Available online at www.sciencedirect.com
SCIENCE DIRECT®
Fuzzy Sets and Systems 148 (2004) 85–104
FUZZY sets and systems
www.elsevier.com/locate/fss

Mining web documents to find additional query terms using fuzzy association rules

M.J. Martín-Bautista*, D. Sánchez, J. Chamorro-Martínez, J.M. Serrano, M.A. Vila

Department of Computer Science and Artificial Intelligence, University of Granada, C/Periodista Daniel Saucedo Aranda, Granada 18071, Spain

Abstract

In this paper, we present an application of association rules to query refinement. Starting from an initial set of documents retrieved from the web, text transactions are constructed and association rules are extracted. A fuzzy extension of text transactions and association rules is employed, where the presence of the terms (items) in the documents (transactions) is determined with a value between 0 and 1. The obtained rules offer the user additional terms to be added to the query with the purpose of guiding the search and improving the retrieval.
© 2004 Elsevier B.V. All rights reserved.

Keywords: Association rules; Fuzzy logic; Information retrieval; Query refinement

1. Introduction

Searching the web is not always so successful as users expect. The lack of homogeneity in the structure of documents and in their indexing by the search robots makes difficult to find relevant information in the web. Most of the retrieved sets of documents in a web search, including the multimedia ones, meet the search criteria but do not satisfy the user needs. Moreover, the amount of documents is so huge that the user feels overwhelmed. This is due generally to a lack of specificity in the formulation of the queries. Some causes of this are that most of the times, the user does not know the vocabulary of the topic, or query terms do not come to user's mind at the query moment. In the case of image retrieving, is even more difficult to construct a query due to the lack of search terms related to the content of the image.

* Corresponding author.
E-mail addresses: mbautis@decsai.ugr.es (M.J. Martín-Bautista), daniel@decsai.ugr.es (D. Sánchez), jesus@decsai.ugr.es (J. Chamorro-Martínez), jmserrano@decsai.ugr.es (J.M. Serrano), vila@decsai.ugr.es (M.A. Vila).

0165-0114/$ - see front matter © 2004 Elsevier B.V. All rights reserved.
doi:10.1016/j.fss.2004.03.007

One possible solution to this problem is the process known as *query expansion* or *query reformulation*. After the query process is performed, new terms are added to and/or removed from the query in order to improve the results, i.e., to discard uninteresting retrieved documents and/or to retrieve interesting documents that were not retrieved by the query. A good review of the topic in the Information Retrieval field can be found in [19].

The purpose of this work is to provide a system with a query reformulation ability using mining technologies. Data mining techniques have been applied successfully in the last decade in the field of Databases, but have been also applied to solve some classical Information Retrieval problems such as document classification [33] and query refinement [48]. In the last case, one of the approaches employed is to obtain association rules that suggest new terms that could be added to the query.

In this paper, we propose to use fuzzy association rules and a assessment framework different from the support/confidence for query refinement. These contributions provide some advantages. First, fuzzy rules take into account the degree of importance of terms in the representation of documents. Second, the measures employed are more suitable to determine which rules are useful for our purposes.

The paper is organized as follows: in Section 2, a summary of literature with the same purpose of this work is included. In Section 3, the concepts of association rules, fuzzy association rules and fuzzy transactions are presented briefly. In Section 4, an application of this theory to text framework is given. The application of extracted text association rules to query reformulation in an Information Retrieval framework is proposed in Section 5. An experimental example is shown in Section 6. Finally, concluding remarks and future trends are given in Section 7.

2. Related work

In the field of Information Retrieval, this problem has been treated as query expansion or query refinement. The solutions given to solve it are based mainly on two approaches: the first is the augmentation of query terms to improve the retrieval process without the intervention of the user. The second one is the suggestion of new terms to the user to be added to the original query in order to guide the search towards a more specific document space. The first case is called *automatic query expansion* [8,22], while the second case is called *semi-automatic query-expansion* [39,50].

We can also distinguish different cases based on the set from which the terms are selected. If a document collection is considered as a whole from which the terms are extracted to be added to the query, the technique is called *global analysis*, as in [52]. However, if the expansion of the query is performed based on the documents retrieved from the first query, the technique is denominated *local analysis*, and the set of documents is called *local set*.

Local analysis can also be classified into two types. On the one hand, *local feedback* adds common words from the top-ranked documents of the local set. These words are identified sometimes by clustering the document collection [2]. In this group we can include the relevance feedback process, since the user has to evaluate the top ranked documents from which the terms to be added to the query are selected. On the other hand, *local context analysis* [52], which combines global analysis and context local feedback to add words based on relationships of the top-ranked documents. The calculus of co-occurrences of terms is based on passages (text windows of fixed size), as in global

analysis, instead of complete documents. The authors show that, in general, local analysis performs better than global one.

In the literature, we can find several approaches using different techniques to identify terms that should be added to the original query. The first group is based on their association relation by co-occurrence to query terms [49]. Instead of simply terms, in [52] the authors find co-occurrences of concepts given by noun groups with the query terms. Another approach based on the concept space approach is [10]. The statistical information can be extracted from a clustering process and a ranking of documents from the local set, as it is shown in [11] or by similarity of the top-ranked documents [37]. All these approaches where a co-occurrence calculus is performed has been said to be suitable to construct specific knowledge base domains, since the terms are related, but they cannot be distinguished how [5].

The techniques in the second group search terms on the basis of their similarity to the query terms, by constructing a similarity term thesaurus [41]. Other approaches in this same group use techniques to find out the most discriminatory terms, which are the candidates to be added to the query. These two characteristics can be combined by first calculating the nearest neighbors and second, by measuring the discriminatory ability of the terms [39]. The last group is formed by approaches based on lexical variants of query terms extracted from a lexical knowledge base such as Wordnet [36]. Some approaches in this group are [5,51] where a semantic network with term hierarchies is constructed. The authors reveal the adequacy of this approach for general knowledge bases, which can be identified in general terms with global analysis, since the set of documents from which the hierarchies are constructed is the corpus, and not the local set of a first query. Previous approaches with the idea of hierarchical thesaurus can be also found in the literature, where an expert system of rules interprets the user's queries and controls the search process [22].

3. Association rules and fuzzy association rules

In this section, we briefly review association rules and some useful extensions able to deal with weighted sets of items.

3.1. Association rules

Let I be a set of elements called "items" and let T be a set of elements called "transactions", each transaction being a set of items. Let us consider two itemsets (sets of items) $I_1, I_2 \subseteq I$, where $I_1 \cap I_2 = \emptyset$. An association rule [1] $I_1 \Rightarrow I_2$ is an implication rule meaning that the apparition of itemset I_1 in a transaction implies the apparition of itemset I_2 in the same transaction. The reciprocal does not have to happen necessarily [30]. I_1 and I_2 are called antecedent and consequent of the rule, respectively.

The problem of obtaining association rules that hold in a set of transactions T is known as the *boolean association rule problem* (BARP). This is an interesting procedure to extract knowledge from data with many different applications depending on the way we instantiate the abstract concepts of item and transaction.

3.1.1. Assessing rules

There are two relevant aspects of association rules that we need to assess. On the one hand, an association rule can be interesting even if there are some exceptions to the rule in the set T, so we are interested in assessing the *accuracy* of the rule and to decide on its basis whether the rule is accurate or not. On the other hand, an accurate rule that holds in few transactions is not interesting since it is not representative of the whole data and its possible application is limited. Hence, we need to measure the amount of transactions supporting the rule and to decide on that basis whether the rule is important or not.

The assessment of association rules is usually based on the *support* and *confidence*. Support is the percentage of transactions where the rule holds, while confidence measures the strength of the rule as the percentage of transactions containing I_1, that contain I_2. The objective of the BARP is to obtain all the rules with support and confidence greater than user-defined thresholds *minsupp* and *minconf*, respectively. These are called *strong rules*.

It is possible to calculate support and confidence from the support of an itemset. We shall note $supp(I_k)$ the support of the itemset I_k, defined as the probability of finding I_k in a transaction of T, i.e.,

$$supp(I_k) = \frac{|\{t \in T \mid I_k \subseteq t\}|}{|T|}. \tag{1}$$

The support and confidence of the rule $I_1 \Rightarrow I_2$ noted by $Supp(I_1 \Rightarrow I_2)$ and $Conf(I_1 \Rightarrow I_2)$, respectively, are calculated as follows:

$$Supp(I_1 \Rightarrow I_2) = supp(I_1 \cup I_2), \tag{2}$$

$$Conf(I_1 \Rightarrow I_2) = \frac{supp(I_1 \cup I_2)}{supp(I_1)}. \tag{3}$$

Confidence is an estimation of the conditional probability of the consequent I_2 with respect to the antecedent I_1. If we note by $\Gamma_{I_j} \subseteq T$ the set of transactions containing I_j, confidence is a measure of the degree of inclusion of Γ_{I_1} in Γ_{I_2}. In particular,

- $Conf(I_1 \Rightarrow I_2) = 1$ iff $\Gamma_{I_1} \subseteq \Gamma_{I_2}$,
- $Conf(I_1 \Rightarrow I_2) = 0$ iff $\Gamma_{I_1} \cap \Gamma_{I_2} = \emptyset$ and $\Gamma_{I_1} \neq \emptyset$.

3.1.2. Certainty factors as an alternative to confidence

Some authors have shown that confidence can yield misleading results in some cases. A summary of papers discussing this problem and the alternative measures proposed is in [4].

Basically, the problem with confidence is that it does not take into account the support of I_2, hence it is unable to detect statistical independence or negative dependence, i.e., a high value of confidence can be obtained in those cases. This problem is specially important when there are some items with very high support. In the worst case, given an itemset I_C such that $supp(I_C) = 1$, every rule of the form $I_A \Rightarrow I_C$ will be strong provided that $supp(I_A) > minsupp$. It has been shown that in practice, a large amount of rules with high confidence are misleading because of the aforementioned problems.

Example 1. Let $supp(I_1)=0.5$, $supp(I_2)=0.8$ and $supp(I_1 \cup I_2)=0.4$. Then $Conf(I_1 \Rightarrow I_2)=0.4/0.5=0.8$ (rather high) but in fact there is statistical independence between I_1 and I_2 since $supp(I_1) * supp(I_2)=0.8 * 0.95=0.76=supp(I_1 \cup I_2)$.

In [4], the use of certainty factors [46] was proposed to avoid the problems introduced by confidence. The certainty factor of $I_1 \Rightarrow I_2$, noted $CF(I_1 \Rightarrow I_2)$, is obtained as follows. If $Conf(I_1 \Rightarrow I_2) > supp(I_2)$ the value of the factor is given by expression (4); otherwise, is given by expression (5), considering that if $supp(I_2)=1$, then $CF(I_1 \Rightarrow I_2)=1$ and if $supp(I_2)=0$, then $CF(I_1 \Rightarrow I_2)=-1$

$$CF(I_1 \Rightarrow I_2)=\frac{Conf(I_1 \Rightarrow I_2)-supp(I_2)}{1-supp(I_2)}, \tag{4}$$

$$CF(I_1 \Rightarrow I_2)=\frac{Conf(I_1 \Rightarrow I_2)-supp(I_2)}{supp(I_2)}. \tag{5}$$

Certainty factor takes values in $[-1,1]$, and measures the variation of our belief that $I_2 \subseteq \tau \in T$ when we know $I_1 \subseteq \tau$. It can be also interpreted as a measure of strength and direction of the dependence between I_1 and I_2. In particular,

- $CF(I_1 \Rightarrow I_2)=1$ means maximum increment of our belief (maximum positive dependence). In addition, $CF(I_1 \Rightarrow I_2)=1$ iff $Conf(I_1 \Rightarrow I_2)=1$ [4].
- $CF(I_1 \Rightarrow I_2)=0$ means no variation of our belief (statistical independence).
- $CF(I_1 \Rightarrow I_2)=-1$ means maximum decrement of our belief (maximum negative dependence).

In Example 1, $CF(I_1 \Rightarrow I_2)=0$, meaning there is no dependence between I_1 and I_2, as expected.

3.2. Fuzzy association rules

Several authors have proposed *fuzzy association rules* as a generalization of association rules when data is fuzzy or has been previously fuzzyfied [3,13,26,31,32]. Though, most of these approaches have been introduced in the setting of relational databases, we think that most of the measures and algorithms proposed can be employed in a more general framework. A broad review, including references to papers on extensions to the case of quantitative attributes and hierarchies of items, can be found in [14].

In this paper we shall employ the model proposed in [13]. This model considers a general framework where data is in the form of fuzzy transactions, i.e., fuzzy subsets of items. A (crisp) set of fuzzy transactions is called a FT-set, and fuzzy association rules are defined as those rules extracted from a FT-set. Fuzzy relational databases can be seen as a particular case of FT-set. Other datasets, such as the description of a set of documents by means of fuzzy subsets of terms, are also particular cases of FT-sets but fall out of the relational database framework.

Given a FT-set $\tilde{T}$ on a set of items I and a fuzzy transaction $\tilde{\tau} \in \tilde{T}$, we note $\tilde{\tau}(i)$ the membership degree of i in $\tilde{\tau}$ $\forall i \in I$. We also define $\tilde{\tau}(I_0)=\min_{i \in I_0} \tilde{\tau}(i)$ for every itemset $I_0 \subseteq I$.

With this scheme, we have a degree in $[0,1]$ associated to each pair $\langle \tilde{\tau}, I_0 \rangle$. Sometimes it is useful to see this information in a different way by means of what we call the *representation* of an itemset. The idea is to see an itemset as a fuzzy subset of transactions. The representation of an itemset

$I_0 \subseteq I$ in a FT-set $\tilde{T}$ is the fuzzy subset $\tilde{\Gamma}_{I_0} \subseteq \tilde{T}$ defined as

$$\tilde{\Gamma}_{I_0} = \sum_{\tilde{\tau} \in \tilde{T}} \tilde{\tau}(I_0)/\tilde{\tau}. \tag{6}$$

On this basis, a fuzzy association rule is an expression of the form $I_1 \Rightarrow I_2$ that holds in a FT-set $\tilde{T}$ iff $\tilde{\Gamma}_{I_1} \subseteq \tilde{\Gamma}_{I_2}$. The only difference with the definition of crisp association rule is that the set of transactions is a FT-set, and the inclusion above is the usual between fuzzy sets.

The same considerations about assessment of rules must be taken into account for fuzzy rules. We discuss on this issue in the next section.

3.2.1. Measuring accuracy and importance for fuzzy association rules

In order to calculate the support of an itemset I_0 we must obtain the cardinality of the set $\tilde{\Gamma}_{I_0}$. However, this is a fuzzy set, so we must employ some fuzzy cardinality measure. Two main approaches are available in the literature, scalar cardinalities such as the power (sigma-count) [12], that provide a real number as the cardinality of a fuzzy set, and fuzzy cardinalities (see [16] for a review), that provide a fuzzy subset of the non-negative integers.

The scalar approach is employed in [26,31] in order to generalize support and confidence. In the same paper [31] this approach is employed to provide another measure, based on the idea of statistical correlation. In [3], scalar cardinalities are employed to compute a measure called *adjusted difference*, inspired on statistical tests, and *weight of evidence*, a measure of information gain.

Some authors have shown that the scalar approach can yield misleading results, basically because the addition of many small values can yield a high value, meaning that a given itemset is in a transaction with degree 1 when in fact it is in many transactions with a very low degree [18,35].

Example 2. Let $\tilde{T} = \{\tilde{\tau}_1, \tilde{\tau}_2, \ldots, \tilde{\tau}_{1000}\}$, let $i_1, i_2 \in I$ and let

$$\tilde{\Gamma}_{i_1}(\tilde{\tau}_k) = \begin{cases} 1, & k = 1, \\ 0.01, & k \neq 1, \end{cases} \quad \tilde{\Gamma}_{i_2}(\tilde{\tau}_k) = \begin{cases} 1, & k = 2, \\ 0.01, & k \neq 2. \end{cases}$$

Then $\tilde{\Gamma}_{\{i_1,i_2\}} = \sum_{\tilde{\tau} \in \tilde{T}} 0.01/\tilde{\tau}$ and confidence of $\{i_1\} \Rightarrow \{i_2\}$ is

$$Conf(\{i_1\} \Rightarrow \{i_2\}) = \frac{\sum_{\tilde{\tau} \in \tilde{T}} \tilde{\Gamma}_{\{i_1,i_2\}}(\tilde{\tau})}{\sum_{\tilde{\tau} \in \tilde{T}} \tilde{\Gamma}_{\{i_1\}}(\tilde{\tau})} = 0.91$$

that is a quite high value, even though the only transaction that significantly contains i_1 ($\tilde{\tau}_1$) minimally contains i_2.

This problem has been detected when extending other measures by means of sigma-counts. Though we have not tested this (classical) problem on the scalar-based accuracy measures mentioned above, we have followed an approach based on fuzzy cardinalities.

In [13] support and confidence are extended to the fuzzy case by using the evaluation of quantified sentences. In this approach, fuzzy cardinalities are matched against linguistic quantifiers to obtain an accomplishment degree of the sentence. Let Q_M be a fuzzy quantifier defined as $Q_M(x) = x, \forall x \in [0, 1]$. We define the *support of an itemset* I_0 in an FT-set T as the evaluation of the quantified sentence

(7), while the *support of a rule* $I_1 \Rightarrow I_2$ in T is given by the evaluation of (8). Finally, its confidence is the evaluation of the quantified sentence in (9).

$$Q_M \ of \ T \ are \ \tilde{\Gamma}_{I_0}, \tag{7}$$

$$Q_M \ of \ T \ are \ \tilde{\Gamma}_{I_1 \cup I_2} = Q_M \ of \ T \ are \ \tilde{\Gamma}_{I_1} \cap \tilde{\Gamma}_{I_2}, \tag{8}$$

$$Q_M \ of \ \tilde{\Gamma}_{I_1} \ are \ \tilde{\Gamma}_{I_2}. \tag{9}$$

The interpretation of these measures is *the degree to which the support* (*confidence*) *is* Q_M.

We evaluate the sentences by means of method GD presented in [17]. Method *GD* obtains the evaluation of a sentence "Q of F are G" as

$$\sum_{\alpha_i \in \Delta(G/F)} (\alpha_i - \alpha_{i+1}) Q\left(\frac{|(G \cap F)_{\alpha_i}|}{|F_{\alpha_i}|}\right), \tag{10}$$

where $F \cap G$ is computed using the minimum, and $\Delta(G/F) = \Lambda(G \cap F) \cup \Lambda(F)$, $\Lambda(F)$ being the level set of F. We label these values as $\Delta(G/F) = \{\alpha_1, \ldots, \alpha_p\}$ with $\alpha_i > \alpha_{i+1}$ for every $i \in \{1, \ldots, p\}$ and $\alpha_{p+1} = 0$. Finally, the set F is assumed to be normalized, otherwise F is normalized and the same normalization factor is applied to $G \cap F$ before the evaluation.

In addition, let us point out that when we are dealing with crisp transactions, the evaluation of these sentences yield the ordinary measures of support and confidence of association rules. Hence, these measures can be considered an extension of support and confidence to the fuzzy case. Further discussion can be found in [13].

As we mentioned in previous sections, we shall use certainty factors instead of confidence to assess the accuracy of rules. In the fuzzy case, we obtain the certainty factor from support and confidence in the way described in Section 3.1.2.

In Example 2, using *GD* with Q_M we obtain $Conf(\{i_1\} \Rightarrow \{i_2\}) = 0.01$, $supp(\{i_2\}) \approx 0.011$ and $CF(\{i_1\} \Rightarrow \{i_2\}) \approx 0.001$. These values are, in our opinion, more in accordance with the data.

3.3. Ratio rules

Ratio rules [29] take into account quantitative values associated to pairs $\langle attribute, value \rangle$. In this approach the starting point is a matrix $\mathbf{X}$ with N rows and M attributes, where each row corresponds to a transaction and the value for row i and column j, x_{ij}, is a real number. In its simplest form, a ratio rule is an expression of the form $m_j : m_k \Rightarrow r_1 : r_2$ where m_j and m_k are attributes and r_1 and r_2 are real numbers. The rule indicates the direction of the correlation between the N values x_{ij} and the N values x_{ik} by means of an unit vector with coordinates (r_1, r_2). The meaning of such rule is that the ratio between values x_{ij} and values x_{ik} is close to the ratio $r_1 : r_2$.

The discovery of ratio rules is based on eigensystem analysis, i.e., to compute the eigenvectors and eigenvalues of the covariance matrix of the data points. This analysis identifies the orthogonal directions (axes) of greatest variance of the data. Each ratio rule corresponds to such an axe, hence a ratio rule has the general form $m_1 : m_2 : \cdots : m_l \Rightarrow r_1 : r_2 : \cdots : r_l$.

An efficient algorithm is proposed in [29]. The algorithm performs a single pass over the data, using existing efficient methods to compute the eigenvectors of the matrix. The k rules (eigenvectors) with higher eigenvalues such that the addition of their eigenvalues cover 85% of the grand total are obtained.

A very interesting contribution of this approach is the possibility to predict missing values with applications in data cleaning, forecasting, and "what-if" scenarios. Other applications of ratio rules are outlier detection and visualization of data structure. In addition, a measure of goodness for a set of rules on the basis of their ability to predict values is provided in [29].

3.4. Discussion

In this section, we discuss about relations between crisp/fuzzy association rules and ratio rules. In summary, our conclusion is that fuzzy rules are a generalization of crisp ones, and ratio rules are a different, complementary approach to the extraction of patterns in order to describe the data, all of them with different and very interesting applications.

3.4.1. Crisp vs. fuzzy association rules

Crisp rules are a particular case of fuzzy ones, since crisp transactions are a particular case of fuzzy ones. As we commented in previous sections, some of the importance and accuracy measures proposed by several authors turn into the usual measures of support, confidence or certainty factor in the crisp case.

3.4.2. Crisp association rules vs. ratio rules

Some key differences are:

- Ratio rules analyze the relation between quantitative values associated to items in transactions. Crisp association rules analyze the relation between the presence of itemsets in transactions. No quantitative relation between itemsets is analyzed.
- Crisp association rules are asymmetric in nature and measure a degree of inclusion or conditional dependence, while ratio rules are symmetric and measure correlation. [1]
- Support indicates the amount of transactions where the rule holds. This is not taken into account by ratio rules.
- Confidence is a measure of inclusion in terms of conditional probability. Certainty factors measure statistical dependence/independence, and can be seen as a measure of non-trivial inclusion. Coordinates of ratio rules measure the ratio between quantities for several attributes in a set of transactions.

3.4.3. Fuzzy association rules vs. ratio rules

The starting point of both is very similar: a matrix containing real values. Fuzzy rules restrict the possible values to $[0, 1]$ and interpret them as the degree to which an itemset is in a transaction (i.e. verify a certain property represented by the transaction). For example, suppose item = term and

[1] One approach to the discovery of rules based on statistical chi-square test that provide a symmetric degree of statistical dependence is presented in [47]. These are called *dependence rules* and are symmetric.

transaction = representation of a document. The appearance of the item in the transaction means that "it describes in some degree the document for retrieval purposes". A $[0,1]$ value measures this presence of the term in the description/representation of the document. Ratio rules neither bound the domain of values nor restrict the interpretation of transactions.

Again, the objective of fuzzy rules is to analyze inclusion and dependence, not correlation. In fact, the rule $I_1 \Rightarrow I_2$ holds when $\tilde{I}_{I_1} \subseteq \tilde{I}_{I_2}$, i.e., $\tilde{I}_{I_1}(\tilde{\tau}) \leqslant \tilde{I}_{I_2}(\tilde{\tau})\ \forall \tilde{\tau} \in \tilde{T}$, while a ratio rule analyzes the correlation between values of $\tilde{I}_{I_1}$ and $\tilde{I}_{I_2}$ in the set of transactions $\tilde{T}$. Both tasks are independent, in fact it is easy to design a case where inclusion holds but not correlation, and vice versa. Of course, double inclusion of fuzzy sets implies correlation (equality in fact) of membership degrees, but this is a very particular and rare case.

As we shall see, our approach to query refinement relies on inclusion and statistical dependence rather than correlation, so we shall employ fuzzy association rules.

4. Definition of mining elements in a text framework

In this section we define several concepts related to data mining in a text framework.

4.1. Text items

Initially, we could consider term- and document-level items, to find relations among terms in the first case or among documents in the second one [30]. In this approach, we consider term-level items. Different representations of text for association rules extraction at term-level can be found in the literature: bag of words, indexing keywords, term taxonomy and multi-term text phrases [15]. In our case, we use automatic indexing techniques coming from Information Retrieval [44] to obtain *word items*, that is, single words appearing in a document where stop-list and/or stemming processes can be applied.

4.1.1. Item weighting schemes

We represent each document by a set of terms with a weight meaning the presence of the term in the document. Some weighting schemes for this purpose can be found in [43].

In this work, we consider three different weighting schemes [30]:

Boolean weighting scheme: It takes values $\{0,1\}$ indicating the absence or presence of the word in the document, respectively.

Frequency weighting scheme: It associates to each term a weight meaning the relative frequency of the term in the document. In a fuzzy framework, the normalization of this frequency can be carried out by dividing the number of occurrences of a term in a document by the number of occurrences of the most frequent term in that document [7].

TFIDF weighting scheme: It is a combination of the within-document word frequency (TF) and the inverse document frequency (IDF). The expressions of these schemes can be found in [43]. We use this scheme in its normalized form in the interval $[0,1]$ according to [6]. In this scheme, a term that occurs frequently in a document but infrequently in the collection is assigned a high weight.

4.2. Text transactions

In a text framework, we identify each transaction with the representation of a document. Therefore, from a collection of documents $D = \{d_1, \ldots, d_n\}$ we can obtain a set of terms $I = \{t_1, \ldots, t_m\}$ which

is the union of the keywords for all the documents in the collection. The weights associated to these terms in a document d_i are represented by $W_i = (w_{i1}, \ldots, w_{im})$. For each document d_i, 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.

Considering these elements, we can define a *text transaction* $\tau_i \in T$ as the extended representation of document d_i. Without loosing generalization, we can write the set of transactions associated to the collection of document D as $T_D = \{d_1, \ldots, d_n\}$.

When the weights $W_i = (w_{i1}, \ldots, w_{im})$ associated to the transactions take values in $\{0, 1\}$, that is, following the boolean weighting scheme of the former section, the transactions can be called boolean or crisp transactions, since the values of the tuples are 1 or 0 meaning that the attribute is present in the transaction or not, respectively.

4.2.1. Fuzzy text transactions

As we have explained in Section 4.1.1, we can consider a weighted representation of the presence of the terms in the documents. In the fuzzy framework, a normalized weighting scheme in the unit interval is employed. We call them *fuzzy weighting schemes*. Concretely, we consider two fuzzy weighting schemes, namely the frequency weighting scheme and the TDIDF weighting scheme, both normalized. Therefore, analogously to the former definition of text transactions, we can define a set of *fuzzy text transactions* $FT_D = \{d_1, \ldots, d_n\}$, where each document d_i corresponds to a fuzzy transaction $\tilde{\tau}_i \in FT$, and where the weights $W = \{w_{i1}, \ldots, w_{im}\}$ of the keyword set $I = \{t_1, \ldots, t_m\}$ are fuzzy values from a fuzzy weighting scheme.

5. Association rules and fuzzy association rules for query refinement

Before query refinement can be applied, we assume that a retrieval process is performed, i.e., we shall start from a set of documents obtained from an initial query (see Fig. 1). From that collection of documents, their representation is obtained as in classical information retrieval, and a transformation of this representation into a transactional one is carried out. Transactions are processed to extract association rules (fuzzy association rules in our case), and based on certain criteria, as we explain below, a list of terms from some of these rules is obtained. Finally, the user selects from that list the terms to add to the query so the query process starts again.

The representation of the documents is obtained following one of the weighting schemes proposed in Section 4.1.1. The document representation building process is shown in Algorithm 1. Given a query and a set of documents, the query representation is matched to each document representation in order to obtain a relevance value for every document. If a document term does not appear in the query, its value will be assumed to be 0. In the crisp case, the considered model is the Boolean one [44], while in the fuzzy case the considered model is the generalized Boolean model with fuzzy logic [9].

The user's initial query generates a set of ranked documents. If the top-ranked documents do not satisfy user's needs, the query improvement process starts. Since we start from the initial set of documents retrieved from a first query, we are dealing with a *local analysis* technique. And, since we just considered the top-ranked documents, we can classify our technique as a *local feedback* one.

From the initial retrieved set of documents, called *local set*, association rules are found and additional terms are suggested to the user in order to refine the query. As we have explained in

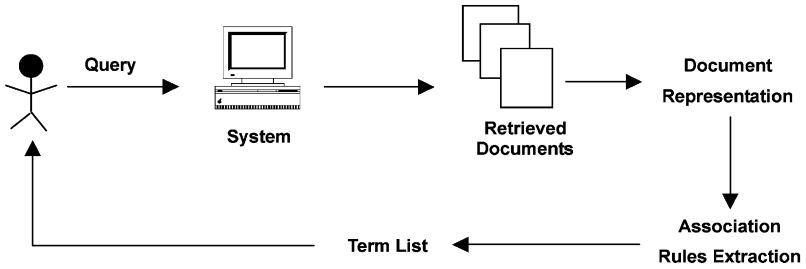

Fig. 1. Scheme of the process of query refinement using association rules.

Section 2, there are two general approaches to query refinement: automatic and semi-automatic. In our case, as we offer to the user a list of terms to add to the query, the system performs a semi-automatic process. We must point out that, since the user cannot understand stemming terms, we do not apply stemming our system (step 4 of Algorithm 2).

As described in the previous subsection, we consider each document as a transaction. Let us consider $T_D = \{d_1, \ldots, d_n\}$ as the set of transactions from the collection of documents D, and $I = \{t_1, \ldots, t_m\}$ as the text items obtained from all the representation documents $d_i \in D$ with their membership to the transaction expressed by $W_i = (w_{i1}, \ldots, w_{im})$. On this set of transactions we apply Algorithm 2 to extract the association rules. We must note that we do not distinguish in this algorithm the crisp and the fuzzy case, but we give general steps to extract association rules from text transactions. The specific cases will be given by the item weighting scheme that we consider in each case.

The whole process is detailed in the following:

Semi-automatic query refinement process using association rules:

(1) the user queries the system;
(2) a first set of documents is retrieved;
(3) from this set, the representation of documents is extracted following Algorithm 1 and association rules are generated following Algorithm 2 and the extraction rule procedure;
(4) terms that appear in certain rules are shown to the user (Section 5.1);
(5) the user selects those terms more related to her/his needs;
(6) the selected terms are added to the query, which is used to query the system again.

Algorithm 1. Basic algorithm to obtain the representation of documents in a collection

Input: a set of documents $D = \{d_1, \ldots, d_n\}$.
Output: a representation for all documents in D.
(1) Let $D = \{d_1, \ldots d_n\}$ be a collection of documents
(2) Extract an initial set of terms S from each document $d_i \in D$
(3) Remove stop words
(4) Apply stemming (*via* Porter's algorithm [40])
(5) The representation of d_i obtained is a set of keywords $S = \{t_1, \ldots, t_m\}$ with their associated weights $(w_{i1}, \ldots, w_{im})$

We must point out that, as it has been explained in [20,42], in the applications of mining techniques to text, documents are usually categorized, in the sense of documents which representation is a set of keywords, that is, terms that really describe the content of the document. This means that usually a full text is not considered and its

Algorithm 2. Basic algorithm to obtain the association rules from text

Input: a set of transactions $T_D = \{d_1, \ldots d_n\}$

a set of term items $I = \{t_1, \ldots, t_m\}$ with their associated weights $W_i = (w_{i1}, \ldots, w_{im})$ for each document d_i.

Output: a set of association rules.

(1) Construct the itemsets from the set of transactions T.
(2) Establish the threshold values of minimum support *minsupp* and minimum confidence *minconf*
(3) Find all the itemsets that have a support above threshold *minsupp*, that is, the *frequent itemsets*
(4) Generate the rules, discarding those rules below threshold minconf

description is not formed by all the words in the document, even without stop words, but also by keywords. The authors justify the use of keywords because of the appearing of useless rules. Some additional commentaries about this problem regarding the poor discriminatory power of frequent terms can be found in [39], where the authors comment the fact that the expanded query may result worst performance than the original one due to the poor discriminatory ability of the added terms.

Therefore, the problem of selecting good terms to be added to the query have two faces. On the one hand, if the terms are not good discriminators, the expansion of the query may not improve the result. But, on the other hand, in dynamic environments or systems where the response-time is important, the application of a pre-processing stage to select good discriminatory terms may not be suitable. In our case, since we are dealing with a problem of query refinement in Internet, information must be shown on-line to the user, so a time constraint is present.

Solutions for both problems can be given. In the first case, discriminatory schemes almost automatic can be used alternatively to a preprocessing stage for selecting the most discriminatory terms. This is the case of the TFIDF weighting scheme (see Section 4.1.1). In the second case, when we work in a dynamic environment, we have to remind that to calculate the term weights following the TFIDF scheme, we need to know the presence of a term in the whole collection, which limits in some way its use in dynamic collections, as usually occurs in Internet. Therefore, instead of improving document representation in this situation, we can improve the rule obtaining process. The use of alternative measures of importance and accuracy such as the ones presented in Section 3 is considered in this work in order to avoid the problem of non-appropriate rule generation.

Additionally to the representation of the documents by terms, an initial categorization of the documents can be available. In that case, the categories can appear as items to be included in the transactions with value $\{0, 1\}$ based on the membership of the document to that category. This way, the extracted rules not only provide additional terms to the query, but also information about the relation between terms and categories.

5.1. The selection of terms for query refinement

The extraction of rules is usually guided by several parameters such as the minimum support (*minsupp*), the minimum value of certainty factor (*mincf*), and the number of terms in the antecedent and consequent of the rule. Rules with support and certainty factor over the respective thresholds are called *strong rules*.

Strong rules identify dependence in the sense of non-trivial inclusion of the set of transactions where each itemset (set of terms in this case) appears. This information is very useful for us in order to refine the query. First, the minimum support restriction ensures that the rules apply to a significant set of documents. Second, the minimum accuracy restriction, though allowing for some exceptions, ensures that the inclusion holds to an important degree.

Once the strong association rules are extracted, the selection of useful terms for query refinement depends on the appearance in antecedent and/or consequent of the terms. Let us suppose that *qterm* is a term that appears in the query and let $term \in S$, $S_0 \subseteq S$. Some possibilities are the following:

- Rules of the form $term \Rightarrow qterm$ such that $qterm \Rightarrow term$ has low accuracy. This means that the appearance of *term* in a document "implies" the appearance of *qterm*, but the reciprocal does not hold significantly, i.e., $\Gamma_{\text{term}} \subseteq \Gamma_{q\text{term}}$ to some extent. Hence, we could suggest the word *term* to the user as a way to restrict the set of documents obtained with the new query.
- Rules of the form $S_0 \Rightarrow qterm$ with $S_0 \subseteq S$. We could suggest the set of terms S_0 to the user as a whole, i.e., to add S_0 to the query. This is again uninteresting if the reciprocal is a strong rule.
- Rules of the form $qterm \Rightarrow term$ with $term \in S$ and $term \Rightarrow qterm$ a not strong rule. We could suggest the user to replace *qterm* with *term* in order to obtain a set of documents that include the actual set (this is interesting if we are going to perform the query again in the web, since perhaps *qterm* is more specific that the user intended).
- Strong rules of the form $S_0 \Rightarrow qterm$ or $term \Rightarrow qterm$ such that the reciprocal is also strong. This means co-occurrence of terms in documents. Replacing *qterm* with S_0 (or *term*) can be useful in order to search for similar documents where *qterm* does not appear. This case could be also accomplished by using ratio rules with ratio 1:1, provided that there is a real lineal correlation between degrees (a goodness measure of the correlation is needed).

The utility of the rules can be improved if a previous categorization of the documents is available, and items meaning that the document is in a given category are employed in the document representation. Rules containing category labels can give us new information about the category itself. For instance, if a rule of the form $term \rightarrow category$ appears with enough accuracy, we can assert that documents where that term appears can be classified in that category.

6. An experimental example

To carry out the experimental stage, we have made an initial query to the search engine *Alltheweb* (*http://www.alltheweb.com*) with the search and results in Spanish. For the query terms, we have taken a short query (only one term with more than one meaning). The term query is '*fresas*' that translates to '*strawberries*' but also '*milling cutter*' in English. The purpose of this kind of query is to find additional terms that can broad the query but narrow the set of retrieved documents.

Table 1
Classes associated to the different meanings of 'fresas' in Spanish

Class	Description
Class I	Industrial (milling cutter)
Class F	Fruit (strawberry cultivation)
Class C	Recipes (how to cook strawberries)
Class M	Class F ∪ Class C
Class X	Others

Therefore, if the user retrieves a set of 100 documents with the term '*fresas*' with the intention of looking for the industrial tool and she/he does not know more vocabulary related to that concept, the resulting rules can suggest her/him some terms to add to the query. This new query can discard the documents related to other meanings (always that the additional terms are not in the vocabulary of the other meanings).

Moreover, a term with different meanings can retrieve several documents belonging to different categories. For instance, documents with the term '*fresas*' related to the fruit meaning would be in a category different from the meaning of the industrial tool. Even in the same concept, we can categorize again by separating those documents related to the strawberries as the fruit cultivation, production and market from those about recipes with strawberries.

From the more than 61.000 retrieved documents, we analyze the 100 top-ranked documents, which is our *local set*. After the application of Algorithm 1 (see Section 6), we obtain 832 terms. If we obtain the text transactions, we have 100 transactions with 832 items. We must point out the difference in the length of the dimensions of the set of transactions obtained. In traditional data mining, the number of transactions is usually greater while the number of items is lower. In our case it is the opposite, although the goodness of the rules has not to be affected.

In order to study the goodness of the rules connecting categories and terms, we have categorized the local set in five categories (see Table 1). The first one, noted by *Class I*, is composed by documents related to the meaning of the industrial milling cutter. The second one, noted by *Class M* is related to the meaning of strawberries as a fruit. Inside this category, we can distinguish between the meaning of strawberries as a product to cultivate and sell, noted by *Class F* and those documents related to recipes with strawberries, noted by *Class C*. Finally, the documents which are not related to any of these categories, but also have no relation among them, have been all categorized in the same class noted by *Class X*. Therefore, after adding these categories as items to the transactions, we have 837 items in each transaction.

Considering the possible weighting schemes we have proposed in Section 4.1.1, we can distinguish broadly between the crisp and the fuzzy case. This last case can have a frequency weighting scheme or a TFIDF weighting scheme. Based on the selected case, we obtain different numbers of rules applying Algorithm 2, without establishing a threshold for the confidence or the certainty measure. The level for all the cases is 5, which implies that the number of components appearing in the rule (antecedent and consequent) cannot be more than 5 adding both sides of the rule. In the boolean case, the number of rules extracted is 87 954 (with a support of 5%); in the case of the normalized frequency scheme, we obtained 68 rules (also with a support of 5%); and, in the case fuzzy TFIDF

Table 2
Rules obtained with different term weighting schemes

	Boolean	Normalized frequency	TFIDF
minsupp (%)	5	5	2
Number of rules	87 964	68	3686

scheme we obtained 3686 rules with a support of 2%. We decide to decrease the support in this last case because the number of obtained rules with a support of 5% was only 4 (see Table 2).

These results reveal one of the main advantages of the fuzzy approach: the selection of good terms, as we commented in Section 5.1. In the crisp case, the weight of an item in a transaction can be only 0 or 1. This means that, if a term appears only one time in a document but other term appears 10 times in the same document, both of them will have a weight of 1. This generates a huge number of rules that, on one hand does not reflect the real presence relation among terms in the documents, and on the other hand, overwhelms the user, who is not able to hand and understand so many rules. The fact that in the TFIDF scheme with a support of 5% only 4 rules have been obtained shows that, really, this scheme discard in some way those terms with a poor discriminatory power, so the terms appearing very frequently in the whole collection have a low weight. The principle of this scheme agrees with the selection of rules in the sense that those rules where a high frequent term appears do not give new information. For instance, those rules where the term *fresas* appear do not provide information about the relation of presence with the other terms in the rule, since the term *fresas* appears in all the documents (otherwise they would not been retrieved). When the TFIDF scheme is used, the term *fresas* is assigned a weight of 0, since it appears in all the documents of the collection. This means that no rule with the term *fresas* will appear in the set of extracted rules when the TFIDF weighting scheme is applied.

However, the terms appearing together with *fresas* in the same rule can decrease the number of documents retrieved. For instance, in the case of the frequency weighting scheme, the rule $frontales^{2} \rightarrow fresas$ appears with a certainty factor of 1. Although from the point of view of new information the interpretation of this rule does not provide anything new, from the point of view of reducing the number of documents, the term *frontales* can suggest to the user a new term related to the meaning of the industrial tool, which she/he did not know before due to a lack of vocabulary in the topic. Other rule that provides new vocabulary terms about the industrial tool with the same weighting scheme is, for instance, $herramientas^{3} \rightarrow fresas$ with a confidence of 70% and a certainty of 0.68. These results are shown in Tables 3 and 4, where the terms appearing in the antecedent of the rules are shown in the left column and the terms appearing in the consequent of the rules are shown in the first row of the table.

From the point of view of effectiveness in information retrieval, we can observe that the first query with *fresas* over the 100 top-ranked documents has a recall value of 1 and a precision value of 0.48. If we narrow the query by adding the term *frontales*, the recall changes to 0.12 and the precision changes to 1. This is reasonable since as the query is more specific, the precision value increases

[2] '*frontales*' in Spanish means '*profiles*'.
[3] '*herramientas*' in Spanish means '*tools*'.

Table 3
Support values of rules for the normalized frequency weighting scheme

	Fresas
Frontales	5.3
Herramientas	5.5

Table 4
Confidence/certainty factor values of rules for the normalized frequency weighting scheme

	Fresas	*Frontales*	*Herramientas*
Fresas	—	0.057/0.0048	0.057/0.0022
Frontales	1/1	—	—
Herramientas	0.7/0.68	—	—

Table 5
Support values of rules for the normalized frequency weighting scheme

	Class I
Frontales	5.3
Herramientas	6.65
Accesorios	5.05
Brocas	6.11

while the recall one decreases. The same phenomenon occurs when we add the term *herramientas* to the query. In this case, the precision value is 1 and the recall value is 0.2.

Regarding the categories, there are many rules where the class label appears in the antecedent and/or the consequent. These rules are quite interesting to know which terms are related to which categories (only when the class label appears in the consequent). For instance, in the frequency weighting scheme, the following rules with a certainty factor of 1 appear: $frontales \rightarrow Class\ I$, $herramientas \rightarrow Class\ I$, $accesorios$[4] $\rightarrow Class\ I$, $brocas$[5] $\rightarrow Class\ I$ (see Tables 5 and 6).

As for the accuracy measures, some results are counterintuitive when we compare the values or confidence and certainty, which reveals that when the rules relate two very frequent items, the confidence is quite high by the certainty is not. For instance, in the frequency weighting scheme, the rule $fresas \rightarrow Class\ I$ has a confidence of 0.47 while the certainty value is of 0.067. The results for this rule are shown in Tables 7 and 8.

[4] '*accesorios*' in Spanish means '*accesories*'.
[5] '*brocas*' in Spanish means '*drills*'.

Table 6
Confidence/certainty factor values of rules for the normalized frequency weighting scheme

	Class I	*Frontales*	*Herramientas*	*Accesorios*	*Brocas*
Class I	—	0.1/0.059	0.13/0.075	0.10/0.056	0.12/0.069
Frontales	1/1	—	—	—	—
Herramientas	1/1	—	—	—	—
Accesorios	1/1	—	—	—	—
Brocas	1/1	—	—	—	—

Table 7
Support values of rules for the normalized frequency weighting scheme

	Class I
Fresas	43.12

Table 8
Confidence/certainty factor values of rules for the normalized frequency weighting scheme

	Class I	*fresas*
Class I	—	0.88/0.8
Fresas	0.47/0.067	—

7. Conclusion and future work

We have presented an application of association rules to query refinement. The extension of classical association rules and transactions to the fuzzy framework and the definition of these concepts in the text framework allows to manage documents with a representation given by weighting schemes reflecting the presence of the term items in the text transactions. Rules of different forms are extracted from the set of transactions and a selection process of the most suitable rules following the user needs is carried out. The terms of these selected rules are shown to the user who chooses those more related to her/his preferences. If a previous categorization of the documents is available, relations among terms and categories can also be found. Further results show also the suitability of certainty as accuracy measure instead of the confidence.

As future research, we will develop an application for automatic query refinement and we will compare the results with other approaches to the same problem found in the literature. Further improvements of the system can be carried out such as to consider an intrinsical structure of the documents as in [27,38]. Another future work will be to consider image features as new terms to be added to the query to improve image retrieving processes via query expansion. Making a relation between image features and query terms [45], we may retrieve new images with similar features to those retrieved by the first query.

Acknowledgements

This work is supported by the research project Fuzzy-KIM, CICYT TIC2002-04021-C02-02.

References

[1] R. Agrawal, T. Imielinski, A. Swami, Mining association rules between set of items in large databases, in: Proc. 1993 ACM SIGMOD Conf., 1993, pp. 207–216.
[2] R. Attar, A.S. Fraenkel, Local feedback in full-text retrieval systems, J. Assoc. Comput. Mach. 24 (3) (1977) 397–417.
[3] W.H. Au, K.C.C. Chan, An effective algorithm for discovering fuzzy rules in relational databases, in: Proc. IEEE Internat. Conf. on Fuzzy Systems, vol. II, 1998, pp. 1314–1319.
[4] F. Berzal, I. Blanco, D. Sánchez, M.A. Vila, Measuring the accuracy and importance of association rules: a new framework, Intell. Data Anal. 6 (2002) 221–235.
[5] R.C. Bodner, F. Song, Knowledge-based approaches to query expansion in information retrieval, in: G. McCalla (Ed.), Advances in Artificial Intelligence, Springer, NY, USA, 1996, pp. 146–158.
[6] G. Bordogna, P. Carrara, G. Pasi, Fuzzy approaches to extend Boolean information retrieval, in: Bosc., J. Kacprzyk (Eds.), Fuzziness in Database Management Systems, Physica-Verlag, Germany, 1995, pp. 231–274.
[7] G. Bordogna, G. Pasi, A fuzzy linguistic approach generalizing Boolean information retrieval: a model and its evaluation, J. Amer. Soc. Inform. Sci. 44 (2) (1993) 70–82.
[8] C. Buckley, G. Salton, J. Allan, A. Singhal, Automatic query expansion using SMART: TREC 3, Proc. Third Text Retrieval Conf., NIST Special Publication, 500-225, Gaithersburg, MD, 1994, pp. 69–80.
[9] D.A. Buell, D.H. Kraft, Performance measurement in a fuzzy retrieval environment, in: Proc. Fourth Internat. Conf. on Information Storage and Retrieval, ACM/SIGIR Forum 16(1), Oakland, CA, USA, 1981, pp. 56–62.
[10] H. Chen, T. Ng, J. Martinez, B.R. Schatz, A concept space approach to addressing the vocabulary problem in scientific information retrieval: an experiment on the worm community system, J. Amer. Soc. Inform. Sci. 48 (1) (1997) 17–31.
[11] W.B. Croft, R.H. Thompson, I^3R: a new approach to the design of document retrieval systems, J. Amer. Soc. Inform. Sci. 38 (6) (1987) 389–404.
[12] A. De Luca, S. Termini, Entropy and energy measures of a fuzzy set, in: M.M. Gupta, R.K. Ragade, R.R. Yager (Eds.), Advances in Fuzzy Set Theory and Applications, vol. 20, 1979, pp. 321–338.
[13] M. Delgado, N. Marín, D. Sánchez, M.A. Vila, Fuzzy association rules: general model and applications, IEEE Trans. Fuzzy Systems 11 (2003) 214–225.
[14] M. Delgado, N. Marín, M.J. Martín-Bautista, D. Sánchez, M.A. Vila, Mining fuzzy association rules: an overview, 2003 BISC Internat. Workshop on Soft Computing for Internet and Bioinformatics, 2003.
[15] M. Delgado, M.J. Martín-Bautista, D. Sánchez, M.A. Vila, Mining text data: special features and patterns, in: Proc. EPS Exploratory Workshop on Pattern Detection and Discovery in Data Mining, London, September 2002.
[16] M. Delgado, M.J. Martín-Bautista, D. Sánchez, M.A. Vila, A probabilistic definition of a nonconvex fuzzy cardinality, Fuzzy Sets and Systems 126 (2) (2002) 41–54.
[17] M. Delgado, D. Sánchez, M.A. Vila, Fuzzy cardinality based evaluation of quantified sentences, Internat. J. Approx. Reason. 23 (2000) 23–66.
[18] D. Dubois, H. Prade, T. Sudkamp, A discussion of indices for the evaluation of fuzzy associations in relational databases, in: T. Bilgic, et al., (Eds.), Proc. IFSA 2003, Lectures Notes on Artificial Intelligence, vol. 2715, Springer, Berlin, Heidelberg, 2003, pp. 111–118.
[19] E. Efthimiadis, Query expansion, Ann. Rev. Inform. Systems Technol. 31 (1996) 121–187.
[20] R. Feldman, M. Fresko, Y. Kinar, Y. Lindell, O. Liphstat, M. Rajman, Y. Schler, O. Zamir, Text mining at the term level, in: Proc. 2nd European Symp. of Principles of Data Mining and Knowledge Discovery, 1998, pp. 65–73.
[21] R. Feldman, H. Hirsh, Mining associations in text in the presence of background knowledge, in: Proc. Second Internat. Conf. on Knowledge Discovery from Databases, 1996.
[22] S. Gauch, J.B. Smith, An expert system for automatic query reformulation, J. Amer. Soc. Inform. Sci. 44 (3) (1993) 124–136.

[23] D. Harman, Towards interactive query expansion, in: Proc. Eleventh Ann. Internat. ACMSIGIR Conf. on Research and Development in Information Retrieval, ACM Press, New York, 1998, pp. 321–331.
[24] M. Hearst, Untangling text data mining, in: Proc. 37th Annual Meeting of the Association for Computational Linguistics (ACL'99), University of Maryland, MA, USA, 1999.
[25] M. Hearst, Next generation web search: setting our sites, IEEE Data Engineering Bulletin 23 (3) (2000) 38–48.
[26] T.P. Hong, C.S. Kuo, S.C. Chi, Mining association rules from quantitative data, Intell. Data Anal. 3 (1999) 363–376.
[27] M.M. Jiang, S.S. Tseng, C.J. Tsai, Intelligent query agent for structural document databases, Expert Systems Appl. 17 (1999) 105–133.
[28] Y. Kodratoff, Knowledge discovery in texts: a definition, and applications, in: Z.W. Ras, A. Skowron (Eds.), Foundation of Intelligent Systems, Lectures Notes on Artificial Intelligence, vol. 1609, Springer, Berlin, 1999.
[29] F. Korn, A. Labrinidis, Y. Kotidis, C. Faloutsos, Quantifiable data mining using ratio rules, The VLDB J. 8 (2000) 254–266.
[30] D.H. Kraft, M.J. Martín-Bautista, J. Chen, M.A. Vila, Rules and fuzzy rules in text: concept, extraction and usage, Internat. J. Approx. Reasoning 34 (2003) 145–161.
[31] C.-M. Kuok, A. Fu, M.H. Wong, Mining fuzzy association rules in databases, SIGMOD Record 27 (1) (1998) 41–46.
[32] J.H. Lee, H.L. Kwang, An extension of association rules using fuzzy sets, in: Proc. IFSA'97, Prague, Czech Republic, 1997.
[33] S.H. Lin, C.S. Shih, M.C. Chen, J.M. Ho, M.T. Ko, Y.M. Huang, Extracting classification knowledge of internet documents with mining term associations: a semantic approach, in: Proc. ACM/SIGIR'98, Melbourne, Australia, 1998, pp. 241–249.
[34] M.J. Martín-Bautista, D. Sánchez, J.M. Serrano, M.A. Vila, Text mining using fuzzy association rules, in: V. Loia, M. Nikravesh, L.A. Zadeh (Eds.), Fuzzy Logic and the Internet, Studies in Fuzziness and Soft Computing, Springer-Verlag, Heidelberg, 2004, pp. 173–190.
[35] M.J. Martín-Bautista, D. Sánchez, M.A. Vila, H.L. Larsen, Measuring effectiveness in fuzzy information retrieval, in: H.L. Larsen, J. Kacprzyk, S. Zadrozny, T. Andreasen, H. Christiansen (Eds.), Flexible Query Answering Systems, Recent Advances, Proc. FQAS'2000: Advances in Soft Computing Series, Springer, Berlin, 2000, pp. 396–404.
[36] G. Miller, WordNet: an on-line lexical database, Internat. J. Lexicogr. 3 (4) (1990) 235–312.
[37] M. Mitra, A. Singhal, C. Buckley, Improving automatic query expansion, in: Proc. ACM SIGIR, Melbourne, Australia, 1998, pp. 206–214.
[38] A. Moliniari, G. Pasi, A fuzzy representation of HTML documents for information retrieval system, Proc. Fifth IEEE Internat. Conf. on Fuzzy Systems, vol. I, New Orleans, EEUU, 1996, pp. 107–112.
[39] H.P. Peat, P. Willet, The limitations of term co-occurrence data for query expansion in document retrieval systems, J. Amer. Soc. Inform. Sci. 42 (5) (1991) 378–383.
[40] M.F. Porter, An algorithm for suffix stripping, Program 14 (3) (1980) 130–137.
[41] Y. Qui, H.P. Frei, Concept-based query expansion, in: Proc. Sixteenth Ann. Internat. ACM-SIGIR'93 Conf. on Research and Development in Information Retrieval, 1993, pp. 160–169.
[42] M. Rajman, R. Besançon, Text mining: natural language techniques and text mining applications, in: Proc. Third Internat. Conf. on Database Semantics (DS-7), IFIP Proceedings Serie, Chapman & Hall, London, 1997.
[43] G. Salton, C. Buckley, Term weighting approaches in automatic text retrieval, Inform. Process. Manage. 24 (5) (1988) 513–523.
[44] G. Salton, M.J. McGill, Introduction to Modern Information Retrieval, McGraw-Hill, New York, 1983.
[45] D. Sánchez, J. Chamorro-Martínez, M.A. Vila, Modelling subjectivity in visual perception of orientation for image retrieval, Inform. Process. Manage. 39 (2) (2003) 251–266.
[46] E. Shortliffe, B. Buchanan, A model of inexact reasoning in medicine, Math. Biosci. 23 (1975) 351–379.
[47] C. Silverstein, S. Brin, R. Motwani, Beyond market baskets: generalizing association rules to dependence rules, Data Mining Knowledge Discovery 2 (1998) 39–68.
[48] P. Srinivasan, M.E. Ruiz, D.H. Kraft, J. Chen, Vocabulary mining for information retrieval: rough sets and fuzzy sets, Inform. Process. Manage. 37 (2001) 15–38.
[49] C.J. Van Rijsbergen, D.J. Harper, M.F. Porter, The selection of good search terms, Inform. Process. Manage. 17 (1981) 77–91.

[50] B. Vélez, R. Weiss, M.A. Sheldon, D.K. Gifford, Fast and effective query refinement, in: Proc. 20th ACM Conf. on Research and Development in Information Retrieval (SIGIR'97), Philadelphia, Pennsylvania, 1997.
[51] E. Voorhees, Query expansion using lexical-semantic relations, Proc. 17th Internat. Conf. on Research and Development in Information Retrieval (SIGIR), Dublin, Ireland, July 1994.
[52] J. Xu, W.B. Croft, Query expansion using local and global document analysis, in: Proc. 19th Ann. Internat. ACM SIGIR Conference on Research and Development in Information Retrieval, 1996, pp. 4–11.
[53] L.A. Zadeh, A computational approach to fuzzy quantifiers in natural languages, Comput. Math. Appl. 9 (1) (1983) 149–184.