# Fuzzy Query: An Impression in Query processing

Ankita Srivastava[*1], Sonali Yadav[*2], Niharika Srivastava[#3], Zaina Khan[*4]

[*]*CSE Dept., Integral University, Lucknow, India*

[#]*SRMSCET, Lucknow, India*

***Abstract*: Fuzzy queries can be used to access and modify data in both conventional & fuzzy databases, through this paper, the benefits of fuzzy query over crisp query will be illustrated using a classical database, on which both crisp and fuzzy queries are executed so that the efficiency of fuzzy query may be shown. For some situations, fuzzy queries more efficiently represent how human interpret the reality through their perception and language. This paper gives the basic information about fuzzy queries and also discusses the work done by many researchers in the area of fuzzy query processing and points towards the opportunity of further research in this area.**

***Keywords*: Conventional database, fuzzy predicate, membership function, fuzzy database.**

## I. Introduction

Precise (Crisp) data refers to data that is accurate and is without any ambiguity. Crisp queries are intended to retrieve data elements which qualify for a given condition. Crisp queries are based on typical logic and set theory, restricting the expression of user preferences. It is possible to deal only that information which is completely true or completely false through classic logic. It is not possible to deal with information that is imprecise or vague, but this type of information contains data, which can provide a better solution to a given problem. In classic logic, if an element does not belong to a set, the membership of that element is represented by 0 and 1 if it belongs, having the set as {0, 1}. This is not appropriate for handling new applications and needs.

On the other hand, in fuzzy logic the membership of an element may be any value of interval [0, 1]. Fuzzy logic is an extension of the classic systems. Instead of exact reasoning, fuzzy logic is the logic behind approximate reasoning. Its importance lies in the fact that many types of human interpretation, particularly the way of thinking based on common sense, are by nature approximate. Fuzzy Logic was proposed by L.A. Zadeh in 1965. The basic idea behind fuzzy logic is sets or classes whose boundaries are vague. For example: old people, good students etc.

Queries having imprecise or vague terms are called fuzzy queries, for example: find the name of persons who are very old. Fuzzy queries are based on Fuzzy Logic i.e. a powerful tool for dealing with preferences. Fuzzy query helps users to express the requirements involving preferences. With this tool, the user may express their query using linguistic terms and sentences. As the application of database technology moves outside the area of a crisp mathematical world to the area of the real world, the significance of handling imprecise information increases.

## II. Literature Review

Human, when interacting with the database, want to make complex queries that have a lot of vagueness present in it. In real world applications we often need to test the queries based on fuzzy data. For example, some one can specify as find employee whose age is around 25 years old, find young person, find employee with good salary etc. A query is flexible if the databases contain imprecise and uncertain information or the query condition is imprecise and uncertain.

In [2] the application of fuzzy set theory in DBMS can be classified into two main classes.

Class 1 deals with the study of fuzzy query processing in conventional (non-fuzzy) DBMS

Class 2 deals with DBMS which, having the ability to store and manipulate fuzzy data directly and also supports fuzzy query.

### *A. Conventional DBMS with Fuzzy Queries:*

Early DBMS's with uncertainty handling are developed within the framework of non-fuzzy DBMS. Generally, these systems deal with the construction and evaluation of fuzzy query against a crisp Database, and ignore the problem of direct representation of fuzzy data in DBMS.

Consequently, this model is less powerful, as it is less suitable in handling fuzzy query involving fuzzy data which are vague and ambiguous.

### *B. Fuzzy Databases with Fuzzy Queries:*

Recent DBMS's with uncertainty handling are more advanced when compared to the earlier ones. They address the problem of direct representation of fuzzy data in the DBMS as well as the construction and evaluations of fuzzy query.

Various fuzzy database models have been proposed over last 35 years. We are mainly interested in query processing on relational data model and its extensions because at present relational database model is worldwide used tool for building database system. Buckles [3] propose one of the earliest versions of Fuzzy Relational Database System (FRDBS) by merging the theory of fuzzy set and Relational Database System (RDBS)

Many researchers develop many techniques for the processing of fuzzy queries. Here we will discuss some of the main techniques to process the fuzzy query.

1) Janusz Kacprzyk and Andrzej Ziolkowski [4] proposed a new approach for database querying system for handling imprecise queries using fuzzy-logic-based calculus of linguistically quantified propositions. An algorithm was proposed to deal with queries involving linguistic quantifiers and importance of the attributes. The meaning of imprecise term is elicited from user and then record of specified term are fetched and matched with query. The formula for finding the matching degree of different types of query was defined in the algorithm.
2) Leonid José and Tineo Rodríguez [5] provide the extension of relational database management systems which support fuzzy quantified queries so that users may address queries involving preferences. An interpretation of fuzzy quantifiers was presented that is based on a simple linguistic transformation principle. With this interpretation, it is possible to derive Boolean queries, which return a α-cut of the initial fuzzy query. Then, the fuzzy query can be processed on this set thus avoiding the exhaustive scan of the entire database.

3) Tien-Chin Wang et al. [6] proposed a fuzzy query language (FQL) which can be applied to query imprecise data. Fuzzy language architecture based on SQL and fuzzy sets for the evaluation of records was developed that significantly improve the robustness of database query operations. In this architecture first data is transform into some clusters based on membership degree then possibility value is calculated for comparing it with α threshold. There also exists WEIGHT clause in the query.

4) V.Balamurugan and K.Senthamarai Kannan [7] proposed a framework to handle the complexities by using fuzzy set theory. The framework is divided into two stages-preprocessing and query implementation stage. The preprocessing stage involves collecting information; identify fuzziness associated with attributes and form different fuzzy sets. In the query implementation stage presence of linguistic variables and hedges are checked and then membership values are calculated and manipulated based on hedges. Finally defuzzification is carried out by finding the α-cut. Main aim of this model is to make use of the standard facilities available in the conventional DBMS.

## III. Need Of Fuzzy Queries

Information in real-world applications is often imprecise, vague and uncertain, if this is ignored, may cause deformation of human perception of real-world and loss of important information. Mechanisms such as fuzzy query, are closer to natural language expression and human being thinking help in improving the access information. Recently new applications have come in to existence that require database management systems with uncertainty capabilities, for example decision support systems.

## IV. Rigidity Problem In Crisp Queries

Crisp queries are suffered from a lack of flexibility. Here by using a simple example we will show that in our daily life some situations may arrive when fuzzy queries provide more user friendly than crisp queries.

Suppose there is a relation HOTEL and user want to find the hotel *cheap* and *near* to station. In the case of crisp querying system user reformulate it as not more than 5 km away and price less than 500.

TABLE I:
Relation HOTEL

| Name | Distance(Km.) | Price(Rs.) |
|---|---|---|
| **A** | 4.5 | 500 |
| **B** | 5 | 500 |
| **C** | 1 | 520 |
| **D** | 5.1 | 350 |
| **E** | 6 | 650 |
| **F** | 7 | 600 |

SQL statement is written as SELECT * FROM HOTELS WHERE PRICE <=500 AND DISTANCE<=5;
The result of the above query will be:-

TABLE II

| Name | Distance(Km.) | Price(Rs.) |
|---|---|---|
| **A** | 4.5 | 500 |
| **B** | 5 | 500 |

Here we can see that although the other two options i.e. C, D are much better for the user but the system doesn't help him in finding the "best" answer. The problem of rigidity arrives from Boolean Logic. It could be solved allowing the specification of fuzzy queries.

Now we will solve this problem using fuzzy logic.
First we will define the membership function for fuzzy predicates: cheap and near respectively:

$$\mu_{cheap}(x) = \begin{cases} 1 & x \leq 400 \\ \dfrac{b-x}{b-a} & 400 < x < 600 \\ 0 & x \geq 600 \end{cases}$$

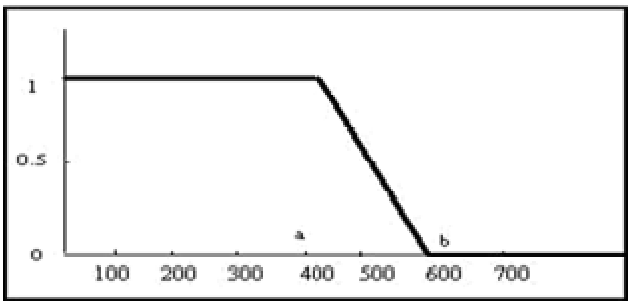

Figure 1: membership function for fuzzy predicate "cheap".

$$\mu_{cheap}(x) = \begin{cases} 1 & x \leq 400 \\ \dfrac{b-x}{b-a} & 400 < x < 600 \\ 0 & x \geq 600 \end{cases}$$

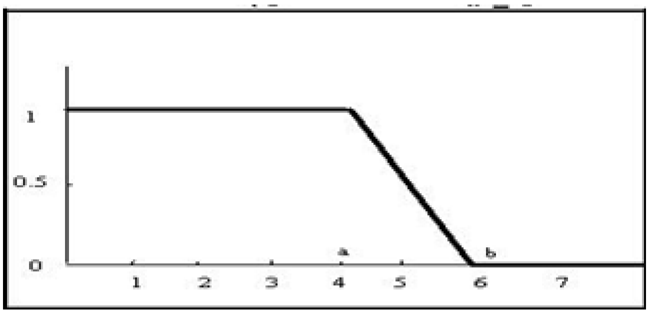

Figure 2: membership function for fuzzy predicate "near".

After calculating the membership value of each element to the fuzzy predicate near and cheap, for finding the solution we will apply the arithmetic mean [1] i.e. a type of operator allowing for compromises between the predicates used as parameters.

TABLE III

| Name | Distance (Km.) | Price (Rs.) | | | AM |
|---|---|---|---|---|---|
| **A** | 4.5 | 500 | 0.75 | 0.5 | 0.625 |
| **B** | 5 | 500 | 0.5 | 0.5 | 0.500 |
| **C** | 1 | 520 | 1.0 | 1.0 | 0.700 |
| **D** | 5.1 | 350 | 0.45 | .40 | 0.725 |
| **E** | 6 | 650 | 0.0 | 0.0 | 0.0 |
| **F** | 7 | 600 | 0.0 | 0.0 | 0.0 |

Arithmetic mean: AM(P1, . . . , Pn)(x) = (PI (x) + . . . + Pn(x))/n , Where P is a fuzzy predicate

Here

$$\text{AM}=\left(\mu_{cheap} + \mu_{near}\right)/2$$

Result of the above calculation shown in the table III

## V. Conclusion

Imprecise, uncertain or vague information is a common phenomenon in real world. Many authors have proposed framework for processing the fuzzy query on the conventional database system but these system do not allow storing the fuzzy data in the database. The available technology of conventional database system for efficient data access and query solving may not be used directly in fuzzy querying. So the Fuzzy databases are required in order to store fuzzy data and process the fuzzy queries efficiently. Based on the previous study we can say that processing of fuzzy query over fuzzy database and development of an efficient query evaluation mechanism is an open area of further research.

## References

[1] Patrick Bosc and Olivier Pivert, "Sqlf a relational database language for fuzzy querying," IEEE Transactions on Fuzzy Systems, Vol. 3, No. 1, pp. 1-17, February 1995.

[2] K. K. Phang, Mashkuri Hj. Yaacob, T. C. Ling "Development of fuzzy database systems," Malaysian Journal of Computer Science, Vol. 10 No. 1, pp. 42-46, June 1997.

[3] B.P. Buckles, F.E. Petry, and H.S. Sachar. "A domain calculus for fuzzy relational databases". Fuzzy Sets and Systems, 29:327–340, 1989.

[4] Janusz Kacprzyk and Andrzej Ziolkowski "Database queries with fuzzy linguistic quantifiers," IEEE Transactions on Systems, Man, and Cybernetics, Vol.Smc-16, No. 3, May/June 1986.

[5] Leonid Jose "Extending RDBMS for allowing fuzzy quantified queries," Lecture Notes in Computer Science, Vol. 1873, Mohamed Ibrahim-Josef Küng-Norman Revell (Eds.) Springer Verlag, (2000), Pp. 407-416.

[6] Tien-Chin Wang, Hsien-Da Lee and Chun-Ming Chen, "Intelligent queries based on fuzzy set theory and SQL," unpublished.

[7] V.Balamurugan and K.Senthamarai Kannan, "A framework for computing linguistic hedges in fuzzy queries," international journal of database management system vol 2 no.1 february 2010.

[8] Jaydev Mishra, "Fuzzy query processing," International Journal of Research and Reviews in Next Generation Networks Vol. 1, No. 1, March 2011.

[9] Galindo J., Urrutia A., Piattini M., "Fuzzy databases: modeling, design and implementation," published by Idea Group Publishing Hershey, USA, 2005.