

Queries for Detailed Information System Selection

Agnieszka Dardzinska

Bialystok University of Technology
 Department on Mechanics and Computer Science
 ul. Wiejska 45c, 15-351 Bialystok, Poland
 Email: a.dardzinska@pb.edu.pl

Anna Romaniuk

Bialystok University of Technology
 Department on Mechanics and Computer Science
 ul. Wiejska 45c, 15-351 Bialystok, Poland
 Email: a.romaniuk@doktoranci.pb.edu.pl

Abstract—In this paper we assume there is a group of connected information systems forming distributed information system (DS). They work under the same ontology. At the same time, each information system has its own knowledge base. Values of attributes in each information system S form atomic expressions of a language used for communication with others. Collaboration among them is initiated when one of information system S is asked by user to resolve a query containing some nonlocal attributes for S . Therefore it has to contact other information systems to obtain additional, helpful knowledge for finding finally objects satisfying given query. Because there is a set of different information systems connected with a given one, we have to decide which of them is the closest with its knowledge, and which one should be selected by user, for further investigation.

I. INTRODUCTION

IN this paper we assume that there is a group of collaborating incomplete information systems, which are connected with a Query Answering System (QAS) and a knowledge base (K), empty at the beginning. By incomplete information system we mean an information system, where attribute values are completely unknown or are connected with corresponding weights [4]. From the definition [3], [4] all the weights for each value of attribute has to sum up to 1. The definition of an information system of type λ presented in this work was initially proposed in [11]. The type λ was presented with a purpose to check all the weights assigned to all values of attributes using Chase algorithm [4], [11]. If a weight is lower than minimal threshold value λ , the corresponding attribute value has to be eliminated. All remaining weights assigned to the rest of attribute values are equally adjusted so their sum is again equal 1. In information systems we take into consideration, we force many semantic inconsistencies. It is related to various interpretations of attributes and their values among sites. In some sites the concept *healthy* can be defined and described completely in a different way than in others. Moreover, in some sites can handle hidden or missing values of attribute *healthy* can be interpreted in a different way.

Quite common method for finding set of objects satisfying the given query from QAS is to replace null values using suggested values obtained from some statistical or rule-based methods. Ontologies [1], [2], [7], [8], [9], [14], [15], [16], [17] are widely used as a part of semantical connection between information systems built independently so they can

understand and collaborate with each other. In [11], the notion of the rough semantics which is optimal and a method of its construction was presented. The rough semantics can be easily used to model and handle semantic inconsistencies among sites due to interpretations of incomplete values. As the result of collaborations among information systems, a knowledge base of any system fills up continuously and contains knowledge in form of rules extracted from all information systems [6]. As we mentioned earlier, the names of attributes can be the same among different information systems, but at the same time their granularity levels may differ. Therefore the knowledge base has to satisfy certain properties in order to be used by Chase[11]. We will show that while solving a query, it is worth first to use the knowledge extracted from systems which are semantically close to the given information system.

II. INCOMPLETE INFORMATION SYSTEM

Working with information systems in real world, most of data are collected and stored in different independent locations, connected with each other. They form distributed information systems. Therefore it is very possible that some attributes can be missing in some systems and can occur in others.

We say that information system $S(A) = (X, A, V)$ is incomplete and of type λ , if $S(A)$ is an incomplete information system defined by Pawlak in [10] and these conditions hold:

- X is a set of objects, A is a set of attributes, and $V = \cup\{V_a : a \in A\}$ is a set of all values of attributes
- $(\forall x \in X)(\forall a \in A)[a_S(x) \in V_a \text{ or } a_S = \{(a_i, p_i) : a_i \in V_a \wedge p_i \in \langle 0, 1 \rangle \wedge 1 \leq i \leq m\}]$
- $(\forall x \in X)(\forall a \in A)[(a_S(x) = \{(a_i, p_i) : 1 \leq i \leq m\}) \Rightarrow \sum_i p_i = 1]$
- $(\forall x \in X)(\forall a \in A)[(a_S(x) = \{(a_i, p_i) : 1 \leq i \leq m\}) \Rightarrow (p_i \geq \lambda)]$.

Now, let us assume that $(S_i(A), K)$ for any $i \in I$ are i different incomplete information systems of type λ , and the knowledge base K (empty at the beginning). These systems are represented by the same set of attributes A . The meaning and granularity level of values of attributes from A in information systems S_i is also the same. All information systems $(S_i(A), K)$ communicate with each other, setting up distributed information system (DIS). We assume that, if $a \in A_i \cap A_j$, then the granularity levels of attribute a in two independent information systems S_i and S_j may differ but the meaning of this attribute in them remains the same.

This work was supported by MB/WM/6/2015

The user submits a query q to the information system S . We extract rules which are in form of a set $R = \bigcup L(R_i)$. They can be later used by any Chase algorithm [4], [5], associated with any of the sites of DIS . Set $L(R_i)$ contains all rules extracted from system S_i . If we want Chase algorithm to be applicable to S , it has to be strictly connected with rules from R which satisfy the following conditions:

- attribute value used in the decision part of a rule from set of rules R has the granularity level either equal to or finer than the granularity level of the corresponding attribute in S .
- attribute used in the decision part of a rule from set of rules R either does not belong to A or is incomplete in S .
- the granularity level of any attribute used in the classification part of a rule from set of rules R is either equal or softer than the granularity level of the corresponding attribute in S .

If we will find a match between the attribute value in information system and the attribute value used in a description of a rule, then two options can be further taken into consideration:

- attribute a involved in matching is the classification attribute in rule. If two attribute values have different granularity level, then the value of attribute a has to be replaced by a finer value (its granularity has to match the granularity of a in S).
- attribute d involved in matching is the decision attribute in rule. If two attribute values have different granularity level, then the decision value d has to be replaced by a softer value (its granularity has to match the granularity of d in S).

In this paper we also take into consideration two information systems and assume that:

Information system S_1 can be transformed into S_2 by containment mapping Ψ ($\Psi(S_1) = S_2$) if following conditions hold:

- $(\forall x \in X)(\forall a \in A)[card(a_{S_1}(x)) \geq card(a_{S_2}(x))]$
- $(\forall x \in X)(\forall a \in A)[card(a_{S_1}(x)) = card(a_{S_2}(x)) \Rightarrow [\sum_{i \neq j} |p_{2i} - p_{2j}| > \sum_{i \neq j} |p_{1i} - p_{1j}|]]$

So, if containment mapping Ψ converts an information system S_1 to S_2 , then we can say S_2 is more complete than S_1 . It means that for a minimum one pair $(a, x) \in A \times X$, either function Ψ has to decrease the number of attribute values in $a_{S_1}(x)$ or the average difference between any two confidences assigned to each attribute value in $a_{S_2}(x)$ has to be increased by this function.

Let us take two information systems S_1, S_2 both of the type λ , represented as Table 1 and Table 2.

It can be easily noticed that some values assigned to objects in both information systems S_1 and S_2 are different. For example values $a(x_3), a(x_5), a(x_8), b(x_2), c(x_2), c(x_7), e(x_1), e(x_4)$. In each of these cases, an attribute value assigned to an object in S_2 is less general than the value assigned to the same object in S_1 . It means that Ψ is a correct containment mapping ($\Psi(S_1) = S_2$).

III. QUERY PROCESSING BASED ON CHASE AND COLLABORATION BETWEEN SYSTEMS

Assume that user submits a query $q(B)$ to one of the information system (S, K) , which cannot be answered directly

as some of the attributes used in a given query are unknown in the primary information system. Assume, we have a group of collaborating information systems working under the same ontology, and user resubmits a query $q(B)$ to other information systems $(S(A), K)$, where $S(A) = (X, A, V)$, $K = \emptyset$, B are the attributes used in $q(B)$, and $A \cap B \neq \emptyset$. Since $(S(A), K)$ can collaborate with itself and other information systems, definitions of hidden or missed attributes for $(S(A), K)$ can be extracted from this information system or from other information systems. It was shown [10] that $(S(A), K)$ can answer the query $q(B)$ assuming that definitions of all values of foreign attributes can be extracted locally or globally and used to finding answer for $q(B)$. Assume now that we have three collaborating information systems with knowledge bases connected with them: $(S, K), (S_1, K_1), (S_2, K_2)$, where $S = (X, A, V)$, $S_1 = (X_1, A_1, V_1)$, $S_2 = (X_2, A_2, V_2)$, and knowledge bases are initially empty $K = K_1 = K_2 = \emptyset$. If the consensus between (S, K) and (S_1, K_1) on the knowledge extracted from $S(A \cap A_1)$ and $S_1(A \cap A_1)$ is closer than the consensus between (S, K) and (S_2, K_2) on the knowledge extracted from $S_2(A \cap A_2)$ and $S_2(A \cap A_2)$, then (S, K) chooses (S_1, K_1) as the detailed information system, more helpful in solving user's queries. All the rules defining unknown attribute values for S are then extracted at S_1 and placed in corresponding K . The problem here is with the predicted values: are they really correctly extracted, and if not, how close they are to the correct values? How can we test this? The classical approach works as follows. First we start with a complete information system, from which we can extract knowledge in form of rules. Then we remove randomly some percent of its values and try to resolve new information system using one of the imputation algorithms. Next we make comparison of the descriptions of objects in both systems: the original one and the system which is the outcome of the imputation algorithm.

Before we will go deeper in the analyzing, we have to make the interpretation of two main functors *and* and *or*, denoted in this paper by $*$ and $+$ correspondingly. We can use the semantics of terms proposed in [12] since it preserves distributive property:

For any queries t_1, t_2, t_3 we have $t_1 * (t_2 + t_3) = (t_1 * t_2) + (t_1 * t_3)$. If we consider that $S = (X, A, V)$ is an information system of type λ and t is a term constructed in a standard way from constant values of attributes in V and two functors " $*$ " and " $+$ ", then by the standard interpretation of a term t in S we mean $N_S(t)$ defined as [12]:

- $N_S(v) = \{(x, p) : (v, p) \in a(x)\}$ for any $v \in V_a$
- $N_S(t_1 * t_2) = N_S(t_1) \otimes N_S(t_2)$
- $N_S(t_1 + t_2) = N_S(t_1) \oplus N_S(t_2)$

where

- $N_S(t_1) \otimes N_S(t_2) = \{(x_i, p_i \cdot q_i)_{i \in I \cap J}\}$
- $N_S(t_1) \oplus N_S(t_2) = \{(x_i, p_i)_{i \in I \setminus J}\} \cup \{(x_j, q_j)_{j \in J \setminus I}\} \cup \{(x_i, \max(p_i, q_i))_{i \in I \cap J}\}$

Assume that (S, K) is an information system, where $S = (X, A, V)$ and K contains definitions of attribute values in

TABLE I
INFORMATION SYSTEM S_1

X	a	b	c	d	e
x_1	$\{(a_1, \frac{2}{3}), (a_2, \frac{1}{3})\}$	$\{(b_1, \frac{1}{4}), (b_2, \frac{3}{4})\}$	c_1	d_1	$\{(e_1, \frac{1}{2}), (e_2, \frac{1}{2})\}$
x_2	$\{(a_2, \frac{1}{4}), (a_3, \frac{3}{4})\}$	$\{(b_1, \frac{1}{3}), (b_2, \frac{2}{3})\}$		d_1	e_1
x_3		b_2	$\{(c_1, \frac{1}{2}), (c_3, \frac{1}{2})\}$	d_2	e_2
x_4	a_3		c_2	d_1	$\{(e_1, \frac{2}{3}), (e_2, \frac{1}{3})\}$
x_5	$\{(a_1, \frac{2}{3}), (a_2, \frac{1}{3})\}$	b_1	c_2		e_1
x_6	a_2	b_2	c_3	d_2	$\{(e_2, \frac{1}{3}), (e_3, \frac{2}{3})\}$
x_7	a_2	$\{(b_1, \frac{1}{4}), (b_2, \frac{3}{4})\}$	$\{(c_1, \frac{1}{3}), (c_2, \frac{2}{3})\}$	d_2	e_2
x_8		b_1	c_2	d_1	e_3

TABLE II
INFORMATION SYSTEM S_2

X	a	b	c	d	e
x_1	$\{(a_1, \frac{2}{3}), (a_2, \frac{1}{3})\}$	$\{(b_1, \frac{1}{4}), (b_2, \frac{3}{4})\}$	c_1	d_1	$\{(e_1, \frac{1}{3}), (e_2, \frac{2}{3})\}$
x_2	$\{(a_2, \frac{1}{4}), (a_3, \frac{3}{4})\}$	b_1	$\{(c_1, \frac{1}{4}), (c_2, \frac{3}{4})\}$	d_1	e_1
x_3	a_1	b_2	$\{(c_1, \frac{1}{2}), (c_3, \frac{1}{2})\}$	d_2	e_2
x_4	a_3		c_2	d_1	e_2
x_5	$\{(a_1, \frac{3}{4}), (a_2, \frac{1}{4})\}$	b_1	c_2		e_1
x_6	a_2	b_2	c_3	d_2	$\{(e_2, \frac{1}{3}), (e_3, \frac{2}{3})\}$
x_7	a_2	$\{(b_1, \frac{1}{4}), (b_2, \frac{3}{4})\}$	c_1	d_2	e_2
x_8	$\{(a_1, \frac{2}{3}), (a_2, \frac{1}{3})\}$	b_1	c_2	d_1	e_3

B. Clearly $A \cap B = \emptyset$. The null value imputation algorithm Chase converts information system $S(A \cup B)$ of type λ to a new more complete information system $Chase(S(A \cup B))$ of the same type. The proposed strategy is new in comparison to known strategies for chasing missing values in relational tables because of the assumption about partial incompleteness of data (sets of weighted attribute values can be assigned to an object as its value). We use ERID algorithm [4] in Chase method to extract rules from such type of data.

IV. SEARCHING MORE DETAILED INFORMATION SYSTEM

Assume again the information system (S, K) . As we already mentioned, the knowledge base K , contains rules extracted earlier either locally (inside primary information system) or globally (obtained from distributed information systems). We want to find the most optimal information system (S_i, K) for primary information system (S, K) , where by optimal we mean the closest system. The distance between two information systems is calculated using the formula:

$$d(S_1, S_2) = \frac{\sum_r d_r(S_1 \rightarrow S_2) + \sum_r d_r(S_2 \rightarrow S_1)}{\sum_r \sup S_1 \cdot \text{conf} S_1 + \sum_r \sup S_2 \cdot \text{conf} S_2}$$

where

$$d_r(S_1 \rightarrow S_2) = \left| \frac{\sup S_2 \cdot \text{conf} S_2}{\max(\sup S_1, 1)} - \frac{\sup S_1 \cdot \text{conf} S_1}{\max(\sup S_2, 1)} \right|.$$

Information system with the minimal value of $d(S_1 \rightarrow S_2)$ has to be chosen. Then it means that S_1 system is the closest to primary information system.

Let us assume we have distributed information system consisting of three different information systems, collaborating

TABLE III
INFORMATION SYSTEM S

X	a	b	c	d
x_1	2	2	–	
x_2	2	1	+	
x_3	0	1	–	
x_4	1	2	+	
x_5	0	1	–	
x_6	1	3	–	

TABLE IV
INFORMATION SYSTEM S_1

Y	a	b	c	d	e
y_1	1	1	+	1	L
y_2	1	2	–	1	H
y_3	2	3	–	1	L
y_4	2	1	+	1	L
y_5	1	2	–	0	H
y_6	2	3	–	0	H
y_7	0	1	+	1	L

with each other S, S_1, S_2 , represented as Table3, Table4 and Table5, respectively.

Information system S received a query $q(B) = (a, 0) * (b, 1) * (d, 1)$ and it has no information about attribute d , which is hidden. Meanwhile this attribute appears in other systems. The purpose is to choose one of the systems: either

TABLE V
INFORMATION SYSTEM S_2

Z	a	b	c	d
z_1	2	1	+	0
z_2	2	1	+	1
z_3	1	2	-	0
z_4	1	3	-	1
z_5	0	2	-	0
z_6	0	3	+	1

S_1 or S_2 , from which values of attribute d in system S can be predicted. After this step we will be able to find set of objects satisfying given query $q(B)$. Because attributes a, b, c appear in all the systems, first we extract, from each information system independently, rules describing a, b, c in terms of other attributes, using method similar to LERS. Next, for each rule we calculate support and confidence in a standard way [4], [5].

For system S_1 we have:

$$\begin{aligned}
 (b, 1) &\rightarrow (a, 1) \text{ with } sup = 1, conf = \frac{1}{3} \\
 (b, 1) &\rightarrow (a, 2) \text{ with } sup = 1, conf = \frac{1}{3} \\
 (b, 1) &\rightarrow (a, 0) \text{ with } sup = 1, conf = \frac{1}{3} \\
 (b, 2) &\rightarrow (a, 1) \text{ with } sup = 2, conf = 1 \\
 (b, 3) &\rightarrow (a, 2) \text{ with } sup = 2, conf = 1 \\
 (c, +) &\rightarrow (a, 1) \text{ with } sup = 1, conf = \frac{1}{3} \\
 (c, +) &\rightarrow (a, 2) \text{ with } sup = 1, conf = \frac{1}{3} \\
 (c, +) &\rightarrow (a, 0) \text{ with } sup = 1, conf = \frac{1}{3} \\
 (a, 1) &\rightarrow (b, 1) \text{ with } sup = 1, conf = \frac{1}{3} \\
 \dots \\
 (a, 1) * (c, +) &\rightarrow (b, 1) \text{ with } sup = 1, conf = 1 \\
 (a, 1) * (c, -) &\rightarrow (b, 2) \text{ with } sup = 2, conf = 1 \\
 \dots
 \end{aligned}$$

For system S_2 we have:

$$\begin{aligned}
 (b, 1) &\rightarrow (a, 1) \text{ with } sup = 2, conf = 1 \\
 (b, 1) &\rightarrow (a, 2) \text{ with } sup = 2, conf = 1 \\
 (b, 2) &\rightarrow (a, 1) \text{ with } sup = 2, conf = \frac{1}{2} \\
 (b, 2) &\rightarrow (a, 0) \text{ with } sup = 2, conf = \frac{1}{2} \\
 (b, 3) &\rightarrow (a, 1) \text{ with } sup = 2, conf = \frac{1}{2} \\
 (b, 3) &\rightarrow (a, 0) \text{ with } sup = 2, conf = \frac{1}{2} \\
 (c, +) &\rightarrow (a, 2) \text{ with } sup = 2, conf = \frac{1}{3} \\
 (c, +) &\rightarrow (a, 0) \text{ with } sup = 1, conf = \frac{1}{3} \\
 (c, -) &\rightarrow (a, 1) \text{ with } sup = 2, conf = \frac{1}{3} \\
 (c, -) &\rightarrow (a, 0) \text{ with } sup = 1, conf = \frac{1}{3} \\
 (a, 1) &\rightarrow (b, 2) \text{ with } sup = 1, conf = \frac{1}{2} \\
 \dots \\
 (a, 0) * (b, 3) &\rightarrow (c, +) \text{ with } sup = 1, conf = 1 \\
 (a, 0) * (d, 1) &\rightarrow (c, -) \text{ with } sup = 1, conf = 1 \\
 \dots
 \end{aligned}$$

We do the similar procedure for system S .

Next the distance between S and S_1 is calculated: $d(S \rightarrow S_1) = \frac{33.36+20.3130+34.66}{36.66+45.3130+39.45} = 0.686$ and so between S and S_2 : $d(S \rightarrow S_2) = \frac{33.36+17.17}{30+29.66} = 0.85$ Because the distance between S and S_1 is smaller than between S and S_2 (the factor is smaller), we choose S_1 as more detailed information

system for contact with S . From chosen information system S_1 , rules describing attribute d in terms of a, b, c are extracted. We can use algorithm ERID [4], [11], [12] for extracting rules from incomplete information system and put them into knowledge base K . These rules also allow us to uncover some hidden attribute values in information system S . Therefore the submitted query $q(B)$ can be answered and the set of objects is x_3, x_5 .

This method can enhance and expand the scope of decision support system, which assists patients and physicians with the challenge of managing pancreas diseases. It involves data collection of pancreatic cancer, risk factors, common characteristics and survival rates. The objective of the system is to detect problems in pancreatic management and to recommend changes to correct these detected problems. It is very important problem, as the survival rate of pancreatic cancer is very poor and those that survive are due to early detection. Early detection of pancreatic cancer is very difficult and often involves some sort of invasive testing procedure. One of the main symptoms of pancreatic cancer is chronic pancreatitis. If physicians are prompted to order additional testing at the time they entered a pancreatitis diagnosis the cancer has a chance to be detected earlier.

V. FINAL REMARKS

In the paper we proposed the method of finding the closest information system to the client. Our goal was to find the best information system, which will be helpful in answering the query submitted to the client. Once we find more detailed information system to the given one, we are able to build knowledge base consisting of rules extracted in distributed systems. Then we can apply the rules, so changes of values of attributes in a query $q(B)$ can be made. The unknown attributes can be replaced by mixture of attributes which are present in both of the systems, the query will transform into more clear and understandable form and can be then answered.

This method was initially tested in medical databases with special preferences related to different pancreas diseases and gives some promising results. Several information systems, keeping different information were connected with other in hospital systems. Each information system consists of a big set of knowledge not necessary connected with patients with pancreas diseases. From the whole set of databases - systems with the best knowledge about particular diseases should be chosen for communication. The best treatment for most of pancreatic diseases, especially cancer, depends on how far it has spread, or its stage. The stages of pancreatic cancer are quite easy to understand and work on. The problem which appears in pancreas cancer treatment is how to describe the stage of pancreatic cancer without previously resorting to major surgery and how to minimize the risk of the disease. In practice, doctors choose pancreatic cancer treatments based upon imaging studies, surgical findings, and an individual's general state of well being. Determining pancreatic cancer's stage is often not so easy. Imaging tests like CT scans and ultrasound of course give doctors some information, but

knowing exactly how far pancreatic cancer has spread already, usually requires deep surgery. Therefore finding the closest information system help to work with the latest, most advanced therapies for pancreatic diseases and can be used to ensure the most advanced treatment with the least impact on patient's body. Using our method we can suggest some restrictions showing how to reduce the risk of pancreatic disease. We obtained knowledge which gives information that smoking is the most important avoidable risk factor for pancreatic cancer, and quitting smoking helps to lower risk. Also getting physical activity and eating well can help patient to stay at a healthy weight and also reduce the risk of illness. The third possibility to minimize the risk for pancreatic cancer is to avoid workplace exposure to harmful substances such as certain pesticides and other chemicals.

REFERENCES

- [1] V.R. Benjamins, D. Fensel, A.G. Prez, "Knowledge management through ontologies", *Proceedings of the 2nd International Conference on Practical Aspects of Knowledge Management (PAKM-98)*, Basel, Switzerland, 1998
- [2] B. Chandrasekaran, J.R. Josephson, V.R. Benjamins, "The ontology of tasks and methods", *Proceedings of the 11th Workshop on Knowledge Acquisition, Modeling and Management*, Alberta, Canada, 1998
- [3] A. Dardzinska, "Action rules mining", *Springer Verlag*, 2013
- [4] A. Dardzinska, Z. Ras, "Chase2, Rule based chase algorithm for information systems of type lambda", *Proceedings of the Second International Workshop on Active Mining*, Maebashi City, Japan, LNAI SpringerVerlag no3430, 2005
- [5] A. Dardzinska, Z. Ras, "On Rules Discovery from Incomplete Information Systems", *Proceedings of ICDM'03 Workshop on Foundations and New Directions of Data Mining*, Melbourne, Florida, IEEE Computer Society, 2003
- [6] A. Dardzinska, A. Romaniuk, "ncomplete distributed information systems optimization based on queries", *Advances in Swarm and Computational Intelligence : 6th International Conference : ICSI 2015*, LNCS Springer, Pekin, China, 2015
- [7] D. Fensel, "Ontologies: a silver bullet for knowledge management and electronic commerce", Springer-Verlag, 1998
- [8] N. Guarino, "Formal Ontology in Information Systems", IOS Press, Amsterdam, 1998
- [9] N. Guarino, P. Giaretta, "Ontologies and knowledge bases, towards a terminological clarification", *Towards Very Large Knowledge Bases: Knowledge Building and Knowledge Sharing*, IOS Press, 1995
- [10] Z. Pawlak, "Information systems - theoretical foundations", *Information Systems Journal*, Elsevier, Vol. 6, 1981
- [11] Z. Ras, A. Dardzinska, "Ontology based distributed autonomous knowledge systems", *Information Systems International Journal*, Elsevier, Vol.29, No. 1, 2004
- [12] Z. Ras, A. Dardzinska, "Solving Failing Queries through Cooperation and Collaboration", *World Wide Web Journal*, Springer, Vol. 9, No. 2, 2006
- [13] Z. Ras, S. Joshi, "Query approximate answering system for an incomplete DKBS", *Fundamenta Informaticae Journal*, IOS Press, Vol. 30, No. 3/4, 1997
- [14] J. Sowa, "Ontology, metadata, and semiotics", *Conceptual Structures: Logical, Linguistic, and Computational Issues*, LNAI, No. 1867, Springer-Verlag, Berlin, 2000
- [15] J. Sowa, "Knowledge Representation: Logical, Philosophical and Computational Foundations", *Brooks/Cole Publishing Co.*, Pacic Grove, CA, 2000
- [16] J. Sowa, "Ontological categories", *Shapes of Forms: From Gestalt Psychology and Phenomenology to Ontology and Mathematics*, Kluwer Academic Publishers, Dordrecht, 1999
- [17] G. Van Heijst, A. Schreiber, B. Wielinga, "Using explicit ontologies in KBS development", *International Journal of Human and Computer Studies*, Vol. 46, No. 2/3, 1997