Term-Based Ontology Alignment

Virach Sornlertlamvanich, Canasai Kruengkrai, Shisanu Tongchim,
Prapass Srichaivattana, and Hitoshi Isahara

Thai Computational Linguistics Laboratory
National Institute of Information and Communications Technology
112 Paholyothin Road, Klong 1, Klong Luang, Pathumthani 12120, Thailand
{virach,canasai,shisanu,prapass}@tcllab.org, isahara@nict.go.jp

Abstract. This paper presents an efficient approach to automatically align concepts between two ontologies. We propose an iterative algorithm that performs finding the most appropriate target concept for a given source concept based on the similarity of shared terms. Experimental results on two lexical ontologies, the MMT semantic hierarchy and the EDR concept dictionary, are given to show the feasibility of the proposed algorithm.

1 Introduction

In this paper, we propose an efficient approach for finding alignments between two different ontologies. Specifically, we derive the source and the target ontologies from available language resources, i.e. the machine readable dictionaries (MDRs). In our context, we consider the ontological concepts as the groups of lexical entries having similar or related meanings organized on a semantic hierarchy. The resulting ontology alignment can be used as a semantic knowledge for constructing multilingual dictionaries.

Typically, bilingual dictionaries provide the relationship between their native language and English. One can extend these bilingual dictionaries to multilingual dictionaries by exploiting English as an intermediate source and associations between two concepts as semantic constraints.

Aligning concepts between two ontologies is often done by humans, which is an expensive and time-consuming process. This motivates us to find an automatic method to perform such task. However, the hierarchical structures of two ontologies are quite different. The structural inconsistency is a common problem [1]. Developing a practical algorithm that is able to deal with this problem is a challenging issue.

The rest of this paper is organized as follows: Section 2 discusses related work. Section 3 provides the description of the proposed algorithm. Section 4 presents experimental results and findings. Finally, Section 5 concludes our work.

2 Related Work

Chen and Fung [2] proposed an automatic technique to associate the English FrameNet lexical entries to the appropriate Chinese word senses. Each FrameNet

Universal Network Language: Advances in Theory and Applications.
Research on Computing Science 12, 2005, pp. 138–144.

lexical entry is linked to Chinese word senses of a Chinese ontology database called HowNet. In the beginning, each FrameNet lexical entry is associated with Chinese word senses whose part-of-speech is the same and Chinese word/phrase is one of the translations. In the second stage of the algorithm, some links are pruned out by analyzing contextual lexical entries from the same semantic frame. In the last stage, some pruned links are recovered if its score is greater than the calculated threshold value. Ngai *et al.* [3] also conducted some experiments by using HowNet. They presented a method for performing alignment between HowNet and WordNet. They used a word-vector based method which was adopted from techniques used in machine translation and information retrieval. Khan and Hovy [4] presented an algorithm to combine an Arabic-English dictionary with WordNet. Their algorithm also tries to find links from Arabic words to WordNet first. Then, the algorithm prunes out some links by trying to find a generalization concept.

3 The Algorithm

In this section, we describe an approach for ontology alignment based on term distribution. To alleviate the structural computation problem, we assume that the considered ontology structure has only the hierarchical (or taxonomic) relation. One may simply think of this ontology structure as a general tree, where node of each tree is equivalent to a concept.

Given two ontologies called the source ontology $\mathcal{T}_s$ and the target ontology $\mathcal{T}_t$, our objective is to align all the concepts (or semantic classes) between these two ontologies. Each ontology consists of concepts, denoted by $\mathcal{C}_1, \ldots, \mathcal{C}_k$. In general, the concepts and their corresponding relations of each ontology can be significantly different due to the theoretical background used in the construction process. However, for the lexical ontologies such as the MMT semantic hierarchy and the EDR concept dictionary, it is possible that the concepts may contain shared members in terms of English words. Thus, we can match the concepts between two ontologies using the similarity of the shared words.

In order to compute the similarity between two concepts, we must also consider their related child concepts. Given a root concept $\mathcal{C}_i$, if we flatten the hierarchy starting from $\mathcal{C}_i$, we obtain a nested cluster, whose largest cluster dominates all subclusters. As a result, we can represent the nested cluster with a feature vector $\mathbf{c}_i = (w_1, \ldots, w_{|\mathcal{V}|})^T$, where features are the set of unique English words $\mathcal{V}$ extracted from both ontologies, and w_j is the number of the word j occurring the nested cluster i. We note that a word can occur more than once, since it may be placed in several concepts on the lexical ontology according to its sense.

After concepts are represented with the feature vectors, the similarity between any two concepts can be easily computed. A variety of standard similarity measures exists, such as the *Dice coefficient*, the *Jaccard coefficient*, and the *cosine* similarity [5]. In our work, we require a similarity measure that can reflect the degree of the overlap between two concepts. Thus, the Jaccard coefficient

Algorithm 1: OntologyAlignment

input : The source ontology $\mathcal{T}_s$ and the target ontology $\mathcal{T}_t$.
output : The set of the aligned concepts $\mathcal{A}$.

begin
  Set the starting level, $l \leftarrow 0$;
  while ${\mathcal{T}_s}^{\langle l \rangle} \leq {\mathcal{T}_s}^{\langle max \rangle}$ **do**
    Find all child concepts on this level, $\{\mathcal{C}_i\}_{i=1}^{k} \in {\mathcal{T}_s}^{\langle l \rangle}$;
    Flatten $\{\mathcal{C}_i\}_{i=1}^{k}$ and build their corresponding feature vectors, $\{\mathbf{c}_i\}_{i=1}^{k}$;
    For each $\mathbf{c}_i$, find the best matched concepts on $\mathcal{T}_t$,
      $\mathcal{B} \leftarrow$ FindBestMatched($\mathbf{c}_i$);
      $\mathcal{A} \leftarrow \mathcal{A} \cup \{\mathcal{B}, \mathcal{C}_i\}$;
    Set $l \leftarrow l + 1$;
  end
end

Algorithm 2: FindBestMatched($\mathbf{c}_i$)

begin
  Set the starting level, $l \leftarrow 0$;
  $BestConcept \leftarrow \mathcal{T}_t$(root concept);
  repeat
    $s_{tmp} \leftarrow JaccardSim(\mathbf{c}_i, BestConcept)$;
    if ${\mathcal{T}_t}^{\langle l \rangle} \leq {\mathcal{T}_t}^{\langle max \rangle}$ **then**
      return *BestConcept;*
    Find all child concepts on this level, $\{\mathcal{B}\}_{j=1}^{h} \in {\mathcal{T}_t}^{\langle l \rangle}$;
    Flatten $\{\mathcal{B}_j\}_{j=1}^{h}$ and build corresponding feature vectors, $\{\mathbf{b}_j\}_{i=1}^{h}$;
    $s_{j^*} \leftarrow \text{argmax}_j JaccardSim(\mathbf{c}_i, \{\mathbf{b}_j\}_{j=1}^{h})$;
    if $s_{j^*} > s_{tmp}$ **then**
      $BestConcept \leftarrow \mathcal{B}_{j^*}$;
    Set $l \leftarrow l + 1$;
  until *BestConcept does not change*;
  return *BestConcept*;
end

is suitable for our task. Recently, Strehl and Ghosh [7] have proposed a version of the Jaccard coefficient called the *extended Jaccard similarity* that can work with continuous or discrete non-negative features. Let $\|\mathbf{x}_i\|$ be the L_2 norm of a given vector $\mathbf{x}_i$. The extended Jaccard similarity can be calculated as follows:

$$JaccardSim(\mathbf{x}_i, \mathbf{x}_j) = \frac{\mathbf{x}_i^T \mathbf{x}_j}{\|\mathbf{x}_i\|^2 + \|\mathbf{x}_j\|^2 - \mathbf{x}_i^T \mathbf{x}_j} \ . \tag{1}$$

We now describe an iterative algorithm for term-based ontology alignment. As mentioned earlier, we formulate that the ontology structure is in the form of the general tree. Our algorithm aligns the concepts on the source ontology $\mathcal{T}_s$ to

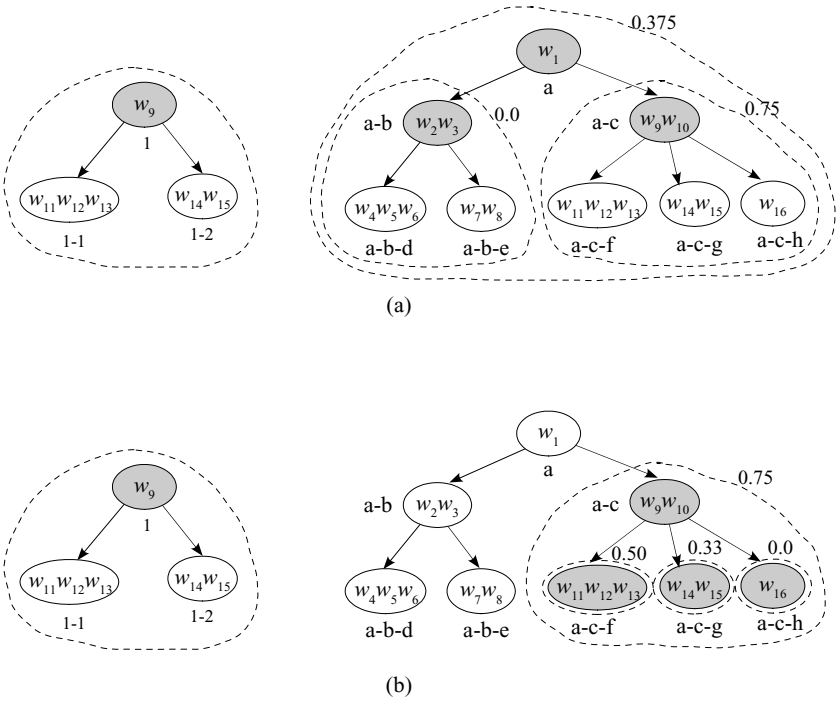

Fig. 1. An example of finding the most appropriate concept on $\mathcal{T}_t$ for the root concept $1 \in \mathcal{T}_s$

the concepts on the target ontology $\mathcal{T}_t$ by performing search and comparison in the top-down manner.

Given a concept $\mathcal{C}_i \in \mathcal{T}_s$, the algorithm attempts to find the most appropriate concept $\mathcal{B}^* \in \mathcal{T}_t$, which is located on an arbitrary level of the hierarchy. The algorithm starts by constructing the feature vectors for the current root concept on the level l and its child concepts on the level $l + 1$. It then calculates the similarity scores between a given source concept and candidate target concepts. If the similarity scores of the child concepts are not greater than the root concept, then the algorithm terminates. Otherwise, it selects a child concept having the maximum score to be the new root concept, and iterates the same searching procedure. Algorithms 1 and 2 outline our ontology alignment process.

Figure 1 shows a simple example that describes how the algorithm works. It begins with finding the most appropriate concept on $\mathcal{T}_t$ for the root concept $1 \in \mathcal{T}_s$. By flattening the hierarchy starting from given concepts ('1' on $\mathcal{T}_s$, and 'a', 'a-b', 'a-c' for $\mathcal{T}_t$), we can represent them with the feature vectors and measure their similarities. On the first iteration, the child concept 'a-c' obtains the maximum score, so it becomes the new root concept. Since the algorithm cannot find improvement on any child concepts in the second iteration, it stops

the loop and the target concept 'a-c' is aligned with the source concept '1'. The algorithm proceeds with the same steps by finding the most appropriate concepts on $\mathcal{T}_t$ for the concepts '1-1' and '1-2'. It finally obtains the resulting concepts 'a-c-f' and 'a-c-g', respectively.

4 Evaluation

4.1 Data Sets

In order to study the behavior of the proposed algorithm, two dictionaries are used in our experiments. The first one is the EDR Electronic Dictionary [6]. The second one is the electronic dictionary of Multilingual Machine Translation (MMT) project [8].

The EDR Electronic Dictionary consists of lexical knowledge of Japanese and English divided into several sub-dictionaries (e.g., the word dictionary, the bilingual dictionary, the concept dictionary, and the co-occurrence dictionary) and the EDR corpus. In the revised version (version 1.5), the Japanese word dictionary contains 250,000 words, while the English word dictionary contains 190,000 words. The concept dictionary holds information on the 400,000 concepts that are listed in the word dictionary. Each concept is marked with a unique hexadecimal number.

For the MMT dictionary, we use the Thai-English Bilingual Dictionary that contains around 60,000 lexical entries. The Thai-English Bilingual Dictionary also contains sematic information about the case relations and the word concepts. The word concepts are organized in a manner of semantic hierarchy. Each word concept is a group of lexical entries classified and ordered in a hierarchical level of meanings. The MMT semantic hierarchy is composed of 160 concepts.

In our experiments, we used a portion of the MMT semantic hierarchy and the EDR concept dictionary as the source and the target ontologies, respectively. We considered the 'animal' concept as the root concepts and extracted its related concepts. However, in the EDR concept dictionary, the relations among concepts are very complex and organized in the form of the semantic network. Thus, we pruned some links to transform the network to a tree structure. Starting from the 'animal' concept, there are more than 200 subconcepts (containing about 7,600 words) in the EDR concept dictionary, and 14 subconcepts (containing about 400 words) in the MMT semantic hierarchy. It is important to note that these two ontologies are considerably different in terms of the number of concepts and words.

4.2 Preliminary Results

Table 1 shows alignment results generated by our algorithm. Here we divide the mapping into two types: *exact* and *subset*. The exact mapping occurs when the MMT concept exactly matches the EDR concept. The subset mapping occurs when the definition of a given MMT concept does not appear in the EDR concept

Table 1. Results of aligned concepts between the MMT semantic hierarchy and the EDR concept dictionary

MMT concept	EDR concept	Mapping
vertebrate	vertebrate	exact
warm-blood	mammal	subset
mammal	mammal	exact
bird	bird	exact
cold-blood	reptile	subset
fish	fish	exact
amphibian	toad	subset
reptile	reptile	exact
snake	snake	exact
invertebrate	squid	subset
worm	leech	subset
insect	hornet	subset
shellfish	crab	subset
other sea creature	squid	subset

dictionary, so the algorithm tries to find the most suitable concept. Since the EDR concepts are more fine-grained than the MMT concepts, the definition of the resulting concept often is the subset of the source concept.

From 14 MMT concepts, 6 concepts are exactly matched with the EDR concepts, e.g. 'mammal', 'bird', and 'fish' concepts. The remaining 8 concepts are mapped to the closely related EDR concepts. For example, the 'warm-blood' concept in MMT is mapped to the 'mammal' concept in EDR. Although the 'warm-blood' concept does not occur in the EDR concept dictionary, some words in this concept appear to be a part of the 'mammal' concept in EDR. Moreover, a child concept of the 'warm-blood' concept is the 'mammal' concept. Thus, the algorithm decides to align the 'warm-blood' concept with the most similar EDR concept, which is the 'mammal' concept.

Figure 2 shows an example of aligned concepts found by our algorithm. The exact mapping can be found if two ontologies have the equivalent concepts and their elements overlap enough for resulting the maximum matching score. Also, the algorithm can yield the most appropriate concepts located on an *intermediate* level of the target ontology.

5 Conclusion and Future Work

This paper has described our first attempt to deal with the problem of automated ontology alignment. We present an efficient algorithm to align concepts between two ontologies based on the similarity of the shared terms. Our algorithm aligns the concepts between the source ontology and the target ontology by performing search and comparison in the top-down manner. Preliminarily experimental results show that the proposed algorithm can find reasonable concept mappings between two ontologies.

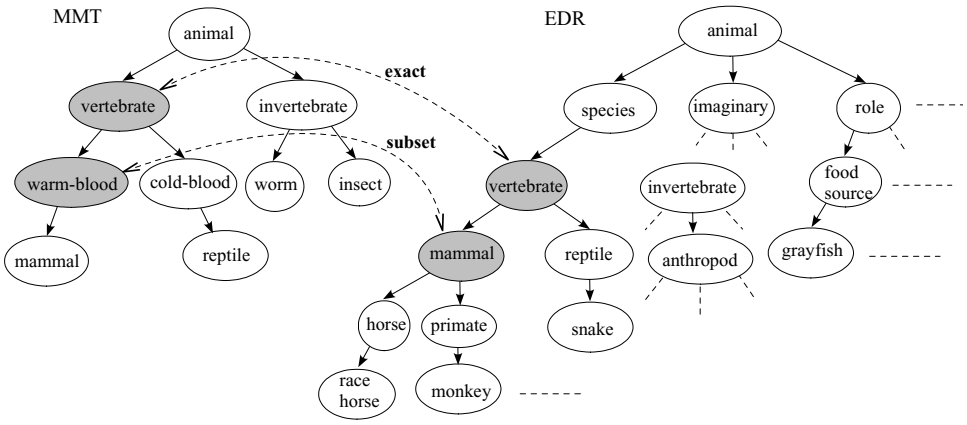

Fig. 2. An example of aligned concepts found by our algorithm

In future work, we plan to investigate our algorithm with larger data sets. Furthermore, we anticipate to apply a model selection technique such as Minimum Description Length (MDL) for generalizing the resulting concepts onto more coarse-grained concepts.

References

1. Ide, N. and Véronis, J.: Machine Readable Dictionaries: What have we learned, where do we go?. Proceedings of the International Workshop on the Future of Lexical Research, Beijing, China (1994) 137–146
2. Chen, B. and Fung, P.: Automatic Construction of an English-Chinese Bilingual FrameNet. Proceedings of Human Language Technology conference, Boston, MA (2004) 29–32
3. Ngai, G., Carpuat , M. and Fung, P.: Identifying Concepts Across Languages: A First Step towards a Corpus-based Approach to Automatic Ontology Alignment. Proceedings of the 19th International Conference on Computational Linguistics, Taipei, Taiwan (2002)
4. Khan, L. and Hovy, E.: Improving the Precision of Lexicon-to-Ontology Alignment Algorithms. Proceedings of AMTA/SIG-IL First Workshop on Interlinguas, San Diego, CA (1997)
5. Manning, C. D., and Schütze, H.: Foundations of statistical natural language processing. MIT Press. Cambridge, MA (1999)
6. Miyoshi, H., Sugiyama, K., Kobayashi, M. and Ogino, T.: An Overview of the EDR Electronic Dictionary and the Current Status of Its Utilization. Proceedings of the 16th International Conference on Computational Linguistics (1996) 1090–1093
7. Strehl, A., Ghosh, J., and Mooney, R. J.: Impact of similarity measures on web-page clustering. In Proceedings of AAAI Workshop on AI for Web Search (2000) 58-64
8. CICC: Thai Basic Dictionary. Center of the International Cooperation for Computerization, Technical Report 6-CICC-MT55 (1995)