Available at
www.ElsevierComputerScience.com
POWERED BY SCIENCE DIRECT®

INTERNATIONAL JOURNAL OF APPROXIMATE REASONING

ELSEVIER International Journal of Approximate Reasoning 34 (2003) 145–161
www.elsevier.com/locate/ijar

Rules and fuzzy rules in text: concept, extraction and usage

D.H. Kraft [a], M.J. Martín-Bautista [b,*], J. Chen [a], D. Sánchez [b]

[a] *Department of Computer Science, Louisiana State University, Baton Rouge, LA, USA*
[b] *Department of Computer Science and Artificial Intelligence, University of Granada, C/Periodista Daniel Saucedo Aranda, Granada 18071, Spain*

Received 1 January 2003; accepted 1 July 2003

Abstract

Several concepts and techniques have been imported from other disciplines such as Machine Learning and Artificial Intelligence to the field of textual data. In this paper, we focus on the concept of rule and the management of uncertainty in text applications. The different structures considered for the construction of the rules, the extraction of the knowledge base and the applications and usage of these rules are detailed. We include a review of the most relevant works of the different types of rules based on their representation and their application to most of the common tasks of Information Retrieval such as categorization, indexing and classification.

Keywords: Rules; Fuzzy logic; Information retrieval; Association rules

1. Introduction

The problem of providing intelligence to retrieval systems has been solved in different ways since the eighty decade, generally with techniques coming from

* Corresponding author. Fax: +34-9-58-243317.
E-mail addresses: kraft@bit.csc.lsu.edu (D.H. Kraft), mbautis@decsai.ugr.es (M.J. Martín-Bautista), jianhua@bit.csc.lsu.edu (J. Chen), daniel@decsai.ugr.es (D. Sánchez).

0888-613X/$ - see front matter © 2003 Elsevier Inc. All rights reserved.
doi:10.1016/j.ijar.2003.07.005

Artificial Intelligence and Machine Learning [20,29,42]. The vertiginous expansion of new technologies, specially the information access through the Internet, and the application of "intelligence" to improve the retrieval processes have propitiated the use of techniques from other disciplines such as Soft Computing [45] and Data Mining [17,23].

The lack of homogeneity in some imported concepts and techniques managed in different processes and even the difference of terms like Information Retrieval, Information Access, Text Mining,... [21] makes difficult the generalization of concepts which may be used with the same structures in each one of these fields but called and applied for different purpose. For instance, the concept of association rule is very close to Text Mining, but can also be used for query expansion, which is a typical problem of Information Retrieval.

One of these concepts utilized in several processes of documentary systems is the concept of rule, due to the fact that it is a very easy form for knowledge representation and it is compatible with human-expressed knowledge [3]. Other knowledge representation techniques such as frames have been also used for this purpose [41]. The construction of intelligent documentary systems via application of rule-base technology from Artificial Intelligence and Machine Learning is one of the first uses of the rule concept in the area of Information Retrieval [41]. Since this first application, different concepts have emerged related to rule-base technology. Classification rules, expert rules, heuristic rules, inference rules, production rules... are all terms used in applications of Machine Learning and Artificial Intelligence to text, sometimes without their real meaning or usage. We can make the first difference here. In Artificial Intelligence, predefined rules are applied to decide upon patterns and/or actions, while in Machine Learning, systems learn from previously seen pattern and/or actions.

1.1. Dealing with uncertainty

The essential point in Information Retrieval is that the user wants information about some topic, that is, has an information need. However, a problem arises with traditional information systems, since the user information needs are vague in nature. Fuzzy logic is a valid tool to deal with this vagueness. This fact has propitiated the extension of different interpretations and definition of rules in a fuzzy framework when dealing with Fuzzy Information Retrieval Systems [24,25].

Several aspects of the Information Retrieval task involve vagueness, which may be best dealt with by Fuzzy Logic approach. First, as we just mentioned, user's information needs are vague: a user may be interested to retrieve documents which are primarily about "Information Technology", in particular, about "Internet" and "Online Information Retrieval", and yet the interested

documents should also be somehow relevant to "Social Changes" and "Music". Thus, the formulation of the user's query may be a fuzzy one, in which the above-mentioned keywords are attached with various fuzzy weights. The second source of vagueness in Information Retrieval comes from the characterization of the documents as to their topical categories. A document may be strongly relevant to "Fuzzy Logic", and somewhat relevant to "Humanity" and "Politics". Finally, the matching of the user's query and the characterization of textual documents is also vaguely defined. It is hard in general to assert the exact degree of fit between the user's query and a document. Given the vagueness spread over the specification of user's information need, the document characterization, and the match between the two, it is quite natural to consider the application of Fuzzy Logic (in particular, rules and fuzzy rules) to handle textual Information Retrieval tasks [43].

The purpose of this paper is to review the different ways of definition, extraction and usage of rules and fuzzy rules in a text framework. This paper is organized as follows: in the next section, the concept of rule is defined as a generalization of the rule approaches and, a study of the representation of rules in a general way with special features in the text framework is given. The different forms of acquisition of rules are detailed in Section 3, and the forms of representation of the antecedent and consequent in the text case are given in Section 4. The usage of rules in text and different applications, specially in the field of Information Retrieval is explained in Section 5. Finally, concluding remarks are given in Section 6.

2. Concept of rule and rule representation

A rule can be defined as a form of knowledge representation which expresses an implication. Rules can be differentiated by their representation, their acquisition and usage. Several approaches combining different forms in each of these categories can be found in the literature, specially in Artificial Intelligence, where they take an important role with the construction of knowledge bases. We study the most relevant types of rules in the following.

2.1. Rule representation

2.1.1. Logical rules

The first type of rules to be considered here is *logical rules*. These rules have a clear syntax and semantics, which make them a good knowledge representation formalism in Artificial Intelligence. A logical rule is an implication of the form:

$$\langle logicalrule \rangle ::== \langle A \rangle \rightarrow \langle C \rangle$$

and can be read *if A then C*, where A (antecedent) and C (consequent) are well-formed formulae (wff) and ($\rightarrow$) being the material implication.

Logical rules can be used for inference due to their well-formed syntax and semantics to deal with logical formulae for reasoning and deriving new knowledge [32].

2.1.2. Production rules

The origin of the use of rules in Artificial Intelligence comes from the construction of expert systems to compensate, in some sense, the lack of action and temporality expression in logic. These expert systems have been also called production systems consisting of a database, a rule base and a rule interpreter. The rules used in these systems are called *production rules*, and represent general knowledge about the problem domain [19]. Each production rule has the same form that logical rules, but formally, the antecedent and consequent of a production rule can be described by *object–attribute-value* tuples (one tuple or a group of them in the antecedent and one in the consequent) as follows:

$$A ::== (\langle object\rangle, \langle attribute\rangle, \langle value\rangle)$$

$$C ::== (\langle object\rangle, \langle attribute\rangle, \langle value\rangle)$$

The instantiation of the values of the $\langle object\rangle$, $\langle attribute\rangle$ (which can be single or multi-valued) and $\langle value\rangle$ in the antecedent and consequent of the rules generates the different application of rules and their different names. For instance, in classification rules, the object of the antecedent is the example to classify, the attribute is the feature with a certain value to be consider. In the consequent, the object is specified by an attribute meaning the class with a certain value.

In the text framework, an example for this kind of rules may be the classification rules for boolean retrieval into two predefined classes, relevant and non-relevant, which may be defined as follows:

$$A ::== (\langle document\rangle, \langle query\rangle, \langle value\rangle)$$

$$C ::== (\langle document\rangle, \langle relevanceclass\rangle, \langle value\rangle)$$

where $\langle value\rangle$ in the condition of the rule is the matching value between the document and the query and the action part means the membership of the document to the relevant or non-relevant class.

Production rules are the result of a process of translation from *heuristic rules*, which are used to be given by the expert in natural language, and can be considered as an informal way to describe production rules. The process of translation from heuristic rules to production rules is not trivial, but necessary since experts give their knowledge in form of heuristic rules, but the system deals with production ones. However, since the rules do not reflect the relations among objects, additionally to the rule base, an object schema can be con-

structed to define interrelationships among the objects and among the objects and their attributes. Finally, we must point out that the definition of production rules does not include a specific semantics, but it is given by the inference method utilized for applying the rules [32].

Production rules and uncertainty. One of the main advantages of the use of production rules is the fact of considering an uncertainty degree associated to the rule. Their formal representation is given by

$$\langle productionrule\rangle ::== \langle antecedent\rangle \rightarrow \langle consequent\rangle[\langle UncertaintyDegree\rangle]$$

where the $\langle UncertaintyDegree\rangle$ can be expressed by a probability-based measure (for instance a certainty factor), linguistic scales of uncertainty, belief measures,... or a combination among them.

Rules with uncertain information. These rules have been traditionally called *fuzzy rules* to express that information appearing in antecedent and/or consequent of the rule has uncertain information [16]. These rules are production rules and can have associated an uncertainty degree as well.

2.1.3. Association and fuzzy association rules

Association rules are production rules where antecedent and consequent are items or group of items with a semantic of presence in a transaction. Given a database of transactions where each transaction is an itemset, we can extract association rules.

We have to distinguish clearly between association rule and co-occurrence: in an association rule, the presence of an item A in a transaction implies the presence of item B in the same transaction but the reciprocal does not have to happen necessarily. This is different from a co-occurrence between terms, which represents that the presence of A and B in the same transaction are reciprocal. However, the term *association* is sometimes used to mean co-occurrence, but do not have to be confused with an association rule. These rules can also have associated an uncertainty degree which is usually expressed by an uncertainty measure such as probability-based measures (conditional probability or certainty factor, for instance) and possibility measures [31].

Formally, let T be a set of transactions containing items of a set of items I. An association rule is defined as a link of the form $A \Rightarrow B$ such that $A, B \subset I$, and $A \cap B = \emptyset$, where A is the antecedent and B is the consequent of the rule, both of them being itemsets coming from a set of transactions T-set.

In a fuzzy framework, we can consider fuzzy association rules [1] where the rule holds in a FT-set (Fuzzy Transaction set). If we call $\widetilde{\Gamma}_A$ and $\widetilde{\Gamma}_B$ the fuzzy

[1] The name *fuzzy association rules* is used because they have implicit vagueness which arises from the origin of the rules (extracted from a set of fuzzy transactions) and/or from the accomplishment degree associated to the rule, for instance.

sets of transactions where A and B appear respectively, we can assert that the rule $A \Rightarrow B$ holds with total accuracy in FT when $\widetilde{I}_A \subseteq \widetilde{I}_B$ [15].

In the case of a text framework, from a collection of documents $D = \{d_1, \ldots, d_n\}$ we can obtain a set of terms $I = \{t_1, \ldots, t_s\}$ which is the union of the keywords for all the documents in the collection. The weights associated to these terms are represented by $W = \{w_1, \ldots, w_s\}$. Therefore, for each document d_i, $1 \leqslant i \leqslant n$, we consider an extended representation where a weight of 0 will be assigned to every term appearing in some of the documents of the collection but not in d_i. This representation of documents, terms and weights is similar to the one of the vector space model [38]. As we are dealing with this representation in a mining context, we call it *text transaction* [14]. An analogous extension to the one above can be done with fuzzy weights in $[0, 1]$ representing presence of terms in documents to obtain *fuzzy text transactions*.

2.1.4. Co-occurrences

The statistical nature of most of the processes to be carried out in text analysis has propitiated the search of associations among the elements [2] summarized in these statistical processes. In a general way, what is intended to get is an attribution of the same properties to elements that appear together or are related to other elements by some kind of association (lexical, semantic, co-occurrence).

Co-occurrences can be considered as relations between facts with a bi-implication ($A \leftrightarrow B$), which indeed represents the accomplishment of two rules ($A \rightarrow B$) and ($B \rightarrow A$). Since these correspondences represent a statistical dependence between A and B, some authors have called them *dependence rules* [40]. The degree of dependence between A and B can be expressed by a numerical value.

3. Acquisition of rules

By acquisition of rules we understand the origin of the rules. There exist two main sources of knowledge to be expressed by rules, namely from experts and from data.

Acquisition from experts. It is given by the knowledge that the user expresses sometimes in form of heuristic rules, which are translated in some way into one of the types of rules reviewed in the former section.

[2] With elements, we refer to the information unit considered in the acquisition of rules. This element can be at a document level, term level, phrases, n-grams, etc. More about this point is discussed in Section 4.

Acquisition from data. This type of acquisition is given by rules extracted automatically. In the text framework, they are usually obtained by an statistical process over the documents or a representation of the documents in a collection.

Independently of the acquisition form of the rules, the antecedent and consequent of the rule can have a concrete representation. In a text framework, we can find different text representation in both sides of the rules such as a single word, a phrase, and so on. Different forms of representation in the rule are given in the next section.

4. Representation forms of antecedent and consequent

As important as the selection of a suitable technique to enhance the retrieval process, is the selection of the structure of the element or information unit to represent the document collection. In this work, we do not consider initially non-textual information. We call *item* to the element to be considered as unit to be part of the antecedent or consequent of the rule in our case or, in general the unit to be considered in whatever the retrieval process and the applied technique. Independently of the different types of items to consider, we have to specify the their level. Initially, we distinguish two types of item sets categorized by their level, namely term-level items and document-level items explained in the following.

4.1. Term-level items

Word items. These items are single words appearing in a document (stop-list or stemming processes can be assumed to be applied).

Phrase items. These items are composed words that syntactically have to be together to represent a concept, for instance, *world wide web*.

n-Grams items. These items are words occurring in a document as a sequence, and can be considered from *1-gram* to *n-grams*, with n equal to 5, as is proposed in [35], or can be used in a more general way, as in [11]. They can be considered as a generalization of the former cases. These items include sequences of words that syntactically have not to correspond to a concept, but reflecting how words appear in some order with a certain nearness. For instance, the sentence 'grammatical inference can be performed by recurrent neural networks' would be represented by the 6-gram 'grammatical inference performed recurrent neural networks'. Let us point out the difference between this kind of n-grams, where sequences of complete words in the sentence are considered, and the n-grams defined in [7], where characters inside a word are sliced. For instance, the bi-grams of the word 'web' would be '_w', 'we', 'eb', 'b_'.

These three types of items can be generalized to what is called *bag of words*, which can be enriched by *n*-grams and categories describing the document content [13,35].

Associated to these elements, a weight can be assigned following a certain scheme. We include here the most used:

- *Boolean weighting scheme*. It indicates the presence or absence of the word in the document. It can take value in {0,1}.
- *Frequency weighting schemes*. These are words with an associated weight meaning the relative frequency of the word in the document, or other weighted scheme generally based on the relative number of occurrences of the word in the document. Some weighted schemes are binary weight, within-document word frequency (TF), inverse document frequency (IDF), and the combination of the two former (TFIDF). The expressions of these schemes can be found in [37].
- *Fuzzy weighting schemes*. Initially, the fuzzy framework seems to be the perfect landscape to find text representations that explicitly deal with uncertainty, besides the inherent flexibility to represent human thinking. However, the inclusion of fuzzy logic in the Information Retrieval framework is, in some sense, quite limited, coming only from the extension of the boolean model. The current potentiality of this extension remains basically in the extension of the logical operators, the interface user–system with the facility of the use of labels to express weights, and the extension of the classical measures of recall and precision to their fuzzy counterparts, which allows to consider all the degrees of relevance of the documents, shown in a ranked list [24]. Nevertheless, the representation of documents in this model does not reflect uncertainty itself. In the literature, there are some options of scheme representations with fuzzy logic. Both [24] and [6] proposals are mathematical normalizations of classical weights. In [33] some opinion about the relevance of the document is reflected in the weighted scheme. These fuzzy weighting schemes correspond to term-level attributes with associated fuzzy values, and their applications are carried out with single words almost always.

4.2. Document-level items

These items are elements which take a whole document as an unit to deal with in text representation.

Citation attributes. These attributes indicate the presence or absence of the citation of a document belonging to the collection.

Link attributes. In a web framework, this kind of attributes indicate the presence or absence of a link to other web document belonging to the collection.

5. Usage of rules in text

Several applications can be found in the literature where the concept of rule have been utilized in a text framework.

Usage of logical rules. The use of logical rules in Information Retrieval takes place more as a consequence of a model than as a tool to build applications. Logical models of Information Retrieval have been proposed in the literature, where logical rules have been defined to make inference about user needs. A good study about these approaches can be found in [39].

Usage of production rules. This type of rules is used in one of the first and most extended applications of the rules in their classical concept of production rules: the expert systems for retrieval. The general idea of most of these systems is to help the user to formulate her/his queries by expert knowledge stored in a knowledge base. The origin of the rules is from an expert or from the user and the elements that take part in the rule have a conceptual nature. Some approaches of this usage can be found in Section 5.1.

Usage of association and fuzzy association rules. Several applications of this kind of rules in the text framework have recently appeared. In all of them, the *A priori* [1] or a similar algorithm is used to discover frequent term-sets (or other representation element, as we have seen in Sections 2 and 4), although not all of the works present a direct application in the field of Information Retrieval. Based on the purpose of the application, the algorithm is guided by the constraints of the rules to be found. For instance, for a categorization system, we desire rules where their consequent is a category label. Some approaches of this usage can be found in Section 5.2.

Usage of co-occurrences. In the Information Retrieval field, these associative mechanisms have been applied to augment the vocabulary of indexes and queries to enhance the retrieval process. Both in query and indexing enriching, the idea is to augment the terms of the query or the terms of the indexing to enrich the vocabulary and document representation and enhance the retrieval process. We must point out, however, that although the enriched representations of text are providing slightly better results, the *Equal Effectiveness Paradox* is still present, that is, with both usual and enriched text representations, the effectiveness of the retrieval task is very similar [30]. Some limitations about the use of co-occurrence in query expansion can also be found in [36], where the authors assert that the high frequencies of the terms to be added for query expansion are not good discriminators for enhancing the retrieval process. Therefore, a compromise between the enhancement of the retrieval process and the time consumed by enriching the document representation must be considered.

Therefore, rules have three broad groups of applications in Information Retrieval: first, expert systems which usually help users with query construction; second, the classification tasks, including retrieval and categorization; and

finally, the expansion or enriching tasks, which helps in automatic indexing and query formulation processes. In the next section some approaches for these retrieval applications are reviewed.

5.1. Expert systems

One of the first rule-base retrieval system was called RUBRIC (Rule Based Information Retrieval by Computer) developed by [34]. In RUBRIC, a set of production rules was used to capture user query concepts in a rule-base tree as a little knowledge base to help users to develop comprehensive queries. The main difficulty of the system is the generation of rules that capture the user query concepts and the lack of semantics of the rules.

An enhanced approach in this same line is the work of [22], in which an automatic method for rule construction is proposed. The authors define three types of rules based on the concept to represent, namely specific, general and possible. These rules are constructed automatically from a thesaurus, establishing the closeness between terms using predefined relationships of the thesaurus: narrow, broad and related terms. Finally, a comparative experiment is carried out by comparing a rule set given by an expert and one generated by the system, concluding the good performance of the latter.

These two works have a similar definition of a rule of the form:

$$\langle rule\rangle ::== \langle pattern\rangle \rightarrow \langle concept\rangle[\langle CertaintyDegree\rangle]$$

and their main difference is the use of logical rules by the material implication in the RUBRIC system (noted by symbol $\rightarrow$) while a new implication more assertive than logical (noted by symbol $\Rightarrow$) is defined in [22], where the authors reason about the inadequacy of the material implication when an inference process with certainty degrees is carried out.

Another approach to consider is the I^3R (Intelligent Intermediary for Information Retrieval) by Croft and Thompson [12], where a blackboard is used for reflecting the user interaction. Two types of rules are defined in this work. The first type are rules denominated *recognition* rules, and are applied over the stemmed index terms to be related to the concepts represented in the domain knowledge. The form of these rules is

$$\langle rule\rangle ::== \langle stem\rangle \rightarrow \langle concept\rangle[\langle CertaintyDegree\rangle]$$

The second type of rules are called relationship rules and can describe synonym, generalization, instantiation, part-of and cross-reference relationships. They are defined as follows:

$$\langle rule\rangle ::== \langle conceptA\rangle and/or \langle conceptB\rangle \rightarrow \langle conceptC\rangle[\langle CertaintyDegree\rangle]$$

In CODER (Composite Document Expert/Extended/Effective Retrieval) presented in [18], the author uses also the blackboard technique. A book is rep-

resented by means of facts, a hierarchical organization and different relations are defined to describe the index entries of the book. Rules and facts are represented in a Prolog system but the author does not describe the rule syntax.

In the fuzzy framework, there also some expert systems approaches for retrieval, although they are described by fuzzy relations instead of giving the rules specifically. In [4], the authors present a knowledge-based approach to construct queries. The knowledge base contains concepts and fuzzy relations among the concepts. On the one hand, fuzzy implication relations are defined to link more specific concepts to broader ones (these relations are not symmetric); and on the other hand, fuzzy synonym relations are also defined giving a fuzzy degree of synonym between two concepts.

5.2. Text classification tasks

Statistical rules are used in general in the classification of textual information, which include several tasks in Information Retrieval. It includes not only the determination of good documents in terms of relevance attending to user needs but also the classification of documents into categories (topics) attending to predefined classes [30]. In the following, we include studies found in the literature about both the retrieval and the categorization tasks.

Retrieval task. Information retrieval itself is a form of classification or categorization since the collection of documents is divided into two categories of documents, a category with relevant documents to the query and another category with the non relevant ones.

For example, in the generalized retrieval scheme proposed by Kraft and Buell [24], the computation of a matching value for a document to a query can be seen as computing the degree of firing for a classification rule. In this model, given $D = \{D_1, D_2, \ldots, D_n\}$, the set of textual documents in the database, and $T = \{t_1, t_2, \ldots, t_s\}$, the set of index terms, the indexing function W is $W : D \times T \to [0, 1]$. Note that if the value of W is 1, it implies that a document is in the set of documents about the concept(s) of a term, while if the value of W is 0, it implies that the document is not in the set, and values in the middle, if allowed, represent partial or weighted membership. This means that each document D_i is represented as a vector of dimension s, the number of terms:

$$D_i = \langle w_{i1}, w_{i2}, \ldots, w_{is} \rangle$$

Here, each w_{ij} is a real number (typically positive), characterizing the *weight* of the term t_j in D_i. These weights, called indexing weights and defined by the W function, can be estimated subjectively, or computed from the term frequencies (TF) or TFIDF which is the combination of TF with inverse document frequencies (IDF) (see Section 4.1 for more details). Consider Q, the set of user queries for information from the database, so that a: $Q \times T \to [0, 1]$ is the

query term weighting function. Thus each query q is represented in the same way as an s-dimension vector:

$$q = \langle w_{q1}, w_{q2}, \ldots, w_{qs} \rangle$$

Here, the w_{qj} weights are called query weights. In addition to the query weighting function, each query is also associated with a Boolean expression ϕ with terms as the Boolean variables. For instance, a query can be of the form $q = \langle 0.9, 0.8, 0, 0, 0.85, \ldots, \rangle$ with $\phi =$ "t_1 AND t_2 OR t_5". The intention of the above query q is to get documents which satisfy Boolean expression ϕ, under the specified term weights. Clearly, this retrieval model generalizes Boolean retrieval, and it also differs from traditional vector-space model of retrieval. To process queries, we have $g : W \times a \to [0, 1]$ to evaluate a given document along the dimensions of a single given keyword. Here by abuse of notation, we use $W \times a$ to denote $\text{range}(W) \times \text{range}(a) = [0, 1] \times [0, 1]$. Various forms for g have been developed, based on a being representative of term importance, or being a term threshold, or viewing the query as an ideal document, or hybrids of these forms [24]. Finally, $e : g_1 \times g_2 \times \cdots \times g_s \to [0, 1]$ is the retrieval status value (RSV), the evaluation of the relevance of the given document based on the Boolean structure (expression ϕ) of the entire query. Here the function e can be seen as a specification of a general rule schema, with each instantiation by a specific query q being a rule. The antecedent of such a rule is the formula ϕ for the query, and the consequent of the rule is the predicate "relevant". The rule maps the matching scores of a document's individual indexing terms to the matching score of the document in the "relevant" category.

Categorization task. The use of rules for categorization comes from a process of classification of documents into different categories regarding their topics in order to optimize a posteriori retrieval process. One of the most relevant works of categorization using rules is the one of [3]. The general idea of this work is the discovery of classification patterns automatically for document categorization. The aim of the induction process is to find sets of decision rules to distinguish among different categories which documents belong to. The attributes of the rules can be one word or a pair of words constructing a dictionary where an elimination process of the less frequent words is carried out.

Other interesting work for categorization is [10], where a method to classify electronic mails by their topic is proposed. The construction of rules comes from the RIPPER algorithm [9] in which a set of rules is built by adding rules to an empty set iteratively until all positive examples are covered. The form of a rule is based on single words in the antecedent belonging to a part of the mail where the word appears (*subject, body, from, to, ...*) and a word indicating the class of mail (a call for papers, for instance) in the consequent. The words appearing in the antecedent are given by a high frequent term selection process.

A generalization of the use of RIPPER for document categorization can be found in [11].

Finally, association rules have been also used for categorization [2], where the authors propose a solution for text categorization based on the application of the best generated association rules to build a classifier.

5.3. Text enriching tasks

With text enriching tasks, we refer to text applications where the vocabulary is augmented with related words to enhance the processes of indexing or querying. Some of the most relevant works for these purposes are included in the following.

Enriched indexing. In the indexing, an enriched representation of documents can be generated by including, for instance, the topic categories the document belongs to [35]. This way, the retrieval process is optimized indirectly since the representation of documents is enriched with terms related to the topic.

Query expansion. In querying, the terms that finally appear in the query are usually not very specific due to the lack of background knowledge of the user about the topic or just because in the moment of the query, the terms do not come to the user's mind. To help the user with the query construction, terms related to the words of a first query may be added to the query. The approaches given in the literature for this purpose are more related to association and co-occurrence expressed as relationships rather than rules. The most interesting ones are those using lexical and semantic relations given by *WordNet* [5,44].

This application has been also extended to the fuzzy framework. In [26,27], fuzzy rules have been used to expand user queries by adding new query terms or increasing query term weights. The fuzzy rules capture the associations and co-occurrences among important indexing terms. The fuzzy rules are of the form:

$$r : [t_i \geqslant w_i] \rightarrow [t_j \geqslant w_j]$$

where t_i, t_j are terms and w_i, w_j are numbers in $[0,1]$ interval specifying the weights of terms t_i and t_j in a potential query. The intuitive meaning of the above rule is that whenever a query term t_i's weight is equal to or greater than w_i in a query q, the query term t_j in query q should have weight at least w_j. These rules are obtained in [26,27] from the significant terms in centers of fuzzy clusters of the document collection. Consider a query q which is specified as

$$q = \langle w_{q1}, w_{q2}, \ldots, w_{qs} \rangle$$

For this query q, the above rule r is applicable to modify q if $w_{qi} \geqslant w_i$ and $w_{qj} \leqslant w_j$. The application of this rule to q (by the modus-ponens rule) will yield

q', which coincides with q on each dimension except $w_{q'j} = w_j$. The final modified query $q*$ is obtained from q by repeatedly applying the applicable rules until convergence. Note that each modification of the query can be viewed as an inference step in using the sound and complete fuzzy logic system in [8], where an inference step is typically of the form:

$$\text{If } (\phi \geqslant \alpha) \text{ and } (\phi \geqslant \alpha \rightarrow \psi \geqslant \beta), \text{ then infer } (\psi \geqslant \beta)$$

Following this same reasoning, an application of fuzzy inferencing and fuzzy clustering to user profile construction can be found in [28].

Fuzzy association rules have been also applied to query expansion [14]. The most accurate rules that include the original query words in the antecedent/consequent of the rule, are used to modify the query by automatically adding these terms to the query or, by showing to the user the related terms in those rules, so the modification of the query depends on the user's decision. A generalization or specification of the query will occur when the terms used to reformulate the query appear in the consequent/antecedent of the rule, respectively. This suggestion of terms helps the user to reduce the set of documents, leading the search through the desired direction.

6. Conclusions

The use of rules in the text framework is very extended due to the additional facility of constructing intelligent systems. The study of the rules have been done from different perspectives, including the management of uncertainty since user information needs have an inherent nature of vagueness. From their representation and acquisition, the rules have been classified in a general way, although some examples in the text framework have been given. From their usage, three broad applications have been found, namely, experts, classification tasks and enriching tasks.

Independently of their representation, acquisition and usage, the possible representation forms of antecedent and consequent have been given. However, different representations retrieve different documents and it is difficult to identify the one working best, besides the additional component of context dependence. From a semantic point of view, documents are related to concepts, but documents are formed by words, not by concepts. On the one hand, a concept can be expressed by more than one word appearing or not in the document, as well as by a group of words. On the other hand, the same word can be related to different concepts. Human mind works inferring concepts from the words of a document [3]. However, the simulation of this process by computers automatically is not so direct and represents one of the big challenges of the present and future to enhance retrieval systems.

References

[1] R. Agrawal, R., Skirant, Fast algorithms for mining association rules, in: Proc. of the 20th International Conference on Very Large Databases. Santiago, Chile, September 1994.

[2] M.L. Antonie, O.R. Zaíane, Text document categorization by term association, in: Proc. of the IEEE International Conference on Data Mining (ICDM). Maebashi City, Japan, December 2002.

[3] C. Aptè, F. Damerau, S.M. Weiss, Automated learning of decision rules for text categorization, ACM Transactions on Information Systems 12 (3) (1994) 233–251.

[4] G. Biswas, J.C. Bezdeck, M. Marques, V. Subramanian, Knowledge-Assisted document retrieval: I. The natural-language interface and II. The retrieval process, Journal of the American Society for Information Science 38 (2) (1987) 83–96, 97–110.

[5] R.C. Bodner, F. Song, Knowledge-based approaches to query expansion in information retrieval, in: G. McCalla (Ed.), Advances in Artificial Intelligence, Springer-Verlag, New-York, USA, 1996, pp. 146–158.

[6] G. Bordogna, P. Carrara, G. Pasi, Fuzzy approaches to extend boolean information retrieval, in: P. Bosc, J. Kacprzyck (Eds.), Fuzziness in Database Management Systems, Physica-Verlag, Germany, 1995, pp. 231–274.

[7] W.B. Cavnar, Using an *n*-gram-based document representation with a vector processing retrieval model, in: NIST Special Publication 500-226: Overview of the third Text Retrieval Conference (TREC-3), 1994, pp. 269–278.

[8] J. Chen, S. Kundu, A sound and complete fuzzy logic system using Zadeh's implication operator. Foundations of Intelligent Systems: Lecture Notes in Computer Science 1079 (1996) 233–242.

[9] W.W. Cohen, Fast effective rule induction, in: Proc. of the Twelfth International Conference on Machine Learning. Lake Taho, California, USA, 1995.

[10] W.W. Cohen, Learning rules that classify e-mail, in: Proc. of the 1996 AAAI Spring Symposium on Machine Learning and Information Access. Palo Alto, California, USA, 1996.

[11] W.W. Cohen, Y. Singer, Context-Sensitive learning methods for text categorization, ACM Transactions on Information Systems 17 (2) (1999) 141–173.

[12] W.B. Croft, R.H. Thompson, I^3R: a new approach to the design of document retrieval systems, Journal of the American Society for Information Science 38 (6) (1987) 389–404.

[13] M. Delgado, M.J. Martín-Bautista, D. Sánchez, M.A. Vila, Mining text data: special features and patterns, in: Proc. of EPS Exploratory Workshop on Pattern Detection and Discovery in Data Mining, London, September 2002.

[14] M. Delgado, M.J. Martín-Bautista, D. Sánchez, J.M. Serrano, M.A. Vila, Association rule extraction for text mining. Lecture Notes in Artificial Intelligence (LNAI) 2522 (2002) 154–162.

[15] M. Delgado, N. Marín, D. Sánchez, M.A. Vila, Fuzzy association rules: general model and applications, IEEE Transactions on Fuzzy Systems 11 (2) (2003) 214–225.

[16] D. Dubois, H. Prade, What are fuzzy rules an how to use them, Fuzzy Sets and Systems 84 (1996) 169–185.

[17] U.M. Fayyad, G. Piatetsky-Shapiro, P. Smyth, From data mining to knowledge discovery: an overview, in: U.M. Fayyad, G. Piatetsky-Shapiro, P. Smyth, R. Uthurusamy (Eds.), Advances in Knowledge Discovery and DataMining, AAAI Press, California, USA, 1996.

[18] E.A. Fox, Development of the coder system: a textbed for artificial intelligence methods in information retrieval, Information Processing and Management 23 (4) (1987) 341–366.

[19] R. Frost, Introduction to knowledge base systems, William Collins Sons & Co., 1986.

[20] S. Gauch, Intelligent information retrieval: an introduction, Journal of the American Society for Information Science 43 (2) (1991).

[21] M. Hearst, Untangling text data mining, in: Proc. of the 37th Annual Meeting of the Association for Computational Linguistics (ACL'99), University of Maryland, USA, 1999.
[22] M. Kim, F. Lu, V.V. Raghavan, Automatic construction of rule-based trees for conceptual retrieval, in: Proc. of String Processing and Information Retrieval (SPIRE2000), A Coruña, Spain, September 2000, pp. 153–161.
[23] Y. Kodratoff, Comparing machine learning and knowledge discovery in databases: an application to knowledge discovery in texts, in: G. Paliouras, V. Karkaletsis, C.D. Spyropoulos (Eds.), ACAI'99, Lecture Notes on Artificial Intelligence 2049, Berlin: Springer-Verlag, 2001, pp. 1–21.
[24] D.H. Kraft, D.A. Buell, Fuzzy sets and generalized boolean retrieval systems, in: D. Dubois, H. Prade (Eds.), Readings in Fuzzy Sets for Intelligent Systems, Morgan Kaufmann Publishers, San Mateo, CA, 1993, pp. 648–659.
[25] D.H. Kraft, G. Bordogna, G. Pasi, Fuzzy set techniques in information retrieval, in: J.C. Bezdeck, D. Dubois, H. Prade (Eds.), Fuzzy Sets in Approximate Reasoning and Information Systems, in Series The Handbooks of Fuzzy Sets, Kluwer Academic Publishers, Massachusetts, USA, 1999.
[26] D.H. Kraft, J. Chen, Integrating and extending fuzzy clustering and inference to improve text retrieval performance, in: H.L. Larsen, J. Kacprzyk, S. Zadrozny, T. Andreasen, H. Christiansen (Eds.), Flexible Query Answering Systems. Recent advances, in series Advances in Soft Computing. Heildelberg, New York: Physica-Verlag, 2000, pp. 386–395.
[27] D.H. Kraft, J. Chen, A. Mikulcic, Combining fuzzy clustering and fuzzy inferencing in information retrieval, in: Proc. of FUZZ-IEEE'2000. San Antonio, TX, May 2000.
[28] D.H. Kraft, J. Chen, M.J. Martín-Bautista, M.A. Vila, Textual information retrieval with user profiles using fuzzy clustering and inferencing, in: P.S. Szczepaniak, J. Segovia, J. Kacprzyk, L.A. Zadeh (Eds.), Intelligent Exploration of the Web, Physica-Verlag, Heidelberg Germany, 2002, pp. 152–165.
[29] D.D. Lewis, Learning in intelligent information retrieval, in: Proc. of the Eight International Workshop on Machine Learning (ML'91). San Mateo, CA: Morgan Kaufmann, 1991, pp. 235–239.
[30] D.D. Lewis, Representation and learning in information retrieval. Doctoral Dissertation, Department of Computer and Information Science, University of Massachusetts, 1992.
[31] R. López de Mántaras, Approximate Reasoning Models, Ellis Horwood Limited, England, 1990.
[32] P. Lucas, L. Van Der Gaag, Principles of Expert Systems, Addison Wesley, Great Britain, 1991.
[33] M.J. Martín-Bautista, M.A. Vila, D. Sánchez, H.L. Larsen, Intelligent filtering with genetic algorithms and fuzzy logic, in: B. Bouchon-Meunier, J. Gutiérrez-Ríos, L. Magdalena, R.R. Yager, Technologies for constructing Intelligent Systems 1, Germany: Physica-Verlag, 2002, pp. 351–362.
[34] B.P. McCune, R.M. Tong, J.S. Dean, D.G. Shapiro, RUBRIC: a system for rule-based information retrieval, IEEE Transaction on Software Engineering 11 (9) (1985).
[35] D. Mladenic, Machine learning on non-homogeneous, distributed data. Doctoral Dissertation, Faculty of Computer and Information Science, University of Ljubljana, 1998.
[36] H.P. Peat, P. Willet, The limitations of term co-occurrence data for query expansion in document retrieval systems, Journal of the American Society for Information Science 42 (5) (1991) 378–383.
[37] G. Salton, C. Buckley, Term weighting approaches in automatic text retrieval, Information Processing and Management 24 (5) (1988) 513–523.
[38] G. Salton, M.J. McGill, Introduction to modern information retrieval, McGraw Hill, USA, 1983.

[39] F. Sebastiani, On the role of logic in information retrieval, Information Processing and Management 34 (1) (1998) 1–18.
[40] C. Silverstein, S. Brin, R. Motwani, Beyond market baskets: generalizing association rules to dependence rules, Data Mining and Knowledge Discovery 2 (1998) 39–68.
[41] K. Sparck Jones, The role of artificial intelligence in information retrieval, Journal of the American Society for Information Science 42 (8) (1991) 558–565.
[42] K. Sparck Jones, Information retrieval and artificial intelligence, Artificial Intelligence 114 (1999) 257–281.
[43] H.R. Turtle, W.B. Croft, Uncertainty in information retrieval systems, in: A. Motro, P. Smets (Eds.), Uncertainty management in information systems: from needs to solutions, Massachusetts, USA, 1997.
[44] E. Voorhees, Query expansion using lexical-semantic relations, in: Proc. of the 17th International Conference on Research and Development in Information Retrieval (SIGIR). Dublin, Ireland, July 1994.
[45] L.A. Zadeh, Fuzzy logic, neural networks, and soft computing, Communications of the ACM 37 (3) (1994) 77–84.