International Journal of Advance Research in Science and Engineering
Vol. No.5, Issue No. 02, February 2016
www.ijarse.com
IJARSE
ISSN 2319 - 8354

LITERATURE SURVEY AN APPROACH TOWARDS FUZZY QUERY TO SUBSTITUTE SKYLINE QUERIES

ShailjaChauhan[1] ,DrAjay Agarwal[2]

[1]Mtech Student, [2]Professor, Head IQAC and Co-Chair IEDC,

Department of Computer Science and Engineering , SRM University (India)

ABSTRACT

The fuzzy logic is basically an approach to compute information based on "degrees of truth", instead of the usual "true or false" logic. It deals with approximate reasoning, rather than the precise ones to solve problems in a way that more resembles human logic. The approach is to do the semantics rather than the mechanics and thus, instead of using precise expressions, the approach uses membership predicates to select the items from database. Major advantage is that it is not only a querying tool, but also it improves the meaning of a query. There are various fuzzy query processing techniques ranging from cluster analysis to iterative range search based algorithms but most of themare focused on implementation rather then the complexity.

This research focuses on the Skyline queries using relationaldatabase . It is found that the complexity of Skyline Query Processing is still significantly high which is comparable to nested queries.

The research proposes an approach towards fuzzy query using relational database to reduce complexity of skyline queries in order to optimize performance.

Keywords: Fuzzy Queries, Skyline, Pareto Dominance property, Weighted product of membership compatibility method, SFS algorithm.

I.INTRODUCTION

As we all know that the preference queries always have the special attention in the field of database and thus many extentionsof SQL have been introduced like SQLf and Skyline.

SQL evolved methods of creating, accessing, and manipulating relational databases. In this, user tells the computer what columns they want, which tables are involved, how the tables should be joined, and which rows from the tables will participate in query.

It also provides a clear method of performing joins among all specified tables. SQL is a relational calculusapproach : it selects and organizes sets of records through a high-level language that tells the system whatrecords to select, not howto select. It is basically a tool used for expressing the desired result of a database retrieval, instead of the mechanics necessary to achieve this result. Expressing the mechanics of a query is a relational algebraprocess.

Also in the other hand, we believe that statistical analysis is less precise than calculus or numbers are more precise than words. We judge our weather forecasts as highly imprecise and less accurate because they cannot predict the exact weather[1]. In fact, it can be shown thata decrease in precision gives an unexpected increase in accuracy.

1.1 Fuzzy Queries

A fuzzy SQL request gives a new path of retrieving useful data from relational databases. These concepts are sometimes imprecise. For example, we can look for risky projects in the corporate project database using a query such as :-

select *

from projects

whereprojects.budget is High

andprojects.duration is Short

With fuzzy SQL, terms such as High replace the more restrictive method of specifying a range of budget values, such as the following.

project.budget>= 30 and project.budget =< 70

The idea behind a fuzzy query process is simply this: “you tells the computer what you wants to, in terms that mean something to your way of thinking or expectations”. By expressing your thoughts through semantics rather than arithmetic, Boolean, and comparison operators, we can find information that expresses the intent of your query.

1.2 Skyline Operator

A Skyline is defined as those points which are not dominated by any other point. Also, if a point dominates another point than it is good or better in all dimensions and/or better in at least one dimension.

In Skyline, the concept of multicriteria must be satisfied simultaneously, in order to obtain the best rows[1].

Multicritariaanalyais is a common approach to address the needs of decision making applications where the set of dimensions and alternatives are finite. For example,A car shopper considers the price, make, model and milage of the car, as well as the vehicles currently in stoks. The analysis identifies the best, most preferred alternatives, which are obtained by eliminating those that are dominated by other alternatives.

II. LITERATURE SURVEY

This section provides an overview of the techniquesthat has been applied till now for evaluating the skyline queries:

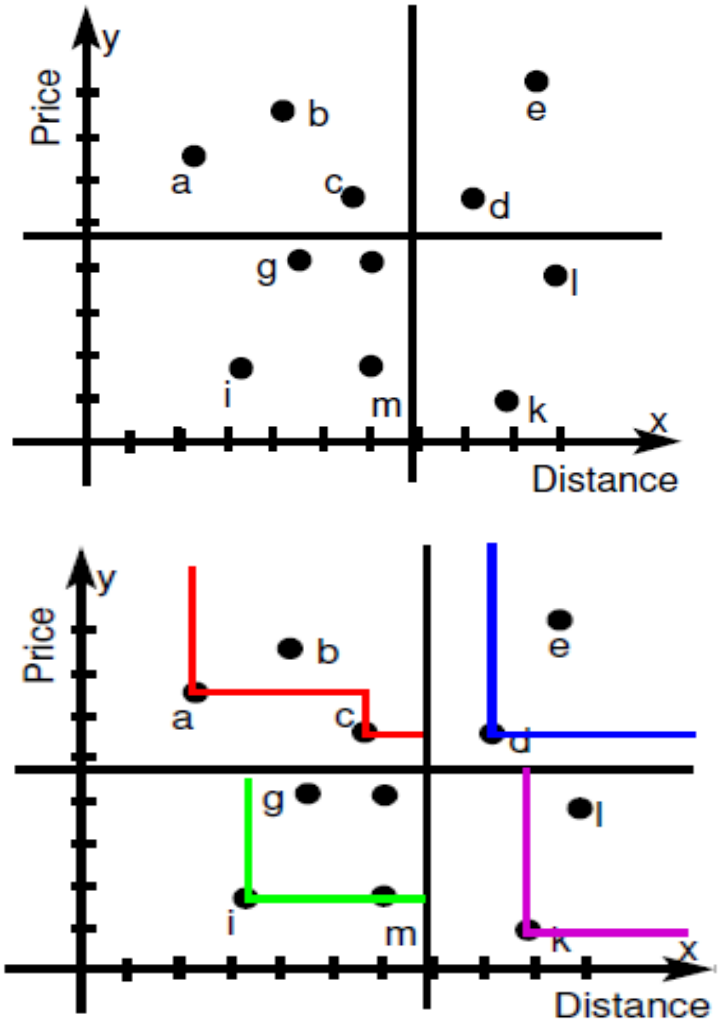
I. Block Nested Loop

In this technique,scanning is done through a list of points and test each point for dominance criteria and than the active list of potential skyline points seen thus far are maintainedalso eachvisited point is compared with all elements in the list. The list is suitablyupdated. This method does not require a precomputed index. Execution is independent on the dimensionality of the space and the total work done depends on the order in which points

were encountered. The method performs redundant work and there is no provision placed for the early termination.

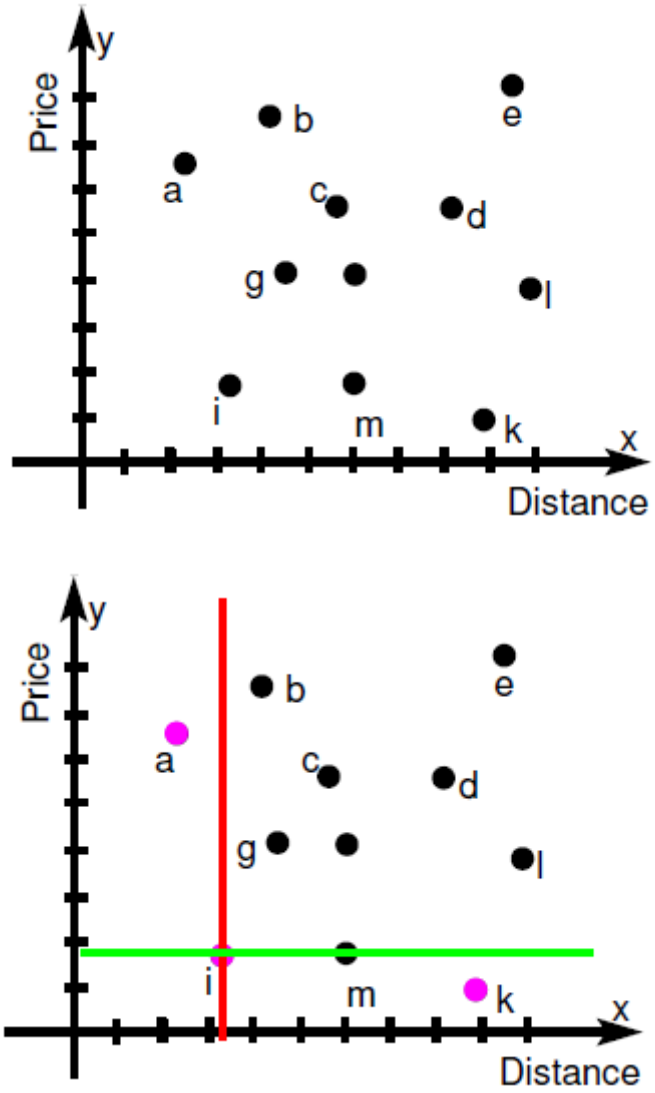
II. Divide-and-Conquer

In this technique firstly we recursively break up large datasets into smaller partition and continue till each smaller partition of the dataset in the main memory remains.After that we compute the partial skyline for each partition using any in-memoryapproach and later combine these partial skyline points to form the final skyline query, as shown in the following



III.Plane-sweep

The Plane sweep is done along each of the d dimensions and can detect the early termination.

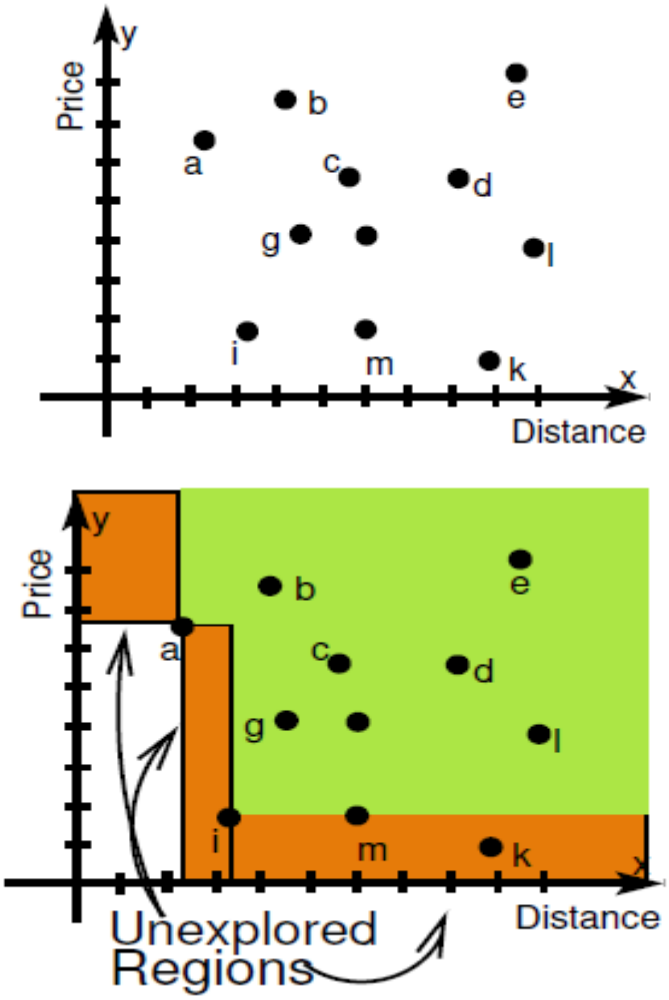


IV. Nearest Neighbor Search

In this, it is assumes that a spatial index structure on thedata points is available for use and thanidentiesthe skyline points by repeated applicationof a nearest neighbor search techniqueon the data points, using a suitably defined L_1distance norm.

The nearest neighbor in the data-point is I asit is closest to the origin when an L_1distancemeasure is assumed. Than after idivides the space into 2^n non-disjoint region, which now must not be recursivelysearched for more skyline points and recursively apply the search on region-1. The nearest neighbor in region-1 would be*a*, explode region to form additional regions.

International Journal of Advance Research in Science and Engineering
Vol. No.5, Issue No. 02, February 2016
www.ijarse.com
IJARSE
ISSN 2319 - 8354

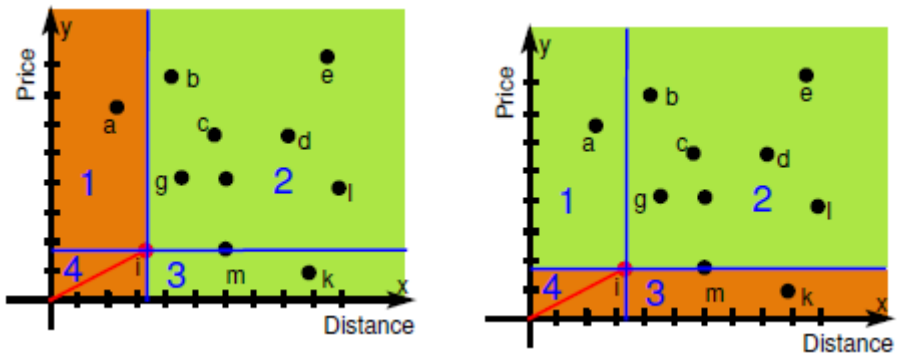


V. Overlapping the search regions

In this we firstly relax the restriction that the regions are non-overlapping and assume that thepoint query splits each dimension intotwo regions; instead of exploding aregion to 2^d, it reduces to 2d. We have traded lesser number of regionsto search at the expense dealingwith duplicate and their removalany one of these duplicate removaltechnique can be employed :

1. Laisser-Faire: maintain ainmemoryhash table that keys ineach point and _ags it a duplicateif already present in thehash-table.
2. Propagate: when a pointis found, remove all instances offrom all unvisited nodes.
3. Merge: merge partitions to formnon-overlap regions.

International Journal of Advance Research in Science and Engineering
Vol. No.5, Issue No. 02, February 2016
www.ijarse.com
IJARSE
ISSN 2319 - 8354



Also, the following section provides an overview of the algorithms that has been applied for evaluating the fuzzy queries :

I. ITERATIVE RANGE-SEARCH-BASED ALGORITHM

We study how to answer a ranking query by answering (possibly multiple) range selection queries. Each selection query has a threshold on the similarity between the given string and a string in the collection. In this way, we canleverage existing approximate-string-selection techniques [12, 6, 9] without modifying their implementations. We develop an algorithm called"Iterative Range Search"("IRS" for short). Algorithm 1 shows the pseudo-code of the algorithm. We start with an initial similarity threshold, t, which could bea fixed value or a value computed based on the query (line 5). The algorithm has two steps.

Algorithm 1 :IRS Algorithm for a top-k query

1: Let k be the number of results requested;

2: Let wmaxbe the maximum weight of a string in the dataset;

3: Let f >= 1 be a multiplication factor;

4: Let R←Θbe the range-search-result set;

5: Let tbe the initial similarity threshold;

{Step 1: Computing initial candidates}

6: while size(R) < f · k do

7: R ←ApproxRangeSearch(t);

8: if size(R) < f · k

9: then Decrease t;

10: end while

{Step 2: Finalizing results}

11: Compute scores for elements in R and keep the first k;

12: Let t1 be the minimum similarity for which

Score(t1,wmax) > Score(R[k]);

13: if t1 < tthen

14: R ←ApproxRangeSearch(t1);

15: Compute scores for elements in R and keep the first k;

16: end if

17: Return R[1..k];

Advantages and Limitations:

The IRS algorithm hasthe advantage that it can utilize any of the existing algorithms for approximate-string range search. It is easy to implement as it uses the range-search algorithm as a blackboxfunction. One main limitation of the algorithm is that it needs to run multiple search queries, which may take a lot of time. In addition, it is not easy to choose a good initial similarity threshold t(line 5) and decrease tproperly for the next query (line 9).

II. SINGLE-PASS SEARCH ALGORITHM

A better way of traversing the lists is to use a heap. The algorithm is called "Single-Pass Search" ("SPS" for short). It traverses the lists in a sorted order using a heap of the current top elements of the lists. This traversal order has two advantages: we do not have the overhead of maintaining the candidate set, and we have more chances to skip elements. Algorithm 2 shows the pseudo-code of the algorithm.

Algorithm 2 :SPS Algorithm for a top-k query

1: Let n be the number of grams in the query;
2: Let l[0..n − 1] be the lists of ids for the query grams;
3: Let g ←1 be the frequency threshold;
4: Insert the top element on each list to a heap, H;
5: Topk←Θ;
6: while H is not empty do
7: Let Tbe the top element on H;
8: Pop from H those elements equal to T;
9: Let p be the number of popped elements;
10: if p>=g then
11: if Score(T) > Score(kthin Topk) then
12: Insert T into Topkand pop the last one;
13: Recompute threshold g;
14: if g > n
15: then break;
16: end if
17: Push next element (if any) of each popped list to H;
18: else
19: Pop additional g − p − 1 elements from H;
20: Let T′be the current top element on H;
21: for each of the g − 1 popped lists do
22: Locate its smallest element E >=T′(if any);
23: Push E to H;
24: end for
25: end if
26: end while

27: Return the elements in Topk;

III. TWO PHASE ALGORITHM:

We can combine the IRS algorithm and the SPS algorithm. This new algorithm is called "two-phase" algorithm (2PH). In the first phase, we execute a single range search with a tight similarity threshold,t. In the second phase, we run the SPS algorithm, but the initial bound on the number of common grams is computed based on the records retrieved in phase 1. The algorithm is based on the following two observations. (1) Retrieving the records very similar to the query could be done efficiently using existing range-search algorithms. (2) The SPS algorithm is efficient since it can skip many elements. Still, a low initial frequency threshold makes the algorithm process a lot of elements at the beginning. The initial top-k candidates computed in phase 1 could give as a higher initial frequency threshold. Moreover, the traversal might stop earlier since the records very similar to the query have already been considered.

Also,there are three methods of combining results in database to producea compatibility ranking: minimum of memberships, weighted average ofmemberships, and weighted product of memberships.

Minimum-of-Memberships Compatibility

The simplest method of calculating the overall compatibility index for a query *Xi* takes the minimum of the predicate truth memberships (this assumes, of course, that the predicates are connected with an *And*operator). Expression below shows how this QCIX is computed for the complete query.

$$X_{qcix} = \bigwedge_{i=1}^{n} \min(\mu_i(p_i))$$

Here,

*Xqcix*is the query compatibility index for query X

*n*is the number of evaluated predicates in the queryμi( ) is a function that returns the minimum membership value

*pi*is the *i*-th predicate in the query

The minimum approach functions like the traditional *And*operator. Each query predicate is treated as an expression that is dependent on the complete set of predicates.

Weighted-Average-of-MembershipsCompatibility

A more robust method of viewing the compatibility measurement forquery *Xi* takes the weighted average of the predicate truth values. In thisapproach, each column is given a bias weight in the interval [*1*,*n*], where*n* is a large number relative to all weights greater than 1. Expression below shows how the weighted average compatibility index is computed.

$$X_{qcix} = \frac{\sum_{i=1}^{n} (\mu_i(p_i)) \times w_i}{\sum_{i=1}^{n} w_i}$$

Here,

*Xqcix*is the query compatibility index for query *X*

*n*is the number of evaluated predicates in the query

μi( ) is a function that returns min membership value

*pi*is the *i*-th predicate in the query

*wi*is contribution weight associated with *i*-th predicate

When the weights for columns are ignored (each column weight is one.

Weighted-Product-of-Membership Compatibility

In this approach, like the weighted average, each column is given a bias weight in the interval [*1*,*n*], where *n* is a large number relative to all weights greater than 1.This method biases the complete compatibility of the query in accordance to the weakest truth memberships and also takes the weighted product of the predicate truth values.

$$X_{qcix} = \prod_{i=1}^{n} \min(\mu_i(p_i)) \times w_i$$

where,

Xqcixis query compatibility index for query X

nis number of evaluated predicates in X

μi( ) is function that returns the min. membership value

pi isi-th predicate in the X

wiis contribution weight associated with i-th predicate

III. PROBLEM DESCRIPTION

This research focuses on recently introduced skyline operator. It is found that the complexity of Skyline Query processing is as high as of nested relational queries.The use of fuzzy approach is suggested by this research paper so that the complexty of the query processing may be reduced.

The problem of identifying skyline is $O(n^2)$ where "n"

is the number of rows in the data set.

The skyline uses the 'crisp' logic in query processing. It means that the record would have not been selected, even if it is extremely close to the intent of the query criterion.Also Skylines consist of only those input tuples, having better or equal values i.e. it contains only the best choices.

Further, this research is focused upon the improvement in expressivity, optimization of the output size and reduction of the complexity in queries, using the fuzzy Logic techniques[6].

IV. METHODOLOGY

One of the criteria to choose the query processing methodology is the expressivity and understanding. For humans, the only natural means of communication is natural language but this is quite difficult because the information to be processed is imperfect i.e. incomplete or ambiguous.

At this point, fuzzy logic provides powerful means to account for many aspects of imperfect information.The real potential of fuzzy queries becomes apparent when we move fromone fuzzy set to queries that combine many fuzzy sets. With multiple fuzzysets we want to find candidates in the data that have, to some degree,characteristics of our combined concepts. For example if we take the concept of Talland Heavy men. In our crisp query we need to establish a boundary pointfor the definition of Heavy. Suppose we select 193 pounds. Then the SQLquery is as follows:

```
select name
fromieh.customers
wherecustomers.height>= 6
andcustomers.weight>= 193;
```

The result of this query is reflected in Table 4.1. Only a single customeris both Talland Heavy.But ifweglance down the table, it appears that several of the customersmight be potential candidates. They are excluded simply because of thesharp boundary points imposed by the crisp SQL query. We can do betterby replacing the query with its corresponding fuzzy version, as follows:

```
select name
fromieh.customers
wherecustomers.height is Tall
andcustomers.weight is Heavy;
```

The intent of this query is obvious: we want customers that are both *Tall*and *Heavy*. To resolve the query we need to tell the computer what constitutesa *Heavy* person.

Fig.4.1

Name	Height	Weight	Age	(Tall and Heavy)
Jackson	5′3″	170	38	
Sanders	6′2″	215	52	←
Miller	5′11″	157	25	
Cassey	6′1″	188	40	
O′Malley	5′5″	163	48	
Freeman	5′10″	202	34	

Table outlines the membership values for each of the weights in the table (ranked by their membership values). (Fig.4.2)

Fig.4.2

Name	Height	Weight	Age	μ(Heavy)	μ(Tall)
Sanders	6′2″	215	52	**.9**	**.91**
Freeman	5′10″	202	34	**.55**	**.83**
Cassey	6′1″	188	40	**.25**	**.90**
Miller	5′11″	157	25	**.00**	**.87**
O′Malley	5′5″	163	48	**.00**	**.62**
Jackson	5′3″	170	38	**.00**	**.53**

Recently the Skyline Query is widely used in data analysis to derive the results that meets more than one specific condition simultaneously. But an effective skyline query process in terms of time and space complexity over uncertain data streams becomes crucial.

We prefers the Pareto Dominance property[2].This is the basic property according to which the skyline queries are implemented and this property also ensures that the items which are selected are in the skyline of the database.

However, A one-dimensional Skyline is trivial because it is equivalent to computing min, max or distinct[1]. Using it on top of relational database system: using existing SQL Queries results in poor performance.

The skyline of a relation instance r with n tuples and attributes A = {A1, …,Ad} can be naively computed in $O(n^2d)$ time, which is $O(n^2)$ if the relation schema is fixed.

The research proposes the use of Fuzzy Queries in order to substitute Skyline Queries for the sake of better performance. Most of the algorithms in Skyline operation have the performance complexities in order on $O(n^2)$ for

example, NL(Nested loops) algorithm, BNL(Block Nested Loops) etc.

Therefore, the fuzzy operations should be introduced in context of query processing to the Databases.

In consequence, some efficient algorithems have been defined in order to evaluate queries in the relational system.

In this research paper, we used the SFS i.e Sort Filter Skyline[11] algorithm.This algorithm begins sorting the table,after it passes a cursor over the sorted rows and finally it discards dominated rows.

So far, some of the approaches are being researched upon such as SQLf[3], Fuzzy Databases[4] and Fuzzy logic based query optimizations in Distributed Databases[5]. One of the main achievements has been a fuzzy extention of the most popular relational database querying language SQL to SQLf, initiated by Bosc and Pivert[6][7][8].

Also,in this research paper, we prefer theWeighted product of membership compatibility method to measure the query compatibility in fuzzy datasets.

This method biases the complete compatibility of the query in accordance to the weakest truth memberships and also takes the weighted product of the predicate truth values.

$$X_{qcix} = \prod_{i=1}^{n} \min(\mu_i(p_i)) \times w_i$$

where,

Xqcix :query compatibility index for query X

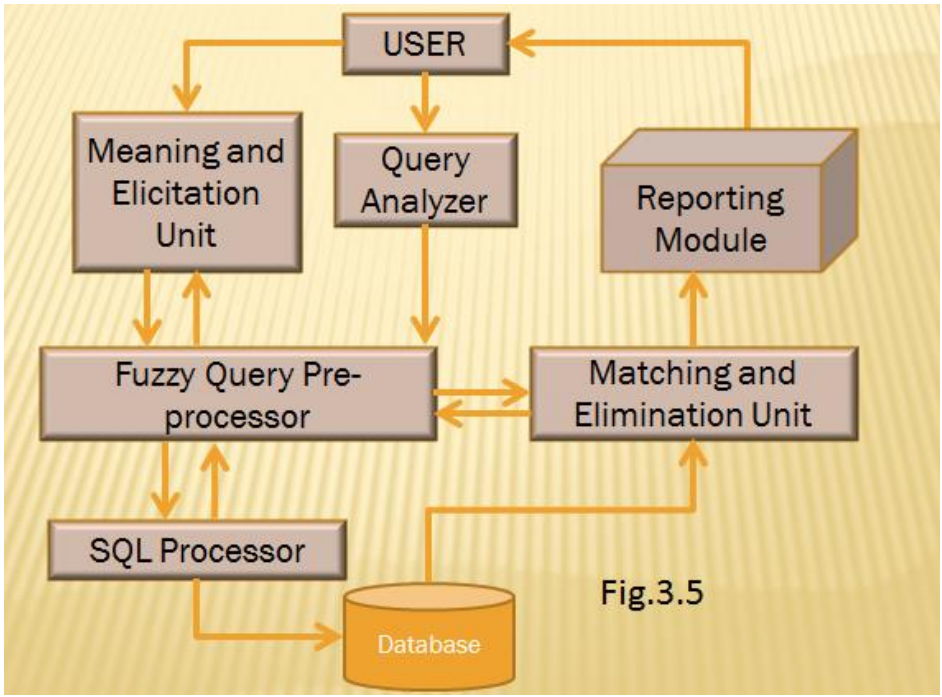
n :number of evaluated predicates in X

μi( ) : function that returns the min. membership value

pi :i-th predicate in the X

wi:contribution weight associated with i-th predicate

Below figure describes the overall methodology and process flow of a Fuzzy query Processor.



V.CONCLUSION AND FUTURE WORKS

The studies done so far suggests that the Fuzzy Logic approach in query processing is better and efficient than the recently introduced Skyline Operator. Various researchers have tried to optimize the Skyline and reduce its complexity. However, the Fuzzy approach should still perform better then any of the suggested enhancements in Skyline Methods.

[4] suggests that from the point of view of the use of naturallanguage to express and handle intentions and preferences of humans, fuzzy logic has an excellent advantage over the Skyline queries.Skyline queries results depend upon high nesting levels which results in high complexity.

Fuzzy query is an alternative to skyline queries. Non-nested queries can be achived with the help of fuzzy logic, which results in less execution cost. This research intends to provide empirical evidence based upon thorough experimentation and comparision between fuzzy queries and skyline queries.

REFERENCES

[1] Fuzzy Modeling and Genetic Algorithm, Earl Cox

[2] “Fuzziness in database management systems: Half a century ofdevelopments and future prospects”, JanuszKacprzyka,∗ , SławomirZadro˙znya, GuyDeTréb, June 2015, Elsevier

[3] About Fuzzy Query Processing, 2013, CLEI, Ricardo Pereira et al

[4] Fuzziness in DBMS: Half a century of Development and Future Prospects(Article in Press), ELSEVIER 2015

[5] Fuzzy logic based query otimizations in Distributed Databases, 2013, IJIRCCE, G.R. Bamnoteet al

[6] P. Bosc, O. Pivert, Some approaches for relational databases flexible querying, J. Intell. Inf. Syst. 1(3–4) (1992) 323–354.

[7] P. Bosc, O. Pivert, Fuzzy querying in conventional databases, in: L.A. Zadeh, J. Kacprzyk (Eds.), Fuzzy Logic for the Management of Uncertainty, Wiley, New York, USA, 1992, pp.645–671.

[8] P. Bosc, O. Pivert, SQLf: a relational database language for fuzzy querying, IEEE Trans. Fuzzy Syst. 3 (1995) 1–17.

[9] Siler, W., Buckley, J.: Fuzzy expert systems and fuzzy reasoning. John Wiley &Sons, Inc.,USA. (2005).

[10] Skyline with Presorting, Chomicki and Liang 2002.

[11] Fuzzy Dominance Skyline Queries, Marlene Goncalves and Leonid Tineo