Analysis and Usage of Fuzzy Logic for Optimized Evaluation of Database Queries

Sardar Sathpal Singh	Prof. Rishi Sayal	P. Venkat Rao
Computer Science & Engineering	Prof. & Head	Asst. Professor
Guru Nanak Engineering College	Computer Science & Engineering Dept.	Computer Science & Engineering Dept
Ibrahimpatnam, R.R. District, Andhra Pradesh.	Guru Nanak Engineering College, Ibrahimpatnam, R.R. Dist, Andhra Pradesh	Guru Nanak Engineering College, Ibrahimpatnam, R.R.Dist, Andhra Pradesh

ABSTRACT

The majority of existing information systems deals with crisp data through crisp database systems. Retrieval of specific queries through fuzzy logic systems have been a rare approach in research. Use of fuzzy logic rule based systems on data bases has shown efficient results experimentally. As such no early work has been done on combination of rule based systems with the database queries to give an efficient output. This paper brings such concise theme of evaluating a student's performance through such rule based systems. The aim of this paper is to present various fuzzification logic combined with rule based systems. Further experimentally this paper proves that for specific applications on databases using rule based systems can give much better results rather than using simple crisp queries with simple database management. This paper also gives a brief overview of various fuzzy logics, concepts fuzzification and defuzzification. This novel property of rule based system on any database system to evaluate for the performances is the main theme of this paper.

Keywords: Fuzzification, Defuzzification, Rule-Based System, Database, Fuzzy Sets.

1. INTRODUCTION

Fuzzy Logic was introduced in 1965 [1], [2], [3], by Lotfi A. Zadeh, professor for computer science at the University of California in Berkeley. Basically, Fuzzy Logic (FL) is a multivalued logic that allows intermediate values to be defined between conventional evaluations like true/false, yes/no, high/low, etc. Fuzzy Relational Databases (FRDB) is introduced in order to overcome the lack of ability of relational databases to model uncertain and incomplete data. The use of fuzzy sets and fuzzy logic to extend existing database models to include these possibilities has been utilized since the 1980s. In [1] and [13], authors mention one of the first approaches to integrate fuzzy logic with ER model. Their model allows fuzzy attributes in entities and relationships. Furthermore, the FRDB model was developed in [4, 5] i.e. a way to use fuzzy EER model to model the database and represent modeled fuzzy knowledge using relational database in detail was founded. Following these attempts, in [10, 11, 12] authors defined a new type of fuzzy SQL language based on the FRDB model developed specifically for this purpose. Formal development of fuzzy logic is a well worked area. Terms like rather high or very slow can be computed mathematically and processed by computers, in order to apply a more human-like way of thinking in the programming of computers. Expert systems, neural networks and fuzzy systems are some more ways of increasing machine intelligence, but, in most cases, the effectiveness of a solution heavily depends on the effectiveness of data access. Fuzzy logic [1, 2, 6, 7, 8] is useful both in AI and human reasoning as well as in decision making and is well suited in searching problems when we can define a set of search conditions, but we can accept a result which resembles, at some degree, the expressed conditions. A common DBMS is expressed to process exact queries, e. g. select * from T where name = "Rahul". You are often allowed to use queries with pattern matching schemes, but more than often such searching features are restricted to the simplest select * from T where name like "ra", meaning to retrieve all records with the field name beginning with the string "Rahul", such as "Raman" or "Ram". Moreover, if a query "fails" to fetch any data and display the output, since there are no records with such word in the data base with the specified attributes, there is no way of making user defined query to retrieve data from Data bases. Therefore, using a "Conventional" DBMS, it is not so possible to express similarity queries. The purpose of Fuzzy Base is to obtain results when the traditional query fails. The process is carried on until the new query retrieves a non-empty set of records in the data base: this set is the most similar to the one originally searched by the user. While a fuzzy similarity query matches required data with grade lower than one: the closer the similarity grade is to one, the greater the retrieved data are like the ones searched by the user [9]. The precision of mathematics owes its success in large part to the efforts of Aristotle and the philosophers who preceded him. In their efforts to devise a concise theory of logic, and later mathematics, the so-called "Laws of Thought" were posited [5]. But it was Lukasiewicz who first proposed a systematic alternative to the bi–valued logic of Aristotle [6]. Even in the present time some Greeks are still outstanding examples for fussiness and fuzziness, (note: the connection to logic got lost somewhere during the last 2 millenniums [7]).

2. CLASSIFICATION OF FUZZY SET THEORY WITH RESPECT TO DBMS

The application of fuzzy set theory in DBMS can be classified into two main categories. Category 1 concerns the study of fuzzy query processing in conventional (non fuzzy) DBMS; Category 2 deals with DBMS which, besides having the ability

to store and manipulate fuzzy data directly, also supports fuzzy query.

2.1 Category 1: Traditional DBMS with Fuzzy Queries

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. This section gives an overview of the direction and problems that researcher in this area attempt to address. Chang et al. [14] explore the use of fuzzy query and propose the Database Skeleton concept which allows user to specify the contents and meaning of a collection of data. Database Skeleton is later used as a semantic base which supports fuzzy query. In Chang's paper, the term "fuzzy" refers to incompletely specified information in the query, such as the lack of access path. It should be noted that fuzzy set theory is not used explicitly in formulating the methodology. Wong [15] proposes a framework to handle incomplete information in non-fuzzy DBMS. Sometimes, due to incomplete or imprecise information in the DBMS, a Database cannot provide answers to some queries. Tahani [16] develops a high-level conceptual framework for processing fuzzy query in a conventional non-fuzzy Database environment. Kacprzyk et al. [17, 18] present a fuzzy query system called Fquery III. Through Fquery III, Dbase III plus (a commercial non-fuzzy micro computer-based RDBS) data can be operated on using fuzzy query. Fquery III is based on the framework of fuzzy set theory. Wong et al. [19] develop a fuzzy query language for VAX Rdb/VMS (a conventional non-fuzzy mini computer-based DBMS). Bosc et al. [20] discuss the extension of the SQL language to handle fuzzy query based on the framework of fuzzy set theory. The extended SQL has the following format: Select n/t <attribute> from <relation> where <fuzzy condition> where n and t are output regulating parameters. As a result, fuzzy query processing tends to become more efficient. In general, under the conventional non-fuzzy DBMS environment, fuzzy query processing based on the theory of fuzzy set is more powerful when compared to those using ad hoc approach or probability theory.

2.2 Category 2: Fuzzy Databases with Fuzzy Queries

DBMS's with uncertainty handling have become a problem to address in the recent past. Now they are more advanced when compared to the earlier ones. They solve the problem of representing the fuzzy data directly in the DBMS as well as the framing of fuzzy query. Buckles proposed the version of Fuzzy Relational Database System (FRDBS) by clubbing the theory of Relational DataBase System (RDBS) along with fuzzy set.

The above version is one of the earliest and brings out a strong theoretical framework of similarity-based FRDBS.

It only supports a specific class of fuzzy number but does not support Possibility Distribution data type. Shenoi generalize the similarity-based model. They observe that the preservation of the properties of classical RDBS above can also be done by restricting the components of fuzzy tuples to be nonempty subsets of equivalence classes from domain partitions. Since the notion of equivalence classes is more general than the notion of similarity relation, an equivalence model of FRDBS, which is a generalization of similarity based model, has been proposed.

3. FUZZY SET

Generally, in our day-to-day life problems are solved using binary paradigm. There are certain problems where we need to consider a vast set of options instead of the only possible two: "true" or "false". The basic idea of fuzzy set theory is that elements or entities can be assigned to sets of varying degrees. That is, instead of either including an element in a given set or excluding it from the set, a membership function is used to express the degree to which the element is a member of the set. For instance to express that a 'laptop' is new using only two-valued logic, it is necessary to decide on an exact limit on age. For example, to distinguish "new laptop" from "not new laptop", one may use the criteria that laptops developed within the last 3 years are new, and laptops older than that are not new. But the allay of this concept is that it classifies a laptop "not new" even if the laptop is developed hours after the 3 years criteria mentioned earlier. The laptops can be better classified if one uses more classes, namely, new, relatively, somewhat old, very old etc. (more than two values or states) and this is not possible with the classical two-valued logic. But one can use fuzzy logic to achieve this better classification. One could decide the simple function, to express that, laptops developed within latest two years are new, laptops developed within two years and four years ago are new to some degree, and laptops older than that, are not new. The fuzzy set approach can be used to classify the documents into fuzzy affinity classes and also to control the actual retrieval process. For example, consider first a document Dc and a particular term X. If X denotes the "concept class" of all items dealing with the subject denoted by X, then the membership function of documents Dc in set X may be denoted as fX (Dc). In the usual terminology fX (Dc) represents the weight of term X in document Dc. Given a number of concept classes A, B, C, ………, Z representing various subject areas, it is now possible to identify each document by giving its membership function with respect to each of the concept classes, that is,

$$D = (fA(Dc), fB(Dc), \ldots\ldots\ldots, fZ(Dc))$$

In general, the distance (or similarity) between two documents or between the document and a query may be obtained as a function of the differences in the membership functions of two items in the corresponding concept classes. Specifically, given T different concept classes, the fuzzy distance between documents Dc' and Dc'' might be computed as

$$T \, d \, (Dc', Dc'') = | \, ((fx \, (Dc') - fX \, (Dc''))| \, x= 1$$

Ranked retrieval is achieved by retrieving the documents in order of increasing fuzzy distance from the query. One attractive feature of fuzzy set theory is the possibility of extending the definition of the membership function from single terms to combination of terms. Thus, given the membership functions of document Dc with respect to terms A and B, the following rules apply for Boolean combinations of terms.

$$F \, (A \, AND \, B) \, (Dc) = min \, (fA \, (Dc), fB \, (Dc))$$

$$F \, (A \, OR \, B) \, (Dc) = max \, (fA \, (Dc), fB \, (Dc))$$

$$F \, (NOT \, A) \, (Dc) = 1 - fA \, (Dc)$$

Definition: Given a pair of standard sets B and M, a fuzzy set F based on B is a pair (B, f) where f: B ->M. B is the "base set" or "support", M is the "membership space" and f is the

"membership function" mapping any element of the support in the corresponding membership value. It is possible to generalize the basic definition of fuzzy set with a recursive use of fuzzy sets in the definition of the membership space [8].

4. FUZZIFICATION

For each input and output variable selected, we define two or more membership functions (MF), normally three but can be more. We have to define a qualitative category for each one of them, for example: low, normal or high. The shape of these functions can be diverse but we will usually work with triangles and trapezoids (actually usually pseudo-trapezoids) (see Figure 6). For this reason we need at least three (for triangles) or four (for trapezoids) points to define one MF of one variable.

Example 1: If we take **x** like a variable and low, normal and high as trapezoidal, triangle and trapezoidal MFs, respectively (Figure 6), - the MF **low** will be defined by three points: (x1, x2, and x3). However, in order to have a real trapezoid, we need a four point at the left of x1 (**any** negative one, e.g. x0) - following the same reasoning, the MF **high** will be defined by four points: (x3, x4, x5, x6) (x6 **any** positive > x5, being x5 the higher possible value for x) - finally, the MF **normal** (like any other triangular MF) will be defined by three points: (x2, x3, x4). In case the MF are trapezoids (or pseudo-trapezoids) (in this case 'low' and 'high'), the MF can be defined as: In case the MF are triangles (in this case 'normal'), the MF can be defined as:

$$y^{low}_{(trap)}(x;x_0,x_1,x_2,x_3) = \max\left(\min\left(\frac{x-x_0}{x_1-x_0},1,\frac{x_3-x}{x_3-x_2}\right),0\right)$$

$$y^{high}_{(trap)}(x;x_3,x_4,x_5,x_6) = \max\left(\min\left(\frac{x-x_3}{x_4-x_3},1,\frac{x_6-x}{x_6-x_5}\right),0\right)$$

In case the MF are triangles (in this case 'normal'), the MF can be defined as:

$$y^{normal}_{(tri)}(x;x_2,x_3,x_4) = \max\left(\min\left(\frac{x-x_2}{x_3-x_2},\frac{x_4-x}{x_4-x_3}\right),0\right)$$

It is important to emphasize that the computation of all the functions/equations for all the MFs of all variables has to be done every time the shape and interval of the MFs are changed (on the contrary, once computed for the first time, these computations do not have to be done if MFs are not changed). *How the fuzzification step works* Next question to be solved is how to fuzzificate all the real values of the variable **x**. First, for a given value of **x,** for example **x$_n$** which can belong to one or more MF we calculate the **y** value for each of the MF/s which **x$_n$** belong**.** This **y** value has to be between **0** and **1**. For example: Consider three MF: low, normal and high and a given value of **x$_n$**, then the degrees of membership to each MF (y values) for **xn** can be, for example: 0.6 for the MF low and 0.4 for the MF normal (see Figure 6). Likewise, we can fuzzificate all the values of any variable. Any of the values will belong to at least one MF with a certain degree of membership.

Figure 6. Example for three MF for a given input
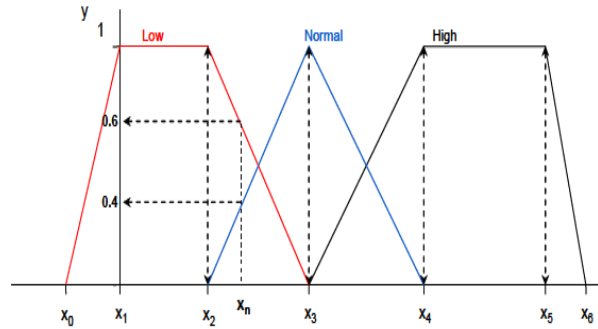
5. DEFUZZIFICATION

In many situations, for a system whose output is fuzzy, it is easier to take a crisp decision if the output is represented as a single scalar quantity. This conversion of a fuzzy set to single crisp value is called defuzzification and is the reverse process of fuzzification. Several methods are available in the literature (Hellendoorn and Thomas, 1993) of which we illustrate a few of the widely used methods, namely centroid method, center of sums, and mean of maxima.

5.1 Centroid method

Also known as the center of gravity or the center of area method, it obtains the center of area (x *) occupied by the fuzzy set. It is given by the expression.

$$X* = \frac{\int \mu(x)x\,dx}{\int \mu(x)\,dx}$$

For a continuous membership function, and

$$X* = \frac{\sum x_i . \mu(x_i)}{\sum \mu(x_i)}$$

For a discrete membership function.

Here, n represents the number of elements in the sample, xi's are the elements, and μ (xi) is its membership function.

5.2 Center of Sums (COS) method

In the centroid method, the overlapping area is counted once where as in center of sums, the overlapping area is counted twice. COS builds the resultant membership function by taking the algebraic sum of outputs from each of the contributing fuzzy sets A1 , A2 , …. , etc. The defuzzified value x* is given by.

$$X* = \frac{\sum x_i . \sum \mu A_k(x_i)}{\sum \sum \mu A_k(x_i)}$$

Here n is the number of fuzzy sets and N the number of fuzzy variables. COS is actually the most commonly used dufuzzification method. It can be implemented easily and leads to rather fast inference cycles.

5.3 Mean of maxima (MOM) defuzzification

One simple way of defuzzifying the output is to take the crisp value with the highest degree of membership. In cases with more than one element having the maximum value, the mean value os the maxima is taken. The equation of the defuzziffied value x*is given by

$$X *= \frac{\sum xi}{|M|}$$

Where M = {xi | μ (xi) is equal to the height of fuzzy set}

|M| is the cardinality of the set M. In the continuous case, M could be defined as

M = {x Є [-c, c] | μ(x) is equal to the height of the fuzzy set}

In such a case, the mean of maxima is the arithmetic average of mean values of all intervals contained in M including zero length intervals. The height of a fuzzy set A, i.e. h (A) is the largest membership grade obtained by any element in the set.

6. FUZZY RULE BASED EVALUATION

Any Fuzzy model application can be built up based on following three phases as per our database parameters consisting of marks and attendance.

Phase 1: Fuzzification of students marks, students attendance giving a final output value in terms of fuzziness.

Phase 2: Generation of inference rules on student's marks and attendance.

Phase 3: Process of Defuzzification of final output value.

Phase1: Every student's performance and final output value is based on two database attribute values that are his overall marks and attendance which are the input parameters to our fuzzy logic system (Figure 1).

Figure 1
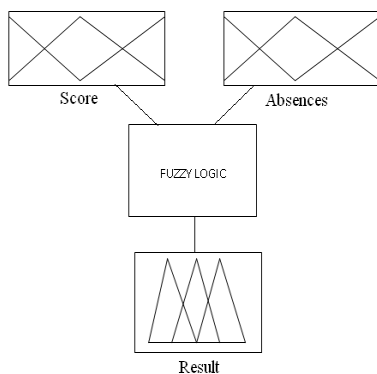
As per the database more than two attributes can also be used to act as input to the fuzzy logic system. For our experimental results we have used a student's database comprising marks and attendance only. More the number of database parameters, greater the number of inference rules generated with the system becoming more complex.

The membership function of student's marks and attendance are shown in Table 1 & figure 2. Generating triangular membership functions.

Table 1.

Linguistic Variables	Key	Interval
Least	L	(0,0,25)
Min	Mn	(0,25,50)
Mid	Md	(25,50,75)
Max	Mx	(50,75,100)
Extreme	Ex	(75,100,100)

Figure 2.
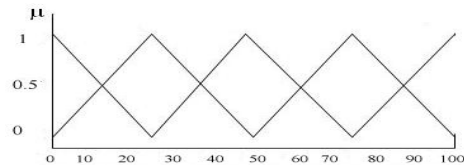
Table 2.

Linguistic Variables	Key	Interval
Very Poor	VP	(0,0,0.25)
Poor	P	(0,0.25,0.5)
Satisfactory	S	(0.25,0.5,0.75)
Good	G	(0.5,0.75,1)
Excellent	E	(0.75,1,1)

Figure 3.
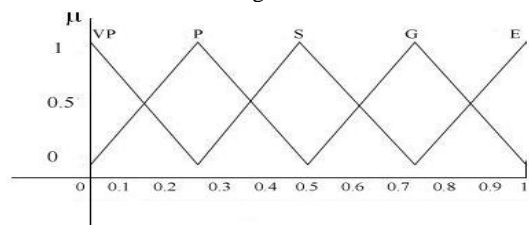
Phase2:

Fuzzy linguistic descriptions are a way of formal representations through rules popularly known as fuzzy rules. These fuzzy rules determine the input and output membership function using IF….THEN rules.(Altrock,1995: Semerci,2004).

These If….Then rules are coded in the form

IF (a_1 is $\tilde{A}_1$, a_2 is $\tilde{A}_2$,......, a_n is $\tilde{A}_n$) THEN (b_1 is $\tilde{B}_1$, b_2 is $\tilde{B}_2$,......, b_n is $\tilde{B}_n$)

where linguistic variables a_i, b_i take the values of fuzzy sets A_i and A_i respectively.

Example:

If there are clouds and thunder storm
Then there will be heavy rain.

As per our student application of marks and attendance, following rules are generated:

Rules of Inference when marks are Least:

L1: If Marks: L and Absences: L then Outcome: VP
L2: If Marks: L and Absences is Mn then Outcome: VP
L3: If Marks: L and Absences: Md then Outcome: P
L4: If Marks: L and Absences: Mx then Outcome: P
L5: If Marks: L and Absences: Ex then Outcome: S

Rules of Inference when marks are Min:

Mn1: If Marks: Mn and Absences: L then Outcome: VP
Mn2: If Marks: Mn and Absences: Mn then Outcome: P
Mn3: If Marks: Mn and Absences: Md then Outcome: P
Mn4: If Marks: Mn and Absences: Mx then Outcome: S
Mn5: If Marks: Mn and Absences: Ex then Outcome: S

Rules of Inference when marks are Mid:

Md1: If Marks: Md and Absences: L then Outcome: P
Md2: If Marks: Md and Absences: Mn then Outcome: P
Md3: If Marks: Md and Absences: Md then Outcome: S
Md4: If Marks: Md and Absences: Mx then Outcome: G
Md5: If Marks: Md and Absences: Ex then Outcome: G

Rules of Inference when marks are Max:

Mx1: If Marks: Mx and Absences: L then Outcome: P
Mx2: If Marks: Mx and Absences: Mn then Outcome: S
Mx3: If Marks: Mx and Absences: Md then Outcome: G
Mx4: If Marks: Mx and Absences: Mx then Outcome: G
Mx5: If Marks: Mx and Absences: Ex then Outcome: E

Rules of Inference when marks are Extreme:

Ex1: If Marks: Ex and Absences: L then Outcome: S
Ex2: If Marks: Ex and Absences: Mn then Outcome: G
Ex3: If Marks: Ex and Absences: Md then Outcome: G
Ex4: If Marks: Ex and Absences: Mx then Outcome: E
Ex5: If Marks: Ex and Absences: Ex then Outcome: E

Incase when more than one rule being active for the same output membership function, only one membership value is taken. Such process where decision plays an important role of choosing the right membership value is known as fuzzy decisions. Our application uses the decision making method proposed by Mamdami.

If the final output S is to be drawn is from the conjunction (AND) of all individual consequents Si then

$$S = S1 \cap S2 \cap \text{........} \cap Sn$$

where

$$\mu_{c(x)} = \min(\mu_{c1(x)}, \mu_{c2(x)},\text{.......} \mu_{cn(x)}), \forall x \in X$$

If the final output S has to be drawn from disjunction (OR) of individual consequents of each rule then

$$S = S1 \cup S2 \cap \cup \text{........} \cap \cup Sn$$

where

$$\mu_{c(x)} = \max(\mu_{c1(x)}, \mu_{c2(x)},\text{.......} \mu_{cn(x)}), \forall x \in X$$

Phase 3:

Defuzzification :

After completion of the fuzzy decision process the fuzzy value should be converted to a single scalar quantity. This conversion of fuzzy set to a single crisp value is called defuzzification. In our database application we have a centroid method (center of Area) which is shown in the figure 4. The areas have been partitioned with dotted lines and one instance of an area calculation is mentioned in the table 3. The final crisp value is calculated as below figure,equation (1).

Figure 4
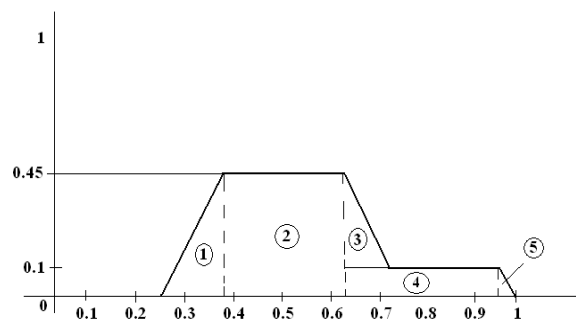
$$X^* = \frac{\sum A\bar{x}}{\sum A}$$

$$\sum A = 0.1975 \ \& \ \sum A\,\bar{x} = 0.098 \quad (1)$$

$$X^* = 0.098/0.1975$$

$$= 0.496$$

Experiments Results:

Our paper has used a database of 10 students marks with their attendance as shown in the datasheet.

Table 4.

Roll No.	Name	Marks (%)	Attendance (%)
07B91A0589	Rahul Dawda	78	80
07B91A0566	K Sindhuja	53	64
07B91A0596	S Gurnam Singh	70	82
07B91A0523	Deepti Kaur	15	25
07B91A0577	Vinay Kanth reddy	48	63
07B91A0507	Ch. Anusha	35	58
07B91A0582	Pargat Singh	69	60
07B91A05C7	Vishnu Lal	24	50
07B91A05A6	Sathpal Singh	97	92
07B91A05A5	Harpreet Singh	64	53

Each students final outcome is based on his marks and attendance only. The marks and attendance were fuzzified using our rules of inferences as shown in phase 2 and then the active membership functions were generated using mamdami fuzzy decision technique. The overall outcome of phase 2 was then defuzzified by calculating the centroid of the partitioned areas of geometrical figures. This process was repeated for all 10 students.

A student with final outcome value of (53 & 64) and (64 & 53) would not change the overall calculated value (Table 5).

Table 5.

Area Segment No.	Area (A)	$\bar{x}$	A$\bar{x}$
1	$\frac{1}{2} \times 0.11 \times 0.45 = 0.02$	0.32	0.006
2	$0.28 \times 0.45 = 0.126$	0.5	0.063
3	$\frac{1}{2} \times 0.08 \times 0.35 = 0.014$	0.6	0.008
4	$0.36 \times 0.1 = 0.036$	0.8	0.02
5	$\frac{1}{2} \times 0.03 \times 0.1 = 0.0015$	0.98	0.001

Our membership function for attendance ranges from 0 to 25, 25 to 50, 50 to 75 & 75 to 100. Let us vary the range of the attendance (Figure 5), but the criteria for the marks remain the same. By the change in arrangement of the value range of attendance the Mid membership function gets closer. The top value of Min remains same as 25 but the value range of L is moved to 45. Also the top value of Max remains same as 75 but the value range of Ex is moved to 65.

Figure 5.
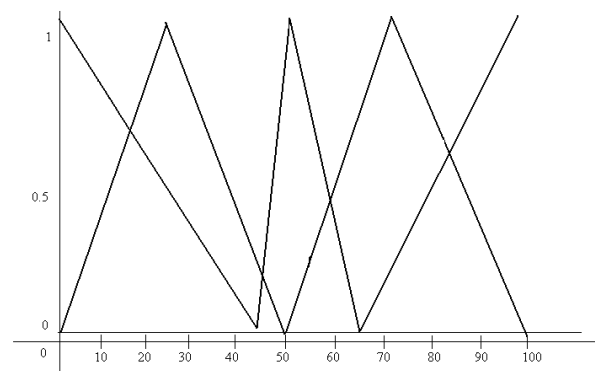
From Table 6 you can observe that the final outcome value is dropped for the students having attendance below 45 and is increased for attendance value above 65. This change in the

value range of attendance is to inflict the punishment on attendance below 45 and compensate attendance above 65. From phase 2 the active rules for the marks and attendance of the student "Sindhuja" are Md3, Md4 & Mx3, Mx4. The final outcome value is obtained as 0.511.

Table 6.

Marks	Absences	Performance Value
78	80	0.841
53	64	0.511
70	82	0.754
15	25	0.192
48	63	0.5283
35	58	0.3972
69	60	0.498
24	50	0.3541
97	92	0.9431
64	53	0.4972

Marks	Attendance	Performance Value
78	80	0.743
53	64	0.496
70	82	0.6817
15	25	0.217
48	63	0.5902
35	58	0.401
69	60	0.524
24	50	0.381
97	92	0.911
64	53	0.496

This shows that a fuzzy rules applied to a database will always lead to an efficient outcome combined with the right decisions which alone may not possible with database queries. An instance of a database queries which can be sampled with our application is

Query: get the student's Final outcome value whose marks are max and attendance is least.

Such database queries when applied to fuzzy set may lead to better decision processes as in spite of student's attendance being the least his marks are max is he really gaining anything from the classes or he is a student of high IQ.

7. CONCLUSION AND FUTURE WORK

In this paper the concept of rule based system has been applied on databases for evaluation and analysis. An application of student based system has been taken into consideration experimentally and the results have shown that considering various parameters and attributes of any entity, the performance can be optimally seen that results are efficient and better with such systems rather than using simple relational queries on databases.

Using rule based systems on a specific application has been experimentally proved in this paper to be more efficient but how far this concept is reasonable and realistic on data sets with varying theme of application systems. Comparative study of experimentally evaluating other techniques to evaluate performances needs to be empirically studied.

8. ACKNOWLEDGMENTS

9. REFERENCES

[1] L.A. Zadeh, Fuzzy Sets, Information and Control, 1965

[2] L.A. Zadeh, Outline of a New Approach to the Analysis of of Complex Systems and Decision Processes, 1973

[3] L.A. Zadeh,"Fuzzy algorithms," Info. & Ctl. Vol. 12, 1968, pp. 94-102.

[4] L.A. Zadeh,"Making computers think like people," IEEE. Spectrum, 8/1984, pp. 26-32.

[5] S. Korner,"Laws of thought," Encyclopedia of Philosophy, Vol. 4, MacMillan, NY: 1967, pp. 414-417.

[6] C. Lejewski,"Jan Lukasiewicz," Encyclopedia of Philosophy, Vol. 5, MacMillan, NY: 1967, pp.104-107.

[7] A. Reigber,"My life with Kostas", unpublished report, Neverending Story Press, 1999

[8] J.F. Baldwin,"Fuzzy logic and fuzzy reasoning," in Fuzzy Reasoning and Its Applications, E.H.Mamdani and B.R. Gaines (eds.), London: Academic Press, 1981.

[9] W. Bandler and L.J. Kohout, "Semantics of implication operators and fuzzy relational products," in Fuzzy Reasoning and Its Applications, E.H. Mamdani and B.R. Gaines (eds.), London: AcademicPress, 1981.

[10] M. Eschbach and J. Cunnyngham, "The logic of fuzzy Bayesian influence," paper presented at the International Fuzzy Systems Association Symposium of Fuzzy information Processing in Artificial Intelligence and Operational Research, Cambridge, England: 1984.

[11] F. Esragh and E.H. Mamdani, "A general approach to linguistic approximation," in Fuzzy Reasoning and Its Applications, E.H. Mamdani and B.R. Gaines (eds.), London: Academic Press, 1981.

[12] J. Fox, "Towards a reconciliation of fuzzy logic and standard logic," Int. Jrnl. of Man-Mach. Stud., Vol. 15, 1981, pp. 213-220.

[13] S. Haack, "Do we need fuzzy logic?" Int. Jrnl. of Man-Mach. Stud., Vol. 11, 1979, pp.437-445. Proceedings of a Symposium Organized by the Austrian Society for Cybernetic Studies, Hemisphere Publ. Co., NY: 1982.

[14] Shi-Kuo Chang and Wu-Haung Cheng. "DataBase Skeleton and Its Application to Logical DataBase Synthesis", *IEEE Transaction on Software Engineering*, Vol. SE-4, No. 1, January 1978.

[15] Eugene Wong. "A Statistical Approach to Incomplete Information in DataBase Systems", *ACM Transactions on DataBase Systems*, Vol. 7, No. 3, September 1982, pp. 470-488.

[16] Valiollah Tahani. "A Conceptual Framework for Fuzzy Query Processing - A Step toward very Intelligent DataBase Systems", *Information Processing & Management*, Vol. 13, 1977.

[17] Janusz Kacprzyk and A. Ziolkowski. "DataBase Queries with Fuzzy Linguistic Quantifiers", *IEEE Transactions on Systems, Man, and Cybernetics*, Vol. SMC-16, No. 3, May/June 1986.

[18] Janusz Kacprzyk et al. "FQUERY lll+: A 'Human-Consistent' DataBase Querying System Based on Fuzzy Logic with Linguistic Quantifiers", *Information Systems*, Vol. 14, No. 6, 1989.

[19] M. H. Wong and K. S. Leung. "A Fuzzy DataBase- Query Language", *Information Systems*, Vol. 13, No. 5, 1990.

[20] P. Bosc et al. "Fuzzy Querying with SQL: Extensions and Implementation Aspect", *Fuzzy Sets and Systems*, Vol. 28, 1988, pp. 333-349.