RESEARCH ARTICLE OPEN ACCESS

FUZZY SQL for Linguistic Queries

Poonam Rathee
Department of Computer Science
Aim &Act, Banasthali Vidyapeeth
Rajasthan
India

ABSTRACT

For Many Years, achieving unambiguous knowledge has been turned to a serious challenge for human being. The aim of this paper is to emphasize situation when classical {true, false} logic is not adequate for data selection and data classification. Linguistic expression like: high salary, young etc are very often used in life and in statistics. The goal of this paper is brief study of fuzzy logic and sets and how to make it suitable for database queries and classification tasks. Fuzzy approach is introduced along with usual relational database model to handle linguistic queries. The purposed fuzzy approach provides flexibility when users cannot unambiguously set hidden boundaries between data. In this paper, we extend the work of medina et al. to implement a new architecture of fuzzy DBMS based on the GEFRED model. This architecture is based on the concept of weak coupling with the DBMS SQL Server.

Keywords:- Fuzzy SQL, Fuzzy Classification, Database.

I. INTRODUCTION

Databases are a very important component in computer systems. Because of their increasing number and volume, good and accurate accessibility to a database becomes even more important. Organizations work with very large data collections mainly stored in relational databases. Linguistic expressions are interesting for data extraction, analysis, dissemination and decision making. The research area of fuzziness in Data Base Management Systems (DBMS) has resulted in a number of models aimed at the representation of imperfect information in Data Bases, or at enabling non-precise queries (often called flexible queries) on conventional database schemas. However, few works have been done from a practical point of view. Statistical indicators are often collected with some errors and vagueness and classical techniques may involve some inadequately selected or classified data.

II. IMPERFECT INFORMATION AND FUZZY THEORY

2.1. Imperfect Information

Fuzzy databases are used basically to handle imperfect information. Imperfect information can be inconsistent, imprecise, ambiguous, uncertain or vague. According to Z.M. Ma, imperfect information is [1]

1. Inconsistent when some real world aspect is having more than one value. (Example – the age of a student is stored as 25 and 27.)

2. Imprecise when attributes value choice has to be made from any given interval or range. (Example – the age of a student is the set [23, 24, 25, 26] or is in interval [50-65].

3. Ambiguous when some elements of information lead to various possible interpretations.

4. Uncertain when there exists degree of truth in attribute value. It occurs due to lack of information. (Example- the possibility that age of a student is 30 is 80 %.)

5. Vague when attribute value is represented by linguistic variables. (Example – the age of student is „Young".)

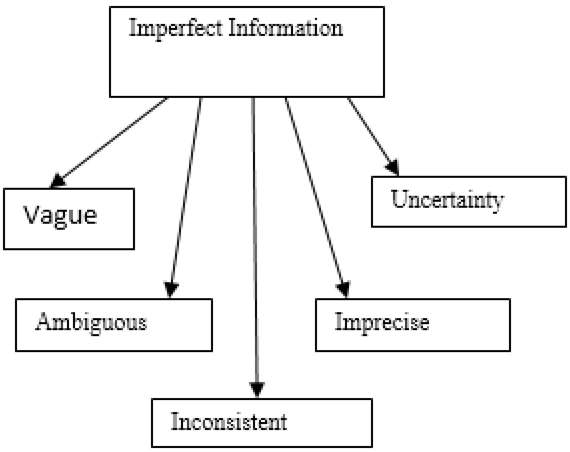

Figure 1. Types Of Imperfect Information

2.2 Classical Query

The SQL is used to obtain data from relational databases. Its advantages among others are the optimized work with RDBMS and understandable interpretation for users. The simple SQL is as follows:

select attribute_1,….., attribute_n

from T

where attribute_p>P and attribute_r<R.

The result of the query is shown in graphical mode in Figure 2. Values P and R delimit the space of interesting data. Small squares on the graph show records that satisfy and not satisfy the query criteria. In the graph is obviously shown that two records are very close to satisfy query criteria whereas other records either satisfy the query or are far for satisfying it.

To retrieve data from database user has to identify the interested data .Boundary values which are present in WHERE clause separates interesting data from non interesting data. But it is not always the case where user identifies interested data unambiguously. In cases when the user can not unambiguously separate data he is interested in from data he is not interested in by sharp boundaries or when user wants to obtain data that are very close to satisfy queries and to know the index of distance to full query satisfaction, it is necessary to adapt the SQL to these requirements.

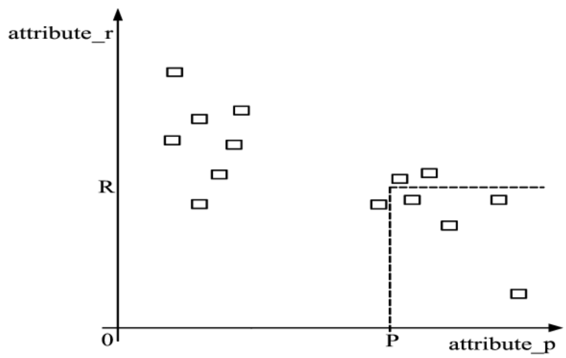

Figure 2 Result Of Classical Query

2.1.1Drawbacks of Classical Query System

The main drawback of classical query is that even the nearby values are present in database for a given query but the search show no result found. To get the result user have to expand the query again and again until he get the exact match in database. In short, classical query system makes a brittle selection. In this way more data from database are selected ,but user has lost the accuracy of his query. These two problems show the instead of changing the boundary conditions in the WHERE clause, it is necessary to change the way in which WHERE clause is evaluated.

The aim of this work is to present the query improvement with the fuzzy SQL approach. This development enables supporting queries based on linguistic expressions from user's point of view and also enables accessing classical relational databases in the unchanged structure.

2.2. Basic Preliminaries on Fuzzy Set Theory

Let X = {x1, x2,..., xn} be universe of discourse.

2.2.1. Fuzzy Sets

Fuzzy sets are extension and generalization of the basic concepts of crisp sets. It allows partial

Membership. A fuzzy set F over a universe of discourse X can be defined as set of ordered pairs[2].

$F = \{\mu F(x) / x: x \in X, \mu F(x) \in [0,1]\}$

Here, µF(x) is membership function of the element x to the fuzzy set F. So , a fuzzy set is thus defined by a function that maps objects in a domain of concern to their membership value in a set. Such a function is

known as membership functions (µ) whose value lies in range [0,1] i.e.

$\mu F(x): X \rightarrow [0,1]$

µF(x) = 0 indicates that x does not belong to the fuzzy set F.

µF(x) = 1 indicates that x completely belongs to the fuzzy set F.

The universe of discourse X can be classified as

1 .Finite or discreet universe of discourse X = {x1, x2,..., xn}, where a fuzzy set F can be represented by:

$F = \mu 1 / x1 + \mu 2 / x2 + ... + \mu n / xn$

Here, µi with i = 1, 2, n represents the membership degree of the element xi..

2.2.2 Different Forms Of Fuzzy Sets To Calculate Membership function

We are using fuzzy membership function to express fuzziness in the query. Zadeh proposed a series of membership functions that could be classified into two groups: those made up of straight lines, or "linear," and Gaussian forms, or "curved." In our work we are using linear Trapezoid function. For implementing fuzzy conditions following membership functions have been used in our thesis work which describes small values, about values and big values respectively.

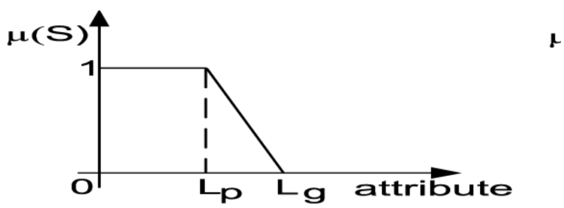

Figure 3(a) Fuzzy set for small value

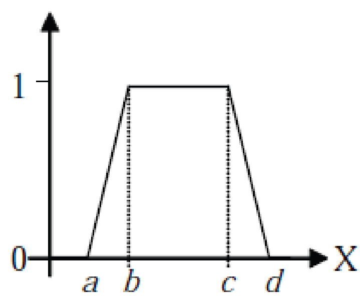

Figure 3(b) Fuzzy Sets For About Values

Method To Calculate membership function for about values

$$T(x) = \begin{cases} 0 & \text{if } (x \le a) \text{ or } (x \ge d) \\ (x-a)/(b-a) & \text{if } x \in (a,b) \\ 1 & \text{if } x \in [b,c] \\ (d-x)/(d-c) & \text{if } x \in (c,d) \end{cases}$$

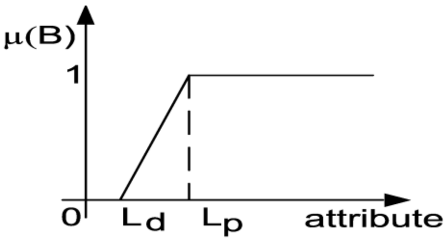

Figure 3(c) Fuzzy Set For Big Value

2.2.3. Fuzzy Union or t-conorms

If A and B are two fuzzy sets of the universe X, then the fuzzy union of A and B is denoted by A∪B and is defined as
A∪B = {(x, max { μA(x) , μB(x) }): x□X}

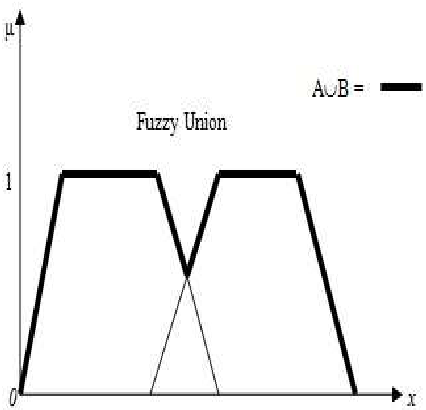

Figure 4(a) Fuzzy Union

2.2.4. Fuzzy Intersection or t-norms

If A and B are two fuzzy sets of the universe X, then the fuzzy intersection of A and B is denoted by A∩B and is defined as
A ∩B = {(x, min{ μA(x) , μB(x) }): x□X}

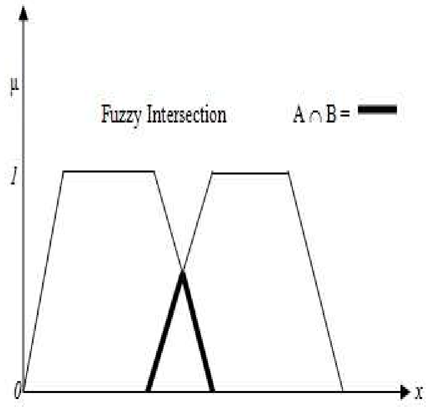

Figure 4(b) Fuzzy Intersection

2.2.5 THE GEFRED MODEL

The GEFRED (Generalized Fuzzy Relational Database) model was published in 1994 by Medina-Pons-Vila [11]. It is developed in possibility framework, so fuzzy domains are considered. It also includes the case where the underlying domain is not fuzzy i.e. numeric. Various data types given in GEFRED are:

1. A single scalar (e.g., Age = Young, represented by the possibility distribution 1/Young).
2. A single number (e.g., Height = 160, represented by the possibility distribution 1/160).
3. A set of mutually exclusive possible scalars (e.g., Age = {Young, Old}, represented by {1/Young,
1/Old}).

4. A set of mutually exclusive possible numbers (e.g., Age = {14, 50}, represented by {1/14, 1/50}).
5. A possibility distribution in a scalar domain (e.g., Age = {0.6/Young, 1.0/Middle}).
6. A possibility distribution in a numeric domain (e.g., Age = {0.5/23, 1.0/26, 0.8/24}, fuzzy numbers or
linguistic labels). It includes Umano-Fukami models‟s datatypes UNKNOWN ,UNDEFINED and NULL also.
7. An Unknown value with possibility distribution: Unknown = {1/x : x ∈ D}
8. An Undefined value with possibility distribution:
Undefined = {0/x : x ∈ D}
9. A NULL value given by:
NULL = {1/Unknown,1/Undefined}

As GEFRED suggests, generalized fuzzy domain (D) and generalized fuzzy relations (□)) concepts are being included. Generalized fuzzy domain (D) includes classic domain, with the possibility distributions defined for this domain and the NULL value also. If X is the universe of discourse and P (X) is set of all possibility distributions including unknown, undefined types and Null type. Then, generalized fuzzy domain (D) can be represented as D □ P(X)□ □NULL.
Generalized fuzzy relations are relations whose attributes are having generalized fuzzy domain. Fuzzy attributes may have associated with compatibility attribute where compatible degree can be stored. Generalized fuzzy relations are given by two sets: Head H and Body B. The head includes the name of each one of the attributes, their domains, and their compatibility attributes (which are optional). The body includes the values of the m tuples:

$$R=\begin{cases} H= \{(A1:D1[,C1], ..., An:Dn[,Cn])\} \\ B = \{(A1:di1[,ci1], ..., An:din[,cin])\}\ i= 1,..., m \end{cases}$$

GEFRED model also defines fuzzy comparators which are based on the classic comparators (<, >, =etc). When these fuzzy comparators are used on non-fuzzy values, meanings of comparators does not change i.e. classical output should come during crisp values.

III. SQL WITH FUZZY COMPONENT

3.1 Concept Of GLC (Generalized logical condition)

The generalized logical condition (GLC) for the WHERE part of the SQL based on linguistic expressions is created. This GLC enables matching fuzzy and classical constraints in the same WHERE clause and to select only records that have the query satisfaction greater than zero. These records are transferred to the client side where t-norm and t-conorm functions, which can be easy aggregated to n variable case, are used to calculate query satisfaction index for each of these records. The query compatibility index (QCI) indicates how the selected record satisfies a query request. If the record fully satisfies query, the QCI value is 1 and if record partially satisfies query conditions, QCI value is in (0,1) interval and represents the distance to the full query satisfaction. The QCI value 0 means that the record does not satisfy a query. It is also possible to use additional filtering functions to choose appropriate number of records or to set the threshold value of the QCI.

3.2 Concept of Weak Coupling

Weak coupling approach with DBMS SQL Server is being worked upon. The concept of weak coupling is shown in Figure 6. The FRDBMS proposed respects the GEFERD model. The language of description and manipulation of the data is therefore FSQL. Seen that the FSQL language is an extension of the SQL language, a FRDBMS can model a RDB (described in SQL language) or a FRDB (described in FSQL language).

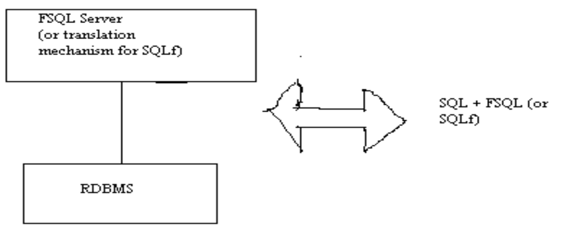

Figure 5. Concept of Weak Coupling

The principle of this coupling is the definition of a software layer that allows the transformation of the command written by the user in FSQL language in their equivalent written in SQL

3.3 STRUCTURE of SQL WITH FUZZY COMPONENT

Fuzzy queries have emerged in the last 20 years to deal with the necessity to soften the Boolean logic in relational databases. A fuzzy query system is an interface to human users to get information from database using (quasi) natural language sentences. Many fuzzy query implementations have been proposed, resulting in slightly different languages. Although there are some variations according to the particularities of different implementations, the answer to a fuzzy query sentence is a generally a list of records, ranked by the degree of matching [2].

In this research the goal is to change values P and R from query (1) with linguistic expressions and to calculate the lower bound of QCI from these linguistic expressions. Thus calculated lower bound is used as a parameter for database queries to select records that have QCI>0. In the next step appropriate t-norms or t-conorms are used to calculate QCI values for all retrieved records. Figure 7 shows steps and modules of this approach. This approach decreases the amount of transferred data across nets and calculation of QCIs is not significant burden for client computers

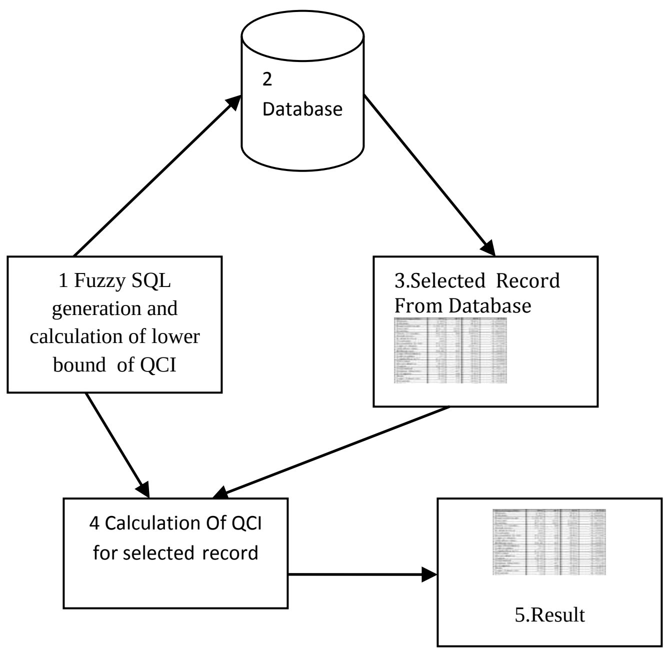

Figure 7. Structure of SQL with fuzzy component

IV. CONCLUSION

This paper outlined the brief introduction of fuzzy relational models. This paper also discusses imperfect information. Fuzzy set theory basics are also being reviewed in brief. Use of fuzzy Logic provides the solution that allows the creation of Queries based on linguistic expression from the users point of view and does not change the structure and concept of obtaining data from relational database, enables an improved usage of Slither SQL and fuzzy approach creates a simple and easy way to use data mining tool. Further search could be directed towards improving the fuzzy SQL by developing the web application with a fuzzy module for data dissemination.

REFERENCES

[1]. Z.M. Ma, A conceptual design methodology for fuzzy relational databases .Journal of Database Management , Vol. 16, No. 2 ,2005 ,66-83.

[2]. Bosc P. and Pivert O., "SQLf Query Functionality on Top of a Regular Relational Database Management," in Proceedings of Knowledge Management in Fuzzy Databases, Heidelberg, pp. 171-190, 2000.

[3]. Galindo, J., Urrutia, A., Piattini, M.: Fuzzy Databases: Modeling Design and Implementation. IDEA Group Publishing, Hershey,USA. (2006)

[4]. Hudec M., "Fuzzy improvement of the SQL", Balcor, 2007

[5]. Amel Grissa T. and Mohamed Hassine , "New Architecture of Fuzzy Database Management Systems," *The International Arab Journal of Information Technology, Vol. 6, No. 3, July 2009* 213

[6]. A.H.M. Sajedul Hoque,Md Sedek Ali et al," Performance Comparision Of Fuzzy Queries on Fuzzy Database and Classical Database," 5th *International Conference on Electrical and Computer Engineering,*ICECE 2008,20-22 Dec 2008

[7]. Galindo J., Urrutia A., and Piattini M., Fuzzy Databases: Modeling, Design and Implementation, Idea Group Publishing, Hershey, 2006

[8]. Galindo J., Urrutia A., and Piattini M., Representation of Fuzzy Knowledge in Relational Databases, IEEE Computer Society, 2004

[9] Dr. Rahul Rishi and Punkaj Gupta,"Database Design for Storage of Fuzzy Information inTraditional Datbase,"IJCA Journal,2011

[10] Handbook of Research on Fuzzy Information Processing in Databases by José Galindo,Volume 1

[11] Medina, J. M., Pons, O., & Vila, M. A., GEFRED: A generalized model of fuzzy relational databases. Information Sciences, 76 (1/2), 1994,87-109.