A NOVEL TERM WEIGHTING SCHEME FOR A FUZZY LOGIC BASED INTELLIGENT WEB AGENT

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

Keywords: Term Weighting, Information Extraction, Information Retrieval, Vector Space Model, Intelligent Agent.

Abstract: Term Weighting (TW) is one of the most important tasks for Information Retrieval (IR). To solve the TW problem, many authors have considered Vector Space Model, and specifically, they have used the TF-IDF method. As this method does not take into account some of the features of terms, we propose a novel alternative fuzzy logic based method for TW in IR. TW is an essential task for the Web Intelligent Agent we are developing. Intelligent Agent mode of operation is also explained in the paper. An application of the Intelligent Agent will be in operation soon for the University of Seville web portal.

1 INTRODUCTION

The spectacular advance of Information Technology and especially of the internet has caused an enormous increase of the available information for the users. Nowadays, it is not only a question of finding the information but of selecting the essential one.

For a long time researchers have faced the problem of managing this high amount of information. IR research deals mainly with documents. Achieving both high recall and precision in IR is one of its most important aims. IR has been widely used for text classification (Aronson et al., 1994; Liu et al., 2001) introducing approaches such as Vector Space Model (VSM), K nearest neighbour method (KNN), Bayesian classification model and Support Vector Machine (SVM) (Lu et al., 2002). VSM is the most frequently used model. In VSM, an object – i.e., a document - is conceptually represented by a vector of keywords extracted from the object, with associated weights representing their importance in the document. Typically, the so-called TF-IDF method is 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. A usual formula to describe the weight of a term j in document i is $w_{ij} = tf_{ij} \times idf_j$. This formula has been modified and improved by many authors to achieve better results in IR (Lee et al., 1997; Liu et al., 2001; Zhao & Karypis, 2002; Xu et al., 2003), but it was never taken into account that some other aspects of keywords may be important for determining term weights apart from TF and IDF.

In this paper, we introduce a novel fuzzy logic based Term Weighting (TW) scheme. This scheme bears in mind some other features that we consider important for calculating the weight of a term, taking advantage of fuzzy logic flexibility.

2 INTELLIGENT WEB AGENT

In previous investigations, we have proposed the use of a fuzzy logic based intelligent agent for Information Extraction (IE) (Ropero et al., 2007). This proposal came from the need of approaching to the contents of an extensive set of accumulated knowledge and it is based on the fact that a set of objects may be qualified on closely related families. The generality of the proposed method is a consequence of the substitution of the objects for their representations in Natural Language (NL). Therefore, the same techniques may be applied to sets of any nature and, particularly, to sets of accumulated knowledge of any kind.

As described below, one of the most controverted tasks that are held to build the

Gómez A., Ropero J., León C. and Carrasco A. (2008).
A NOVEL TERM WEIGHTING SCHEME FOR A FUZZY LOGIC BASED INTELLIGENT WEB AGENT.
In *Proceedings of the Tenth International Conference on Enterprise Information Systems - AIDSS*, pages 496-499
DOI: 10.5220/0001700604960499

intelligent agent is TW. This paper describes how to improve the intuitive TW method that was proposed in our previous investigations by means of an automated process.

2.1 NL Object Representation

As mentioned above, it is necessary to associate a NL representation to every object in a set of accumulated knowledge. The process consists of two steps.

The first step is to build a question in NL, i.e. the object. Its answer is the desired to be represented. This is what we call a standard-question. Knowledge Engineer's expertise with regard to object field jargon is important provided that the higher knowledge the higher reliability of the standard-questions proposed for the representation of the object. This is because they are more similar to possible user consultations. Nevertheless, it is possible to re-define object representations or to add new definitions analyzing user consultations and studying their syntax and their vocabulary. Consequently, the system can refine its knowledge by means of the Knowledge Engineer.

The second step is keyword selection. These keywords are selected from the words in a standard-question whose meaning is strongly related to the represented object.

2.2 Hierarchic Structure

The standard-question collection generated from the whole set of knowledge must be organized in hierarchical groups. This offers the advantage of easier handling. The proposed method classifies objects in a completely vertical way forcing every object to be accessible in only one way across the classification tree, that is to say, classifying it under a unique criterion or group of criteria. Group assignment decision is adopted by a Knowledge Engineer on the basis of common subject matter. This way objects are classified under an only root tree structure where several levels are established.

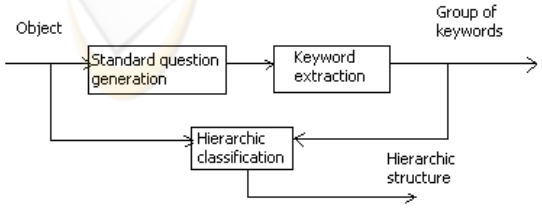

Figure 1: Mode of operation of the system.

The mode of operation of the system is shown in Figure 1. Table 1 shows an example of how keyword extraction takes place.

Table 1: Keyword Extraction Example.

Term	Content	Example
Object	Information to Retrieve	We introduce a novel fuzzy logic based term weighting scheme
Standard-question	NL sentence	Which is the topic of the conference paper?
Keywords	Selected words from Standard-questions	Topic, conference, paper.

3 TERM WEIGHTING

3.1 Term Weighting Parameters

The classification of the objects in the set of knowledge has to be performed according to the structure and the criteria proposed above. A set of keywords representing one or several objects is available. In addition, the degree of belonging of every keyword to the sets where it appears does not have to be the same. Thus, there are several sets for every level. Keywords can belong to more than one set or subset in the same level. It is then necessary to define a term weight to specify a keyword degree of belonging for every set in every level. These weights indicate the degree of belonging of every keyword to the corresponding subset.

The relation between keywords and objects – term weight – is determined on the basis of 4 parameters that are described below:

The first parameter is the number of objects in a considered set containing a keyword. This parameter is related to IR concept of Term Frequency (TF). The more objects in a set a keyword belongs to, the higher value for the corresponding term weight.

The second parameter is the number of objects in different sets to the considered containing a keyword. This parameter is related to IR concept of Inverse Document Frequency (IDF). The more objects in different sets a keyword belongs to, the lower value for the corresponding term weight.

The third parameter is the degree of identification of the object if only the considered keyword is used. This parameter has a strong influence on the final value of a term weight. The more a keyword identifies an object, the higher

value for the corresponding term weight. Nevertheless, this parameter creates two disadvantages in terms of practical aspects when it comes to carrying out a term weight automated and systematic assignment. On the one hand, the degree of identification is not deductible from any characteristic of a keyword, so it must be specified by the Knowledge Engineer. The assigned values could be neither univocal nor systematic. On the second hand, the same keyword may have a different relationship with every object. The solution to both problems is to create a table with all the keywords and their corresponding weights for every object. This table will be created in the phase of keyword extraction from standard-questions. Imprecision practically does not affect the working method due to the fact that both term weighting and information extraction are based on fuzzy logic, what minimizes possible variations of the assigned weights. In the example set in section 2.2, this parameter would rise to a higher value for the word 'topic', but it would not be so high for the word 'conference'.

The fourth parameter is related to what we have called join terms. In the example set in section 2.2, 'conference paper' would constitute two join terms. Join terms have lower weights provided that the appearance of these join terms is what really determines the object with major certainty whereas the appearance of only one the words may refer to another object.

The consideration of these parameters determines the weight of a keyword for every subset in every level. We have to consider that some systems size may change, so that it will be necessary to modify the weights. For this reason, it is a good idea to consider term weighting as an automated task. As the rules for determining term weights are not mathematically precise, this process is a good candidate for a solution by means of fuzzy logic.

3.2 Term Weighting

In order to automate term weighting on the basis of the above described parameters, a set of rules have been defined. These rules are expressed on the If…Then way. Once the values of the above mentioned parameters have been determined, they are taken as the input to a fuzzy logic engine which considers the relative weight of each one and returns a value which corresponds to proposed term weight.

Logically, the more inputs the engine has, the more rules there are. Inputs can take three values: *Low*, *Medium* and *High*. Outputs can take the values of *Low*, *Medium-Low*, *Medium-High* and *High*. Some examples of the defined inference rules are:

- If all inputs are *Low*, Then output is *Low*.
- If input2 is *Low*, input1 is *Medium* or *High* and the others are High, Then output is *Medium-Low*.
- If input2 is *High* and one of parameters 3 and 4 is *High*, Then output is *Medium-High*.
- If inputs 2, 3 and 4 are *High*, Then output is *High*.

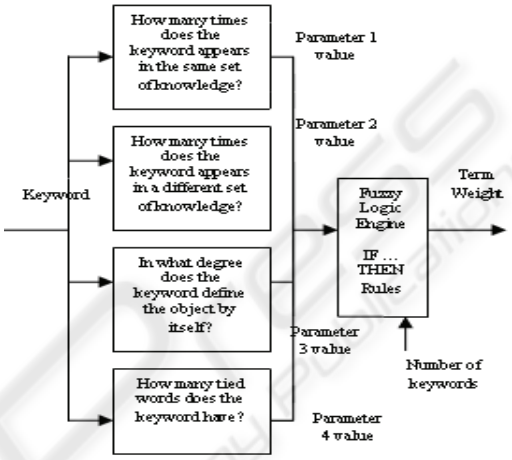

Figure 2: Term weighting process.

The possible combinations generate 81 rules. The fuzzy engine uses triangular fuzzy sets, a singleton fuzzifier and centroid defuzzifier. The whole process is shown in Figure 2.

Besides, due to the fact that IE system uses two fuzzy engines which do not work in the same way as it depends on the number of keywords, (Ropero, 2007) the proposed weight suffers a later modification depending on the number of keywords that represent the object.

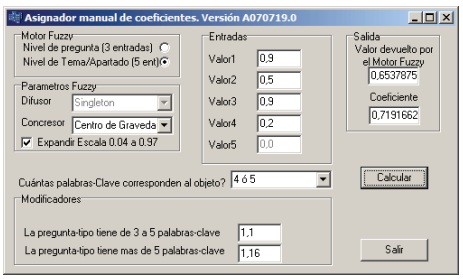

Figure 3: Term Weight Generator interface.

For the comparative study of the automatically obtained weight values and the ones that were proposed intuitively by the Knowledge Engineer, a specific application has been developed. Weights are generated according to the described parameters and

rules managing to fit fuzzy logic rules and verifying that equivalent results are obtained. In Figure 3, the interface for the term weight generator is shown.

4 RESULTS

As we did in (Ropero et al., 2007) tests are based on the use of standard-questions as user consultations. The first goal of these tests is to check that the system makes a correct identification of standard-questions with an index of certainty higher than 0.7. The use of fuzzy logic makes it possible to identify not only the corresponding standard-question but others as well. This is related to *recall*, though it does not match that exact definition (Ruiz & Srinivasan, 1998). The second goal is to check if the required standard-question is among the three answers with higher degree of certainty. These three answers should be presented to the user, with the correct one among these three options. This is related to *precision*. The obtained results are shown in Table 2.

Table 2: Obtained Results.

Type of system	First answer	Among the first three answers	Out of the first three answers	Failed answer
Intuitive Term Weighting	77 %	20.5 %	1 %	1.5 %
Automatic Term Weighting	79 %	17 %	2 %	2 %

Comparative tests between the results obtained with the fuzzy logic engine and the ones proposed by the Knowledge Engineer System are very satisfactory as it is observed that rules fit correctly to produce a few functionally equal coefficients to the wished ones.

Besides, there are two important advantages for the new method: on the one hand, term weighting is automatic; on the second hand, the level of required expertise is much lower, as there is no need for an operator to know very much about the way fuzzy logic engine works, but only to know how many times a keyword appears in every set and the answer to some simple questions – Does a keyword undoubtedly define an object by itself? Is a keyword tied to another one? - .

5 CONCLUSIONS

As said above, comparative tests between the results obtained with the fuzzy logic engine and the ones obtained by Knowledge Engineer System's expertise are very similar. Taking into account that there are two fundamental advantages with our new method - automation and less level of expertise required – we must conclude that the method is suitable for Term Weighting in Information Retrieval. This method will be used for the design of a Web Intelligent Agent which will soon start to work for the University of Seville web page.

ACKNOWLEDGEMENTS

The work described in this paper has been supported by the Spanish Ministry of Education and Science (MEC: Ministerio de Educación y Ciencia) through project reference number DPI2006-15467-C02-02.

REFERENCES

Aronson, A.R, Rindflesch, T.C, Browne, A. C., 1994. *Exploiting a large thesaurus for information retrieval.* Proceedings of RIAO, pp. 197-216.

Lee, D.L., Chuang, H., Seamons, K., 1997. *Document ranking and the vector-space model*. IEEE Software, Vol. 14, Issue 2, p. 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, 2001. Beijing. Vol 3, pp. 95 – 100.

Lu, M., Hu, K., Wu, Y., Lu, Y., Zhou, L., 2002. *SECTCS: towards improving VSM and Naive Bayesian classifier.* IEEE International Conference on Systems, Man and Cybernetics, Vol. 5, p. 5.

Ropero J., Gomez, A., Leon, C., Carrasco, A. 2007. *Information Extraction in a Set of Knowledge Using a Fuzzy Logic Based Intelligent Agent*. Lecture Notes in Computer Science. Vol. 4707, pp. 811-820.

Ruiz, M.E., 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. 1998. pp 59-72.

Xu, J., Wang, Z. 2003. TCBLSA: A new method of text clustering. *International Conference on Machine Learning and Cybernetics. Vol. 1, pp. 63-66.*

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, 2002. pp 16 – 21.