 = S^5$), los pesos de importancia relativa asociados con los términos de la consulta en el primero ($S_2 = S^3$) y las recomendaciones en el tercero ($S_3 = S^9$) y los agentes pueden valorar los grados de relevancia de los documentos recuperados en el tercer nivel ($S_4 = S^9$) y los grados de satisfacción de una consulta en el segundo ($S_5 = S^5$).

Para el desarrollo del ejemplo, vamos a considerar el punto de vista de un usuario i y cuatro agentes de información. Supongamos que el usuario está interesado en *Agentes*, y más concretamente en *Agentes Web*, para lo cual introduce la siguiente consulta:

$$(t_1, p_1^1, p_1^2) = (Agentes, H, T)$$

$$(t_2, p_2^1, p_2^2) = (Web, M, M)$$

donde $p_i^1 \in S_1$ y $p_i^2 \in S_2$. Con ello el usuario está indicando su preferencia por documentos que traten sobre *agentes* en un contexto *Web*, al menos con grados *alto* y *medio* respectivamente, y también indica su preferencia por documentos en los que el término *agente* sea más importante que *Web* asignando a estos términos los pesos de importancia relativa T y M respectivamente. Además, la primera vez

que el usuario accede al sistema define su perfil \mathcal{P}_i , seleccionando el tópico *Web Mining* con un peso $T \in S_4$. El usuario introduce esta información junto con su identidad \mathcal{ID} , en el agente de interface.

El agente de interface transfiere toda esta información al agente de filtrado colaborativo que comunica al agente de tareas la consulta multi-ponderada. El agente de tareas pasa al nivel de agentes de filtrado basado en contenidos los términos introducidos por el usuario junto con sus correspondientes pesos de umbral. Cada agente de filtrado basado en contenidos envía los términos a su correspondiente agente de información que se encarga de buscar en las fuentes de información aquellos documentos relacionados con los términos de la consulta, y obtiene una lista con los enlaces más relevantes. Por ejemplo, cada agente de información h , con $h = 1, \dots, 4$ podría recuperar un total de 5 enlaces ($j = 1, \dots, 5$), D^h y su relevancia R^h , con $r_j^h \in S_4 \times [-0.5, 0.5]$ (ver tabla 4.1).

Para este ejemplo, hemos supuesto que los documentos están representados tal y como se muestra en la tabla 4.2 ($F(d_j^h, Agentes)$ y $F(d_j^h, Web)$).

Con los valores de la tabla 4.2, calculamos los valores r_j^h de la tabla 4.1. Por ejemplo, veamos cómo se calculan los r_1^1 y con el resto se procedería de forma similar:

$$\begin{aligned}
 r_1^1 &= \bar{x}^e[\Delta(8 \cdot F(d_1^1, Agentes)), \Delta(8 \cdot F(d_1^1, Web))] = \Delta(8 \cdot \frac{0.9+0.5}{2}) = (H, -0.4) \\
 r_2^1 &= \bar{x}^e[\Delta(8 \cdot F(d_2^1, Agentes)), \Delta(8 \cdot F(d_2^1, Web))] = \Delta(8 \cdot \frac{0.6+0.8}{2}) = (H, -0.4) \\
 r_3^1 &= \bar{x}^e[\Delta(8 \cdot F(d_3^1, Agentes)), \Delta(8 \cdot F(d_3^1, Web))] = \Delta(8 \cdot \frac{0.2+1}{2}) = (MLH, -0.2) \\
 r_4^1 &= \bar{x}^e[\Delta(8 \cdot F(d_4^1, Agentes)), \Delta(8 \cdot F(d_4^1, Web))] = \Delta(8 \cdot \frac{0.9+0.1}{2}) = (M, 0) \\
 r_5^1 &= \bar{x}^e[\Delta(8 \cdot F(d_5^1, Agentes)), \Delta(8 \cdot F(d_5^1, Web))] = \Delta(8 \cdot \frac{0.4+0.4}{2}) = (MLL, 0.2).
 \end{aligned}$$

(D^h, R^h)	d_j^h	r_j^h
(D^1, R^1)	http://phonebk.duke.edu/clients/bnfaqent.html http://webhound.www.media.mit.edu/projects/webhound/doc/Webhound.html http://www.elet.polimi.it/section/compeng/air/agents/ http://www.cs.bham.ac.uk/~amw/agents/links/ http://groucho.gsfc.nasa.gov/Code\ 520/Code\ 522/Projects/Agents/	(H,-0.4) (H,-0.4) (MLH,-0.2) (M,0) (MLL,0.2)
(D^2, R^2)	http://lcs.www.media.mit.edu/people/lieber/Lieberary/Letizia/Letizia.html http://www.osf.org/ri/contracts/6.Rationale.frame.html http://www.info.unicaen.fr/~serge/sma.html http://www.cs.umbc.edu/~cikm/1994/ia/papers/jain.html http://www.hinet.com/realty/edge/gallery.html	(VH,0.2) (M,0.4) (M,0.4) (MLL,0.2) (L,-0.4)
(D^3, R^3)	http://activist.gpl.ibm.com/WhitePaper/ptc2.htm http://www.cs.umbc.edu/~cikm/ia/submitted/viewing/chen.html http://www.psychology.nottingham.ac.uk:80/aigr/research/agents/agents.html http://netq.rowland.org/isab/isab.html http://maple.net/gbd/salagnts.html	(VH,0.2) (MLH,-0.2) (MLH,-0.2) (M,0) (VL,-0.2)
(D^4, R^4)	http://www.ncsa.uiuc.edu/SDG/IT94/Proceedings/Agents/spetka/spetka.html http://mmm.wiwi.hu-berlin.de/MMM/cebit\ _engl.html http://foner.www.media.mit.edu/people/foner/Julia/subsection3\ _2\ _2.html http://www.cs.bham.ac.uk/~amw/agents/index.html http://www.ffly.com/html/About1.html	(VH,0.2) (MLH,-0.2) (MLL,0.2) (MLL,0.2) (L,-0.4)

Tabla 4.1: Conjuntos de documentos para los términos *Agentes* y *Web*.

Cada agente de información h devuelve a su correspondiente agente de filtrado basado en contenidos un conjunto de documentos D^h junto con su relevancia R^h y su representación original con respecto a los términos de la consulta. Entonces, cada agente de filtrado basado en contenidos h filtra los documentos aplicando los pesos de umbral por medio de una función de similaridad lingüística e_h , y así vuelve a calcular su relevancia. Esta función requiere una transformación previa de los pesos de umbral que están expresados como etiquetas de S_1 y deben ser transformados en etiquetas de S_4 obteniendo así los valores $p_i^{1'}$:

$$p_1^{1'} = TF_{S_4}^{S_1}(H, 0) = \Delta\left(\frac{\Delta^{-1}(H, 0) \cdot 8}{4}\right) = \Delta\left(\frac{3 \cdot 8}{4}\right) = (H, 0)$$

$$p_2^{1'} = TF_{S_4}^{S_1}(M, 0) = \Delta\left(\frac{\Delta^{-1}(M, 0) \cdot 8}{4}\right) = \Delta\left(\frac{2 \cdot 8}{4}\right) = (M, 0)$$

	Agentes	Web
$F(d_1^1, _)$	0.9	0.5
$F(d_2^1, _)$	0.6	0.8
$F(d_3^1, _)$	0.2	1
$F(d_4^1, _)$	0.9	0.1
$F(d_5^1, _)$	0.4	0.4

	Agentes	Web
$F(d_1^2, _)$	1	0.8
$F(d_2^2, _)$	0.5	0.6
$F(d_3^2, _)$	0.6	0.5
$F(d_4^2, _)$	0.4	0.4
$F(d_5^2, _)$	0.1	0.3

	Agentes	Web
$F(d_1^3, _)$	0.7	0.9
$F(d_2^3, _)$	0.8	0.4
$F(d_3^3, _)$	0.6	0.6
$F(d_4^3, _)$	0.3	0.7
$F(d_5^3, _)$	0.1	0.1

	Agentes	Web
$F(d_1^4, _)$	1	0.8
$F(d_2^4, _)$	0.4	0.8
$F(d_3^4, _)$	0.6	0.2
$F(d_4^4, _)$	0.6	0.2
$F(d_5^4, _)$	0.2	0.2

Tabla 4.2: Representaciones de D^h .

A continuación, cada agente de filtrado basado en contenidos calcula un nuevo conjunto de relevancia, para lo cual usamos por ejemplo la función de similaridad lingüística e^2 que vimos en el paso 7:

$$\begin{aligned}
 rn_1^1 &= \bar{x}^e[e^2(\Delta(8 \cdot 0.9), H), e^2(\Delta(8 \cdot 0.5), M)] = \Delta(\frac{1}{2} \cdot (\Delta^{-1}(VH, 0.2) + \Delta^{-1}(M, 0))) = (H, -0.4) \\
 rn_2^1 &= \bar{x}^e[e^2(\Delta(8 \cdot 0.6), H), e^2(\Delta(8 \cdot 0.8), M)] = \Delta(\frac{1}{2} \cdot (\Delta^{-1}(N, 0) + \Delta^{-1}(H, 0.4))) = (MLL, 0.2) \\
 rn_3^1 &= \bar{x}^e[e^2(\Delta(8 \cdot 0.2), H), e^2(\Delta(8 \cdot 1), M)] = \Delta(\frac{1}{2} \cdot (\Delta^{-1}(N, 0) + \Delta^{-1}(T, 0))) = (M, 0) \\
 rn_4^1 &= \bar{x}^e[e^2(\Delta(8 \cdot 0.9), H), e^2(\Delta(8 \cdot 0.1), M)] = \Delta(\frac{1}{2} \cdot (\Delta^{-1}(VH, 0.2) + \Delta^{-1}(N, 0))) = (M, -0.4) \\
 rn_5^1 &= \bar{x}^e[e^2(\Delta(8 \cdot 0.4), H), e^2(\Delta(8 \cdot 0.4), M)] = \Delta(\frac{1}{2} \cdot (\Delta^{-1}(N, 0) + \Delta^{-1}(N, 0))) = (N, 0)
 \end{aligned}$$

$$\begin{aligned}
 rn_1^2 &= \bar{x}^e[e^2(\Delta(8 \cdot 1), H), e^2(\Delta(8 \cdot 0.8), M)] = \Delta(\frac{1}{2} \cdot (\Delta^{-1}(T, 0) + \Delta^{-1}(H, 0.4))) = (VH, 0.2) \\
 rn_2^2 &= \bar{x}^e[e^2(\Delta(8 \cdot 0.5), H), e^2(\Delta(8 \cdot 0.6), M)] = \Delta(\frac{1}{2} \cdot (\Delta^{-1}(N, 0) + \Delta^{-1}(MLH, -0.2))) = (L, 0.4) \\
 rn_3^2 &= \bar{x}^e[e^2(\Delta(8 \cdot 0.6), H), e^2(\Delta(8 \cdot 0.5), M)] = \Delta(\frac{1}{2} \cdot (\Delta^{-1}(N, 0) + \Delta^{-1}(M, 0))) = (L, 0) \\
 rn_4^2 &= \bar{x}^e[e^2(\Delta(8 \cdot 0.4), H), e^2(\Delta(8 \cdot 0.4), M)] = \Delta(\frac{1}{2} \cdot (\Delta^{-1}(N, 0) + \Delta^{-1}(N, 0))) = (N, 0) \\
 rn_5^2 &= \bar{x}^e[e^2(\Delta(8 \cdot 0.1), H), e^2(\Delta(8 \cdot 0.3), M)] = \Delta(\frac{1}{2} \cdot (\Delta^{-1}(N, 0) + \Delta^{-1}(N, 0))) = (N, 0)
 \end{aligned}$$

$$\begin{aligned}
 rn_1^3 &= \bar{x}^e[e^2(\Delta(8 \cdot 0.7), H), e^2(\Delta(8 \cdot 0.9), M)] = \Delta(\frac{1}{2} \cdot (\Delta^{-1}(N, 0) + \Delta^{-1}(VH, 0.2))) = (M, -0.4) \\
 rn_2^3 &= \bar{x}^e[e^2(\Delta(8 \cdot 0.8), H), e^2(\Delta(8 \cdot 0.4), M)] = \Delta(\frac{1}{2} \cdot (\Delta^{-1}(H, 0.4) + \Delta^{-1}(N, 0))) = (MLL, 0.2) \\
 rn_3^3 &= \bar{x}^e[e^2(\Delta(8 \cdot 0.6), H), e^2(\Delta(8 \cdot 0.6), M)] = \Delta(\frac{1}{2} \cdot (\Delta^{-1}(N, 0) + \Delta^{-1}(MLH, -0.2))) = (L, 0.4) \\
 rn_4^3 &= \bar{x}^e[e^2(\Delta(8 \cdot 0.3), H), e^2(\Delta(8 \cdot 0.7), M)] = \Delta(\frac{1}{2} \cdot (\Delta^{-1}(N, 0) + \Delta^{-1}(H, -0.4))) = (MLL, 0.2) \\
 rn_5^3 &= \bar{x}^e[e^2(\Delta(8 \cdot 0.1), H), e^2(\Delta(8 \cdot 0.1), M)] = \Delta(\frac{1}{2} \cdot (\Delta^{-1}(N, 0) + \Delta^{-1}(N, 0))) = (N, 0)
 \end{aligned}$$

$$\begin{aligned}
 rn_1^4 &= \bar{x}^e[e^2(\Delta(8 \cdot 1), H), e^2(\Delta(8 \cdot 0.8), M)] = \Delta(\frac{1}{2} \cdot (\Delta^{-1}(T, 0) + \Delta^{-1}(H, 0.4))) = (VH, 0.2) \\
 rn_2^4 &= \bar{x}^e[e^2(\Delta(8 \cdot 0.4), H), e^2(\Delta(8 \cdot 0.8), M)] = \Delta(\frac{1}{2} \cdot (\Delta^{-1}(N, 0) + \Delta^{-1}(H, 0.4))) = (MLL, 0.2) \\
 rn_3^4 &= \bar{x}^e[e^2(\Delta(8 \cdot 0.6), H), e^2(\Delta(8 \cdot 0.2), M)] = \Delta(\frac{1}{2} \cdot (\Delta^{-1}(N, 0) + \Delta^{-1}(N, 0))) = (N, 0) \\
 rn_4^4 &= \bar{x}^e[e^2(\Delta(8 \cdot 0.6), H), e^2(\Delta(8 \cdot 0.2), M)] = \Delta(\frac{1}{2} \cdot (\Delta^{-1}(N, 0) + \Delta^{-1}(N, 0))) = (N, 0) \\
 rn_5^4 &= \bar{x}^e[e^2(\Delta(8 \cdot 0.2), H), e^2(\Delta(8 \cdot 0.2), M)] = \Delta(\frac{1}{2} \cdot (\Delta^{-1}(N, 0) + \Delta^{-1}(N, 0))) = (N, 0)
 \end{aligned}$$

Como podemos observar, la aplicación de los pesos de umbral puede variar la relevancia de los documentos y por tanto, afectar a la ordenación entre los mismos. Por otro lado, en cada agente de filtrado basado en contenido, también calculamos los grados de satisfacción de los términos de la consulta. Ello requiere aplicar la función de transformación $TF_{S_5}^{S_1}$ para expresar los p_i^1 como etiquetas de S_5 , pero en este ejemplo no es necesario este paso pues recordemos que tanto S_1 como S_5 se corresponden con el segundo nivel de la jerarquía, S^5 . Entonces, los c_i^h son

calculados de la siguiente forma:

$$\begin{aligned} c_1^1 &= \bar{x}^e [e^2(\Delta(4 \cdot 0.9), H), e^2(\Delta(4 \cdot 0.6), H), e^2(\Delta(4 \cdot 0.2), H), e^2(\Delta(4 \cdot 0.9), H), e^2(\Delta(4 \cdot 0.4), H)] = \\ &= \Delta(\frac{1}{5}(\Delta^{-1}(T, -0.4) + \Delta^{-1}(N, 0) + \Delta^{-1}(N, 0) + \Delta^{-1}(T, -0.4) + \Delta^{-1}(N, 0))) = \Delta(\frac{7.2}{5}) = (L, 0.44) \end{aligned}$$

$$\begin{aligned} c_2^1 &= \bar{x}^e [e^2(\Delta(4 \cdot 0.5), M), e^2(\Delta(4 \cdot 0.8), M), e^2(\Delta(4 \cdot 1), M), e^2(\Delta(4 \cdot 0.1), M), e^2(\Delta(4 \cdot 0.4), M)] = \\ &= \Delta(\frac{1}{5}(\Delta^{-1}(M, 0) + \Delta^{-1}(H, 0.2) + \Delta^{-1}(T, 0) + \Delta^{-1}(N, 0) + \Delta^{-1}(N, 0))) = \Delta(\frac{9.2}{5}) = (M, -0.16) \end{aligned}$$

$$\begin{aligned} c_1^2 &= \bar{x}^e [e^2(\Delta(4 \cdot 1), H), e^2(\Delta(4 \cdot 0.5), H), e^2(\Delta(4 \cdot 0.6), H), e^2(\Delta(4 \cdot 0.4), H), e^2(\Delta(4 \cdot 0.1), H)] = \\ &= \Delta(\frac{1}{5}(\Delta^{-1}(T, 0) + \Delta^{-1}(N, 0) + \Delta^{-1}(N, 0) + \Delta^{-1}(N, 0) + \Delta^{-1}(N, 0))) = \Delta(\frac{4}{5}) = (L, -0.2) \end{aligned}$$

$$\begin{aligned} c_2^2 &= \bar{x}^e [e^2(\Delta(4 \cdot 0.8), M), e^2(\Delta(4 \cdot 0.6), M), e^2(\Delta(4 \cdot 0.5), M), e^2(\Delta(4 \cdot 0.4), M), e^2(\Delta(4 \cdot 0.4), M)] = \\ &= \Delta(\frac{1}{5}(\Delta^{-1}(H, 0.2) + \Delta^{-1}(M, 0.4) + \Delta^{-1}(M, 0) + \Delta^{-1}(N, 0) + \Delta^{-1}(N, 0))) = \Delta(\frac{7.6}{5}) = (M, -0.48) \end{aligned}$$

$$\begin{aligned} c_1^3 &= \bar{x}^e [e^2(\Delta(4 \cdot 0.7), H), e^2(\Delta(4 \cdot 0.8), H), e^2(\Delta(4 \cdot 0.6), H), e^2(\Delta(4 \cdot 0.3), H), e^2(\Delta(4 \cdot 0.1), H)] = \\ &= \Delta(\frac{1}{5}(\Delta^{-1}(N, 0) + \Delta^{-1}(H, 0.2) + \Delta^{-1}(N, 0) + \Delta^{-1}(N, 0) + \Delta^{-1}(N, 0))) = \Delta(\frac{3.2}{5}) = (L, -0.36) \end{aligned}$$

$$\begin{aligned} c_2^3 &= \bar{x}^e [e^2(\Delta(4 \cdot 0.9), M), e^2(\Delta(4 \cdot 0.4), M), e^2(\Delta(4 \cdot 0.6), M), e^2(\Delta(4 \cdot 0.7), M), e^2(\Delta(4 \cdot 0.1), M)] = \\ &= \Delta(\frac{1}{5}(\Delta^{-1}(T, -0.4) + \Delta^{-1}(N, 0) + \Delta^{-1}(M, 0.4) + \Delta^{-1}(H, -0.3) + \Delta^{-1}(N, 0))) = \Delta(\frac{8.8}{5}) = (M, -0.24) \end{aligned}$$

$$\begin{aligned} c_1^4 &= \bar{x}^e [e^2(\Delta(4 \cdot 1), H), e^2(\Delta(4 \cdot 0.4), H), e^2(\Delta(4 \cdot 0.6), H), e^2(\Delta(4 \cdot 0.6), H), e^2(\Delta(4 \cdot 0.2), H)] = \\ &= \Delta(\frac{1}{5}(\Delta^{-1}(T, 0) + \Delta^{-1}(N, 0) + \Delta^{-1}(N, 0) + \Delta^{-1}(N, 0) + \Delta^{-1}(N, 0))) = \Delta(\frac{4}{5}) = (L, -0.2) \end{aligned}$$

$$\begin{aligned} c_2^4 &= \bar{x}^e [e^2(\Delta(4 \cdot 0.8), M), e^2(\Delta(4 \cdot 0.8), M), e^2(\Delta(4 \cdot 0.2), M), e^2(\Delta(4 \cdot 0.2), M), e^2(\Delta(4 \cdot 0.2), M)] = \\ &= \Delta(\frac{1}{5}(\Delta^{-1}(H, 0.2) + \Delta^{-1}(H, 0.2) + \Delta^{-1}(N, 0) + \Delta^{-1}(N, 0) + \Delta^{-1}(N, 0))) = \Delta(\frac{6.4}{5}) = (L, 0.28) \end{aligned}$$

A continuación, el agente de tareas calcula el número de documentos que va a recuperar de cada agente de filtrado basado en contenidos. Primeramente, ordena los documentos D^h con respecto a la nueva relevancia RN^h y después calcula λ^h , un grado de satisfacción global de la consulta para cada agente de filtrado basado en contenidos h . Este valor es calculado agregando los grados de satisfacción y los

pesos de importancia relativa, usando para ello el operador \bar{x}_l^w que vimos en el modelo de representación de 2-tuplas en el capítulo 3. Para ello, los $(p_i^2, \alpha_i) \in S_2$ y los $(c_i^h, \alpha_i^w) \in S_5$ deben ser transformados en etiquetas de S_4 , el conjunto de etiquetas lingüísticas que usamos para expresar la relevancia:

$$(p_1^{2'}, \alpha_1') = TF_{S_4}^{S_2}(T, 0) = \Delta\left(\frac{\Delta^{-1}(T,0) \cdot 8}{2}\right) = (T, 0)$$

$$(p_2^{2'}, \alpha_2') = TF_{S_4}^{S_2}(M, 0) = \Delta\left(\frac{\Delta^{-1}(M,0) \cdot 8}{2}\right) = (M, 0)$$

$$(c_1^{1'}, \alpha_1^{w'}) = TF_{S_4}^{S_5}(L, 0.44) = \Delta\left(\frac{\Delta^{-1}(L,0.44) \cdot 8}{4}\right) = (MLL, -0.12)$$

$$(c_2^{1'}, \alpha_2^{w'}) = TF_{S_4}^{S_5}(M, -0.16) = \Delta\left(\frac{\Delta^{-1}(M,-0.16) \cdot 8}{4}\right) = (M, -0.32)$$

$$(c_1^{2'}, \alpha_1^{w'}) = TF_{S_4}^{S_5}(L, -0.2) = \Delta\left(\frac{\Delta^{-1}(L,-0.2) \cdot 8}{4}\right) = (L, -0.4)$$

$$(c_2^{2'}, \alpha_2^{w'}) = TF_{S_5}^{S_9}(M, -0.48) = \Delta\left(\frac{\Delta^{-1}(M,-0.48) \cdot 8}{4}\right) = (MLL, 0.04)$$

$$(c_1^{3'}, \alpha_1^{w'}) = TF_{S_4}^{S_5}(L, -0.36) = \Delta\left(\frac{\Delta^{-1}(L,-0.36) \cdot 8}{4}\right) = (VL, 0.28)$$

$$(c_2^{3'}, \alpha_2^{w'}) = TF_{S_4}^{S_5}(M, -0.24) = \Delta\left(\frac{\Delta^{-1}(M,-0.24) \cdot 8}{4}\right) = (M, -0.48)$$

$$(c_1^{4'}, \alpha_1^{w'}) = TF_{S_4}^{S_5}(L, -0.2) = \Delta\left(\frac{\Delta^{-1}(L,-0.2) \cdot 8}{4}\right) = (L, -0.4)$$

$$(c_2^{4'}, \alpha_2^{w'}) = TF_{S_4}^{S_5}(M, 0.28) = \Delta\left(\frac{\Delta^{-1}(M,0.28) \cdot 8}{4}\right) = (MLL, -0.44)$$

Entonces se calculan los λ^h como sigue:

$$\lambda^1 = \bar{x}_l^w [((MLL, -0.12), (T, 0)), ((M, -0.32), (M, 0))] = \Delta\left(\frac{2.88 \cdot 8 + 3.68 \cdot 4}{12}\right) = \Delta(3.15) = (MLL, 0.15)$$

$$\lambda^2 = \bar{x}_l^w [((L, -0.4), (T, 0)), ((MLL, 0.04), (M, 0))] = \Delta\left(\frac{1.6 \cdot 8 + 3.04 \cdot 4}{12}\right) = \Delta(2.08) = (L, 0.08)$$

$$\lambda^3 = \bar{x}_l^w [((VL, 0.28), (T, 0)), ((M, -0.48), (M, 0))] = \Delta\left(\frac{1.28 \cdot 8 + 3.52 \cdot 4}{12}\right) = \Delta(2.03) = (L, 0.03)$$

$$\lambda^4 = \bar{x}_l^w [((L, -0.4), (T, 0)), ((MLL, -0.44), (M, 0))] = \Delta\left(\frac{1.6 \cdot 8 + 2.56 \cdot 4}{12}\right) = \Delta(1.92) = (L, -0.08)$$

Para recopilar los mejores documentos de cada agente de filtrado basado en con-

tenidos h , se selecciona un número de documentos $k(D^h)$ de cada uno de ellos que es proporcional a su grado de satisfacción λ^h , y para ello se calcula la probabilidad de selección de los documentos de cada agente de filtrado basado en contenidos:

$$\begin{aligned}
 P_s^1 &= \frac{\Delta^{-1}(\lambda^1)}{\Delta^{-1}(\lambda^1)+\Delta^{-1}(\lambda^2)+\Delta^{-1}(\lambda^3)+\Delta^{-1}(\lambda^4)} = \frac{3.15}{9.18} = 0.3431 \\
 P_s^2 &= \frac{\Delta^{-1}(\lambda^2)}{\Delta^{-1}(\lambda^1)+\Delta^{-1}(\lambda^2)+\Delta^{-1}(\lambda^3)+\Delta^{-1}(\lambda^4)} = \frac{2.08}{9.18} = 0.2265 \\
 P_s^3 &= \frac{\Delta^{-1}(\lambda^3)}{\Delta^{-1}(\lambda^1)+\Delta^{-1}(\lambda^2)+\Delta^{-1}(\lambda^3)+\Delta^{-1}(\lambda^4)} = \frac{2.03}{9.18} = 0.2211 \\
 P_s^4 &= \frac{\Delta^{-1}(\lambda^4)}{\Delta^{-1}(\lambda^1)+\Delta^{-1}(\lambda^2)+\Delta^{-1}(\lambda^3)+\Delta^{-1}(\lambda^4)} = \frac{1.92}{9.18} = 0.2091
 \end{aligned}$$

Con estas probabilidades:

$$\begin{aligned}
 k(D^1) &= \text{round}\left(\frac{\sum_{i=1}^4 \#(D^i)}{4} \cdot P_s^1\right) = \text{round}(5 \cdot 0.3431) = 2 \\
 k(D^2) &= \text{round}\left(\frac{\sum_{i=1}^4 \#(D^i)}{4} \cdot P_s^2\right) = \text{round}(5 \cdot 0.2265) = 1 \\
 k(D^3) &= \text{round}\left(\frac{\sum_{i=1}^4 \#(D^i)}{4} \cdot P_s^3\right) = \text{round}(5 \cdot 0.2211) = 1 \\
 k(D^4) &= \text{round}\left(\frac{\sum_{i=1}^4 \#(D^i)}{4} \cdot P_s^4\right) = \text{round}(5 \cdot 0.2091) = 1
 \end{aligned}$$

lo que indica que hay que obtener los dos documentos de mayor relevancia de D^1 , y el de mayor relevancia de D^2 , D^3 y D^4 :

$$\begin{aligned}
 D^1 &\equiv (d_1^1, rn_1^1) = (d_1^1, (H, -0.4)) \text{ y } (d_3^1, rn_3^1) = (d_3^1, (M, 0)) \\
 D^2 &\equiv (d_1^2, rn_1^2) = (d_1^2, (VH, 0.2)) \\
 D^3 &\equiv (d_1^3, rn_1^3) = (d_1^3, (M, -0.4)) \\
 D^4 &\equiv (d_1^4, rn_1^4) = (d_1^4, (VH, 0.2))
 \end{aligned}$$

Por tanto, la lista de documentos DV ordenados por relevancia RV que el agente de filtrado colaborativo recibe del agente de tareas, es la siguiente:

$$\begin{aligned} (d_1^V, r_1^V) &= (d_1^2, rn_1^2) = (d_1^2, (VH, 0.2)) \\ (d_2^V, r_2^V) &= (d_1^4, rn_1^4) = (d_1^4, (VH, 0.2)) \\ (d_3^V, r_3^V) &= (d_1^1, rn_1^1) = (d_1^1, (H, -0.4)) \\ (d_4^V, r_4^V) &= (d_3^1, rn_3^1) = (d_3^1, (M, 0)) \\ (d_5^V, r_5^V) &= (d_1^3, rn_1^3) = (d_1^3, (M, -0.4)) \end{aligned}$$

Ahora, el agente de filtrado colaborativo filtra esos documentos, volviendo a calcular la relevancia pero ahora teniendo en cuenta las recomendaciones que sobre ellos hayan realizado los usuarios que tengan un perfil similar. Supongamos que el SR colaborativo almacena, con respecto a los documentos recibidos, las recomendaciones mostradas en la tabla 4.3.

Documentos para el tópico "Web Mining"	Valoraciones de los usuarios con el perfil P_i	Recomendación $RC_j^{P_i} \in S_3$
d_1^V	(Id3,M), (Id5,MLH)	(MLH,-0.5)
d_2^V	---	---
d_3^V	(Id1,H), (Id3,MLL), (Id4,M)	(M,0.33)
d_4^V	(Id2,M), (Id4,H)	(MLH,0)
d_5^V	(Id5,H)	(H,0)

Tabla 4.3: Recomendaciones existentes de usuarios con un perfil similar.

En la tabla anterior vemos que para el documento d_2^V no existe ninguna valoración almacenada, y para el resto de documentos, las recomendaciones $RC_j^{P_i}$ se obtienen de la siguiente forma mediante el uso del operador \bar{x}^e (definición 3.13 del capítulo 3):

$$RC_1^{\mathcal{P}_i} = \bar{x}^e[(M, 0), (MLH, 0)] = \Delta(\frac{4+5}{2}) = (MLH, -0.5)$$

$$RC_3^{\mathcal{P}_i} = \bar{x}^e[(H, 0), (MLL, 0), (M, 0)] = \Delta(\frac{6+3+4}{3}) = (M, 0.33)$$

$$RC_4^{\mathcal{P}_i} = \bar{x}^e[(M, 0), (H, 0)] = \Delta(\frac{4+6}{2}) = (MLH, 0)$$

$$RC_5^{\mathcal{P}_i} = \bar{x}^e[(H, 0)] = \Delta(\frac{6}{1}) = (H, 0)$$

Recordemos que aunque los $r_j^V \in S_4$ y los $RC_j^{\mathcal{P}_i} \in S_3$, en este ejemplo ambos conjuntos equivalen al tercer nivel de la jerarquía, es decir a S^9 , por lo que no es necesario aplicar ninguna función de transformación; en caso contrario habría que aplicar la función $TF_{S_4}^{S_3}(RC_j^{\mathcal{P}_i})$. Entonces, usando estas recomendaciones el agente de filtrado colaborativo vuelve a calcular la relevancia de los documentos DV , usando el operador de media ponderada \bar{x}^w (definición 3.14 del capítulo anterior) con el vector de pesos [0.6,0.4]:

$$r_1^{NV} = \bar{x}^w[(VH, 0.2), (MLH, -0.5)] = \Delta(7.2 \cdot 0.6 + 4.5 \cdot 0.4) = \Delta(6.12) = (H, 0.12)$$

$$r_2^{NV} = (VH, 0.2) = r_2^V$$

$$r_3^{NV} = \bar{x}^w[(H, -0.4), (M, 0.33)] = \Delta(5.6 \cdot 0.6 + 4.33 \cdot 0.4) = \Delta(5.092) = (MLH, 0.092)$$

$$r_4^{NV} = \bar{x}^w[(M, 0), (MLH, 0)] = \Delta(4 \cdot 0.6 + 5 \cdot 0.4) = \Delta(4.4) = (M, 0.4)$$

$$r_5^{NV} = \bar{x}^w[(M, -0.4), (H, 0)] = \Delta(3.6 \cdot 0.6 + 6 \cdot 0.4) = \Delta(4.56) = (MLH, -0.44).$$

donde vemos que el valor de la relevancia para el documento 2 no cambia, puesto que para él no había ninguna valoración. Así, el listado de documentos DW ordenados por relevancia RW que recibe el agente de interface es el siguiente:

$$(d_1^W, r_1^W) = (d_2^V, r_2^{NV}) = (d_2^V, (VH, 0.2))$$

$$(d_2^W, r_2^W) = (d_1^V, r_1^{NV}) = (d_1^V, (H, 0.12))$$

$$(d_3^W, r_3^W) = (d_3^V, r_3^{NV}) = (d_3^V, (MLH, 0.092))$$

$$(d_4^W, r_4^W) = (d_5^V, r_5^{NV}) = (d_5^V, (MLH, -0.44))$$

$$(d_5^W, r_5^W) = (d_4^V, r_4^{NV}) = (d_4^V, (M, 0.4))$$

En el último paso del algoritmo, el agente de interface filtra este último listado y devuelve al usuario de internet los documentos más relevantes (Df, Rf) . Por ejemplo, si el número de documentos a devolver está establecido en $K = 3$, el sistema devolvería al usuario estos enlaces:

$$\begin{aligned}
 (d_1^f, r_1^f) &= (d_1^A, r_1^W) = \\
 &(\text{http} : // \text{www.ncsa.uiuc.edu/SDG/IT94/Proceedings/Agents/spetka/spetka.html}, (VH, 0.2)) \\
 (d_2^f, r_2^f) &= (d_1^2, r_2^W) = \\
 &(\text{http} : // \text{lcs.www.media.mit.edu/people/lieber/Lieberary/Letizia/Letizia.html}, (H, 0.12)) \\
 (d_3^f, r_3^f) &= (d_1^1, r_3^W) = (\text{http} : // \text{phonebk.duke.edu/clients/tnfagent.html}, (MLH, 0.092))
 \end{aligned}$$

A continuación, tendría lugar la fase de realimentación en la que el usuario de internet introduce sus valoraciones sobre los documentos recuperados como etiquetas lingüísticas pertenecientes a S_3 . Esta actividad se realiza fácilmente y consiste en que cuando el usuario proporciona sus valoraciones, el SR colaborativo las almacena y vuelve a calcular las recomendaciones para estos documentos, agregando mediante el uso del operador \bar{x}^e las nuevas valoraciones con las ya existentes proporcionadas por los usuarios con un perfil similar, tal y como hemos visto un poco más arriba.

Capítulo 5

Un Sistema de Recomendaciones sobre Recursos de Investigación: SIRE2IN

En este capítulo proponemos el diseño e implementación de SIRE2IN, un SR de convocatorias sobre recursos de investigación, dirigido a los investigadores de la Universidad de Granada y empresas del entorno productivo, con el objetivo de ayudarles en sus procesos de acceso a información personalizada sobre recursos de investigación (convocatorias, proyectos, noticias, eventos, etc.) en sus áreas de interés. El sistema ha sido diseñado integrando herramientas de filtrado basadas en contenidos, junto con un modelado lingüístico difuso multi-granular.

5.1. Introducción

La Oficina de Transferencia de Resultados de Investigación (OTRI) de la Universidad de Granada, está integrada en el Vicerrectorado de Investigación y Tercer Ciclo. Es responsable de promover y gestionar las actividades de generación de conocimiento y colaboración técnica y científica, fomentando la interrelación entre investigadores de la universidad y el mundo empresarial, así como su participación

en diversos programas diseñados para llevar a cabo actividades de I+D+i (Investigación, Desarrollo e Innovación). Uno de sus objetivos fundamentales es fomentar y ayudar en la generación de conocimiento, así como en su difusión y transferencia a la sociedad, con el propósito de identificar las demandas y necesidades del entorno productivo. En la figura 5.1 podemos ver una representación gráfica de esta misión.

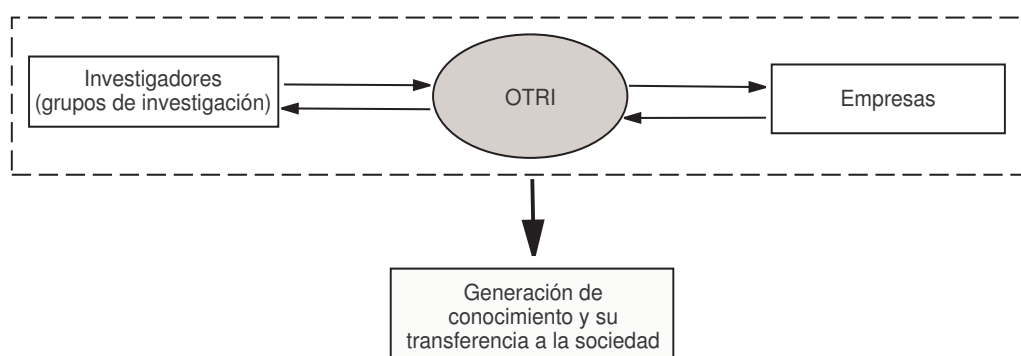


Figura 5.1: Principal misión de la OTRI.

Este objetivo se concreta a través de las siguientes funciones:

- Fomentar entre los profesores e investigadores de la universidad que deben empezar a considerar que el conocimiento que ellos generan, fruto de sus investigaciones, debe ser transferido para su aplicación en el entorno que les rodea.
- Respaldar a los grupos de investigación para que desarrollen, dentro de la sociedad, sus actividades de producción, difusión y transferencia de resultados científicos, técnicos y artísticos.
- Gestionar las relaciones entre los grupos de investigación y su entorno, proporcionándoles conjuntamente instrumentos, mecanismos y métodos efi-

cientes que les permitan incrementar la calidad y el número de los proyectos llevados a cabo y los recursos usados por los grupos, así como el uso social y económico de sus resultados.

- Realizar la difusión de la oferta científica, tecnológica y artística de los grupos de investigación, para hacerla llegar al resto de agentes que intervienen en los procesos de I+D+i.

De cara a la realización de dichas funciones, la OTRI cuenta con una cartera de servicios de entre los que destacan los siguientes:

- Fuente de información: proporcionar información sobre convocatorias, eventos, noticias, etc.
- Orientación para la financiación de la I+D y la transferencia de tecnología.
- Ayuda en la preparación de ofertas.
- Negociación y elaboración de contratos y convenios con empresas.
- Elaboración, difusión y promoción de la oferta científico tecnológica.
- Evaluación, protección y transferencia de derechos de propiedad intelectual e industrial.
- Ayudas para la creación de nuevas empresas.
- Gestión de contactos.

Para llevar a cabo las funciones descritas y gestionar su cartera de servicios,

la OTRI se compone de un equipo de Técnicos en Transferencia de Tecnología. Cada uno de los técnicos gestiona una o varias tareas específicas, pero todos ellos tienen la tarea común de la difusión de recursos de investigación (noticias, eventos, convocatorias, congresos, cursos, etc.) entre los investigadores de la universidad y las empresas del entorno. Ello implica la selección por parte de los técnicos, de los investigadores o empresas a los que más les podría interesar cada una de las convocatorias que vayan surgiendo. En este proceso de selección encontramos un primer problema: la gran cantidad de información y recursos de investigación a los que los técnicos de OTRI pueden acceder, está provocando que no sean capaces de difundir la información a los usuarios adecuados (sean investigadores o empresas) de forma rápida y sencilla. Como hemos visto en capítulos previos, los *sistemas de FI* o SR son herramientas cuyo objetivo es evaluar y filtrar la gran cantidad de información disponible en un ámbito concreto y así asistir a los usuarios en sus procesos de acceso a la información. Por tanto, este tipo de sistemas pueden facilitar la labor de difusión personalizada que tienen que realizar los técnicos de OTRI y de este modo ser sistemas de acceso a la información sobre investigación para los asociados a la OTRI.

Por otro lado, encontramos el problema de la gran variedad de representaciones y evaluaciones de la información, lo que puede ser aún más acusado cuando los usuarios forman parte del proceso como es el caso que nos ocupa. Por tanto, para mejorar las posibilidades de representación de la información y la interfaz con los usuarios, se hace necesaria una mayor flexibilidad en el procesamiento de la información. Para conseguir dicha flexibilidad, proponemos el uso de técnicas de *modelado lingüístico difuso* que representan y gestionan información flexible por medio de etiquetas lingüísticas.

En este capítulo proponemos el diseño e implementación de un Sistema de Recomendaciones sobre Recursos de Investigación, denominado SIRE2IN. El sistema, que está dirigido a investigadores de la Universidad de Granada y empresas del entorno, les permite obtener información personalizada sobre recursos de investigación en el ámbito de sus áreas de interés y recomienda sobre posibilidades de colaboración con otros investigadores o empresas de cara a poder acceder conjuntamente a las diversas convocatorias que puedan ir apareciendo. Para ello, hemos diseñado el sistema a partir de las técnicas de filtrado de información y del modelado lingüístico difuso. Concretamente, integramos el uso de una de las dos principales técnicas de filtrado, el filtrado basado en contenidos, con una de las técnicas de modelado lingüístico difuso, el llamado modelado lingüístico difuso multi-granular, que nos permite representar y gestionar la información lingüística de distinta naturaleza que pueden usar los investigadores o empresas para describir sus perfiles. Para probar la funcionalidad del sistema, hemos implementado una primera versión, a la que se puede acceder en esta dirección: *<http://otri.ugr.es/sire2in>*.

El capítulo se estructura de la siguiente forma. En la Sección 2, presentamos la arquitectura del sistema. Posteriormente, en la Sección 3 analizamos las estructuras de datos necesarias para la gestión de la información. En la Sección 4 nos centramos en el funcionamiento del sistema, describiendo cómo desarrolla su actividad. La Sección 5 presenta el proceso de desarrollo donde tratamos tanto los detalles de su implementación, como la descripción de la interfaz de usuario. Por último, en la Sección 6 debatiremos sobre las mejoras futuras que pensamos incorporar.

5.2. Arquitectura del Sistema

En la introducción hemos estado describiendo el problema al que pretendemos dar solución con el diseño e implementación de nuestro sistema. De esa descripción podemos abstraer la arquitectura de SIRE2IN, representada en la figura 5.2. Como podemos observar en la figura, el sistema consta de tres componentes principales:

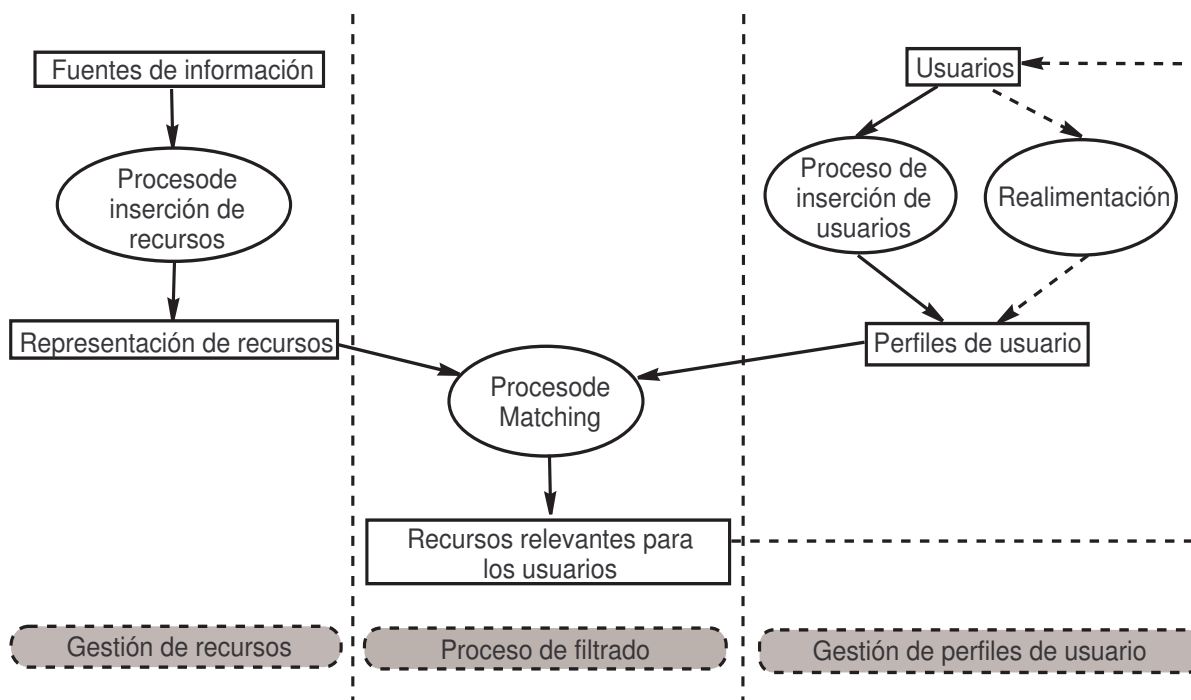


Figura 5.2: Arquitectura de SIRE2IN.

- **Gestión de recursos.** Este módulo es el responsable de la gestión de las fuentes de información de las que los técnicos de OTRI reciben toda la información sobre recursos de investigación, y obtienen una representación interna de estos ítems. Ejemplos de fuentes de información son Internet, boletines

de noticias especializados, listas de distribución, convocatorias públicas o privadas, proyectos, foros, etc. Para gestionar los recursos los representamos de acuerdo con su ámbito, y para obtener dicha representación usamos la *clasificación de códigos UNESCO para la Ciencia y de la Tecnología* [107]. Esta clasificación se compone de 3 niveles, de forma que cada uno es un refinamiento del nivel previo. El primer nivel abarca aspectos generales y se usan 2 dígitos para representar cada una de las 24 categorías que lo componen. Cada categoría general del primer nivel incluye algunas disciplinas, codificadas por 4 dígitos en el segundo nivel, que se compone de un total de 248 disciplinas. El tercer nivel se compone de subdisciplinas que representan las actividades que se desarrollan en cada una de las disciplinas del segundo nivel; estas subdisciplinas son en total 2253 y están codificadas por 6 dígitos. Nosotros vamos a trabajar únicamente con los dos primeros niveles, porque consideramos que el tercer nivel proporciona un nivel de discriminación demasiado elevado y ello podría dificultar la interacción con los usuarios. Además de estos códigos UNESCO, para cada recurso almacenamos otro tipo de información que veremos más adelante y que también será tomada en cuenta en el proceso de filtrado.

- **Gestión de perfiles de usuario.** Los usuarios pueden ser investigadores de la Universidad o bien personas de contacto de las empresas del entorno. En ambos casos el sistema trabaja con una representación interna de los usuarios en la que se incluyen sus preferencias o necesidades, es decir, que el sistema representa a cada usuario por medio de un perfil de usuario. Para definir los perfiles de los usuarios vamos a usar información general sobre cada uno de ellos y además tendremos en cuenta los tópicos o aspectos que
-

más les interesan. Para ello también usamos la clasificación UNESCO [107], por lo que cada usuario tendrá asociada una lista de códigos UNESCO con aquellas disciplinas que mejor representan sus intereses o necesidades de información; en definitiva, almacenamos el conjunto de códigos UNESCO que mejor definen su actividad investigadora. Inicialmente el sistema asigna a cada usuario los códigos UNESCO que tenga asignados por defecto el grupo de investigación o la empresa a la que pertenezca, y posteriormente podrá actualizar su perfil por medio de la fase de realimentación en la que expresa especificaciones explícitas sobre sus preferencias.

- **Proceso de filtrado.** Es el proceso principal por el que el sistema filtra la información entrante para enviarla únicamente a los usuarios a los que más les pueda interesar. Al tratarse de un sistema de filtrado basado en contenidos, filtra la información realizando un proceso de cálculo de similitud entre los términos usados en la representación de los perfiles de usuario y los términos usados en la representación de los recursos. Posteriormente estudiaremos en detalle este proceso teniendo en cuenta las estructuras de datos definidas para el desarrollo del sistema.

5.3. Estructuras de Datos

En este apartado se van a definir las estructuras de datos que necesitamos para representar toda la información relativa a usuarios y recursos de investigación.

Vamos a trabajar con recursos (convocatorias, noticias, eventos, etc.) y con perfiles

de usuario, por lo que los usuarios no van a introducir consultas explícitas y puntuales sino que trabajamos con necesidades de los mismos a más largo plazo. Es por ello, que para diseñar nuestro sistema hemos optado por un **modelo vectorial** según el cual tanto el ámbito de cada recurso como los temas de interés de cada usuario son representados por medio de un vector, es decir una lista ordenada de términos que definen dichas áreas.

En primer lugar vamos a considerar la información necesaria para caracterizar un recurso:

- titular: titular o encabezado que da título al recurso,
 - resumen: resumen acerca del recurso,
 - texto: texto principal,
 - fecha: fecha de publicación,
 - origen: fuente de la información,
 - enlace: enlace a la página Web donde se puede encontrar toda la información sobre el recurso. Es un dato necesario puesto que cuando el sistema envía a los usuarios información sobre un recurso, no envía toda la información sino un email con la información resumida y el enlace al recurso para poder ampliarla,
 - tipo de recurso: clasifica el recurso según sea una convocatoria de proyecto, una noticia, un curso, un evento o un congreso,
 - objetivo: este campo indica el tipo de usuarios a los que principalmente está dirigido el recurso, es decir, si está dirigido a investigadores, empresas o ambos,
-

- rango de cantidades: en el caso de convocatorias de proyectos, indica las cantidades mínima y máxima que el usuario puede solicitar,
- ámbito: es el campo de aplicación del recurso y por tanto la principal información que se tendrá en cuenta a la hora de realizar el proceso de filtrado. Para representar el ámbito de los recursos, vamos a usar el modelo vectorial, donde para cada recurso almacenamos un vector VR . Para construir este vector, usamos la clasificación UNESCO [107] y concretamente trabajamos con el segundo nivel, que recordemos se compone de 248 disciplinas. Por lo tanto, el vector que representa el ámbito de los recursos tendrá 248 posiciones, una para cada disciplina de la clasificación UNESCO de nivel 2, y en cada posición se almacena el grado de importancia del código UNESCO representado en esa posición para el ámbito del recurso. Por ejemplo, si observamos el listado de códigos UNESCO de nivel 2 [107], vemos que el primero que aparece es el código *1101* correspondiente a *aplicaciones de la lógica*, por lo que el grado de importancia de dicho código para el ámbito del recurso i se almacenará en la primera posición del vector, $VR_i[1]$.

Para caracterizar a un usuario, debemos distinguir si se trata de un investigador o de una persona de contacto de alguna empresa, que en definitiva representa a la empresa. Para los investigadores, vamos a trabajar con la siguiente información:

- usuario: identidad del usuario que se usará para el acceso identificado al sistema,
 - password: clave de acceso al sistema,
 - dni: documento nacional de identidad,
-

- nombre y apellidos,
 - departamento y centro donde trabaja,
 - dirección,
 - teléfono fijo, teléfono móvil y fax,
 - web: si el usuario dispone de página Web,
 - email: dato necesario puesto que el mail es el medio que vamos a usar para enviar a los usuarios la información sobre los recursos y las recomendaciones,
 - grupo de investigación: grupo de investigación al que pertenece el usuario. Seguimos la nomenclatura del Plan Andaluz de Investigación (PAI) según la cual cada grupo se identifica mediante 6 dígitos, los 3 primeros son letras e identifican el área de investigación, y los 3 últimos son números que identifican al grupo en ese área,
 - preferencias de colaboración: si el investigador desea colaborar con investigadores de otros grupos, con empresas, con ambos o sencillamente no desea colaborar con otra gente,
 - preferencias sobre recursos: el investigador especifica el tipo de recursos que desea, por ejemplo si sólo quiere que se le envíe información sobre convocatorias de proyectos, o noticias, etc.
 - rango de cantidades: en el caso de convocatorias de proyectos, se permite que el usuario especifique las cantidades mínima y máxima en las que estaría interesado solicitar,
-

- temas de interés: es la información sobre las áreas de investigación de interés para los usuarios, por lo que constituye un aspecto clave en el proceso de filtrado. Para representar los temas de interés también vamos a usar el modelo vectorial, almacenando un vector VU para cada usuario. Para construir estos vectores, usamos la clasificación UNESCO de nivel 2 [107], que dispone de 248 disciplinas. Por tanto, el vector tendrá 248 posiciones y en cada una se almacena el grado de importancia de la disciplina representada en esa posición para el área de investigación de interés para el investigador. Por ejemplo, para un usuario x en la primera posición del vector $VU_x[1]$ se almacena el grado de importancia del código 1101 correspondiente a la disciplina *aplicaciones de la lógica* con respecto a los temas de interés del usuario.

De igual forma, si el usuario es una persona de contacto de alguna empresa, el sistema trabaja con la siguiente información:

- usuario: identidad del usuario que se usará para el acceso identificado al sistema,
 - password: clave de acceso al sistema,
 - dni: documento nacional de identidad,
 - nombre y apellidos,
 - empresa: nombre de la empresa para la que trabaja,
 - dirección,
 - teléfono fijo, teléfono móvil y fax,
 - web: página Web de la empresa,
-

- email: dato necesario puesto que el mail es el medio que vamos a usar para enviar a los usuarios la información sobre los recursos y las recomendaciones,
- preferencias de colaboración: indica si la empresa desea colaborar con otras empresas, con investigadores de la universidad, con ambos o sencillamente no desea colaborar con nadie,
- preferencias sobre recursos: la empresa especifica el tipo de recursos que desea, por ejemplo si sólo quiere que se le envíe información sobre convocatorias de proyectos, o noticias, etc.
- rango de cantidades: en el caso de convocatorias de proyectos, se permite que el usuario especifique las cantidades mínima y máxima en las que estaría interesado solicitar,
- temas de interés: es la información sobre las áreas de investigación de interés para la empresa. Al igual que en los casos anteriores, para representar los temas de interés también vamos a usar el modelo vectorial, almacenando un vector VU para cada usuario. Para construir estos vectores, usamos la clasificación UNESCO de nivel 2 [107]. Por tanto, el vector tendrá 248 posiciones y en cada una se almacena el grado de importancia de la disciplina representada en esa posición para el área de investigación de interés para la empresa. Por ejemplo, para un usuario x en la primera posición del vector $VU_x[1]$ se almacena el grado de importancia del código 1101 correspondiente a la disciplina *aplicaciones de la lógica* con respecto a los temas de interés de dicho usuario.

Con toda esta información, tanto de investigadores como de empresas, el sistema establece los perfiles de usuario.

Por otro lado vamos a definir cómo se va a representar y gestionar la información lingüística. Como ya hemos comentado, para facilitar la interacción con los usuarios, vamos a trabajar con información lingüística y concretamente para conseguir una mayor flexibilidad en los procesos de comunicación del sistema, hemos adoptado el modelado lingüístico difuso multi-granular. Recordemos que en este modelo, para representar la información lingüística, en lugar de usar siempre el mismo conjunto de etiquetas, se usan conjuntos de etiquetas distintos (S_1, S_2, \dots) lo cual es muy útil en casos en los que se tienen que realizar distintas valoraciones de la información como es el caso que nos ocupa. Estos conjuntos de etiquetas S_i van a ser seleccionados de los conjuntos de etiquetas que conforman una determinada jerarquía lingüística (ver capítulo 3), LH , que habrá que definir previamente, es decir, $S_i \in LH$. Recordemos que el número de conjuntos de etiquetas distintos que podemos usar, está limitado por el número de niveles de la jerarquía LH , por lo que en ciertos casos los conjuntos de etiquetas S_i y S_j pueden estar asociados a un mismo conjunto de etiquetas de LH pero con diferentes interpretaciones, según el concepto que sea modelado. En nuestro sistema, distinguimos tres conceptos que pueden ser evaluados:

- **grado de importancia** (S_1) de cada código UNESCO con respecto al ámbito de cada recurso o respecto a las preferencias de los usuarios,
 - **grado de relevancia** (S_2) de un determinado recurso para un investigador o para una empresa,
 - **grado de compatibilidad** (S_3) entre usuarios, es decir, entre un investigador y una empresa, entre investigadores de distintos grupos y entre diferentes empresas.
-

En el diseño de nuestro sistema, partimos de la jerarquía lingüística que ya vimos en el capítulo 3, compuesta de tres niveles de 3, 5 y 9 etiquetas cada uno de ellos. Usamos el segundo nivel (5 etiquetas lingüísticas) para asignar grados de importancia ($S_1 = S^5$) y el tercer nivel (9 etiquetas lingüísticas) para asignar grados de relevancia ($S_2 = S^9$) y grados de compatibilidad ($S_3 = S^9$). Los términos lingüísticos de cada uno de los niveles de esta jerarquía son los siguientes:

- 1^{er} nivel: $S^3 = \{a_0 = Nulo = N, a_1 = Medio = M, a_2 = Total = T\}$.
- 2^o nivel: $S^5 = \{b_0 = Nulo = N, b_1 = Bajo = L, b_2 = Medio = M, b_3 = Alto = H, b_4 = Total = T\}$
- 3^{er} nivel: $S^9 = \{c_0 = Nulo = N, c_1 = Muy_Bajo = VL, c_2 = Bajo = L, c_3 = Algo_Bajo = MLL, c_4 = Medio = M, c_5 = Algo_Alto = MLH, c_6 = Alto = H, c_7 = Muy_Alto = VH, c_8 = Total = T\}$

Por lo tanto, para representar un recurso i , tendremos un vector que representa su ámbito:

$$VR_i = (VR_i[1], VR_i[2], \dots, VR_i[248]),$$

donde cada componente $VR_i[j] \in S_1, j = 1, \dots, 248$, almacena una etiqueta lingüística que indica el grado de importancia del código UNESCO de la posición j con respecto al recurso i . Esta información va a ser introducida por los técnicos de OTRI cada vez que vayan a insertar un nuevo recurso en el sistema.

Por otro lado, para representar los temas de interés usados para definir los perfiles de los usuarios, seguimos el mismo método, usando un vector VU para cada

usuario del sistema. Entonces para un determinado usuario x representamos sus temas de interés mediante un vector:

$$VU_x = (VU_x[1], VU_x[2], \dots, VU_x[248]),$$

donde cada componente $VU_x[y] \in S_1$, con $y = 1, \dots, 248$, almacena una etiqueta lingüística que indica el grado de importancia de la disciplina representada por el código UNESCO de la posición y con respecto al usuario x . Esta información es inicialmente asignada por el sistema según los códigos UNESCO que tenga asignados por defecto el grupo de investigación o la empresa a la que pertenece cada usuario, pero posteriormente cada uno de ellos podrá editarla según sus necesidades y preferencias.

5.4. Actividad del Sistema

Una vez que ya hemos establecido las estructuras básicas sobre las que se sustenta nuestro sistema, podemos pasar ya a describir el funcionamiento del mismo. La actividad del sistema puede ser descrita brevemente en tres pasos:

1. Un técnico de OTRI encuentra en las fuentes de información un determinado recurso (noticia, convocatoria, etc.) y lo inserta en el sistema.
 2. Entonces el sistema realiza el proceso de filtrado para seleccionar aquellos usuarios a los que más les puede interesar la información recién insertada en el sistema y automáticamente les envía un mail con la información, el grado de relevancia que se ha calculado para que el usuario pueda comprobar que
-

el recurso es relevante para él y recomendaciones sobre posibilidades de colaboración.

3. Una vez que los usuarios reciben la información, y basándose en su grado de satisfacción con la información recibida, pueden seleccionar el tipo de información que desean recibir en un futuro. Para ello tienen que actualizar sus perfiles, accediendo al sistema y modificando sus preferencias.

Para desarrollar dicha actividad, necesitamos primeramente haber introducido en el sistema los usuarios con los que contemos inicialmente. Posteriormente podremos ir añadiendo nuevos usuarios.

Por lo tanto, el funcionamiento del sistema contempla la realización de los siguientes procesos:

- Proceso de inserción de usuarios.
- Proceso de inserción de recursos.
- Proceso de filtrado.
- Proceso de realimentación (feedback).

5.4.1. Proceso de Inserción de Usuarios

Como ya hemos comentado, los usuarios del sistema van a ser tanto los investigadores de la universidad, como personas de contacto de empresas que estén interesadas en recibir información relevante y personalizada sobre recursos de investigación. Suelen ser empresas habituadas o interesadas en mantener algún tipo

de relación con grupos de investigación de la universidad, como puede ser a través de la firma de contratos o convenios de investigación.

De cara a reunir información sobre los usuarios y así poder establecer sus perfiles, aspecto éste fundamental en cualquier SR, hemos adoptado un enfoque híbrido entre el explícito y el implícito. Según este enfoque, cuando se inserta un nuevo usuario, el sistema le asigna automáticamente y de forma implícita (sin intervención del usuario) un perfil según la actividad del grupo de investigación o de la empresa a la que pertenezca, y posteriormente el usuario podrá actualizar dicho perfil mediante la inserción en el sistema de información explícita. De esta forma, al insertar un nuevo usuario, automáticamente tendrá un perfil de manera que el sistema empezará a enviarle información acorde con dicho perfil. Conforme el usuario vaya recibiendo información, podrá ir actualizando su perfil según el grado de satisfacción de la información que va recibiendo, y así irá adaptando su perfil de forma adecuada para únicamente recibir la información que realmente desea e ir adaptándose a los cambios que vayan surgiendo en cuanto a nuevas preferencias o necesidades de información se refiere.

Entonces, al insertar un nuevo usuario en el sistema, introducimos toda la información disponible sobre el mismo (nombre y apellidos, datos de contacto, etc.) y definimos sus temas de interés de forma implícita. En el caso de que el usuario sea un investigador, el sistema usa la clasificación en grupos de investigación del Plan Andaluz de Investigación (PAI), según la cual cada investigador pertenece a un grupo de investigación y en la que cada grupo además tiene asignados una serie de códigos UNESCO definidos en función de sus principales líneas de investigación. De esta forma, y como los perfiles de usuario se definen en función

de códigos UNESCO de nivel 2, al insertar un nuevo investigador se le asignan automáticamente los códigos UNESCO de nivel 2 del grupo al que pertenece dicho investigador, junto con grado de importancia $Total (b_4 \in S_1)$. En el caso de que el usuario pertenezca a una empresa, el procedimiento es similar, pero serán los técnicos de OTRI quiénes asignen manualmente estos códigos UNESCO en función de la actividad de la empresa a la que pertenezca el usuario.

Posteriormente los usuarios pueden actualizar sus perfiles siempre que lo deseen, accediendo al sistema y editando cualquier información sobre sí mismos, especialmente la referida a las preferencias y temas de interés. En concreto, para modificar sus temas de interés deben editar los códigos UNESCO que tienen asignados o bien las etiquetas lingüísticas que indican su grado de importancia.

De esta forma SIRE2IN define y actualiza los perfiles de usuario que usará en el proceso de filtrado, cada vez que se inserta en el sistema un nuevo recurso.

Ejemplo 5.1. Inserción de un usuario.

En este ejemplo vamos a ver cómo se inserta un nuevo usuario. Al técnico se le muestra una pantalla con un formulario en el que introduce la información que tenga disponible sobre el usuario junto con su identificador ID (normalmente su e-mail) y una clave de acceso necesaria para acceder al sistema. Supongamos que el usuario pertenece a un grupo de investigación que trabaja en Ciencias de la Nutrición y que por tanto tiene asignado el código UNESCO 3206 - *Ciencias de la Nutrición*. Este código, además, ocupa la posición 100 de la lista de códigos UNESCO de nivel 2 por lo que será almacenado en la posición 100 del vector de temas de interés. Por lo tanto, el sistema asigna al usuario ID el código 3206 con

grado de importancia $Total(b_4 \in S_1)$, de forma que el perfil de dicho usuario queda representado mediante un vector de temas de interés VU que tiene los siguientes valores:

$$VU_{\mathcal{ID}}[x] = b_4, \text{ si } x = 100$$
$$VU_{\mathcal{ID}}[x] = b_0, \text{ en otro caso.}$$

Posteriormente, cuando el usuario acceda por primera vez al sistema introducirá sus preferencias sobre posibilidades de colaboración, tipos de recursos deseados y rango de cantidades en que está interesado.

5.4.2. Proceso de Inserción de Recursos

Este proceso es llevado a cabo por los técnicos de OTRI cada vez que reciben información sobre un nuevo recurso, que puede ser una noticia, una convocatoria, un evento, un curso, etc. y desean difundir dicha información. El técnico inserta dicho recurso en el sistema y automáticamente se envía a los usuarios más apropiados, que pudieran estar interesados, la información básica junto con un enlace a la fuente de información para obtener más detalles; además, también se envía el grado de relevancia que se ha obtenido y recomendaciones sobre las posibilidades de colaboración con otros investigadores o empresas.

Como ya vimos en la sección anterior, el sistema almacena información general sobre los recursos y su ámbito. Dicho ámbito va a estar representado mediante un vector de códigos UNESCO por lo que al insertar un nuevo recurso en el sistema, el técnico debe decidir qué códigos UNESCO le va a asignar y qué grado de importancia va a tener cada uno. Para ello el técnico tiene que asignar junto

con cada código una etiqueta lingüística, perteneciente al conjunto S_1 .

Por lo tanto, cuando un técnico va a insertar un nuevo recurso primeramente introduce toda la información que tenga disponible, es decir, titular, resumen, texto detallado, fecha, fuente de información, enlace al recurso, tipo de recurso, objetivo y rango de cantidades, y a continuación valora los grados de importancia de cada código UNESCO de nivel 2 con respecto al ámbito del recurso. Para ello, el sistema muestra una lista desplegable de códigos UNESCO de nivel 2 de los que el técnico va seleccionando los que considere más oportunos y a su vez les va asignando una etiqueta lingüística $b_i \in S_1$, con $i = 0, \dots, \#(S_1)$ que refleje su grado de importancia. Para facilitar la labor del técnico, este paso se divide en dos, mostrando antes la clasificación UNESCO de nivel 1 y a continuación únicamente se muestran los códigos de nivel 2 correspondientes al del nivel 1 que previamente haya seleccionado el técnico. Por último, se pueden continuar añadiendo nuevos códigos UNESCO o bien finalizar la inserción del recurso.

Una vez que el técnico ha completado estos pasos, el sistema construye una representación interna del nuevo recurso, que como ya vimos consiste en un vector de etiquetas lingüísticas correspondientes a cada uno de los códigos UNESCO de nivel 2.

Ejemplo 5.2. Inserción de un recurso.

Supongamos que un técnico de OTRI recibe información sobre una determinada convocatoria i consistente en un congreso sobre Ciencias de la Nutrición. Entonces, el técnico inserta este recurso en el sistema, introduciendo toda la información de la que dispone y seleccionando de la lista de códigos UNESCO de nivel 2

aquellos a los que considera pertenece la convocatoria junto con sus grados de importancia. En este ejemplo, el técnico podría seleccionar los códigos *3206 - Ciencias de la Nutrición* con un grado de importancia *Total* ($b_4 \in S_1$) y *3309 - Tecnología de los Alimentos* con un grado de importancia *Alto* ($b_3 \in S_1$). Una vez que el técnico inserta esta información, el sistema construye el vector VR_i que va a usar internamente como representación del ámbito de la convocatoria, y que va a tener los siguientes valores:

$$VR_i[j] = b_4, \text{ si } j = 100$$

$$VR_i[j] = b_3, \text{ si } j = 118$$

$$VR_i[j] = b_0, \text{ en otro caso.}$$

teniendo en cuenta que los códigos *3206* y *3309* están en las posiciones 100 y 118 de la lista de códigos UNESCO de nivel 2, por lo que son almacenados en $VR_i[100]$ y $VR_i[118]$ respectivamente.

5.4.3. Proceso de Filtrado

En esta fase se realiza el filtrado de la información de forma que ésta sea entregada únicamente a los usuarios a los que más les pueda interesar, lo que se consigue mediante un proceso de cálculo de similaridad entre los términos usados en la representación de los perfiles de usuario y los términos usados en la representación de los recursos.

Según vimos en la sección anterior, tanto para representar los recursos como los perfiles de usuario vamos a seguir el modelo vectorial [62]. Este modelo usa medidas sofisticadas tales como la Distancia Euclídea o la Medida del Coseno para

realizar el proceso de cálculo de similaridad. Concretamente, para el diseño de nuestro sistema hemos optado por el uso de la medida del coseno que describimos a continuación.

La **Medida del Coseno** es una medida angular para calcular similitud entre vectores, definida por el coseno del ángulo formado entre los vectores que representan un documento y la consulta del usuario. En nuestro caso, al no trabajar con consultas explícitas, está definido por el coseno del ángulo formado entre el vector que representa el ámbito de un recurso VR y el vector que representa los temas de interés del usuario VU . Esta definición se formaliza matemáticamente de la siguiente manera:

$$\sigma(VR, VU) = \frac{\sum_{k=1}^n (r_k \times u_k)}{\sqrt{\sum_{k=1}^n (r_k)^2} \times \sqrt{\sum_{k=1}^n (u_k)^2}}$$

donde n es el número de términos usado para definir los vectores (en nuestro caso 248, el número de códigos UNESCO de nivel 2), r_k es el valor del término k en el vector del recurso VR y u_k es su valor en el vector VU que representa los temas de interés del usuario. En términos matemáticos, se trata del producto interno entre los vectores de los recursos y de las preferencias de los usuarios, normalizado según sus longitudes. Usando esta medida podemos obtener un rango que va desde el valor 1 para los casos de similaridad más alta hasta 0 en los que la similaridad sea nula. Esta medida del coseno también se puede aplicar de la misma manera para calcular la similaridad entre usuarios entre sí o recursos entre sí.

Las medidas angulares se caracterizan por representar una vista del espacio de los documentos desde un punto fijo, el origen. Además, una medida angular no

considera la distancia de cada documento respecto del origen, sino que únicamente tiene en cuenta su dirección. Por tanto, dos documentos que siguen el mismo vector desde el origen serán juzgados como idénticos, a pesar que estuvieran alejados en el espacio de los documentos. Esto significa que un anuncio de un párrafo y un artículo más detallado y extenso sobre un mismo asunto, podrían ser considerados igualmente relevantes para un usuario a pesar de que uno esté mucho más detallado que el otro.

Ejemplo 5.3. Medida del coseno.

Supongamos que tenemos tres recursos representados por los mismos dos términos mediante los siguientes vectores:

$$VR_1 = \langle 1, 3 \rangle,$$

$$VR_2 = \langle 100, 300 \rangle, \text{ y}$$

$$VR_3 = \langle 3, 1 \rangle .$$

Aplicando la medida del coseno, $\sigma(VR_1, VR_2) = 1.0$ y $\sigma(VR_1, VR_3) = 0.6$, según lo cual R_2 y R_1 son más similares que R_3 y R_1 . Esto se debe a que en R_1 y R_2 los dos términos tienen la misma importancia relativa, es decir, que conceptualmente están en la misma temática.

Siguiendo este enfoque, cada vez que se inserta un nuevo recurso i en el sistema, primeramente se construye el vector que va a representar el ámbito del nuevo recurso VR_i y a continuación se calcula la medida del coseno entre VR_i y todos los vectores que representan los temas de interés de los usuarios, VU_j , con $j = 1, \dots, m$ donde m es el número de usuarios registrados en el sistema, para de esta

forma encontrar los usuarios más apropiados a los que enviar información sobre el nuevo recurso. Si $\sigma(VR_i, VU_j) \geq \alpha$, entonces el sistema selecciona al usuario j . Previamente hemos tenido que definir el valor de umbral α que usamos para discriminar entre información relevante e irrelevante. En nuestro primer diseño hemos establecido que $\alpha = 0.5$, pero ya decimos que se trata de un valor que se puede ir ajustando según los resultados que se vayan obteniendo. Las preferencias de los usuarios sobre tipos de recursos y rango de cantidades, son tenidas en cuenta para considerarlos o no en el proceso de selección. Por otra parte, las preferencias de colaboración son usadas para clasificar a los usuarios seleccionados en dos conjuntos, los que no desean colaborar \mathcal{U}_S y los que están dispuestos a colaborar con otros \mathcal{U}_C .

En este punto del proceso, el sistema ha seleccionado una serie de usuarios, los ha dividido en dos conjuntos y para cada uno de ellos tiene un valor $\sigma(VR_i, VU_j) \geq \alpha$. Usamos dicho valor para calcular el grado de relevancia del recurso i para el usuario j , para lo cual tendremos que expresar el valor $\sigma(VR_i, VU_j)$ como una etiqueta lingüística del conjunto S_2 aplicando la función de transformación $TF_{S_2}^{S_1}$ definida en el capítulo 3 (definición 3.17). Entonces el sistema envía a los usuarios del conjunto \mathcal{U}_S la información del recurso y el grado de relevancia que se ha calculado expresado mediante una etiqueta lingüística.

Para los usuarios que admiten algún tipo de colaboración con otros, incluidos en el conjunto \mathcal{U}_C , queda calcular dichas posibilidades de colaboración. Para ello, entre cada dos usuarios $x, y \in \mathcal{U}_C$:

- Analizar si los usuarios son investigadores o alguien de una empresa para tenerlos o no en cuenta según las preferencias de cada usuario. Por ejemplo, un investigador podría querer colaborar únicamente con investigadores de otros grupos, pero no colaborar con empresas.
- Si cuadran las preferencias de ambos, calcular la similaridad entre ellos, mediante la medida del coseno $\sigma(VU_x, VU_y)$.
- Obtener el grado de compatibilidad entre x e y , para lo cual expresamos $\sigma(VU_x, VU_y)$ como una etiqueta lingüística de S_3 .

Por último, el sistema envía a los usuarios del conjunto \mathcal{U}_C la información del recurso, su grado de relevancia expresado mediante una etiqueta lingüística y recomendaciones sobre las posibilidades de colaboración en las que se indica con quién colaborar y el grado de compatibilidad, también expresado mediante una etiqueta lingüística para ayudarle a aclararse con la recomendación suministrada por el sistema.

En la figura 5.3 presentamos un esquema de cómo se lleva a cabo todo el proceso.

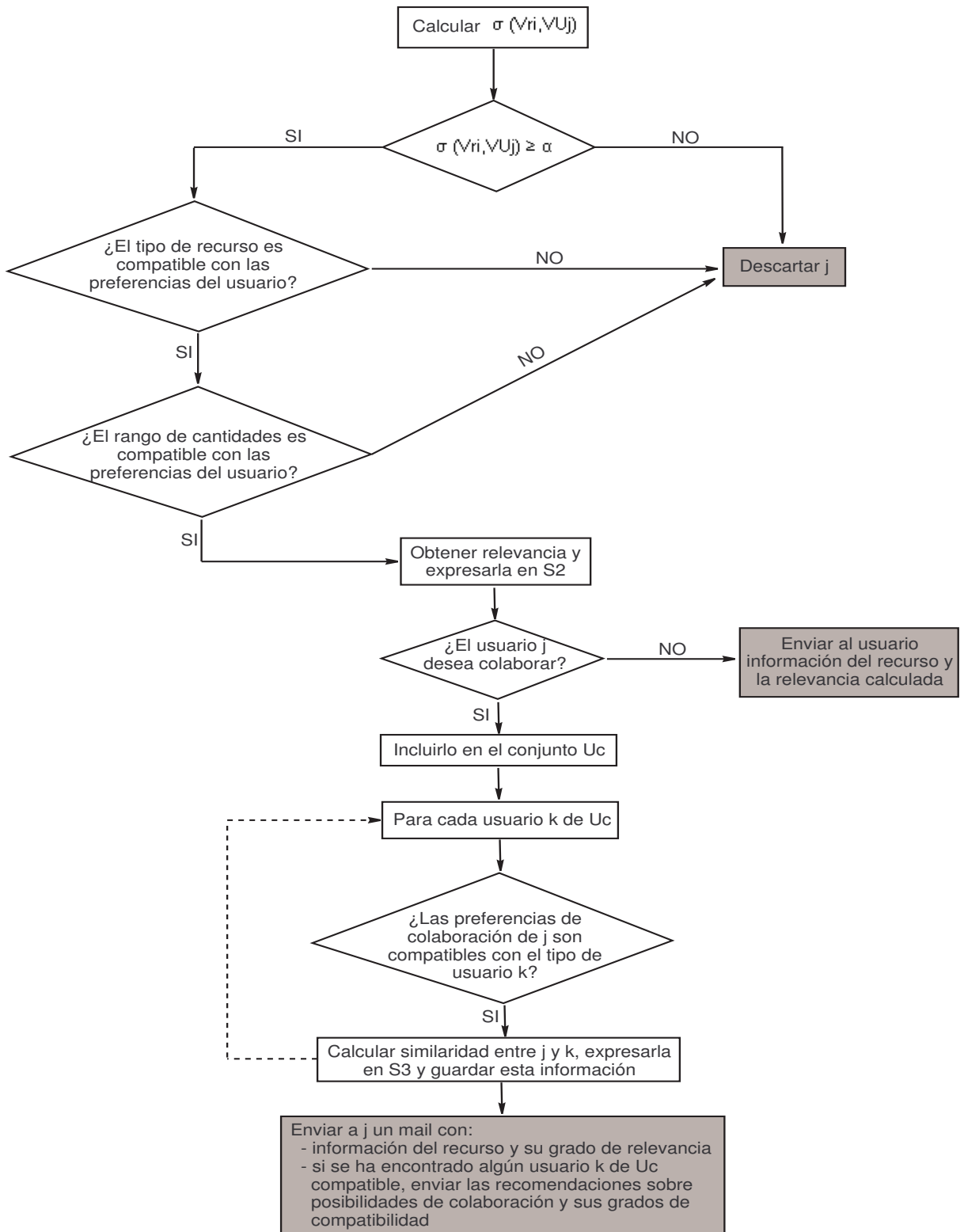


Figura 5.3: Esquema del proceso de filtrado para un usuario j .

5.4.4. Proceso de Realimentación de los Usuarios

Esta fase abarca la actividad desarrollada por el sistema de filtrado una vez que los usuarios comienzan a hacer uso de la información que ha sido enviada por el sistema. Como ya hemos dicho, los perfiles de los usuarios representan necesidades o intereses de información de los mismos a largo plazo, por lo que una propiedad deseada es que los perfiles se puedan ir adaptando puesto que las necesidades o preferencias de los usuarios van a ir cambiando a lo largo del tiempo. Por ese motivo, el sistema permite a los usuarios actualizar sus perfiles según los intereses de cada uno y así mejorar el proceso de filtrado. En SIRE2IN este proceso de realimentación es desarrollado de la siguiente forma:

- El usuario accede al sistema introduciendo su ID y su password.
 - Ahora el usuario puede actualizar su perfil mediante las siguientes operaciones:
 - editar sus preferencias de colaboración,
 - editar sus preferencias sobre el tipo de recursos deseados,
 - editar sus preferencias sobre el rango de cantidades que desea solicitar en el caso de convocatorias de proyectos,
 - editar sus temas de interés:
 - añadiendo un nuevo código UNESCO a su perfil con su correspondiente grado de importancia, es decir, asignando a dicho código una etiqueta lingüística $b_i \in S_1$,
-

- eliminando un código UNESCO que tuviera ya asignado y del que ya no desea obtener más información,
- modificando los grados de importancia de los códigos UNESCO que ya tiene asignados, asignándoles nuevas etiquetas lingüísticas $b_i \in S_1$.

Ejemplo 5.4. Realimentación del usuario.

Retomando el ejemplo 5.1, supongamos ahora que el usuario \mathcal{ID} desea actualizar su perfil porque considera que también debería pertenecer a la categoría 3309 - *Tecnología de los Alimentos*, por lo que debe añadir dicho código a su perfil y además asignarle un grado de importancia. Por ejemplo, consideremos que el usuario se asigna un grado de importancia *Alto* ($b_3 \in S_1$) a dicho código, que recordemos ocupa la posición 118 del listado de códigos UNESCO de nivel 2. Después de este paso, el usuario \mathcal{ID} tendría un nuevo perfil en el que su vector de temas de interés quedaría con los siguientes valores:

$$VU_{\mathcal{ID}}[x] = b_4, \text{ si } x = 100$$

$$VU_{\mathcal{ID}}[x] = b_3, \text{ if } x = 118$$

$$VU_{\mathcal{ID}}[x] = b_0, \text{ en otro caso.}$$

5.5. Desarrollo del Sistema

5.5.1. Implementación

Dada la gran funcionalidad que aporta un sistema de este tipo en cualquier organización o empresa y ante la necesidad real que se nos planteó en el ámbito de la OTRI de la Universidad de Granada, decidimos implantarlo. Para ello, nos pusimos a desarrollar e implementar una primera versión del sistema.

Desde el inicio nos planteamos que SIRE2IN funcionase desde el punto de vista de una aplicación Web trabajando mediante una arquitectura cliente-servidor, de manera que se pueda acceder a ella desde cualquier puesto y únicamente necesitamos un navegador Web. Ello implica añadir los correspondientes módulos de acceso identificado al sistema.

Hemos implementado esta primera versión del sistema usando el servidor propio de la OTRI. El sistema operativo instalado en el servidor es Windows 2003 Server, por lo que el servidor de aplicaciones con el que trabajamos es Internet Information Server. Para el almacenamiento de la información usamos una base de datos Microsoft Access alojada en el servidor. El lenguaje utilizado es ASP, lanzando consultas SQL para comunicarnos con la base de datos.

Una vez realizadas y superadas las pruebas de funcionamiento a nivel interno, la idea es iniciar una fase de pruebas en la que únicamente intervendrán los directores de departamento (o quién ellos designen) para posteriormente ampliar al resto de investigadores. A continuación procederemos a realizar una nueva carga en la base de datos, para incluir ya a todos los investigadores y personas de contacto de empresas a los que les pueda interesar darse de alta en el sistema.

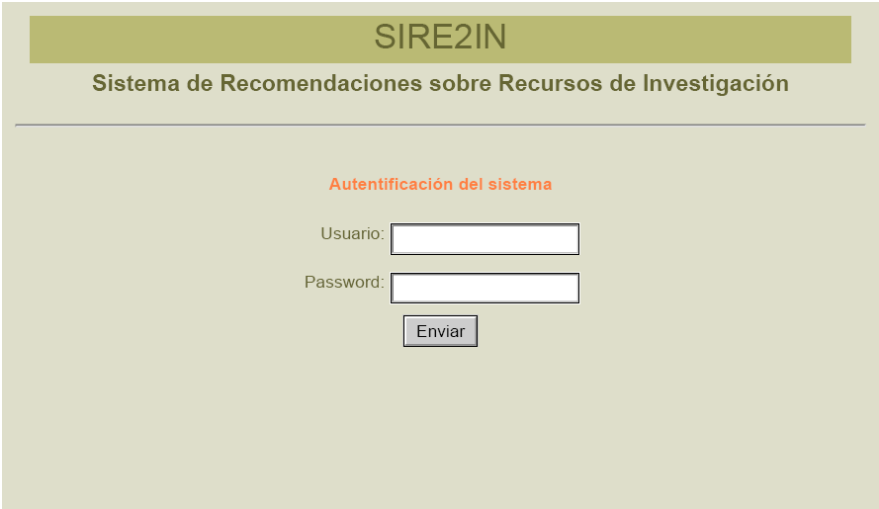
Actualmente estamos en una fase de introducir mejoras al sistema por lo que

hemos aprovechado para migrarlo a un entorno de software libre, implantándolo en máquinas que tienen instalado el sistema operativo Linux. Para ello usamos el servidor Apache, MySQL como sistema de gestión de bases de datos y PHP como lenguaje para la programación de las páginas Web.

5.5.2. Descripción de la Aplicación

La idea de este apartado es ofrecer una visión general de la aplicación, desde el punto de vista de su interfaz con el usuario. Comentar que aunque aquí describimos la primera versión, en la nueva versión en la que actualmente trabajamos, la interfaz apenas cambiará pues únicamente nos centramos en mejoras internas de funcionamiento que nos permitan mejorar el proceso de filtrado. Podemos acceder al sistema a través de la siguiente dirección: <http://otri.ugr.es/sire2in>, donde se nos mostrará una pantalla (ver figura 5.4) en la que se nos pide que nos registremos como usuario, introduciendo el nombre de usuario y password que previamente nos habrá proporcionado un administrador del sistema. Introducimos nuestro usuario y password y accedemos a un menú distinto en función del tipo de usuario, que recordemos puede ser: suscriptor (investigador o persona de contacto de una empresa), técnico de OTRI o administrador del sistema. Suponiendo que tuviésemos acceso total como administrador del sistema, nos aparecería un menú con todas las opciones posibles, tal y como se muestra en la figura 5.5.

A continuación describimos los distintos tipos de acceso, así como las posibles acciones a realizar en cada uno de ellos.



SIRE2IN
Sistema de Recomendaciones sobre Recursos de Investigación

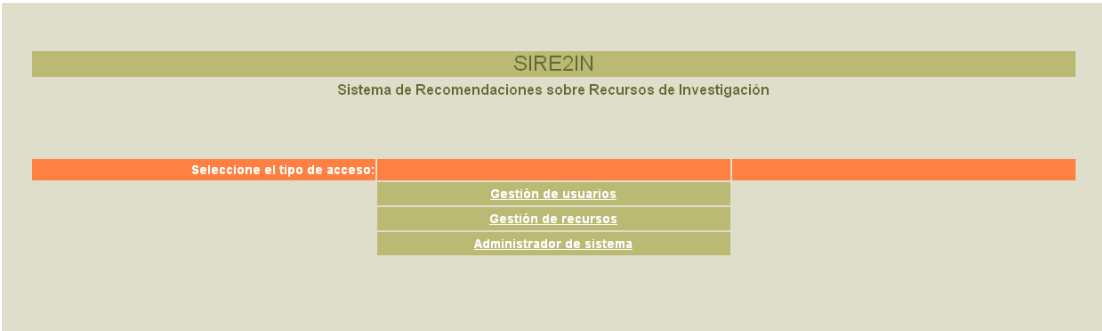
Autenticación del sistema

Usuario:

Password:

Enviar

Figura 5.4: Pantalla de autenticación de SIRE2IN.



SIRE2IN
Sistema de Recomendaciones sobre Recursos de Investigación

Seleccione el tipo de acceso:

- Gestión de usuarios
- Gestión de recursos
- Administrador de sistema

Figura 5.5: Menú principal, accediendo como administrador.

Suscriptor

Menú de acceso para los investigadores y empresas que estén registrados en el sistema. El usuario podrá ver sus datos personales, así como su perfil y si lo desea podrá modificar ambos para ajustarlos según sus necesidades o preferencias de información. En la figura 5.6 podemos ver la pantalla correspondiente a este tipo de acceso.

Técnico de OTRI

Opción a la que acceden los técnicos de OTRI y que les permite gestionar

SIRE2IN
Sistema de Recomendaciones sobre Recursos de Investigación

[Volver](#)

Datos del suscriptor					
DNI					
Nombre	Carlos Porcel Gallego				
Grupo de investigación	TIC186				
Departamento	Ciencias de la Computación e Inteligencia Artificial				
Centro	E.T.S. Ingeniería Informática				
Dirección	Periodista Daniel Saucedo Aranda s/n, 18071, Granada				
Tfn. fijo	958244258				
Tfn. móvil					
Fâx					
Email	cporcel@gmail.com				
Preferencias de colaboración					
Tipo de recursos					
Rango cantidades					
Temas de interés	<table border="1"> <thead> <tr> <th>Códigos UNESCO asignados al usuario</th> <th>Grado de importancia</th> </tr> </thead> <tbody> <tr> <td>1203</td> <td>total</td> </tr> </tbody> </table>	Códigos UNESCO asignados al usuario	Grado de importancia	1203	total
Códigos UNESCO asignados al usuario	Grado de importancia				
1203	total				

[Editar datos personales](#)
 [Editar preferencias](#)
 [Editar temas de interés](#)

Figura 5.6: Acceso como suscriptor.

los contenidos del sistema en lo que se refiere a recursos de investigación a los que deben dar difusión. Se accede a un listado de recursos ordenados por fechas, desde el que podrán acceder a información más detallada de cada recurso pinchando con el ratón sobre el titular que se desee. También se podrán realizar búsquedas, añadir nuevos recursos o bien modificar o eliminar alguno de los ya existentes. Cada vez que se añade un nuevo recurso o se modifican los datos de alguno ya existente, se debe realizar el proceso de filtrado (pulsando el botón "Filtrar información") para enviar la nueva información a los usuarios a los que más les pueda interesar. En la figura 5.7 podemos ver la pantalla a la que accedemos como técnicos de OTRI.

Administrador del sistema

Módulo dirigido exclusivamente a los encargados de la administración del sistema, que tienen un control total sobre todos los contenidos del mismo, tanto relativos a usuarios como a recursos. Simplemente indicar que desde



The screenshot shows the SIRE2IN web interface. At the top, there is a header with the title 'SIRE2IN' and the subtitle 'Sistema de Recomendaciones sobre Recursos de Investigación'. Below the header, there are two links: 'Insertar recurso' and 'Búsqueda de recursos', and a 'Menú' link on the right. The main content area features a table with two rows of data. Each row has columns for 'Titular' and 'Fecha', followed by three buttons: 'Filtrar información', 'Editar', and 'Eliminar'.

Titular	Fecha			
Congreso nacional de Ciencias de la Nutrición	01/12/2005	Filtrar información	Editar	Eliminar
Convocatoria de proyectos en colaboración con empresas	07/12/2005	Filtrar información	Editar	Eliminar

Figura 5.7: Acceso como técnico de OTRI.

aquí se realizará toda la gestión relativa a los usuarios (añadir, modificar o eliminar usuarios, o simplemente modificar datos personales) para lo cual cuenta con una herramienta de búsqueda de usuarios por diversos campos que facilita el acceso a los datos de los mismos (ver la pantalla en figura 5.8). En este módulo se incluyen también los procedimientos de carga o de actualización masiva de información. La idea es ir añadiendo funcionalidades dentro de este módulo para que en una versión definitiva, desde aquí podamos realizar toda la gestión de la base de datos a través de formularios Web y sin necesidad de acceder físicamente a la base de datos alojada en el servidor.

5.6. Discusión

En este capítulo hemos presentado un sistema de recomendaciones personalizado, cuyo propósito es ayudar a los técnicos de transferencia de tecnología en sus labores de difusión de información sobre recursos de investigación únicamente a aquéllos (investigadores o empresas) que les pueda interesar según sus áreas de interés.

The screenshot shows a web interface for 'SIRE2IN Sistema de Recomendaciones sobre Recursos de Investigación'. At the top right, there is a 'Menú' link. The main content area features a search form titled 'Búsqueda de usuarios' with the instruction 'Introduzca algunos datos para realizar la búsqueda'. The form includes a dropdown menu for 'Tipo de usuario' currently set to 'Investigador', and text input fields for 'DNI', 'Apellidos', and 'Nombre'. A 'Búsqueda' button is located at the bottom right of the form.

Figura 5.8: Pantalla de búsqueda de usuarios.

Además, el sistema proporciona un valor añadido, puesto que por un lado envía a los usuarios un grado lingüístico de relevancia que ayuda a justificar el envío por mail de la información al usuario y por otro lado, recomienda a los usuarios otros investigadores o empresas con los que podría colaborar.

Sin embargo, consideramos que el sistema podría mejorar incorporando nuevas características. En concreto, destacamos las siguientes:

- En la versión actual, es el técnico de OTRI quien asigna manualmente los códigos UNESCO a los recursos para así definir su ámbito de aplicación. Una posible mejora sería incorporar un módulo que realice automáticamente esta asignación de códigos UNESCO, es decir, que el sistema establezca automáticamente el ámbito de un recurso teniendo en cuenta el contenido del mismo.
- Otra posible mejora sería considerar la incorporación de grados lingüísticos de importancia relativa entre diferentes términos. De esta forma, en la

representación de los tópicos de interés de los usuarios o del ámbito de los recursos, para cada código UNESCO el sistema valorará tanto el grado de importancia (que actúa como un valor de umbral) como el grado de importancia relativa con respecto a otros códigos.

- Otro aspecto importante sería considerar la incorporación en el sistema de un nuevo módulo de filtrado colaborativo y así obtendríamos un enfoque híbrido. Para ello, tendríamos que determinar el conjunto de usuarios cuyo perfil sea similar a uno dado sin tener en cuenta el contenido de los recursos. Entonces, cuando le enviamos la información sobre un recurso a un usuario (filtrado basado en contenidos), haríamos un seguimiento de las acciones de dicho usuario y en caso de que le parezca interesante la información recibida (por ejemplo si vemos que se interesa solicitando información más detallada o directamente solicita un proyecto), le enviaríamos dicha información al conjunto de usuarios que tengan un perfil similar, aunque hayan sido seleccionados previamente por el módulo de filtrado basado en contenidos, pero ahora se les indica que dicha información se les envía porque le ha resultado de interés a un usuario con un perfil similar.
-

Capítulo 6

Comentarios Finales

6.1. Conclusiones

El objetivo fundamental que nos ha guiado en esta memoria ha sido el de estudiar el diseño de sistemas de acceso a la información que combinen técnicas de RI tradicionales con las técnicas más actuales de FI, con el propósito de crear sistemas de acceso a la información que proporcionen a los usuarios información relevante y personalizada y con ello incrementar su satisfacción. Los sistemas de FI ayudan a los usuarios a evaluar y filtrar la gran cantidad de información disponible para que únicamente accedan a información relevante para ellos. En este sentido es de destacar la importancia de una adecuada caracterización de los perfiles de los usuarios, de cara a obtener una personalización adecuada y así mejorar la satisfacción de los usuarios en sus procesos de acceso a la información.

Hemos visto también que las técnicas de Soft Computing, y en particular la Lógica Difusa, es una de las técnicas de IA que más se están usando en el diseño de sistemas de acceso a la información, dados los buenos resultados que se obtienen

mediante su aplicación. En particular, el modelado lingüístico difuso [115, 117] está siendo utilizado para modelar la subjetividad y la incertidumbre existentes en las actividades de acceso a la información (p.e, en la estimación de la relevancia de un documento respecto al perfil de un usuario o en la construcción de dicho perfil que representa las necesidades o preferencias de información del usuario).

Atendiendo a estos aspectos, los resultados obtenidos en esta memoria pueden resumirse en los siguientes puntos:

- Hemos analizado y estudiado los sistemas de FI, destacando la funcionalidad de la inclusión de estos sistemas en distintos ámbitos de aplicación (empresas, organizaciones, centros de I+D, etc.) como herramientas útiles en la distribución del conocimiento entre sus integrantes.
 - Se ha estudiado, analizado y presentado el Modelado Lingüístico Difuso para el manejo de información lingüística, y propuesto su aplicación en el diseño de sistemas de FI.
 - Hemos propuesto un sistema multi-agente para el acceso a la información en la Web, basado en técnicas de filtrado y en un modelado lingüístico difuso multi-granular.
 - Se ha diseñado e implementando un sistema automático de difusión de recursos de investigación, basado en la aplicación conjunta de técnicas de filtrado basadas en contenidos junto con un modelado lingüístico difuso multi-granular.
-

Algunos de los desarrollos de esta memoria se encuentran ya publicados en [42, 43, 44, 47, 48, 49, 50].

6.2. Trabajos Futuros

En esta sección destacamos algunas de las líneas que consideramos más interesantes de cara a nuestros futuros trabajos en el desarrollo de sistemas de FI:

1. Una posible línea de investigación estaría relacionada con el estudio de aspectos relacionados con la credibilidad de las recomendaciones en los sistemas de FI colaborativos, para considerar o no la validez de dichas recomendaciones.
 2. Otra posible línea de investigación sería introducir métodos de evaluación de la calidad informativa de los documentos y de las Webs que los almacenan, para conseguir generar recomendaciones más completas. De esta forma se podrían realizar recomendaciones relevantes, personalizadas y además de calidad.
 3. Avanzar en el estudio del modelado lingüístico difuso de los sistemas de FI, usando información lingüística no balanceada tanto para representar los ítems como la información sobre las preferencias de los usuarios, así como las recomendaciones finales que se obtengan, de modo que la información se represente de una forma más natural y así mejorar la interacción entre el sistema y los usuarios.
 4. Profundizar en el estudio de las técnicas de construcción y actualización de los perfiles de los usuarios. En concreto, sería una mejora importante la
-

caracterización de los perfiles de usuario mediante la aplicación de técnicas automáticas.

5. Por último, otra línea especialmente interesante, es el diseño de nuevos sistemas de FI para aplicaciones en la Web o para ámbitos más concretos, como una determinada organización. En particular nos interesa el diseño de sistemas de FI para recomendar los documentos y las Webs en función de su calidad informativa.
-

Bibliografía

- [1] Amazon: online shopping. <http://www.amazon.com>
- [2] B. Arfi. Fuzzy decision making in politics: A linguistic fuzzy-set approach (LFSA). *Political Analysis*, 13 (1), 23-56, 2005.
- [3] R. Baeza-Yates. Information Retrieval in the Web: Beyond Current Search Engines. *International Journal of Approximate Reasoning*, 34(2-3), 97-104, 2003.
- [4] R. Baeza-Yates, B. Ribeiro-Neto. *Modern Information Retrieval*. Addison-Wesley, 1999.
- [5] G. Bafoutsou, G. Mentzas. Review and Functional Classification of Collaborative Systems. *International Journal of Information Management*, 22, 281-305, 2002.
- [6] N.J. Belkin, W.B. Croft. Information Filtering and Information Retrieval: Two Sides of the Same Coin?. *Communications of the ACM*, 35(12), 29-38, 1992.
- [7] P.P. Bonissone, K.S. Decker. Uncertainty in artificial intelligence. *Ch. Selecting Uncertainty Calculi and Granularity: an experiment in trading-off*

-
- precision an complexity*, L.H. Kanal and J.F. Lemmer. Eds. North-Holland, 217-247, 1986.
- [8] G. Boone. Concept features in RE:Agent, an intelligent email agent. *Proc. of Autonomous Agents*, 1998, 141-148.
- [9] G. Bordogna, M. Fedrizzi, G. Pasi. A Linguistic Modeling of Consensus for a Fuzzy Majority in Group Decision Making. *IEEE Transactions on Systems, Man and Cybernetics. Part A: systems and humans*, 27, n.1, 126-132, 1997.
- [10] G. Bordogna, G. Pasi. A Fuzzy Linguistic Approach Generalizing Boolean Information Retrieval: A Model and Its Evaluation. *J. of the American Society for Information Science*, 44, 70-82, 1993.
- [11] G. Bordogna, G. Pasi. An Ordinal Information Retrieval Model. *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, 9(1), 63-75, 2001.
- [12] G. Bordogna, G. Pasi, R.R. Yager. Soft approaches to distributed information retrieval. *International Journal of Approximate Reasoning*, 34, 105-120, 2003.
- [13] W. Brenner, R. Zarnekow, H. Witting. Intelligent Software Agent, Foundations and Applications. *Springer-Verlag*, Berlin Heidelberg, 1998.
- [14] M. Claypool, A. Gokhale, T. Miranda, P. Murnikov, D. Netes, M. Sartin. Combining Content-Based and Collaborative Filters in an Online Newspaper. *Proc. of the ACM SIGIR Workshop on Recommender Systems-Implementation and Evaluation*, 1999.
-

-
- [15] C.W. Cleverdon, E.M. Keen. Factors Determining the Performance of Indexing Systems, Vol. 2—Test Results. *ASLIB Cranfield Res. Proj.*, Cranfield, Bedford, England, 1966.
- [16] O. Cerdón, F. Herrera, I. Zvir. Linguistic modelling by hierarchical systems of linguistic rules. *IEEE Transactions on Fuzzy Systems*, 10 (1), 2-20, 2001.
- [17] O. Cerdón, E. Herrera-Viedma. Preface to the Special Issue on Soft Computing Applications to Intelligent Information Retrieval on the Internet. *International Journal of Approximate Reasoning*, 34(2-3), 89-95, 2003.
- [18] F. Crestani, G. Pasi. *Soft Computing in Information Retrieval: Techniques and Applications*. Physica-Verlag, New York, 2000.
- [19] F. Crestani, G. Pasi. Handling Vagueness, Subjectivity, and Imprecision in Information Access: An Introduction to the Special Issue. *Information Processing & Management*, 39(2), 161-165, 2003.
- [20] M. Chau, D. Zeng, H. Chen, M. Huang, D. Hendriawan. Design and evaluation of a multi-agent collaborative Web mining system. *Decision Support Systems* 35, 167-183, 2003.
- [21] H. Chen. Preface to the Special Issue: "Web Retrieval and Mining". *Decision Support Systems*, 35, 1-5, 2003.
- [22] R. Degani, G. Bortolan. The Problem of Linguistic Approximation in Clinical Decision Making. *Int. J. of Approximate Reasoning*, 2, 143-162, 1988.
- [23] M. Delgado, F. Herrera, E. Herrera-Viedma, L. Martínez. Combining numerical and linguistic information in group decision making. *Information Sciences*, 107, 177-194, 1998.
-

-
- [24] M. Delgado, F. Herrera, E. Herrera-Viedma, M.J. Martín-Bautista, M.A. Vila. Combining linguistic information in a distributed intelligent agent model for information gathering on the Internet. *P.P. Wang, Ed., Computing with Words*, (John Wiley & Son, 2001), 251-276, 2001.
- [25] M. Delgado, F. Herrera, E. Herrera-Viedma, M.J. Martín-Bautista, L. Martínez, M.A. Vila. A communication model based on the 2-tuple fuzzy linguistic representation for a distributed intelligent agent system on Internet. *Soft Computing*, 6, 320-328, 2002.
- [26] M. Delgado, J.L. Verdegay, M.A. Vila. On aggregation operations of linguistic labels. *Int. Journal of Intelligent Systems*, 8, 351-370, 1993.
- [27] B. Fazlollahi, R.M. Vahidov, R.A. Aliev. Multi-agent distributed intelligent system based on fuzzy decision making. *Int. J. of Intelligent Systems*, 15, 849-858, 2000.
- [28] J. Ferber, *Multi-Agent Systems: An Introduction to Distributed Artificial Intelligence*. Addison-Wesley Longman, New York, 1999.
- [29] Film Conseil. <http://fconseil.lip6.fr/>
- [30] J. Furner. On Recommending. *Journal of the American Society for Information Science and Technology*, 53(9), 747-763, 2002.
- [31] D. Goldberg, D. Nichols, B. M. Oki, D. Terry. Using Collaborative Filtering to Weave an Information Tapestry. *Communications of the ACM*, 35, 61-70, 1992.
- [32] N. Good, J.B. Schafer, J.A. Konstan, A. Borchers, B.M. Sarwar, J.L. Herlocker, J. Riedl. Combining collaborative filtering with personal agents for
-

- better recommendations. *Proc. of the Sixteenth National Conference on Artificial Intelligence*, 439-446, 1999.
- [33] Google. <http://www.google.com>
- [34] U. Hanani, B. Shapira, P. Shoval. Information Filtering: Overview of Issues, Research and Systems. *User Modeling and User-Adapted Interaction* 11, 203-259, 2001.
- [35] J.L. Herlocker. Understanding and Improving Automated Collaborative Filtering Systems. A thesis submitted to the faculty of the graduate school of the University of Minnesota by Jonathan Lee Herlocker, 2000.
- [36] J.L. Herlocker, J.A. Konstan, J. Riedl. Explaining Collaborative Filtering Recommendations. *ACM 2000 Conference on Computer-Supported Collaborative Work*, 241-250.
- [37] J.L. Herlocker, J.A. Konstan, J. Riedl. An Empirical Analysis of Design Choices in Neighborhood-Based Collaborative Filtering Algorithms. *Information Retrieval*, 5, 287-310, 2002.
- [38] E. Herrera-Viedma. Modeling the retrieval process of an information retrieval system using an ordinal fuzzy linguistic approach. *J. of the American Society for Information Science and Technology*, 52(6), 460-475, 2001.
- [39] E. Herrera-Viedma. An information retrieval system with ordinal linguistic weighted queries based on two weighting elements. *Int. J. of Uncertainty, Fuzziness and Knowledge-Based Systems*, 9, 77-88, 2001.
- [40] E. Herrera-Viedma, O. Cordon, M. Luque, A.G. López, A.M. Muñoz. A Model of Fuzzy Linguistic IRS Based on Multi-Granular Linguistic Infor-
-

- mation. *International Journal of Approximate Reasoning*, 34 (3), 221-239, 2003.
- [41] E. Herrera-Viedma, F. Herrera, L. Martínez, J.C. Herrera, A.G. López. Incorporating Filtering Techniques in a Fuzzy Linguistic Multi-Agent Model for Information Gathering on the Web. *Fuzzy Sets and Systems* 148 (1), 61-83, 2004.
- [42] E. Herrera-Viedma, A.G. López-Herrera, L. Hidalgo, C. Porcel. Improving the Computation of Relevance Degrees in a Linguistic Information Retrieval System. *I Congreso Español de Informática (CEDI 2005)*, Granada (Spain), 2005.
- [43] E. Herrera-Viedma, A.G. López-Herrera, C. Porcel. A New Model of Linguistic Weighted Information Retrieval System. *Joint 4th EUSFLAT & 11th LFA Conference (EUSFLAT-LFA 2005)*, Barcelona (Spain), 2005.
- [44] E. Herrera-Viedma, A.G. López-Herrera C. Porcel. Tuning the Matching Function for a Threshold Weighting Semantics in a Linguistic Information Retrieval System. *International Journal of Intelligent Systems*, 20 (9), 921-937, 2005.
- [45] E. Herrera-Viedma, L. Martínez, F. Mata, F. Chiclana. A Consensus Support System Model for Group Decision-making Problems with Multi-granular Linguistic Preference Relations. *IEEE Trans. on Fuzzy Systems*, 2005. To appear.
- [46] E. Herrera-Viedma, E. Peis. Evaluating the informative quality of documents in SGML-format using fuzzy linguistic techniques based on compu-
-

- ting with words. *Information Processing & Management*, 39(2), 195-213, 2003.
- [47] E. Herrera-Viedma, E. Peis, M.D. Olvera, C. Porcel. Revisión de los Sistemas de Recomendaciones para la Recuperación de Información. *VI Congreso ISKO-España*, 507-514, Salamanca (Spain), 2003.
- [48] E. Herrera-Viedma, C. Porcel, A.G. López, M.D. Olvera, K. Anaya. A Fuzzy Linguistic Multi-Agent Model for Information Gathering on the Web Based on Collaborative Filtering Techniques. *Lecture Notes in Artificial Intelligence 3034*, 3-12, 2004.
- [49] E. Herrera-Viedma, C. Porcel, L. Hidalgo. Sistemas de Recomendaciones: Herramientas para el Filtrado de Información en Internet. *Hipertext.net*, 2, 2004.
- [50] E. Herrera-Viedma, C. Porcel, F. Herrera, L. Martínez, A.G. López-Herrera. Techniques to Improve Multi-Agent Systems for Searching and Mining the Web. In *Intelligent Data Mining: Techniques and Applications*, Da Ruan, Gouqing Chen, Etienne E. Kerre, Geert Wets (Eds) Springer Publisher, 463-486, 2005.
- [51] F. Herrera, E. Herrera-Viedma. Aggregation operators for linguistic weighted information. *IEEE Trans. on Systems, Man and Cybernetics*, Part A: Systems, 27, 646-656, 1997.
- [52] F. Herrera, E. Herrera-Viedma, L. Martínez. A Fusion Approach for Managing Multi-Granularity Linguistic Term Sets in Decision Making. *Fuzzy Sets and Systems*, 114, 43-58, 2000.
-

-
- [53] F. Herrera, E. Herrera-Viedma, L. Martínez. An Information Retrieval System with Unbalanced Linguistic Information Based on the Linguistic 2-tuple Model. *8th International Conference on Information Processing and Management of Uncertainty in Knowledge-Bases Systems (IPMU'2002)*. Annecy (France), 23-29.
- [54] F. Herrera, E. Herrera-Viedma, L. Martínez, P.J. Sánchez. A Methodology for Generating the Semantics of Unbalanced Linguistic Term Sets. *9th International Conference on Fuzzy Theory and Technology*, Florida, 151-154, 2003.
- [55] F. Herrera, E. Herrera-Viedma, J.L. Verdegay. Direct approach processes in group decision making using linguistic OWA operators. *Fuzzy Sets and Systems*, 79, 175-190, 1996.
- [56] F. Herrera, L. Martínez. A 2-tuple fuzzy linguistic representation model for computing with words. *IEEE Transactions on Fuzzy Systems*, 8 (6), 746-752, 2000.
- [57] F. Herrera, L. Martínez. A model based on linguistic 2-tuples for dealing with multigranularity hierarchical linguistic contexts in multiexpert decision-making. *IEEE Transactions on Systems, Man and Cybernetics*. Part B: Cybernetics, 31(2), 227-234, 2001.
- [58] F. Herrera, L. Martínez. The 2-tuple linguistic computational model. Advantages of its linguistic description, accuracy and consistency. *Int. J. of Uncertainty, Fuzziness and Knowledge-Based Systems*, 9, 33-48, 2001.
- [59] N. Jennings, K. Sycara, M. Wooldridge. A roadmap of agent research and development. *Autonomous Agents and Multi-Agents Systems*, 1, 7-38, 1998.
-

-
- [60] M. Kobayashi, K. Takeda. Information retrieval on the web. *ACM Computing Surveys*, 32(2), 148-173, 2000.
- [61] J.A. Konstan, B. Miller, D. Maltz, J. Herlocker, L. Gordon, J. Riedl. GroupLens: applying collaborative filtering to Usenet news. *Communications of the ACM*, 40 (3), 77-87, 1997.
- [62] R.R. Korfhage. *Information Storage and Retrieval*. New York: Wiley Computer Publishing, 1997.
- [63] T. Kuflik, B. Shapira, P. Shoval. Stereotype-Based versus Personal-Based Filtering Rules in Information Filtering Systems. *Journal of the American Society for Information Science and Technology*, 54(3), 243-250, 2003.
- [64] T. Kuflik, P. Shoval. Generation of User Profiles for Information Filtering - Research Agenda. *Proceedings of the 23rd Annual International ACM SIGIR Conference on Research and Development Information Retrieval*, July 24-28 2000, Ahtens (Greece), 313-315.
- [65] K. Lang. NewsWeeder: learning to filter Netnews. *Proceeding of the 12th International Conference on Machine Learning*, San Francisco, CA, 1995.
- [66] S. Lawrence, C.L. Giles. Searching the World Wide Web. *Science*, 280(5360), 98- 100, 1998.
- [67] S. Lawrence, C.L. Giles. Searching the Web: General and Scientific Information Access. *IEEE Communications Magazine*, 37 (1), 116-122, 1999.
- [68] H. Lieberman. Personal assistants for the Web: A MIT perspective. *M.Klusch (Ed.) Intelligent Information Agents* (Springer-Verlag, 1999), 279-292.
-

- [69] S.A. Macskassy. New Techniques in Intelligent Information Filtering. A dissertation submitted by Sofus Attila Macskassy to the Graduate School - New Brunswick Rutgers, The State University of New Jersey, 2003.
- [70] P. Maes. Agents that reduce work and information overload. *Comm. of the ACM*, 37, 31-40, 1994.
- [71] P. Maes. Modeling adaptive autonomous agents. In: C.G. Langton, Ed., *Artificial Life: an overview* (MIT Press, 1995), 135-162.
- [72] G. Marchioni. *Information Seeking in Electronic Environments*. Cambridge University Press, 1995.
- [73] M.J. Martín-Bautista, H.L. Larsen, M.A. Vila. A Fuzzy Genetic Algorithm to an Adaptive Information Retrieval Agent. *J. of the American Society for Information Science*, 50 (9), 760-771, 1999.
- [74] G.A. Miller. The magical number seven or minus two: some limits on our capacity of processing information. *Psychological Rev.* 63, 81-97, 1956.
- [75] S. Mitra, H. L. Larsen. Special Issue on Web Mining using Soft Computing. *Fuzzy Sets and Systems*, 148 (1), 2004.
- [76] S. Miyamoto. *Fuzzy Sets in Information Retrieval and Cluster Analysis*. Kluwer Academic Publishers, 1990.
- [77] M. Montaner. Collaborative Recommender Agents Based On Case-Based Reasoning and Trust. Thesis presented by Miquel Montaner. Department of Electronics, Computer Science and Automatic Control. Universitat de Girona.
-

-
- [78] M. Montaner, B. López, J.L. de la Rosa. A Taxonomy of Recommender Agents on the Internet. *Artificial Intelligence Review* 19, 285-330, 2003.
- [79] A. Moukas, G. Zacharia, P. Maes. Amalthea and Histos: Multiagent systems for WWW sites and representation recommendations. *M. Klusch (Ed.) Intelligent Information Agents*, Springer-Verlag, 293-322, 1999.
- [80] Moviefinder.com. <http://www.moviefinder.com>
- [81] MusicStrands, Inc. <http://www.musicstrands.com>
- [82] MusicSurfer - UPF. <http://musicsurfer.iaa.upf.edu/>
- [83] D.W. Oard, J. Kim. Implicit Feedback for Recommender systems. *Proceedings of the AAAI Workshop on Recommender Systems (AAAI '98)* Madison, Wisconsin, 26-30 July 1998.
- [84] G. Pasi. Intelligent Information Retrieval: Some Research Trends. J.M. Benítez, O. Cordon, F. Hoffman, R. Roy. Eds. *Advances in Soft Computing. Engineering Design and Manufacturing* (Springer), 157-171, 2003.
- [85] P. Perny, J.D. Zucker. Collaborative Filtering Methods based on Fuzzy Preference Relations. *Proceedings of the conference EUROFUSE-SIC'99*, Budapest, 25-28.
- [86] P. Perny, J.D. Zucker. Preference-based Search and Machine Learning for Collaborative Filtering: the Film Conseil Movie Recommender System. *Information - Interaction - Intelligence* 13, vol. 1, n. 1, 2001.
- [87] C.J. Petrie. Agent-based engineering, the Web and Intelligence. *IEEE Expert*, 24-29, 1996.
-

-
- [88] L.M. Quiroga, J. Mostafa. An experiment in building profiles in information filtering: the role of context of user relevance feedback. *Information Processing and Management*, 38, 671-694, 2002.
- [89] *Recommender Systems*. Papers and Notes from the 2001 Workshop New Orleans, LA. In conjunction with ACM SIGIR Conference on Research and Development in Information Retrieval.
- [90] Reel.com: Your Connection to the Movies. <http://www.reel.com/>
- [91] P. Reisnick, H.R. Varian. Recommender Systems. *Special issue of Comm. of the ACM*, 40 (3), 56-59, 1997.
- [92] G. Salton, M.J. McGill. *An Introduction to Modern Information Retrieval*, McGraw- Hill, 1983.
- [93] B.M. Sarwar, G. Karypis, J.A. Konstan, J. Riedl. Analysis of Recommender Algorithms for E-Commerce. *ACM E-Commerce 2000 Conference*, 2000.
- [94] B.M. Sarwar, J.A. Konstan, A. Borchers, J. Herlocker, B. Miller, J. Riedl. Using Filtering Agents to Improve Prediction Quality in the GroupLens Research Collaborative Filtering System. *Proceedings of the ACM Conference on Computer Supported Cooperative Work (CSCW-98)*, 345-354, 1998.
- [95] J.B. Schafer. MetaLens: A Framework for Multi-source Recommendations. A thesis submitted to the faculty of the graduate school of the University of Minnesota by John Benjamin Schafer, 2001.
- [96] J.B. Schafer, J.A. Konstan, J. Riedl. Recommender Systems in E-Commerce. *ACM Conference on Electronic Commerce (EC-99)*, november 3-5, 1999.
-

-
- [97] J.B. Schafer, J.A. Konstan, J. Riedl. Electronic Commerce Recommender Applications. *Journal of Data Mining and Knowledge Discovery*, vol.5, 115-152, 2001.
- [98] J.B. Schafer, J.A. Konstan, J. Riedl. MetaRecommendation Systems: User-controlled Integration of Diverse Recommendations. *ACM Conference on Information and Knowledge Management (CIKM-02)*, november 5-7, 2002.
- [99] B. Shapira, U. Hanani, A. Raveh, P. Shoval. Information Filtering: A New Two-Phase Model Using Stereotypic User Profiling. *Journal of Intelligent Information Systems*, 8, 155-165, 1997.
- [100] B. Shapira, P. Shoval, U. Hanani. Hypertext browsing: a new model based on hypergraph dynamic construction using data analysis methods. *Proceedings of NGfTS-95, the Second International Workshop on Next Generation Information Technologies and Systems*. Naharia, Israel, 1995.
- [101] B. Shapira, P. Shoval, U. Hanani. Stereotypes in Information Filtering Systems. *Information Processing & Management*, 33 (3), 273-287, 1997.
- [102] B. Shapira, P. Shoval, U. Hanani. Experimentation with an information-filtering system that combines cognitive and sociological filtering integrated with user stereotypes. *Decision Support Systems*, 27, 5-24, 1999.
- [103] L.C. Smith. Artificial Intelligence and Information Retrieval. *Annual Review of Information Science and Technology*, 22, 41-77, 1987.
- [104] K. Sycara, A. Pannu, M. Williamson, D. Zeng. Distributed Intelligent Agents. *IEEE Expert*, 36-46, 1996.
-

-
- [105] V. Torra. Negation functions based semantics for ordered linguistic labels. *International Journal of Intelligent Systems*, 11, 975-988, 1996.
- [106] V. Torra. Aggregation of Linguistic Labels when Semantics is based on Antonyms. *International Journal of Intelligent Systems*, 16, 513-524, 2001.
- [107] Clasificación UNESCO. Ministerio de Educación y Ciencia.
<http://www.mec.es/ciencia/jsp/plantilla.jsp?area=proyectos/invest&id=53>
- [108] M. Wooldridge, N. Jennings. Intelligent agents: theory and practice. *The knowledge engineering review*, 10, 115-152, 1995.
- [109] R.R. Yager. On Ordered Weighted Averaging Aggregation Operators in Multicriteria Decision Making. *IEEE Trans. on Systems and Cybernetics*, 18 (1), 183-190, 1988.
- [110] R.R. Yager. Protocol for negotiations among multiple intelligent agents. *J. Kacprzyk, H. Nurmi and M. Fedrizzi Eds., Consensus Under Fuzziness* (Kluwer Academic Publishers), 165-174, 1996.
- [111] R.R. Yager. Intelligent agents for World Wide Web advertising decisions. *International J. of Intelligent Systems*, 12, 379-390, 1997.
- [112] R.R. Yager. Fusion of multi-agent preference orderings. *Fuzzy Sets and Systems*, 112, 1-12, 2001.
- [113] R.R. Yager. Fuzzy Logic Methods in Recommender Systems. *Fuzzy Sets and Systems*, 136 (2), 133-149, 2003.
- [114] Yahoo!. <http://www.yahoo.com>
- [115] L.A. Zadeh. Fuzzy Sets. *Information and Control*, 8(3), 338-353, 1965.
-

-
- [116] L.A. Zadeh. The concept of a linguistic variable and its applications to approximate reasoning. Part I. *Information Sciences*, 8, 199-249, 1975. Part II, *Information Sciences*, 8, 301-357, 1975. Part III, *Information Sciences*, 9, 43-80, 1975.
- [117] L.A. Zadeh. Fuzzy Logic, Neural Networks and Soft Computing. *Communications of the ACM*, 37(3), 77-84, 1994.
- [118] L.A. Zadeh. What is Soft Computing?. *Soft Computing*, 1(1), 1, 1997.
- [119] L.A. Zadeh, J. Kacprzyk. *Fuzzy Logic for the Management of Uncertainty*. John Wiley, New York, 1992.
-