Fuzzy Logic In DBMS

- Advances in Computer Science and the recognition ofdata as a crucial organisation resource have led to the rapid progress and development of information systems
- Existing DBMS can only handle crisp, precise and non-ambiguous data
- In other words, these systems do not cater for vague and ambiguous data which are fuzzy in nature

- The application of fuzzy set theory in DBMS can be classified into two main classes.
- Class 1 concerns the study of fuzzy query processing in conventional (nonfuzzy) DBMS; Class 2 deals with DBMS which, besides having the ability to store and manipulate fuzzy data directly, also supports fuzzy query.
- 00100

- 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

- Chang *et al.* [10] 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

- The methodology proposed is able to process fuzzy query such as the following:
- *Query:* get supplierName ; goods equal 'tv'.
- This query is fuzzy because it does not provide enough information to the DBMS - no access path is specified.
- Table is Supply (suppliername,FK)
- Stock(goods,PK)

- A full query for the above query takes the form
- *Query:* get supply.supplierName ; stock.goods equal 'tv'.
- In Chang's paper, the term "fuzzy" refers to incompletelyspecified information in the query, such as the lack of access path.

- A number of algorithms have been developed to improve the performance of searching using query. But those algorithms have been developed only for classical DB. The traditional DBMS cannot manipulate incomplete, imprecise and vague data such as very high, about 30, etc. properly. To overcome this problem, FDBMS (Fuzzy Database Management System) has been introduced.

- For representing fuzzy data, possibility-***distribution-fuzzy-relational model*** (pd-fr model),
- In this fuzzy data are represented by fuzzy relations whose grades of
  - Degree of Membership
  - attribute values can be possibility distributions,
  - where the fuzziness of an attribute value is represented by a possibility distribution and its association by a grade of membership.
  - (a,1)(b,0.9)

- We shall have a fuzzy relation RA whose attribute(PD) and membership values(Mfunction) are possibility distribution
- **(Create RA <NAME, AGE>**
- **1/<Jack, 27>,**
- **0.6/<John, 35>,**
- **{1/0.8, 0.6/0.7}/<Judy, 0.6/23<, 1/24)>,**
- **0.8/<Mike, {0.6/47, 1/48, 0.5/49}>,**
- **{1/ 1, 0.7/0.9)/<Richard, about25>})**

- Function **creates** defines a relation schema and creates an instance of fuzzy relation with setting entities. We have possibility distributions **as** attribute values such **as** (0.6/23, 1/24} and (0.6/47, 1/48, 0.5/49) and **as** membership values such **as** (1/0.8, 0.6/0.7} and {1/1,0.7/0.9}. We have a name of possibility distributions such **as about25,** which have been defined **as (term (about 25 {0.6/24, 1/25, 0.6/26}))**

Fuzzy Data Model

- A fuzzy database consists of relations: a relation is a relation
- *R(t1........, ,tn)* a Cartesian product P1 xP2 x ... x *Pn,* of domains Pi;

Fuzzy Attribute Value

- Attribute values such as age have nonfuzzy values such as fuzzy predicates such as 20 in relational database; attribute values are defined as fuzzy predicate such as “young” and “about forty” in the fuzzy database. For example, a fuzzy attribute value of “age of Dr. *x* is nearto 49” expressed as a possibility distribution {0.8/48,1/49,0.8/50}
- *p* (age of *x)* = **YOUNG; YOUNG** denotes a **fuzzy** set that represent the fuzzy predicate “young”. Thus attribute values are identified with fuzzy sets such as YOUNG.

Fuzzy Truth Values

- Truth values of any tuples are either 1(=true) or 0(=false) in the relational database; truth values of any tuples are defined as fuzzy predicates such as **“0.7”** and “completely true” in the fuzzy database. Consider, for example, a tuple *t* that asserts fuzzy proposition: “It is completely true that Dr. X is very much older than twenty.”The truth values of t is expressed as a possibility distribution P[T(t)]=N;T(t) denotes a truth value of t and N denotes a fuzzy set that represents the fuzzy predicate “completely true.” Thus the truth values T(t) are identified with fuzzy sets such as N over z[0,1];the values z [0,1]
- has the following meaning.
- 1)z=0 means the tuple is completely false
- 2)0<z<1 means that the tuple t is true to degree expressed by the real number z
- 3)z=1 means that the tuple t is completely true

THE GEFRED MODEL

- The GEFRED (Generalized Fuzzy Relational Database) model was published in 1994 by Medina-Pons-Vila[11]. It is developed in possibilistic 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 data types UNKNOWN ,UNDEFINED and NULL also.
- 7. An Unknown value with possibility distribution:
- Unknown = $\{1/x : x \in D\}$
- 8. An Undefined value with possibility distribution:
- Undefined = $\{0/x : x \in D\}$
- 9. A NULL value given by:
- NULL = {1/Unknown,1/Undefined}

Suppliers

supplier	location	material	quality
DEWAG	Paris	802.025	medium
DEWAG	Paris	802.020	medium
MAM	Berlin	802.025	high
KBA	Hamburg	802.025	high
INFORM	Aachen	802.025	low

Reliability

supplier	material	reliability
DEWAG	802.025	high
DEWAG	802.020	medium
MAM	802.025	medium
KBA	802.025	low
INFORM	802.025	high

Materials

material	description	standard
802.020	engine XL	EURO
802.025	engine L	EURO
802.020	engine XL	ISO

Select *Companies* where *Material* = EURO-NORM and Location = Paris

Answer{DEWAG}

Reliability

supplier	material	reliability	μ_R
DEWAG	802.025	high	.8
DEWAG	802.020	medium	.7
MAM	802.025	medium	.6
KBA	802.025	low	.8
INFORM	802.025	high	.9

- Q={(H,M),(S,L)}
- D=[1-10]
- Da=[1-5]
- Dun=[6-10]

supl.	supplier	material	quality	delay
	BAW	802.025	sufficient	8
	DEWAG	802.025	medium	5
	DEWAG	809.200	high	8
	KBA	802.025	sufficient	7
	KBA	809.200	sufficient	3
	KBA	840.024	low	9
	MD	802.025	sufficient	8
	MD	809.200	medium	4
	MTX	802.025	high	2
	MTX	840.024	high	4
	MAM	802.025	low	7
	MAM	840.024	medium	6
	ZT	809.200	high	8
	ZT	840.024	medium	2

Supplier	**Material**	**Q**	**D**	**RE**
DE,MAM,ZT		HM	Dun	C1
DE,MD,M,ZT		HM	Da	C2
BAW,K,MD,MAM		SL	Dun	C3
K,		SL	Da	C4

Supplier	Material	Q	D	RE
DE,MAM,ZT		HM	Dun	C1
DE,MD,M,ZT		HM	Da	C2
BAW,K,MD,MAM		SL	Dun	C3
K,		SL	Da	C4

Dun	**C1 Decrease the delays**	**C3 Terminate**		
Da	C2 New orders	C4 Provide good quality		
	HM	SL		

$$\mu_{acc.}(u) = \begin{cases} 1 & for \quad 1 \le u < 3 \\ (7-u)/(7-3) & for \quad 3 \le u < 7 \\ 0 & for \quad 7 \ge u \end{cases}$$

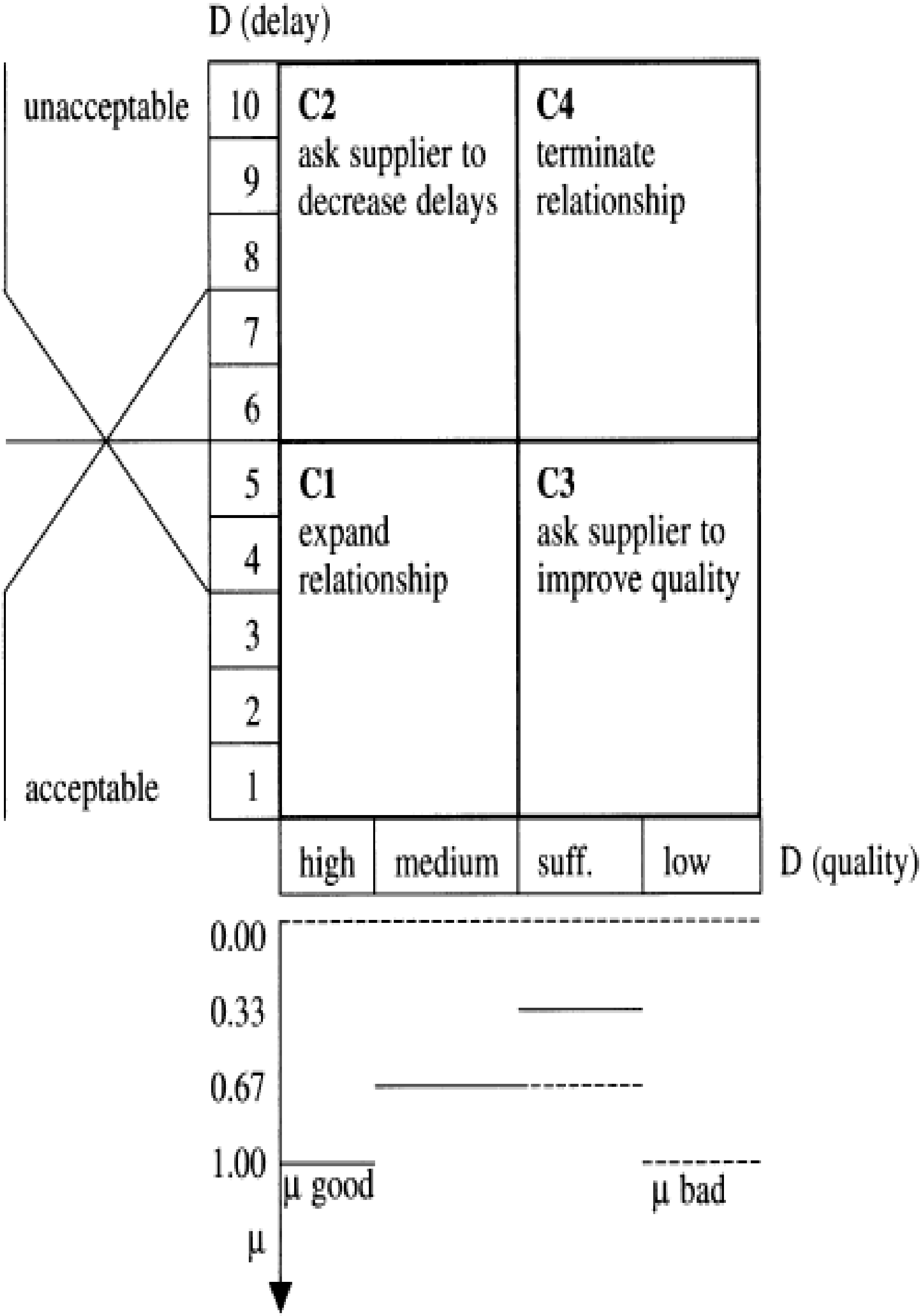
$$\mu_{unacc.}(u) = \begin{cases} 0 & for \quad u < 3 \\ (u-3)/(7-3) & for \quad 3 \le u < 7 \\ 1 & for \quad 7 \le u \end{cases}$$

For the linguistic variable "delay" we shall define the two terms "good" and "bad" with the following membership functions:

$$\mu_{\mathrm{good}}(u) = \{(\mathrm{high}, 1), (\mathrm{medium}, .67), (\mathrm{sufficient}, .33)\}$$

$$\mu_{\mathrm{bad}}(u) = \{(\mathrm{medium}, .33), (\mathrm{sufficient}, .67), (\mathrm{low}, 1)\}.$$

Graphically the class matrix would now book as follows:



Let us consider BAW
Q=sufficient D=8

Terminate
Ug=0.33
Ub=0.67

ZT Q=medium, D=2
Expand

Ug=0.67
Ub=0.33

- Fuzzy Object Oriented Database (FOOD)
- complex objects
- various hierarchies like aggregation, generalization and inheritance to be implemented on the database.
- Along with it various types of uncertainties present in the data can be well handled by the proposed model.
- Due to the implementation of fuzziness, the proposed model is enhanced to access the objects fast from the database.
- To increase the performance based on speed to access the objects from FOOD model, indexing technique based on R tree indexing is implemented.

Information Retrieval (IR)

- Information Retrieval is a way to access information available on the internet, intranet, databases and data repositories. An Information Retrieval system takes the user's query as input and returns a set of documents sorted by their relevance to the query.

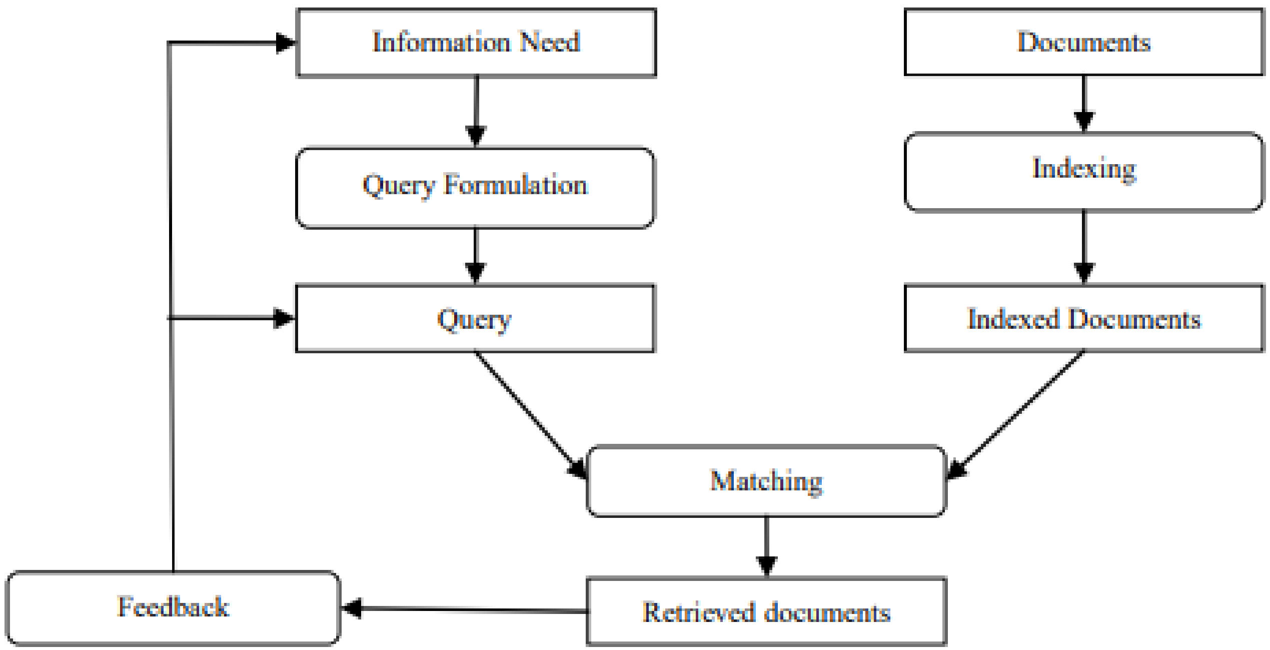


Fig. 2 information Retrieval Processes [14]

Fuzzy logic in IR

- Fuzzy logic has been in extensively use in the domain of Information retrieval from the early days of Networking. It has become more relevant in today's scenario because of its inherent ability to solve the complex problems faced by exponentially increasing size of web and web information.

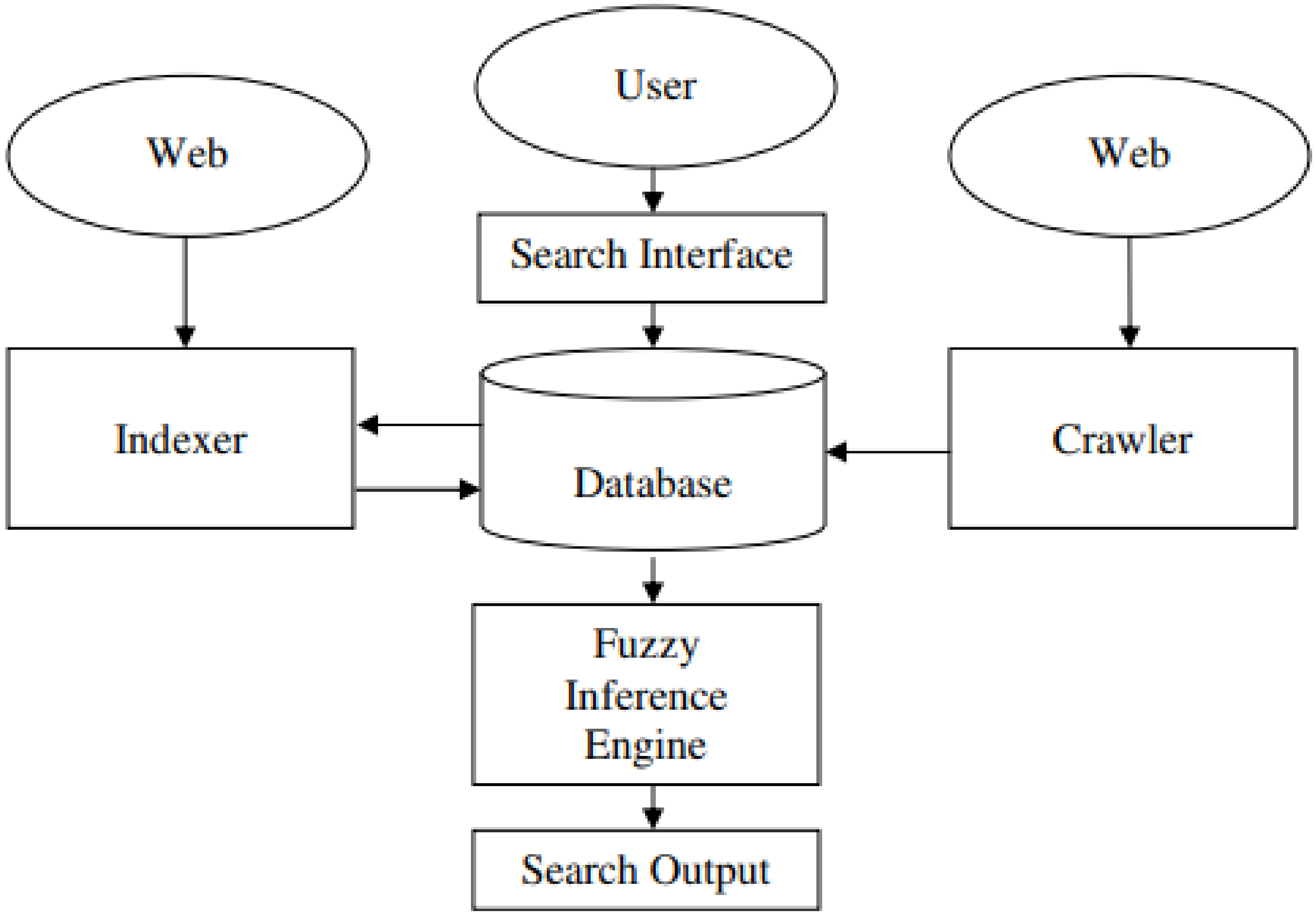


Figure 2: System Flow Diagram

Examples

- Example2: Suppose we have two documents d1 and d2 shown below. The documents are represented by two pairs of index terms and its weights as:
- Document 1, d1 ={(Desktop,0.40),(Laptop,0.40)}, =0.40
- Document 2, d2 = {(Desktop,0.99),(Laptop,0.39)}, =.39
- Query, q1 = Desktop AND laptop
- The use of MIN operator for the AND operation returns the document value for document d1 and d2 as 0.40 and 0.39 and thus d1 is retrieved with a higher rank than d2. But as one can see that the document d2 should be the obvious choice for this particular query.

- Example3: Suppose that we have two documents d3 and d4 and a query q2 as follows: :
- d3 = {(t1,0),(t2,.8,)(t3,1),…..(t100,1)},
- d4 = {(t1,0),(t2,.1,)(t3,.1),…..(t100,1)},
- Query, q2 = t2 OR t100
- In the above query, the degree of satisfaction of both the document d3 and d4 are same while a careful examination can tell us that the document d3 is more appropriate solution to the query q2.