EXTRACTION OF SEMANTIC RELATIONSHIPS STARTING FROM SIMILARITY MEASUREMENTS

Mohamed Frikha, Mohamed Mhiri and Faiez Gargouri
MIRACL Laboratory, ISIM Institute, BP 1030 - 3018, Sfax, Tunisia

Keywords: Information System, Ontology, semantic relationships, similarity measurement, Conceptual Schema.

Abstract: Current applications' modelling becomes increasingly complex. Indeed, it requires a hard work to study the particular domain in order to determine its main concepts and its relationships. The designers can have, in certain case many ambiguities concerning the comprehension of the domain to be modelled and the concepts to be used. In order to solve these ambiguities, we used ontology like a reference to give more semantics to conceptual schemas. For that, we used an approach for an ontology building to represent the pertinent concepts for a domain. In this paper, we propose a set of allowing determining the resemblance between the concepts of a conceptual schema and the ontology. Then, we propose an algorithm using these similarity measurements to determine the semantic relationships.

1 INTRODUCTION

In general, the majority of conceptual schemas (CS) of information systems (IS) are created from scratch, wasting time and many resources. These CS can contain errors, due to the ignorance of the domain to be modelled. Several approaches were proposed to make more semantics and to solve these errors. The ontology is the most known. It represents « *a description of the concepts and relations which can exist* » (Gruber, 1993).

The solution which we proposed consists in the definition and the building of an ontology known as *IS design ontology* (Mhiri[1] et al., 2005). The latter constitutes a means of help and assistance in the realization of design task. Such ontology makes it possible to solve the problems of knowledge representation of IS, such as the problems of the expressivity, the comprehensibility, the sharing and the reuse.

In this paper, we present the complementarity between the design of IS and ontologies. Then in the following section, we present the various interactions between ontology and a conceptual schema. Then, we present our contribution consisting in integrating ontologies in the IS design process. For that, similarity measurements must be calculated and an algorithm of search in ontology must be established.

2 ONTOLOGY VERSUS INFORMATION SYSTEMS

Modelling consists in creating a virtual representation of a reality in order to emphasize the interesting points. There are several methods and languages of IS design like MERISE, UML language, etc. However, the design of the complex applications handles a significant quantity of changed information. It risks to have ambiguities and to contain errors (Gargouri, 2002).

To solve these problems, we used ontology. It makes possible to study the semantics of the whole concepts and the relationships between these concepts of the various CS for a given domain.

The existing conceptual schema can be used to create ontology (figure 1), whereas existing ontology can be used to improve CS (Fonseca et al., 2003).

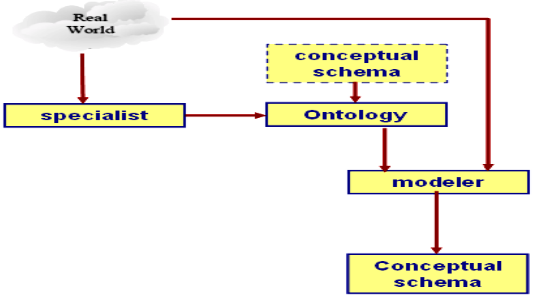

Figure 1: Ontology-based modelling process.

Frikha M., Mhiri M. and Gargouri F. (2007).
EXTRACTION OF SEMANTIC RELATIONSHIPS STARTING FROM SIMILARITY MEASUREMENTS.
In *Proceedings of the Ninth International Conference on Enterprise Information Systems - ISAS*, pages 603-606
DOI: 10.5220/0002392706030606

Each conceptual schema to integrate is compared to the ontology, and for each conflict to find the system calculates similarities measurements between the concept of ontology and the component of the conceptual schema to check.

In the following paragraph, we present our approach to ensure an interaction between a built ontology and a given CS.

3 DIFFERENT INTERACTIONS BETWEEN ONTOLOGY AND CS

To verify a conceptual schema, it is necessary, at first, to check any component of this CS with its correspondence in the ontology. A component can be a class, a conceptual relationship (association, aggregation, composition and inheritance).

This checking requires a comparison step by using the similarities measurements calculation between the ontology concepts and CS components.

3.1 Similarities Measurements

We define a similarity measurement between two concepts (C1, C2) like a value ranging between 0 and 1 allowing reassembling these concepts. To reach a more precise level of similarity between the concepts, we propose a set of formulas used during the algorithm of comparison.

To measure the similarities between two concepts, we combine syntactic matching between strings and semantic matching.

3.1.1 Syntactic Similarities

For syntactic matching, a function of distance is applied to a pair of strings, to determine the dissimilitude between them. In this work we adopted the Levenshtein distance. It is applied to the calculation of the similarity between the concept names (SimName(Cc,Co)) and between the attribute names (SimNameAt(Cc_{at}, Co)).

$$\mathrm{SimName}\ (Cc_n, Co_n) = 1 - \ (\mathrm{Lev}\ (Cc_n, Co_n) / \mathrm{Max}(\mathrm{long}(Cc_n), \mathrm{long}(Co_n))) \quad (1)$$

$$\mathrm{SimNameAt}\ (Cc_{at}, Co) = \underset{\mathbf{j}\,=\,1}{\overset{\mathbf{K}}{\mathbf{MAX}}} \ (1 - \mathrm{Lev}(Cc_{at}, Co_{at\,j}) / \mathrm{Max}(\mathrm{long}(Cc_{at}), \mathrm{long}(Co_{atj}))) \quad (2)$$

To calculate the similarity between attribute names of two concepts, we use a formula for the comparison of two concepts Cc and Co. Cc_{at} is an attribute from the Cc and Co_{at} is an attribute from the Co. *n* is the number of attributes in the Cc and *K* is the number of attributes in the Co.

$$\mathrm{SimNameAtts}(Cc,Co) = \frac{1}{n} . \sum_{i=1}^{n} \underset{j=1}{\overset{K}{\mathrm{Max}}} \ \mathrm{SimName}\ (\mathrm{CCat}\ i, \mathrm{COat}\ j) \quad (3)$$

3.1.2 Semantic Similarities

We consider the nearest neighbour (Holt, 2000), which is used to calculate the similarity in terms of the attributes each concept presents, and is given by the formula:

$$\mathrm{SimAt}(Cc,Co) = \sum_{i=1}^{n} \mathrm{SimNameAt}(\mathrm{CCat}\,i, \mathrm{CO})\ \mathrm{xWati} \quad (4)$$

Where Cc and Co are, respectively, the conceptual schema and the ontology concepts. *n* is the number of attributes considered, *i* is the index of the attribute being processed, SimNameAt($Ccat_i$,Co) is the function which calculates the similarities between the attributes of the compared concepts and Wat_i is the weight of the i^{th} attribute in the ontology.

The weight of an attribute is given by following formula:

$$W_{at} = \frac{\Sigma\ Co_{at}}{\Sigma\ Co} \quad (5)$$

Where Co_{at} is the number of concepts that the attribute has and Co is the total number of concepts belonging to ontology.

Three types of relationships are considered for the similarities measurement of a pair of concepts. The first one is the taxonomic associations (IS-A), and the other two are the aggregation and composition ones. The similarity in terms of the place in the hierarchy, where each concept is located, is obtained by the formula:

$$\mathrm{SimHier}(Cc,Co) = \frac{\Sigma(\mathrm{Hier}(Cc,Pc).\mathrm{Wt}(c,p))}{\mathrm{Nhier}(Cc,Pc)} \quad (6)$$

Where Hier(Cc,Pc) is each one of the taxonomic relationships existing in both the conceptual schema and the ontology. Wt(c,p) is the weight of the hierarchical relationship arc, and Nhier(Cc,Pc) is the number of IS-A associations in both the ontology and the conceptual schema.

The weight Wt(c,p) of a taxonomic arc is given by the following formula (Jiang and Conrath,1997):

$$Wt(c,p) = \frac{(E)}{E(p)} \cdot \frac{(d(p)+1)}{d(p)} \cdot (IC(c) - IC(p)) \qquad (7)$$

Where d(p) is the depth of the parent node (p) of the node corresponding to the concept being compared. E is the density of the whole ontology's hierarchy, that is, the number of nodes it has. E(p) is the density of the taxonomy considering the node p as the root concept, that is, the number of direct and indirect children it has. Finally, IC (Information Content) represents the amount of information the node has (Resnik, 1998), and its value is given by:

$$IC(c) = -\log(\,(\,\Sigma(1/sup(c)\,)\,).1/N) \qquad (8)$$

Where sup(c) is the number of super classes (direct or indirect) the class c has, and N is the total number of concepts of the ontology. The more specialized a concept is, the more information it intrinsically possesses.

Finally, the aggregation and composition links are considered to calculate the similarity between two concepts, by the simple formula:

$$SimRel(Cc,Co) = (\Sigma(Rel(Cc,Co))/Rel(Cc)) \qquad (9)$$

Where Rel(Cc,Co) is each composition/ aggregation link existing both in the ontology and in the conceptual schema and Rel(Cc) is concerned with the ones present only in the conceptual schema.

3.1.3 Extractions of the Semantic Relationships between the Concepts

After having calculated the similarities between the concepts, we will use these formulas to determine the semantic relations (Mhiri[2] et al., 2006) which can exist between these concepts. These relationships are defined as follow:

Table 1: Determination of semantic relationships between concepts.

	Syntactic formulas		Semantic formulas		
formula relation	SimName(C,C)	SimNameAtts(C,C)	SimAt(C,C)	SimHier(C,C)	SimRel(C,C)
Identity	= 1	=1	= 1	= 1	= 1
Synonymy	< SAn	> SAc	> SAc	> SAc	> SAc
Homonymy	= 1	< SAn	< SAn	< SAn	< SAn
Equivalence	< SAn	< SAn	> SAc	> SAc	> SAc
Kind of	< SAn	> SAn	-	> SAc	-

SAn is the threshold of Analysis. It is the minimal value which two concepts can have between them, by being applied to a formula, to analyze them. SAc is the threshold of Acceptance, if the two concepts have a value higher than the threshold of acceptance for a formula, then we can say that they satisfied this formula. These values are given by a domain's expert.

This table determines the semantic relationships which can exist between two concepts according to the values of the syntactic and semantic formulas.

Each formula is calculated for each concept in the conceptual diagram with ontology concepts. The ontology can also be dynamically updated depending on the similarities measurements carried out with every new conceptual schema. Attributes and relationships can be added to existing concepts, and even new concepts may be inserted.

3.2 The Algorithm of Search in Ontology

The algorithm is described then illustrated in figure 2 details in a high level of abstraction, the relationships that can be exist between two concepts.

The concept which does not exist in our ontology and which does not have semantic relations with another concept in the ontology, will be added as being new concept.

If a candidate has a semantic relationship with a concept of ontology it will not be automatically considered as a new concept. Any semantic relation must be presented at the expert.

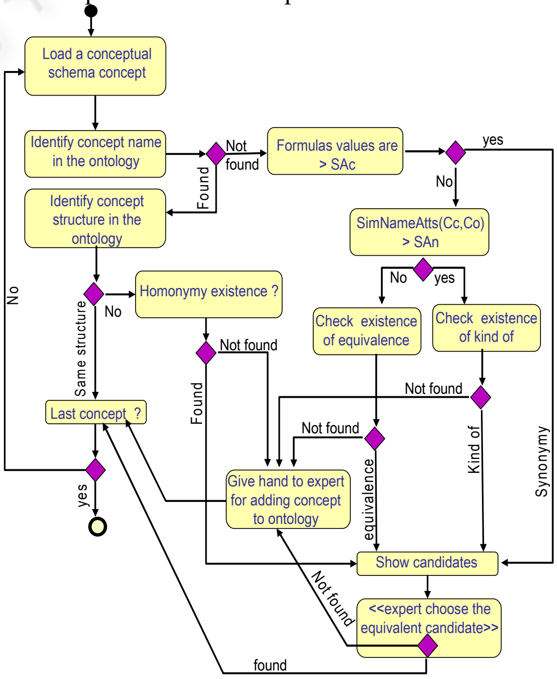
Load a conceptual schema concept
Identify concept name in the ontology
Not found
Found
Formulas values are > SAc
yes
No
Identify concept structure in the ontology
SimNameAtts(Cc,Co) > SAn
No
yes
Homonymy existence ?
No
Same structure
Check existence of equivalence
Check existence of kind of
Not found
Found
Not found
Last concept ?
Not found
yes
Give hand to expert for adding concept to ontology
equivalence
Kind of
Synonymy
Show candidates
Not found
<<expert choose the equivalent candidate>>
found

Figure 2: Algorithm search of semantic relationships.

Thereafter, we will present the steps of progress of this algorithm:

Step 1 – Search concept's name in the ontology: If the concept's name is found in the ontology, go to step 2. Else go to step 5.

Step 2 – Search concept's structure in the ontology: Once the term which nominates the concept is found in the ontology, its structure is compared against the ontology, attribute by attribute. The algorithm verifies if there is a one to one correspondence between each input concept's attribute and the ontology concept's attribute, if we find the same structure, it is the identity and we go to step 3. If there are differences in at least one of the attributes, go to step 4.

Step 3 – Tests if it is the last concept: If the current concept is the last one of the schema, go to the end. Else go back to step 1 to processes of the next concept.

Step 4 – determine the existence of a homonymy: we calculate measurements of similarities presented in the preceding step in order to check if it is a homonymy relation. If yes, go to step 8, else go to step 10.

Step 5 – calculate the similarities: The formulas of similarity will be calculated between the CS concept and each concept of ontology. Go to the following step.

Step 6 – Verify threshold: Check if the similarity value of these formulas *SimNameAtts(Cc,Co), SimAt(Cc,Co), SimHier(Cc,Co)* and *SimRel(Cc,Co).* If all formulas values are higher than the acceptance threshold, it is the synonymy and go to step 8, else go to the following step.

Step 7 – check if it is an equivalence or a kind of relation: compare the value of *SimNameAtts(Cc, Co)* with the analysis threshold. If it is higher than the analysis threshold, check if it is a kind of relation by comparing the other formulas with the thresholds, else check if it is an equivalence relation. Go to the following step, else go to step 10.

Step 8 – present the Candidates: present each candidate found, with his relationship with the CS concept. Go to step 9.

Step 9 – relation selection: At this point the domain expert intervention is necessary. He selects the concept he judges as the most equivalent to the input schema's concept. If the expert chooses a non-existing relation in our ontology, then we add this relation to ontology. Go to step 3.

Step 10 - Addition of a new concept to ontology: In this step, we give the hand to an expert for adding the new concept in ontology, with all its attributes and their new semantic relation. Go again to step 3.

4 CONCLUSION

We presented an approach of ontology building for the IS design. It allows the extraction of the concepts and its relations starting from CS of UML. Then, we presented the role of ontology in the modelling phase. Then, we showed the different interactions between ontology and an unspecified CS in order to check the CS and assist the designer in his work and guarantee the reuse, the extensibility and the comprehension of a conceptual diagram. For that, we defined some formulas using similarities measurements to compare and integrate different CS modelling the same part of reality.

Like perspective for this work, we will use these formulas in a case study for a particular domain. Then, we will integrate these rules in design process to ensure the coupling between CS and ontology.

REFERENCES

Fonseca, F. T., Davis, C., Câmara, G.,2003. Bridging Ontologies and Conceptual Schemas in Geographic Information Integration. *GeoInformatica. v.7*, n.4, p.355-378. Kluwer Academic Publishers, 2003.

Gargouri F. , 2002. Modélisation de la complexités des systèmes d'information à travers la coopération par intégration de représentation conceptuelles. *Habilitation universitaire en informatique*. Tunis 02.

Gruber T. R., 1993. Towards principle for the design of ontology used for knowledge sharing. In N. Guarino and R. Poli, editors, *Formal Ontology in Conceptual Analysis and Knowledge Representation, International Workshop on Ontology*.Kluwer Academic.

Hess G., Iochpe N.,Cirano, 2004. Ontology-driven resolution of semantic heterogeneities in GDB conceptual schemas. *GEOINFO 2004 VI Brasilian Symposium of GeoInformatic.*

Holt, A., 2000. Understanding environment and geographical complexities trough similarity matching. *In Complexity International, number 7.*

Jiang, J., and Conrath, D.,1997. Semantic Similarity Based in Corpus Statistics and Lexical Taxonomy. *In Proc of International Conference Reasearch in Computational Linguistics*. Taiwan.

Mhiri[1], M., Mtibaa, A., Gargouri, F., 2005. Towards an approach for building information systems'ontologies. *FOMI'2005,* Verona, Italy, 9-10, June.

Mhiri[2] M.,Chabaane S.,Mtibaa A.,Gargouri F.,2006. An Algorithm for Building Information System's Ontologies, *ICEIS 2006.*

Resnik, P., 1998. Semantic Similarity in a Taxonomy: An Information-Based Measure and its Application to Problems of Ambiguity in Natural Language. *Journal of Artificial Intelligence Research*, p. 95-130.