Ontology Mapping for Web-Based Educational Systems Interoperability

Amel Bouzeghoub, Abdeltif Elbyed
GET/INT, 9, rue Charles Fourier,
91011 Evry, France
{Amel.Bouzeghoub, Abdeltif.Elbyed}@int-evry.fr

Abstract. In order to deal with the need of sharing learning objects within and across learning object repositories most of the recent work argue for the use of ontologies as a means for providing a shared understanding of common domains. But with the proliferation of a great number of different ontologies even for the same domain, it becomes necessary to provide a mapping process to perform interoperability. Although many efforts in ontology mapping have already been carried out, few of them use resources properties to generate relations between local concepts. Our approach exploits these properties and uses inference rules to produce correspondences between concepts from source and target ontology.

Introduction

Web-based Educational Systems (WBES) is one of the leading domains where interoperability and sharing is in high demand. Indeed, the abundance of learning resources in the web involves the necessity of sharing and reusing content. Typically, these digital learning objects (LO) may be content stored as text, audio or video media files. Some efforts, deriving from organizations such as Ariadne [18] or EducaNext [19], have developed repositories for storing learning objects (LORs) described using a set of metadata (based on a standard, LOM [20] in most cases). Although these repositories organize the content of their resources and exchange resources, a problem of search and answer accuracy still remain. Semantic has to be associated to metadata values to tackle linguistic, inconsistent use of terms and cultural differences. Tools coming from semantic Web – ontologies– have to be integrated into repositories to organize the different concepts covered by the resources stored in a so called "knowledge domain ontology". Moreover, they do not offer powerful tools for reusing and composing existing resources. Finally, as the number of LORs increases, the problem of interoperability becomes more and more important, revealing problems of similarities, overlapping and cooperation of knowledge domains. It becomes increasingly difficult for users to obtain relevant information. Nowadays, domain ontologies are recognized as the most important issue in web semantic interoperability. The problem is that users are more familiar with their own domain ontology. It is not easy for them to use multiple ontologies in the remote repositories.
Most of the approaches treat the interoperability at a low level. In [16] the interoperability is based on common protocols, which define the interactions between repositories. A set of methods referred to as Simple Query Interface (SQI) has been proposed as a universal interoperability layer for educational networks. In other projects like Elena/Edutella [22] and eduSource [21] the openness is supported by a communication protocol.

We consider in our work that the interoperability may be supported at an ontological level and we propose as a first result an algorithm to generate mapping among different ontologies.

In this paper we present our approach based on either rules derived by human experts or basic deduction rules. The hypotheses generation combines different similarity measures to find mapping candidates between two ontologies.

The rest of the paper is structured as follows: First, we motivate our work by showing how ontologies are used in WBES in general and in our system named SIMBAD in particular. The proposed mapping algorithm is presented in section three. After a comparison to related work in section four, the paper ends with conclusion and remarks on further work.

Motivations and Context

Ontologies offer a great potential in higher education providing in particular the sharing and reusing of information across educational systems and enabling intelligent and personalized learner support. The increased functionality that ontologies imply will bring new opportunities to e-learning. Learners will be able to interact with distant educational systems easily and in a personalized way. An overview of ontologies for education field and an initial report on the development of an ontology-driven web portal O4E are presented in [3].
We have developed a WBES named SIMBAD based on a domain ontology. To facilitate resources exchange between SIMBAD and other WBES it becomes necessary to find solutions allowing the cooperation between various repositories of learning resources. The user may seek resources out of his/her private reference ontology. The problem is that the comprehension of a new classification (a new ontology) is expensive and does not constitute a justified investment. It is thus necessary to propose mechanisms to permit the user to access to resources of other repositories in a transparent way using his/her favourite WBES (and the associated shared reference ontology).

Ontologies for SIMBAD

This section presents the logical architecture of our system. A more detailed description is given in [2].
Our system is aimed at two categories of users that is authors of resources and learners. It is based on three models: (1) the domain model represented by an ontology which represent a normalized and common referential among all users of the system. This ontology is based on the ACM/CCS classification for the computer science domain [17]. This model will serve to semantically index the learner and the learning resources (2) the learner model is a view on the domain model. The set of learner knowledge is modeled by links to the domain model (3) the learning object model gives a semantic description of a learning object. In order to be found and re-used, a resource must be described by a set of metadata. We distinguish two types of metadata: the first one describes general characteristics of the resource (e.g., author, title, language, media) using LOM standard and the

second one describes the semantic of the resource. This semantic is structured in three parts and described in the same way as software components: prerequisites are the resource inputs (what is required by the resource) whereas content and acquisition function are its exits (what is provided by the resource).
A resource can be a set of web pages, a file or a program (a simulator for example). We just suppose that it is a unit accessible via an URI and we consider it as an instance of an ontology concept.

Algorithm Principles

In this paper we present an approach to combine different similarity measures to find mapping candidates between two ontologies.
This algorithm is based on three steps: the first step consists in generating information from the ontology. It uses the instances comparisons for deducing relations between concepts (convergence, divergence) of the same ontology. The second step calculates the similarity between candidate couples of concepts by using inference rules and rules derived by human experts. Before describing the algorithm, we need some more definitions and notations.

Definitions and Notations

φ-relation. *A φ-relation describes semantic properties among resources. These properties are presented in [1]. We define $\varphi : I \rightarrow I'$ where I and I' are sets of instances. Given r and r' any element of I and I' respectively, a φ-relation can be one of following relation types presented in table1 (pre(r) and cont(r) denote respectively prerequisites and content sets):*

Table 1. *φ-relation definition.*

φ-relation	Definition
substitution	r is substituted to r', if pre(r) = pre(r')
equivalence	r is equivalent to r', if r is substituted to r' and cont(r)= cont(r')
weak-precedence	r weakly-precedes r' if cont(r) $\subseteq$ pre(r')
strong-precedence	r strongly-precede r', if cont(r)=pre(r')

These new relations among resources will enrich the ontology relationships. This yields a corresponding series of definitions
Enriched Ontology Definition. An ontology O is a tuple O=(C, R, <, σ, ⊥, |) where (i) R and C denotes two disjoint sets called concept identifiers and relation identifiers respectively, (ii) < denotes a partial order on C called concept hierarchy or taxonomy, (iii) $\sigma : R \rightarrow C \times C$ denotes a function called signature that associate a relation to a couple of concepts, (iv) ⊥ denotes a relation called divergence between two concepts (i.e. there is no φ-relation between resources associated to

the concepts), (v) | denotes a relation called convergence between two concepts (i.e. there is at least one φ-relation between associated instances). We adopt the following logic notation presented in table 2 to express relations between concepts; c and d are two concepts:

Table 2. *Logical relation between concepts.*

Condition	Logical notation
$c < d$	$c \mapsto d$
$c \perp d$	$c,d \mapsto$
$c \mid d$	$\mapsto c,d$

Degree of Convergence. We define the degree of convergence notion to measure the convergence among concepts. Two concepts are more or less convergent depending upon the number of common resources.
The degree of convergence between two concepts c and d, noted $Đ_{c,d}$ is given in the following formula :

$$Đ_{c,d} = \frac{number\ of\ common\ insances\ between\ c\ and\ d}{\min(|c|,|d|)}$$

Where |c| (resp. |d|) is the total number of instances linked to the concept c (resp. d).

Ontology Morphism Definition. The comparison of two concepts in the same ontology is equivalent to the comparison of their images in a different ontology. For example, if c precedes d in the first ontology, their corresponding concepts F(c) and F(d) respect the same relation. This leads to the following definitions:
An ontology morphism between two ontologies O=(C, R, <, $\perp$, |, σ) and O'=(C', R', <', $\perp$', |', σ') is the couple of function (F,G) such that F : C → C' and G : R → R'
Given c and d two elements of C, the following relations are generated:

Table 3. *Morphism properties.*

Condition in O	New relation in O'
$c < d$	$F(c) <' F(d)$
$c \perp d$	$F(c) \perp' F(d)$
$c \mid d$	$F(c) \mid' F(d)$
$\sigma(r)=(c,d)$	$\sigma'(G(r))=(F(c),F(d))$

Mapping Process

Our mapping approach is based on multiple iterations. Different similarity measures are used by applying inference rules. In this section we describe the different steps of the mapping process after a brief definition of the ontology mapping.

Given two ontologies O and O', mapping one ontology onto another means that for each entity concept c in source ontology O, we try to find a corresponding concept c', which has the same intended meaning, in the target ontology O'.
The mapping process illustrated in Fig. 1 includes four main steps, starting with two ontologies, which are going to be mapped as its input: The derivation of ontology mappings takes place in a search of candidate mappings. The similarity computation determines similarity values of candidate mappings. Hypotheses are then generated using a rule base. This rule base contains a set of deductive rules which may be enriched with new rules proposed by domain experts. The "best" similarity hypothesis is selected. Each step can be repeated for multiple rounds and exchanges messages with previous step if necessary.

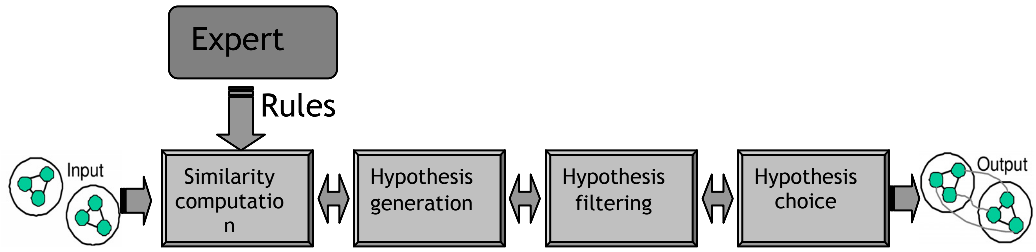

Fig. 1*. Different steps of mapping process*

Similarity Computation: The similarity computation is an iteration process. The first iteration consists in providing a basic similarity between concepts. In this iteration we use linguistic tools to compare concepts' names. In the i^{th} iteration we use the similarity produced in (i-1)th iteration and we apply the inference rules. These inference rules are either rules inferred from morphism ontological definition or rules proposed by the domain expert.

1st Iteration: Similarity Computation Using Linguistic Comparisons. In the first step, basic similarities are set via measures based on linguistic comparisons which are independent of the next similarities measures. Several ideas have been developed using concept names comparisons [11], dictionaries (e.g.WordNet [24]), identifiers such as URIs, etc.
We present below examples of methods and functions:

1. The use of existing tools based on dictionaries, like Nuno and Rocha in [15] who use WordNet to identify four type of relations between two concepts A and B:
   - $A \equiv B$ (i.e sim(A,B)=1.) if there exists a meaning of A synonym to a meaning of B
   - $A \supseteq B$ (i.e sim(A,B)=0.7) if there exists a meaning of A hyponym to a meaning of B
   - $A \subseteq B$ (i.e sim(A,B)=0.3) if there exists a meaning of B hyponym to a meaning of A
   - $A \perp B$ (i.e sim(A,B)=0.) if there is no relation between the meaning of A and the meaning of B
   - String equality :

$$\text{SimStrequ}(c,d) = \begin{cases} 1 \text{ if } c.char(i)=d.char(i)\ \forall\ i \in [0,|c|] \text{ with } |c| = |d| \\ 0 \text{ otherwise} \end{cases}$$

2. Similarity measure between two strings on a scale from 0 to 1

$$\text{SimStr}(c, d) = Max(0, \frac{Min(c,d) - ed(c,d)}{Min(c,d)})$$

Combining these methods will bring better results.

i[nd] iteration : Similarity Computation Using a Rule Base. After getting some relations between concepts based on linguistic solution, we use a rule base to find new similarities between ontologies. The rule base contains two sets of rules: a first set of basic deduction rules and a second set of rules proposed by the domain expert. Each rule shall give an indication on whether two concepts are similar but none provides for itself the mapping. The rules give only a similarity weight between two compared entities. A threshold is defined on the similarity values to determine the correspondence or the non-correspondence. Thanks to the ontology morphism definition, the set of deduction rules will generate, at the iteration i, new similarity relations from the iteration i-1.

For a progressive and dynamic similarity generation we define a similarity function F~: $C \times C' \to [0,1]$ which associate for each couple of concept a degree of similarity comprise between 0 and 1. The application of any comparison method (structural or semantic) will increase the similarity value and therefore the F~() value.

The following examples of deduction rules will illustrate the mechanism of similarity computation; c and d (respectively, c' and d') designs two concepts of the ontology O (respectively, O') and nbr-child(c) is the number of sub-concepts of c.

R1
IF F~(c,c') increases its value
THEN
$\forall (d,d') \in C \times C'$ such that ($c \mapsto d$ and $c' \mapsto d'$):
F~(d,d')= F~(d,d') + (F~(c,c')/nbr-child(c))

R2
IF F~(c,c')) increases its value
THEN
$\forall d' \in C'$ such that $\mapsto c', d'$: F~(c,d')= F~(c,d') + (F~(c,c')$\times$ $Đ_{c',d'}$)
&
$\forall d \in C$ such that $\mapsto c, d$: F~(c',d)= F~(c',d) + (F~(c,c')$\times$$Đ_{c,d}$)

Besides these rules, the domain expert can propose other rules to ameliorate the result quality. In the next iteration the overall similarities between concepts are calculated based on new similarity measures proposed by the expert and processed with an inference engine. Table 4 shows examples of rules for concepts comparisons. This set of rules may evolve dynamically. The manual effort is necessary because ontology mapping is too complex to be directly mapped by defaults rules.

Table 4. *Example of similarity rules between concepts.*

Rules	Description
R1	Two concepts are similar, if their names are similar
R2	Two concepts are similar, if their URI is similar
R3	Two concepts are similar, if their “father” concept are similar
R4	Two concepts are similar, if their “child” concept are similar
R5	Two concepts are similar, if their associated instances are similar

Hypothesis Generation. The hypotheses generation at iteration (i) is based on either the mapping set or the similarities generated at the iteration (i-1). We use the deduction rules and the comparison rules to propose new correspondences between concepts. Indeed, mapping hypotheses are generated for all couple of concepts depending on the similarity value (F~).

Hypothesis Filtering. During this step hypotheses which do not verify certain constraints (e.g. c⊥d and F(c) |’ F(d)) are removed. These are examples of rules which compare two hypotheses in order to eliminate the weakest one. Given two hypotheses hyp1 and hyp2 such that: hyp1: F~(c ;c’) and hyp2 : F~(d ;d’) with F~(d ;d’)>F~(d ;d’);

IF $c \mapsto d$ and $d' \mapsto c'$ **THEN** eliminate hyp2.
IF ($c, d \mapsto$ and $\mapsto c', d'$) or ($\mapsto c, d$ and $c', d' \mapsto$) **THEN** eliminate hyp2.

Furthermore, we can define a similarity threshold below which the hypotheses are not considered.

Hypothesis Choice. In this step, the hypotheses list provided by the filter is browsed and the best similarities are chosen. If none of the received hypothesis is selected, the precedent steps 3 and 4 are repeated for multiple rounds by decreasing the threshold.

In the last step, only the best similarities are considered in the final mapping table.

Prototype

We are currently implementing the different mapping process using a multi-agent system. We associate one agent to each step of the process (e.g. similarity computation).

Agent definition: An agent is a software component that has a role to play in the functioning of the system [13]. The degree of granularity is not equal for all agents: some of them play more important roles than others. An agent should have the ability of interacting with other agents and possibly humans (Expert) via an agent-communication language [9] [7]. Therefore the following performances can be reached:

- High performance: agents can run in parallel. They can be cloned when their work is too important;
- High flexibility: an agent can be programmed for any context; this means that the agent can directly interfaces different ontologies;
- High modularity: the number of interconnected sources can increases with no limits.

We use a JADE (Java Agent Development Framework) platform [25] to implement all the agents. JADE is conforming to the FIPA standard (Foundation for Intelligent Physical Agents). The agents communicate by exchanging messages in ACL language (Agent Communication Language) [16]. We include a rule-based inference engine called JESS [23] and to deal with ontologies and provide a programming environment for RDF, RDFS and OWL, we use the Jena framework [26] (A Semantic Web Framework for Java).

Related Work

Various works have been developed for supporting the mapping of ontologies. An interesting survey which gathered 35 works is presented in [10]. In [4,10] we can find other surveys on ontology alignment. In most approaches heuristics are described for identifying corresponding concepts in different ontologies, e.g. comparing the names or the natural language definition of two concepts, and checking the closeness of two concepts in the concept hierarchy.

PROMPT [12] is an algorithm for ontology merging and alignment based on identification of matching class names. A few approaches like RDFT [14] use the comparison of the resources to determine a similarity between concepts, but the problem is that the structures of all data instances are heterogonous. RDFT proposes an approach to the integration of product information over the web by exploiting the data model of RDF, which is based on directed label graphs. RDFT discovers a similarity between classes (concepts) based on the instance information for this class, using a machine-learning approach.

Like RDFT, GLUE [10] is a system which employs machine learning technologies to semi-automatically create mappings between heterogeneous ontologies. An ontology is considered here as a taxonomy of concepts and the problem of matching is reduced to: “for each concept node in one taxonomy, find the most similar node in the other taxonomy”. The problem of Glue is that the reliability of the results is related on the quantities and the degree of correction of all examples used by machine learning.

S-Match Semantic Matching [8] is an approach to matching classification hierarchies. The problem addressed by Semantic Matching is the following: say you have two different classification hierarchies, where each hierarchy is used to describe a set of documents, i.e. each term in the classification hierarchy describes a set of documents. How do the terms in one hierarchy relate to the terms in the other hierarchy? The proposed algorithm returns all possible similarities between both graphs based on synonyms sets from thesauri, using a SAT solver.

The tools described above offer mappings between heterogeneous ontologies. Most of them are based on syntactic and semantic matching heuristics given by an expert. None uses deduction rules which can be used for different application domains. Deduction rules offer more flexibility to the system. In addition, the closest work to our approach is QOM [5, 6] which is considered as a way to trade off between effectiveness and efficiency. One of the conclusions presented by the authors is "Using an approach combining many features to determine mappings, clearly leads to significantly higher quality mappings".
In our mapping approach, we try to use as much as possible available information contained in the ontology. This information consists of identifiers names of concept/relation, ontology structure, resources (concepts instances) and manual/automatic rules. Resources properties generate new semantic relations between concepts (concepts of the same ontology).

Conclusion

Nowadays, myriad of Web-Based Educational Systems exists, each of them storing their own learner's models and resources. Solutions have to be defined to open these systems to each others. Learners should be able to access to distant learning resources in a transparent way (without changing their usual reference ontology). Our objective is to be able to query different LOR and thus improving the interoperability of such systems.
In this paper we have introduced a mapping approach for bridging gaps between learning object repositories based on ontologies. This algorithm is applied on an existing WBES that allows learners and teachers searching, adding and composing new resources in a local repository. The particularity of the algorithm is that (i) it uses information on the resources to enrich the local ontology by generating relations between local concepts (ii) it is based on inference rules. Some of them are basic ones; others can be added by a domain expert. This flexibility permits its application to other domains.
The prototype is based on multi-agents technology. It is implemented with the JADE platform and the Jess rule-based reasoning engine. In future work, we plan to add other match and techniques in order to resolve more complex mapping problems (e.g. cardinality n:m).

References

[1] Bouzeghoub A., Defude B., Ammour S., Duitama J.F., Lecocq C.: A RDF Description Model for Manipulating Learning Objects, Proc. IEEE International Conference on Advanced Learning Technologies, Joensuu, Finland, August 2004
[2] Bouzeghoub, A., Carpentier, C., Defude, B., Duitama, J.F.: A Model of Reusable Educational Components for the Generation of Adaptive Courses. Proc. Of the International Workshop on Semantic Web for Web-based Learning, with CAISE'03 Conference, Klagenfurt, Austria, June 2003

[3] Dicheva1, D., Sosnovsky, S., Gavrilova, T., Brusilovsky, P.: Ontological Web Portal for Educational Ontologies. International Workshop on Applications of Semantic Web technologies for E-Learning (SW-EL'05) july 2005
[4] Euzenat J., Le Bach, T., Barrasa J., Bouquet P., De Bo J., Dieng-Kuntz R., Ehrig M., Hauswirth M., Jarrar M., Lara R., Maynard D., Napoli A., Stamou G., Stuckenschmidt H., Shvaiko P., Tessaris S., Van Acker S., Zaihrayeu I.: State of the art on ontology alignment. Deliverable 2.2.3, IST Knowledge Web NoE, 21p., June 2004
[5] Ehrig, M., Sure Y.: Ontology Mapping - An Integrated Approach April 21,2004, ESWS 2004: 76-91
[6] Ehrig, M., and Staab, S., QOM— Quick Ontology Mapping, ISWC2004 7-11 Nov. 2004, Hiroshima, Japan
[7] Franklin S. and Graesser A.: Is it an Agent, or just a Program? A Taxonomy for Autonomous Agents. Proceedings of the Third International Workshop on Agent Theories, Architectures, and Languages, Springer-Verlag, Berlin, Germany, 1996.
[8] Giunchiglia, F., Shvaiko P.: Semantic matching. The Knowledge Engineering Review, 18(3):265-280, 2004.
[9] Jennings, N. R.: On agent-based software engineering. Artificial Intelligence, 2000.
[10] Kalfoglou Y., Schorlemmer M.,Ontology mapping: the state of the art, The Knowledge Engineering Review 18(1):1--31, January 2003
[11] Levenshtein, I.V.: Binary codes capable of correcting deletions, insertions, and reversals.Cybernetics and Control Theory (1966)
[12] Natalya, F., Noy, A., Mark, A.: Musen. Ontology versioning as an element of an ontology-management framework. To be published in IEEE Intelligent Systems, 2003.
[13] Nwana, H..S.: Software Agents: An Overview. Knowledge Engineering Review 1996.
[14] Omelayenko, Borys.: Integrating vocabularies. Discovering and representing vocabulary maps. In Proceedings of the First International Semantic Web Conference (ISWC2002), Sardinia, Italy, 2002.
[15] Silva, N., Rocha, J.A.: Ontology mapping for interoperability in semantic web. In Proceedings of the IADIS International Conference WWW/Internet 2003 (ICWI'2003), Algarve, Portugal, 2003.
[16] Simon, B., Massart, D., Van Asche, F., Ternier, S., Duval, E., Brantner, S., Olmedilla, D., Miklós, S.: A Simple Query Interface for Interoperable Learning Repositories. WWW 2005, may 10–14, 2005, Chiba, Japan

Web references

[17] The ACM Computing Classification System (1998), available from http://www.acm.org/class/1998/ccs98.html accessed in feb. 2006
[18] ARIADNE, Ariadne foundation for the European Knowledge Pool, available from http://www.ariadne-eu.org/ accessed in feb. 2006
[19] EducaNext, available from http://www.educanext.org/ accessed in feb. 2006
[20] IEEE. Draft Standard for Learning Object Metadata. (IEEE P1484.12.1.), available from <http://ltsc.ieee.org> accessed in feb 2006
[21] Edusource http://www.edusource.ca/ accessed in feb. 2006
[22] Edutella http://edutella.jxta.org/ accessed in feb. 2006
[23] Jess http://herzberg.ca.sandia.gov/jess/index.shtml
[24] Wordnet http://www.wordnet.com
[25] Jade http://jade.tilab.com/
[26] Jena http://jena.sourceforge.net/

Biographies

Amel Bouzeghoub received a degree of Ph.D. in Computer Sciences at Pierre et Marie Curie University, (France). She is currently an associate professor in the department of Computer Science at GET-INT (Institut National des Télécommunications, Evry, France). Her research interests include topics related to semantic web, knowledge representation (ontologies and their interoperability, knowledge sharing and reuse), and personalized web-based learning systems. She is currently involved in the AIDA virtual multidisciplinary research team.

Abdeltif Elbyed received his M.Sc. degree in Computer Science from the University of Evry (France). He is currently a PhD student at GET-INT (Institut National des Télécommunications, Evry, France). His research interests include distributed ontology engineering, data management and agent-based technology.