Vol.2011-IFAT-104 No.3
2011/11/22

Japanese Hyponymy Extraction based on a Term Similarity Graph

Takuya Akiba[†1] and Tetsuya Sakai[†2]

We present a new method for automatic extraction of hyponymy relations between Japanese words from large-scale web corpora. Our method utilizes a term similarity graph, as well as information from Wikipedia. Our experimental results based on tens of millions of web pages show that our method can extract Japanese hyponymy relations with 82% precision.

1. Introduction

Semantic relations between words, such as hyponymy, synonymy and meronymy, have various information access applications (e.g. Web search) and the automatic extraction of such relations from corpora is an important research problem in natural language processing.

For the Japanese language, there exist several linguistic resources that contain these relations, such as the Japanese Wordnet, *Nihongo Goitaikei* and EDR electric dictionary. However, the cost of maintaining such knowledge and of adapting to linguistic phenomena that keep evolving is very high. Therefore, many studies have been conducted for automatic extraction of these semantic relationships.

In this study, we focus on automatic extraction of hyponymy relations from large corpora. Here, a word A is a hypernym of a word B (or the word B is a hyponym of the word A) if B is a kind of A or B is an instance of A. The relation is also called as *is-a* relation. Although there are already many existing studies on automatic hyponymy extraction, there still is a lot of room for improvement in terms of precision, recall or the trade-off between the two. Hyponymy relations are useful, for example, for an information access system that supports generalization and specialization of the user's information needs.

We focus on the problem of automatic hyponymy extraction from Japanese corpora, and propose a method that utilizes a term similarity graph, as well as information from Wikipedia. Our experimental results based on tens of millions of web pages show that our method can extract Japanese hyponymy relations with 82% precision.

2. Related Work

Research on automatic hyponymy extraction started with pattern-based approaches. Hearst[2] created lexico syntactic patterns in English like "A such as B" and applied these patterns to texts of an encyclopedia. Similar work in Japanese was conducted by Ando et al.[1]. They created Japanese patterns and applied them on news articles.

Though these pattern-matching approaches are important baselines or building blocks, their precision and recall performances are quite limited, mainly because of *sparsity*. That is, hyponymy relations that do not occur in the corpora in the context of the specific patterns can never be obtained.

Besides the aforementioned plain-text corpora, structured texts may also be utilized for hyponymy extraction. Shinzato et al.[6] proposed a method which utilizes HTML structures. Ponzetto et al.[4] used categories of Wikipedia, and Sumida et al.[7] used tables of contents of Wikipedia.

One way to improve the effectiveness of the above approaches is to use term similarity graphs, which are weighted graphs whose vertices represent words and edges represent similarities between words. Yamada et al.[8] combined a term similarity graph with the aforementioned Wikipedia-based method by Sumida et al.[7]. Zhang et al.[9] combined a term similarity graph with the pattern-based methods, and this is the work which we build on.

3. Proposed Methods

3.1 Pattern-based Hyponymy Extraction

As a starting point, we use a pattern-based method, using the patterns by Ando et al.[1] specified in Table 1.

Our method first locates a pattern in the corpus, and then check the words on

[†1] The University of Tokyo
[†2] Microsoft Research Asia

Table 1 Patterns which we use for hyponymy extraction.

Japanese Patterns	English Translation
A *nado* B	B such as A
A *nado-no* B	
A *ni-nita* B	B which is imilar to A
A *no-youna* B	
A *to-iu* B	B which is called A
A *to-yoba-reru* B	
A *igai-no* B	B other than A

either side of that pattern. If they are both nouns, we extract the word pair as a candidate.

3.2 Weight Models

By the above pattern-based method, we may obtain multiple hypernym candidates for a hyponym candidate. We therefore rank the candidate hypernyms by computing a weight for each of them. The most basic formula, adopted from 2), is the following:

$$w(T \to L) = c(T \to L) \times IDF(L) \tag{1}$$

$$= c(T \to L) \times \frac{1+N}{1+DF(L)} \tag{2}$$

where $w(T \to L)$ is the weight of word L as a hypernym of word T, $c(T \to L)$ is the number of occurrences of patterns that suppport the word L as a hypernym of word T, N is the number of words, and $DF(L)$ is the number of different hyponyms that have word L as a hypernym.

More sophisticated formulas were proposed after that. The one used in 9) is the following:

$$w(T \to L) = \left(\sum_i \sqrt{c(T \to L, P_i)} \right) \times IDF(L) \tag{3}$$

where $c(T \to L, P_i)$ is the number of occurrences of pattern P_i that suppports the word L as a hypernym of word T. The idea behind this formula is that, if a particular pattern has occurred once, then there is a high probability that the same pattern will occur again. Hence we discount the later occurrences.

Through a preliminary experiment, we noticed that the effect of the IDF factor in Eq. (3) is too strong. Therefore, we modified the formula as follows:

Table 2 Examples of the entries in the term similarity graph. The numbers written in parentheses describe the similarity.

Original Word	*gyorai* (torpedo)	*tonyu* (soymilk)
Rank 1	*jirai* (mine, 0.36)	*tohu* (bean curd, 0.42)
Rank 2	*kirai* (submarine mine, 0.36)	*okara* (okara, 0.41)
Rank 3	*bakurai* (depth charge, 0.30)	*daietto* (diet, 0.40)

$$w(T \to L) = \left(\sum_i \sqrt{c(T \to L, P_i)} \right)^2 \times IDF(L). \tag{4}$$

Note that this formula is more consistent with the original formula (Eq. (1)) than Eq. (3) in terms of the balance between the number of occurrences and the IDF factor.

3.3 Evidence Propagation

One of the problems of the pattern-matching approaches is *sparsity*. For rare terms, it is hard to have a number of supporting patterns even if we use tens of millions of web pages. Moreover, for common terms, still it is desirable to increase the amount of evidence to improve the precision. To cope with that, we use a term-similarity graph for evidence propagation.

A term similarity graph is a weighted graph whose vertices represent words and edges represent similarities between words. There are several studies on generating term similarity graphs from large web corpora: we used a term similarity graph generated by the method described in 5). Table 2 shows some examples of the entries in this graph.

We use term-similarity graphs to "borrow" the supporting patterns from other hyponyms. In 9), the following formula is proposed:

$$w'(T \to L) = w(T \to L) + \sum_{T_1 \neq T} \mu \times Sim(T, T_1) \times w(T_1 \to L) \tag{5}$$

where $w'(T \to L)$ is the new weight of word L as a hypernym of word T, $w(T \to L)$ is the previous weight without evidence propagation, μ is a constant called the propagation factor and $Sim(T, T_1)$ is the similarity between word T and word T_1.

Because the number of occurrences of words varies drastically between different words by orders of magnitude, we found that the effect of frequent words tend to

Table 3 Examples of first sentences of Wikipedia entries.

Sentence	Hyponym	Hypernym
toukeigaku toha, toukei ni kansuru kenkyuu wo okonau gakumon de aru	*toukeigaku* (statistics)	*gakumon* (study)
UNIX ha, konpyu-ta you no operethingu sisutemu no isshu de aru.	UNIX	*operethingu sisutemu* (opeprating system)

dominate in Eq. (5). Therefore, we propose to *scale* the previous weight before evidence propagation using the following formula:

$$\tilde{w}(T \to L) = \frac{\log\left(\sum_{L'} w(T \to L')\right)}{\sum_{L'} w(T \to L')} \times w(T \to L) \qquad (6)$$

and compute the final weight as the following formula

$$w'(T \to L) = \tilde{w}(T \to L) + \sum_{T_1 \neq T} \mu \times Sim(T, T_1) \times \tilde{w}(T_1 \to L). \qquad (7)$$

The idea behind Eq. (6) is the following. Because we do not want frequent terms to dominate the weight, we divide the score by the total score the term has as a hyponym. On the other hand, because a higher total score means that the hypernyms of the term are correct with a higher probability, we want frequent terms to have a greater impact than rare terms. Therefore, we multiply the score by the logarithm of the total score.

3.4 Exploiting Wikipedia

Because Wikipedia has many entries, actually large amount of important nouns are contained as an entry in Wikipedia. Therefore, we devised techniques to exploit Wikipedia entries to improve the accuracy of hypernym extraction for words included in Wikipedia.

3.4.1 First Sentences in Entries

A Japanese Wikipedia description page of a term often starts with a sentence that states that the term is a kind/instance of X, where X is a hypernym of the term. The hypernym is often located near the end of that sentence. Table 3 shows two examples: the first example shows that statistics is a kind of study; the second shows that UNIX is a kind of operating system.

We enumerated typical suffixes of the first sentences that follows the hypernyms as shown in Table 4. For each of the first sentence from the Japanese Wikipedia pages, we repeatedly strip the suffixes shown in Table 4 until a noun is found. (The repetition is necessary for handling expressions such as *no-hitotsu dearu* that combine multiple suffixes.) And if the noun is among our candiate hypernyms, we consider the Wikipedia entry as a new supporting pattern.

Table 4 Suffix patterns for hypernymy extraction from Wikipedia first sentences.

de-aru	*no-koto*	*wo-iu*	*wo-sasu*
no-hitotsu	*no-isshu*	*no-ichi-bunya*	

Table 5 Examples of edges in the link label graph we generated from Wikipedia.

From	To
senkei risuto (linear lists)	*rinkudo risuto* (linked lists)
kankoku (Korea, abbreviated)	*daikan minkoku* (Korea, formally)
manga (comics, in Katakana)	*manga* (comics, in Kanji)
2shinho (binary number system)	*2shinsu* (binary number system)

3.4.2 Link Label Graph

In Wikipedia, entries have links to other entries, and these links often have different labels from the title of the target entry. We use the labels there as different strings that refer to the title to obtain more information about hypernyms. We generate a link label graph from Wikipedia texts, which is a graph with vertices of terms and directed edges of relationship that a term is used as a label of another term. Table 5 shows some examples. We can easily use it by combining with the term similarity graph using a constant instead of similarity.

The graph mainly contains information about the different words for the same meaning. Ideally, this type of information should be included in the term similarity graphs. However, in practice, the precision of term similarity graphs can be far from perfect. Though our link label graph contains fewer entries than the term similarity graph, the precision of our link label graph seems relatively high, and incorporating the graph can improve the precision of the result.

4. Experimental Setup

4.1 Linguistic Resources

We used publicly available web corpora: the Japanese subset of ClueWeb09 Dataset and Japanese Wikipedia. ClueWeb09 Dataset is a very large dataset of web documents collected by Carnegie Mellon University in 2009 and has been

Table 6 Examples of the stopwords we removed from the dictionary.

Problem	Examples
Too general	*kotoba* (word), *namae* (name), *aishou* (nickname)
Noisy for web corpora	*naiyou* (contents), *ichiran* (catalog), *daunro-do* (download)
Noisy for the patterns	*koto*, *mono*, *wake*

used by several tracks of the Text Retrieval Conference (TREC). The Japanese subset we used contains about sixty million documents.

We generated a dictionary of nouns from the titles of Japanese Wikipedia, Hatena Keywords[⋆1] and Japanese Wordnet[3]. The dictionary contains about 1.3 million words, including general words from Wordnet and proper names or latest words from Wikipedia and Hatena Keywords. We manually prepared some stopwords to avoid too much noise in the generated relations: some examples are shown in Table 6.

From the Japanese subset of ClueWeb09, we generated a term similarity graph using methods describe in 5).

4.2 Evaluation Set and Criteria

For evaluation, we selected 100 words as hyponyms from the dictionary. We first randomly selected 400 terms from the dictionary with the probability of a term T being selected to be $\log(1 + F(T))$, where $F(T)$ is the frequency of term T in the corpora. Then we manually selected 100 terms by considering diversity while eliminating words that are inappropriate as hyponyms. Basically this method follows the evaluation conducted in 9).

For each of the term from the test set, we apply one of the aforementioned automatic hyponym extraction methods and rank the hypernym candidates. Then the top ranked candidate was assessed by the first author of this paper. If the sentence "<term> is a (kind of) <candidate>" is semantically correct, this is counted as a correct output. Otherwise the candidate is treated as incorrect.

5. Results

5.1 Overall Performance

First, we present the evaluation results of our method and two methods following existing studies (see Table 7). Our first baseline is the pattern-based method by Ando et al.[1] and our second baseline is the Japanese version of the method described in Zhang et al.[9], which uses a term similarity graph. As the method by Zhang et al.[9] originally targeted English, our implemention of this method also uses the Japanese patterns of Ando et al.[1].

Table 7 Comparison of precision between our method and existing methods.

Method	Precision
Ando et al.[1]	0.63
Zhang et al.[9]	0.67
Ours	0.82

Our results show that our method outperforms the two baselines by 30% and 22% in terms of precision, respectively. We conducted two-tailed pairwise sign tests to test the statistical significance of the differences. Our method significantly outperforms other two methods at $\alpha = 0.01$.

One reason that may explain the large performance gap between our method and the baselines is that while our test set mainly contains Wikipedia title words and our method exploits Wikipedia, the baselines were not targeted specifically for Wikipedia. Even so, our first experimental results are encouraging.

It should be noted, however, that our implementations of the two baselines perform far worse than the corresponding results reported in 9). One possible reason is the difference of the size of corpora. In 9) they used approximately ten times as many web documents as these we used, and this can severely affect the precision. Another possible explanation would be that the precision for English does not easily carry over to other languages such as Japanese. For example, the patterns that signal hyponym relations are probably never equivalent across languages.

5.2 Comparison of Weight Models

Second, we compare the results between different weight models we discussed in Section 3.2 (see Table 8). *Linear* is one of the baselines using Eq. (1), *Original Nonlinear* is the other baseline using Eq. (3) and *New Nonlinear* is our method using Eq. (4). We used the term similarity graph without the link label graph for propagation with the scaling described in Section 3.3.

The table shows that the two nonlinear methods outperform *Linear* by 12%

[⋆1] http://d.hatena.ne.jp/keyword/

Table 8 Comparison of precision between different models for combining different evidence.

Formula	Precision
Linear	0.69
Original Nonlinear	0.77
New Nonlinear	0.80

Table 9 Comparison of precision with and without the scaling before the propagation.

Setting	Precision
No Scaling	0.74
Scaling	0.80

and 16%, respectively. Moreover, *New Nonlinear* outperforms *Original Nonlinear* by 4%.

We conducted two-tailed pairwise sign tests to test the statistical significance of the differences. Both of the nonlinear methods significantly outperform *Linear* at $\alpha = 0.05$. The two nonlinear methods are not significantly different from each other.

5.3 Comparison of Propagation Models

Next, we compare the results with and without the scaling we discussed in Section 3.3 (see Table 9). *No Scaling* is the result using Eq. (5) and *Scaling* is the result using Eq. (7). We used the term similarity graph without the link label graph and we used Eq. (4) for the weight model.

The table shows that *Scaling* outperforms *No Scaling* by 8%. However, according to a two-tailed pairwise sign test, the difference is not statistically significant.

5.4 Comparison of Propagation Sources

Next, we compare the evaluation results of different propagation sources (see Table 10). In the table, *Base* is the baseline without any propagation, *TSG* is the result with the propagation using the term similarity graph described in Section 3.3, *LLG* is the result with the propagation using the link label graph described in Section 3.4.2, and *TSG+LLG* is the result with the propagation combining the term similarity graph and the link label graph.

The table shows that *TSG*, *LLG* and *TSG+LLG* outperform *Base* (i.e. no propagation) by 16%, 4% and 19%, respectively.

We conducted two-tailed pairwise sign tests to test the statistical significance of the differences. *TSG+LLG* significantly outperforms *Base* at $\alpha = 0.05$. All other pairwise differences are not statistically significant.

It is also worth noting that while *TSG+LLG* managed to return one or more hypernym candidates for every term in the test set, *Base*, *TSG* and *LLG* failed to return a candidate for three, one and two terms, respectively. Thus, evidence propagation can improve not only precision but also recall.

Table 10 Comparison of precision between different sources for evidence propagation.

Source	Precision
Base	0.69
TSG	0.80
LLG	0.72
TSG+LLG	0.82

Table 11 Comparison of precision with and without the technique using first sentences in Wikipedia Entries as new patterns.

Setting	Precision
Only normal patterns	0.79
With the technique	0.82

5.5 Effect of First Sentences in Wikipedia Entries

Finally, we evaluate the effect of the technique using first sentences in Wikipedia entries as new patterns we discussed in Section 3.4.1 (see Table 11). We used Eq. (4) for weighting the relations and we used both the term similarity graph and the link label graph for propagation with the scaling described in Section 3.3.

The table shows that the Wikipedia-based technique improves the precision by 4%. However, according to a two-tailed pairwise sign test, the difference is not statistically significant.

6. Further Discussions

6.1 Examples and Analyses

Table 12 shows some examples of hyponymy relations extracted by our method.

It shows the method correctly extracted the hypernyms for both general terms like *daigaku* (university) or *nezumi* (mouse) and proper names like *nikkei-restaurant* (Nikkei Restaurant, a magazine about restaurants in Japan) or *saibouzu* (Cybozu, a popular groupware in Japan).

Gyorai (torpedo) is one of the examples that the propagation worked well. Without the propagation, the top hypernym for *gyorai* was *tasuu* (a large amount), which is an inappropriate hypernym of *gyorai.* However, all the top neighbors in the term similarity graph are weapons as we described in Table 2, and they supported to extract *buki* (weapon) as the hypernym.

Sanuki-shi (Sanuki City) and *nakama-shi* (Nakama City) are examples of the

Table 12 Examples of the extracted hypernyms.

Hyponym	Hypernym	Correct?
daigaku (university)	*kyouiku-kikan* (education institution)	Yes
nezumi (mouse)	*doubutu* (animal)	Yes
gyorai (torpedo)	*buki* (weapon)	Yes
nikkei resutoran (Nikkei Restaurant)	*zassi* (magazine)	Yes
saibouzu (Cybozu)	*gurupu-wea* (groupware)	Yes
sanuki-shi (Sanuki City)	*sougi-sha* (funeral parlor)	No
nakama-shi (Nakama City)	*sougi-sha* (funeral parlor)	No
soutai onkan (relative hearing)	*ongaku* (music)	No
seiki bunpu kyokusen (normal distribution curve)	*yasashisa* (grace)	No

words that we failed to mine the correct hypernyms, and it is interesting that both have *sougi-sha* (funeral parlor) as the wrong hypernym. It may be that some web pages that provide specific services such as "funeral parlor search" had adversely affected the results.

Soutai onkan (relative hearing) also has an incorrect hypernym *ongaku* (music). Including the previous examples, to eliminate such incorrect hypernyms that plays different roles in the world, combining similarity of dependency between verbs[6] may be promising future work.

Seiki bunpu kyokusen (normal distribution curve) is an example of the words that we failed to mine any correct hypernym. For this particular example, applying some prefix stripping rules (e.g. reducing *seiki bunpu kyokusen* to *kyokusen* (curve)) may be used for obtaining hypernyms. However, in general, it is a challenging problem to automatically determine that there is no appropriate hypernym for a given term.

6.2 Other Viewpoints

As we described in Section 5.1, the precision of our method in our experiment was 0.82. Note that the set of nouns we prepared includes about 1.3 million words, and this precision is based on the set of words that we selected to some extent randomly. Therefore, although we have not evaluated the entire set of extracted hypernym relations exhaustively, we expect this large data set to be of reasonable accuracy.

Moreover, although our evaluation examined only the top ranked hypernym candidate, our data set contains many alternative candidates which may also be useful. Retaining multile hypernym candidates for each hyponym may be useful for some applications.

7. Conclusions

In this paper, we proposed a method for automatically extracting hyponymy relations between Japanese nouns from web documents. Our method is based on a existing method utilizing a term similarity graph, but it has improved formulas for deciding weight and propagating evidence, and it treats Wikipedia as a special data source and exploit the information there. Our experiments showed that our method can extract large-scale hyponymy relations with 82% precision and it improved the precision statistically significantly from existing methods.

Our future work includes combining similarity of syntactic dependency to further improve the precision. Moreover, we also would like to evaluate the results with different criteria such as precision in the top-5 results.

References

1) Ando, M., Sekine, S. and Ishiza, S.: Automatic extraction of hyponyms from Japanese newspaper using lexico-syntactic patterns, *Proc. of LREC* (2004).
2) Hearst, M.: Automatic acquisition of hyponyms from large text corpora, *Proc. of COLING* (1992).
3) Kyoko, K., Bond, F., Tomuro, N. and Hitoshi, I.: Extraction of attribute concepts from Japanese adjectives., *Proc. of LREC* (2010).
4) Ponzetto, S.P. and Strube, M.: Deriving a large scale taxonomy from Wikipedia, *Proc. of AAAI* (2007).
5) Shi, S., Zhang, H., Yuan, X. and Wen, J.: Corpus-based semantic class mining: distributional vs. pattern-based approaches, *Proc. of COLING* (2010).
6) Shinzato, K. and Torisawa, K.: Acquiring hyponymy relations from web documents, *Proc. of HLT-NAACL* (2004).
7) Sumida, A. and Torisawa, K.: Hacking Wikipedia for hyponymy relation acquisition, *Proc. of IJCNLP* (2008).
8) Yamada, I., Torisawa, K., Kazama, J., Kuroda, K., Murata, M., DeSaeger, S., Bond, F. and Sumida, A.: Hypernym discovery based on distributional similarity and hierarchical structures, *Proc. of EMNLP* (2009).
9) Zhang, F., Shi, S., Liu, J., Sun, S. and Lin, C.: Nonlinear evidence fusion and propagation for hyponymy relation mining, *Proc. of ACL-HLT* (2011).