Identifying Beginner Problems in Expressing Domain Semantics when Developing Ontologies

Sandra Lovrenčić, Mirko Maleković
University of Zagreb
Faculty of organization and informatics
HR-42000 Varaždin, Croatia
sandra.lovrencic@foi.hr, mirko.malekovic@foi.hr

Abstract—**Ontologies as a knowledge representation method are already being applied in various areas. Therefore, this method is introduced to new developers constantly. The paper investigates possibility for overlooking ontology features that can enable users to properly represent semantics of the domain of interest. In initial research, ontology development using frames was considered and evaluation was made based on criteria connected to classes, hierarchy and attributes. Possible beginner oversights are identified. Suggestions considering chosen semantic criteria are also described.**

Keywords-knowledge representation; ontology development; domain semantics; semantic criteria

I. Introduction

Knowledge representation with the help of ontologies is a subject of research for two decades already. With new technologies emerging every day and Semantic Web vision, they have become an important part of various research areas, including knowledge management. Along with their features, their application is also spreading.

In a well known paper [1] ontology is defined as "an explicit specification of a conceptualization" and frame systems are described as knowledge representation framework for "describing hierarchies of classes with slots" that ontologies consist of. Over years, many ontology development methods, languages and tools have been evolved [2]. Ontology evaluation is, of course, an integrating part of their development and was in centre of research about five years ago. Evaluation of ontology content is concentrated on consistency, completeness, conciseness, expandability and sensitivity, whereas ontology taxonomy evaluation considers inconsistency, incompleteness and redundancy [3][4]. Well known OntoClean method evaluates ontologies according to rigidity, identity, unity and dependency, concepts introduced from philosophy [5][6]. Ontology evaluation can be based on structural, functional and usability-profiling measures [7] as well as on "coverage of a particular domain and the richness, complexity and granularity of that coverage; the specific use cases, scenarios, requirements, applications, and data sources it was developed to address"[8]. Factors considered in the evaluation process can be features of languages and tools used [9], but also user demands and simplicity of use [10]. Evaluation methods can be combined to explore various ontology characteristics [11].

Some aforementioned evaluation methods are designed to be conducted independently of ontology development methods, tools or languages used and others consider them as possible biases that can influence on richness of knowledge representation. However, a factor of knowledge and experience of ontology engineer is rarely taken into consideration. To new developers ontologies are constantly introduced as a knowledge representation method. If accustomed to different means of representing knowledge, such as classical databases, they may not use all features that ontologies offer for representing the semantics of the domain of interest, for example, description of classes with the use of instances of other class. Defining and focusing on potential oversights when teaching or learning how to develop ontologies can reduce initial mistakes. Therefore, the main goal of presented research was to evaluate basic ontology elements, such as classes, hierarchy and attributes (slots) with the purpose to discover how well beginners can understand and exploit the concept of ontologies when managing and representing knowledge.

The paper is organized as follows: in second section semantic criteria for ontology evaluation are introduced; afterwards, research process as well as analysis and results are described; conclusion and future work are in the final section.

II. Semantic Criteria

When considering the use of ontologies, "the most important aspect of the ontological representation is its capacity of expressing domain semantics"[12]. Generally, ontologies represent semantic knowledge of a certain domain through hierarchies of classes and attributes (and their constraints) that describe them. Therefore, those features should be used for a proper domain description, but some of them may be overlooked, especially with the lack of experience. Detection of those oversights can give valuable information about important parts of ontology development lessons.

At the Faculty of organization and informatics in Varaždin, Croatia, ontologies are taught at two levels:

- second or third year undergraduate students learn ontology development at simple level within Knowledge Management course, where only frame

 ISBN: 978-1-61208-104-5

systems as knowledge representation formalism are introduced and students have no prerequisites that include formal logical systems;

- second year graduate students learn ontology development with OWL and description logic reasoning within Knowledge Bases and Semantic Web course and have prerequisites that include formal logical systems.

For initial research with the purpose of simplicity and further guidelines, only first case is considered. Since the goal was to discover how easy beginners can grasp the concept of ontology and how many semantics will they be able to represent with it, ontology elements used within frames - classes, hierarchy and attributes (slots) were obvious choice for analysis.

During previous years, it was noted that students who develop ontologies for the first time tend to describe classes only with simple string or integer attributes and that they are inclined to either develop very poor hierarchies with many attributes or very rich ones where even instances are mixed for classes. For that reason lectures were organized in a manner that each covered one specific development part: detailed description of development process with examples, development of classes and hierarchy and the use of attributes. The hypothesis was that beginners will to some extent overlook the use of more complex ontology features – complex attributes, use of more hierarchies and their connection for better domain description. According to important ontology elements, several criteria were taken into consideration for the evaluation:

Total number of hierarchies – Although this is not commonly, for the purpose of research, class hierarchy was divided into two parts: main hierarchy (describing the domain of interest) and support hierarchy (used to better describe the domain of interest). For example, University studies ontology has several such hierarchies: types of studies, teaching participants, courses, conduction places and enrolment requirements [13]. Because the domain of interest was types of studies, this would be main hierarchy. Other hierarchies would help in its description – their classes or instances would be used as values for class attributes in main hierarchy. Therefore, this criterion can imply more semantically versatile description.

Number of support hierarchies – There can be several main hierarchies in complex domains, as well as support ones. Because support hierarchies are those that designate more complexity in domain description, it is necessary to determine their actual number (if any).

Depth of main and support (where applicable) hierarchies – It is obvious that hierarchies with more branches and more depth give better description of domain structure and class relations and therefore represent a valuable criteria. Main hierarchy can be the only hierarchy in one-hierarchy ontology or one or more of those that directly describe the domain of interest in multiple-hierarchy ontology. These criteria are considered with hypothesis that support classes will have lesser depth than main ones.

Total number of classes, number of classes in main and support hierarchies (where applicable) – The number of support hierarchies and hierarchy depth cannot itself give complete information about the degree of semantics represented: main hierarchy should obviously have a number of classes, but support hierarchy can actually consist of only one, whose instances must be values for a certain attribute. Only ontologies that have support hierarchies were evaluated according to these criteria, whereas all ontologies were used for analysis of total number of classes.

Total number of attributes – It is needless to say that attributes are the real descriptors of classes and that their greater number should mean better semantic representation. For this criterion the total number of attributes is taken, regardless whether they belong to main or support hierarchy.

Number of attributes in main and support (where applicable) hierarchies – These criteria gives even better insight in how well is which part of ontology described. Of course, it is applicable only on ontologies that have support hierarchies.

Number of connecting attributes – Connecting attributes are those that connect classes together, primarily meaning that the attribute value of one class is the instance of the other (regardless whether it is a part of main or support hierarchy). They show how well are represented connections among various parts of the domain, and the actual effect of support classes – how much semantic they add.

Number of simple and complex attributes – The last two criteria show the complexity of class description. Attributes are divided into two groups, simple and complex. Simple attributes are any, boolean, float, integer and string whilst complex are class, instance and symbol.

It should be noted that those criteria are chosen according to main ontology elements using frames. They can be proven more or less useful after the research and need for other criteria can be discovered.

III. RESEARCH

Research was conducted at Faculty of organization and informatics during spring semester of year 2009/2010.

A. Participants

As already described, students are taught knowledge representation with ontologies during laboratory exercises in course Knowledge Management. For that reason, research was conducted with second and third year undergraduate students at this course (year of course enrollment is not fixed; only prerequisites are). Participants had no prior experience with ontologies, but were familiar with knowledge representation methods for knowledge management in general. Total number of students was 152 in 10 groups. Ontology development is part of their final grade, but several irregular students decided to apply for the regular exam and not to present their work.

Laboratory exercises were divided into two parts. First part is not the subject of this research, but was good introductory for ontologies: students had to collect knowledge about some topic in knowledge management domain, represent it in a wiki system and tag important concepts that were then visualized in graph.

 ISBN: 978-1-61208-104-5

Assignments from both parts of laboratory exercises were included in student grades with 25% in total. Also, students had to obtain at least 12 of 25% to be able to apply even for a regular exam. This ensured their motivation to accomplish given tasks.

B. *Research Process*

As a tool for ontology development was chosen Protégé [14], as one of most used open-source tools that has good user interface and support and is being developed for more than 20 years [15]. Version that was used is Frames without Protégé Axiom Language (subset of first order logic axioms), for several reasons:

- participants were undergraduate students with no prerequisites that included knowledge of first order logic;
- although they had mostly the same courses in their first year, students can choose between two directions in their undergraduate studies, information systems and business systems – therefore, their interest and knowledge of informatics topics is not the same;
- Protégé editor is very intuitive and allows easy manipulation with ontology elements of interest for the research.

Because of grading, each student had to choose a different domain for ontology, according to hers/his interests. Domains could be similar, but not exactly the same (for example, car models from two different manufacturers). Their task was to represent the chosen domain with ontology as best as possible and to incorporate into it all features that were taught to them.

Laboratory exercises consisted of four sessions. Activities at each session are described below:

Session 1 – Students were taught about ontologies through example of University studies ontology [13]. Firstly, the role of classes and their attributes in hierarchy was explained to them. Then they had a task to create a small hierarchy example. Protégé-Frames tool was also presented to them with step by step explanation how to create ontology. Their next task was to try out the tool. Students also had enough time to start searching for a suitable domain according to their preferences and interests. They had to find a domain for ontology development until next session. As already explained, they had to have different ontologies.

Session 2 – The most important task for this session was to create one or more class hierarchies. Each student's hierarchy was individually controlled and they were given suggestions for better arrangement of classes. Also, at least one support class for better semantic description of the specific domains was proposed to each of them.

Session 3 – For this session the most important task was to create appropriate attributes and connect with them all hierarchies together (where applicable). Suggestions for more use of complex attributes and explanations how to use attributes to connect different classes were also given to each student.

Session 4 – The last session was actually used for presentation and grading of ontologies. Students had to finish ontologies at home (create frames for instance entry window, populate ontology with enough instances to be able to make queries, create several queries, visualize ontology). About half of students already created some attributes at second session and populated instances at third, so they had enough time for the completion of the task.

IV. Analysis and Results

142 students delivered their ontology. As mentioned before, one of goals was also to determine whether some changes in semantic criteria definition have to be done. Therefore, for initial analysis 50 randomly selected ontologies were used. With purpose of better understanding of research results, information about values that have small range (0-4) is presented in Table 1. Following statistical measures were used for semantic criteria analysis: arithmetic mean, median, mode, standard deviation and skewness.

TABLE I. Selected Criteria Values

Criteria	**Values**				
	0	***1***	***2***	***3***	***4***
N. of hierarchies	-	30	14	1	5
N. of main hierarchies	-	49	1	0	0
N. of support hierarchies	30	14	2	4	0
Depth of main hierarchy	2	5	21	16	6
Depth of support hierarchy	14	4	2	0	0

The raw data from Table 1 already shows that less than a half of ontologies have support hierarchies and that almost all of them have only one main hierarchy. The support hierarchies generally have 0 depth (one class), and the depth of main ones is satisfactory. First two rows for value 0 are empty, because all ontologies have at least one main hierarchy. Detailed analysis is given in next subsections.

A. *Classification Analysis*

Table 2 shows average values obtained for selected ontologies according to following criteria: total number of hierarchies, number of support hierarchies, depth of main and support hierarchies and number of classes in total, and in main and support classes, where applicable.

It can be seen that most ontologies had only one hierarchy (median and mode are 1). Actually, 20 of 50 (40%) had at least one support hierarchy, meaning that more than half of students did not use this ontology feature to better describe domain knowledge. The average depth of main hierarchies was 2,38 with mode of 2 and their skewness showed that there was only a small asymmetry in sample distribution. As expected, the depth of support ontologies was mainly 0, indicating only one supporting class in most hierarchy cases.

Since to all students at least one support hierarchy or class was suggested, it can be concluded that the above result is influenced by this suggestion. With next generation no individual suggestions should be made. Instead, more

 ISBN: 978-1-61208-104-5

detailed explanation and more examples of support hierarchies should be included in the teaching process.

Average number of classes was 21,76, but other measures, especially a standard deviation of 17,8309, showed that there are some extreme values (inclining more to greater values, as can be seen from skewness). A number of classes in ontologies that have support classes was also very variable for main classes, but not for support ones. This is understandable, because most of them had only one class, although several of them had as many as 11 or 12.

TABLE II. HIERARCHY ANALYSIS

Criteria	Raw values		Statistical measures				
	Minimum	*Maximum*	*Average*	*Median*	*Mode*	*Standard deviation*	*Skewness*
Number of hierarchies	1	4	1,62	1	1	0,9452	1,6023
Number of support hierarchies	0	3	0,6	0	0	0,9035	1,5910
Depth of main hierarchies	0	4	2,38	2	2	0,9666	-0,2809
Depth of support hierarchies	0	2	0,31	0	0	0,6806	1,5139
Number of classes	6	110	21,76	16,5	14	17,8309	3,1115
Number of classes in main hierarchies	1	105	22,05	16	16	22,9858	2,6314
Number of classes in support hierarchies	1	12	3,23	2	1	3,4296	1,3704

Results obtained for hierarchy analysis showed that other criteria have to be included for class analysis because of large range of number of classes – from 6 to 110. Since, according to prior notions, students in a certain number of cases tend to represent even instances as classes, this can result in such a large range. Therefore, ontologies with different development mistakes should be analyzed separately. Diversity of the domains represented can be used for grouping of ontologies before ontology analysis.

Hierarchy information could not be affected by number of classes and it showed relatively even distribution. But it also pointed out that beginners do not understand a concept of support classes and their usefulness for better knowledge representation. This ontology feature demands more practice to be exploited.

B. Attributes Analysis

Information about the attributes analysis is presented in Table 3. Criteria used are as follows: total number of attributes, number of attributes in main and support hierarchies (where applicable), number of connecting attributes and number of simple and complex attributes.

Average number of attributes was 12,8, but standard deviation and skewness showed discrepancies of that value. For ontologies with support hierarchies results were the same for attributes used in main hierarchies. In support hierarchies there were no big discrepancies and number of attributes was very small. In most cases there were two attributes (mode value 2), but arithmetic mean of 5,38 and other measures showed variation of attribute number (which was actually from 1 to 17).

A smaller number of attributes in support hierarchies shows that only those for basic description of classes were used (sometimes only instance name). Although those classes help in better description of main hierarchy, the question arises whether they should be also fully described. In that case the description of the main class would also be better. Again, the importance and possibilities that support hierarchies have remain unused.

TABLE III. ATTRIBUTE ANALYSIS

Criteria	Raw values		Statistical measures				
	Minimum	*Maximum*	*Average*	*Median*	*Mode*	*Standard deviation*	*Skewness*
Number of attributes	4	60	12,18	9	8	9,1377	3,3203
Number of attributes in main hierarchies	2	54	10,13	7,5	8	11,2339	3,4564
Number of attrributes in support hierarchies	1	17	5,38	4	2	3,8580	1,9580
Number of connecting attributes	1	8	3,4	2	2	2,4902	1,0398
Number of simple attributes	2	52	9,4	7,5	8	8,2293	3,4407
Number of complex attributes	0	10	2,78	2	0	2,7575	0,9809

The number of connecting attributes showed that most of ontologies had 2 of them with average of 3,4 and values ranging from 1 to 8. As explained above, connecting attributes can be within main or support hierarchy. More analysis is necessary for determining whether the most often value of 2 attributes in support classes and 2 connecting attributes can indicate the following:

- support hierarchy – one general attribute for defining instance name and the other for reverse connection with the class described with that instance (value of that attribute is the instance of the class which attribute is instance of the class it belongs to – so called reverse slots in Protégé);
- connecting attributes – one attribute in described class and one reverse in class that describes it.

When comparing simple and complex attributes, regardless the values that show asymmetry of the distribution, it is obvious that mostly simple attributes were used. As mentioned above, this was noted during previous years of teaching this course. According to average number of complex attributes, they were probably those used as connection attributes. Obviously, they should be analyzed separately from the rest of complex attributes so that the percentage of usage of each of them can be calculated. Nevertheless, the small number of complex attributes in general showed that all their possibilities for better class description were not used.

In general, high standard deviation and skewness values indicate that distribution asymmetry does not allow accurate results interpretation. Aforementioned problem of representing instances as classes in a certain number of cases can have influence on large number of attributes in some ontologies, underlining that ontologies with different development mistakes should be analyzed separately. After grouping of ontologies according to domain similarity (as suggested in hierarchy analysis) it has to be determined how this will affect attribute analysis results and whether other criteria or ontology manipulation is necessary.

V. CONCLUSION AND FUTURE WORK

Results of conducted research pointed out several problems with oversights of new ontology developers. According to evaluated ontologies, common beginner oversights are:

- about 60% of users do not understand the value of support hierarchies in representation of semantic information (with the notion that the result is influenced by individual suggestions to include support hierarchies and that results could have been worse);
- users that created support hierarchies do not exploit their full potential (mostly only one class and less attributes for description of classes in support hierarchies);
- very small number of complex attributes shows that users possibly consider the number of attributes as main feature for embedding semantic information and not their complexity or that they do not fully understand their potential.

Some suggestions for improvement of semantic criteria can also been given, regarding prior analysis:

- the number of classes in general and also in main and support classes – large range in number of classes prevents correct interpretation of results and therefore ontologies with different development mistakes should be analyzed separately with additional semantic criteria;
- grouping of ontologies according to domain similarity can be conducted also with additional semantic criteria;
- the number of attributes – standard deviation shows more or less uneven distribution of values, also disabling correct interpretation, although some general conclusions can be made; after corrections in hierarchy analysis, effects of those changes should be analyzed with possible adjustment of semantic criteria.

Obtained results show that to certain aspects of ontology features more focus should be given when learning or teaching this formalism for representing domain knowledge. The future work in research of this problem will include:

- separation of ontologies with mistakes that cause extreme values in number of classes and/or attributes;
- grouping of ontologies according to domain similarity;
- adjustment of existing and establishment of new criteria;
- trial analysis of 50 ontologies with new settings and full analysis of all ontologies;
- change of focus in ontology development exercises with next generation of students and comparison of results;
- inclusion of second year graduate students that learn Protégé-OWL and description logics with adjustment of semantic criteria.

Given that knowledge representation using ontologies is integral part of Semantic Web and given that incorporating semantics in domain description is a precondition for its success, minimizing oversights that influence on proper representation of semantic information is of high importance. To new ontology developers all features that can aid in this effort should be pointed out.

REFERENCES

[1] T. Gruber, “A Translation Approach to Portable Ontology Specifications”, Knowledge Acquisition, vol. 5, pp. 199-220, June 1993.

[2] A. Gómez-Pérez, M. Fernández-López, and O. Corcho, Ontological Engineering, London: Springer-Verlag, 2004.

[3] A. Gómez-Pérez, “Evaluation of Ontologies”, International Journal of Intelligent Systems, vol. 16, pp. 391-409, September 2001.

[4] A. Gómez-Pérez, “Ontology Evaluation”, in Handbook on Ontologies, S. Staab and S. Studer, Eds. Berlin: Springer-Verlag, 2004, pp. 251-273.

[5] C. Welty and N. Guarino, “Supporting ontological analysis of taxonomic relationships”, Data & Knowledge Engineering, vol. 39, pp. 51-74, October 2001.

[6] N. Guarino and C. Welty, “Overview of OntoClean”, in Handbook on Ontologies, S. Staab and S. Studer, Eds. Berlin: Springer-Verlag, 2004, pp. 151-171.

[7] A. Gangemi, C. Catenacci, M. Ciaramita, and J. Lehmann, "Modelling Ontology Evaluation and Validation", Proceedings of the 3rd European Semantic Web Conference (ESWC2006), Springer, Budva, Montenegro, June 2006, pp. 140-154.

[8] L. Obrst, W. Ceusters, I. Mani, S. Ray, and B. Smith, "The Evaluation of Ontologies: Toward Improved Semantic Interoperability", in Semantic Web: Revolutionizing Knowledge Discovery in the Life Sciences, C. J. O. Baker and K-H. Cheung, Eds., London: Springer, 2007, pp. 139-158.

[9] Y. Sure et al., "Why Evaluate Ontology Technologies? Because it Works!", IEEE Intelligent Systems, vol. 19, pp. 74-81, July 2004.

[10] N. F. Noy, R. Guha, and M. A. Musen, "User ratings of ontologies: Who will rate the raters?", Proceedings of the AAAI 2005 Spring Symposium on Knowledge Collection from Volunteer Contributors, Stanford, CA, March 2005., www.stanford.edu/~natalya/papers/SS505NoyN.pdf, visited July 26, 2010.

[11] S. Lovrenčić and M. Čubrilo, "Ontology Evaluation – Comprising Verification and Validation", Proceedings of the 19th Central European Conference on Information and Intelligent Systems (CECIIS 2008), FOI, Varaždin, Croatia, Sep. 2008, pp. 657-663.

[12] C. d'Amato, S. Staab, and N. Fanizzi, "On the influence of Description Logics Ontologies on Conceptual Similarity", in Knowledge Engineering: Practice and Patterns: Proceedings of the 16th International Conference on Knowledge Engineering and Knowledge Management (EKAW 2008), Springer Verlag, Acitrezza, Italy, Sep.– Oct. 2008, pp. 48-63.

[13] S. Lovrenčić, Formal ontology of university studies, Ph.D. Thesis, Varaždin, 2007, in Croatian

[14] ***, Protégé, http://protege.stanford.edu, visited July 26, 2010.

[15] J. H. Gennari et al., "The evolution of Protégé: an environment for knowledge-based systems development", International Journal of Human-Computer Studies, vol. 58, pp. 89-123, Jan. 2003.