Fuzzy Logic System for Querying a Database

Sunanda Chowdhary\textsuperscript{1}, Vipin Sharma\textsuperscript{2}, A. B. Patki\textsuperscript{3} and Suman Mann\textsuperscript{4}
\textsuperscript{1,2}Student, Computer Science Department, MSIT, GGS Indraprastha University
\textsuperscript{3}Senior Director/Scientist G&HOD, Department of Information Technology, Govt. of India
\textsuperscript{4}Reader, Computer Science Department, MSIT, GGS Indraprastha University
E-Mail: 1sunandachowdhary.2010@msit.in, 2vipinsharma.2010@msit.in, 3apatki@mit.gov.in, 
4sumanmann2007@gmail.com

\textbf{ABSTRACT}

Classical query languages are too restrictive in data extraction, search and deduction. Ever growing size of databases in organizations, universities, public databases, and internet has made it difficult, even for experienced users, to extract desired information. Classical SQL queries have remarkable capabilities in terms of data deduction and answer formation from information stored at widely dispersed databases. Ironically, human queries are rarely crisp which poses challenges in efficient answer formation and data retrieval. These are based on human perception which is grossly inexact and imprecise based on world knowledge. Integration of query languages with fuzzy logic can increase their scalability in data retrieval based on human perception. This paper presents a method for retrieval of data using extended fuzzy theory based version of SQL-like languages. We also propose two tier database administrator schemes for formation and maintenance of fuzzy databases in corporate computing environment. The eventual aim of this paper is to enlighten the importance of fuzzy technique in database queries and filter the role of database administrator. We also propose use of XML as a technology for constructing and maintaining fuzzy database owing to its capability of accessing and transferring information over web which further adds to its corporate viability. Feasibility of the project is established with the help of an illustration on stock market, with relational database maintained using XML in response of fuzzy SQL-like queries viz., most recent IPO, highest gainer of the week. Effective use of graphical user interface program is used to display results from relational database supporting ‘what-if’ scenarios.

\textbf{KEYWORDS}

Fuzzy query, linguistic variables, database administrator, XML

\textbf{1. MOTIVATION}

Usually, a user knows what he/she needs from a database depending on the grossly imprecise perception of the world knowledge. Classical query languages give either exact result or null result and do not inform about approximately similar results. In many cases user executes a script with a series of queries with different parameter values to extract desired information. Such an approach is devoid of cognitive support to infer from the databases.

In the past, attempts have been made to extend capability of classical query languages by integrating them with fuzzy logic system in Fuzzy Queries \cite{4,5,6,9}. Such amalgamation of two techniques allows user to view information in databases, which provides cognitive support to reduce information overloading allowing the user to use linguistic variables with hedges in data extraction. We introduce a concept of two-tier Database Administrator (DBA) schemes for better management of data in corporate computing environment for context sensitive information processing. Database administrator at regional/zonal level will be responsible for maintaining crisp database while corporate level DBA (fuzzy DBA) will seek maintenance of fuzzy database. Crisp database will be a base over which fuzzy database is constructed. In the subsequent sections, we introduce the relevant aspects Fuzzy set theory and illustrate through a case study of Fuzzy logic for querying a database.

\textbf{2. FUZZY SET THEORY AND FUZZY DATABASES}

The concept of fuzzy set theory, introduced by Lotfi A. Zadeh in 1965, is generalization of classical set theory. An element in fuzzy set represents a grade of its membership in fuzzy set \cite{5}. Let $U$ be universe of discourse, $x$ is generic element of $U$ such that $U = \{x, x \in U\}$. Then a fuzzy set $A$ in $U$ is characterized by membership function $\mu_A(x)$, which associates $x$ a real number in interval $[0, 1]$, where $\mu_A(x)$ represents grade of membership of $x$ in $A$. Nearer the value of $\mu_A(x)$ to unity, higher the grade of membership of $x$ in set $A$. In case of classical set theory membership function contains any of two values i.e. $\{0, 1\}$ \cite{2}.

When $A$ is a fuzzy set and $x$ is a relevant object, the proposition \textbf{“}$x$ is a member of $A$’ is not necessarily either true or false, as required by the two-valued logic, but it may be true only to some degree, the degree to which $x$ is actually a member of $A$, is a real number in the interval $[0, 1]$. Theoretically, if $X$ is a collection of objects denoted generically by $x$, then a fuzzy set $F$ in $X$ is a set of ordered pairs,

$$F = \{(x, \mu_A(x)) | x \in X\},$$

$\mu_A(x)$ is called the membership function (or grade of membership) of $x$ in $F$. The range of the membership function is a subset of the nonnegative real numbers whose supremum is finite.
Each input variable in fuzzy system have a membership function which can be plotted over membership graph like in Figure 1 [10]. The meaning of cold, warm, and hot is represented by membership function values over temperature scale. The vertical line in image shows that this temperature t is “not hot” (since hot line is at zero). The cold line describe this temperature as “very cold” and warm line as “slightly warm”.

Fuzzy logic mimics human control logic. It provides a simple way to arrive at a definite conclusion based upon vague, imprecise, ambiguous, noisy, or missing input information [8]. Fuzzy logic not only provides meaningful and powerful representation of measurement of uncertainties but also provides for a meaningful and powerful representation of vague concepts expressed in natural language as in the work of [4]. Fuzzy logic enables us to use linguistic variables like hot, cold, young, good etc. It also allow us to use hedges, adjective of variables, to further filter out elements like very hot, less cold, too young etc.

For retrieval of information and answer deduction capability, human perception of approximation has become highly momentous as never before. This has led to a shift from realm of crisp relational database model to realm of fuzzy real world database model. This has led to need of modeling imprecise information in relational databases as opposed to use of fuzzy queries, which is another viable option.

There can be two broad approaches:

- Allow for fuzzy queries on databases storing crisp data.
- Adding fuzzy information to the databases.

3. FUZZY QUERIES TO RELATIONAL DATABASES

The easiest way to incorporate world knowledge which is grossly imprecise and inaccurate is to present a database model that stores crisp entries but allows for fuzzy queries with the usage of an interactive graphical user interface program. Classical query languages are designed to extract data which fully satisfies given constraint e.g. “SELECT * FROM table WHERE age < 25 AND salary > 70000”. This kind of query will retrieve all tuples in database which satisfy given constraint, but if database does not contain any such tuples then there is no way to extract data which are somewhat similar to desired output.

Traditional query languages are incapable to retrieve data which are similar to the one desired by user. Therefore we have introduced concept of fuzzy logic in query language. Combination of fuzzy and query language enables user to extract desired information even in case where user doesn’t know exact denomination for querying. A fuzzy query will look like “SELECT * FROM table WHERE young age AND high salary”. Fuzziness corresponding to age is shown in figure 2. Membership value corresponding to age is denoted by variables alpha, beta, and gamma. This query extracts those tuples where age attribute is less and salary attribute is high in database. Fuzzy query relaxes constraint imposed on selection conditions by executing fuzzy algorithm. Tuples extracted in fuzzy query shares most resemblance to information desired by the user compared to rejected tuples.

One limitation to fuzzy query is that they are strictly related to application domain. It would be impossible to perform an effective, non-trivial, relaxing activity for generic application domain [6]. We developed an application on “Stock Market” to test our fuzzy query; snapshot of software is shown in figure 2.

In our implementation of fuzzy query for stock market we made a fuzzy class. It contains function to convert an attribute column into membership function values of that attribute. Our fuzzy class implements various other fuzzy operations like union, intersection, complement, sum, difference to name a few. It also support uses of hedges like very, extremely, moderately, slightly, vaguely [3].

4. FUZZY DATA IN RELATIONAL DATABASES

In this section we present another approach of capturing fuzziness present in the world. In previous section usage of fuzzy queries on crisp database was demonstrated. Here, we introduce not only fuzzy queries but also information stored in databases can itself be fuzzy (imprecise, inaccurate) [1].
There can be two approaches to it based on type of values associated with each attribute:

- An attribute can store crisp values
- An attribute can take crisp value or fuzzy value (linguistic values) as shown in table 1.

5. TWO LEVEL DATABASE ADMINISTRATOR SCHEMA

Corporate computing necessitates presence of data over widely spread network has simulated the need for two-tier architecture of database administrator. This concept is presented as overshadowing conventional one-tier architecture, in which database stores only crisp data values. Instead, in two-tier architecture, at higher level DBA maintains a database with crisp values. Whereas, at lower level a fuzzy database is constructed, this is maintained by fuzzy DBA. Fuzzy database stores membership values pertaining to each attribute. Thus, higher-level database act as input for lower level fuzzy database for calculating degree of membership corresponding each attribute in database.

This division of tasks has a profound influence on efficient maintenance of database and retrieval of information based on human perception of approximation. Also, transfer and access of large fuzzy database over network have become a feasible option.

6. USING XML AND FUZZY LOGIC TO ACCESS A RELATIONAL DATABASE: STOCK MARKET ILLUSTRATION

With the focus on corporate computing environment, XML is proposed as an enabling technology for transfer of data over World Wide [7]. Thus, in this illustration we use XML for constructing, maintaining and accessing of data from geographically dispersed databases. As mentioned earlier, two approaches can be used viz., fuzzy database or fuzzy queries to illustrate human perception based on experience and knowledge. In this illustration we present the later one (refer to table 1). C sharp is used for presenting a graphical user interface through which user can issue fuzzy queries. Amalgamation of C sharp and XML to implement fuzzy queries allows us to enter into the realm of real world which is inherently imprecise in nature.

Consider a query which was executed in our software:

```
SELECT company_name FROM stock_table WHERE dividend is very high AND opening_price is extremely low
```

From this we extracted attributes to be displayed in final result – company_name, name of table – stock_table, and predicates - dividend very high & opening_price extremely low. Our software first grades whole column values listed in predicates on scale zero to one to get membership function values of each attribute. Now hedges are applied to these membership functions to further refine data. Since predicate in query involves more than one attribute fuzzy operations are applied. At the end software generates name of companies which are in conformity with user query.

7. CONCLUSION AND FUTURE SCOPE

We have proposed two methods of incorporating fuzziness in relational database to do justice to approximation based human knowledge. Fuzzy queries to relational database are proposed as one candidate model and fuzzy relational database is presented as another relational model. With the help of illustration of stock market, we have successfully implemented both the methods using C sharp (GUI) and XML (constructing and maintaining database). This establishes feasibility of the concepts proposed.

This paper is an attempt to overcome restrictions posed by classical query languages by providing ample scope for queries based on one’s experience and education. A deviation from conventional one tier architecture of database administrator is presented in the form of two-tier database administrator architecture, which has further provided a concrete base for the two models proposed in this paper. Corporate computing
viability is established with the usage of XML as an enabling programming language (technology).

8. REFERENCES


