Hacking Wikipedia for Hyponymy Relation Acquisition

Asuka Sumida **Kentaro Torisawa**
Japan Advanced Institute of Science and Technology
1-1 Asahidai, Nomi-shi, Ishikawa-ken, 923-1211 JAPAN
{a-sumida,torisawa}@jaist.ac.jp

Abstract

This paper describes a method for extracting a large set of hyponymy relations from Wikipedia. The Wikipedia is much more consistently structured than generic HTML documents, and we can extract a large number of hyponymy relations with simple methods. In this work, we managed to extract more than 1.4×10^6 hyponymy relations with 75.3% precision from the Japanese version of the Wikipedia. To the best of our knowledge, this is the largest machine-readable thesaurus for Japanese. The main contribution of this paper is a method for hyponymy acquisition from hierarchical layouts in Wikipedia. By using a machine learning technique and pattern matching, we were able to extract more than 6.3×10^5 relations from hierarchical layouts in the Japanese Wikipedia, and their precision was 76.4%. The remaining hyponymy relations were acquired by existing methods for extracting relations from definition sentences and category pages. This means that extraction from the hierarchical layouts almost doubled the number of relations extracted.

1 Introduction

The goal of this study has been to automatically extract a large set of hyponymy relations, which play a critical role in many NLP applications, such as Q&A systems (Fleischman et al., 2003). In this paper, hyponymy relation is defined as a relation between a hypernym and a hyponym when "the *hyponym* is a (kind of) *hypernym*."[1].

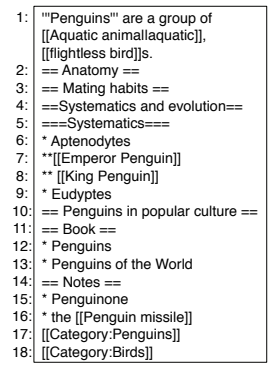
Currently, most useful sources of hyponymy relations are hand-crafted thesauri, such as WordNet (Fellbaum, 1998). Such thesauri are highly reliable, but their coverage is not large and the costs of extension and maintenance is prohibitively high. To reduce these costs, many methods have been proposed for automatically building thesauri (Hearst, 1992; Etzioni et al., 2005; Shinzato and Torisawa, 2004; Pantel and Pennacchiotti, 2006). But often these methods need a huge amount of documents and computational resources to obtain a reasonable number of hyponymy relations, and we still do not have a thesaurus with sufficient coverage.

In this paper, we attempt to extract a large number of hyponymy relations without a large document collection or great computational power. The key idea is to focus on Wikipedia[2], which is much more consistently organized than normal documents. Actually, some studies have already attempted to extract hyponymy relations or semantic classifications from Wikipedia. Hyponymy relations were extracted from definition sentences (Herbelot and Copestake, 2006; Kazama and Torisawa, 2007). Disambiguation of named entities was also attempted (Bunescu and Pasca, 2006). Category pages were used to extract semantic relations (Suchanek et al., 2007). Lexical patterns for semantic relations were learned (Ruiz-Casado et al., 2005).

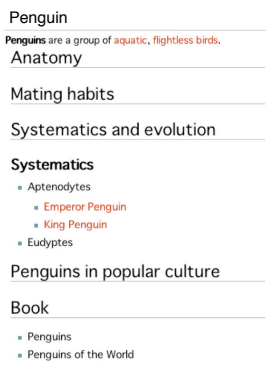
The difference between our work and these attempts is that we focus on the hierarchical layout of normal articles in Wikipedia. For instance, the article titled "Penguin" is shown in Fig. 1(b). This article has a quite consistently organized hierarchical structure. The whole article is divided into the sections "Anatomy", "Mating habits", "Systematics and evolution", "Penguins in popular culture" and so on. The section "Systematics and evolution" has the

[1]This is a slightly modified definition of the one in (Miller et al., 1990). Linguistic literature, e.g. (A.Cruse, 1998), distinguishes hyponymy relations, such as "national university" and "university", and concept-instance relations, such as "Tokyo University" and "university". However, we regard concept-instance relations as a part of hyponymy relations in this paper because we think the distinction is not crucial for many NLP applications.

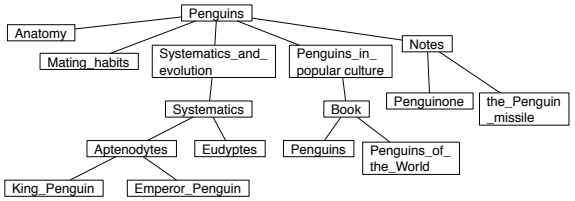
[2]http://ja.wikipedia.org/wiki



(a) The source code of the article "Penguin' in 'Wikipedia



(b) The example of the article "Penguin" in Wikipedia



(c) The displayed page of the article "Penguin" in Wikipedia

Figure 1: The example of a Wikipedia article

subsection "Systematics", which is further divided to "Aptenodytes", "Eudyptes" and so on. Some of such section-subsection relations can be regarded as valid hyponymy relations. In the article about "Penguin", relations such as the one between "Aptenodytes" and "Emperor Penguin" and the one between "Book" and "Penguins of the World" are valid hyponymy relations. The main objective of this work is to develop a method to extract *only* such hyponymy relations.

The rest of the paper is organized as follows. We first explain the structure of Wikipedia in Section 2. Next, we introduce our method in Section 3. Some alternative methods are presented in Section 4. We then show the experimental results in Section 5.

2 The Structure of Wikipedia

The Wikipedia is built on the MediaWiki software package[3]. MediaWiki interprets the source code written in the MediaWiki syntax to produce human-readable web pages. For example, Fig. 1(b) is a result of interpreting the source code in Fig. 1(a). An important point is that the MediaWiki syntax is stricter than the HTML syntax and usage of the syntax in most Wikipedia articles are constrained by editorial policy. This makes it easier to extract information from the Wikipedia than from generic HTML documents.

[3] `http://www.mediawiki.org/wiki/MediaWiki`

Usually, a Wikipedia article starts with a definition sentence, such as "Penguins are a group of aquatic, flightless birds" in Fig. 1(a). Then, the hierarchical structure marked in the following manner follows.

Headings Headings describe the subject of a paragraph. See line 2-5, 10-11, 14 of Fig. 1(a). Headings are marked up as "=+*title*=+" in the MediaWiki syntax, where *title* is a subject of the paragraph. Note that "+" here means a finite number of repetition of symbols. "=+*section*=+" means that "=*section*=", "==*section*==" and "===*section*===" are legitimate mark up in the Wikipedia syntax. We use this '+' notation in the following explanation as well.

Bulleted lists Bulleted lists are lists of unordered items. See line 6-9, 12-13, 15-16 of Fig. 1. Bulleted lists are marked as "*+*title*" in the MediaWiki syntax, where *title* is a subject of a listed item.

Ordered lists Ordered lists are lists of numbered items. Ordered lists are marked up as "#+*title*" in MediaWiki syntax, where *title* is a subject of a numbered item.

Definition lists Definition lists contain terms and its definitions. Our method focuses only on the terms. Definition lists are marked as ";*title*" where *title* is a term.

The basic hierarchical structure of a Wikipedia article is organized by a pre-determined ordering among the above items. For instance, a bulleted list item is assumed to occupy a lower position in the hierarchy than a heading item. In general, items occupy a higher position in the order of headings, definition lists, bulleted lists, and ordered lists. In addition, recall that headings, bullet list and ordered list allowed the repetitions of symbols "=", "*" and "#". The number of repetition indicates the position in the hierarchy and the more repetition the item contains, the lower the position occupied by the item becomes. For instance, "==Systematics and evolution==" occupies a higher position than "===Systematics===" as illustrated in Fig. 1(a) (b).

Then, it is easy to extract a hierarchical structure based on the order among the mark-up items by parsing the source code of an article. Fig. 1(c) illustrates the hierarchical structure extracted from the source code in Fig. 1(a).

3 Proposed Method

This section describes our method for extracting hyponymy relations from hierarchical structures in Wikipedia articles. The method consists of three steps:

Step 1 Extract hyponymy relation candidates from hierarchical structures in the Wikipedia.

Step 2 Select proper hyponymy relations by applying simple patterns to the extracted candidates.

Step 3 Select proper hyponymy relations from the candidates by using a machine learning technique.

Each step is described below.

3.1 Step 1: Extracting Relation Candidates

The Step 1 procedure extracts the *title* of a marked-up item and a *title* of its (direct) subordinate marked-up item as a hyponymy relation for each marked-up item. For example, given the hierarchy in Fig. 1(c), the Step1 procedure extracted hyponymy relation candidates such as "Aptenodytes/Emperor Penguin"and "Book/Penguins of the World". (Note that we denote hyponymy relations or their candidates as "*hypernym*/*hyponym*" throughout this paper.) However, these relation candidates include many wrong hyponymy relations such as "Penguins in popular culture/Book". Steps 2 and 3 select proper relations from the output of the Step 1 procedure.

3.2 Step 2: Selecting Hyponymy Relations by Simple Patterns

Step 2 selects plausible hyponymy relations by applying simple patterns to hyponymy relation candidates obtained in Step 1. This is based on our observation that if a hypernym candidate matches a particular pattern, it is likely to constitute a correct relation. For example, in Japanese, if a hypernym candidate is " `omona` X (Popular *or* typical X)", X is likely to be a correct hypernym of the hyponym candidates that followed it in the article. Fig.2 shows a Japanese Wikipedia article about a zoo that includes "`omona doubutsu` (Popular animals)", " `Mazeran Pengin` (Magellanic Penguin)", "`Raion` (Lion)" and so on. From this article, the Step 1 procedure extracts a hyponymy relation candidate "Popular Animals/Magellanic Penguin", and the Step 2 procedure extracts "Animals/Magellanic Penguin" after matching "Popular" to the hypernym candidate and removing the string "Popular" from the candidate. Fig. 3 lists all the patterns we used. Note that the non-variable part of the patterns is removed from the matched hypernym candidates.

Figure 2: Example for Step2

`Xno ichiran`(list of X), `Xichiran`(list of X), `Xsyousai`(details of X), `Xrisuto`(X list), `daihyoutekinaX`(typical X), `daihyouX`(typical X), `syuyounaX`(popular or typical X), `omonaX`(popular or typical X), `syuyouX`(popular or typical X), `kihontekinaX`(basic X), `kihon`(basic X), `chomeinaX`(notable X), `ookinaX`(large X), `omonaX`(popular or typical X), `ta noX`(other X), `ichibuX`(partial list of X)

Figure 3: Patterns for Step 2

3.3 Step 3: Selecting Proper Hyponymy Relations by Machine Learning

The Step 3 procedure selects proper hyponymy relations from the relation candidates that do not match the patterns in Step 2. We use Support Vector Machines (SVM) (Vapnik, 1998) for this task. For each hyponymy relation candidate, we firstly apply morphological analysis and obtain the following types of features for each hypernym candidate and hyponym candidate, and append them into a single feature vector, which is given to the classifier.

POS We found that POS tags are useful clues for judging the validity of relations. For instance, if a hypernym includes proper nouns (and particularly toponyms), it is unlikely to constitute a proper relation. We assigned each POS tag a unique dimension in the feature space and if a hypernym/hyponym consists of a morpheme with a particular POS tag, then the corresponding element of the feature vector was set to one. When hypernyms/hyponyms are multiple morpheme expressions, the feature vectors for every morpheme were simply summed. (The obtained feature vector works as *disjunction* of each feature vector.) An important point is that, since the last morpheme of hypernyms/hyponyms works as strong evidence for the validity of relations, the POS tag of the last morpheme was mapped to the dimension that is different from the POS tags of the other morphemes.

MORPH Morphemes themselves are also mapped to a dimension of the feature vectors. The last morphemes are also mapped to dimensions that are different from those of the other morphemes. This feature is used for recognizing particular morphemes that strongly suggest the validity of hyponymy relations. For instance, if the morpheme "`zoku` (genus)" comes in the end of the hypernym, the relation is likely to be valid, as exemplified by the relation "`koutei pengin zoku` (Aptenodytes genus)/`koutei pengin` (Emperor Penguin)".

EXP Expressions of hypernym/hyponym candidates themselves also give a good clue for judging the validity of the relation. For instance, there are typical strings that can be the title of a marked-up item but cannot be a proper hypernym or a proper hyponym. Examples of these strings include "Background" and "Note". By mapping each expression to an element in a feature vector and setting the element to one, we can prevent the candidates containing such expressions from being selected by the classifier.

ATTR We used this type of features according to our observation that if a relation candidate includes an *attribute*, it is a wrong relation. The attributes of an object can be defined as "what we want to know about the object". For instance, we regard "Anatomy" as attributes of creatures in general, and the relation such as "Penguin/Anatomy" cannot be regarded as proper hyponymy relations. To set up this type of features, we automatically created a set of attributes and the feature was set to one if the hypernym/hyponym is included in the set. The attribute set was created in the following manner. We collected all the titles of the marked-up items from all the articles, and counted the occurrences of each title. If a title appears more than one time, then it was added to the attribute set. Note that this method relies on the hypothesis that the same attribute is used in articles about more than one object (e.g., "Penguin" and "Sparrow") belonging to the same class (e.g., "animal"). (Actually, in this counting of titles, we excluded the titles of items in the bulleted lists and the ordered lists in the bottom layer of the hierarchical structures. This is because these items are likely to constitute valid hyponymy relations. We also excluded that match the patterns in Fig. 3.) As a result, we obtained the set of 40,733 attributes and the precision of a set was 73% according to the characterization of attributes in (Tokunaga et al., 2005).

LAYER We found that if a hyponymy relation is extracted from the bottom of the hierarchy, it tends to be a correct relation. For example, in Fig. 1(c), the hyponymy relation "Penguin/Anatomy" which is extracted from the top of hierarchy is wrong, but the hyponymy relation "Aptenodytes/Emperor Penguin " which is extracted from the bottom of the layer is correct. To capture this tendency, we added the mark that marks up a hypernym and a hyponym to the features. Each mark is mapped to a dimension in the feature vector, and the corresponding element was set to one if a hypernym/hyponym candidate appears with the mark.

As the final output of our method, we merged the results of Steps 2 and 3.

4 Alternative Methods

This section describes existing methods for acquiring hyponymy relations from the Wikipedia. We compare the results of these methods with the output of our method in the next section.

4.1 Extraction from Definition Sentences

Definition sentences in the Wikipedia article were used for acquiring hyponymy relations by (Kazama and Torisawa, 2007) for named entity recognition. Their method is developed for the English version of the Wikipedia and required some modifications to the Japanese version. These modification was inspired by Tsurumaru's method (Tsurumaru et al., 1986).

Basically, definition sentences have forms similar to "*hyponym word* wa *hypernym word* no isshu de aru(*hyponym* is a kind of *hypernym*)" in dictionaries in general, and contain hyponymy relations in them. In the Wikipedia, such sentences usually come just after the titles of articles, so it is quite easy to recognize them. To extract hyponymy relations from definition sentences, we manually prepared 1,334 patterns, which are exemplified in Table 4, and applied them to the first sentence.

4.2 Extraction from Category Pages

Suchanek et al. (Suchanek et al., 2007) extracted hyponymy relations from the category pages in the Wikipedia using WordNet information. Although we cannot use WordNet because there is no Japanese version of WordNet, we can apply their idea to the Wikipedia only.

The basic idea is to regard the pairs of the category name provided in the top of a category page and the

hyponym wa.∗*hypernym* no hitotsu.
(*hyponym* is one of *hypernym*)
hyponym wa .∗*hypernym* no daihyoutekina mono dearu.
(*hyponym* is a typical *hypernym*)
hyponym wa.∗*hypernym* no uchi no hitotsu.
(*hyponym* is one of *hypernym*)

Note that *hyponym* and *hypernym* match only with NPs.

Figure 4: Examples of patterns for definition sentences

items listed in the page as hyponymy relation.

Thus, the method is quite simple. But the relations extracted by this are not limited to hyponymy relations, unfortunately. For instance, the category page "football" includes "football team". Such loosely associated relations are harmful for obtaining precise relations. Suchanek used WordNet to prevent such relations from being included in the output. However, we could not develop such a method because of the lack of a Japanese WordNet.

5 Experiments

For evaluating our method, we used the Japanese version of Wikipedia from March 2007, which includes 820,074 pages[4]. Then, we removed "user pages","special pages", "template pages", "redirection pages", and "category pages" from it.

In Step 3, we used TinySVM[5] with polynomial kernel of degree 2 as a classifier. From the relation candidates given to the Step 3 procedure, we randomly picked up 2,000 relations as a training set, and 1,000 relations as a development set. We also used the morphological analyzer MeCab [6] in Step 3.

Table 1 summarizes the performance of our method. Each row of the table shows A) the precision of the hyponymy relations, B) the number of the relations, and C) the expected number of correct relations estimated from the precision and the number of the extracted relations, after each step of the procedure. Note that Step 2' indicates the hyponymy relation candidates that *did not* match the pattern in Fig.3 and that were given to the Step 3 procedure. The difference between Step 2' and Step 3 indicates the effect of our classifier. Step 2&3 is the final result obtained by merging the results of Step 2 and Step 3. As the final output, we obtained more than 6.3×10^5 relations and their precision was 76.4%. Note that the precision was measured by checking 200 random samples for each step except for Step 3 and Step 2&3, for which the precision was obtained in a way described later. Note that all the numbers were obtained after removing duplicates in the relations. Example of the relations recognized by Step 2 or Step 3 are shown in Fig. 5.

[4]This pages include "incomplete pages" that are not counted in the number of pages presented in the top page of the Wikipedia.

[5]http://chasen.org/ taku/software/TinySVM/index.html

[6]http://mecab.sourceforge.net

Table 1: Performance of each step

	Precision	# of rels.	estimated # of correct rels.
Step 1	44%	2,768,856	1,218,296
Step 2	71.5%	221,605	158,447
Step 2'	40.0%	2,557,872	1,023,148
Step 3	78.1%	416,858	325,670
Step 2 & 3	76.4%	633,122	484,117

aatisuto	/	erubisu puresurii
Artist	/	Elvis Presley
sakura	/	someiyoshino
Cherry Blossom	/	Yoshino Cherry
heiya	/	nakagawa heiya
Plain	/	Nakagawa Plain
ikou oyobi kenzoubutsu	/	tsuki no piramiddo
Ruins and buildings	/	the Pyramid of the Moon
suponsaa	/	genzai*
Sponsors	/	Present*
shutsuen sakuhin	/	taidan go*
Art work	/	After leaving a group*

"*" indicates an incorrectly recognized relation.

Figure 5: Examples of acquired hyponymy relations

Table 2 shows the effect of each type of features in Step 3. Each row indicates the precision, recall and F-measure against 400 samples that are randomly selected from the relation candidates given to Step 3, when we removed a type of features from feature vector and when we used all the types. (The 400 samples included 142 valid relations.) We can see that all types except for LAYER contributed to an improvement of the F-measure. When the LAYER features were removed, the F-measure was improved to 1.1 but the precision was on an unacceptable level (55%) and cannot be used in actual acquisition.

Table 3 summarizes the statistics of all the methods for acquisition from Wikipedia. It shows A) the pre-

Table 2: Effect of each features in Step3

Feature Type	a Precision	Recall	F-measure
-POS	60.0%	57.0%	58.4
-MORPH	85.0%	47.8%	61.2
-EXP	82.2%	35.9%	50.0
-ATTR	79.7%	47.1%	59.2
-LAYER	55.0%	76.7%	64.1
ALL	78.1%	52.8%	63.0

Table 3: The result for extracting hyponymy relations from definition sentences, category structures,and hierarchy structures

	Precision	# of rels.	# of correct rels.
Hierarchy (Proposed)	76.4 %	633,122	484,117
Definition snts	77.5%	220,892	171,191
Category	70.5%	596,463	420,506
Total	75.3%	1,426,861	1,075,814

cision of the relations (200 random samples), B) the number of relations, and C) the expected number of correct relations estimated from the precision and the number of extracted relations. We obtained 1.4×10^6 hyponymy relations without duplication in total with 75.3% precision from definition sentences, category structures, and hierarchical structures. They covered 6.6×10^5 distinct hyponyms and 1.0×10^5 distinct hypernyms. Note that the number of duplicated relations in these results was just 23,616. This suggests that we could extract different types of hyponymy relations from each of these methods.

6 Conclusion

This paper described a method for extracting a large set of hyponymy relations from the hierarchical structures of articles in Wikipedia. We could extract 633,122 relations from hierarchical layouts in the Japanese Wikipedia and their precision was 76.4%. Combining with existing methods that extract relations from definition sentences and category structures, we were able to extract 1,426,861 relations with 75.3% precision in total without duplication. To the best of our knowledge, this is the largest machine-readable thesaurus for Japanese available.

References

D. A.Cruse. 1998. *Lexical Semantics*. Cambridge Textbooks in Linguistics.

Razvan C. Bunescu and Marius Pasca. 2006. Using encyclopedic knowledge for named entity disambiguation. In *Proceedings of the 11th Conference of the EACL*, pages 9–16.

O. Etzioni, M. Cafarella, D. Downey, A. Popescu, T. Shaked, S. Soderland, D. S. Weld, and A. Yates. 2005. Unsupervised named-entity extraction from the web: an experimental study. *Artif. Intell.*, 165(1):91–134.

Christiane Fellbaum, editor. 1998. *WordNet: an electronic lexical database*. MIT Press.

Michael Fleischman, Eduard Hovy, and Abdessamad Echihabi. 2003. Offline strategies for online question answering: Answering questions before they are asked. In *ACL2003*, pages 1–7.

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

Aurelie Herbelot and Ann Copestake. 2006. Acquiring ontological relationships from wikipedia using rmrs. In *Proceedings of the ISWC 2006 Workshop on Web Content Mining with Human Language Technologies*.

Jun'ichi Kazama and Kentaro Torisawa. 2007. Exploiting wikipedia as external knowledge for named entity recognition. In *Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, pages 698–707.

George A. Miller, Richard Beckwith, Christiane Fellbaum, Derek Gross, and Katherine J. Miller. 1990. Introduction to wordnet: An on-line lexical database. In *Journal of Lexicography*, pages 235–244.

Patrick Pantel and Marco Pennacchiotti. 2006. Espresso: leveraging generic patterns for automatically harvesting semantic relations. In *ACL '06 : Proceedings of the 21st International Conference on Computational Linguistics and the 44th annual meeting of the ACL*, pages 113–120.

Maria Ruiz-Casado, Enrique Alfonseca, and Pablo Castells. 2005. Automatic extraction of semantic relationships for wordnet by means of pattern learning from wikipedia. In *NLDB*, pages 67–79.

Keiji Shinzato and Kentaro Torisawa. 2004. Acquiring hyponymy relations from web documents. In *HLT-NAACL '04 : Proceedings of Human Language Technology Conference/North American chapter of the Association for Computational Linguistics annual meeting*, pages 73–80.

Fabian M. Suchanek, Gjergji Kasneci, and Gerhard Weikum. 2007. Yago: A core of semantic knowledge unifying wordnet and wikipedia. In *WWW '07 : Proceedings of the 16th International World Wide Web Conference*.

Kosuke Tokunaga, Jun'ichi Kazama, and Kentaro Torisawa. 2005. Automatic discovery of attribute words from web documents. In *IJCNLP 2005*, pages 106–118.

Hiroaki Tsurumaru, Toru Hitaka, and Sho Yoshida. 1986. An attempt to automatic thesaurus construction from an ordinary japanese language dictionary. In *Proceedings of the 11th conference on Computational linguistics*, pages 445–447.

Vladimir N. Vapnik. 1998. *Statistical Learning Theory*. Wiley-Interscience.