THE ANNALS OF "DUNAREA DE JOS" UNIVERSITY OF GALATI
FASCICLE III, 2009, Vol.32, No.2, ISSN 1221-454X
ELECTROTECHNICS, ELECTRONICS, AUTOMATIC CONTROL, INFORMATICS

INTELLIGENT INTERFACES FOR DATABASE FUZZY QUERYING

Cornelia TUDORIE

Department of Computer Science, University "Dunarea de Jos", Galati
2 Stiintei, 800146 Galati, Tel, Fax: 0236460182
email: Cornelia.Tudorie@ugal.ro

Abstract: Intelligent interfaces to database querying must offer to the user a flexible way to express the selection criteria. Some software tools are developed in our department for scientific purpose: to be used for studying flexible queries. There are two major aspects of the queries' flexibility that are interesting: natural language (Romanian) querying and vague querying[1] . They are studied and implemented either separately or both combined. For the natural language processing, the intermediary language approach is adopted. The queries vagueness is modelled and processed using the fuzzy logic technique. All presented software tools (and more others not presented here) are implemented in collaboration with our Computer Science students.

Keywords: Database, Intelligent Interface, Fuzzy Queries, Natural Language Interface

1. INTRODUCTION

The conventional database systems using query languages typically offer a mean to specify the selection criteria, as complex as it can be, very precise expressed, using Boolean expressions. The rigidity and specificity of the commonly used query languages can cause an empty result or a too complex one; in both cases the information is useless to the user.

A similar situation can be found when the domain of an attribute is very wide, the values are too varying and concrete, so the user has difficulties knowing or expressing precise criteria.

The solution would be accepting approximate or vague criteria in the search query; so only objects of a certain area of interest would be retrieved from the database. A natural consequence of accepting such type of criteria will be a result as a list of database objects, ordered by the grade of satisfaction of the original query criteria.

Generally following the idea to offer a much more natural access to a database, we set our point of view to the two aspects giving the flexibility of the interface:

- The possibility to express the database query using the natural language (Romanian language). The vocabulary that can be used by the operator will be necessarily restricted to those specific terms of the domain the database is operating. The natural language query is translated into an equivalent form, using the command language for databases (standard SQL may be chosen), and then processed by the database server.
- The possibility to include vague terms into the query, more exactly into the selection criterion of the database tuples. Those tuples will then be correctly interpreted, accordingly to their signification, and the satisfaction degree of the selection criterion will be computed for each of them.

[1] Most of the presented software tools are documented and interfaced in Romanian language.

This paper was recommended for publication by Severin Bumbaru

Some software tools are developed in our department for scientific purpose: to be used for studying flexible queries. There are two major aspects of the queries' flexibility that are interesting: natural language (Romanian) querying and vague querying. They are studied and implemented either separately or both combined. For the natural language processing, the intermediary language approach is adopted (Cristea, 1987). The queries vagueness is modelled and processed using the fuzzy logic technique.

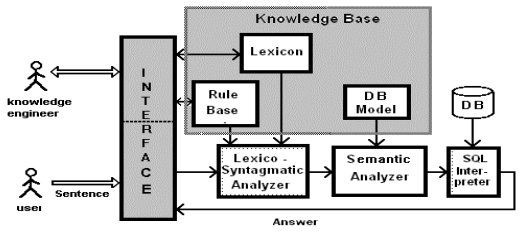
The presented intelligent systems are general enough to enable any time the connection to any database and/or to any (or associated, if case) knowledge base, new created or already used. The generality character of the interface; it will act as an intelligent system, able to be connected to any relational database; provided that a particular set of meta-knowledge has been prepared, specifically to the database and the linguistic context.

All presented software tools (and more others not presented here) are implemented in collaboration with our Computer Science students.

2. INTELLIGENT INTERFACE FOR DATABASE QUERYING IN ROMANIAN LANGUAGE (RoLQuery)[2]

Goal: A natural language query is analyzed and evaluated. The user request is understood and translated into the equivalent SQL query, sent and processed by the database system. The system's answer is presented to the user. (see more Cristea, 1987; Tudorie, Neacsu, Manolache, 2005; Tudorie, 2006a)

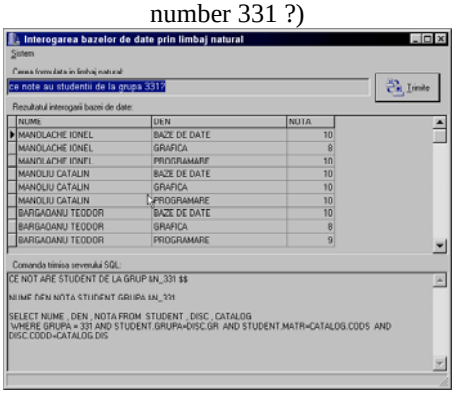
Architecture:



Example:

Ce note au studenții de la grupa 331?

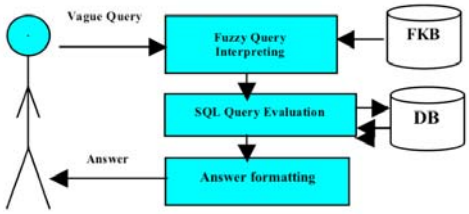
(What are the marks for the students in group number 331 ?)



3. INTELLIGENT INTERFACE FOR DATABASE VAGUE QUERYING (FuzzySQL)[3]

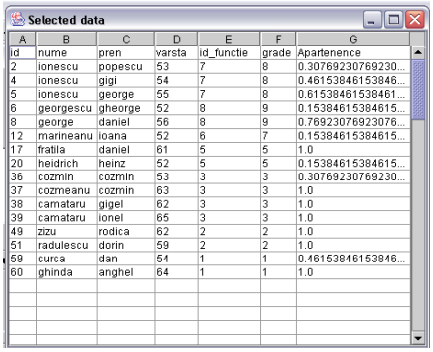
Goal: Through a graphical interface the user can ask the database including vague criteria. For each selected database row, a criterion satisfaction degree is computed accordingly to the fuzzy terms definitions (linguistic values, modifiers, numbers), stored in the knowledge base. (see more Dubois, Prade, 1996; Tudorie, 2003; Tudorie, 2006a)

Architecture:



Example:

Select * from angajat where varsta$foarte#mare



A	B	C	D	E	F	G
id	nume	pren	varsta	id_functie	grade	Apartenence
2	ionescu	popescu	53	7	8	0.307692307692307...
4	ionescu	gigi	54	7	8	0.461538461538461...
5	ionescu	george	55	7	8	0.615384615384615...
6	georgescu	gheorge	52	8	9	0.153846153846153...
8	george	daniel	56	8	9	0.769230769230769...
12	marineanu	ioana	52	6	7	0.153846153846153...
17	fratila	daniel	61	5	5	1.0
20	heidrich	heinz	52	5	5	0.153846153846153...
36	cozmin	cozmin	53	3	3	0.307692307692307...
37	cozmeanu	cozmin	63	3	3	1.0
38	camataru	gigel	62	3	3	1.0
39	camataru	ionel	65	3	3	1.0
49	zizu	rodica	62	2	2	1.0
51	radulescu	dorin	59	2	2	1.0
59	curca	dan	54	1	1	0.461538461538461...
60	ghinda	anghel	64	1	1	1.0

[2] In collaboration with Ionel Manolache (manolachei@yahoo.com)

[3] In collaboration with Gabriel Ion (ion_gabriel2000@yahoo.com)

4. INTERFACE FOR FUZZY KNOWLEDGE ACQUISITION AND FUZZY EXPRESSION EVALUATION (FuzzyKAEE)[4]

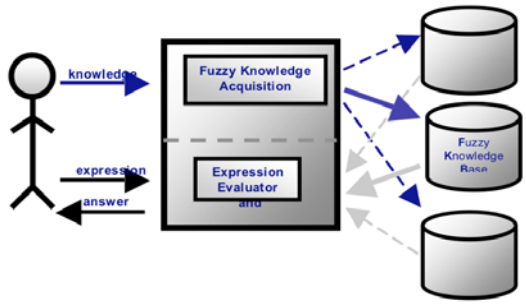
Goal: Two main utilities are possible:

- **Fuzzy knowledge acquisition**; it consists in vague terms (linguistic values, modifiers, numbers) defining as fuzzy sets, accordingly to the particular application domain (context).

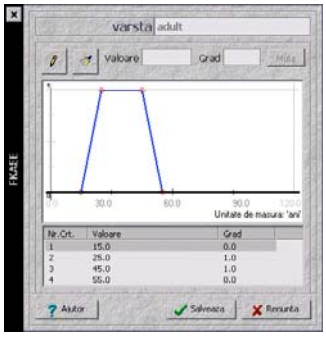
- **Fuzzy expression evaluation**; it consists in a graphical construct of expressions containing vague terms and their global satisfaction degree computing; the already defined fuzzy sets are taking into account.

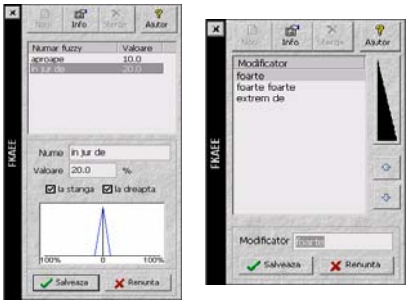
(see more Kacprzyk, Zadrozny, 2001; Tudorie, 2003; Tudorie, 2004a; Tudorie, 2006a; Yager, 1991)

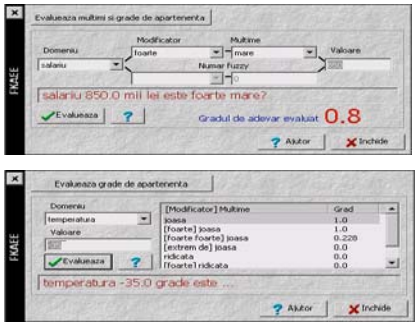
Architecture:



Example:







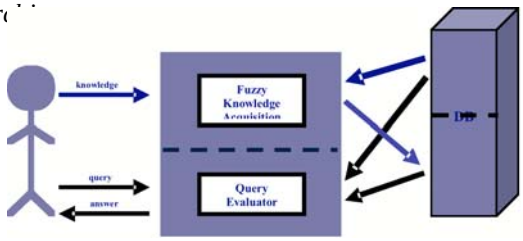
5. ASSISTED FUZZY KNOWLEDGE ACQUISITION FOR DATABASE VAGUE QUERYING (FuzzyKAA)[5]

Goal: Two main utilities are possible:

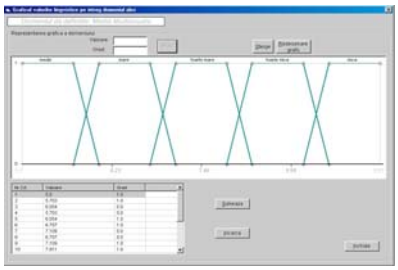
- **Fuzzy knowledge acquisition**; it consists in vague terms (linguistic values) defining as fuzzy sets, accordingly to the particular application domain (context) of each database. An algorithm for semi-automate definitions extracting is implemented.
- **Fuzzy query evaluation**; it consists in a graphical vague query construct and the global satisfaction degree computing for each selected database row; the already defined fuzzy sets are taking into account.

(see more Kacprzyk, Zadrozny, 2001; Tudorie, 2003; Tudorie, 2004a; Tudorie, 2006a)

Ar



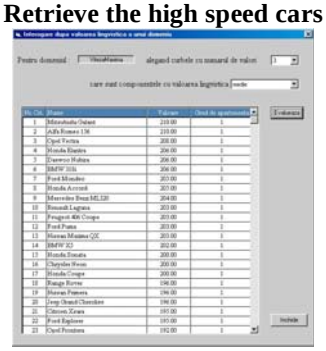
Example:



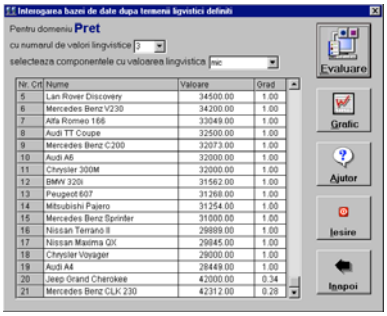
[4] In collaboration with Mirel Lazar (lt_mirel@yahoo.com)

[5] In collaboration with Sebastian Cretanu

Retrieve the high speed cars



Retrieve the *inexpensive* cars



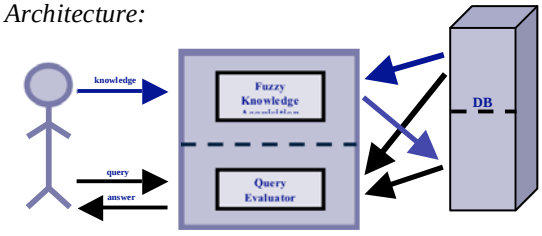
6. INTERFACE FOR FUZZY KNOWLEDGE ACQUISITION AND FUZZY DATABASE QUERYING (SAECFBD)[6]

Goal: Three main utilities are possible:

- **Fuzzy knowledge acquisition**; it consists in vague terms (linguistic values) defining as fuzzy sets, accordingly to the particular application domain (context) of each database.
- **Fuzzy query evaluation**; it consists in interpreting a simple query vagueness (absolute object qualification) and vague query evaluating by computing the criterion satisfaction degree for each selected database row; the already defined fuzzy sets are taking into account.
- **Complex fuzzy query evaluation**; it consists in interpreting and evaluating complex vague queries, containing two dependent criteria (relative object qualification); the already defined fuzzy sets are taking into account. The new operator WITHIN (Tudorie, 2006a; Tudorie, 2008) is implemented for the first time.

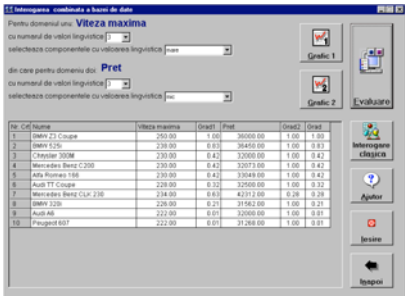
(see more Tudorie, 2004b; Tudorie, 2006a; Tudorie, 2006b)

Architecture:



Examples:

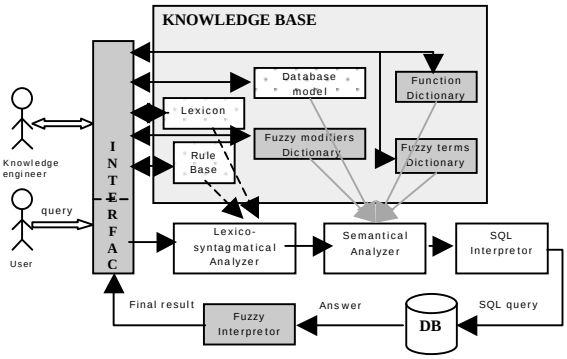
Retrieve the *inexpensive* cars <u>among</u> the *high speed* ones



7. INTELIGENT INTERFACE FOR DATABASE FUZZY QUERYING IN ROMANIAN LANGUAGE (RoFQuery)[7]

Goal: A natural language query is analyzed and evaluated. It may contain vague terms or not. It combines all utilities of RoLQuery and FuzzyKAA systems. (see more Cristea, 1987; Tudorie, Neacsu, Manolache, 2005; Tudorie, 2006a;)

Architecture:

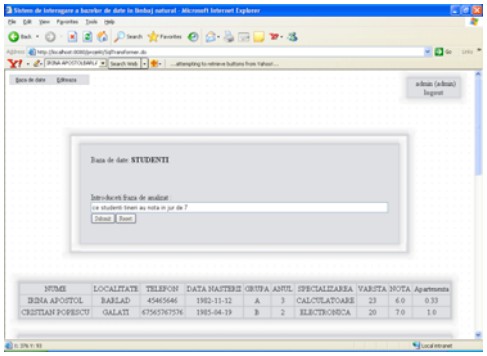


Example:

Ce studenți tineri au nota în jur de 7?
(Retrieve the young students having marks around 7)

[6] In collaboration with Laurentiu Fotache (f_laurentiu@yahoo.com)

[7] In collaboration with Cristian Neacsu (cristian.neacsu@gmail.com)



7. REFERENCES

Cristea, D. (1987). Sistemul QUERNAL, In C. Giumale, D. Preotescu, L.D. Serbanati, D. Tufis, D. Cristea (eds.), *LISP*, Editura Tehnica, Bucuresti, pp 215-229

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

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

Tudorie, C. (2003). Vague criteria in relational database queries, *Annals of "Dunarea de Jos" University of Galati*, III/2003, pp. 43-48

Tudorie, C. (2004a). Modeling Lingustic Modifiers from Romanian for Database Vague Querying, *Proceedings of 8th International Szmposium on Automatic Control and Computer Science, SACCS 2004*, Iaşi , pp. 63-67

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

Tudorie, C., Neacsu, C., Manolache, I. (2005). Fuzzy queries in Romanian language. An intelligent interface, *Annals of "Dunarea de Jos" University of Galati*, III/2005, Galati, Romania

Tudorie, C. (2006a). Contributii la realizarea de interfete pentru interogarea flexibila a bazelor de date (*Contributions to interfaces for database flexible querying*), *PhD Thesis*, "Dunarea de Jos" University of Galati, Romania

Tudorie, C., Bumbaru, S., Segal, C. (2006b). New Kind of Preference in Database Fuzzy Querying, *International Conference on Information Processing and Management of Uncertainty in Knowledge-based Systems, IPMU'06*, Paris, pp. 1389-1395

Tudorie, C. (2008). Qualifying Objects in Classical Relational Database Querying. In Galindo, J. (Ed.): *Handbook of Research on Fuzzy Information Processing in Databases*, Idea Group Publishing, Information Science Reference, Hershey, USA

Yager, R.R. (1991). Connectives and quantifiers in fuzzy sets, In *Fuzzy Sets and Systems*, 40-1, Elsevier Science, pp 39-75