Linguistic Values on Attribute Subdomains in Vague Database Querying

CORNELIA TUDORIE
Department of Computer Science and Engineering
University "Dunărea de Jos"
Domnească 111, 800201 Galați
ROMANIA

Abstract: - Intelligent database interfaces must be able to interpret and evaluate imprecise criteria in queries, containing certain vague terms, currently used in natural language speaking. The simplest vague selection criterion is expressed by a linguistic value, defined as fuzzy set on a database attribute domain. Generally, when a vague query is processed, the definitions of such possible vague terms must already exist in a knowledge base. There are also cases when linguistic variables must be dynamically defined, in accordance with the intermediate results of a query evaluation. A particular type of query is discussed and an evaluating procedure is proposed.

Key-Words: - Artificial Intelligence, Database, Flexible Query, Vague Criteria, Fuzzy Logic, Linguistic Variable

1 Introduction

The access to databases is possible in the following two ways:

- operating with application programs, when a limited set of predetermined functions are available and
- operating directly on data, using relational command languages.

The second one is unavoidable when an occasional operation, in particular terms, is performed. The most usual situation is database querying about various selection criteria. Two major limitations occur in such a database querying access: the rigid formal language syntax and the difficulty to realize and express precise criteria to locate the information. This happens because humans do not always think and speak in precise terms. So, it is very useful to provide intelligent interfaces to databases, able to understand natural language queries and more important, able to interpret and evaluate imprecise criteria in queries.

Including vague criteria in a database query may have two advantages:

- the flexibility of the query expression
- the possibility to refine the results, assigning to each tuple the corresponding degree of criteria satisfaction.

We particularly focus on the possible vagueness of the selection criterion, which involve certain vague terms, currently used in natural language speaking. So, all the discussion in that direction is inspired from the usual necessities of our final database users, expressed in many various linguistic forms.

The fuzzy set theory is already established as the adequate framework to model and to manage vague expressions, or in other words, to evaluate vague queries sent to relational database.

The selection vague criteria may be very simple, but it may also be very complex. We consider mainly the linguistic complexity, not the logical one. The linguistic complexity of the criterion is coming from various categories of vague terms with different semantic effects on the selection criterion, hence the logical complexity.

A review of several categories of linguistically terms with vague meaning, their fuzzy model and specific operations are presented in [4], [5], and many others.

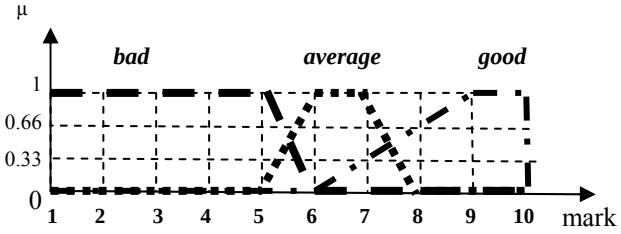

Fig. 1.The linguistic domain for the **mark** attribute

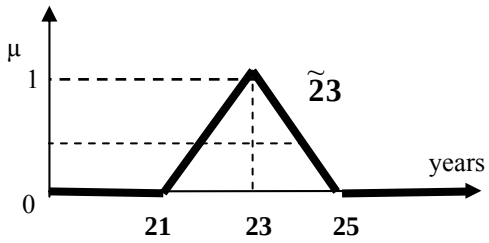

Fig. 2. The definition of a fuzzy predicate by a fuzzy number

STUDENT

Name	**. . .**	**Mark**	**Age**	**. . .**
John		7	21	
Mary		8	22	
George		10	25	
Helen		9	23	
Robby		7	24	
Michael		8	22	
Dan		4	25	
Paul		9	20	
Justin		6	20	
Jerry		5	21	
Mandy		7	26	

Table 1. A relational table

For example, if the table 1 and the definitions in figures 1 and 2 are considered, the response to the query

Retrieve the students ***around 23 years***
having ***average or good*** *mark*

will be

Name	**Mark**	**Age**	$\mu_{average}$	μ_{good}	$\mu_{\widetilde{23}}$	**μ**
Helen	9	23	0	1	1	1
Mary	8	22	0	0.66	0.5	0.5
Robby	7	24	1	0.33	0.5	0.5
Michael	8	22	0	0.66	0.5	0.5

Table 2. Ranked result of a vague query

In order to evaluate this query, each vague term is considered according to its fuzzy model (or its *definition*).

Generally, before evaluating a query, a knowledge base containing such term definitions must already exist. In the following, a particularly case is described: when a dynamic definition of vague terms is needed.

2 Crisp and Linguistic Domain

The **linguistic label** is a word (usually coming from natural language) that designates a fuzzy entity.

A linguistic label may be assigned to a fuzzy set or fuzzy quantity, suggesting a vague term from usual language, typical to the application area where our model is working. Although it is relying on the same representation style, the linguistic label may have various meaning, depending on the problem nature. For example, it may indicate:

- A *gradual property* (‚***good* mark**' for a student), when the membership degree of the fuzzy set is a function defined on an attribute domain (the [1,10] interval for the **mark** attribute), generally a monotonous function; the value of the membership function expresses the *intensity of the property*: between the 0 degree (that is a not good mark), and the 1 degree (that is an absolutely good mark).
- A *category of objects*, eventually having a certain property (‚***intelligent* student**'), when the membership function is not always monotonous and its value expresses the *closeness degree* of the current object to the object considered as *the representative one for that category*.

The linguistic label stands for the semantic model of the fuzzy entity to which it is assigned. Therefore, in order to build the knowledge model for a given application domain using the fuzzy sets formalism, both aspects must be taken into account: the linguistic representation and the numerical representation of the knowledge pieces.

If some fuzzy sets are defined on the same referential domain, with different membership functions, then a ***fuzzy set family*** is formed. If a linguistic label is assigned to each fuzzy set, then the set of these labels may be the definition set of a ***linguistic variable***, and the labels are named ***linguistic values***.

The **linguistic variables** is the quadruple:

$(V, E(V), U, M)$ where

V is the name of the linguistic variable

$E(V)$ is a set of linguistic values for the linguistic variable V

U is the crisp referential domain of the linguistic variable V

M is a mapping $E(V) \rightarrow \mathcal{F}(U)$ which maps a fuzzy set on U for each linguistic values of V.

The choice of numerical representation of a linguistic term is seldom obvious. However, at the qualitative level, the term is well understood and well semantically placed with respect to other linguistic expressions. Thus, an order relation $\underline{\prec}$ on $E(V)$ is easy to define; for example:

little $\underline{\prec}$ *intermediate* $\underline{\prec}$ big

Obviously, $\underline{\prec}$ is a semantic order relation.

In most cases, the set of the linguistic values for a linguistic variable

$$E(V) = \{ e_i \}, i = 1,..., n_e$$

is an ordered set ($e_i \prec e_j$ for $i \leq j$) having an odd cardinality. The definition of the ***intermediate*** value is generally situated in the centre of the referential domain, while the others are placed symmetrically on the two sides (like in figure 3). Some possible operators manipulating linguistic values may be ([3]):

1. $NEG(e_i) = e_j , j = n_e - i$
2. $MAX(e_i , e_j) = e_i$, if $e_j \prec e_i$
3. $MIN(e_i , e_j) = e_i$, if $e_i \prec e_j$

This intuitive order relation $\prec$ between linguistic values is the correspondent at the semantic level of a pre-order relation $\blacktriangleleft$, established between fuzzy sets defining the linguistic values (this relation is defined by Ulrich Bodenhofer in [1])

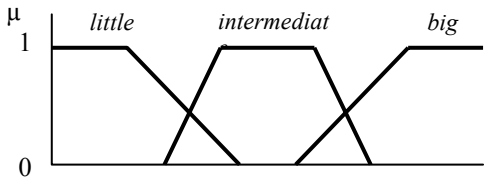

Fig. 3. Definitions of a set of linguistic values

In figure 3, the membership functions for three fuzzy sets corresponding to linguistic values *little*, *intermediate*, *big* are represented on the same axis.
It is easy to show that the relation at the linguistic level:

$$little \prec intermediate \prec big$$

is transferred at the numerical level:

$$M(little) \blacktriangleleft M(intermediate) \blacktriangleleft M(big)$$

The property of a set of linguistic values to preserve the order of the linguistic values at the fuzzy sets definitions too is named ***interpretability*** and is defined in [2].

Let be V a linguistic variable defined on the domain D of the table attribute A. The linguistic values of the V variable form the **linguistic domain** of the A attribute.

So, in a vague query context, ***the crisp domain*** (the domain attribute, according to the relational model theory) and the ***linguistic domain*** must be defined for each table attribute.
For example,
the crisp domain D=[1,10] and
the linguistic domain
L={ ***bad, almost bad, average,almost good, good*** }
may be associated to the attribute **mark** of the STUDENT table (table 1).

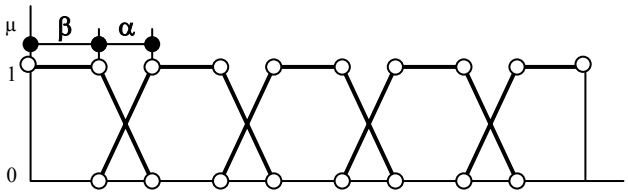

Fig. 4. A set of linguistic values on a referential domain

In most applications, defining the linguistic values set covers almost uniformly the referential domain. There are usually 3, or 5 or another odd number of linguistic values.

Starting from this general idea, we have already proposed a software interface (**FuzzyKAA** system, [6]), able to assist the user for linguistic values defining in a database context. The system proposes a uniform partitioning of the attribute domain, by trapezoidal membership functions, like in figure 4.

The correspondence between β and α parameters is implicit, but it can be changed any time. The definitions implicitly obtained, can be adjusted either by changing numerical coordinates of graphical points, or by directly manipulating of them.

This semi-automate method to create the knowledge base containing vague terms definition for vague query evaluation has a great advantage: details regarding effective attribute domain limits, or distributions of the values, can be easy obtained thanks to directly connecting to the database.

3 Linguistic values defined on a subdomain

The humans use an enormous number of expression variants in their common language. This fact determined us to study the possibility to find the most accurate model for the query sent to the database, thus the computational treatment and the response to be as adequate as possible.

The study has found a new class of problems, which require a partitioning of a limited subset of an attribute domain, but not the whole domain, in order to model the linguistic labels.
The query contains a fuzzy selection criterion addressed to one group of database objects, which requires a domain adjustment, in accordance to the group members; at its turn the group is already obtained by another fuzzy selection applied to the whole database (or table).

Let's consider the query addressed to the STUDENT table (table 1), as follows:

*Retrieve the **good** students within the **young** ones*

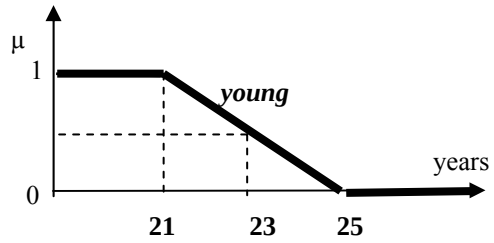

Fig. 5. The ***young*** linguistic label definition

The query evaluation procedure respects the following steps:

1. The selection criterion **age *young*** is evaluated, taking into account the definition in the figure 5; a tuple group as intermediate result is obtained, where the condition $\mu_{young}>0$ is satisfied (table 3). In other words, the students not at all ***young*** are eliminated.

Name	Mark	Age	μ_{good}
George	10	25	1
Helen	9	23	1
Paul	9	20	1
Mary	8	22	0.66
Michael	8	22	0.66

Table 3.

2. The interval containing the mark values for the selected students forms the **mark** subdomain [5,9]; it is considered later, instead [1,10].

3. The linguistic value set { ***bad, almost bad, average, almost good, good*** } will partition this subdomain (figure 6).

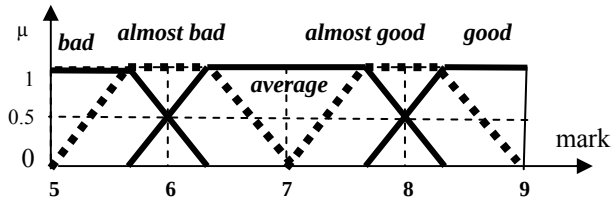

Fig. 6. Linguistic values defined on a subdomain

4. The global criterion satisfaction degree will result for each tuple, taking into account the satisfaction degree for the criterion ***good* mark** (fig 6) and the degree of the ***young age*** for the student (table 3). The table 4 contains the computed global degrees.

Name	Mark	Age	μ_{young}	μ_{good}	μ
Paul	9	20	1	1	1
Justin	6	20	1	0	0
John	7	21	1	0	0
Jerry	5	21	1	0	0
Mary	8	22	0.75	0.5	0.5
Michael	8	22	0.75	0.5	0.5
Helen	9	23	0.5	1	0.5
Robby	7	24	0.25	0	0

Table 4.

We propose a discussion, in order to validate the response provided by the proposed procedure, if it respects the query meaning. Other expression variants, more or less similar to the previous example, will be analysed, and the differences, at both numerical and semantically levels, will be pointed out.

i. The simple criterion query

*Retrieve the **good** students*

can be classically processed by the fyzzy logic, taking into account the linguistic values defined on the whole attribute domain (figure 7).

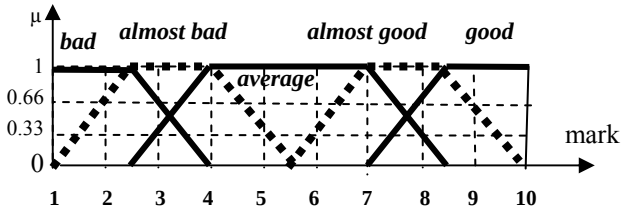

Fig. 7. Linguistic values defined on the **mark** domain

The response contains any age students, respecting the ***good* mark** criterion (table 5). George, for example, not at all ***young***, has got the best **mark.**

Name	Mark	Age	μ_{young}
Paul	9	20	1
Justin	6	20	1
John	7	21	1
Jerry	5	21	1
Mary	8	22	0.75
Michael	8	22	0.75
Helen	9	23	0.5
Robby	7	24	0.25

Table 5.

ii. A query containing a conjunction by two simple criteria, for example:

*Retrieve the **good** **AND** **young** students*

will return the students both ***good*** and ***young***, ranked by the global satisfaction degree (table 6).

It is obvious that table 4 contains a more restrictive response (lower satisfaction degrees), because the ***good*** **mark** criterion is applied on a limited students group, not on the original table.

One can remark that the selected tuples respect the same order in the all three discussed queries.

Name	Mark	Age	μ_{young}	μ_{good}	μ
Paul	9	20	1	1	1
Justin	6	20	1	0	0
John	7	21	1	0	0
Jerry	5	21	1	0	0
Mary	8	22	0.75	0.66	0.66
Michael	8	22	0.75	0.66	0.66
Helen	9	23	0.5	1	0.5
Robby	7	24	0.25	0	0
George	10	25	0	1	0
Dan	4	25	0	0	0
Mandy	7	26	0	0	0

Table 6.

iii. A quite different query expression may be:

*Retrieve the **best** students within the **young** ones*

In the precise query context, the ***best*** **mark** criterion applied on a tuples group is equivalent to the aggregation MAX function and returns the (only one) student having the greatest mark. But in the imprecise context, the ***young*** students group is a fuzzy set too (a membership degree is attached to each student). Here the best mark can correspond to any student, but not necessary to the youngest one. So, a ranked list of the students satisfying the ***good*** **mark** criterion, but also taking into account their ***young age*** degree, is the most adequate response to the above query. In other words, we consider this query can be assimilated with the initially discussed one (*Retrieve the **good** students within the **young** ones***)**, so it can be submitted to the same evaluation procedure. Moreover, it may be even more suggestive for the database user, and semantically adequate to the response in table 4.

In conclusion, the solution given by the proposed query evaluation procedure consists in a *relative selection* of the database tuples, where the selection criterion is not an absolute one, but it is relative to a subclass of already selected tuples.

Similar procedures can be used to evaluate more complex queries, including dynamical aggregation for example, such as:

*How many **best** students are within the **young** ones ?*

4 Conclusion

A particular type of vague query sent to a database is discussed in this paper and a procedure for evaluate it is proposed. The main idea is to dynamically define sets of linguistic labels on limited attribute domains, determined by previous fuzzy selections.

After a comparative analysis including other similar query types, well known as fuzzy model, we conclude that: the evaluation procedure, proposed by this paper, provides an accurate model for the discussed vague expression, with respect to query semantic.

References:

[1] U. Bodenhofer, A general framework for ordering fuzzy alternatives with respect to fuzzy orderings, *8th Int. Conf. on Information Processing and Management of Uncertainty (IPMU 2000)*, Madrid, 2000, pp. 1071-1077

[2] U. Bodenhofer and P. Bauer, Towards an Axiomatic Treatment of „Interpretability", *6th International Conference on Soft Computing*, Iizuka, 2000, pp. 334-339

[3] E. Herrera-Viedma, An Information Retrieval System with Ordinal Linguistic Weighted Queries Based on Two Weighting Semantics, *8th International Conference on Information Processing and Management of Uncertainty in Knowledge-Bases Systems(IPMU 2000)*, Madrid, 2000

[4] C. Tudorie, Cercetări privind aplicarea tehnicilor de inteligență artificială pentru interogarea bazelor de date, *Scientific Report*, University 'Dunărea de Jos', Galați, 2003

[5] C. Tudorie, Vague criteria in relational database queries, In *Bulletin of "Dunarea de Jos" University of Galați*, III/2003, pp. 43-48,

[6] C. Tudorie, Contribuții la realizarea unei interfețe inteligente pentru interogarea bazelor de date, *Scientific Report*, University 'Dunărea de Jos', Galați, 2003