AUTOMATIC MODELING OF THE LINGUISTIC VALUES FOR DATABASE FUZZY QUERYING

Cornelia TUDORIE*, Laurentiu FRANGU, Diana STEFANESCU***

**Department of Computer Science, **Department of Electronics and Telecommunications*
"Dunarea de Jos" University, Domneasca 47, 800008 Galati, ROMANIA
email: Cornelia.Tudorie@ugal.ro, Laurentiu.Frangu@ugal.ro, Diana.Stefanescu@ugal.ro

Abstract: In order to evaluate vague queries, each linguistic term is considered according to its fuzzy model. Usually, the linguistic terms are defined as fuzzy sets, during a classical knowledge acquisition off-line process. But they can also be automatically extracted from the actual content of the database, by an online process. In at least two situations, automatically modeling the linguistic values would be very useful: first, to simplify the knowledge engineer's task by extracting the definitions from the database content; and second, where mandatory, to dynamically define the linguistic values in complex criteria queries evaluation. Procedures to automatically extract the fuzzy model of the linguistic values from the existing data are presented in this paper.

Keywords: Database, Flexible Query, Linguistic Values, Knowledge Acquisition

1. INTRODUCTION

Querying relational databases means selecting the database rows satisfying a Boolean selection criterion. For example, when the following crisp query (i.e. precise, or classical query) is sent to a database

Retrieve the students having the mark greater than 8

the Boolean criterion *mark* > 8 is evaluated for each database row. The answer contains the database rows that satisfy the Boolean formula.

But humans do not always think and speak in precise terms. So, the database user often needs to ask for information, by expressing selection criteria using linguistic terms, or vague expressions, like the following example

Retrieve the students having good *mark*

This time, the criterion 'good *mark'* is no longer Boolean, so for each table row a satisfaction degree has to be computed, expressing the measure of its compatibility to the vague criterion.

The fuzzy set theory is well known as the most adequate framework to model and to manage vague expressions. Generally, in real-world fuzzy applications, a knowledge base containing such term definitions must already exist. The knowledge engineer has to define the linguistic terms, as fuzzy sets, during an off-line knowledge acquisition process.

In a database context, in order to evaluate vague queries, each linguistic term is considered according to its fuzzy model, i.e. its definition (Projet BADINS, 1995,1997; Bosc and Prade, 1997; Dubois and Prade, 1996; Kacprzyk and Zadrozny, 2001; Pivert, 1991; Yager, 1997; Zadeh, 2001). But, in the database query context, an aspect is very important: the data in the database are always available. Moreover, details regarding actual attribute domain limits, or distribution of the values, can easily be obtained online. Therefore, the fuzzy model of the linguistic terms can be automatically extracted from the actual content of the database, by an online process. This may happen in most of the database applications where vague queries are accepted.

Methods to automatically modeling linguistic values starting from the database content are presented in the paper. They are two major usefulness, relating to database fuzzy querying:

This paper was recommended for publication by Dorel Aiordăchioaie

1. during the knowledge acquisition process, when the fuzzy model of the linguistic terms must be defined, the knowledge engineer's task can be simplified, by suggesting him implicit definitions, automatically resulting from the actual data distribution on the database attribute domains.
2. in some query evaluation processes, where the selection criteria are more complex and dynamic modeling procedures for the linguistic values are mandatory – relative object qualification (Tudorie and Dumitriu, 2004; Tudorie, Bumbaru and Segal, 2006; Tudorie, Bumbaru and Dumitriu, 2006) - like the following:

   Retrieve the ***inexpensive*** *cars* **among** *the* ***high speed*** *ones.*

   In fact, this was the reason why we found useful to investigate this subject.

First, this paper will define the attribute linguistic domain; then, some algorithms to extract the linguistic values definition will be proposed. Conclusions rising from the experimental validation and two laboratory implementations will also be presented.

2. ATTRIBUTE. CRISP DOMAIN AND LINGUISTIC DOMAIN

The *linguistic label* is a word (usually coming from natural language) that designates a fuzzy entity. It suggests a vague term from usual language, typical to the application area where the model is working.
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 a number of fuzzy sets are defined on the same referential domain, with different membership functions and a linguistic label is assigned to each of them, then all these labels form the definition set of a *linguistic variable*; the labels are named *linguistic values*. Properties of and relations on linguistic values set are discussed in (Bodenhofer, 2000), (Bodenhofer and Bauer, 2000), (Herrera-Viedma, 2000).

In a database context, a linguistic value assigned to a fuzzy set on an attribute domain, expresses a *gradual property* for the database objects. For example, one may indicate a '*big* capital' for a commercial company (Fig. 1), or a '*young* age' for an employee, etc. The value of the membership function for an attribute value expresses the *intensity of the property* for the database object corresponding to that attribute value: between the 0 degree (for example, that is not a big capital), and the 1 degree (for example, that is an absolutely big capital).

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

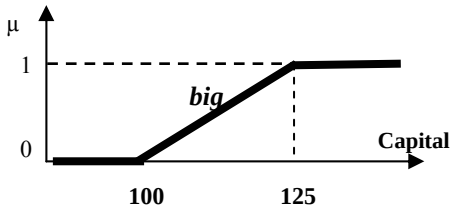


Fig. 1. The *big* linguistic value definition on the 'Capital' attribute domain

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 attribute.

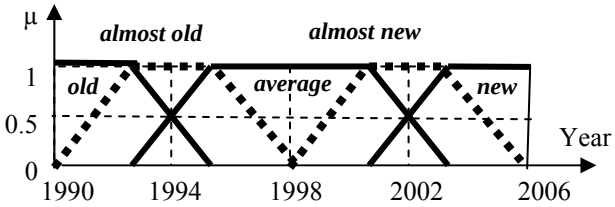


Fig. 2. The linguistic domain of the 'Year' attribute

For example,
the crisp domain D=[1990, 2006] and
the linguistic domain
L={ *old, almost old, average, almost new, new* }
might be associated to the attribute Year of a COMPANY table. They are shown in Fig. 2.

In most applications, defining the linguistic values set of a linguistic variable, covers almost uniformly the referential domain. There are usually 3, or 5 or another odd number of linguistic values. The question is when and how these definitions are found out, because they must already exist before the query is addressed to the database. Usually, the knowledge engineer has to identify and then to describe these definitions, starting from an expert opinion.

In the following, we propose a new method to acquire knowledge, that is an automatic modeling process as well as some algorithms, able to extract the fuzzy model of the linguistic values from the database content.

3. ALGORITHMS TO MODELING THE LINGUISTIC DOMAIN

For most of Artificial Intelligence applications, the required knowledge is obtained during an off-line acquisition process, by the knowledge engineer and

the domain expert. We found that in the database fuzzy querying context, contrarily to all others, the great advantage of the availability of the data may be very important. Starting from the database content, the fuzzy model of the linguistic values can be obtained.

3.1 Algorithm by Uniformly Domain Covering

In most applications, defining the linguistic values set almost uniformly covers the referential domain (Fig. 3). The number 3 for the linguistic domain cardinality of the attributes was accepted, as a simplification.

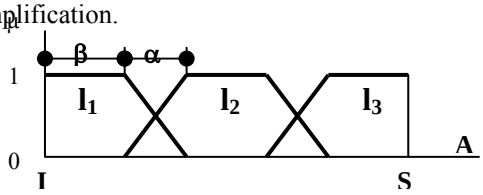


Fig. 3. The linguistic domain of the 'Year' attribute

Obtaining the definition for three linguistic values $\mathbf{l_1}$, $\mathbf{l_2}$, $\mathbf{l_3}$ on a database attribute starts from the attribute crisp domain limits, I and S, coming from the database content, and from the predefined values α and β, for example

$$\alpha = \frac{1}{8}(S - I) \qquad \text{and}$$

$$\beta = 2\alpha = \frac{1}{4}(S - I)$$

The membership functions for $\mathbf{l_1}$, $\mathbf{l_2}$, $\mathbf{l_3}$ are:

$$M_{l_1} = \begin{cases} 1 & , \ I \le t \le I + \beta \\ 1 - \dfrac{t - (I + \beta)}{\alpha} & , \ I + \beta \le t \le I + \beta + \alpha \\ 0 & , \ t \ge I + \beta + \alpha \end{cases}$$

$$M_{l_2} = \begin{cases} 0 & , \ I \le t \le I + \beta \\ 1 - \dfrac{I + \beta + \alpha - t}{\alpha} & , \ I + \beta \le t \le I + \beta + \alpha \\ 1 & , \ I + \beta + \alpha \le t \le I + 2\beta + \alpha \\ 1 - \dfrac{t - (I + 2\beta + \alpha)}{\alpha} & , \ I + 2\beta + \alpha \le t \le I + 2\beta + 2\alpha \\ 0 & , \ t \ge I + 2\beta + 2\alpha \end{cases}$$

$$M_{l_3} = \begin{cases} 0 & , \ It \le I + 2\beta + \alpha \\ 1 - \dfrac{I + 2\beta + 2\alpha - t}{\alpha} & , \ I + 2\beta + \alpha \le t \le I + 2\beta + 2\alpha \\ 1 & , \ t \ge I + 2\beta + 2\alpha \end{cases}$$

where v=t.A is a value in the domain D (that is [I,S]) of a attribute A of a table R.
The FuzzyKAA system (Apendix A) is able to assist the user for linguistic values defining in a database context, starting from a uniform partitioning of the attribute domain, like above.

3.2 Statistical Mean-based Algorithm

A new idea for the other algorithms is to take into account the statistical distribution of the data on the attribute domain. So, the median trapezoid is centered on the statistical mean of the attribute values existing in the database. The other membership functions are distributed on the rest of the interval, at the left and right of the middle one (Fig. 4).

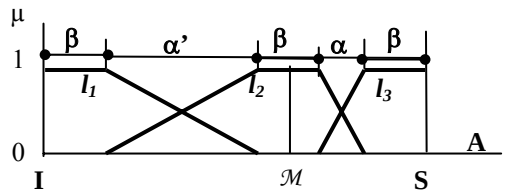


Fig. 4. Defining the linguistic values based on the statistical mean of the attribute values

In this case, the input data that determine the fuzzy models are the attribute crisp domain limits (I and S) and also the statistical mean value into the [I,S] interval.

$\mathcal{M} = \sum_{i-1}^{n} t_i.A$, where n is the cardinality of the relation R and t_i is a tuple $t_i \in R$

The values of α, β, and α' are based on I, S, $\mathcal{M}$ and they can be:

$$\alpha = \frac{1}{4} \cdot \min(\mathcal{M} - I, S - \mathcal{M})$$

$$\beta = 2\alpha = \frac{1}{2} \cdot \min(\mathcal{M} - I, S - \mathcal{M})$$

$$\alpha' = (S - I) - \frac{7}{4} \cdot \min(\mathcal{M} - I, S - \mathcal{M})$$

If $0 < \alpha < \frac{1}{8}(S - I)$, $0 < \beta < \frac{1}{4}(S - I)$

then $\frac{1}{8}(S - I) < \alpha' < (S - I)$

The formulae of the membership functions, M_{l_1}, M_{l_2}, M_{l_3}, for the linguistic values l_1, l_2, l_3 are depending on the asymmetry of the statistical mean value into the [*I*,*S*] interval. They are similarly obtained like in section 3.1.
The symetry $\mathcal{M} - I = S - \mathcal{M}$ corresponds to the particular case discussed in the section 3.1.

Exemple. Let STUD be a table containing data about students.

STUD [Name, …, Age, …]

Name	. . .	**Age**	. . .
Elena		20	
Ioana		23	
Maria		21	
Paul		25	
Vasile		22	
Costel		24	
Ion		20	
Marin		22	
Matei		28	
George		21	
Ana		20	
Sorin		21	
Florin		24	

Remarks.
i. One can observe that the second algorithm provides more accurate definitions of the semantics of the linguistic terms, existing in our common thinking.

ii. The second method is no longer adequate in situations with small amount of data (the available data are irrelevant to the phenomenon); finding a representative domain model is not possible. In this case, the first method, uniformly distributing the linguistic values, is recommended.

iii. In order to improve the algorithms when extracting the linguistic domain model from the database content, one can take into account all historical attribute values in the average computing, not only the current ones. This can be an advantage at least for the low volume databases, with an insufficient perspective on the attribute domain, or for the databases with highly dynamic values.

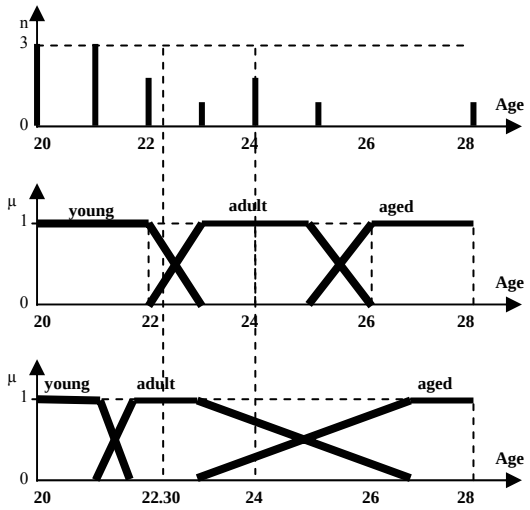


Fig. 5. An example of attribute values distribution in the database. Two possible definitions of the linguistic domain obtained by two different algorithms

3.3 Clustering-based Algorithm

One particular distribution of the effective values in the attribute domain can often be present: it's shape leads to the idea of clustering the objects according to that attribute (Fig. 6).

Clustering is a common technique in the domains of the data analysis and data mining. It is interesting to map the linguistic values of the attribute to the clusters and then to automatically extract their fuzzy models. In order to follow as close as possible the position and the shape of the clusters, some aspects must be taken into account:

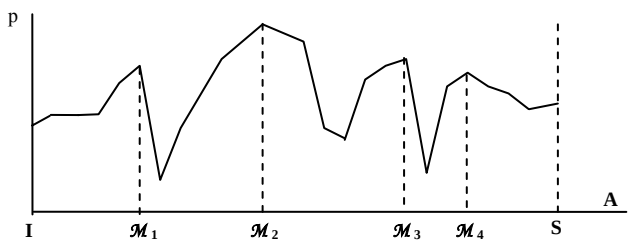


Fig. 6. An example of attribute values distribution in the database, suggesting the clusterisation

- each point corresponding to a local maximum of the distribution will be the center of the trapezoid modeling a linguistic value.
  $\mathcal{M}_k$, $\mathcal{M}_k \in [\, I, S \,]$
- the width of the core (the area corresponding to the value 1 of the satisfaction degree) depends on the minimum of the distance between two neighbor points of local minimum.
  $$\beta = 2 \cdot \frac{1}{3} \cdot \min_k \{ (\mathcal{M}_k - \mathcal{M}_{k-1}) \} ,$$
  $k = 1, \ldots, n$, $\mathcal{M}_0 = I$, $\mathcal{M}_n = S$
- the other parameters and the membership functions can be computed (Fig. 7)
  $\alpha_k = \mathcal{M}_k - \mathcal{M}_{k-1} - \beta$,
  $k = 1, \ldots, n$, $\mathcal{M}_0 = I$, $\mathcal{M}_n = S$

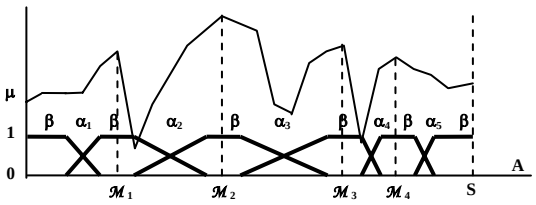


Fig. 7. An example of linguistic values definition according to the clusters

This method is certainly the most adequate for a good representation of the object qualification, in a particular class of situations.

4. EXPERIMENTAL VALIDATION

The aim of the experimental validation is to compare the fuzzy models extracted from the database by the proposed algorithms and the user perspective on the semantic of the same linguistic terms. In order to validate the presented algorithms, an experimental software, MultiDef (Appendix B), was used.

The experiment is based on some surveys with participation of some "domain experts". Each survey consists in filling a form referring to the fuzzy models of the linguistic terms for the same database. The proposed models are compared with each of the tree algorithms automatically generated definitions. The survey data post-processing is as follows:

The difference between the model of the attribute linguistic domain proposed by the expert and the result of the algorithm is evaluated by computing a resemblance coefficient

$$C_{M/m} = 1 - \frac{1}{n \cdot (S - I)} \cdot \sum_{k=1}^{n} \int_{I}^{S} \left| M_k(v) - m_k(v) \right| \, dv$$

where I and S are the crisp limits of the attribute's domain,

n is the number of values in the attribute linguistic domain,

M_k is the membership function of the k-th linguistic value obtained by the algorithm,

m_k is the membership function of the same linguistic value, drawn by the expert.

The *C* coefficient varies between 0 and 1, where

1 means a perfect resemblance (identity),

0 means no resemblance at all,

and the intermediary values can suggest:

0.95 - small difference

0.8 - substantial difference

< 0.65 - major difference

One series of such experiments is presented and discussed in (Tudorie, 2006b). It confirms that the experience of the experts meets almost perfectly the semantic extracted from the database content, by the proposed methods. More conclusions are presented in the following.

5. CONCLUSIONS

The first two algorithms are applicable for any data distribution on the attribute domain, even a uniform one. The number of the linguistic values can be arbitrarily chosen. Usualy, an odd number is proposed for the cardinality of the linguistic domain; almost always three. The principles of modeling are always the same.

The last algorithm is applicable when the linguistic domain corresponds to the clusters standing out on the attribute domain, so the number of linguistic values is now deterministic.

The proposed methods, to extract from the database content the definitions of the linguistic values on the attributes domains, are very useful in two kinds of situations:

- during the knowledge acquisition process, in order to build a fuzzy querying interface
- to evaluate complex queries where dynamic modeling of the linguistic values is necessary.

The permanent availability of the real values existing in the database is the main advantage, allowing to extract the model of the lingustic terms, any time, from the database content.

In order to improve the proposed modeling method, one can develop a global procedure to detect the proper algorithm to apply; a parametric identification process is then needed: if at least two points of local maximum are obvious, the third algorithm is applicable; otherwise, the second algorithm (that covers the first one) can be applied.

The importance of the presented method is not only coming from the effective procedure of knowledge extraction, but especially from its usefulness, at least when this process is inevitable (for example, the relative qualification).

REFERENCES

BADINS, Projet (1995, 1997). *Bases de données multimédia et interrogation souple*. Rapport d'activité scientifique, Institut de recherche en informatique et systèmes aléatoires, Rennes

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

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

Bosc, P., Prade, H. (1997). An Introduction to Fuzzy Set and Possibility Theory-based Approaches to the Treatment of Uncertainty and Imprecision in Data Base Management Systems.. In Motro, A., Smets, P. (eds.): *Uncertainty Management in Information Systems: from Needs to Solutions*, Kluwer Academic Publishers, pp. 285-324

Dubois, D., Prade, H. (1996). Using fuzzy sets in flexible querying: Why and how?. In H. Christiansen, H., Larsen, H.L., Andreasen, T. (eds.): *Workshop on Flexible Query-Answering Systems*, pp. 89-103

Herrera-Viedma, E. (2000). An Information Retrieval System with Ordinal Linguistic Weighted Queries Based on Two Weighting Semantics. In *8th International Conference on Information Processing and Management of Uncertainty in Knowledge-Bases Systems*, Madrid, pp. 454-461

Kacprzyk, J., Zadrozny, S. (2001). Computing with words in intelligent database querying: standalone and Internet-based applications. *Information Sciences*, Vol. 134. Elsevier, pp. 71-109

Pivert, O. (1991). *Contribution à l'interrogation flexible de bases de données - Expression et évaluation de requêtes floues.* Thèse, Université de Rennes I

Tudorie, C., Dumitriu, L. (2004). How are the Attribute Linguistic Domains Involved in Database Fuzzy Queries Evaluation. In *Scientific Bulletin of "Politehnica" University of Timisoara*, Vol. 49(63), Timisoara, Romania, pp. 61-64

Tudorie, C., Bumbaru, S. , Segal, C. (2006). New Kind of Preference in Database Fuzzy Querying. In *International Conference on Information Processing and Management of Uncertainty in Knowledge-based Systems*, Paris, pp. 1389-1395

Tudorie, C. (2006a). Laboratory software tools for database flexible querying. In *International Conference on Information Processing and Management of Uncertainty in Knowledge-based Systems*, Poster Section, Paris, pp. 112-115

Tudorie, C., Bumbaru, S., Dumitriu, L. (2006). Relative Qualification in Database Flexible

Queries. In *IEEE Conference On Intelligent Systems*, London, pp. 83-88

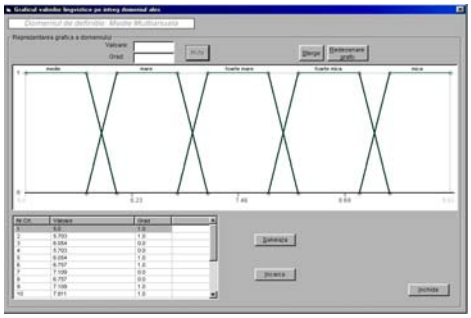
Tudorie, C. (2006b). *Contributions to interfaces for database flexible querying*. PhD Thesis, University 'Dunărea de Jos', Galați, Romania

Yager, R.R. (1997). Fuzzy logics and artificial intelligence. In *Fuzzy Sets and Systems*, Vol. 90(2), Elsevier Science, pp. 193-198

Zadeh, L.A. (2001). From Computing with Numbers to Computing with Words. In *Annals of the New York Academy of Sciences*, Vol. 929, pp. 221-252

APPENDIX A. FuzzyKAA System

This software tool is able to connect the user to any database and to assist him for linguistic values defining in that database's context. The system proposes a uniform partitioning of the attribute domain and then the definitions implicitly obtained can be adjusted either by changing numerical coordinates of graphical points, or by directly manipulating of them. Any fuzzy query can be evaluated accordingly to existing definitions. (see Tudorie, 2006a and Tudorie, 2006b).



APPENDIX B. MultiDef System

This software tool is able to connect several users (for example knowledge engineers) to the database, each of them having the possibility to define the fuzzy model for each linguistic value of the database attributes.

One defining process starts from an initial implicit model; the user may modify it according to his own semantic for the current linguistic term. An administrator, having his own interface, is monitoring and managing all this activity; he has at any moment a global view of all membership functions drawn by the users for the same linguistic terms, on the same attribute domain. (see Tudorie, 2006a and Tudorie, 2006b).