DESIDOC Bulletin of Information Technology, Vol. 24, No. 6, November 2004, pp. 3-9

A Fuzzy Logic Based Search Technique for Digital Libraries

SGK Murthy, Dr RN Biswas

Abstract

Conventional search techniques for searching a database follow Boolean logic. When a database search is made with a given set of keywords, a minor variation in the keyword due to difference in spelling standard, typographical error or difference in inflection of the same stem, results in failure in getting records from the database. This situation demands a special kind of search technique, which should take into account approximation along with exactness. Fuzzy sets are more appropriate to represent imprecise information and matching based on it is better suited to process it. As fuzzy logic is an extension of conventional logic; it handles the concept of partial truth along with true and false. Fuzzy logic more closely follows the way humans think and helps to handle real world complexities more efficiently. This paper presents a new information retrieval support technique used for database search, using fuzzy based matching and is expected to significantly enhance searching benefits over conventional approaches.

1. INTRODUCTION

In today's world most of the information is being stored in digital form. It is created, maintained and retrieved by processes in which, despite large scale automation and mechanization, human interactions are not only unavoidable but desirable and most often essential. A good part of the data is in the form of strings of characters and quite often it becomes necessary to retrieve the information from database on the basis of string data. String data, by its very nature, inherits the attributes and weaknesses of the written language in many occasions. Humanly variations due to style, conventions, standards or even mistakes like omitting or misstating prefixes, suffixes or plurals, or other type of spelling differences and errors at the stages of database creation and data retrieval are quite common in a real world. Conventional search techniques, based on Boolean logic, fail even for minor variations.

More general, typing problems are associated with character misplacement. Missing, extra, transposed or erroneous characters pose a difficulty in identification. The following example describes character misplacement.

L I N E A R T R N A S F O R M

L I N A E R T R A N S F O R M

L E N E A R T R A N S F R M

Standard spelling differences for words like 'colour', 'program' and a host of other words are now well known. So are the different conventions of writing the order of surname, middle name and first name of a person. Similarly, one has the choice from the set of phrases 'carbon resistor', 'resistor made of carbon', 'resistor from/of carbon' or 'resistor carbon' to mean the same item or topic. Apart from simple miss typing, convention, style and standard problems, usages of unfamiliar abbreviations in

technical terms and new additions in vocabulary from ever growing world of technology and their derivatives open chances of differences. Variations due to difference in inflections of the same stem by adding or dropping a prefix or a suffix for moderation in meaning, emphasis or simply style are frequent. The difference in meaning between 'encoding' and 'codification', if any, is imperceptible and quite often both words are used interchangeably. Conventional search techniques are not adequate to identify such small variations. There is a need to have an effective search support technique, which takes care of nearest matches along with exact matches. Considering the above-mentioned practical difficulties, an efficient technique is proposed for searching digital libraries. The search is based on name or title part of the record in the database. Name or title in the database is a string of characters and, without loss of generality, may be assumed to be of a fixed length and is called 'the stored string'. It consists of a few distinct characteristic words otherwise known as keywords. Similarly, the search string, which is normally entered through the keyboard, called 'the target string', is also expected to contain a few keywords. It is assumed that the variations or differences of the type mentioned earlier in the stored string as well as the target string are also transmitted in parts or full to the corresponding set of keywords. It is because of this variations, one has to deal with some amount of uncertainty and lack of exactness in finding the match of the string.

In the search process discussed here, the keywords from stored strings are first extracted and are stored in a separate table with links to corresponding stored strings. In the fuzzy logic based match technique outlined in the subsequent parts of this paper, three metrics - the character count mismatch, the soundEx mismatch and Triad mismatch have been used at word level. Further, it fuses the obtained metrics information with the help of a fuzzy logic based inference system to find closest matches. The three stages, namely, the creation of linked table of keywords, computation of metrics and fuzzy logic based fusion of metrics are finally integrated to produce a fuzzy logic based information retrieval system for a digital library (database).

As the central idea is based on fuzzy sets, fuzzy logic and different types of metrics, a brief introduction is given on these topics. In implementation section, detail explanation on metric computation and fuzzy reasoning is presented.

2. ABOUT FUZZY SYSTEMS

2.1 Fuzzy Logic

Fuzzy logic, an extension of boolean logic, supports values true and false as well as other values in between. The concept of fuzzy logic was introduced by Professor Lotfi A Zadeah. The goal is to develop a model that could be close to natural language process.

Fuzzy logic is an appropriate tool for modeling the kind of uncertainty associated with vagueness with imprecision. The underlying power of fuzzy logic is fuzzy set theory, and linguistic variables that are used for qualitative judgement rather than quantitative judgement.

2.2 Mathematical Uncertainty

Many mathematical disciplines like probability theory, information theory and fuzzy set theory deal with the description of uncertainties. In general, probability theory deals with stochastic uncertainty of occurrence of a particular event. A different type of uncertainty, called lexical uncertainty is inherent in humans language and it deals with the imprecision that is implied in most words humans use to evaluate concepts and derive conclusions. It was proved that humans use words as a "subject category" to classify entities such as "distance", "quality", "performance" and "temperature". Using these subjective categories, humans can evaluate complex problems in the real world.

2.3 Fuzzification

Fuzzification is the first step in a fuzzy system. In this process all crisp inputs are converted into fuzzy variables. Fuzzification

process partitions the universe of discourse spanned by each variable into a number of fuzzy sets, assigning a linguistic variable. Fuzzy sets are represented by fuzzy membership functions (MF). Different types of membership functions can be used for fuzzification process.

2.3.1 Membership Function

In fuzzy systems fuzzy sets are represented by fuzzy membership functions (MF). Triangular, trapezoidal, Gaussian, are some of the membership functions, used as per the convenience and nature of the problem. Figure 1 describes about shape of the membership functions.

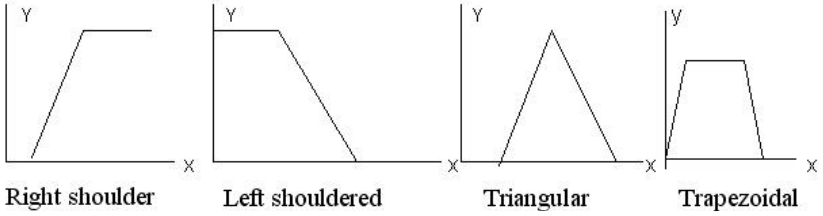

Figure 1. Membership functions

For every crisp value, membership functions represent a degree, which indicates the strength of its concern to the fuzzy set. The degree of membership is always in the range [0,1]. For example, any crisp value has a membership degree (or degree of support) 0, indicating that the value is not in the fuzzy set. Degree 1, means, the value is completely within the fuzzy set. The following equations explain, how to calculate a membership value ($\mu(x)$) for a crisp value x, by using triangular membership function.

$$\mu(x) = \begin{cases} 0, & \text{if } x < a_1 \\ (x - a_1)/(a_2 - a_1), & \text{if } a_1 <= x <= a_2 \\ (a_3 - x)/(a_2 - a_3), & \text{if } a_2 <= x <= a_3 \\ 0, & \text{if } x > a_3 \end{cases}$$

where a_1, a_2 and a_3 are the x co-ordinates of the start, middle end points of a triangular fuzzy membership function. The X–axis represent the crisp values of the parameter intended to be fuzzified. The constants a_1, a_2, a_3 are chosen by the designer from experience and depends on the range of crisp value and the particular fuzzy set LOW, MEDIUM, etc. The number of fuzzy categories for a specific application area is also decided on the basis of experience. Both number of fuzzy categories and the parameters a_1, a_2, a_3 for each category can be fine tuned by variational studies.

2.4 Fuzzy Inference

Fuzzy inference is a process of formulating a mapping from inputs to output using fuzzy logic. In fuzzy inference, knowledge is expressed in terms of fuzzy 'if then' rules. The computation of fuzzy inference consists of two components: Aggregation - computation of 'if' part, and Composition - computation of 'then' part. The most common inference methods AND, OR and NOT are used in majority of fuzzy logic applications.

$$\text{AND}: \mu_{A \cap B} = \min \{\mu_A, \mu_B\}$$

$$\text{OR}: \mu_{A \cup B} = \max \{\mu_A, \mu_B\}$$

$$\text{NOT}: \mu_{-A} = 1 - \mu_A$$

In case a numerical output is required from the inference process, a defuzzification step must follow the fuzzy inference. A weight factor is used to tune the fuzzy logic system for optimal performance. The weight factor represents the strength of the fuzzy rule within the interval [0,1]. It is multiplied with the aggregation result in the composition step.

2.5 Defuzzification

Defuzzification process translates fuzzy values to crisp values. Most of the fuzzy

systems require this process to visualize the result in a numerical form rather than in fuzzy term. Defuzzification step also requires membership functions shown in figure 2.

The following are some of the methods used in fuzzy systems.

1. Center of area
2. Center of largest area
3. Mean of maximum
4. First of maxima

Among these center of area method is used in most of the applications. The following equation represents computation of center of area defuzzification method.

$$U^* = \Sigma (A_i * L_i) / \Sigma A_i \quad (i = 1)$$

where A_i represents area covered under each fuzzy set and L_i represents mid point of the fuzzy set on X-axis.

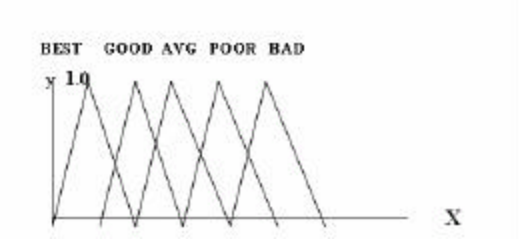

Figure 2. Membership functions for defuzzification

3. METRICS

A metric is a measure and in our case it is a measure of match or mismatch between two strings. A variety of metrics can be designed for computation of match/mismatch between two strings. Careful selection of an appropriate metric is a critical step for the algorithm It was seen that a metric computed on a single property of the strings is inadequate in view of the complexities mentioned. An integrated metric computed from carefully selected two or three base metrics with suitable weight factor gives an acceptable solution in most application areas.

In our system three basic mismatch metrics have been considered – character mismatch, triad/tri gram mismatch and soundEx mismatch.

Character mismatch metric is the usual root mean of squares of differences of alphabet counts in the two strings. Triad/tri gram mismatch is computed by the same expression except that here alphabet count is replaced by triad count in the two strings. SoundEx mismatch is again same as character mismatch where character counts, which includes alphabets and numerals counts are taken for the corresponding soundEx code strings.

4. IMPLEMENTATION

For the implementation of fuzzy logic based search support algorithm, a digital library database is considered on a relational database management system (ORACLE). Figure 3 describes the E-R model of the database.

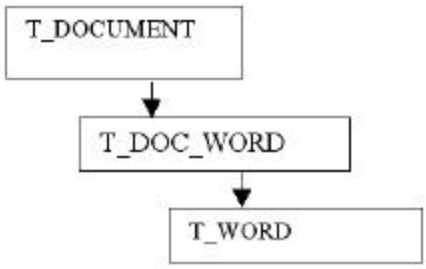

Figure 3. E-R model

The above E-R model contains 3 tables. The following table 1 describes data tables and attributes.

Table 1. Data tables and attributes

Table Name	Attributes
T_DOCUMENT	Acc_no, Title
T_WORD	Word_no, Word
T_DOC_WORD (Link Table)	Word_no, Acc_no

The T_DOCUMENT table is the base database table where the stored string is a field (attribute) in each record (tupple). The system is so designed that with each entry in the T_DOCUMENT table data, other two tables are updated.

The search process starts with entry of a target string. The target string is preprocessed to eliminate the predefined stop words like is, at, for and the remaining set of words (Z) of the target string are used for searching the database.

Starting with the set of words (Z) the search process gives a set of best matches and identifiers from the database. In order to achieve this, it uses the following step by step procedure described in the block diagrams (figure 4 and figure 5). Detailed description of some important steps are given subsequently.

4.1 Computation of Character Mismatch (CM)

Character mismatch is calculated between any two words (word from target string and a word from stored string, in the present context). For each word stored in word table character mismatch is calculated by using the following equation

$$CM = \Sigma(T_C \sim S_C)$$

Where subscript c represent a character from union of character sets for the two words, T_C represents character count for each character C in the target word, S_C represents corresponding character count in the stored word, ~ stands for absolute difference and Σ for summation over all characters represented by subscript c of both words. In order to improve more efficiency, it is better to precompute occurrences of characters used in word. It is scaled to the range (0-100).

4.2 SoundEx Mismatch

SoundEx mismatch is calculated for two given words and it gives mismatch between (0-100). The soundEx code for each word contains a character followed by 5 digits.

4.3 Triad/Tri-grams Mismatch

Triad mismatch (T_M)=Difference in number of occurrences of each/union number of

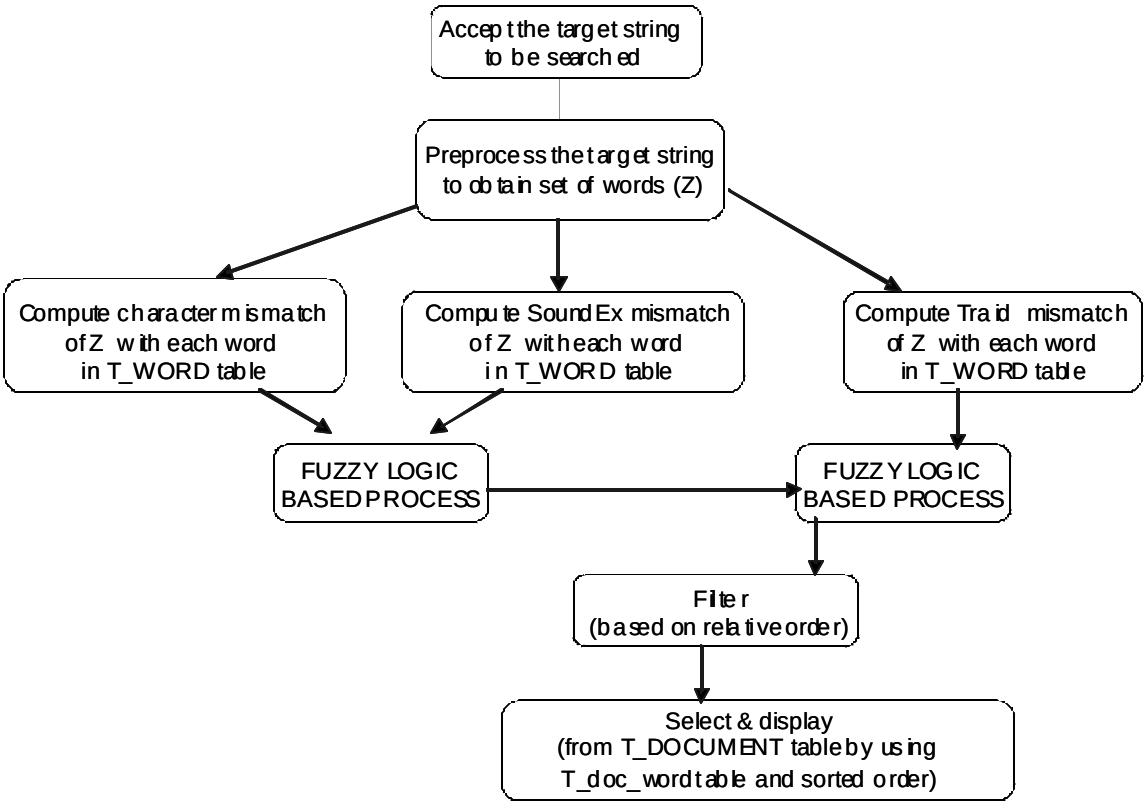

Figure 4. Information flow in the search process

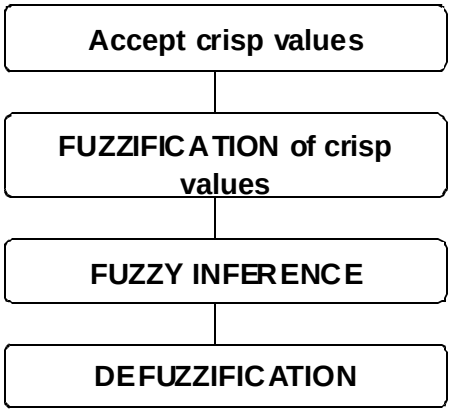

Figure 5. Information flow in fuzzy logic based process

occurrences of all triads. The output is triad mismatch value between two words. Low value indicates high match. The output value is scaled to the range (0-100).

In order to find the relative matching order of the word based on these metrics i.e., using a fuzzy based technique fuses character, soundEx and tri-gram mismatches.

4.4 Fuzzification of Metric Values Using Membership Functions

The following membership functions are defined for all metrics obtained by using the above steps. For the present problem selected MF's are given below.

1. Character mismatch ∈ {LOW, MEDIUM, HIGH}
2. SoundEx mismatch ∈ {LOW, MEDIUM, HIGH}
3. Triad mismatch ∈ {LOW, MEDIUM, HIGH}

This process converts the crisp values into corresponding fuzzy values. For the current problem, three fuzzy sets are recognized in the universe of discourse. These sets are LOW, MEDIUM and HIGH. To represent the above fuzzy sets the following triangular membership functions are chosen. For each scaled crisp input value corresponding to one of the three metrics - Char, SoundEx and Triad mismatches, this process gives fuzzy output sets with degree of support, by using the formula given in 2.3.1. Figure 6 shows membership functions for fuzzification.

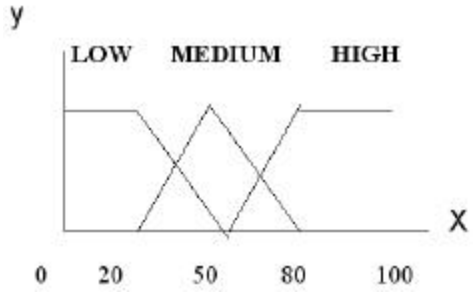

Figure 6. Membership functions for fuzzification

4.5 Fuzzy Inference

The strength of the fuzzy logic system depends on fuzzy inference. The fuzzy rules of the system represent the knowledge of the system. By adjusting the weight values of the fuzzy rules, fuzzy logic system is tuned for better performance. To define fuzzy inference, output fuzzy sets are selected. For the present technique, 5 output fuzzy sets (BEST, GOOD, AVERAGE, BAD and POOR) are considered,. The output fuzzy set "BEST" indicates the nearest match and the set "POOR" indicates farthest match of the strings. Table 2 describes the typical fuzzy inference with two inputs (fuzzy values of char mismatch and soundEx mismatch) and one output. The weight value indicates the strength of the rule. The output of each rule is a fuzzy set.

Table 2. Typical fuzzy inference with two inputs—fuzzy values of char mismatch and soundEx mismatch, output and weight

Rule	Input1	Input2	Output	Weight
1	LOW	LOW	BEST	1.0
2	LOW	MED	AVG	1.0
3	LOW	HIGH	BAD	1.0
4	MED	LOW	GOOD	1.0
5	MED	MED	AVG	1.0
6	MED	HIGH	BAD	1.0
7	HIGH	LOW	GOOD	1.0
8	HIGH	MED	POOR	0.5
9	HIGH	HIGH	POOR	1.0

By using aggregation and composition steps this process gives output fuzzy sets with degree of support.

4.6 Defuzzification

This process accepts the output of fuzzy inference process in terms of fuzzy sets with

Figure 7. Defuzzification Process

degree of support. Center of the area method is chosen for defuzzification, in order to compute crisp output from the fuzzy inputs. Figure 7 shows defuzzification process.

4.7 Filtering

This process receives the match percentage of user search word with database words from the defuzzification process. For each search word it selects a finite number of best matches based on a predefined cutoff number.

4.8 Select & Display

Based on the best matches (words from the T_word table), corresponding information is selected by using T_document and T_doc_word tables. One word in T_doc_word table should, in general, have a number of links to T_DOCUMENT table. Intersection of the sets of link strings corresponding to each matched word in T_doc_word is displayed on the screen. In case a display set is null, then the cutoff number may be adjusted and in extreme cases union of sets may be taken instead of intersection for display.

5. CONCLUSIONS

In this paper, an effort has been made to apply fuzzy logic to produce a search support technique for searching a database. Based on the algorithm designed, software was developed and tested on a database with different search strings by introducing noise. The results convey that fuzzy logic based search support technique is able to retrieve the information in the presence of noise in the search strings. However, this algorithm in the present form suffers from search speed problem. It is possible to overcome computation time problem by partition of search space. Fuzzy logic system is not capable of learning of inference rules, it is therefore necessary that proper fuzzy rules are identified at the beginning and tuned for the weights separately. Selection of appropriate metrics is critical in the search process.

BIBLIOGRAPHY

1. Altrock, Constantine Von. Fuzzy logic and neurofuzzy applications in business and finance. Prentice Hall PTR, New Jersy. 1997.
2. Reznik, Leonid. Fuzzy controllers. Newnes Oxford ox2 BDP. 1997.
3. Klir, G.J., *et al.* Fuzzy set theory, foundations and applications. INTL ED, Prentice Hall. 1997.
4. Valluru, B.R. & Hayagriva, V.R. C++ neural networks and fuzzy logic. Ed.2, BPB Publications. 1996.
5. Swaminathan, R, *et al*. Computer recognition method as applied to codification process for inventory system of a large multidisciplined laboratory. *Def Sci J,* 1988, **38**(2), 105-17.

Contributors: **Sh. SGK Murthy**, Scientist, Defence Research & Development Laboratory (DRDL), Kanchanbagh PO, Hyderabad–500 058, Tele: 91-040-24583394, e-mail: sgk_murthy@rediffmail.com

Dr. RN Biswas, Retd Scientist, DRDL, Kanchanbagh PO, Hyderabad–500 058.