A Method for the Access to the Contents in a Set of Knowledge Using a Fuzzy Logic Based Intelligent Agent

Jorge Ropero[1], *Ariel Gómez*[1], *Carlos León*[1] *and Alejandro Carrasco*[1]
[1] Department of Electronic Technology, University of Seville, Spain.
jropero@dte.us.es, ariel@us.es, cleon@us.es, acarrasco@us.es.
Escuela Técnica Superior de Ingeniería Informática
Avda. Reina Mercedes, s/n
41012 - Sevilla (Spain)

Abstract

This paper proposes a method for the classification of the contents in a set of knowledge in order to answer to user consultations using natural language. The system is based on a fuzzy logic engine, which takes advantage of its flexibility for managing sets of accumulated knowledge. These sets can be built in hierarchic levels by a tree structure. A method of consultation based on a fuzzy logic application provided with an interface that one may interact with in natural language is also proposed. The eventual aim of this system is the implementation of an intelligent agent to manage the information contained in an internet portal.

1. Motivations

The access to the contents of an extensive set of accumulated knowledge – a database, a summary of documents, web contents, pictures, etc – is an important concern nowadays. Information Retrieval (IR) deals with large collections of textual material, and its aim is to satisfy user queries and needs [1]. These needs are increased when the user in question is not familiar with the matter or there are ambiguous contents, bad organization or, simply, complex topics or a great amount of information difficult to manage.

Eventually, unsuccessful attempts can turn out to be frustrating if the exact term or terms are not used to make the consultations - a machine only will answer adequately if it is asked in an exact way -, and one can eventually end in a paradox: the less one knows the more difficult it is to find the answers. In many cases the solution is to seek help from an expert on the topic. In fact the person asked to help is an interpreter who is able to generate a syntactically and semantically a correct search obtaining the desired answers. Consequently, there is the need for an agent to interpret the vague information we provide, giving us concrete answers related to the existing contents of the set of knowledge. This should be based on an estimation of the certainty of the relation between what we have expressed in natural language and the contents stored in the set of knowledge.

To solve this, we have developed a method of classification of contents by creating a few indexes based on key words, and a method of consultation based on a fuzzy logic application with an interface that one may interact with in natural language. We then propose an artificial intelligence (AI) application based on the use of fuzzy logic.

2. Mode of operation

All printed material, including text, illustrations, and charts, must be kept within a print area of 6.9 inches (175 mm) wide by 8.9 inches (226 mm) high. Do not write or print anything outside the print area. The top margin must be 1 inch (25 mm), except for the title page, and the left margin must be 0.75 inch (19 mm). All text must be in a two-column format. Columns are to be 3.27 inches (83 mm) wide, with a 0.37 inch (8 mm) space between them. Text must be fully justified.

2.1. Objectives

The main objective of the system designed must be to let the users find possible answers to what they are looking for in a huge set of knowledge. With this aim, the whole set of knowledge must be classified into different objects, as shown in Figure 1. These objects are the possible user consultations, organized in hierarchic groups. A standard question is assigned for every object, and different key words from each standard question must then be selected in order to differentiate one object from the others. Finally,

Fourth International Conference on Fuzzy Systems and Knowledge Discovery (FSKD 2007)
0-7695-2874-0/07 $25.00 © 2007 IEEE

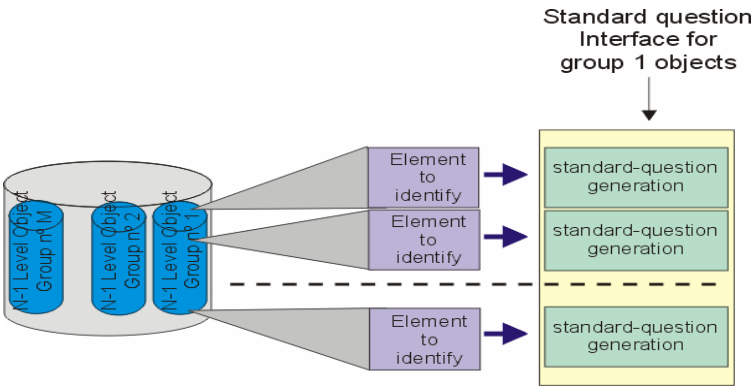

Figure 1. Generated standard question interface.

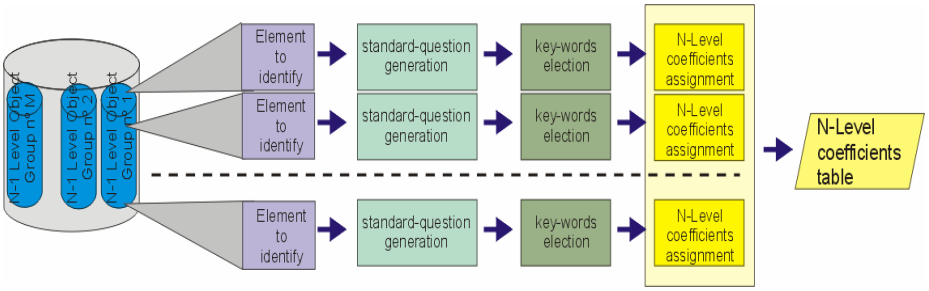

Figure 2. Term weighting scheme.

term weights are assigned to every word for every level of hierarchy in a scheme based on a vector space model. These term weights are the inputs to a fuzzy logic system, which must detect the object to which the correspondent user consultation refers.

2.2. Hierarchic structure

The standard-questions collection generated from the whole set of knowledge must be organized in hierarchic groups. This offers the advantage of easier handling. For tests, we decided to use the questions-answers database of the University of Seville to create a list of the most frequently asked questions. The whole knowledge was organized into three levels: topic, section and question. For any web page, it would be enough to create a bank of possible questions–answers, arrange them hierarchically, and identify them as objects.

Every possible user question must be related with standard questions in order to present its standard answers as possible answers to this user question. This may be seen in Figure 3. There are obviously many ways of asking, so the aim of the system is to identify real user consultations and what we have called standard questions. As there may be several questions similar to the one asked by the user or as it may be interesting for the user to get related answers, several answers will be presented.

When a question is made, the first processing step consists of distinguishing the key words in the consultation. These words must be searched in a database which must contain all the words that are related somehow to the content of the subject we are dealing with –see Figure 4. Another database with the possible answers becomes necessary.

Key words are assigned to every standard question in order to identify it. These words are chosen among those that could appear in a possible consultation. As mentioned above, the whole knowledge is grouped in various hierarchic levels, so the belonging of these words to different levels is determined by a few numerical coefficients indicating how significant the considered word is within the level in question. With this aim, weight vectors are assigned to each word. Each vector contains the certainty of belonging to every fuzzy set. It is important to notice that the same word can belong to several different sets.

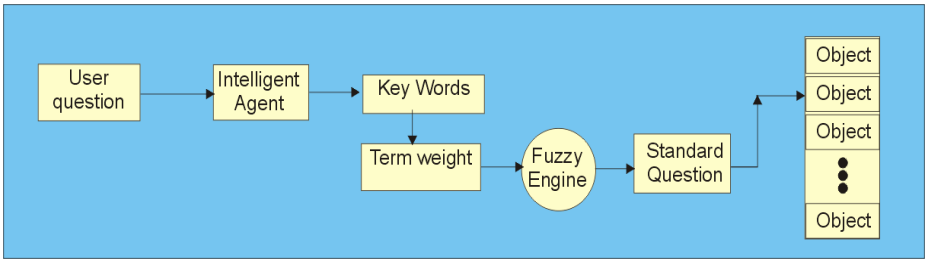

Figure 3. Mode of operation.

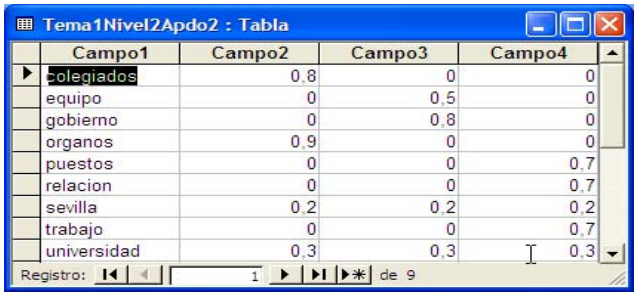

Figure 4. Index words database.

2.3. Term weighting

As the system must work using a natural language interface, we propose the use of a system based on vector space modeling for term weighting. Automatic document indexing is usually based on the frequency of occurrence, that is to say, words with a high frequency of occurrence are less significant than words with a low one, and the inverse document frequency, that is, how often it occurs in the whole document. The main limitation of this scheme is created by terms which are unhelpful for identification but have the same frequency of occurrence as the relevant ones [2-8].

Thus, we propose a novel scheme based on a vector space model which also takes into account if the word is important for the meaning of the question itself or if the word is linked to other words and its relationship with the whole set of knowledge for term weighting [7,9].

The success of the proposed method depends to a great extent on a correct assignment of the coefficients to the key words, that is, the building of the weight vectors. The process consists of 3 stages: election of key words, coefficient assignment of the chosen words; and modification of the index values in order to obtain the desired minimal certainties.

2.3.1. Election of key words. As mentioned earlier, for every element that has to be determined, a question is assigned in natural language. These questions are called *standard questions*. From every standard question key words are chosen. These key words will allow the user consultation and these standard questions to be matched and, therefore, to indicate the required element. Word election is based on its *concretion,* meaning the degree of relation of the word with the element to be identified. This excludes articles, conjunctions, verb forms, etc, unless they are strongly significant for the question structure.

2.3.2. Coefficient assignment. For every key word, coefficients corresponding to every level of hierarchy of the whole set of knowledge must be assigned. The higher the relationship between the present level and the key word, the higher the coefficient of that key word will be for that level. These coefficients are the inputs to every fuzzy logic system, as described in Section 2.4.

2.3.3. Standard question recognition test. Once the coefficient assignment is made, standard question recognition tests must take place. These tests are based on using standard questions as user consultations.

2.3.4. Coefficient modification. The results obtained for standard question recognition tests represent the relationship of every standard-question relationship with itself. The higher it is the better the results are. However, it must be taken into account that similar questions must have a high degree of certainty too. This allows for the modification of coefficients in order to achieve better results. An example of how we have modified some coefficients is shown in Section 3.1.

2.4. Fuzzy logic system

Fuzzy logic arises as a response to the inflexibility of the classic binary logic [10-11]. By means of a set of functions, a degree of flexibility may be given to these epithets: what may be cool for a Sevillian might be mild for a Berliner.

A fuzzy logic system gives flexibility for term weighting. More important than having a concrete value for weights, what really matters is that a feature is represented by a word. It is not so important that a weight is 0.8 or 0.9: the weight is HIGH in both cases.

All the key words are extracted for comparison with the ones contained in our key word database. As mentioned earlier, the whole set of knowledge is arranged in levels of hierarchy. The inputs to the fuzzy logic system are the coefficients of belonging to every level. The set "belonging to every level 1" is analyzed bearing in mind the value returned by the fuzzy engine. If the level of certainty is lower than a predefined value, the content of the corresponding set is rejected. Starting from level one and using a tree structure makes it possible to reject a great amount of content which will not be considered in future searches.

For every set that has overcome a minimum certainty threshold, the process is repeated and the coefficients of belonging corresponding to every level 2 set are evaluated. Sets where the degree of certainty returned by the fuzzy engine does not surpass a certain minimal threshold are rejected. If they surpass the threshold, the method for determining the belonging to the following level is applied to them. This process is repeated until the last level. The answers correspond to those last level elements in which certainty has overcome the definite threshold. There can be more than one answer. The vaguer the questions, the more answers we will obtain. This process may be seen in Figure 5.

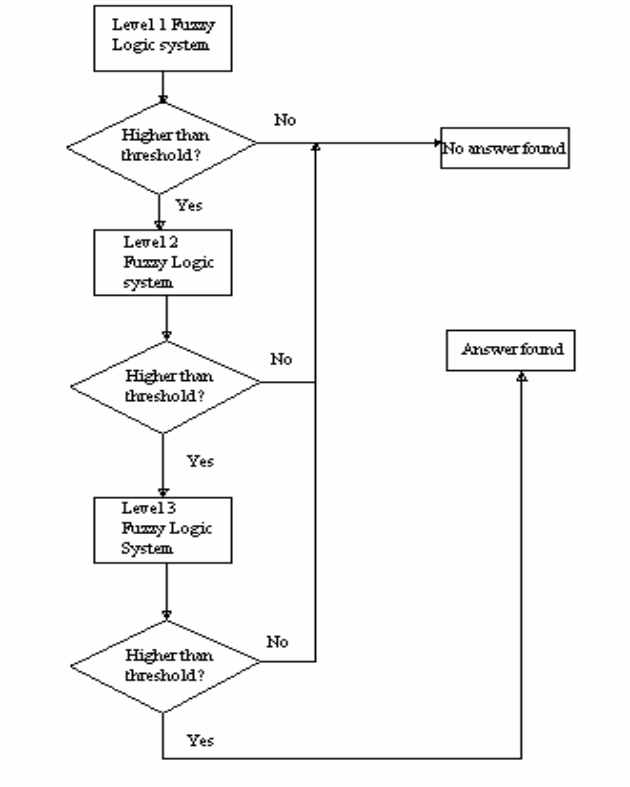

Figure 5. Fuzzy logic system: operation mode.

The heart of the fuzzy logic system is the fuzzy engine. This engine is responsible for determining the probability that the key words contained in a consultation will belong to a certain fuzzy set in a specific level. The engine must evaluate the belonging to every set for the corresponding level. Thus, the engine takes the coefficients of the key words for that set as inputs. The fuzzy engine output will be determined by the defined rules. These rules are of the IF … THEN type. An example of a rule might be this:

IF word_index1 is HIGH AND word_index2 is MEDIUM AND word_index3 is LOW, THEN output is HIGH.

2.5. System administrator

The system requires a system administrator who has basically three functions:

a) Defining and modifying term weights and rules.
b) Adding new words to the database when necessary.
c) Creating a system feedback which asks the eventual users for their opinion about the answers given by the assistant in order to take the necessary steps in each case.

Thus far, an administrator interface has been created – Figure 6 - .

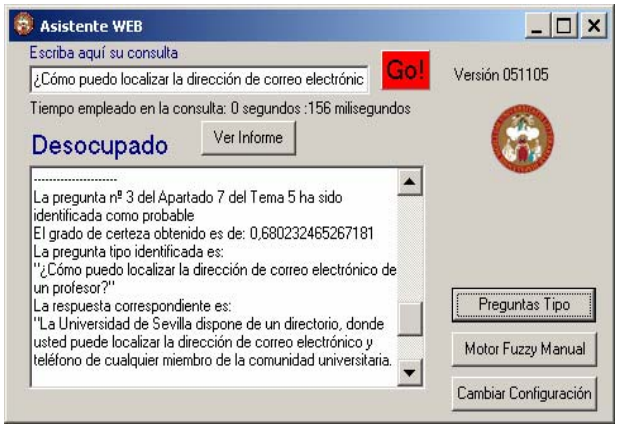

Figure 6. Administrator interface.

3. Tests and results

3.1. Test definition

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 [12]. 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. The correct answer must be among these three options. This is related to *precision*, though it does not match that exact definition either.

To achieve these goals, it will be necessary to modify the assigned coefficients in principle. These modifications will be carried out again using a bottom-up design, so that the certainty returned by the fuzzy engine is at least of 0.7 for the correct decision in the lowest level, and of 0.5 for other levels. Likewise, in order to limit the number of possible answers, incorrect question indexes may be modified making them lower.

This method has been proven considering the most frequent questions asked in the portal web administrator as our set of knowledge, which consists on a set of 117 questions. The elements to be found are the questions themselves and possibly related questions.

Test results for standard question recognition fit into five categories:

1.-The correct question is the only one found or the one that has the highest degree of certainty.
2.-The correct question is between the two with the highest certainty or is the one that has the second highest degree of certainty.
3.-The correct question is among the three with the highest degree of certainty or is the one that has the third highest certainty.
4.-The correct question is found but not among the three with the highest degree of certainty
5.-The correct question is not found.

3.2. Rule definition

As mentioned above, rule definition corresponds to the administrator. Logically, the more inputs the engine has, the more rules there are. For example, for a three input engine, the inputs can take three values: LOW, MEDIUM and HIGH. The outputs can take the values of LOW, MEDIUM-LOW, MEDIUM-HIGH and HIGH. The inference rules defined are:

If all inputs are LOW, output is LOW.
If one input is MEDIUM and the others are LOW, output is MEDIUM-LOW.
If two inputs are MEDIUM and the others are LOW, output is MEDIUM-HIGH.
If all the input are MEDIUM or one input is HIGH, output is HIGH.

The possible combinations generate 27 rules for the fuzzy engine. A five input engine generates 243 rules.

3.3. Input definition

For every question, 3 to 5 key words are defined. Nevertheless, the user may include in his consultation anywhere from 1 to 5 of these words. Defining an engine with only a few inputs causes rapid saturation of the system. This is a great handicap for precision: 90 % of correct answers are detected but only half of them are the first option as may be seen in Table 1. Defining a five input engine produces values with a very low degree of certainty. Precision grows to 55 % but recall falls.

A solution to this problem is the use of variable thresholds. When all eventual outputs are below the fixed threshold, this threshold goes down until an output is found. If thresholds are variable, the system is more flexible and results are better.

Finally, the solution provided is to implement a flexible system with variable inputs. If the user consultation has at the most three key words, a three input fuzzy engine is used, whereas if the user consultation includes four or

more key words, the system will use a five input fuzzy engine. Results are much better in terms of recall and precision. The user obtains the correct answer 97.75 % of the times and the first option 77.45 % of the times.

Type of system / Test result category	First answer	Among the first two answers	Among the first three answers	Out of the first three answers	Failed answer
3 input engine	45 %	24 %	9 %	12 %	10 %
5 input system with fixed thresholds	55 %	12 %	3 %	1 %	29 %
5 input system with variable thresholds	70 %	14 %	3 %	1 %	12 %
Variable input system with fixed thresholds	77 %	16 %	4.5 %	1 %	1.5 %

Table 1. Test results.

4. Conclusions

A method of consultation based on a fuzzy logic application provided with an interface that one may interact with in natural language has been presented. The system takes advantage of converting any kind of object in a text object; this allows the application of text retrieval techniques. The other advantage of the method arises from fuzzy logic flexibility, which makes it possible to have a non-rigid term weighting in the stage of classification of the contents in the set of knowledge.

A method for the classification of the contents in a set of knowledge is also proposed. We also present a modification of the classical vector space model to define term weights, which are used as inputs to a fuzzy logic based system.

An eventual application of this system is the implementation of an intelligent agent to manage the information contained in an internet portal. So far the results obtained are good enough, as the number of correctly detected consultations is high.

5. Acknowledgments

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.

6. References

[1] K.L. Kwok. "A neural network for probabilistic information retrieval". Proceedings of the 12th annual international ACM SIGIR conference on Research and development in information retrieval. Cambridge, Massachusetts, United States. 1989

[2] G. Salton. Introduction to Modern Information Retrieval. McGraw-Hill, 1983.

[3] C.J. van Rijsbergen, . Information retrieval. Butterworths, 1979.

[4] G. Salton, C. Buckley. "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), p. 431-443, 1996.

[5] G. Salton. Automatic Text Processing. Addison-Wesley Publishing Company, 1988.

[6] D.L Lee, H. Chuang; K. Seamons. "Document ranking and the vector-space model". Software, IEEE. Vol. 14, Issue 2, Mar/Apr 1997 p. 67 – 75

[7] S. Liu, M. Dong; H. Zhang, R. Li, Z. Shi. "An approach of multi-hierarchy text classification". International Conferences on Info-tech and Info-net, 2001. Proceedings. ICII 2001 - Beijing. Vol 3, Oct/ Nov. 2001 p. 95 – 100.

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

[9] Y. Zhao, G. Karypis. "Improving precategorized collection retrieval by using supervised term weighting schemes". Proceedings of the International Conference on Information Technology: Coding and Computing, 2002. 8-10 April 2002 Page(s):16 – 21.

[10] B. Martín del Brío, A. Sanz Molina. Redes neuronales y sistemas borrosos. Ra-Ma, 2001.

[11] T. Bouaziz, A. Wolski, "Applying Fuzzy Events to Approximate Reasoning in Active Databases". Proc. Sixth IEEE International Conference on Fuzzy Systems (FUZZ-IEEE'97). July 1-5, Barcelona, Spain.

[12] M.E. Ruiz, P. Srinivasan. "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.