Extending WordNet with Hypernyms and Siblings Acquired from Wikipedia

Ichiro Yamada[†‡] **Jong-Hoon Oh**[†] **Chikara Hashimoto**[†] **Kentaro Torisawa**[†]
Jun'ichi Kazama[†] **Stijn De Saeger**[†] **Takuya Kawada**[†]
[†]Information Analysis Laboratory, National Institute of
Information and Communications Technology, 619-0289 Kyoto, Japan
{rovellia,ch,torisawa,kazama,stijn,tkawada}@nict.go.jp
[‡]NHK Science & Technology Research Laboratories, 157-8510 Tokyo, Japan
yamada.i-hy@nhk.or.jp

Abstract

This paper proposes a method for extending WordNet with terms in Wikipedia. Our method identifies a WordNet synset by integrating evidence derived from the structure of an article in Wikipedia and distributional similarity of terms. Unlike previous methods, utilizing the hypernym and siblings of the target term acquired from Wikipedia, the proposed method can deal with terms other than Wikipedia article titles and can work well even when reliable distributional similarity of a target term is unavailable. Experiments show that the proposed method can identify synsets for 2,039,417 inputs at precision rate of 84%. Furthermore, it is estimated from the experimental results that there should be 328,572 terms among all the inputs whose synset our method can correctly identify, while previous methods relying only on distributional similarity and lexico-syntactic patterns cannot.

1 Introduction

As a comprehensive repository of word senses, WordNet (Fellbaum, 1998) has played an important role in many natural-language-processing (NLP) tasks. Hand-crafted semantic knowledge, however, has low coverage of named entities and domain-specific knowledge. To address this issue, many researchers have proposed methods for extending WordNet by mapping new terms to a WordNet "synset."[1]

In this paper, we propose a novel method that extends WordNet by integrating the *Wikipedia article structure* and *distributional similarity*. With this method, an appropriate synset for a term in Wikipedia is identified by using the distributional similarity of the term and that of the term's hypernym and siblings, which are automatically acquired from Wikipedia. (Hereafter, **trg** is used for a target term for which we identify the appropriate synset, **hyper** is used for the hypernym of **trg** and **sib** is used for the sibling of **trg**.)

The reason for using **hyper** and **sib** can be explained as follows. In general, when an unknown term is encountered, its context helps in interpreting the term. Especially, if the unknown term's hypernym and/or its semantically similar terms (its **sib**s) were somehow learned as context, it would often be possible to successfully guess its meaning. In WordNet expansion, **trg** may correspond to terms for which reliable distributional similarity is unavailable. In such cases, the distributional similarity of **hyper** and **sib** can help. Even when the distributional similarity of **trg** is available, that of **hyper** and **sib** can boost the performance of synset identification by providing additional sources of information.

In this study, **trg**, **hyper** and **sib** are derived from hyponymy relation instances acquired from Wikipedia. Acquisition of hyponymy relations from Wikipedia is based on the internal structure of Wikipedia articles. For example, the Wikipedia article "Jack Black" is composed of the article title "Jack Black", section titles "Career" and "Films", and an itemized list under the "films" section as shown in Fig. 1. Hyponymy relation instances like (films, *Kung Fu Panda*), (films, *Airborne*) and (films, *Johnny Skidmarks*) can be acquired from this structure (Sumida et al., 2008). Most hyponyms in these hyponymy relation instances are not in WordNet and thus should be added to WordNet.

Trg, **hyper** and **sib** obtained from Wikipedia are the inputs ($\mathcal{I}$s. See Section 3) to candidate generation of appropriate synset for **Trg**, whose outputs, in turn, become the inputs to candidate selection (Figure 2). The candidate generation relies on multiple *synset identification modules*.

[1] A synset is a set of cognitive synonyms, each expressing a distinct concept.

Proceedings of the 5th International Joint Conference on Natural Language Processing, pages 874–882,
Chiang Mai, Thailand, November 8 – 13, 2011.

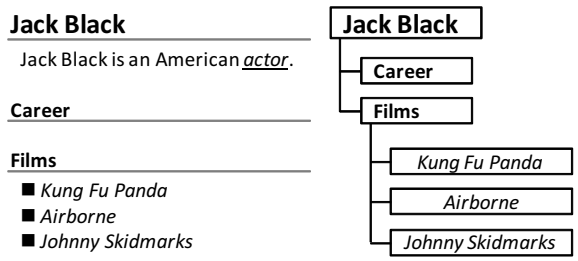

Figure 1: Internal structure of Wikipedia article "Jack Black".

Each module uses a different combination of information sources for generating appropriate synset candidates. This difference between the modules makes it possible to generate diverse synset candidates from the different viewpoint of each module. The most appropriate synset is selected by the candidate selection using a classifier that can discriminate better synset candidates from worse ones among outputs of the multiple synset identification modules. The candidate selection is expected to improve the precision of synset identification.

Experiments show that the proposed method can identify synsets for 2,039,417 $\mathcal{I}$s with 84% precision rate. In contrast, our implementation of Yamada et al. (2009), which uses the distributional similarity of **trg** only, has a precision rate of only 50.3% and covers less than half of the whole input from Wikipedia. Furthermore, it is estimated from the experimental results that there should be 328,572 terms among all the 1,231,172 unique **trg**s in the 2,039,417 $\mathcal{I}$s whose synset can be correctly identified by our method, but not by previous methods relying only on distributional similarity and lexico-syntactic patterns (Snow et al., 2006; Yamada et al., 2009).

2 Related Work

As a resource for extending WordNet, Wikipedia has recently received growing interest (Ruiz-Casado et al., 2005; Suchanek et al., 2007; Toral et al., 2008; Wu and Weld, 2008; Toral et al., 2009; Ponzetto and Navigli, 2009). These studies link a Wikipedia article title to a WordNet synset by using the Wikipedia category system, Wikipedia infoboxes, or similarity between Wikipedia article contents as evidence. However, these methods cannot handle terms that are not Wikipedia-article titles, and thus their coverage is limited.

On the other hand, distributional similarity between terms has been used in extending an existing taxonomy like WordNet (Snow et al., 2006; Yamada et al., 2009). Snow et al. (2006) identified a hypernym for a target term by using lexico-syntactic patterns and distributionally similar terms of the target term. Then the target term is linked to a WordNet synset by using its hypernym and distributionally similar terms as evidence. Yamada et al. (2009) linked a target term to its hypernym in a given taxonomy by using distributional similarity between the target term and terms in the taxonomy.

However, it is often the case that we cannot obtain reliable distributional similarity of a term and we cannot acquire hypernyms of a term co-occurring with lexico-syntactic patterns, especially when the term is infrequent in a corpus. As a result, we can hardly expect that the previous methods (Snow et al., 2006; Yamada et al., 2009) work well for this infrequent term. Nonetheless, we believe that it is important to deal with such infrequent terms, since they constitute the long-tail of the Web. Our method exploits not only the distributional similarity of a target term but also that of hypernym and siblings of the target term. Accordingly, as indicated by the experimental results in Section 4, our method achieves a higher precision and a broader coverage.

Many researchers have proposed methods for hyponymy relation acquisition from texts (Hearst, 1992; Shinzato and Torisawa, 2004; Sumida and Torisawa, 2008; Sumida et al., 2008; Oh et al., 2009; Oh et al., 2010). Recently, Wikipedia has gained attention as a source for hyponymy relations (Sumida and Torisawa, 2008; Sumida et al., 2008; Oh et al., 2009; Oh et al., 2010). Hyponymy relation instances acquired from Wikipedia are relevant to hypernyms and siblings of a term in our proposed method and the method of Sumida and Torisawa (2008) is used for preprocessing in the proposed method.

3 Proposed Method

The proposed method is overviewed in Figure 2. In the preprocessing stage (Section 3.1), hyponymy relation instances are acquired by using a method of acquiring hyponymy relations (proposed by Sumida et al. (2008)) from Wikipedia articles. From these relations, **trg**, **hyper** and **sib**, which are denoted as $\mathcal{I}$ $=<$ target term, hypernym, siblings $>$ or $\mathcal{I}$ $=<$ $trg, hyp, N_{sib}(trg)$ $>$ in short, are obtained. In

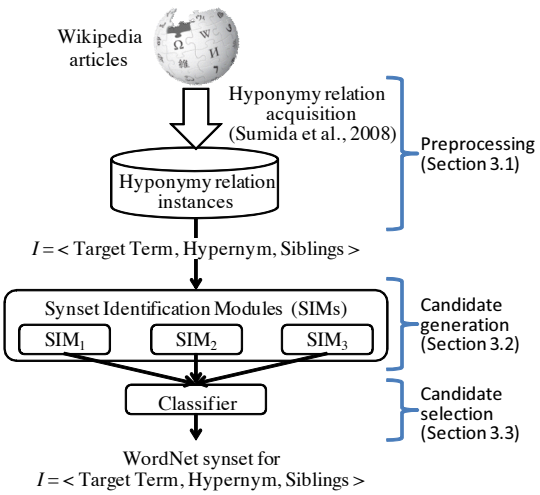

Figure 2: Overview of proposed method.

the candidate-generation stage (Section 3.2), three synset identification modules (SIMs) generate candidates of appropriate synsets for a given $\mathcal{I}$. Finally, in the candidate-selection stage (Section 3.3), a classifier produces the final output by selecting the best one among the output of the three SIMs. In our task, "one sense per one hyponymy relation instance" is assumed. For example, *Airborne* in Fig. 1 represents the meaning of "film", while the term itself has several other meanings.

3.1 Acquisition of Hypernyms (hyper) and Siblings (sibs)

A set of hyponymy relation instances are acquired by using "A tool for hyponymy relation acquisition from Wikipedia"[2] (Sumida et al., 2008). This tool extracts hyponymy-relation candidates from a Wikipedia article structure and then applies SVM to select the correct hyponymy relation instances from the candidates. Among the resulting hyponymy relation instances, reliable ones are selected by using a threshold value for a SVMs score with 90% precision [3]. Furthermore, hyponymy relation instances are restricted to ones whose hyponym comes from leaf nodes in the hierarchical layout of Wikipedia articles (i.e., *Kung Fu Panda* in Fig. 1). By means of this restriction, terms that are not named entities are filtered out. The method used for acquisition of hyponymy relations is explained in detail in Sumida et al. (2008).

[2]Available at `http://alaginrc.nict.go.jp/hyponymy/index.html`

[3]To ensure the results had 90% precision, an SVM score (distance from hyperplane) of 0.49 was used as the threshold value.

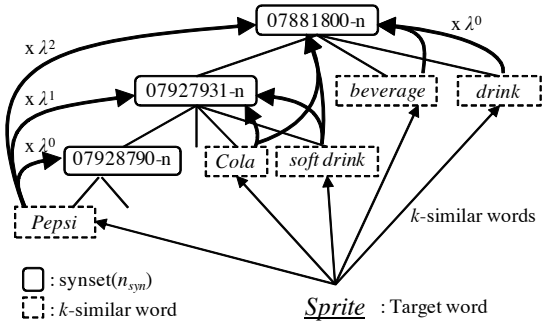

Figure 3: Example of score propagation by **trg** *Sprite* and its **sib** *Coke*.

Some **hyper**s in hyponymy relations extracted from Wikipedia are often very long noun sequences like *wild south-China tiger*. Consequently, their reliable distributional similarity is unavailable owing to their low-frequency or their absence in corpora. This problem is addressed by applying a longest-suffix match against frequently-appearing terms in a corpus for which reliable distributional similarity is measured (Section 3.2.2) under the assumption that the longest suffix can be regarded as the superordinate concept of **hyper**. For example, *south-China tiger* is the longest-suffix match result for *wild south China tiger*. If the longest suffix for **hyper** cannot be found, the hyponymy relation instances containing **hyper** are ignored.

Sibs are extracted from a Wikipedia article structure under the condition that their **hyper**s are the same.

3.2 Synset Identification Modules

Each synset identification module (SIM) generates synset candidates for a given $\mathcal{I} =< trg, hyp, N_{sib}(trg) >$. The candidates are determined by a *score propagation method*. A WordNet synset gets a higher score if it is considered to be the appropriate synset for trg.

3.2.1 Scoring by Propagation

The score for input $\mathcal{I} =< trg, hyp, N_{sib}(trg) >$ and WordNet synset syn is defined as $PS(\mathcal{I}, syn)$ in Eq. (1), which represents the weighted sum of the sub-scores for target term trg, its hypernym hyp, and a set of siblings, $N_{sib}(trg)$. Each sub-score, $S(n, syn)$ (Eq. (2)), is computed by score propagation through the hierarchical structure of WordNet synsets, where n is either **trg**, **hyper** or **sib**. Figure 3 illustrates the score propagation. It is assumed that "*Sprite*" is a **trg** and "07881800-

$$PS(\mathcal{I}, syn) = \alpha \times S(trg, syn) + \beta \times S(hyp, syn) + \gamma \times \sum_{sib \in N_{sib}(trg)} \frac{S(sib, syn)}{|N_{sib}(trg)|} \quad (1)$$

$$S(n, syn) = \sum_{n_k \in TopK(n)} \sum_{syn_k \in SYN(n_k)} \lambda^{d(syn, syn_k)} \times sim(n, n_k) \quad (2)$$

n" is a synset for which the sub-score $S(n, syn)$ is computed. First, a set of k terms in the Japanese WordNet that is the most similar to **trg** is obtained. This set is extracted by using the distributional similarity measure mentioned in Section 3.2.2 and is denoted by $TopK(n)$ in Eq. (2). Second, the synset *receives* the penalized distributional similarity from the k terms ($n_k \in TopK(n)$ in Eq. (2)). The score propagation is controlled by $\lambda^{d(syn, syn_k)}$, where $0 < \lambda < 1$, $d(syn, syn_k)$ is the distance between two synsets syn and syn_k in the WordNet hierarchy, and syn_k is a synset to which $n_k \in TopK(n)$ belongs. More precisely, $d(syn, syn_k)$ is the minimum length of any ancestral path between syn and syn_k, and $d(syn, syn_k) = 0$ if $syn = syn_k$.

Note that as distance $d(syn, syn_k)$ increases, $\lambda^{d(syn, syn_k)}$ becomes smaller and $sim(n, n_k)$ therefore makes less contribution to $S(n, syn)$. On the other hand, $S(n, syn)$ tends to *recieve* a penalized similarity from more synsets distant from syn than those close to syn. The score therefore has the largest value when these two tendencies are balanced.

The score propagation for **trg** and **sib**, in Figure 3, is done only in the direction to ancestors in the Wordnet hierarchy. On the contrary, the score propagation for **hyper** is done in a slightly different way. That is, the penalized distributional similarity is propagated to not only the ancestors but also the descendants.

The final score value, $PS(\mathcal{I}, syn)$, is the sum of sub-score values S for **trg**, **hyper**, and **sib** weighted by constants α, β, and γ, where $\alpha + \beta + \gamma = 1$. These constants are optimized in the experiment, which is described in Section 4.1. This score-propagation scheme is an extension of Yamada et al. (2009).

3.2.2 Measuring Distributional Similarity

The distributional similarity between two terms (n_1 and n_2) is defined as

$$sim(n_1, n_2) = 1 - D_{JS}(P(a|n_1) \| P(a|n_2)) \quad (3)$$

where a denotes a class to which the term belongs, and $D_{JS}(P(a|n_1) \| P(a|n_2))$ is the Jensen-Shannon divergence between two probability distributions, $P(a|n_1)$ and $P(a|n_2)$.

To calculate probability distribution $P(a|n)$, Torisawa (2001) conducted noun clustering using the triple $< v, p, n >$ obtained from a parsed corpus, where v, n, and p represent a verb, a noun, and a postposition that attaches to the noun. The noun and the postposition constitute a phrase that depends on the verb. The probability of occurrence of the triple $< v, p, n >$ is defined as

$$P(< v, p, n >) \quad (4)$$
$$=_{def} \sum_{a \in A} P(< v, p > |a) P(n|a) P(a)$$

where a denotes a class of $< v, p >$ and n. $P(< v, p > |a)$, $P(n|a)$, and $P(a)$ are estimated by the EM-based clustering method, which estimates these probabilities by using a given corpus. In the E-step, probability $P(a| < v, p >)$ is calculated. In the M-step, probabilities $P(< v, p > |a)$, $P(n|a)$, and $P(a)$ are updated to arrive at the maximum likelihood using the results of the E-step. From the results of estimation by this EM-based clustering method, probabilities $P(< v, p > |a)$, $P(n|a)$, and $P(a)$ for $< v, p >$, n, and a are obtained. $P(a|n)$ is then calculated by the following equation:

$$P(a|n) = \frac{P(n|a)P(a)}{\sum_{a \in A} P(n|a)P(a)} \quad (5)$$

With the aim of enabling large-scale clustering and using the resulting clusters in named entity recognition, Kazama and Torisawa (2008) proposed parallelization of this EM-based clustering method. Kazama et al. (2009) then reported the calculation of distributional similarity by using the clustering results. We applied their method to the TSUBAKI corpus (Shinzato et al., 2008), a collection of 100-million Japanese Web pages containing 6×10^9 sentences. We prepared

about 1,000,000 terms for calculating the distributional similarity. These one million terms consist of the following three sets of terms: (1) sets of hyponymy relation instances extracted from Wikipedia, (2) sets in WordNet, and (3) sets from the TSUBAKI corpus that are neither (1) nor (2). Terms in sets (1) or (2) were required to syntactically depend on 10 different $< v, p >$ in the TSUBAKI corpus for reliably calculating the distributional similarities. Terms in set (3) have been chosen from those that have the largest number of dependency relations in the corpus, so the total number of terms is one million.

3.2.3 Definition of SIMs

Three synset identification modules (SIMs), namely, SIM_1, SIM_2, and SIM_3, were developed. These three modules make it possible to generate diverse candidates of an appropriate synset by using different evidence derived from **trg**, **hyper**, and **sib**. Table 1 summarizes the information that each SIM uses as a trigger for score propagation. SIM_1 relies on $PS(\mathcal{I}, syn)$, whose β and γ are zero, while SIM_2 is defined by $PS(\mathcal{I}, syn)$, whose γ is zero.

Information sources	SIM_1	SIM_2	SIM_3
Target term (**trg**)	✓	✓	✓
Hypernym (**hyper**)		✓	✓
Sibling (**sib**)			✓

Table 1: Information sources used in each module.

Basically, each SIM generates top-n synsets that maximize $PS(\mathcal{I}, syn)$ over all WordNet synsets. Here, one heuristics is used for SIM_2 and SIM_3 when **hyper** is available. When **hyper** belongs to WordNet synsets, one of the synsets is usually the appropriate synset of the **trg**. According to this observation, the top-n synsets among the synsets that contain **hyper** are generated. However, if the hyponymy relation is wrong (like *musician* as a hypernym of *acoustic guitars*), this heuristics will have a negative effect on the performance of synset identification.

To avoid this effect, the following additional condition is set: At least one of the synsets to which **hyper** belongs has score $PS(\mathcal{I}, syn) > 0$. Under this condition, the synset which contains the **hyp** but is not supported by the **trg** and its **sib**s is not preferred in generating candidates of an appropriate synset.

3.3 Selecting Appropriate Synset among Outputs of Multiple SIMs

Once SIMs generate WordNet synset candidates for a **trg**, to select the most appropriate synset, a classifier is applied to these candidates. As the classifier, SVMs trained with a polynomial kernel of degree 2 are used[4]. Moreover, the following features are used for training SVMs, which are selected by the ablation test reported in Section 4.4.

f1: **hyper**

f2: Name of SIM used for generating appropriate synset candidates

f3: Value of $PS(\mathcal{I}, syn)$ given by each SIM

f4: Synset ID of WordNet synset candidate

f5: Suffix of **trg**

f6: Suffix of **hyper**

Regarding f5 and f6, the suffixes are obtained in the same way as the procedure for longest-suffix matching described in Section 3.1. Finally, the synset which has the largest SVM score is selected as appropriate synset for **trg**.

4 Experiments

4.1 Experimental Set up

For our experiments, 4,057,879 hyponymy relation instances were acquired from the 2009-09-27 version of the Japanese Wikipedia dump (containing about 0.9 million articles). Hyponymy relation instances whose hyponyms are not found in the Japanese WordNet and are from a leaf node in a layout structure of a Wikipedia article (which usually corresponds to an itemized list like the movie names in Fig. 1) were then selected. After these processes, 2,039,417 hyponymy relation instances containing 1,231,172 unique hyponyms (**trg**), of which about 80% are not Wikipedia article titles, were acquired. **Sib**s for each hyponym (**trg**) were then acquired from the hyponymy relation instances. Finally, 2,039,417 $\mathcal{I}$s (note that $\mathcal{I} =< trg, hyp, N_{sib}(trg) >$ composed of **trg**, **hyper**, and **sib**s) were used for the experiments.

800 $\mathcal{I}$s were randomly selected for development data, and 1,800 $\mathcal{I}$s were selected for test data from the 2,039,417 $\mathcal{I}$s, where a **trg** in the selected $\mathcal{I}$s is unique over both development and test data. Candidate generation was applied to these development and test data, and the appropriate synsets

[4]TinySVM, available at `http://chasen.org/taku/software/TinySVM`, `wasused`

among candidates for each $\mathcal{I}$ were then manually labeled. In the labeling, three judges were asked to mark synset candidates as the correct synset for $\mathcal{I}$ if one of terms in the synset candidate is an appropriate hypernym[5]. Finally, the correct synset for $\mathcal{I}$ was determined by the judges' majority vote. The interrater agreement between the three judges (Siegel's Kappa) was 0.785, indicating substantial agreement.

We performed parameter optimization by using the development data. The parameters used in our method showing the best performance for development data were used in our experiments, namely, the number of similar terms $k = 60$, the parameter for score propagation $\lambda = 0.6$, and weights for $S(n, syn)$ in $PS(\mathcal{I}, syn)$ ($\alpha = 0.6$ and $\beta = 0.4$ for SIM_2 and $\alpha = 0.5$, $\beta = 0.4$, and $\gamma = 0.1$ for SIM_3).

4.2 Results

In the experiment, the eleven systems listed as follows (ten baseline systems and our proposed system) were evaluated.

- $B1$–$B3$: Bi represents a system that outputs the best candidate of SIM_i ($1 \leq i \leq 3$).
- $SB1$–$SB3$: SBi represents a system based on a classifier that selects the best synset among the top-5 candidates of SIM_i ($1 \leq i \leq 3$)
- $CB1$: randomly selects one of the outputs of B1–B3 as the appropriate synset.
- $CB2$: selects the most frequently observed synset in the training data among the outputs of B1–B3.
- $EB1$: randomly selects one of synsets, which contains a **hyper** in $\mathcal{I}$.
- $EB2$: selects the most frequently observed synset in the training data among synsets, which contains a **hyper** in $\mathcal{I}$.
- Proposed method: The proposed method using B1–B3

$B1$ is our implementation of the method proposed by Yamada et al. (2009). Evaluation of B1–B3 shows the performance of candidate generation by SIM_1, SIM_2, and SIM_3. $EB1$ and $EB2$ can be considered simple extensions of an existing research of Sumida et al. (2008) for estimating Wordnet synsets.

Table 2 shows the precision rate of each system. We could not evaluate all 1,800 samples for $B1$, $SB1$, $EB1$ and $EB2$. $B1$ and $SB1$ were able to generate outputs for 614 $\mathcal{I}$s, where trg in $\mathcal{I}$s was included in the target terms for calculating the distributional similarity. $EB1$ and $EB2$ can select the synset when **hyper** or the suffix of **hyper** is registered in WordNet. For this reason, it was not possible to select a synset for 174 **trg**s out of the 1,800 samples in $EB1$ and $EB2$. As a result, we used 1,636 samples in evaluating $EB1$ and $EB2$. $SB1$–$SB3$, $CB2$, $EB2$, and the proposed method were evaluated by five-fold cross validation with test data because these systems need training data for learning their classifier or finding the most frequently observed synset.

The precision rate of the proposed method is the highest among those of all the systems in Table 2. Comparison of the proposed method and one of $SB1$–$SB3$ shows the effectiveness of integration of different information generated by multiple SIMs. In the results for $B1$–$B3$ and $SB1$–$SB3$, $SB1$ and $SB2$ (which use a classifier), respectively, attain higher precision than systems $B1$ and $B2$ (which do not use a classifier). The precision of $SB3$ is, however, lower than that of $B3$, indicating that using a classifier is not always effective for synset identification.

System	Precision
$B1$	50.3 (309/ 614)
$B2$	70.9 (1,276/1,800)
$B3$	78.2 (1,408/1,800)
$SB1$	59.3 (364/ 614)
$SB2$	72.4 (1,303/1,800)
$SB3$	76.3 (1,374/1,800)
$CB1$	71.9 (1,294/1,800)
$CB2$	80.2 (1,444/1,800)
$EB1$	56.5 (924/1,636)
$EB2$	79.2 (1,296/1,636)
Proposed method	84.2 (1,515/1,800)

Table 2: Experimental results of each system.

4.3 Evaluation by Ranking

Figure 4 shows precision rates by their ranking for the system outputs of $B1$–$B3$, $SB1$–$SB3$, and the proposed method. The vertical axis indicates precision rate; the horizontal axis indicates the rank

[5]Judges were required to label the following ten synsets (00001740, 00001930, 00002137, 00002684, 00003553, 00004258, 00004475, 00023100, 00007347, and 00021939) as wrong one. These synsets were selected in descending order of the number of their lower nodes in the WordNet hierarchy.

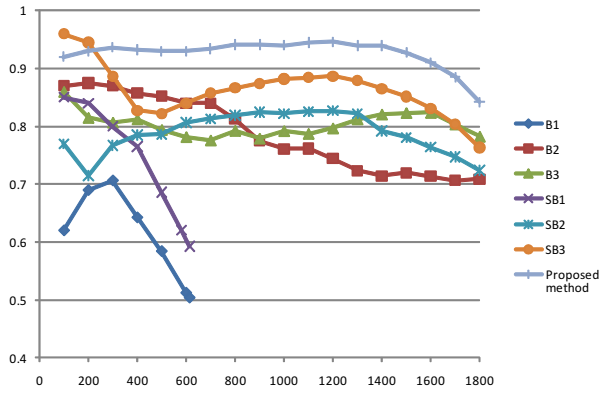

Figure 4: Precision rate by ranking.

of **trg** in scores, i.e., all the outputs are sorted in the descending order of scores. For $B1$, $B2$, and $B3$, $PS(\mathcal{I}, syn)$ is used as a score, while for $SB1$, $SB2$, $SB3$, and the proposed method, the distance from the hyperplane of an SVM is used as a score. For $SB1$, $SB2$, $SB3$, and the proposed method, averaged scores over the five folds were used.

It should be noted that the proposed method outperforms the other methods for almost all the ranks and keeps a precision rate of about 84%. This result implies that the method can identify synsets with a precision rate of about 84% for all 2,039,417 $\mathcal{I}$s.

Note also that the proposed method keeps a precision rate of about 91% until the top 88.9% (1600/1800). From this result, it is estimated that the method can identify synsets for 1,813,042 $\mathcal{I}$s (88.9% of all 2,039,417 $\mathcal{I}$s) with a precision rate of about 91%.

4.4 Contribution of Each Feature

To examine the effectiveness of each of the six features used for the classifier, ablation tests using the development data (which examined the change in the performance of the classifier when one of the features was ignored) were conducted. Table 3 lists the results of these tests. This table shows that

Feature set	Precision
All	82.6 (661/800)
w/o **Hyper** (f1)	82.1 (657/800)
w/o Name of SIM (f2)	82.5 (660/800)
w/o Value of $PS(\mathcal{I}, syn)$ (f3)	82.0 (656/800)
w/o Synset ID (f4)	81.1 (649/800)
w/o Suffix of the **trg** (f5)	82.1 (657/800)
w/o Suffix of the **hyper** (f6)	82.4 (659/800)

Table 3: Results of ablation test.

f4 (the synset ID of a WordNet synset candidate) is the most effective for the classifier.

4.5 Distribution of Output Synset IDs

freq.	Synset ids	Example terms in WordNet
336	04599396	work, piece of work
336	00007846	someone, person
132	06613686	moving picture,movie
68	06619428	broadcast, programme
44	08237863	cast, cast of characters
44	08008335	organisation, organization
28	06376154	drama
25	08276720	school
25	03315023	installation, facility
23	06616806	docudrama, documentary
22	07020895	music
22	04341686	construction, structure
19	00455599	game
17	06362953	writing, piece of writing
17	03129123	creation

Table 4: Distribution of output synset IDs.

Table 4 lists the distribution of output synset IDs determined by our method. About two-thirds of the results were assigned a very specific synset. The remaining terms were all assigned to the two most-frequent synsets: 04599396 or 00007846. In the case when a term was assigned to these most-frequent synsets, a more specific synset, like 06613686 (moving picture, movie) or 09765278 (actor), should have been chosen. However, our current scoring process does not take the synsets' granularity into account, so it sometimes favors the more general synsets, like 06613686 or 00007846. Fine-tuning our algorithm to compensate for this tendency will be an important future work.

4.6 Analysis

4.6.1 Advantage of Proposed Method

The advantage of the proposed method compared to previous methods that use nothing but either distributional similarity of a **trg** or co-occurrence with their hypernyms via lexico-syntactic patterns (Snow et al., 2006; Yamada et al., 2009) (or both) was demonstrated as follows. Specifically, it was shown that many terms do not have reliable distributional similarity (owing to their infrequency in a corpus) and do not co-occur with their hypernyms via any lexico-syntactic pattern in a sentence. Even so, our method can correctly identify the synset of such terms thanks to their **hy-**

pers and **sib**s acquired from the internal structure of Wikipedia articles.

First, 1,515 terms were extracted from all 1,800 $\mathcal{I}$s whose synset our method could correctly identify. The occurrence of each term in a corpus that consists of 600 million Japanese Web pages (a super set of the one-million-page TSUBAKI corpus) was then counted. According to the result of this counting, 430 terms occur less than 10 times. We believe that it is not possible to obtain reliable distributional similarity for these terms owing to their infrequency. Note that the distributional similarity of the suffix of **hyper** was used instead of that of the **hyper** itself. (Section 3.1). If the suffix of a term is used, it might be possible to obtain reliable distributional similarity even for the 430 terms. Accordingly, it was determined whether the suffix of a term can help the synset identification for each of the 430 terms by checking whether the suffix of each of the 430 terms is actually its subordinate concept. According to the results of this checking, 342 terms out of the 430 do not have a suffix that is their correct subordinate concept. Reliable distributional similarity is thus not available for them even when the suffix technique is used.

Next, the co-occurrence of each of the 342 terms and its **hyper** via some lexico-syntactic pattern within a sentence was checked by using the 600-million-page Japanese Web corpus. According to the results of this check, 290 terms out of 342 do not co-occur with their **hyper** within a sentence; thus, the co-occurrence with their **hyper** cannot be used to identify their synset.

In conclusion, it is difficult for the previous methods to correctly identify the synset of the 290 terms that do not co-occur with their **hyper** within a sentence, while our method can. From this result, it is estimated that the number of such terms in all the 2,039,417 $\mathcal{I}$s is 328,572.

4.6.2 Error Analysis

From the 285 incorrect synsets output by the proposed method, 124 were selected from B3, 110 were selected from B2, and 51 were selected from B1. For 232 of these 285 errors, all outputs of B1, B2, and B3 were judged incorrect. Because the proposed method's classifier chooses the final result from the outputs of B1, B2, and B3, it cannot help selecting the wrong candidate in these cases. 100 erroneous synsets were randomly selected from our results, and the following three types of error were found.

Missing terms for some senses in the Japanese WordNet (20/100): For instance, the Japanese term *anime* is defined in synset *06616464-n* as *animation originating in Japan*, but no term in the Japanese WordNet is linked to this synset (Bond et al., 2009). The Japanese WordNet does contain other meanings for the term *anime*, such as *a hard copal derived from an African tree* (synset *14896018-n*) and *any of various resins or oleoresins* (synset *14766265-n*). As a result, Japanese animation films with the hypernym *anime* are linked to either *14896018-n* or *14766265-n*. This type of error should be avoidable by adding such missing terms to the Japanese WordNet.

Terms incorrectly identified as persons (16/100): Many named entities such as companies or movie titles are often mistaken for references to people. For example, a movie titled "BROTHER" is distributionally similar to other movies as well as family terms like "sister" and "mother". Moreover, WordNet does not contain many movie titles, so the "family term" sense is selected as the dominant sense, and "BROTHER" was given the synset of *person*. We expect such problems can be alleviated by adding more named entities to WordNet.

Hyponymy relation acquisition error (10/100): The precision of hyponymy-relation acquisition was 90%, which accounted for the remaining 10% of the errors. For example, the term *acoustic guitar* was given the wrong hypernyms, namely, *musician*, which results in misclassification.

5 Conclusion

This paper proposed a method for extending WordNet with terms in Wikipedia, by exploiting hypernyms (**hyper**s) and siblings (**sib**s) of target terms (**trg**s) acquired from Wikipedia as additional sources of information. Experimental results showed that the proposed method could identify synsets for 2,039,417 inputs at precision rate of 84%. Furthermore, it was estimated that there were 328,572 terms among all the inputs whose synsets the proposed method could correctly identify. In contrast, previous methods relying on distributional similarity and lexico-syntactic patterns only could not identify these synsets.

References

Francis Bond, Hitoshi Isahara, Sanae Fujita, Kiyotaka Uchimoto, Takayuki Kuribayashi, and Kyoko Kanzaki. 2009. Enhancing the Japanese WordNet. In *Proceedings of the 7th Workshop on Asian Language Resources*, pages 1–8.

Christiane Fellbaum. 1998. *WordNet: An Electronical Lexical Database*. The MIT Press.

Marti A. Hearst. 1992. Automatic acquisition of hyponyms from large text corpora. In *Proceedings of the 14th International Conference on Computational linguistics (COLING-92)*, pages 539–545.

Jun'ichi Kazama and Kentaro Torisawa. 2008. Inducing gazetteers for named entity recognition by large-scale clustering of dependency relations. In *Proceedings of the 46th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies (ACL-08:HLT)*, pages 407–415.

Jun'ichi Kazama, Stijn De Saeger, Kentaro Torisawa, and Masaki 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)*, pages 84–87.

Jong-Hoon Oh, Kiyotaka Uchimoto, and Kentaro Torisawa. 2009. Bilingual co-training for monolingual hyponymy-relation acquisition. In *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP (ACL-IJCNLP 2009)*, pages 432–440.

Jong-Hoon Oh, Ichiro Yamada, Kentaro Torisawa, and Stijn De Saeger. 2010. Co-star: A co-training style algorithm for hyponymy relation acquisition from structured and unstructured text. In *Proceedings of the 23rd International Conference on Computational Linguistics (COLING 2010)*, pages 842–850.

Simone Paolo Ponzetto and Roberto Navigli. 2009. Large-scale taxonomy mapping for restructuring and integrating wikipedia. In *Proceedings of the 21th International Joint Conference on Artificial Intelligence (IJCAI 2009)*, pages 2083–2088.

Maria Ruiz-Casado, Enrique Alfonseca, and Pablo Castells. 2005. Automatic assignment of wikipedia encyclopedic entries to wordnet synsets. In *Advances in Web Intelligence, volume 3528 of Lecture Notes in Computer Science*, pages 380–386.

Keiji Shinzato and Kentaro Torisawa. 2004. Acquiring hyponymy relations from web documents. In *Proceedings of the Human Language Technology Conference and North American chapter of the Association for Computational Linguistics annual meeting (HLT-NAACL04)*, pages 73–80.

Keiji Shinzato, Tomohide Shibata, Daisuke Kawahara, Chikara Hashimoto, and Sadao Kurohashi. 2008. Tsubaki: An open search engine infrastructure for developing new information access. In *Proceedings the Third International Joint Conference on Natural Language Processing (IJCNLP 2008)*, pages 189–196.

Rion Snow, Daniel Jurafsky, and Andrew Y. Ng. 2006. Semantic taxonomy induction from heterogenous evidence. In *Proceedings of the 43nd Annual Meeting of the Association for Computational Linguistics (ACL-06)*, pages 801–808.

Fabian M. Suchanek, Gjergji Kasneci, and Gerhard Weikum. 2007. Yago: A Core of Semantic Knowledge. In *Proceedings of the 17th international conference on World Wide Web (WWW '07)*, pages 697–706.

Asuka Sumida and Kentaro Torisawa. 2008. Hacking Wikipedia for hyponymy relation acquisition. In *Proceedings of the Third International Joint Conference on Natural Language Processing (IJCNLP 2008)*, pages 883–888.

Asuka Sumida, Naoki Yoshinaga, and Kentaro Torisawa. 2008. Boosting precision and recall of hyponymy relation acquisition from hierarchical layouts in Wikipedia. In *Proceedings of the 6th International Conference on Language Resources and Evaluation (LREC 2008)*.

Antonio Toral, Rafael Munoz, and Monica Monachini. 2008. Named entity wordnet. In *Proceedings of the Sixth International Conference on Language Resources and Evaluation (LREC 2008)*, pages 741–747.

Antonio Toral, Oscar Ferrandez, Eneko Agirre, and Rafael Munoz. 2009. A study on linking Wikipedia categories to WordNet using text similarity. In *Proceedings of Recent Advances in Natural Language Processing (RANLP 2009)*, pages 449–454.

Kentaro Torisawa. 2001. An unsuperveised method for canonicalization of Japanese postpositions. In *Proceedings of the 6th Natural Language Processing Pacific Rim Symposium (NLPRS 2001)*, pages 211–218.

Fei Wu and Daniel S. Weld. 2008. Automatically refining the wikipedia infobox ontology. In *Proceeding of the 17th international conference on World Wide Web (WWW '08)*, pages 635–644.

Ichiro Yamada, Kentaro Torisawa, Jun'ichi Kazama, Kow Kuroda, Masaki Murata, Stijn De Saeger, Francis Bond, and Asuka Sumida. 2009. Hypernym discovery based on distributional similarity and hierarchical structures. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing (EMNLP 2009)*, pages 929–937.