INFORMATION RETRIEVAL USING XQUERY PROCESSING TECHNIQUES

E.J.Thomson Fredrick[1] and G.Radhamani[2]

[1] Research Scholar, Research & Development Centre, Bharathiar University, Coimbatore
thomson500@yahoo.com
[2] Director & Professor, Dr.G.R.D College of Science, Coimbatore

ABSTRACT

In recent years, the extraction of data from XML documents is an important issue for XML research and development. Fuzzy processing techniques have been proposed for flexible querying to Native XML Databases. We propose the fuzzy XQuery processing techniques for Native XML database systems, where the weights of attributes can be described by linguistic terms represented by fuzzy numbers. The proposed fuzzy XQuery processing techniques allow the users to use linguistic terms in the XQueries represented by fuzzy sets. The proposed Fuzzy XQuery processing applies the arithmetic operations of fuzzy sets. The proposed fuzzy query processing techniques can deal with the users' fuzzy queries in a more flexible and more intelligent manner. Our proposed research work would be a new step towards a more flexible XQuery language for Native XML Databases.

.

KEYWORDS

XML, XQuery, Fuzzy XQuery,Native XML Database

1. INTRODUCTION

XML(Extensible Markup Language) is becoming a dominant standard for storing and exchanging information. With its increasing use in areas such as data warehousing and e-commerce, there is a rapidly growing need for rule-based technology to support reactive functionality on XML repositories. Since XML is used as a defacto standard for communicating information on the Web, we need new techniques to process and retrieve XML data from XML repositories. XML Database vendors rushed to enrich their products with more flexible and advanced features to make them satisfy the requirements of modern applications [8]. X*ML* has become a standard format to exchange information over the Internet, and the importance of database technologies that support storage, processing, and delivery of XML is still increasing [5].

Most of the existing XML query languages are based on SQL. Unlike queries on traditional relational databases whose results are always flat relations, the results for XML queries are complex. Querying XML data involves two key steps: query formulation and efficient processing of the formulated query. The current state of the art in querying XML data is represented by XPath and XQuery, both of which rely on Boolean conditions. Boolean selection is too restrictive when users do not use or even know the data structure precisely. In this paper we describe a XML querying framework, called FuzzyXQuery, based on Fuzzy Set Theory, relying on fuzzy conditions for the definition of flexible constraints on stored data.

The rest of the paper is organized as follows. In Section 2, we review the research work done on Fuzzy SQL operations on Databases and Fuzzy Logic based operations on XML. In

DOI: 10.5121/ijdms.2011.3104

section 3, we briefly review some basic definitions of fuzzy sets from [9]. In section 4, we present Fuzzy XQuery processing techniques for XML database. The conclusion and future work is discussed in section 6.

2. Literature Review

In [2], Bosc, P. Pivert,O. (1995) proposed SQLf as a flexible querying language for relational databases conceived to be a complete extension of SQL with fuzzy logic. Fuzzy queries supported by SQLf involve fuzzy terms whose semantic depends of the user and the application domain. In [7], Shyi-Ming Chen and Yu-Chuan Chen,2003 presented new fuzzy query processing techniques for fuzzy database systems, where the weights of attributes of the user's queries can be represented by fuzzy numbers. The proposed fuzzy query processing techniques also allow the users to use linguistic terms in the queries represented by fuzzy sets. In [3], Buche *et al.* 2006 proposed a fuzzy-based XML querying system that performs approximate comparisons between query and data trees. This technique supports imprecise data via possibility distributions. In [4], Calms *et al.* 2007 discussed some issues raised in fuzzy querying by handling semi-structured information.

In [6], Marlene Goncalves and Leonid Tineo, 2007 were inspired by the research work of P. Bosc and O. Pivert,1995 and proposed more flexibility to XQuery by means of fuzzy logic use. As a step for the XQuery extension, they have focused their attention to XPath expressions. They specified fuzzy terms through XML using the XML Schema language. Bhowmick,S.S. and Prakash,S. (2006) proposed a efficient and faster XML Query processing in RDBMS using GUI-driven approach using prefetching algorithm. But this paper proposes automatic generation of XQuery and Fuzzy XQuery using GUI based approach. The proposed fuzzy XQuery processing techniques also allow the users to use linguistic terms in the XQueries represented by fuzzy sets. Then the fuzzy set values will be defuzzified by using the centroid method. In our earlier work, we demonstrate how Fuzzy Logic based XQuery operations provide better output than normal XQuery operations [8]. But this paper extends our earlier work. This paper proposes Fuzzy XQuery processing techniques based on Fuzzy sets for Native XML Database systems.

3. Basic Concepts of Fuzzy Sets

The theory of fuzzy sets was proposed by Zadeh in 1965 [9]. Let U be the universe of discourse, $U = \{u_1, u_2, \ldots, u_n\}$. A fuzzy set N in the universe of discourse U can be represented by

$$N = \{u_i, f_N(u_i)) \mid u_i \in U\} \qquad (1)$$

Let N and L be two fuzzy sets of the universe of discourse U, $U = \{u_1, u_2, \ldots, u_n\}$ and let f_N and f_L be the membership functions of the fuzzy sets N and L, respectively, where $f_N : U \to [0,1]$,
$f_L : U \to [0,1]$, $N = \{u_i, f_N(u_i)) \mid u_i \in U\}$, and $L = \{u_i, f_L(u_i)) \mid u_i \in U\}$. The union of the fuzzy sets N and L, denoted as $N \cup L$, is defined by

$$N \cup L = \{u_{i,} f_{N \cup L}(u_i)) \mid f_{N \cup L}(u_i) = Max\ (f_N(u_i), f_L(u_i)), u_i \in U \qquad (2)$$

The intersection of the fuzzy sets N and L, denoted as N ∩L, is defined by

$$N \cap L = \{u_{i,} f_{N \cap L}(u_i)) \mid f_{N \cap L}(u_i) = Min\ (f_N(u_i), f_L(u_i)), u_i \in U \qquad (3)$$

The cardinality |*N*| of the fuzzy set N is defined by

$$\left| N \right| = \sum_{i=1}^{n} f_N(u_i) \qquad (4)$$

The simplified arithmetic operations of triangular fuzzy numbers are described in the following.

Let E and L be two triangular fuzzy numbers, where

$E = (a_1, b_1, c_1)$

$L = (a_2, b_2, c_2)$

(1) Fuzzy Numbers Addition $\oplus$:

$$E \oplus L = (a1, b1, c1) \oplus (a2, b2, c2) = (a1 + a2, b1 + b2, c1 + c2). \qquad (5)$$

(2) Fuzzy Numbers Subtraction

$$\mathrm{E} \ \Theta \ \mathrm{L} = (a1,\ b1,\ c1)\ \Theta\ (a2,\ b2,\ c2) \ = \ (a1 - c2,\ b1 - b2,\ c1 - a2). \qquad (6)$$

(3) Fuzzy Numbers Multiplication $\otimes$:

$$E \otimes L = (a1, b1, c1) \otimes (a2, b2, c2) = (a1 \times a2, b1 \times b2, c1 \times c2) \qquad (7)$$

(4) Fuzzy Numbers Division Ø:

$$E \ Ø \ L \ = \ (a1,\ b1,\ c1)\ Ø\ (a2,\ b2,\ c2) \ = \ (a1/c2,\ b1/b2,\ c1/a2). \qquad (8)$$

4. Fuzzy XQuery Processing Techniques For XML Database

Fuzzy XQueries are based on fuzzy set theory proposed by [3], whose goals are to store imprecise data, to process user's imprecise queries, and to provide proper information to users to overcome the drawbacks of the normal XQuery Operations. Fuzzy XQueries provide a representation scheme for dealing with vague or uncertain concepts.

The following are the limitations of XQuery with Boolean logic

- XQuery is forced to make arbitrary determinations about what it can do.
- XQuery does not fit the exact criteria people have in their minds.
- XQuery commands are executed on the basis of only crisp or classical logic.

Unlike Boolean logic Fuzzy XQueries deal with data that is vague, ambiguous, incomplete and imprecise. Instead of applying crisp boundaries to delineate the search space, the space can be represented linguistically using the concept of fuzzy logic [8]. In a relational fuzzy database system, the users can use linguistic terms to describe the weights of query items. For example, consider the following fuzzy SQL statements of the user's query shown as follows:

SELECT rollno,name from students where height='Very Tall' and Weight='Heavy'

In the above Fuzzy SQL, Very Tall and Heavy are linguistic terms represented by triangular fuzzy numbers [7].

Fuzzy logic provides a flexible and fluid method of defining semantic concepts within the Native XML database and provides the basis for a much richer and much more powerful method of looking through a XML database. In a fuzzy XQuery, the selected records are ranked according to their compatibility with the semantics – the intent - of the query. This provides a measure of how well a record fits in with the complete set of XML records retrieved.

The fuzzy membership value for the Fuzzy XQuery will be calculated using the following formula. If we know the attribute value x, the lower range value a, and higher range value b,

Fuzzy membership value for x can be calculated using the following formula :

$$f_{close\ to\ a}(x) = \frac{1}{1+\left(\frac{x-a}{b-a}\right)^2} \qquad \textbf{(9)}$$

For example, assume that the value of the attribute AGE of a tuple in a XML database is “25” and the query condition of the user’s query is "AGE = young", then the degree of matching of the tuple with respect to the user’s query "AGE = young" is equal to

$$f_{young}(25) = \left(1+\left((25-20)/15\right)^2\right)^{-1} = 0.9$$

In a XML database system also, the users can use linguistic terms to describe the weights of query items same like Fuzzy Database systems. These linguistic terms are represented by fuzzy triangular sets. The linguistic terms based fuzzy weights and the corresponding triangular fuzzy triangular sets are shown in the following Table 1.

TABLE I
TRIANGULAR FUZZY NUMBER CORRESPONDING TO EACH LINGUISTIC VALUE

Linguistic Variable	Triangular Fuzzy Number
Absolutely Low (AL)	(0.0,0.0,0.0)
Extremely Low (EH)	(0.0,0.1,0.2)
Very Low (VL)	(0.1,0.2,0.3)
Low (L)	(0.2,0.3,0.4)
Medium (M)	(0.4,0.5,0.6)
High (H)	(0.6,0.7,0.8)
Very High (VH)	(0.7,0.8,0.9)
Extremely High (EH)	(0.8,0.9,1.0)
Absolutely High (AH)	(1.0,1.0,1.0)

In Fuzzy XQuery, if the weight γ of an attribute A is a crisp value represented as "WEIGHT A = γ", where γ is a real value between 0 and 1, then we can extend the crisp value γ into the triangular fuzzy number representation (γ, γ, γ). For example, if the weight of an attribute is 0.6, then we can extend the value 0.6 into the triangular fuzzy number representation (0.6, 0.6, 0.6).

Let W_1, W_2 be two triangular fuzzy numbers representing the fuzzy weights of the attributes $\boldsymbol{A_1}$ and $\boldsymbol{A_2}$, respectively, where
W_1 = (a_1,b_1,c_1)
W_2 = (a_2,b_2,c_2)
and $\overline{W_1}$ and $\overline{W_2}$ are "fuzzy scaled weights" of W_1 *and* W_2. Then, by using the fuzzy number arithmetic operations, we can get

$$\overline{W_1} = W_1 \varnothing \left(W_1 \oplus W_2\right)$$
$$\overline{W_1} = \left(a_1, b_1, c_1\right) \varnothing \left(a_1 + a_2, b_1 + b_2, c_1 + c_2\right)$$
$$\overline{W_1} = \left(\frac{a_1}{c_1 + c_2}, \frac{b_1}{b_1 + b_2}, \frac{c_1}{a_1 + a_2}\right) \qquad (10)$$

$$\overline{W_2} = W_2 \varnothing \left(W_1 \oplus W_2\right)$$
$$\overline{W_2} = \left(a_2, b_2, c_2\right) \varnothing \left(a_1 + a_2, b_1 + b_2, c_1 + c_2\right)$$
$$\overline{W_2} = \left(\frac{a_2}{c_1 + c_2}, \frac{b_2}{b_1 + b_2}, \frac{c_2}{a_1 + a_2}\right) \qquad (11)$$

$\varnothing$ and $\oplus$ are fuzzy sets division and fuzzy sets addition operators respectively. Both the fuzzy sets addition and fuzzy sets division operations are explained in the section 3. Let us assume that W_1 is assigned with the linguistic variable based weight "high" and W_2 is assigned with the linguistic variable based weight "very high". According to Table 1, the triangular fuzzy set for "high" is (0.6,0.7,0.8) and the triangular fuzzy set for "very high" is (0.7,0.8,0.9). Then the fuzzy weights $\overline{W_1}$ and $\overline{W_2}$ are calculated according to the equations (10) and (11).

$$W_1 = \left(a_1, b_1, c_1\right) = (0.6, 0.7, 0.8)$$
$$W_2 = \left(a_2, b_2, c_2\right) = (0.7, 0.8, 0.9)$$

$$\overline{W_1} = \left(0.6, 0.7, 0.8\right) \varnothing \left(1.3, 1.5, 1.8\right)$$
$$\overline{W_1} = \left(\frac{0.6}{1.7}, \frac{0.7}{1.5}, \frac{0.8}{1.3}\right)$$
$$\overline{W_1} = \left(0.3, 0.4, 0.6\right) \qquad (12)$$
$$\overline{W_2} = \left(0.7, 0.8, 0.9\right) \varnothing \left(1.3, 1.5, 1.8\right)$$
$$\overline{W_2} = \left(\frac{0.7}{1.7}, \frac{0.8}{1.5}, \frac{0.9}{1.3}\right)$$
$$\overline{W_2} = \left(0.4, 0.5, 0.6\right) \qquad (13)$$

The implementation of the Equations (10) and (11) in the Fuzzy XQuery operation is explained using an XQuery example.

Assume that the user wants to find the Employee id,name,Age and Salary of employees who are young and whose salary is high, then the Fuzzy XQuery can be expressed as follows:

```
<output> {
for $emp in doc(emp.xml)/employees/record
let $eid := $record/empid/text()
let $en := $record/ename/text()
 let $age:=$record/age/text()
 let $salary:=$record/salary/text()
 return if $ age=very young and $salary=low and   $ fuzzy_age weight=W1  and $
fuzzy_salary_weight= W2  then
<record>  <empid> {$eid} </empid>
<ename> {$en} </ename>
<salary>{$sa}</salary>
<age>{$a}</age> </record>
else () }
</output>
```

where $\boldsymbol{W_1}$ and $\boldsymbol{W_2}$ are the fuzzy weights of the attributes Age and Salary.

Let us assume that W_1 is assigned with “high” and $\boldsymbol{W_2}$ is assigned with “very high”. These linguistic terms are assigned from Table 1. But $\overline{W_1}$ and $\overline{W_2}$ are the fuzzy scaled weights of W_1 and W_2 respectively , where

$$\overline{W_1} = W_1 \O (W_1 \oplus W_2\ ) \ = \ (a_1, b_1, c_1) .$$

$$\overline{W_2} = W_2 \O (W_1 \oplus W_2\ ) \ = \ (a_2, b_2, c_2) .$$

Assume that the Fuzzy membership value of the query condition "AGE = very young" is 0.9 and assume that the fuzzy membership value of the query condition "SALARY = low" is 0.7, then we extend the value 0.9 to the triangular fuzzy number (0.9, 0.9, 0.9) and extend the value 0.7 to the triangular fuzzy number (0.7, 0.7, 0.7). After performing the fuzzy number arithmetic operations, we can get the degree of matching of the XML record with respect to the user’s weighted fuzzy query represented by a fuzzy number F, where the calculation process is shown as follows:

$$F = \overline{W_1} \otimes (0.9, 0.9, 0.9) \oplus \overline{W_2} \otimes (0.7, 0.7, 0.7)$$

The values for $\overline{W_1}$ and $\overline{W_2}$ are assigned from Equations (12) and (13).

$$= (0.3, 0.4, 0.6) \otimes (0.9, 0.9, 0.9) \oplus$$

$$(0.4, 0.5, 0.6) \otimes (0.7, 0.7, 0.7)$$

$$= (0.27, 0.36, 0.54) \oplus (0.28, 0.35, 0.42)$$

$$= (0.55, 0.71, 0.96)$$

Then, we can use a defuzzification method to defuzzify the triangular fuzzy number F into a crisp value. The value is regarded as the matching degree of the XML record with respect to the user’s weighted fuzzy query. In the following, we describe how to defuzzify a triangular fuzzy set into a crisp value [7]. Assume that F is a triangular fuzzy number, G = (a, b, c), and Def(F) denotes the defuzzified value of the triangular fuzzy number G, then

$$Def(F) = \frac{a + 2b + c}{4} \qquad (14)$$

4.1. The Algorithm for Fuzzy XQuery Processing

If K_i is a linguistic term represented by a fuzzy set and the Data D_i is a crisp value **then** compute the Fuzzy matching degree $F(D_i(A))$ using the formula (9);

If $F(D_i(A))$ is a crisp value, then extend the crisp value $F(D_i(A))$ into the triangular fuzzy number representation $(F(D_i(A)), F(D_i(A)), F(D_i(A)))$;

If $F(D_i(B))$ is a crisp value, then extend the crisp value $F(D_i(B))$ into the triangular fuzzy number representation $(F(D_i(B)),F(D_i(B)),F(D_i(B)))$;

If W_1 is a linguistic term **then** find the corresponding triangular fuzzy number of the linguistic term based on Table 1;

If W_2 is a linguistic term **then** find the corresponding triangular fuzzy number of the linguistic term based on Table 1;

If W_1 is a crisp value, where $W_1 \in [0, 1]$, **then** extend W_1 into the triangular fuzzy number representation (W_1, W_1, W_1);

If W_2 is a crisp value, where $W_2 \in [0, 1]$, **then** extend W_2 into the triangular fuzzy number representation (W_2, W_2, W_2);

Let W_α and **W_β** be the Fuzzy scaled weights of **W_1** and **W_2** respectively, where

$$W_\alpha = W_1 \varnothing (W_1 \oplus W_2),$$
$$W_\beta = W_2 \varnothing (W_1 \oplus W_2);$$

Find the fuzzy matching degree $F(R_i)$ of the XML Record R_i where

$$F(R_i) = (\mathrm{F}(\mathrm{D}_i(\mathrm{A})), \mathrm{F}(\mathrm{D}_i(\mathrm{A})), \mathrm{F}(\mathrm{D}_i(\mathrm{A})))$$
$$\otimes W_\alpha \oplus (\mathrm{F}(\mathrm{D}_i(\mathrm{B})), \mathrm{F}(\mathrm{D}_i(\mathrm{B})), \mathrm{F}(\mathrm{D}_i(\mathrm{B}))) \otimes W_\beta$$

W_α and W_β are the fuzzy scaled weights of W_1 and W_2 respectively, and "$\otimes$" and "$\oplus$" are the multiplication operator and the addition operator of the fuzzy numbers respectively.
Calculate the defuzzified value $Def(F(R_i))$ of $F(R_i)$ based on formula (14), where $Def(F(R_i)) \in [0,1]$

If any XML record satisfies the defuzzified fuzzy value, then display the XML record.
Else
  the XML Record Ri does not satisfy the user's query;
End For;
End;

The above algorithm will display the result of the user's Fuzzy XQuery according to the Fuzzy membership values of the XML records satisfying the user's query in a descending sequence.

The working of Fuzzy XQuery is illustrated in the following diagram.

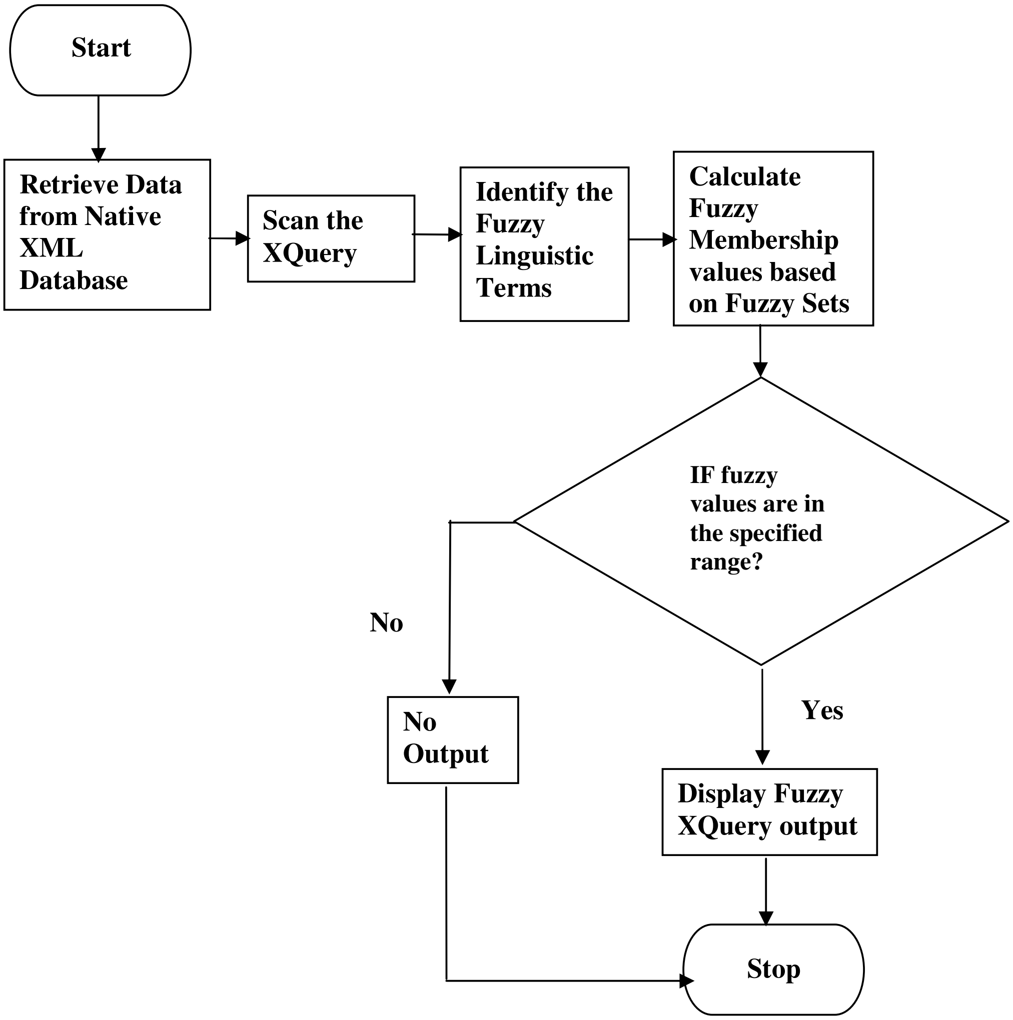
Start
Retrieve Data from Native XML Database
Scan the XQuery
Identify the Fuzzy Linguistic Terms
Calculate Fuzzy Membership values based on Fuzzy Sets
IF fuzzy values are in the specified range?
No
No Output
Yes
Display Fuzzy XQuery output
Stop

Figure. 1. Working of fuzzy xquery

REFERENCES

[1] Alessandro Campi1, Ernesto Damiani, Sam Guinea, Stefania Marrara, Gabriella Pasi, and Paola Spoletini, 'A Fuzzy Extension of the XPath Query Language', Journal of Intelligent Information Systems,Vol.33 Issue 3, December 2009.

[2] Bosc, P., Pivert,O., 'SQLf: A Relational Database Language for Fuzzy Querying', IEEE Transactions on Fuzzy Systems,Vol 3, No.1,February,1995.

[3] Buche, P., Dibie-Barthèlemy, J., and Wattez, F.. 'Approximate querying of XML fuzzy data'. In springer (Ed.), *Proceedings of the 7th international conference FQAS ,* (Vol. 4027/2006). Milan,Italy, 2006.

[4] Calms, M. D., Prade, H., & Sdes, F. 'Flexible querying of semistructured data: A fuzzy-set based approach'. International Journal of Intelligent systems, Vol.22, pp. 723-737, July,2007.

[5] Gang Gou, Chirkova,R '*Efficiently Querying Large XML Data Repositories: A Survey'*, IEEE Transactions on Knowledge and Data Engineering, October,2007.

[6] Marlene Goncalves and Leonid Tineo, 'A new step towards Flexible XQuery", Journal of Revista Avances en Sistemas e Informática',Vol.4 No.3,December,2007

[7] Shyi-Ming Chen and Yu-Chuan Chen, '*New fuzzy query processing techniques for fuzzy database systems'*, International Journal of Fuzzy systems, Vol.5, pp. 161- 170,2003.

[8] Thomson Fredrick,E.J, G.Radhamani,G. "Fuzzy Logic based XQuery Operations for Native XML Database Systems", International Journal Database Theory and Application, Vol 2, No.3, pp.13-20,September,2009

[9] Zadeh,L.A. 'Fuzzy Sets', *Information and Control*, Vol. 8, pp. 338- 353,1965.