Term Weighting for Information Retrieval Using Fuzzy Logic

Jorge Ropero, Ariel Gómez, Alejandro Carrasco,
Carlos León and Joaquín Luque
*Department of Electronic Technology, University of Seville,
Spain*

1. Introduction

The rising quantity of available information has constituted an enormous advance in our daily life. However, at the same time, some problems emerge as a result from the existing difficulty to distinguish the necessary information among the high quantity of unnecessary data. Information Retrieval has become a capital task for retrieving the useful information. Firstly, it was mainly used for document retrieval, but lately, its use has been generalized for the retrieval of any kind of information, such as the information contained in a database, a web page, or any set of accumulated knowledge. In particular, the so-called Vector Space Model is widely used. Vector Space Model is based on the use of index terms, which represent some pieces of knowledge or Objects. Index terms have associated weights, which represent the importance of them in the considered set of knowledge.

It is important that the assignment of weights to every index term - called Term Weighting - is automatic. The so-called TF-IDF method is mainly used for determining the weight of a term (Lee et al., 1997). Term Frequency (TF) is the frequency of occurrence of a term in a document; and Inverse Document Frequency (IDF) varies inversely with the number of documents to which the term is assigned (Salton, 1988). Although TF-IDF method for Term Weighting has worked reasonably well for Information Retrieval and has been a starting point for more recent algorithms, it was never taken into account that some other aspects of index terms may be important for determining term weights apart from TF and IDF: first of all, we should consider the degree of identification of an object if only the considered index term is used. This parameter has a strong influence on the final value of a term weight if the degree of identification is high. The more an index term identifies an object, the higher value for the corresponding term weight; secondly, we should also consider the existance of join terms.

These aspects are especially important when the information is abundant, imprecise, vague and heterogeneous. In this chapter, we define a new Term Weighting model based on Fuzzy Logic. This model tries to replace the traditional Term Weighting method, called TF-IDF. In order to show the efficiency of the new method, the Fuzzy Logic-based method has been tested on the website of the University of Seville. Web pages are usually a perfect example of heterogeneous and disordered information. We demonstrate the improvement introduced by the new method extracting the required information. Besides, it is also possible to extract related information, which may be of interest to the users.

2. Vector Space Model and Term Weighting

In the Vector Space Model, the contents of a document are represented by a multidimensional space vector. Later, the proper classes of the given vector are determined by comparing the distances between vectors. The procedure of the Vector Space Model can be divided into three stages, as seen in Figure 1 (Raghavan & Wong, 1986):

- The first step is document indexing, when most relevant terms are extracted.
- The second stage is based on the introduction of weights associated to index terms in order to improve the retrieval relevant to the user.
- The last stage classifies the document with a certain measure of similarity.

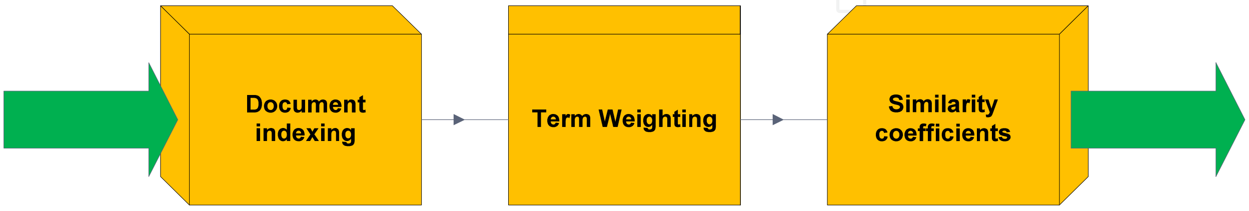


Fig. 1. Vector Space Model procedure

In this chapter, we are focusing in the second stage. It was in the late 50's when the idea of text retrieval came up - a concept that was later extended to general information retrieval -. Text retrieval was founded on an automatic search based on textual content through a series of identifiers. It was Gerard Salton who laid the foundations for linking these identifiers and the texts that they represent during the 70's and the 80's. Salton suggested that every document could be represented by a term vector in the way D = (t_i, t_j,…, t_p), where every t_k identifies a term assigned to a document D. A formal representation of the vector D leads us not to consider only the terms in the vector, but to add a set of weights representing the term weight, it is to say, its importance in the document.

A Term Weighting system should improve efficiency in two main factors, recall and precision. Recall takes into account the fact that the objects relevant to the user should be retrieved. Precision considers the fact that the objects that are not wanted by the user should be rejected. In principle, it is desirable to build a system that rewards both high recall, - retrieving all that is relevant - and high precision - discarding all unwanted objects (Ruiz & Srinisavan, 1998). Recall improves using high-frequency index terms, i.e. terms which occur in many documents of the collection. This way, it is expected to retrieve many documents including such terms, and thus, many of the relevant documents. The precision factor, however, improves when using more specific index terms that are capable of isolating the few relevant articles of the mass of irrelevant. In practice, compromises are utilized; using frequent enough terms to achieve a reasonable level of recall without causing a too low value of precision. The exact definitions of recall and precision are shown in Equations 1 and 2.

$$\text{Recall} = \frac{\textit{retrieved relevant objects}}{\textit{total number of relevant objects}}$$

Equation 1. Definition of recall

$$\text{Precision} = \frac{\textit{retrieved relevant objects}}{\textit{total number of retrieved objects}}$$

Equation 2. Definition of precision

So firstly, terms that are mentioned frequently in individual documents or extracts from a document, appear to be useful for improving recall. This suggests the use of a factor known as Term Frequency (TF) as part of a Term Weighting system, measuring the frequency of occurance of a term in a document. The TF factor has been used for Term Weighting for years in automatic indexing environments. Secondly, the TF factor solely does not ensure an acceptable retrieval. In particular, when the high frequency terms are not concentrated in specific documents, but instead are frequent in the entire set, all documents tend to be recovered, and this affects the precision factor. Thus, there is the need to introduce a new factor that favours the terms that are concentrated in only a few documents in the collection. The Inverse Document Frequency (IDF) is the factor that considers this aspect. The IDF factor is inversely proportional to the number of documents (n) to which a term is assigned in a set of documents N. A typical IDF factor is log (N / n) (Salton & Buckley, 1996). So the best index terms to identify the contents of a document are those able to distinguish certain individual documents from the rest of the set. This implies that the best terms should have high term frequencies, but low overall collection frequencies. A reasonable measure of the importance of a term can be obtained, therefore, by the product of term frequency and inverse document frequency (TF x IDF). It is usual to describe the weight of a term *i* in a document *j* as shown in Equation 3.

$$w_{ij} = tf_{ij} \times idf_j$$

Equation 3. Obtention of term weights; general formula

This formula was originally designed for the retrieval and extraction of documents. Eventually, it has also been used for the retrieval of any object in any set of accumulated knowledge, and has been revised and improved by other authors in order to obtain better results in Information Retrieval (Lee et al., 1997), (Zhao & Karypis, 2002), (Lertnattee & Theeramunkong, 2003), (Liu & Ke, 2007).

In short, term weights must be related somehow to the importance of an index term in the corresponding set of knowledge. There are two options for defining these weights:

- The evaluation of the weights by an expert in the field. This is based on his own perception of the importance of index terms. This method is simple, but it has the disadvantage of relying solely on the criterion of the engineer of knowledge, it is very subjective and is not able of being automated.
- Automated generation of weights using a set of rules. The most widely used method for Term Weighting, as said above, is the TF-IDF method. In this chapter, we propose a novel Fuzzy Logic-based Term Weighting method, which obtains better results for Information Retrieval.

To calculate the weight of a term, the TF-IDF approach considers two factors:

- TF: Frequency of occurrence of the term in the document. So tf_{ik} is the frequency of occurrence of the term T_k in document i.

- IDF: varies inversely with the number of documents n_k where the term T_k has been assigned in a set of N documents. The typical IDF factor is represented by the expression log (N / n_k + 0.01).

Introducing standardization to simplify the calculations, the formula finally obtained for the calculation of the weights is defined in Equation 4 (Liu et al., 2001)

$$W_{ik} = \frac{tf_{ik} \times \log(N / n_k + 0.01)}{\sqrt{\sum_{k=1}^{m} tf_{ik} \times \log(N / n_k + 0.01))^2}}$$

Equation 4. Obtention of term weights. Used formula.

A third factor that is commonly used is the document length normalization factor. Long documents usually have a much larger set of extracted terms than short documents. This fact makes it more likely that long documents are retrieved (Van Rijsbergen, 1979), (Salton & Buckley, 1996). The term weight obtained using a length normalization factor is given by Equation 5.

$$W_{ik} = \frac{w_{ik}}{\sqrt{\sum_{i=1}^{m} (w_i)^2}}$$

Equation 5. Obtention of term weights using a length normalization factor

In Equation 5, w_i correspond to the weights of the other components of the vector.

All Term Weighting tasks are shown in Figure 2.

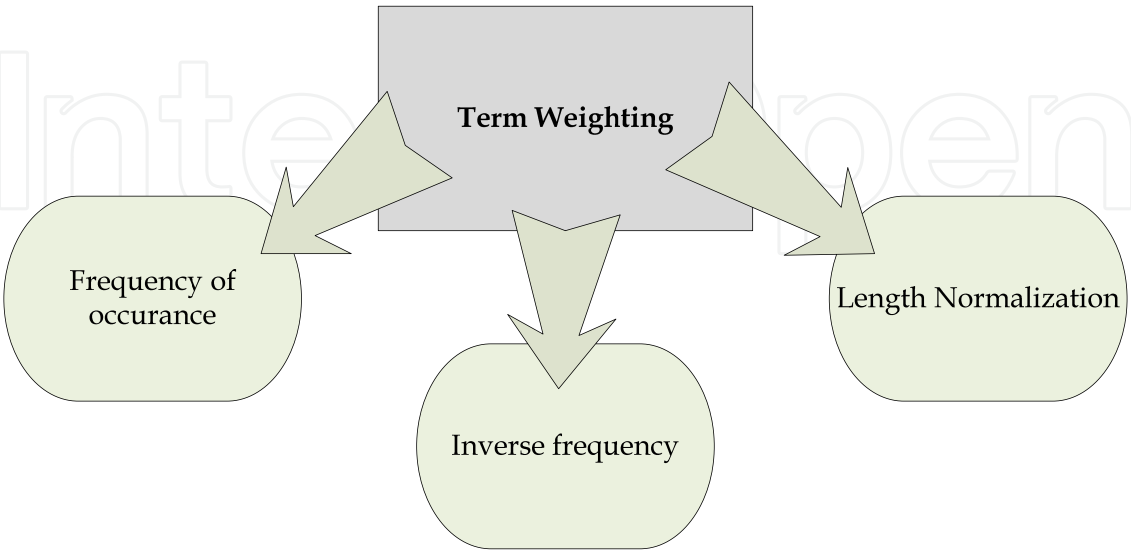


Fig. 2. Term Weighting tasks

3. Term Weighting method comparison

3.1 Term Weighting methods

The TF-IDF method works reasonably well, but has the disadvantage of not considering two aspects that we believe key:

- The first aspect is the degree of identification of the object if a determined index term is solely used in a query. This parameter has a strong influence on the final value of a weight of term if the degree of identification is high. The more a term identifies an object, a higher value has its correspondent weight. However, this parameter creates two disadvantages in terms of practical aspects when a systematic, automated Term Weighting scheme is neccessary. On the one hand, the degree of identification is not deductible from any feature of the index term, so it must be specified by the Knowlegde Engineer. The assigned values may therefore be subjective, not systematic and not univocal. On the other hand, the same index term may have a different relationship with different objects.
- The second aspect is related to the join index terms, i.e. terms that are linked to others. Join terms have lower weights as the fact that these keywords are linked is what really determines the principal object. The appearance of one of these words could refer to another object.

This chapter describes, firstly, the operation of TF-IDF method. Then, the new Term Weighting Fuzzy Logic-based method is introduced. Finally, both methods are implemented for the particular case of Information Retrieval for the University of Seville web portal, obtaining specific results of the operation of both of them. A web portal is a typical example of a disordered, vague and heterogenous set of knowledge. With this aim, an intelligent agent was designed to allow an efficient retrieval of the relevant information. This system should be valid for any set of knowledge. The system was designed to enable users to find possible answers to their queries in a set of knowledge of a great size. The whole set of knowledge was classified into different objects. These objects represent the possible answers to user queries and were organized into hierarchical groups (called Topic, Section and Object). One or more standard questions are assigned to every object and some index terms are extracted from them.

The last step is Term Weigthing; the assigned weight depends on the importance of an index term for the identification of the object. The way in which these weights are assigned is the main issue of this chapter. All the process is shown in Figure 3.

As an example of the classical TF-IDF Term Weighting method functioning, we are using the term 'library', used in the example shown in Table 1.

At Topic hierarchic level:

- 'Library' appears 6 times in Topic 6 (tf_{ik} = 6, K=6).
- 'Library' appears 10 times in other Topics (n_k = 3)
- There are 12 Topics in total (N=12) - for normalizing, it is only necessary to know the other tf_{ik} and n_k for the Topic-.
- Substituting, W_{ik} = 1.00.

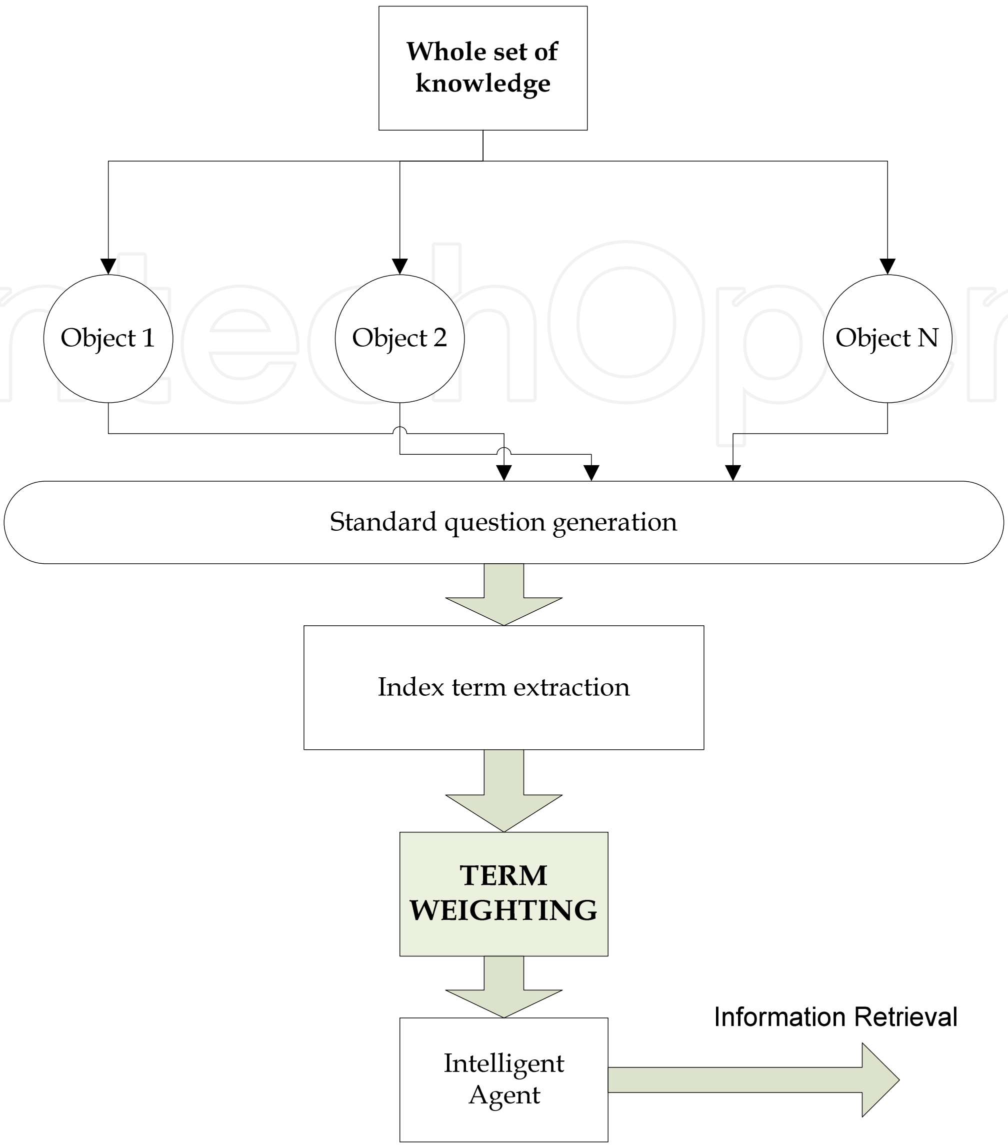


Fig. 3. Information Retrieval process.

As well, an example of the followed methodology is shown in Table 1.

STEP	EXAMPLE
Step 1: Web page identified by standard/s question/s	- Web page: http://bib.us.es/index-ides-idweb.html - Standard question: What online services are offered by the Library of the University of Seville?
Step 2: Locate standard/s question/s in the hierarchical structure.	Topic 6: Library Section 3: Online services Object 1.
Step 3: Extract index terms	Index terms: 'Library', 'services', 'online'
Step 4: Term weighting	Explained below

Table 1. Example of the followed methodology.

At Section hierarchic level:

- ‘Library’ appears 6 times in Section 6.3 (tf_{ik} = 6, K= 3)
- ‘Library’ appears 4 times in other Sections in Topic 6 (n_k = 6)
- There are 6 Sections in Topic 6 (N=6).
- Substituting, W_{ik} = 0.01. In fact, ‘Library’ appears in most of the Sections in Topic 6, so it is not very relevant to distinguish the desired Section inside the Topic.

At Object hierarchic level:

- ‘Library’ appears once in Object 6.3.1 (tf_{ik} = 1, K = 1). – Logically a term can only appear once in an Object -.
- ‘Library’ appears 3 times in other Topics (n_k = 3).
- There are 4 Objects in Section 6.3 (N=3).
- Substituting, W_{ik} = 0.01.

Consequently, ‘Library’ is relevant to find out that the Object is in Topic 6, but not very relevant to find out the definite Object, which should be found according to other terms in a user consultation.

As said above, TF-IDF has the disadvantage of not considering the degree of identification of the object if only the considered index term is used and the existance of join terms. The FL-based method provides a solution for these problems: the solution is to create a table of all the index terms and their corresponding weights for each object. This table will be created in the process of extracting the index words from the standard questions. Imprecision practically does not affect the method due to the fact that Term Weighting is based on fuzzy logic. This fact minimizes the effect of possible variations of the assigned weights.

Furthermore, the Fuzzy Logic-based method provides two important advantages:

- Term Weighting is automatic.
- The level of expertise required is much lower. Moreover, there is no need for an operator of any kind of knowledge about Fuzzy Logic, but only has to know how many times an index term appears in a certain subset and the answer to two simple questions:
- • How does an index term define an object by itself?
- • Are there any join terms tied to the considered index term?

For example, in the case of a website, the own web page developer may define standard questions. These questions are associated with the object - the web page -. He also should define the index for each object and answer the two questions proposed above. This greatly simplifies the process and leaves the possibility of using collaborative intelligence.

Fuzzy Logic based Term Weighting method is defined below. Four questions must be answered to determine the weight of an Index Term:

- Question 1 (Q1): How often does an index term appear in other subsets? - Related to IDF factor -.
- Question 2 (Q2): How often does an index term appear in its own subset? - Related to TF factor -.
- Question 3 (Q3): Does an index term undoubtedly define an object by itself?
- Question 4 (Q4): Is an index term joined to another one?

With the answers to these questions, a set of values is obtained. These values are the inputs to a fuzzy logic system, a Term Weight Generator. The Fuzzy Logic system output sets the weight of an index term for each hierarchical level (Figure 4).

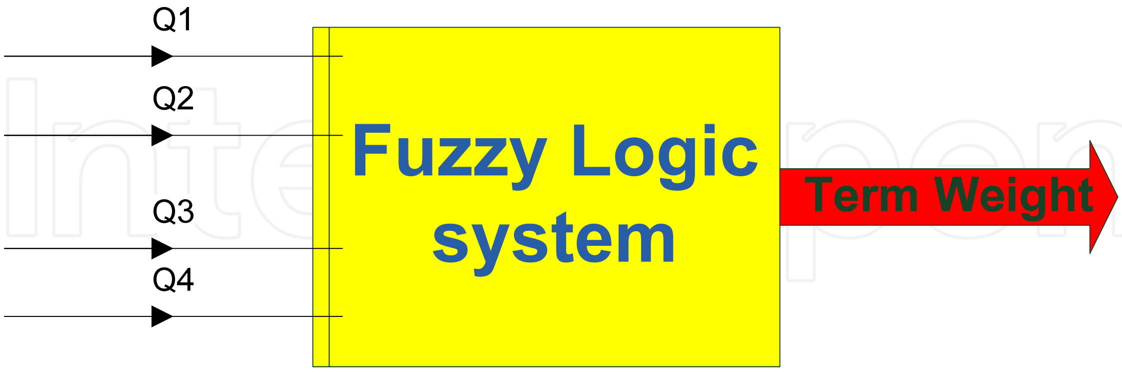


Fig. 4. Term Weighting using Fuzzy Logic.

Next it is described how to define the system input values associated with each of the four questions (Qi). Qi are the inputs to the Fuzzy Logic system

Question 1

Term weight is partly associated to the question 'How often does an index term appear in other subsets?'. It is given by a value between 0 – if it appears many times – and 1 - if it does not appear in any other subset -. To define weights, we are considering the times that the most used terms in the whole set of knowledge appear. The list of the most used index terms is shown in Table 2.

Number of order	Index term	Number of appearances in the accumulated set of knowledge
1	Service	31
2	Services	18
3	Library	16
4	Research	15
5	Address	14
	Student	14
7	Mail	13
	Access	13
9	Electronic	12
	Computer	12
	Resources	12
12	Center	10
	Education	10
	Registration	10
	Program	10

Table 2. List of the most used words.

Provided that there are 1114 index terms defined in our case, we think that 1 % of these words must mark the border for the value 0 (11 words). Therefore, whenever an index term appears more than 12 times in other subsets, we will give it the value of 0. Associated values for every Topic are defined in Table 3.

Number of appearances	0	1	2	3	4	5	6
Associated value	1	0.9	0.8	0.7	0.64	0.59	0.53
Number of appearances	7	8	9	10	11	12	≥13
Associated value	0.47	0.41	0.36	0.3	0.2	0.1	0

Table 3. Input values associated to Q1 for topic hierarchic level.

Between 0 and 3 times appearing - approximately a third of the possible values - , we consider that an index term belongs to the so called HIGH set. Therefore, it is defined in its correspondant fuzzy set with uniformly distributed values between 0.7 and 1, as may be seen in Figure 5. Analogously, we distribute all values uniformly according to different fuzzy sets. Fuzzy sets are defined by linguistic variables LOW, MEDIUM and HIGH. Fuzzy sets are triangular, on one hand for simplicity and on the other hand because we tested other more complex types of sets (Gauss, Pi type, etc), but the results did not improve at all.

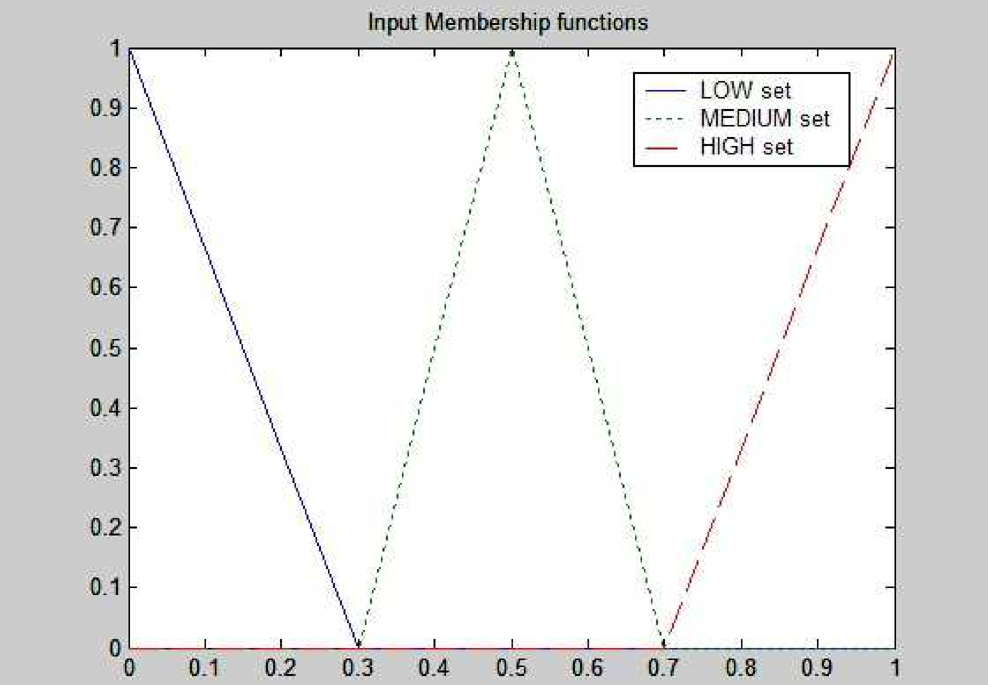


Fig. 5. Input fuzzy sets.

On the other hand, given that at each hierarchical level, a different term weight is defined, it is necessary to consider other scales to calculate the fuzzy system input values for the other hierarchical levels. As for the level of topic was considered the top level - the whole set of knowledge - , for the level of Section we consider the number occurrences of an index term on a given topic. Keeping in mind that all topics are considered, we take as reference the value of the topic in which the index term appears more often. The process is analogous to the above described, obtaining the values shown in Table 4.

Number of appearances	0	1	2	3	4	5	≥ 6
Associated value	1	0.7	0.6	0.5	0.4	0.3	0

Table 4. Input values associated to Q1 for section hierarchic level.

To find the term weight associated with the object level, the method is slightly different. It is also based on the definition of fuzzy sets, but we do not take into account the maximum number of words per secion, but the value associated to Q1 directly passes the border between fuzzy sets when the number of objects in which it appears increases in one unit, as seen in Table 5.

Number of appearances	0	1	2	≥ 3
Associated value	1	0.7	0.3	0

Table 5. Input values associated to Q1 for object hierarchic level.

Question 2

To find the imput value to the FL system of FL with question 2, the reasoning is analogous to the one for Q1, Though, we only have to consider the frequency of occurance of an index term within a single subset of knowledge, and not the frequency of occurrence in other subsets. Logically, the more times a term appears in a subset, the greater the probability that the query is related to it. Question Q2 corresponds to the TF factor.

Looking again at the list of index terms used in a topic, we obtain the values shown in Tables 6 and 7. It has been taken into account that the more times an index term appears in a topic or section, the greater should be the input value. These tables correspond to the values for the hierarchical levels of Topic and Section, respectively.

Number of appearances	1	2	3	4	5	≥ 6
Associated value	0	0.3	0.45	0.6	0.7	1

Table 6. Input values associated to Q2 for topic hierarchic level.

Number of appearances	1	2	3	4	5	≥ 6
Associated value	0	0.3	0.45	0.6	0.7	1

Table 7. Input values associated to Q2 for section hierarchic level.

Q2 is meaningless to determine the input value for the last hierarchical level. At this level, an index term appears only once on every object.

Question 3

For Question 3, the answer is completely subjective. In this chapter, we propose the values "Yes", "Rather" and "No". Table 8, shows the input values associated with Q3. This value is independent of hierarchical level.

Answer (Does the term itself define the Object?)	Yes	Rather	No
Associated value	1	0.5	0

Table 8. Input values associated to Q3.

For example, the developer of a web page would only have to answer "Yes", "Rather" or "No" to Question 3, without complicated mathematical formulas to describe it.

Question 4

Finally, question 4 deals with the number of index terms joined to another one. If an index term is joined to another one, its weight is lower. This is due to the fact that the term must be a join term to refer to the object in question. We propose term weight values for this question in Table 9. Again, the values 0.7 and 0.3 are a consequence of considering the border between fuzzy sets.

Joined terms to an index term	0	1	2	≥ 3
Associated value	1	0.7	0.3	0

Table 9. Input values associated to Q3.

After considering all these factors, fuzzy rules must be defined. In the case of Topic and Section hierarchical levels, we must consider the four input values that are associated with questions Q1, Q2, Q3 and Q4. Four output fuzzy sets have been also defined: HIGH, MEDIUM-HIGH, MEDIUM-LOW AND LOW. For the definition of the fuzzy rules for the Term Weighting system, we have used basically the following criteria:

- A high value of Q1 (IDF-related factor) implies that the term is not very present in other sets of knowledge. In this case, the output will be high, unless the term itself has very little importance (low Q3) or it is joined to many terms (low Q4).
- A high value of Q2 (TF-related factor), usually implies a high output value, since the index term is very present in a set of knowledge. However, if Q1 has a low value means that the term is present throughout the whole set of knowledge, so it is not very useful for extracting information.
- Q3 is a very important parameter, since if one term defines itself very well to a particular object, it is much easier to find the object.
- A low value of Q4 makes an index term less important, since it is associated with other terms. This fact causes a lower output value.

The combination of the four inputs and the three input fuzzy sets provides 81 possible combinations, which are summarized in Table 10.

In the object level (the last hierarchic level), Question 2 is discarded. Therefore, there is a change in the rules, although the criteria for the definition of fuzzy rules are similar to the previous case. An input less reduces the number of rules to twenty seven.

3.2 Example of the followed methodology

An example of the followed methodology is shown below. A comparision with the classical TF-IDF is done, starting from the definition of an object in the database of the Web portal of the University of Seville. The following example shows the difference between applying the TF-IDF method and applying the Fuzzy Logic-based one.

Rule number	Rule definition	Output
R1	IF Q1 = HIGH and Q2 ≠ LOW	At least MEDIUM-HIGH
R2	IF Q1 = MEDIUM and Q2 = HIGH	At least MEDIUM-HIGH
R3	IF Q1 = HIGH and Q2 = LOW	Depends on other Questions
R4	IF Q1 = HIGH and Q2 = LOW	Depends on other Questions
R5	IF Q3 = HIGH	At least MEDIUM-HIGH
R6	IF Q4 = LOW	Descends a level
R7	IF Q4 = MEDIUM	If the Output is MEDIUM-LOW, it descends to LOW
R8	IF (R1 and R2) or (R1 and R5) or (R2 and R5)	HIGH
R9	In any other case	MEDIUM-LOW

Table 10. Fuzzy rules.

In the Web portal database, Object 6.3.1 (http://bib.us.es/index-ides-idweb.html) is defined by the following standard question:

What online services are offered by the Library of the University of Seville?

If we consider the term ‘library’:

At Topic hierarchic level:

- ‘Library’ appears 6 times in other Topics in the whole set of knowledge, so that the value associated to Q1 is 0.53.
- ‘Library’ appears 10 times in Topic 6, so that the value associated to Q2 is 1.
- The response to Q3 is ‘Rather’ in 7 of the 10 times and ‘No’ in the other three, so that the value associated to Q3 is a weighted average: (7*0.5 + 3*0)/10 = 0.35.
- Term ‘Library’ is tied to one term 7 times and it is tied to two terms once. Therefore, the average is 1.1 terms. A linear extrapolation leads to a value associated to Q4 of 0.66.
- With all the values as inputs for the fuzzy logic engine, we obtain a term weight of 0.56.

At Section hierarchic level:

- ‘Library’ appears 6 times in other Sections corresponding to Topic 6, so that the value associated to Q1 is 0.
- ‘Library’ appears 4 times in Topic 6, so that the value associated to Q2 is 0.6.
- The response to Q3 is ‘Rather’ in three of the four cases, so that the value associated to Q3 is (3*0.5 + 1*0)/4 = 0.375.
- Term ‘Library’ is tied to one term 5 times and it is tied to two terms once so that the value associated to Q4 is 0.63.

- With all the values as inputs for the fuzzy logic engine, we obtain a term weight of 0.13.

At Object hierarchich level:

- 'Library' appears 3 times in other Objects corresponding to Section 6.3, so that the value associated to Q1 is 0.
- The response to Q3 is 'Rather', so that the value associated to Q3 is 0.5.
- Term 'Library' is tied to one term twice and it is tied to two terms once so that the value associated to Q4 is 0.57.
- With all the values as inputs for the fuzzy logic engine, we obtain a term weight of 0.33.

A summary of the values for the index term 'library' is shown in Table 11.

Hierarchic levels		Q1 value	Q2 value	Q3 value	Q4 value	Term Weight
Topic level (Topic 6)	**TF-IDF Method**	-	-	-	-	1
	Fuzzy Logic-based method	0.53	1	0.35	0.66	0.56
Section level (Section 3)	**TF-IDF Method**	-	-	-	-	0.01
	Fuzzy Logic-based method	0	0.6	0.375	0.63	0.13
Object level (Object 1)	**TF-IDF Method**	-	-	-	-	0.01
	Fuzzy Logic-based method	0	-	0.5	0.57	0.33

Table 11. Comparison of Term Weight values.

We may see the difference with the corresponding weight for the TF-IDF method - a value W_{ik} = 0.01 had been obtained), but this is just what we were looking for: not only the desired object is found, but also the ones that are more closely related to it. The word 'library' has a small weight for the TF-IDF method because it can not distinguish between the objects of Section 6.3. However, in this case all the objects will be retrieved, as they are interrelated. The weights of other terms determine the object which has a higher level of certainty.

4. Tests and results

4.1 General tests

Tests were held on the website of the University of Seville. 253 objects were defined, and grouped in a hierarchical structure, with 12 topics. Every topic has a variable number of sections and objects. From these 253 objects, 2107 standard questions were extracted. More than half of them were not used for these tests, as they were similar to others and did not contribute much to the results. Finally, the number of standard questions used for the tests was 914. Also, several types of standard questions were defined.

Depending on the nature of the considered object, we defined different types of standard questions, such as:

- A single primary standard question, which is the one that best defines an object. This question must always be associated to every object, the others types of standard questions are optional.
- Standard questions that take into account synonyms of some of the index terms used in the main standard question (e.g., "report" as a synonym for "document").We have called them synonim standard questions.
- Standard questions that take into account that a user may search for an object, but his question may be inaccurate or may be he does not know the proper jargon (e.g., "broken table" for "repairing service"). We have called them imprecise standard questions.
- Standard questions that are related to the main question associated with the object, but are more specific (e.g., "I'd like to find some information about the curriculum of Computer Science" to "I'd like to find some information about the curriculum for the courses offered by the University of Seville "). We have called them specific standard questions.
- Standard questions created by a feedback system. Most frequent user queries may be used.

For our tests, we considered the types of standard questions shown in Table 12.

Type of standard question	**Number of questions**
Main standard questions	252
Synonim standard questions	308
Imprecise standard questions	125
Specific standard questions	229
Feedback standard questions	0
Total standard questions	**914**

Table 12. Types of standard questions.

The standard questions were used as inputs in a Fuzzy Logic-based system. The outputs of the system are the objects with a degree a certainty greater than a certain threshold. To compare results, we considered the position in which the correct answer appears among the total number of answers identified as probable.

First of all, we shall define the thresholds to overcome in the Fuzzy Logic system. Thus, topics and sections that are not related to the object to be identified are removed. This is one of the advantages of using a hierarchical structure. Processing time is better as many subsets of knowledge are discarded. Anyway, it is desirable not to discard too many objects, in order to also obtain the related ones. The ideal is to retrieve between one and five answers for the user. The results of the consultation were sorted in 5 categories:

- Category Cat1: the correct answer is retrieved as the only answer or it is the one that has a higher degree of certainty between the answers retrieved by the system.
- Category Cat2: The correct answer is retrieved between the three answers with higher degree of certainty -excluding the previous case -.
- Category Cat3: The correct answer is retrieved between the five answers with higher degree of certainty - excluding the previous cases -.
- Category Cat4: The correct answer is retrieved, but not between the five answers with higher degree of certainty.
- Category Cat5: The correct answer is not retrieved by system.

Results are shown in Table 13

Method	Cat1	Cat2	Cat3	Cat4	Cat5	Total
TF-IDF method	466 (50.98%)	223 (24.40%)	53 (5.80%)	79 (8.64%)	93 (10.18%)	914
FL-based method	710 (77.68%)	108 (11.82%)	27 (2.95%)	28 (3.06%)	41 (4.49%)	914

Table 13. Information Retrieval results of using both Term Weighting methods.

The results obtained with the TF-IDF method are quite reasonable. 81.18% of the objects are retrieved among the top 5 choices and more than half of the objects are retrieved in the first place, Fuzzy Logic-based method is clearly better. 92.45% of the objects are retrieved and more than three-quarters are retrieved in the first place.

4.2 Tests according to the type of standard questions

In order to refine the conclusions about both Term Weighting methods, it is important to make a more thorough analysis of the results. We submitted to both Term Weighting methods to a comprehensive analysis according to the type of standard question. Results are shown in the Table 14.

According to the results, the TF-IDF method works relatively well considering the number of objects retrieved. Though, the Fuzzy Logic-based method is more precise, retrieving 91.67% of the objects in the first place. On the other hand, good results for this type of questions are logical, since questions correspond to supposedly well-made user queries.

For synonymous standard questions, the conclusions are similar: the results obtained using the Fuzzy Logic-based method are better than those achieved with TF-IDF method, especially in regard to precision. Though, the TF-IDF method also ensures good results. However, queries are not precise, so the performance is worse for the TF-IDF method than it is for the Fuzzy Logic-based method. This fact gives an idea of fuzzy logic as an ideal tool for adding more flexibility to the system. Anyway, the results are quite similar to those obtained for the main standard questions. They are only slightly worse, since synonim standard questions are similar to the main standard questions.

The difference is even more noticeable in regard to imprecise standard questions and specific standard questions. Imprecise standard questions are detected nearly as well as the main standard questions in the case of Fuzzy Logic-based method. This is another reason to confirm the appropriateness of using Fuzzy Logic. As for the specific standard questions, we

Type of standard question		Cat1	Cat2	Cat3	Cat4	Cat5	Total
Main standard questions	TF-IDF Method	171 (67.86%)	58 (23.02%)	6 (2.38%)	6 (2.38%)	11 (4.37%)	252
	Fuzzy Logic-based method	231 (91.67%)	13 (5.16%)	2 (0.79%)	0 (0.00 %)	6 (2.38%)	252
Synonim standard questions	TF-IDF Method	177 (57.46%)	86 (27.92%)	13 (4.22%)	15 (4.87%)	17 (5.52%)	308
	Fuzzy Logic-based method	252 (81.82%)	41 (13.31%)	3 (0.97%)	5 (1.62%)	47 (2.27%)	308
Imprecise standard questions	TF-IDF Method	74 (59.20%)	32 (25.60%)	6 (4.80%)	1 (0.80%)	12 (9.60%)	125
	Fuzzy Logic-based method	111 (88.80%)	5 (4.00%)	0 (0.00 %)	0 (0.00 %)	9 (7.20%)	125
Specific standard questions	TF-IDF Method	46 (20.08%)	49 (21.40%)	26(11.35%)	55(24.01%)	52 (22.71%)	229
	Fuzzy Logic-based method	107 (46.72%)	53 (23.14%)	24 (10.48%)	23(10.04%)	22 (9.61%)	229

Table 14. Information Retrieval results of using both Term Weighting methods, according to the type of standard question.

get the worst result by far among all classes of standard questions. This is a logical fact, considering that these questions are associated with the main standard question, but it is more concrete. In fact, it is usual for such specific questions to belong to a list within a whole. This way, there may be objects that are more related to the query than the requiered object itself. This is hardly a drawback, since both objects are retrieved to the user - the more specific one and the more general one -. The own user must choose which one is the most accurate. This case shows more clearly that the fact of using Fuzzy Logic allows the user to extract a larger number of objects.

4.3 Tests according to the number of standard questions

Another aspect to consider in the analysis of the results is the number of standard questions assigned to every object. Obviously, an object that is well defined by a single standard question is very specific. Thus, it is easy to extract the object from the complete set of knowledge. However, there are objects that contain very vague or imprecise information, making it necessary to define several standard questions for every object. For this study, the objects are grouped into the following:

- Group 1: the object is defined by a single standard question.
- Group 2: the object is defined by two to five standard questions.
- Group 3: the object is defined by six to ten standard questions.
- Group 4: the object is defined by more than ten standard questions.

Obviously, groups 1 and 2 are more numerous, since it is less common that many questions have the same response. However, the objects from the groups 3 and 4 correspond to a wide range of standard questions, so they are equally important. In Table 15 the number of objects for each of these groups is defined.

Group number	Number of standard questions per object	Number of objects
Group 1	1	95
Group 2	2-5	108
Group 3	6-10	22
Group 4	> 10	28

Table 15. Groups according to the number of standard questions per object.

To analyze the results, the position in which the required object is retrieved must be considered. We consider the retrieval of most of the standard questions that define that object. For example, if an object is defined by 15 standard questions and, for 10 of them, the object is retrieved in second place, it is considered that the object has actually been retrieved in second place.

In short, this study does not focus on the answers to standard questions, but on the correctly retrieved objects. This provides a new element for the system analysis. Results are shown in Table 16.

For group 1, the results are almost perfect for the Fuzzy Logic-based method, as nearly all the objects are retrieved in the first place (about 94%). However, the TF-IDF method, though not as accurate, resists the comparison. This behaviour is repeated in group 2. The objects are often retrieved by both methods among the top three items. Though, the Fuzzy Logic-based method is better for its accuracy, retrieving over 92% of the objects in the first place. In view of the tests, we conclude that the results are very good for both methods when up to five standard questions are defined. Although the results are better for the novel Fuzzy Logic-based Term Weighting method, they are also quite reasonable for the classical TF-IDF Term Weighting method.

However, the largest advantage of using Fuzzy Logic for Term Weighting occurs when many standard questions per object are defined, i.e. when the information is confusing, disordered or imprecise. For the case of group 3, where objects are defined by among six and ten standard questions per object type, we observe that there is a significant difference between the TF-IDF classical method and the proposed Fuzzy Logic-based method. Although both methods retrieve all the objects, there is a big difference in the way they are retrieved, especially on the accuracy of the information extraction. 86% of the objects are retrieved in first place using the Fuzzy Logic-based method, while only 45% using the TF-IDF classical method.

Type of standard question		Cat1	Cat2	Cat3	Cat4	Cat5	Total
Group 1	TF-IDF Method	74 (77.89%)	16 (16.84%)	1 (1.05%)	1 (1.05%)	3 (3.16%)	95
	Fuzzy Logic-based method	89 (93.68%)	3 (3.16%)	2 (2.10%)	0 (0.00 %)	1 (1.05%)	95
Group 2	TF-IDF Method	86 (79.63%)	21 (19.44%)	1 (0.93%)	0 (0.00 %)	0 (0.00 %)	108
	Fuzzy Logic-based method	100 (92.59%)	7 (6.48%)	0 (0.00 %)	0 (0.00 %)	1 (0.93%)	108
Group 3	TF-IDF Method	10 (45.45%)	9 (40.91%)	3 (13.63%)	0 (0.00 %)	0 (0.00 %)	22
	Fuzzy Logic-based method	19 (86.36%)	3 (13.63%)	0 (0.00 %)	0 (0.00 %)	0 (0.00 %)	22
Group 4	TF-IDF Method	10 (35.71%)	10 (35.71%)	3 (10.71%)	2 (7.14%)	3 (10.71%)	28
	Fuzzy Logic-based method	21 (75.00%)	4 (14.29%)	1 (3.57%)	1 (3.57%)	1 (3.57%)	28

Table 16. Information Retrieval results of using both Term Weighting methods, according to the number of standard questions per object.

The difference is even more marked when more than ten standard questions per object are defined. In this case, it is obvious that none of the questions clearly define the object, so that information is clearly vague. While using the Fuzzy Logic-based method, more than 96% of the objects are retrieved - with 75% of them in the first place -, with the TF-IDF method correctly, only 82% of the objects are retrieved. Furthermore, only 35.7% of these objects are extracted in the first place.

In view of the table, we observe that the more standard questions per object, the better the results of the Fuzzy Logic-based method, compared with those obtained with the classical TF-IDF method. Therefore, the obvious conclusion is that the more convoluted, messy and confusing is the information, the better the Fuzzy Logic-based Term Weighting method is compared to the classical one. This makes Fuzzy Logic-based Term Weighting an ideal tool for the case of information extraction in a web portal.

5. Future research directions

We suggest the application of other Computational Intelligence techniques apart from Fuzzy Logic for Term Weighting. Among these techniques, we believe that the so-called neuro-fuzzy techniques represent a very interesting field, as they combine human reasoning provided by Fuzzy Logic and the connection-based structure of Artificial Neural Networks, taking advantage of both techniques. One possible application is the creation of fuzzy rules by means of an Artificial Neural Network system.

Another possible future direction is to check the validity of this method in other environments containing inaccurate, vague and heterogeneous data.

6. Conclusion

The difficulty to distinguish the necessary information from the huge quantity of unnecessary data has enhanced the use of Information Retrieval recently. Especially, the so-called Vector Space Model is much extended. Vector Space Model is based on the use of index terms. These index terms are associated with certain weights, which represent the importance of these terms in the considered set of knowledge. In this chapter, we propose the development of a novel automatic Fuzzy Logic-based Term Weighting method for Vector Space Model. This method improves the TF-IDF Term Weighting classic method for its flexibility. The use of Fuzzy Logic is very appropiate in heterogeneous, vague, imprecise, or not in order information environments.

Fuzzy Logic-based method is similar to TF-IDF, but also considers two aspects that the TF-IDF does not: the degree of identification of the object if a determined index term is solely used in a query; and the existance of join index terms. Term Weighting is automatic. The level of expertise required is low, so there is no need for an operator of any kind of knowledge about Fuzzy Logic. Therefore, an operator only has to know how many times an index term appears in a certain subset and the answer to two simple questions.

Although the results obtained with the TF-IDF method are quite reasonable, Fuzzy Logic-based method is clearly superior. Especially when user queries are not equal to the standard query or they are imprecise, we observe that the performance declines more for the TF-IDF method than for the Fuzzy Logic-based method. This fact gives us an idea of how suitable is the use of Fuzzy Logic to add more flexibility to an Information Retrieval system.

7. References

Lertnattee, V. & Theeramunkong, T. (2003). Combining homogenous classifiers for centroid-based text classification. *Proceedings of the 7th International Symposium on Computers and Communications*, pp. 1034-1039.

Lee, D.L., Chuang, H., Seamons, K., 1997. *Document ranking and the vector-space model*. IEEE Software, Vol. 14, Issue 2, pp. 67 – 75.

Liu, S., Dong, M., Zhang, H., Li, R. & Shi, Z. (2001). An approach of multi-hierarchy text classification. *Proceedings of the International Conferences on Info-tech and Info-net*. Beijing. Vol 3, pp. 95 – 100.

Raghavan, V.V. & Wong, S.K. (1986). A critical analysis of vector space model for information retrieval. *Journal of the American Society for Information Science*, Vol.37 (5), p. 279-87.

Ruiz, M. & Srinivasan, P. (1998). Automatic Text Categorization Using Neural Networks. *Advances in Classification Research vol. 8: Proceedings of the 8th ASIS SIG/CR*

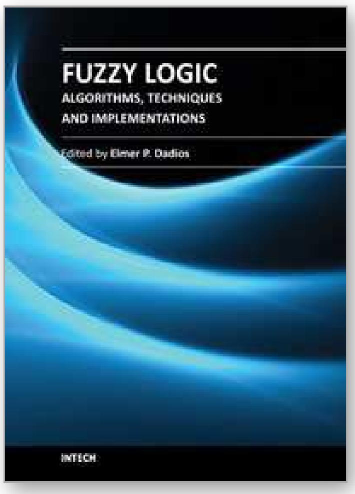
Classification Research Workshop. Ed. Efthimis Efthimiadis. Information Today, Medford:New Jersey, pp 59-72.

Salton, G. (1988). *Automatic Text Processing*. Addison-Wesley Publishing Company.

Salton, G. & Buckley, C. (1996). Term Weighting Approaches in Automatic Text Retrieval. *Technical Report TR87-881, Department of Computer Science, Cornell University, 1987. Information Processing and Management* Vol.32 (4), pp. 431-443.

Van Rijsbergen, C.J. (1979). *Information retrieval.* Butterworths.

Zhao, Y. & Karypis, G. (2002). Improving precategorized collection retrieval by using supervised term weighting schemes. *Proceedings of the International Conference on Information Technology: Coding and Computing*, pp 16 – 21.


FUZZY LOGIC
ALGORITHMS, TECHNIQUES AND IMPLEMENTATIONS
Edited by Elmer P. Dadios
INTECH