

CRIS 2006
Current Research
Information Systems



Anne Gams Steine Asserson /
Eduard J. Simons (eds.)

**Enabling Interaction and Quality:
Beyond the Hanseatic League**

*8th International Conference on
Current Research Information Systems*

promoted by **euroCRIS**
Current Research Information Systems



Leuven University Press

Getting the knowledge to the agent – the Rough Sets approach

Sérgio Tenreiro de Magalhães, Leonel Duarte dos Santos, Luís Amaral

Department of Information Systems
University of Minho, Portugal

Abstract

For a query in a Research Information System (CRIS) to return adequate results, it is necessary that the system can “understand” the intention of the enquiring agent. One possible approach to guarantee the success of this communication is to create an intermediate module, responsible for the knowledge discovery processes, that can define concepts translatable in the languages used by the different agents involved in the use of a CRIS, enhance the queries, construct information over the available information and construct knowledge about the knowledge available in the CRIS and about its use. The Rough Set theory is a powerful tool that can set a path to achieve this goal. This paper describes in what way that is achievable, while describing the approach that is being followed by the portuguese CRIS, Degóis.

1 Introduction

According to euroCRIS [euroCRIS, 2005] a Current Research Information System (CRIS) is “any information tool dedicated to provide access to and disseminate research information”, therefore, it is necessary that the system is able to interact in a satisfactory way with those interested in becoming aware of research information. If not, the CRIS is not accomplishing the proposed objectives. Interacting is always dependent on agreeing on a language and that is the problem that we face when extracting, from the system, knowledge that meets the user’s requirements.

Human interaction is based on a natural language that is beyond the capability of having a conversation in a language. Therefore, more complex interacting systems are being developed to allow human-computer interaction to follow/integrate this human tendency to use tone of voice, face expressions and gestures to complement the available language [Buxton, 1990]. This is a problem that necessarily means dealing with incomplete and semi-structured data and the path presented in this paper can, one day, be extended to deal with those new paradigms in human-computer interaction. For the time being we need to deal with the incomplete and semi-structured data presented by users when naturally querying an information system using only typed words, this is, we need to agree on a language that can help the user to express what he is looking for, in a manner that is simple and consistent with the computer languages set (conditioned by the data-

model). But we need to agree on other languages too, for instance to communicate between the project module and the *curriculum vitae* module. Standardization creates a unified language but what about intercommunicating between different standards? The concepts are not equal but they are almost certainly similar. So, we need to change our vision of communication. Traditionally we see a conversation as the use, under certain rules, of a subset of objects, like words, familiar to those having the conversation (in the sense that they belong to their language set). In this paper we'll argue in favor of a different paradigm where, in a reasonable way, we approach the known languages by recognizing similarities. This can replace the traditional logical approaches, where a query for two words returns the documents that include the first and the second words, or the traditional artificial intelligence processes where an expansion is associated to each individual term [De Cock, 2005], for instance by adding synonyms.

2 The Rough Sets theory

Z. Pawlak [Pawlak] introduced the Rough Sets theory in the 1980's and its applications are becoming popular in many different fields, from electronic commerce data mining [Wang, 2004] to biometric authentication [Magalhaes, 2005]. The main idea is to work with the uncertainty created by undefined sets, where some objects can belong to the set in some occasions and not belong in others.

In Rough Sets theory two main areas are defined: i) the lower approximation, that defines the objects that we are certain that belong to the set; and ii) the upper approximation, that defines the set where we are certain that no object that should belong to the set is left out. This creates a boundary region where we must, and we can, deal with uncertainty.

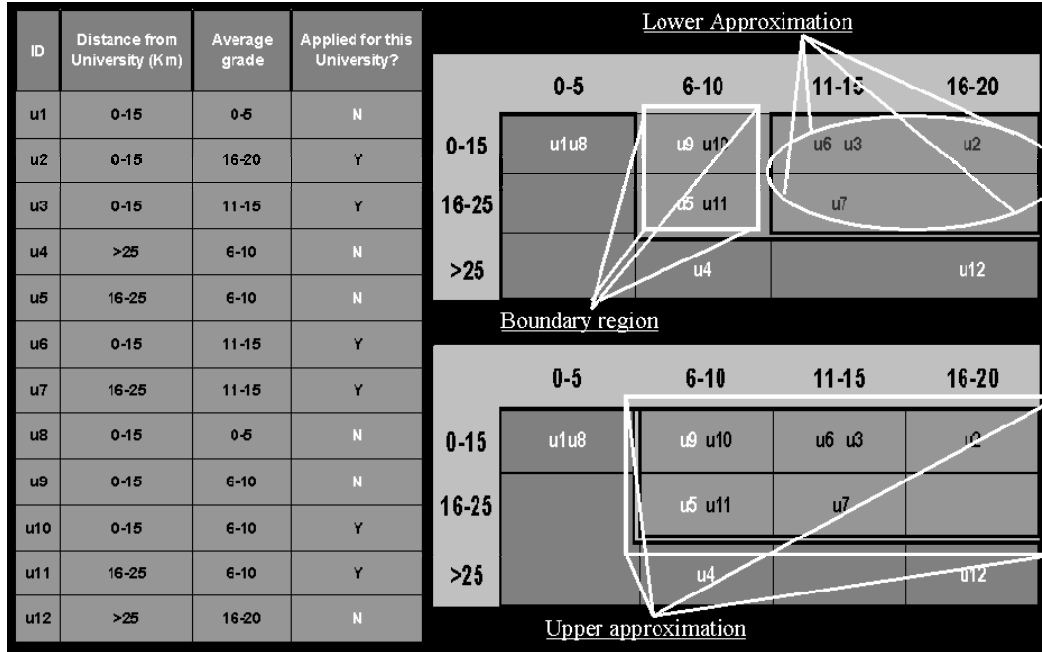


Figure 1 – The Rough Sets view of a decision table with two attributes and a decision

Figure 1 shows the lower and upper approximations as well as the boundary region generated by a table containing data corresponding to several data on senior high school students applying to universities. In this example the decision attribute is the information relative to their choice of a specific university (Y), or not (N). Once there is a boundary region, we can't make a decision on some students choice based on the two values provided (home distance to the university and average grade), so we say they are indiscernible. In our example, a student living 21Km away from the university and with an average of 9 may apply to our university or not, this is, it can belong to the "yes" class or not, it "roughly" belongs to that class (upper approximation). In our example it is simple to conclude a rule (if *distance*<25 and *grade*>10, then *decision*=YES) and several possible rules that include the boundary region or part of it.

Another important capability of Rough Sets is to define reducts on the attributes that will preserve the information but will simplify the problem. Reducts aim to find the minimum amount of information to:

- Discern one object from all other objects;
- Discern all object from each other;
- Determine the outcome of a particular object; or
- Determine the outcome of all objects (Ohrn, 2000).

In complex decision tables, reducts often allow some columns/attributes to be excluded, therefore simplifying the problem without compromising the performance.

This theory is effective, among others applications, in approximating concepts, identifying attribute dependencies, reducing the problem size, constructing decision rules, knowledge discovery and representation, model fusion and in approximate reasoning under uncertainty [Son, 2005]. All of these problems exist in the process of extracting the knowledge from a CRIS and, therefore, it is only natural to use this technology to increase the usability of those systems.

3 The Rough Sets approach

Rough Sets have already been used to improve the quality of the results of a query expansion in the Internet [De Cock, 2005], a similar of those related to querying a CRIS. Our approach takes into account the existing works on query expansion, including the Rough Sets approach as well as others, but tries to go beyond the understanding of the simple universal relation between terms. We aim to define a way for establishing a relation between the CRIS's language (the known objects) and the users (other CRIS included) language.

In an intuitive manner, we can define certain concepts, recognize their intersection and understand their differences. Figure 2 shows some of the relations concerning the concept "JAVA". This scheme is relatively easy to construct in our mind and we easily see that when talking about JAVA, INDONESIA and SUMATRA, we are talking about islands. But the relations are not easy to define when we are dealing with the universe of all the possible words.

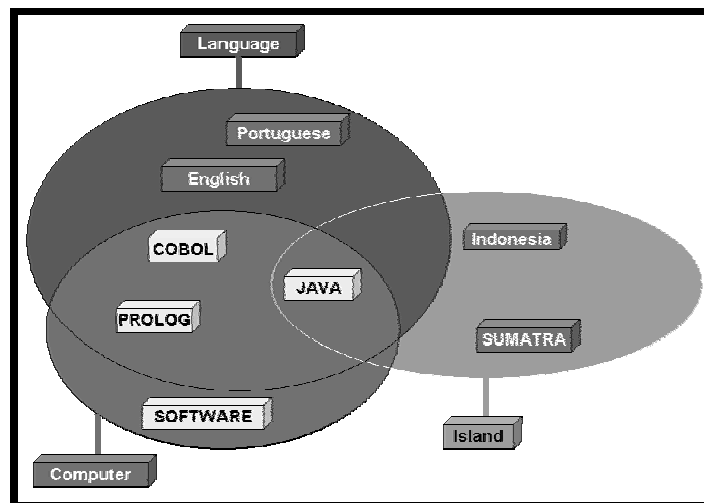


Figure 2 – Some concepts related and/or including the concept "JAVA".

Once we aim a system that is simple to use and we want to use it to extract knowledge from the existing CRIS and of its use, we need an external knowledge discovery module (EKDM) that can, in a transparent way, be the moderator of the communication, the translator. This module must create, along time, its own main concepts that can be expressed in both the users and the CRIS languages. These concepts are, sometimes, dependent on other simpler concepts that the module has already clarified. In this way, a user's query goes through several steps, described in the next sections. Considering the focus of this paper, we'll explore only those steps in which Rough Sets bring new approaches and new results.

3.1. Preparation of the original query

The first step to process the user's query is to translate the terms into a form known by the EKDM's language. This is achieved by using text-mining techniques destined to clean strings and prepare a query. In this stage words too common like "and" or "the" are eliminated, while other words are substituted by a synonym existent in the ontology, a formal definition of entities and their properties, interactions, behaviors and constraints [Reynolds, 2002]. For instance, the term "plants" would be substituted by "plant".

3.2. Alteration/optimization of the query

De Cock [De Cock, 2005] presented a method to query the Internet using a fuzzy Rough Sets system, where the terms are first expanded to the upper approximation, this is to a set of terms that are related to one of them, and then reduced in order to obtain an optimized expanded query that corresponds to the needs of the inquirer without losing any key concepts and without creating redundancy. But a CRIS has particularities that require the change of some formulas used. Given the complexity of the tasks we'll present the original method, the altered method and the consequences of those changes, through functional examples.

3.2.1. De Cock's Alteration/optimization of the query

The base for the optimization of the query is the thesaurus that will represent the relations in the user's language, so it is mandatory that we find ways to create it in a manner that is, in fact, representative. Given the amount of information provided by the Internet, De Cock used it to create the thesaurus by, for each two terms, t_1 and t_2 , counting the number of occurrences of each one of them and of " t_1 AND t_2 ", these values, D_{t_1} , D_{t_2} and $D_{t_1 \cap t_2}$ respectively (Table 1), are

calculated using the search engine GOOGLE. The values returned by $\frac{D_{t_1 \cap t_2}}{\min(D_{t_1}, D_{t_2})}$ (Table 2) are then normalized (Table 3) by the S function (Figure 3). Finally, those relations that obtain a value of 50% or more, and only those, are considered as existing (Table 4).

	Portuguese	English	Language	COBOL	PROLOG	JAVA	Indonesia	Sumatra	Software	Island	Computer
Portuguese	85.100.000	63.400.000	34.500.000	101.000	118.000	2.510.000	4.840.000	197.000	15.600.000	6.820.000	10.800.000
English		1.480.000.000	429.000.000	999.000	1.140.000	30.900.000	99.500.000	1.450.000	247.000.000	70.300.000	211.000.000
Language			1.140.000.000	1.920.000	4.240.000	71.400.000	21.700.000	620.000	331.000	53.100.000	243.000.000
COBOL				4.620.000	709.000	2.360.000	132.000	806	2.510.000	267.000	1.940.000
PROLOG					4.270.000	2.900.000	88.100	567	1.910.000	205.000	2.120.000
JAVA						348.000.000	4.340.000	2.350.000	174.000.000	6.780.000	73.400.000
Indonesia							219.000.000	1.840.000	23.200.000	59.500.000	21.800.000
Sumatra								4.130.000	560.000	2.020.000	608.000
Software									2.450.000.000	50.200.000	707.000.000
Island										414.000.000	59.200.000
Computer											1.650.000.000

Table 1 – Number of simultaneous occurrences in Google.

	Portuguese	English	Language	COBOL	PROLOG	JAVA	Indonesia	Sumatra	Software	Island	Computer
Portuguese	100,00%	74,50%	40,54%	2,19%	2,76%	2,95%	5,69%	4,77%	18,33%	8,01%	12,69%
English		100,00%	37,63%	21,62%	26,70%	8,88%	45,43%	35,11%	16,69%	16,98%	14,26%
Language			100,00%	41,56%	99,30%	20,52%	9,91%	15,01%	0,03%	12,83%	21,32%
COBOL				100,00%	16,60%	51,08%	2,86%	0,02%	54,33%	5,78%	41,99%
PROLOG					100,00%	67,92%	2,06%	0,01%	44,73%	4,80%	49,65%
JAVA						100,00%	1,98%	56,90%	50,00%	1,95%	21,09%
Indonesia							100,00%	44,55%	10,59%	27,17%	9,95%
Sumatra								100,00%	13,56%	48,91%	14,72%
Software									100,00%	12,13%	42,85%
Island										100,00%	14,30%
Computer											100,00%

Table 2 – Calculation of the level of relation between the terms (first stage).

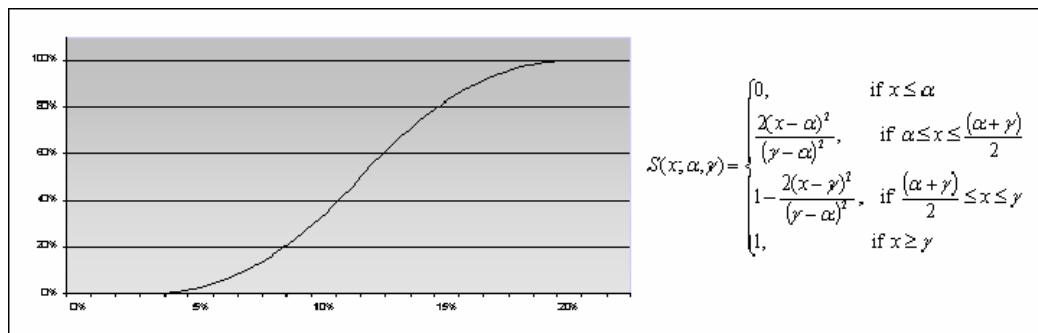


Figure 3 – The S function, used to normalize the values obtained in the first stage

	Portuguese	English	Language	COBOL	PROLOG	JAVA	Indonesia	Sumatra	Software	Island	Computer
Portuguese	100,00%	100,00%	100,00%	0,00%	0,00%	0,00%	5,00%	2,17%	98,07%	17,40%	63,03%
English		100,00%	100,00%	100,00%	100,00%	23,92%	100,00%	100,00%	92,41%	93,69%	77,17%
Language			100,00%	100,00%	100,00%	100,00%	33,03%	82,78%	0,00%	64,38%	100,00%
COBOL				100,00%	92,02%	100,00%	0,00%	0,00%	100,00%	5,35%	100,00%
PROLOG					100,00%	100,00%	0,00%	0,00%	100,00%	2,24%	100,00%
JAVA						100,00%	0,00%	100,00%	100,00%	0,00%	100,00%
Indonesia							100,00%	100,00%	39,91%	100,00%	33,47%
Sumatra								100,00%	71,29%	100,00%	80,72%
Software									100,00%	57,09%	100,00%
Island										100,00%	77,51%
Computer											100,00%

Table 3 – Normalized values of the level of relation between the terms (second stage) obtained through the *S* function.

	Portuguese	English	Language	COBOL	PROLOG	JAVA	Indonesia	Sumatra	Software	Island	Computer
Portuguese	•	•	•						•		•
English		•	•	•	•		•	•	•	•	•
Language			•	•	•	•		•		•	•
COBOL				•	•	•			•		•
PROLOG					•	•			•		•
JAVA						•	•	•	•	•	•
Indonesia							•	•		•	
Sumatra								•	•	•	•
Software									•	•	•
Island										•	•
Computer											•

Table 4 – Two terms are related if the correspondent relation value is at least 50%

Given a user's prepared query, we first obtain an expended query by joining to the concepts used, the other concepts of the user's language that are related to, at least, one of them. In the Table 4 we can verify some of those relations, for instance the term "JAVA" is related to the terms "language", "COBOL", "PROLOG", "Sumatra", "Software", "Computer" and, of course, with it self. The table doesn't include all the concepts of the user's language, so it doesn't present all the relations. This expansion is what can be considered, under the Rough Sets theory, the upper approximation. Then we create a *tight upper approximation* by eliminating all the concepts that are related with other concepts not included in the upper approximation. This will define which part of the boundary region will be included in our final query.

Figure 4 presents two examples of user's prepared queries: "JAVA and SOFTWARE" and "JAVA and PROLOG" and how they would be optimized if the user's language add only the words used in our example of the Table 1, 2 and 3. So, the result is not valid, only the

methodology, once other concepts may be included while some can still be excluded. For instance, “JAVA” can also be related (at least in the intermediate stage) to “coffee” because of that special brand of coffee with that name, and “portuguese” may be eliminated from the final query because of its relation with other concepts not included in the example tables like, for instance, “wine”. Even so, we can verify that there is an inclusion of new relevant concepts in the queries, but also that some can be lost, like the concept software in the first example. So, we need to improve this methodology and we’ll do so by adapting it o the specificities of a CRIS.

Query: JAVA and the software				Query: JAVA and PROLOG			
concepts	Prepared query	upper approximation	tight upper approximation	concepts	Prepared query	upper approximation	tight upper approximation
Portuguese		●	●	Portuguese			
English		●		English		●	
Language		●		Language		●	
COBOL		●	●	COBOL		●	●
PROLOG		●	●	PROLOG	●	●	●
JAVA	●	●	●	JAVA	●	●	●
Indonesia				Indonesia			
Sumatra		●		Sumatra		●	
Software	●	●		Software		●	
Island				Island			
Computer		●	●	Computer		●	

Figure 4 – Example of the generation of the optimised query. The final result is not relevant once the generating rules were applied to a subset of the language.

3.2.2. Alteration/optimization of the query in a CRIS

The simultaneous occurrence of two concepts in a *Curriculum Vitae* included in a CRIS has different meanings if they are found in the some database field or in different ones, so we need to include this concept in the construction of our thesaurus.

Lets consider the number of occurrences in the n fields of a CRIS of two concepts, t_1 and t_2 , and the correspondent relative frequencies for each one of the several fields, F_{t1-j} , and F_{t2-j} for $j=1,2...n$. In the Table 5 we have an example of that counting for seven concepts in three fields, originating the frequencies expressed in Table 6. We define the distance of the terms in the CRIS,

d , through the *Euclidian Distance* as $\sum_{j=1}^n \sqrt{(F_{t1-j} - F_{t2-j})^2}$ and the proximity, p , as $1 - d$, in our

example the distance is calculated in the Table 7. This factor p will be used to weigh the previous formula that establishes the relation between the terms. So, the two terms are related if and only

if: $S \left[\frac{D_{t_1 \cap t_2}}{\min(D_{t_1}, D_{t_2})} \left(1 - \sum_{j=1}^n \sqrt{(F_{t1-j} - F_{t2-j})^2} \right) \right] \geq 50\%$, where S stands for the normalizing

function presented in Figure 3. The values of the discriminating function obtained for the concepts and the corresponding relations are presented in the Table 8 and Table 9, respectively.

	Title_Of_Production	Name_Of_Project	Keywords	SUM
PROLOG	12	0	0	12
JAVA	14	0	0	14
INDONESIA	1	0	0	1
SOFTWARE	73	1	5	79
ILHA	130	3	0	133
SUMATRA	1	0	0	1
COMPUTADOR	73	0	2	75

Table 5 – Occurrence of several terms in three different fields of the portuguese CRIS, Degóis.

	Title_Of_Production	Name_Of_Project	Keywords
PROLOG	100,00%	0,00%	0,00%
JAVA	100,00%	0,00%	0,00%
INDONESIA	100,00%	0,00%	0,00%
SOFTWARE	92,41%	1,27%	6,33%
ILHA	97,74%	2,26%	0,00%
SUMATRA	100,00%	0,00%	0,00%
COMPUTADOR	97,33%	0,00%	2,67%

Table 6 – Relation between the terms and the fields.

distance	PROLOG	JAVA	Indonesia	Software	Island	Sumatra	Computer
PROLOG	0,00%	0,00%	0,00%	9,97%	3,19%	0,00%	3,77%
JAVA		0,00%	0,00%	9,97%	3,19%	0,00%	3,77%
Indonesia			0,00%	9,97%	3,19%	0,00%	3,77%
Software				0,00%	8,34%	9,97%	6,27%
Island					0,00%	3,19%	3,52%
Sumatra						0,00%	3,77%
Computer							0,00%

Table 7 – Distance between the terms in Degóis

	PROLOG	JAVA	Indonesia	Software	Island	Sumatra	Computer
PROLOG	100,00%	100,00%	0,00%	100,00%	1,88%	0,00%	100,00%
JAVA		100,00%	0,00%	100,00%	0,00%	100,00%	100,00%
Indonesia			100,00%	29,58%	100,00%	100,00%	29,95%
Software				100,00%	45,57%	57,98%	100,00%
Island					100,00%	100,00%	73,37%
Sumatra						100,00%	76,45%
Computer							100,00%

Table 8 – Normalized values of the proximity of the terms (using the S function) in Degóis

	PROLOG	JAVA	Indonesia	Software	Island	Sumatra	Computer
PROLOG	●	●		●			●
JAVA		●		●		●	●
Indonesia			●		●	●	
Software				●	●	●	●
Island					●	●	●
Sumatra						●	●
Computer							●

Table 9 – New relation table, specific to the used CRIS (Degóis). Now, there is no relation between “Software” and “Island”.

We can observe in the Table 9 that there is no longer a relation between the concepts “software” and “island” and the consequence of that change is a difference in the final query. Figure 5 shows the calculations of Figure 4 made without a relation between those two concepts and we can verify that, in the first example, the term “software” is brought back into the query. Again, this is only indicative of the power of the methodology and the resulting query is not the real result once we are not working with the complete set of concepts.

Query: JAVA and the software				Query: JAVA and PROLOG			
concepts	Prepared query	upper approximation	tight upper approximation	concepts	Prepared query	upper approximation	tight upper approximation
Portuguese		●	●	Portuguese			
English		●		English		●	
Language		●		Language		●	
COBOL		●	●	COBOL		●	●
PROLOG		●	●	PROLOG	●	●	●
JAVA	●	●	●	JAVA	●	●	●
Indonesia				Indonesia			
Sumatra		●		Sumatra		●	
Software	●	●	●	Software		●	
Island				Island			
Computer		●	●	Computer		●	

Figure 5 – The break in the relation between the terms “Software” and “Island” would include the term “software” in the optimised version of our first query.

3.3. Fitting the contents to the results

We have obtained a query that corresponds to the needs of the inquirer without losing any important concepts and without creating redundancy. Now, we can conclude the translation of the user’s query to the CRIS language.

Each obtained query results in several documents that include the terms in several different fields. For instance, we can find a document with the term “JAVA” in the keywords and “island” in the address, or with “JAVA” and “island” both in the keywords field. The relevance of those two documents is certainly different. The Rough Sets theory can help us to define the degree of relevance induced by each field, either globally or for a particular set of concepts. For that, the

EKDM must keep the records of all the documents retrieved by the queries and consider the users choices of documents as an expert judgment that will constitute the decision attribute that will state if a document should be retrieved or not. This will allow the constitution of rules, under the Rough Sets theory, that will improve future answer to queries.

The set constituted by the terms of the final query and by the rules that apply to them , stating the combination of fields that is more relevant for those concepts, constitutes the translation of the user's query to the CRIS language.

4 Conclusion

In conclusion, this paper will show that Rough Sets are a valid approach to knowledge discovery in CRIS and demonstrates the path to implement it by describing the EKDM and some translating processes that are based on an existing fuzzy Rough Sets thesaurus creation method. By adapting the processes to the particular characteristics of a CRIS, through the inclusion of some specific parameters, we enhance the results.

The information retrieval, while aiming to satisfy the user's needs, also provides information that is used to feed the EKDM in order to generate dependencies between terms and to create more basic and complex concepts that will increase the existing knowledge and improve the quality of the information transmitted to the enquirer. In this way we'll be able to obtain information on the available information and construct knowledge about the knowledge available in the CRIS and about its use.

5 Contact information

Sérgio Tenreiro de Magalhães, Leonel Duarte dos Santos, Luís Amaral
Departamento de Sistemas de Informação
Universidade do Minho
4800-058 Guimarães
Portugal

e-mail: {psmagalhaes, leonel, amaral@dsi.uminho.pt}

6 Bibliography

euroCRIS (2005), www.eurocris.org, [November, 2005]

Buxton, W. (1990). "The Natural Language of Interaction: A Perspective on Non-Verbal Dialogues". In Laurel, B. (Ed.). *The Art of Human-Computer Interface Design*, Reading, MA: Addison-Wesley. 405-416.

- Pawlak, Z. (1991). "Rough Sets: Theoretical Aspects of Reasoning About Data". Kluwer Academic Publishers, Dordrecht.
- Wang, X. Xu, R and Wang, W. (2004). "Rough Set Theory: Application in Electronic Commerce Data Mining," in, pp. 541-544, Proceedings of the IEEE/WIC/ACM International Conference on Web Intelligence (WI'04). 541 – 544
- Revett, K. Magalhães, S. T., Santos, H.D. (2005). "Developing a Keystroke Dynamics Based Agent Using Rough Sets". International Workshop on Rough Sets and Soft Computing in Intelligent Agent and Web Technology. Compiègne University of Technology. 56-61.
- Son, N. H. (2005), "Rough Set Approach to Learning in MAS". UTC Press, France.
- De Cock, M. and Cornelis, C. (2005). "Fuzzy Rough Set Based Web Query Expansion". International Workshop on Rough Sets and Soft Computing in Intelligent Agent and Web Technology, pp. 9-16, Compiègne University of Technology, France.
- R.G. Reynolds, A. Lazar, (2002): "Computational Framework for Modeling the Dynamic Evolution of Large-scale Multi-agent Organizations", at SPIE's 16th Annual International Symposium on Aerospace/Defense Sensing, Simulation, and Controls, 1-5 April 2002, Orlando, Florida, USA.
- Ohrn, A. and Rowland, T. (2000): "Rough Sets: A Knowledge Discovery Technique for Multifactorial Medical Outcomes", American Journal of Physical Medicine & Rehabilitation. 79(1):100-108, January/February 2000