A Method for Integration of WordNet-Based Ontologies Using Distance Measures

Trong Hai Duong[1], Ngoc Thanh Nguyen[2], and Geun Sik Jo[1]

[1] School of Computer and Information Engineering,
Inha University, Korea
haiduongtrong@gmail.com, gsjo@inha.ac.kr
[2] Institute of Information Science and Engineering,
Wroclaw University of Technology, Poland
thanh@pwr.wrov.pl

Abstract. While there is a large body of previous work focused on WordNet-based for finding the semantic similarity of concepts and words, the application of these word oriented methods to ontology integration tasks has not been yet explored. In this paper, we propose a methodology of WordNet-based distance measures, and we apply the meaning of concepts of upper ontologies to an ontology integration process by providing semantic network called *OnConceptSNet*. It is a semantic network of concepts of ontologies in which relations between concepts derived from upper ontology WordNet. We also describe a methodology for conflict in ontology integration process.

Keywords: knowledge integration, information system, ontology integration.

1 Introduction

Ontology integration is an important task which needs to be performed when several information systems share or exchange their knowledge. Because ontology in these systems is a separated element of their knowledge bases, the knowledge integration process very often begins with ontology integration.

Basically, the ontology is defined by the following elements:

- ***C*** - a set of concepts (classes),
- ***I*** - a set of instances of concepts,
- ***R*** - a set of binary relations defined on ***C***,
- ***Z*** - a set of axioms, which can be interpreted as integrity constraints or relationships between instances and concepts.

In general, the problem of ontology integration can be formulated as follows: *For given ontologies $O_1, \ldots, O_n$ one should determine one ontology which could replace them* [3, 10]. Integration of ontologies is such a complex task, since ontologies have various characteristics and forms by nature, i.e., languages, domains, structures of ontologies may differ from each other. Therefore, authors of [4] suggested an Ontology Architecture, which provide a solid basis for studies about

I. Lovrek, R.J. Howlett, and L.C. Jain (Eds.): KES 2008, Part I, LNAI 5177, pp. 210–219, 2008.

ontology integration task. Pinto and Martins [10] identified the activities that should be performed in the ontology integration process. One of the first tools, PROMPT [8] helps in the merge process are now available. It uses labels to extent the structure of ontologies. Their focus is on ontology merging, i.e., how to create one ontology from two source ontologies. Most of the ideas for ontology integration tasks deal with upper ontologies as domain specific of the ontology [2]. The upper ontologies not only provide definitions for general-purpose terms [7, 12], but also extend them as semantic domain layering of the ontology architecture [4]. However, all these approaches often lack the specific application for ontology integration task and significant testing.

In our study, we apply the meaning of concepts of the upper ontologies to ontology integration process by providing semantic network called *OnConceptSNet*. It is a semantic network of concepts of ontologies in which relations between concepts derived from upper ontology WordNet. We propose a methodology for WordNet-based distance measures between the concepts. We also describe a methodology for conflict in ontology integration process.

2 Definitions

2.1 Basic Notions

As stated in above section, by an ontology we understand a quadruple: ($\boldsymbol{C}$, $\boldsymbol{I}$, $\boldsymbol{R}$, $\boldsymbol{Z}$). We assume a real world ($\boldsymbol{A}$, $\boldsymbol{V}$) where $\boldsymbol{A}$ is a finite set of attributes and $\boldsymbol{V}$ is the domain of $\boldsymbol{A}$, that is $\boldsymbol{V}$ is a set of attribute values, and $\boldsymbol{V} = \bigcup_{a\in\boldsymbol{A}} \boldsymbol{V}_a$ ($\boldsymbol{V}_a$ is the domain of attribute a). In this paper, we accept the following assumptions:

1. A concept is defined as a triple:

$$concept = (c, A^c, V^c) \tag{1}$$

where c is a unique name of the concept, $A^c \subseteq \boldsymbol{A}$ is a set of attributes describing the concept and $V^c \subseteq \boldsymbol{V}$ is the attributes' domain: $V^c = \bigcup_{a\in A^c} V_a$. Pair (A^c, V^c) is called the structure of concept c.

2. An instance of a concept c is described by the attributes from set A^c with values from set V_c. Thus, an instance of a concept c is defined as a pair:

$$instance = (id,\ v) \tag{2}$$

where id is a unique identifier of the instance in world (A, V) and v is the value of the instance, which is a tuple of type A^c. All instances of the same concept in an ontology are different with each other.

By $Ins(O, c)$ we denote the set of instances belonging to concept c in ontology O. We have

$$\boldsymbol{I} = \bigcup_{c\in C} Ins(O, c) \tag{3}$$

2.2 Similarity

We present a formal definition of similarity method which derived from [2] as follows: let x, y, z are entities, value of $sim(x, y)$ represents the semantic similarity between x and y.

1. $sim(x, y) \in [0, 1]$
2. if $sim(x, y) = 1$ then $y = x$ or x semantic equivalent y.
3. $sim(x, y) = 0$: two objects are disjoint, i.e., no common characteristics.
4. $sim(x, y) = sim(y, x)$: similarity is symmetric
5. $sim(x, z) \leq (sim(x, y) + sim(y, z))$: The triangular inequation is valid for the similarity measure

However, when we apply similarity characteristics to find similarity between two structures of concepts, we reject characteristics 2, and 4 to satisfy with overlap characteristic of two concepts which mentioned below section.

2.3 Problem

In this section, we present the ontology integration process by providing semantic network of concepts of ontologies, called *OnConceptSNet*. The *OnConceptSNet* builds or extends their representations by acquiring knowledge from WordNet-Base and *static rules*. The knowledge may change the old network by adding and deleting nodes and arcs or by modifying numerical values (similar) or type arcs (relation), called weights, associated with the arcs.

The *OnConceptSNet* is defined as a graph:

$$\boldsymbol{G} = (\boldsymbol{C}^*, \boldsymbol{R}^*) \tag{4}$$

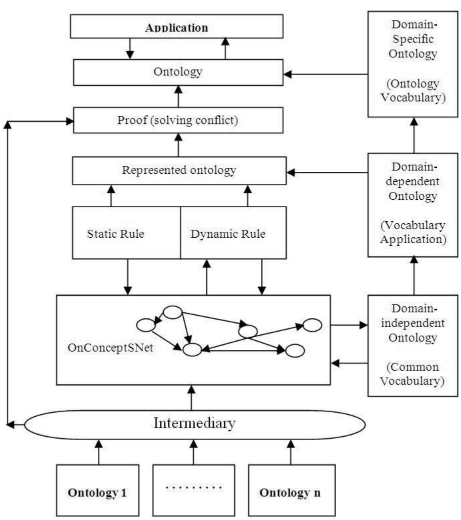

Fig. 1. Ontology integration process

$\boldsymbol{C}^*$ *is a set of nodes representing concepts that come from* $O_1, \ldots, O_n$.
$\boldsymbol{R}^*$ *is a set of arcs representing relations between concepts: semantic equivalent* ($\Leftrightarrow$)*, more general* ($\sqsupseteq$)*, less general* ($\sqsubseteq$)*, overlap* ($\frown$)*. Each arc is associated by a numerical similar value between two concepts.*

Here we denote S_i, l_i, and L_i corresponding to structure, name, and label of concept c_i, where the label of concept is either its name or its comment, or its label. We define the relations on *OnConceptSNet* as follow:

1. $c_1 \frown c_2$, that must be satisfied with one of conditions:
   - $sim(S_1, S_2) = 1$, and $sim(S_2, S_1) = 1$.
   - S_1, and S_2 are empty.
2. $c_1 \Leftrightarrow c_2$, that must be satisfied with one of conditions:
   - $sim(c_1, c_2) = 1$
   - $sim(L_1, L_2) > 0$, and $c_1 \frown c_2$.
   - $sim(L_1, L_2) > 0$, and *super-concept of* $c_1 \Leftrightarrow$ *super-concept of* c_2
   - $sim(L_1, L_2) > 0$, and *sub-concept of* $c_1 \Leftrightarrow$ *sub-concept of* c_2
3. $c_1 \sqsupseteq c_2$, that must be satisfied with one of conditions:
   - $sim(L_1, L_2) > 0$, $sim(S_2, S_1) = 1$, and $sim(S_1, S_2) < 1$
   - l_1 is a hyponym of l_2, $sim(S_2, S_1) = 1$, and $sim(S_1, S_2) < 1$
   - l_2 is a hypernym of l_1, $sim(S_2, S_1) = 1$, and $sim(S_1, S_2) < 1$
4. $c_1 \sqsubseteq c_2$, that of $c_2 \sqsupseteq c_1$.

We apply the upper ontologies as semantic domain layering. The idea of domain-independent ontologies provides basic concepts and relations to build the semantic network of concepts of ontologies which we call *OnConceptSNet*.

Represented ontology is an ontology which represents candidate ontologies. It is derived from *OnConceptSNet*.

Rules include *static rules*, and *dynamic rules*. The main *static rules* that we mentioned above to create *OnConceptSNet*. *Dynamic rules* are used to reduce or extend *OnConceptSNet* into *Represented ontology*, the rules as follows:

1. Rules for Concept:
   - if $c_1 \Leftrightarrow c_2$ then *delete* c_1
   - if $c_1 \sqsupseteq c_2 \wedge \exists\, c_1 \Leftrightarrow c_3$, where c_2 *is a sub-class* c_3 then *delete* c_1
   - if $(c_1 \sqsupseteq c_2) \wedge \neg\exists\, c_1 \Leftrightarrow c_3$, where c_2 *is a sub-class* c_3 then c_2 *sub-class* c_1
   - if $c_1 \sqsubseteq c_2 \wedge \exists\, c_1 \Leftrightarrow c_3$, where c_2 *is a sup-class* c_3 then *delete* c_1
   - if $c_1 \sqsubseteq c_2 \wedge \neg\exists\, c_1 \Leftrightarrow c_3$, where c_2 *is a sup-class* c_3 then c_2 *sup-class* c_1
2. Rules for property, the symbols differ in above-mentioned, $p_1 \Leftrightarrow p_2$ (p_1 *similarity* p_2), $\sqsubseteq$ (*hypernym/holonym*), $\sqsupseteq$ (*hyponym/meronym*), $\perp$ (*antonym*):
   - if $p_1 \Leftrightarrow p_2$ then *delete* p_2 (eg., $job \Leftrightarrow occupation$)
   - if $p_1 \sqsubseteq p_2$ then *delete* p_1 (eg., $age \sqsubseteq birthday$)
   - if $p_1 \sqsupseteq p_2$ then *delete* p_2 (eg., $sex \sqsupseteq female$)
   - if $p_1 \perp p_2$ then *delete* p_1 (eg., $single \perp married$)

Proof (solving conflict) contacts candidate ontologies, and the *represented ontology* to proof conflict in the *represented ontology*. After solving conflict, the *represented ontology* becomes an *ontology* that replaces the candidate ontologies.

Intermediary plays an intermediary role to connect the candidate ontologies and the *OnConceptSNet*. It translates the candidate ontologies into synchronous.

3 Ontology Conflict and Integration

3.1 Conflicts on Instance Level

At this level we assume that 2 ontologies differ from each other only in values of instances. That means they may have the same concepts and relations.

Definition 1. Let O_1 and O_2 be $(\boldsymbol{A},\ \boldsymbol{V})$-based ontologies. Let concept (c, A^c, V^c) belong to both ontologies and let the same instance i belong to concept c in each ontology, that is $(i, v_1) \in Ins(O_1, c)$ and $(i, v_2) \in Ins(O_2, c)$. We say that a conflict takes place if $v_1 \neq v_2$.

For solving conflicts of ontologies on instance level, consensus methods seem to be very useful. Different criteria, structures of data and algorithms have been worked out [5, 6]. For this kind of conflict, the consensus problem can be defined:

Given a set of values $X = \{v_1, \ldots, v_n\}$ where v_i is a tuple of type A^c, that is:

$$v_i : A^c \to V^c \tag{5}$$

for $i = 1, \ldots, n$; $A^c \subseteq \boldsymbol{A}$ and $V = \bigcup_{a \in A^c} V_a$ one should find tuple v of type A, such that one or more selected postulates for consensus are satisfied [6].

One of very popular postulate requires minimizing the following sum.

$$\sum_{i=1}^{n} d(v, v_i) = \min_{v' \in T(A^c)} \sum_{i=1}^{n} (v', v_i) \tag{6}$$

where $T(A^c)$ is the set of all tuples of type A^c.

3.2 Conflicts on Concept Level

At this level we assume that two ontologies differ from each other in the structure of the same concept. That means they contain the same concept but its structure is different in each ontology.

Definition 2. Let O_1 and O_2 be $(\boldsymbol{A},\ \boldsymbol{V})$-based ontologies. Let concept (c_1, A^{c_1}, V^{c_1}) belong to O_1 and concept (c_2, A^{c_2}, V^{c_2}) belong to O_2. We say that a conflict takes place in concept level if $c_1 = c_2$ but $A^{c_1} \neq A^{c_2}$ or $V^{c_1} \neq V^{c_2}$.

Definition 2 specifies such situations in which two ontologies define the same concept in different ways. For example, concept person in one system may be defined by attributes: *Name, Age, Address, Sex, Job*, while in another system it is defined by attributes: *Id, Name, Address, Date_of_birth, Taxpayer identification number, Occupation*.

The problem is the following: For given, a set of pairs $X = \{(A^i, V^i) : (A^i, V^i)$ is the structure of concept c in ontology O_i for $i = 1, \ldots, n\}$, it is needed to determine a pair (A^*, V^*) which at best represents the given pairs.

Words "at best" mean one or more postulates for satisfying by pair (A^*, V^*).

4 Distance Measures

4.1 WorkNet-Base Similarity between Two Words

The Palmer and Wu [9] similarity metric measures the depth of the two concepts in the WordNet taxonomy, and the depth of the least common subsumer. Resnik [11] defines the similarity between two words as the information content of the lowest superordinate in the hierarchy, defining the information content of a concept c (where a concept is the WordNet class containing the word). The Lesk similarity [1] of two concepts is defined as a function of the overlap between the corresponding glosses and those that surround it in the given context.

Our purpose is to apply similarity measure between words for ontology integration tasks. This similarity degree depends on complex candidate ontologies. So another of our approach for WorkNet-base similarity of two words is proposed as follows: the words occur together in a synset, they have the synonym relation with each other in context of a gross. For example, the words *learner* occurs in two noun synsets {*learner, scholar, ass-imilator*} and {*apprentice, learner, prentice*}*; student* occurs in two noun synsets {*student, pupil, educate* } and {*scholar, s-cholarly person, bookman, student*}. Thus, *scholar* is a common word of a *student*'s synset and a *learner*'s synset, so *student* and *learner* have a relation, if we continue finding synonym of words in student's synsets and in learner's synsets, the number of similar words occurs together with *student* and *learner* may be much larger. That means the similarity degree between *student* and *learner* is quite larger. Moreover, each word occur in many synsets that cross part of speech. For example, the word *base* occurs in 7 adjective synsets, 3 verb synsets, and 19 noun synsets, that means the similarity acrosses part of speech. For these reasons, we proposed a formulate for measuring the semantic similarity of words as follows:

$$sim(w_1, w_2) = \max_{level=1,\ldots,n}\left(\frac{\triangle + \sum_{w_i \in Syn_1 \cap E}(\sum_{w_j \in Syn_2 \cap E} Inc(w_i, w_j))}{min(size(Syn_1), size(Syn_2)) + size(E)}\right) \quad (7)$$

where

$$Inc = \begin{cases} 0 & \text{if } w_i \neq w_j \\ 1 & \text{if } w_i = w_j \end{cases}$$

If $Inc(w_1, w_2) = 1$ then $E = E \cup \{w_1\}$.
$\triangle$ is total return value of *Inc* at *level*=1..*k*-1, *k* is current *level.*

The *level* will be increased from 1 to *n*, each increasing time of *level* then,

$Syn_1 = \bigcup_{w \in Syn_1 \cap E} Synonym(w)$ and $Syn_2 = \bigcup_{w \in Syn_2 \cap E} Synonym(w)$.

We experiment with the method to find out similarity between 100 pairs of words with different similarity degree and crossing part of speech, we chose *level* is equal to 3, and limit of size of array *Syn* is 1000, most of similarity between words is found out. Please note that, the more we increase level, the more similarity between words is increased. For example, similarity between *learner* and *student* at *level* 1, 2, 3 corresponding to 0.24, 0.65, 0.84.

4.2 Similarity between Two Properties

p_i is representation identification of property i,
$R_i = \{r_1, r_2, \ldots, r_n\}, r_j$ is a name/value of instance j of priperty p_i
$A_i = \{a_1, a_2, \ldots, a_k\}, a_j$ is a either single word or compound word come from r_i
$G_i = \{g_1, g_2, \ldots, g_m\}, G_i$ is set of more general words of $a_j \in A_i, j = 1, 2, \ldots, k$.
The words of set G_i which come from WordNet through HYPERNYM.

$$H = \bigcup_{i=1}^{n} (G'_i) \tag{8}$$

where $G'_i \subseteq G_i$ and if $g_j \in G'_i, g_j$ exists in at least $\frac{1}{2}n$ sets $G_i, i = 1, 2, \ldots, n$

$$sim(H_1, H_2) = \frac{\sum_{a \in H_1} (max(sim_{b \in H_2}(a, b)))}{min(size(H_1), size(H_2))} \tag{9}$$

Similarity between two properties

$$sim(p_1, p_2) = max(sim(L_1, L_2), sim(H_1, H_2)) \tag{10}$$

where $sim(L_1, L_2)$ is similarity between two labels of properties p_1 and p_2.

4.3 Similarity between Two Concepts

c_i is representation identification of concept i,
$C_i = \{l_1, l_2, \ldots, l_n\}, l_i$ is a name/label of instance i of concept c_i
$A_i = \{a_1, a_2, \ldots, a_k\}, a_j$ is a either single word or compound word come from l_i
$G_i = \{g_1, g_2, \ldots, g_k\}, G_i$ are set of more general words of $a_j \in A_i$. Those words of set which come from WordNet through HYPERNYM.

$$H = \bigcup_{i=1}^{n} (G'_i) \tag{11}$$

where $G'_i \subseteq G_i$ and if $g_j \in G'_i, g_j$ exists in at least $\frac{1}{2}n$ sets $G_i, i = 1, 2, \ldots, n$

$$M = \bigcup_{i=1}^{n} (A'_i) \tag{12}$$

where $A'_i \subseteq A_i$ and if $a_k \in A'_i$ then exist at least $g_j \in G_i$ and g_j is general word of a_k.

We define similarity between 2 structures of two concepts as follows:
S is representation of Structure of concept. $S = \{p_1, p_2, \ldots\}$, where $p_i, i = 1, 2, \ldots, n$ is properties of concept

$$sim(S_1, S_2) = \frac{\sum_{p \in S_1} (max(sim_{p' \in S_2}(p, p')))}{size(S_2)} \tag{13}$$

Similarity between 2 concepts c_1 and c_2

$$sim(c_1, c_2) = max(\frac{sim(L_1, L_2) + sim(S_1, S_2)}{2}, \frac{sim(H_1, H_2) + sim(S_1, S_2)}{2}, \frac{sim(M_1, M_2) + sim(S_1, S_2)}{2}) \quad (14)$$

where $sim(L_1, L_2)$ is similarity between two labels of concepts c_1 and c_2.

5 An Algorithm for Ontology Integration

- Input: n candidate ontologies $O_1, \ldots, O_n$.
- Output: Ontology O^* that replaces $O_1, \ldots, O_n$.

Begin

1. *Intermediary* translates $O_1, \ldots, O_n$ with language $L_1, \ldots, L_n$ into L_0;
2. Create *OnConceptSNet* for $O_1, \ldots, O_n$ pass WordNet-Based;
   - Find similarity between properties of $O_1, \ldots, O_n$;
   - Find similarity between concepts of $O_1, \ldots, O_n$;
   - Create relation between concepts of $O_1, \ldots, O_n$ on *OnConceptSNet* pass *static rules* and similarity between concepts;
3. Create *dynamic rules* that have states come from *OnConceptSNet*;
4. Execute *dynamic rules* for reducing *OnConceptSNet* to *represented ontology* that best represents n candidate ontologies;
5. Execute algorithm for solving conflict in *represented ontology*;
6. Compute O^*, to build the domain-dependent, and domain-specific ontology;
7. Return O^*;

End.

6 Experiments

We used four data sets, each consisting of at least two ontologies which we refer to [2] http://www.aifb.uni-karlsruhe.de/WBS/meh/foam/ontologies.htm for evaluation purposes. From their differences, we expect a representative evaluation. Because of limited space, we only present some compares with author's result [2] (see the table 1, and table 2).

Our purpose is to integrate ontologies of the information systems with knowledge bases which have to be integrated when they want to share or exchange their knowledge. These information systems' knowledge bases include ontologies and their instances. Therefore we assume that there are enough instances for our similarity finding method. However, we have to note that our method is sufficient not only in environments with many instances but also in environments with lack of instanes. An example, although the ontologies of authors [2] are really less instances, but our results are still sufficient. In this paper, we only show some experiment results from which is stressed the role of WordNet-Based as independent-domain ontologies is stressed. Our complete system for ontology integration will be shown in our future work.

Table 1. Some results comparing between our similarity and in [2]

Property 1	*Property 2*	*Author's sim[2]*	*Our sim*
russia#music	meh://8807#music	1.0	1.0
russia#cost_money_eating	meh://8807#cost_money	0,9473	0.98
russia#include_city	meh://8807#include_town	0,9167	1.0
animalsA.owl#hasMaleParent	animalsB.owl#hasFather	1.0	1.0
animalsA.owl#hasFemaleParent	animalsB.owl#hasMother	1.0	1.0
...	...	...	...

Table 2. Some results comparing between our relation and similarity in [2]

Concept 1	*Concept 2*	*Author's sim[2]*	*Our relation*
animalsA.owl#Woman	animalsB.owl#Person	?	sub-class
animalsA.owl#HumanBeing	animalsB.owl#Man	?	sup-class
animalsA.owl#HumanBeing	animalsB.owl#Person	1.0	equivalent
animalsA.owl#TwoLeggedPerson	animalsB.owl#BipedalPerson	1.0	equivalent
animalsA.owl#TwoLeggedThing	animalsB.owl#BipedalThing	1.0	equivalent
...	...	...	...

7 Conclusions

In this paper, we built the semantic network, called OnConceptSNet which is derived from the upper ontologyWordNet to integrate multiple ontologies as reconcile semantic conflicts between the ontologies. We designed the semantic similarities between ontology elements using WordNet-Based. We also described a methodology to solve the conflict in ontology integration process. In future work, we will approach dynamic inference rules for ontology integration tasks that derive and aggregate relation between attributes, instances, concepts, and to insert, remove a derived object when the condition of the deductive rule is satisfied. We will also applied consensus theory for solving conflict in relation level and restriction level. Finally, we will build a auto-ontology integration system.

References

1. Banerjee, S., Pedersen, T.: An adapted Lesk algorithm for word sense disambiguation using Word- Net. In: Proceedings of the Third International Conference on Intelligent Text Processing and Computational Linguistics, Mexico City, Mexico, pp. 136–145 (2002)
2. Ehrig, M., Sure, Y.: Ontology mapping - an integrated approach. In: First European Semantic Web Symposium, pp. 76–91 (2004)
3. Gangemi, A., Pisanelli, D.M., Steve, G.: Ontology Integration: Experiences with Medical Terminologies. In: Guarino, N. (ed.) Formal Ontology in Information Systems, pp. 163–178. IOS Press, Amsterdam (1998)

4. Jeongsoo, L., Heekwon, C., Kwangsoo, K., Cheol-Han, K.: An Ontology Architecture for Integration of Ontologies. In: Processing The Semantic Web - ASWC, pp. 205–211 (2006)
5. Nguyen, N.T.: Processing Inconsistency of Knowledge on Semantic Level. Journal of Universal Computer Science (2), 285–302 (2005)
6. Nguyen, N.T.: Advanced Methods for Inconsistent Knowledge Management. Springer, London (2008)
7. Niles, I., Pease, A.: Towards a standard upper ontology. In: Welty, C., Smith, B. (eds.) FOIS 2001: Proceedings of the international conference on Formal Ontology in Information Systems, New York, NY, USA, pp. 2–9 (2001)
8. Noy, N.F., Musen, M.A.: PROMPT: Algorithm and Tool for Automated Ontology Merging and Alignment. In: AAAI 2000 Proceedings, pp. 450–455 (2000)
9. Palmer, M., Wu: Verb semantics and lexical selection. In: Proceedings of the 32nd Annual Meeting of the Association for Computational Linguistics, Las Cruces, New Mexico, pp. 133–138 (1994)
10. Pinto, H.S., Martins, J.P.: A Methodology for Ontology Integration. In: Proceedings of the First International Conference on Knowledge Capture, pp. 131–138. ACM Press, New York (2001)
11. Resnik: Using information content to evaluate semantic similarity. In: Proceedings of the 14th International Joint Conference on Artificial Intelligence, Montreal, Canada, pp. 448–453 (1995)
12. Semy, S.K., Pulvermacher, M.K., Obrst, L.J.: Toward the use of an upper ontology for U.S. government and U.S. military domains: An evaluation. Technical Report MTR 04B0000063, The MITRE Corporation (2004)