Hypernym Discovery Based on Distributional Similarity and Hierarchical Structures

Ichiro Yamada[†], Kentaro Torisawa[†], Jun'ichi Kazama[†], Kow Kuroda[†], Masaki Murata[†], Stijn De Saeger[†], Francis Bond[†] and Asuka Sumida[‡]

[†]National Institute of Information and Communications Technology
3-5 Hikaridai, Keihannna Science City 619-0289, JAPAN
{iyamada,torisawa,kazama,kuroda,murata,stijn,bond}@nict.go.jp

[‡]Japan Advanced Institute of Science and Technology
1-1 Asahidai, Nomi-shi, Ishikawa-ken 923-1211, JAPAN
a-sumida@jaist.ac.jp

Abstract

This paper presents a new method of developing a large-scale hyponymy relation database by combining Wikipedia and other Web documents. We attach new words to the hyponymy database extracted from Wikipedia by using distributional similarity calculated from documents on the Web. For a given target word, our algorithm first finds *k* similar words from the Wikipedia database. Then, the hypernyms of these *k* similar words are assigned scores by considering the distributional similarities and hierarchical distances in the Wikipedia database. Finally, new hyponymy relations are output according to the scores. In this paper, we tested two distributional similarities. One is based on raw verb-noun dependencies (which we call "RVD"), and the other is based on a large-scale clustering of verb-noun dependencies (called "CVD"). Our method achieved an attachment accuracy of 91.0% for the top 10,000 relations, and an attachment accuracy of 74.5% for the top 100,000 relations when using CVD. This was a far better outcome compared to the other baseline approaches. Excluding the region that had very high scores, CVD was found to be more effective than RVD. We also confirmed that most relations extracted by our method cannot be extracted merely by applying the well-known lexico-syntactic patterns to Web documents.

1 Introduction

Large-scale taxonomies such as WordNet (Fellbaum 1998) play an important role in information extraction and question answering. However, extremely high costs are borne to manually enlarge and maintain such taxonomies. Thus, applications using these taxonomies tend to face the drawback of data sparseness. This paper presents a new method for discovering a large set of hyponymy relations. Here, a word[1] X is regarded as a hypernym of a word Y if Y is a kind of X or Y is an instance of X. We are able to generate large-scale hyponymy relations by attaching new words to the hyponymy database extracted from Wikipedia (referred to as "Wikipedia relation database") by using distributional similarity calculated from Web documents. Relations extracted from Wikipedia are relatively clean. On the other hand, reliable distributional similarity can be calculated using a large number of documents on the Web. In this paper, we combine the advantages of these two resources.

Using distributional similarity, our algorithm first computes *k* similar words for a target word. Then, each *k* similar word assigns a score to its ancestors in the hierarchical structures of the Wikipedia relation database. The hypernym that has the highest score for the target word is selected as the hypernym of the target word. Figure 1 is an overview of the proposed approach.

In the experiment, we extracted hypernyms for approximately 670,000 target words that are not included in the Wikipedia relation database but are found on the Web. We tested two distributional similarities: one based on raw verb-noun dependencies (RVD) and the other based on a large-scale clustering of verb-noun dependencies (CVD). The experimental results showed that the proposed methods were more effective than the other baseline approaches. In addition, we confirmed that most of the relations extracted by our method could not be extracted using the lexico-syntactic pattern-based method.

In the remainder of this paper, we first intro-

[1] In this paper, we use the term "word" for both "a single-word word" and "a multi-word word."

Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing, pages 929–937,
Singapore, 6-7 August 2009.

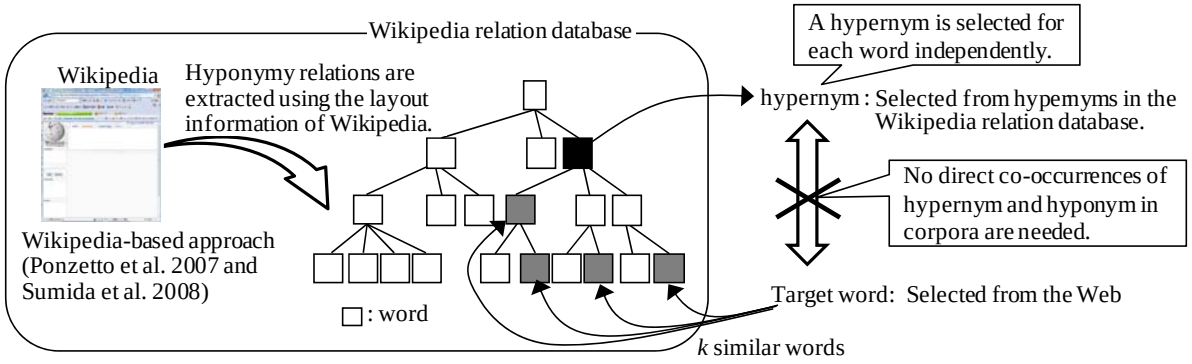


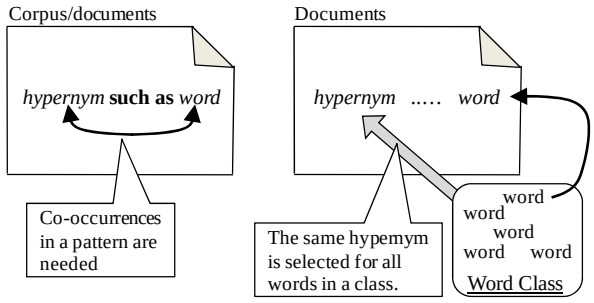
Figure 1: Overview of the proposed approach.

duce some related works in Section 2. Section 3 describes the Wikipedia relation database. Section 4 describes the distributional similarity calculated by the two methods. In Section 5, we describe a method to discover an appropriate hypernym for each target word. The experimental results are presented in Section 6 before concluding the paper in Section 7.

2 Related Works

Most previous researchers have relied on lexico-syntactic patterns for hyponymy acquisition. Lexico-syntactic patterns were first used by Hearst (1992). The patterns used by her included "NP_0 such as NP_1," in which NP_0 is a hypernym of NP_1. Using these patterns as seeds, Hearst discovered new patterns by which to semi-automatically extract hyponymy relations. Pantel et al. (2004a) proposed a method to automatically discover the patterns using a minimal edit distance. Ando et al. (2003) applied predefined lexico-syntactic patterns to Japanese news articles. Snow et al. (2005) generalized these lexico-syntactic pattern-based methods by using dependency path features for machine learning. Then, they extended the framework such that this method was capable of making use of heterogenous evidence (Snow et al. 2006). These pattern-based methods require the co-occurrences of a target word and the hypernym in a document. It should be noted that the requirement of such co-occurrences actually poses a problem when we extract a large set of hyponymy relations since they are not frequently observed (Shinzato et al. 2004, Pantel et al. 2004b).

Clustering-based methods have been proposed as another approach. Caraballo (1999), Pantel et al. (2004b), and Shinzato et al. (2004) proposed a method to find a common hypernym for word classes, which are automatically constructed using some measures of word similarities or hierarchical structures in HTML documents. Etzioni et al. (2005) used both a pattern-based approach and a clustering-based approach. The required amount of co-occurrences is significantly reduced due to class-based generalization processes. Note that these clustering-based methods obtain the same hypernym for all the words in a particular class. This causes a problem for selecting an appropriate hypernym for each word in the case when the granularity or the construction of the classes is incorrect. Figure 2 shows the drawbacks of the existing approaches.



Pattern-based method (Hearst 1992, Pantel et al. 2004a, Ando et al. 2003, Snow et al. 2005, Snow et al. 2006, and Etzioni et al. 2005)

Clustering-based method (Caraballo 1999, Pantel et al. 2004b, Shinzato et al. 2004, and Etzioni et al. 2005)

Figure 2: Drawbacks in existing approaches for hyponymy acquisition.

Ponzetto et al. (2007) and Sumida et al. (2008) proposed a method for acquiring hyponymy relations from Wikipedia. This Wikipedia-based approach can extract a large volume of hyponymy relations with high accuracy. However, it is also true that this approach does not account for many words that usually appear in Web documents; this could be because of the unbalanced topics in Wikipedia or merely because of the incomplete coverage of articles on Wikipedia. Our method can target words that frequently appear on the Web but are not included in the Wikipedia relation database, thus making the results of the Wikipedia-based approach richer and more balanced. Our approach uses distributional similari-

ty, which is computed based on the noun-verb dependency profiles on the Web. The use of distributional similarity resembles the clustering-based approach; however, our method can select a hypernym for each word independently, and it does not suffer from class granularity mismatch or the low quality of classes. In addition, our approach exploits the hierarchical structures of the Wikipedia hypernym relations.

3 Wikipedia Relation Database

Our Wikipedia relation database is based on the extraction method of Sumida et al. (2008). They proposed a method of automatically acquiring hyponymy relations by focusing on the hierarchical layout of articles on Wikipedia. By way of an example, Figure 3 shows part of the source code clipped from the article titled "Penguin." An article has hierarchical structures composed of titles, sections, itemizations, etc. The entire article is divided into sections titled "Anatomy," "Mating habits," "Systematics and evolution," "Penguins in popular culture," and so on. The section "Systematics and evolution" has a subsection "Systematics," which is further divided into "Aptenodytes," "Eudyptes," and so on. Some of these section-subsection relations can be regarded as valid hyponymy relations. In this article, relations such as the one between "Aptenodytes" and "Emperor Penguin" and that between "Book" and "Penguins of the World" are valid hyponymy relations.

First, Sumida et al. (2008) extracted hyponymy relation candidates from hierarchical structures on Wikipedia. Then, they selected proper hyponymy relations using a support vector machine classifier. They used several kinds of features for the hyponymy relation candidate, such as a POS tag for each word, the appearance of morphemes of each word, the distance between two words in the hierarchical structures of Wikipedia, and the last character of each word. As a result of their experiments, approximately 2.4 million hyponymy relations in Japanese were extracted, with a precision rate of 90.1%.

Compared to the traditional taxonomies, these extracted hyponymy relations have the following characteristics (Fellbaum 1998, Bond et al. 2008).

(a) The database includes a more extensive vocabulary.
(b) The database includes a large number of named entities.

Popular Japanese taxonomies *GoiTaikei* (Ikehara et al. 1997) and *Bunrui-Goi-Hyo* (1996)

```
'''Penguins''' are a group of
[[Aquatic animal|aquatic]],
[[flightless bird]]s.
== Anatomy ==
== Mating habits ==
==Systematics and evolution==
===Systematics===
* Aptenodytes
**[[Emperor Penguin]]
** [[King Penguin]]
* Eudyptes
== Penguins in popular culture ==
== Book ==
* Penguins
* Penguins of the World
== Notes ==
* Penguinone
* the [[Penguin missile]]
[[Category:Penguins]]
[[Category:Birds]]
```

Figure 3: A part of source code clipped from the article "Penguin" in Wikipedia.

contain approximately 300,000 words and 96,000 words, respectively. In contrast, the extracted hyponymy relations contain approximately 1.2 million hyponyms and are undoubtedly much larger than the existing taxonomies. Another difference is that since Wikipedia covers a large number of named entities, the extracted hyponymy relations also contain a large number of named entities.

Note that the extracted relations have a hierarchical structure because one hypernym of a certain word may also be the hyponym of another hypernym. However, we observed that the depth of the hierarchy, on an average, is extremely shallow. To make the hierarchy appropriate for our method, we extended these into a deeper hierarchical structure. The extracted relations include many compound nouns as hypernyms, and we decomposed a compound noun into a sequence of nouns using a morphological analyzer. Since Japanese is a head-final language, the suffix of a noun sequence becomes the hypernym of the original compound noun if the suffix forms another valid compound noun. We extracted suffixes of compound nouns and manually checked whether they were valid compound nouns; then, we constructed a hierarchy of compound nouns. The hierarchy can be extended such that it includes the hyponyms of the original hypernym and the resulting hierarchy constitutes a hierarchical taxonomy. We use this hierarchical taxonomy as a target for expansion.[2]

[2] Note that this modification was performed as part of another project of ours aimed at constructing a large-scale and clean hypernym knowledge base by human annotation. We do not think this cost is directly relevant to the method proposed here.

4 Distributional Similarity

The distributional hypothesis states that words that occur in similar contexts tend to be semantically similar (Harris 1985). In this section, we first introduce distributional similarity based on raw verb-noun dependencies (RVD). To avoid the sparseness problem of the co-occurrence of verb-noun dependencies, we also use distributional similarity based on a large-scale clustering of verb-noun dependencies (CVD).

In the experiment mentioned in the following section, we used the TSUBAKI corpus (Shinzato et al. 2008) to calculate distributional similarity. This corpus provides a collection of 100 million Japanese Web pages containing 6×10^9 sentences.

4.1 Distributional Similarity Based on RVD

When calculating the distributional similarity based on RVD, we use the triple <*v*, *rel*, *n*>, where *v* is a verb, *n* is a noun phrase, and *rel* stands for the relation between *v* and *n*. In Japanese, a relation *rel* is represented by postpositions attached to *n* and the phrase composed of *n* and *rel* modifies *v*. Each triple is divided into two parts. The first is <*v*, *rel*> and the second is *n*. Then, we consider the conditional probability of occurrence of the pair <*v*, *rel*>: *P*(<*v*, *rel*>|*n*). *P*(<*v*, *rel*>|*n*) can be regarded as the distribution of the grammatical contexts of the noun phrase *n*. The distributional similarity can be defined as the distance between these distributions. There are several kinds of functions for evaluating the distance between two distributions (Lee 1999). Our method uses the Jensen-Shannon divergence. The Jensen-Shannon divergence between two probability distributions, $P(\cdot \mid n_1)$ and $P(\cdot \mid n_2)$, can be calculated as follows:

$$\begin{aligned} & D_{JS}(P(\cdot \mid n_1) \,\|\, P(\cdot \mid n_2)) \\ & = \frac{1}{2}(D_{KL}(P(\cdot \mid n_1) \,\|\, \frac{P(\cdot \mid n_1) + P(\cdot \mid n_2)}{2}) \\ & + D_{KL}(P(\cdot \mid n_2) \,\|\, \frac{P(\cdot \mid n_1) + P(\cdot \mid n_2)}{2})), \end{aligned}$$

where D_{KL} indicates the Kullback-Leibler divergence and is defined as follows:

$$D_{KL}(P(\cdot \mid n_1) \,\|\, P(\cdot \mid n_2)) = \sum P(\cdot \mid n_1) \log \frac{P(\cdot \mid n_1)}{P(\cdot \mid n_2)}.$$

Finally, the distributional similarity between two words, n_1 and n_2, is defined as follows:

$$sim(n_1, n_2) = 1 - D_{JS}(P(\cdot \mid n_1) \,\|\, P(\cdot \mid n_2)).$$

This similarity assumes a value from 0 to 1. If two words are similar, the value will be close to 1; if two words have entirely different meanings, the value will be 0.

In the experiment, we used 1,000,000 noun phrases and 100,000 pairs of verbs and postpositions to calculate the probability *P*(<*v*, *rel*>|*n*) from the dependency relations extracted from the above-mentioned Web corpus (Shinzato et al. 2008). The probabilities are computed using the following equation by modifying for the frequency using the log function:

$$P(< v, rel > \mid n) = \frac{\log(f(< v, rel, n >)) + 1}{\sum_{< v, rel > \in D} \log(f(< v, rel, n >) + 1} \quad \text{if } f(< v, rel, n >) > 0,$$

where *f*(<*v*, *rel*, *n*>) is the frequency of a triple <*v*, *rel*, *n*> and *D* is the set defined as { <*v*, *rel* > | *f*(<*v*, *rel*, *n*>) > 0 }. In the case of *f*(<*v*, *rel*, *n*>) = 0, P(<*v*, *rel*>|*n*) is set to 0.

Instead of using the observed frequency directly as in the usual maximum likelihood estimation, we modified it as above. Although this might seems strange, this kind of modification is common in information retrieval as a term weighing method (Manning et al. 1999) and it is also applied in some studies to yield better word similarities (Terada et al. 2006, Kazama et al. 2009). We also adopted this idea in this study.

4.2 Distributional Similarity Based on CVD

Rooth et al. (1999) and Torisawa (2001) showed that EM-based clustering using verb-noun dependencies can produce semantically clean noun clusters. We exploit these EM-based clustering results as the smoothed contexts for noun *n*. In Torisawa's model (2001), the probability of occurrence of the triple <*v*, *rel*, *n*> is defined as follows:

$$\begin{aligned} & P(< v, rel, n >) \\ & \quad =_{def} \sum_{a \in A} P(< v, rel > \mid a) P(n \mid a) P(a), \end{aligned}$$

where *a* denotes a hidden class of <*v*,*rel*> and *n*. In this equation, the probabilities *P*(<*v*,*rel*>|*a*), *P*(*n*|*a*), and *P*(*a*) cannot be calculated directly because class *a* is not observed in a given corpus. The EM-based clustering method estimates these probabilities using a given corpus. In the E-step,

the probability $P(a|<v,rel>)$ is calculated. In the M-step, the probabilities $P(<v,rel>|a)$, $P(n|a)$, and $P(a)$ are updated to arrive at the maximum likelihood using the results of the E-step. From the results of estimation of this EM-based clustering method, we can obtain the probabilities $P(<v,rel>|a)$, $P(n|a)$, and $P(a)$ for each $<v, rel>$, n, and a. Then, $P(a|n)$ is calculated by the following equation:

$$P(a \mid n) = \frac{P(n \mid a)P(a)}{\sum_{a \in A} P(n \mid a)P(a)}.$$

$P(a|n)$ can be used to find the class of n. For example, the class that has the maximum $P(a|n)$ can be regarded as the class to which n belongs. Noun phrases that occur with similar pairs $<v,rel>$ tend to be classified in the same class.

Kazama et al. (2008) proposed the parallelization of this EM-based clustering with the aim of enabling large-scale clustering and using the resulting clusters in named entity recognition. Kazama et al. (2009) reported the calculation of distributional similarity using the clustering results. The distributional similarity was calculated by the Jensen-Shannon divergence, which was used in this paper. Similar to the case in Kazama et al., we performed word clustering using 1,000,000 noun phrases and 2,000 classes. Note that the frequencies of dependencies were modified with the log function, as in RVD, described in the previous section.

5 Discovering an Appropriate Hypernym for a Target word

In the Wikipedia relation database, there are about 95,000 hypernyms and about 1.2 million hyponyms. In both RVD and CVD, the words used were selected according to the number (the number of kinds, not the frequency) of $<v, rel>$s that n has dependencies in the data. As a result, 1 million words were selected. The number of common words that are also included in the Wikipedia relation database are as follows:

Hypernyms	28,015 (common hypernyms)
Hyponyms	175,022 (common hyponyms)

These common hypernyms become candidates for hypernyms for a target word. On the other hand, the common hyponyms are used as clues for identifying appropriate hypernyms.

In our task, the potential target words are about 810,000 in number and are not included in the Wikipedia relation database. These include some strange words or word phrases that are extracted due to the failure of morphological analysis. We exclude these words using simple rules. Consequently, the number of target words for our process is reduced to about 670,000.

In the following section, we outline the scoring method that uses k similar words to discover an appropriate hypernym for a target word. We also explain several baseline approaches that use distributional similarity.

5.1 Scoring with *k* similar Words

In this approach, we first calculate the similarities between the common hyponyms and a target word and select the k most similar common hyponyms. Here, we use a similarity threshold value S_{min} to avoid the effect of words having lower similarities. If the similarity is less than the threshold value, the word is excluded from the set of k similar words. Next, each k similar word votes a score to its ancestors in the hierarchical structures of the Wikipedia relation database. The score used to vote for a hypernym n_{hyper} is as follows:

$$score(n_{hyper}) = \sum_{n_{hypo} \in Desc(n_{hyper}) \cap ksimilar(n_{trg})} d^{r(n_{hyper}, n_{hypo})-1} \times sim(n_{trg}, n_{hypo}),$$

where n_{trg} is the target word, $Desc(n_{hyper})$ is the descendant of the hypernym n_{hyper}, $ksimilar(n_{trg})$ is the k similar word of n_{trg}, $d^{r(n_{hyper}, n_{hypo})-1}$ is a penalty that depends on the differences in the depth of hierarchy, d is a parameter for the penalty value and has a value between 0 and 1, and $r(n_{trg}, n_{hypo})$ is the difference in the depth of hierarchy between n_{trg} and n_{hypo}. $sim(n_{trg}, n_{hypo})$ is a distributional similarity between n_{trg} and n_{hypo}.

As a result of scoring, each hypernym has a score for the target word. The hypernym that has the highest score for the target word is selected as its hypernym. The hyponymy relations thus produced are ranked according to the scores.

Figure 4 shows an example of the scoring process. In this example, we use *CitroenAX* as the target word whose hypernym will be identified. First, the k similar words are extracted from the common hyponyms in the Wikipedia relation: *Opel Astra*, *TVR Tuscan*, *Mitsubishi Minica*, and *Renault Lutecia* are extracted. Next, each k similar word votes a score to its ancestors. The words *Opel Astra*, *TVR Tuscan*, and *Renault Lutecia* vote to their parent *car* and the word *Mitsubishi*

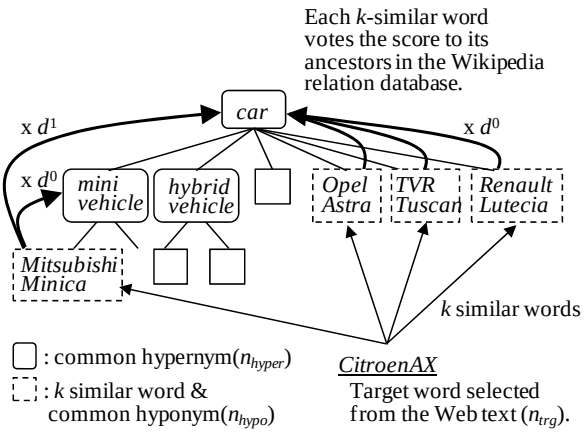


Figure 4: Overview of the scoring process.

Minica votes to its parent *mini-vehicle* and its grandparent *car* with a small penalty. Finally, the hypernym *car*, which has the highest score, is selected as the hypernym of the target word *CitroenAX*.

5.2 Baseline Approaches

Using distributional similarity, we can also develop the following baseline approaches to discover hyponymy relations.

Selecting the hypernym of the most similar hyponym (baseline approach 1)

We use the heuristics that similar words tend to have the same hypernym. In this approach, we first calculate the similarities between the common hyponyms and the target word. The common hyponym most similar to the target word is extracted. Then, the parent of the extracted common hyponym is regarded as the hypernym of the target word. This approach outputs several hypernyms when the most similar hyponym has several hypernyms. This approach can be considered to be the same as the scoring method using k similar words when $k = 1$. We use the distributional similarity between the target word and the most similar hyponym in the Wikipedia relation database as the score for the appropriateness of the resulting hyponymy.

Selecting the most similar hypernym (baseline approach 2)

The distributional similarity between the common hypernym and the target word is calculated. Then, the hypernym that has the highest distributional similarity is regarded as the hypernym of the target word. The similarity is used as the score of the appropriateness of the produced hyponymy.

Scoring based on the average similarity of the hypernym's children (baseline approach 3)

This approach uses the probabilistic distributions of the hypernym's children. We define the probability $P_{child}(\cdot \mid n_{hyper})$ characterized by the children of the hypernym n_{hyper}, as follows:

$$P_{child}(\cdot \mid n_{hyper}) = \frac{\sum_{n_{hypo} \in Ch(n_{hyper})} P(\cdot \mid n_{hypo}) P(n_{hypo})}{\sum_{n_{hypo} \in Ch(n_{hyper})} P(n_{hypo})},$$

where $Ch(n_{hyper})$ is a set of all children of n_{hyper}. Then, distributional similarities between a common hypernym n_{hyper} and the target word n_{hypo} are calculated. The hypernym that has the highest distributional similarity is selected as the hypernym of the word. This distributional similarity is used as the score of the appropriateness of the produced hyponymy.

If a hypernym has only a few children, the reliability of the probabilistic distribution of hypernym defined here will be low because the Wikipedia relation database includes some incorrect relations. For this reason, we use the hypernym only if the number of children it has is more than a threshold value.

6 Experiments

We evaluated our proposed methods by using it in experiments to discover hypernyms from the Wikipedia relation database for the target words extracted from about 670,000 noun phrases.

6.1 Parameter Estimation by Preliminary Experiments

In the proposed methods, there are several parameters. We performed parameter optimization by randomly selecting 694 words as development data in our preliminary experiments. The hypernyms of these words were determined manually. We adjusted the parameters so that each method achieved the best performance for this development data.

The parameters in the scoring method with k similar words were adjusted as follows[3]:

(RVD)

Number of similar words:	$k = 100$.
Similarity threshold:	$S_{min} = 0.05$.
Penalty value for ancestors:	$d = 0.6$.

[3] We tested the parameter values k = {100, 200, 300, 400, 500, 600, 700, 800, 900, 1000}, S_{min}={0, 0.05, 0.1, 0.15, 0.2, 0.25, 0.3, 0.35, 0.4} and d={0.5, 0.55, 0.6, 0.65, 0.7, 0.75, 0.8, 0.85, 0.9, 0.95, 1.0}.

Table 1: Precision of each approach based on the score ranking. CVD represents the method that uses the distributional similarity based on large-scale of clustering of verb-noun dependencies. RVD represents the one based on raw verb-noun dependencies.

	k-similar words (CVD)	*k*-similar words (RVD)	*k*-similar words (CVD, $d = 0$)	Baseline approach 1 (CVD)	Baseline approach 2 (CVD)	Baseline approach 3 (CVD)
1,000	0.940	**1.000**	0.850	0.730	0.290	0.630
10,000	**0.910**	0.875	0.875	0.555	0.300	0.445
100,000	**0.745**	0.710	0.730	0.500	0.280	0.435
670,000	**0.520**	0.500	0.470	0.345	0.115	0.170

(CVD)

Number of similar words: $k = 200$.
Similarity threshold: $S_{min} = 0.3$.
Penalty value for ancestors: $d = 0.6$.

The parameter in baseline approach 3 was adjusted as follows:

Threshold for the number of children: 20.

6.2 Evaluation of the Experimental Results on the Basis of Score Ranking

Using the adjusted parameters, we conducted experiments to extract the hypernym of each target word with the help of the scoring method based on *k* similar words. In these experiments, two kinds of distributional similarity mentioned in Section 4 were exploited individually. The words that were used in the development data were excluded.

We also conducted a comparative experiment in which the parameter value for the penalty of the hierarchal difference, *d*, was set to 0 to clarify the ability of using hierarchal structures in the *k* similar words method. This means each *k* similar word votes only to their parent.

We then judged the quality of each acquired hypernym. The evaluation data sets were sampled from the top 1,000, 10,000, 100,000, and 670,000 results that were ranked according to the score of each method. Then, against 200 samples that were randomly sampled from each set, one of the authors judged whether the hypernym extracted by each method for the target word was correct or not. In this evaluation, if the sentence "*The target word* is a kind of *the hypernym*" or "*The target word* is an instance of *the hypernym*" was consistent, the extracted hyponymy was judged as correct. It should be noted that the outputs of the compared methods are combined and shuffled to enable fair comparison. In addition, baseline approach 1 extracted several hypernyms for the target word. In this case, we judged the hypernym as correct when the case where one of the hypernyms was correct.

The precision of each result is shown in Table 1. The results of the *k* similar words method are far better than those of the other baseline methods. In particular, the *k* similar words method with CVD outperformed the methods of the *k* similar words where the parameter value *d* was set to 0 and the method using RVD except for the top 1,000 results. This means that the use of hierarchal structures and the clustering process for calculating distributional similarity are effective for this task. We confirmed the significant differences of the proposed method (CVD) as compared with all the baseline approaches at the 1% significant level by the Fisher's exact test (Hays 1988).

The precision of baseline approach 2 that selected the most similar hypernym was the worst among all the methods. There were words that were similar to the target word among the hypernyms extracted incorrectly. For example, the word *semento-kojo* (cement factory) was extracted for the hypernym of the word *kuriningu-kojo* (dry cleaning plant). It is difficult to judge whether the word is a hypernym or just a similar word by using only the similarity measure.

As for the results of baseline approach 1 using the most similar hyponym and baseline approach 3 using the similarity of the set of hypernym's children, the noise on the Wikipedia relation database decreased the precision. Moreover, over-specified hypernyms were extracted incorrectly by these methods. In contrast, the method of scoring based on the use of *k* similar words was robust against noise because it uses the voting approach for the similarities. Further, this method can extract hypernyms that are not over-specific because it uses all descendants for scoring.

Table 2 shows some examples of relations extracted by the *k* similar words method using CVD.

Table2: Hypernym discovery results by the *k*-similar words based approach (CVD). The underline indicates the hypernyms which are extracted incorrectly.

Score	Target word	Extracted hypernym
58.6	INDIVI	*burando* (fashion label)
54.3	*kureome* (Cleome)	*hana* (flower)
34.4	UOKR	*gemu* (game)
21.7	*Okido* (Okido)	*machi* (town)
20.5	*Sumatofotsu* (Smart fortwo)	*kuruma* (car)
15.6	*Fukagawameshi* (Fukagawa rice)	*ryori* (dish)
8.9	John Barry	*sakkyokuka* (composer)
8.5	JVM	*sofuto-wea* (software)
6.6	*metangasu* (methane gas)	*genso* (chemical element)
5.4	*me-ru semina* (mail seminar)	*Hon* (book)
3.9	*gurometto* (grommet)	*shohin* (merchandise)
3.1	*supuringubakku* (spring back)	*gensho* (phenomenon)

6.3 Investigation of the Extracted Relation Overlap with a Conventional Method

We randomly sampled 300 hyponymy relations that were extracted correctly using the *k* similar words method exploiting CVD and investigated whether or not these relations can be extracted by the conventional method based on the lexico-syntactic pattern. The possible hyponymy relations were extracted using the pattern-based method (Ando et al. 2003) from the TSUBAKI corpus (Shinzato et al. 2008). From a comparison of these relations, we found only 57 common hyponymy relations. That is, the remaining 243 hyponymy relations were not included in the possible hyponymy relations. This result indicates that our method can acquire the hyponymy relations that cannot be extracted by the conventional pattern-based method.

6.4 Discussions

We investigated the reason for the errors generated by the method of scoring using *k* similar words exploiting CVD. We conducted experiments on hypernym extraction targeting 694 words in the development data mentioned in Section 6.1. Among these, 286 relations were extracted incorrectly. In these relations, there were some frequent hypernyms. For example, the word *sakuhin* (work) appeared 28 times and *hon* (book) appeared 20 times. As shown in Table 2, *hon* (book) was also extracted for the target word *meru-seminah* (mail seminar). It is really difficult even for a human to identify whether the title is that of the book or the event. If we can identify these difficult hypernyms in advance, we can improve precision by excluding them from the target hypernyms. This will be one of the topics for future study.

7 Conclusion

In this paper, we proposed a method for discovering hyponymy relations between nouns by fusing the Wikipedia relation database and words from the Web. We demonstrated that the method using *k* similar words has high accuracy. The experimental results showed the effectiveness of using hierarchal structures and the clustering process for calculating distributional similarity for this task. The experimental results showed that our method could achieve 91.0% attachment accuracy for the top 10,000 hyponymy relations and 74.5% attachment accuracy for the top 100,000 relations when using the clustering-based similarity. We confirmed that most relations extracted by the proposed method could not be handled by the lexico-syntactic pattern-based method. Future work will be to filter out difficult hypernyms for hyponymy extraction process to achieve higher precision.

References

M. Ando, S. Sekine and S. Ishizaki. 2003. Automatic Extraction of Hyponyms from Newspaper Using Lexicosyntactic Patterns. *IPSJ SIG Notes*, 2003-NL-157, pp. 77–82 (in Japanese).

F. Bond, H. Isahara, K. Kanzaki and K. Uchimoto. 2008. Boot-strapping a WordNet Using Multiple Existing WordNets. In *the 6th International Conference on Language Resources and Evaluation* (LREC), Marrakech.

Bunruigoihyo. 1996. The National Language Research Institute (in Japanese).

S. A. Caraballo. 1999. Automatic Construction of a Hypernym-labeled Noun Hierarchy from Text. In *Proceedings of the Conference of the Association for Computational Linguistics* (ACL).

O. Etzioni, M. Cafarella, D. Downey, A. Popescu, T. Shaked, S. Soderland, D. Weld and A. Yates. 2005. Unsupervised Named-Entity Extraction from the Web: An Experimental Study. *Artificial Intelligence*, 165(1):91–134.

C. Fellbaum. 1998. WordNet: An Electronic Lexical

Database. Cambridge, MA: MIT Press.

Z. Harris. 1985. Distributional Structure. In Katz, J. J. (ed.) The Philosophy of Linguistics, *Oxford University Press*, pp. 26–47.

W. L. Hays. 1988. Statistics: Analyzing Qualitative Data, Rinehart and Winston, Inc., Ch. 18, pp. 769–783.

M. Hearst. 1992. Automatic Acquisition of Hyponyms from Large Text Corpora. In *Proceedings of the 14th Conference on Computational Linguistics* (COLING), pp. 539–545.

S. Ikehara, M. Miyazaki, S. Shirai, A. Yokoo, H. Nakaiwa, K. Ogura, Y. Ooyama and Y. Hayashi. 1997. Goi-Taikei A Japanese Lexicon, *Iwanami Shoten.*

J. Kazama and K. Torisawa. 2008. Inducing Gazetteers for Named Entity Recognition by Large-scale Clustering of Dependency Relations. In *Proceedings of ACL-08: HLT*, pp. 407–415.

J. Kazama, Stijn De Saeger, K. Torisawa and M. Murata. 2009. Generating a Large-scale Analogy List Using a Probabilistic Clustering Based on Noun-Verb Dependency Profiles. In *15th Annual Meeting of the Association for Natural Language Processing*, C1–3 (in Japanese).

L. Lee. 1999. Measures of Distributional Similarity. In *Proceedings of the 37th Annual Meeting of the Association for Computational Linguistics*, pp. 25–32.

C. D. Manning and H. Schutze. 1999. Foundations of Statistical Natural Language Processing. The MIT Press.

P. Pantel, D. Ravichandran and E. Hovy. 2004a. Towards Terascale Knowledge Acquisition. In *Proceedings of the 20th International Conference on Computational Linguistics*.

P. Pantel and D. Ravichandran. 2004b. Automatically Labeling Semantic Classes. In *Proceedings of the Human Language Technology and North American Chapter of the Association for Computational Linguistics Conference*.

S. P. Ponzetto, and M. Strube. 2007. Deriving a Large Scale Taxonomy from Wikipedia. In *Proceedings of the 22nd National Conference on Artificial Intelligence*, pp. 1440–1445.

M. Rooth, S. Riezler, D. Presher, G. Carroll and F. Beil. 1999. Inducing a Semantically Annotated Lexicon via EM-based Clustering. In *Proceedings of the 37th annual meeting of the Association for Computational Linguistics*, pp. 104–111.

K. Shinzato and K. Torisawa. 2004. Acquiring Hyponymy Relations from Web Documents. In *Proceedings of HLT-NAACL*, pp. 73–80.

K. Shinzato, D. Kawahara, C. Hashimoto and S. Kurohashi. 2008. A Large-Scale Web Data Collection as A Natural Language Processing Infrastructure. In *the 6th International Conference on Language Resources and Evaluation* (LREC).

R. Snow, D. Jurafsky and A. Y. Ng. 2005. Learning Syntactic Patterns for Automatic Hypernym Discovery. *NIPS 2005*.

R. Snow, D. Jurafsky, A. Y. Ng. 2006. Semantic Taxonomy Induction from Heterogenous Evidence. In *Proceedings of the 21st International Conference on Computational Linguistics and the 44th annual meeting of the Association for Computational Linguistics*, pp. 801–808.

A. Sumida, N. Yoshinaga and K. Torisawa. 2008. Boosting Precision and Recall of Hyponymy Relation Acquisition from Hierarchical Layouts in Wikipedia. In *the 6th International Conference on Language Resources and Evaluation* (LREC).

A. Terada, M. Yoshida, H. Nakagawa. 2006. A Tool for Constructing a Synonym Dictionary using context Information. In *proceedings of IPSJ SIG Technical Reports*, vol.2006 No.124, pp. 87-94. (In Japanese).

K. Torisawa. 2001. An Unsupervised Method for Canonicalization of Japanese Postpositions. In *Proceedings of the 6th Natural Language Processing Pacific Rim Symposium* (NLPRS), pp. 211–218.

K. Torisawa, Stijn De Saeger, Y. Kakizawa, J. Kazama, M. Murata, D. Noguchi and A. Sumida. 2008. TORISHIKI-KAI, An Autogenerated Web Search Directory. In *Proceedings of the second international symposium on universal communication*, pp. 179–186, 2008.