Extending a thesaurus by classifying words

Tokunaga Takenobu **Fujii Atsushi** **Iwayama Makoto**
Sakurai Naoyuki **Tanaka Hozumi**

Department of Computer Science **Advanced Research Lab.**
Tokyo Institute of Technology **Hitachi Ltd.**
take@cs.titech.ac.jp

Abstract

This paper proposes a method for extending an existing thesaurus through classification of new words in terms of that thesaurus. New words are classified on the basis of relative probabilities of.a word belonging to a given word class, with the probabilities calculated using noun-verb co-occurrence pairs. Experiments using the Japanese *Bunruigoihyô* thesaurus on about 420,000 co-occurrences showed that new words can be classified correctly with a maximum accuracy of more than 80%.

1 Introduction

For most natural language processing (NLP) systems, thesauri comprise indispensable linguistic knowledge. *Roget's International Thesaurus* [Chapman, 1984] and *WordNet* [Miller *et al.*, 1993] are typical English thesauri which have been widely used in past NLP research [Resnik, 1992; Yarowsky, 1992]. They are handcrafted, machine-readable and have fairly broad coverage. However, since these thesauri were originally compiled for human use, they are not always suitable for computer-based natural language processing. Limitations of handcrafted thesauri can be summarized as follows [Hatzivassiloglou and McKeown, 1993; Uramoto, 1996; Hindle, 1990].

- limited vocabulary size
- unclear classification criteria
- building thesauri by hand requires considerable time and effort

The vocabulary size of typical handcrafted thesauri ranges from 50,000 to 100,000 words, including general words in broad domains. From the viewpoint of NLP systems dealing with a particular domain, however, these thesauri include many unnecessary (general) words and do not include necessary domain-specific words.

The second problem with handcrafted thesauri is that their classification is based on the intuition of lexicographers, with their classification criteria not always being clear. For the purposes of NLP systems, their classification of words is sometimes too coarse and does not provide sufficient distinction between words, or is sometimes unnecessarily detailed.

Lastly, building thesauri by hand requires significant amounts of time and effort even for restricted domains. Furthermore, this effort is repeated when a system is ported to another domain.

This criticism leads us to automatic approaches for building thesauri from large corpora [Hirschman *et al.*, 1975; Hindle, 1990; Hatzivassiloglou and McKeown, 1993; Pereira *et al.*, 1993; Tokunaga *et al.*, 1995; Ushioda, 1996]. Past attempts have basically taken the following steps [Charniak, 1993].

(1) extract word co-occurrences

(2) define similarities (distances) between words on the basis of co-occurrences

(3) cluster words on the basis of similarities

The most crucial part of this approach is gathering word co-occurrence data. Co-occurrences are usually gathered on the basis of certain relations such as predicate-argument, modifier-modified, adjacency, or mixture of these. However, it is very difficult to gather sufficient co-occurrences to calculate similarities reliably [Resnik, 1992; Basili *et al.*, 1992]. It is sometimes impractical to build a large thesaurus from scratch based on only co-occurrence data.

Based on this observation, a third approach has been proposed, namely, combining linguistic knowledge and co-occurrence data [Resnik, 1992; Uramoto, 1996]. This approach aims at compensating the sparseness of co-occurrence data by using existing linguistic knowledge, such as *WordNet*. This paper follows this line of research and proposes a method to extend an existing thesaurus by classifying new words in terms of that thesaurus. In other words, the proposed method identifies appropriate

word classes of the thesaurus for a new word which is not included in the thesaurus. This search process is facilitated based on the probability that a word belongs to a given word class. The probability is calculated based on word co-occurrences. As such, this method could also suffer from the data sparseness problem. As Resnik pointed out, however, using the thesaurus structure (classes) can remedy this problem [Resnik, 1992].

2 Core thesaurus

Bunruigoihyô (BGH for short) [Hayashi, 1966] is a typical Japanese thesaurus, which has been used for much NLP research on Japanese. BGH includes 87,743 words, each of which is assigned an 8 digit class code. Some words are assigned more than one class code. The coding system of BGH has a hierarchical structure, that is, the first digit represents the part(s) of speech of the word (1: noun, 2:verb, 3: adjective, 4: others), and the second digit classifies words sharing the same first digit and so on. Thus BGH can be considered as four trees, each of which has 8 levels in depth (see figure 1), with each leaf as a set of words.

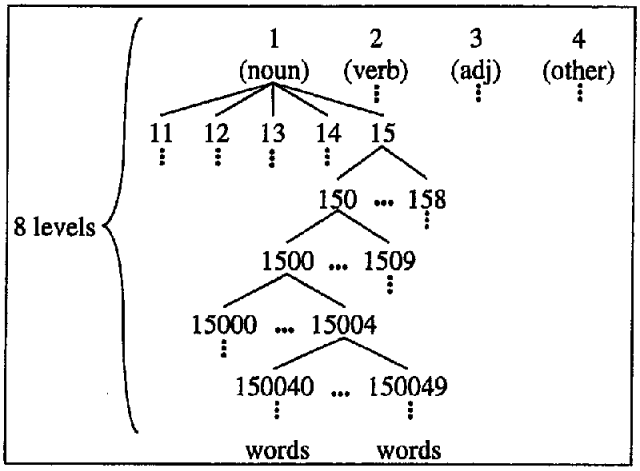

Fig. 1 Structure of *Bunruigoihyô* (BGH)

This paper focuses on classifying only nouns in terms of a class code based on the first 5 digits, namely, up to the fifth level of the noun tree. Table 1 shows the number of words (#words) and the number of 5 digit class codes (#classes) with respect to each part of speech.

Table 1 Outline of *Bunruigoihyô* (BGH)

POS	noun	verb	adj	other	total
#words	55,443	21,669	9,890	741	87,743
#classes	544	165	190	24	842

3 Co-occurrence data

Appropriate word classes for a new word are identified based on the probability that the word belongs to different word classes. This probability is calculated based on co-occurrences of nouns and verbs. The co-occurrences were extracted from the RWC text base RWC-DB-TEXT-95-1 [Real World Computing Partnership, 1995]. This text base consists of 4 years worth of Mainiti Shimbun [Mainichi Shimbun, 1991-1994] newspaper articles, which have been automatically annotated with morphological tags. The total number of morphemes is about 100 million. Instead of conducting full parsing on the texts, several heuristics were used in order to obtain dependencies between nouns and verbs in the form of tuples (frequency, noun, postposition, verb). Among these tuples, only those which include the postposition "*WO*" (typically marking *accusative* case) were used. Further, tuples containing nouns in BGH were selected. In the case of a compound noun, the noun was transformed into the maximal leftmost string contained in BGH[1]. As a result, 419,132 tuples remained including 23,223 noun types and 9,151 verb types. These were used in the experiments described in section 5.

4 Identifying appropriate word classes

4.1 Probabilistic model

The probabilistic model used in this paper is the SVMV model [Iwayama and Tokunaga, 1994]. This model was originally developed for document categorization, in which a new document is classified into certain predefined categories. For the purposes of this paper, a new word (noun) not appearing in the thesaurus is treated as a new document, and a word class in the thesaurus corresponds to a predefined document category. Each noun is represented by a set of verbs co-occurring with that noun. The probability $P(c_i|w)$ is calculated for each word class c_i, and the proper classes for a word w are determined based on it. The SVMV model formalizes the probability $P(c|w)$ as follows.

Conditioning $P(c|w)$ on each possible event gives

$$P(c|w) = \sum_{v_i} P(c|w, V = v_i)P(V = v_i|w). \quad (1)$$

Assuming conditional independence between c and $V = v_i$ given w, that is $P(c|w, V = v_i) = P(c|V = v_i)$, we obtain

$$P(c|w) = \sum_{v_i} P(c|V = v_i)P(V = v_i|w). \quad (2)$$

Using Bayes' theorem, this becomes

$$P(c|w) = P(c) \sum_{v_i} \frac{P(V = v_i|c)P(V = v_i|w)}{P(V = v_i)}. \quad (3)$$

All the probabilities in (3) can be estimated from training data based on the following equations. In the following, $f_r(w, v)$ denotes the frequency that a noun w and a verb v are co-occurring.

[1]For Japanese compound noun, the final word tends to be a semantic head.

$P(V = v_i|c)$ is the probability that a randomly extracted verb co-occurring with a noun is v_i, given that the noun belongs to word class c. This is estimated from the relative frequency of v_i co-occurring with the nouns in word class c, namely,

$$\hat{P}(V = v_i|c) = \frac{\sum_{w \in c} f_r(w, v_i)}{\sum_i \sum_{w \in c} f_r(w, v_i)}. \quad (4)$$

$P(V = v_i|w)$ is the probability that a randomly extracted verb co-occurring with a noun w is v_i. This is estimated from the relative frequency of v_i co-occurring with noun w, namely,

$$\hat{P}(V = v_i|w) = \frac{f_r(w, v_i)}{\sum_i f_r(w, v_i)}. \quad (5)$$

$P(V = v_i)$ is the prior probability that a randomly extracted verb co-occurring with a randomly selected noun is v_i. This is estimated from the relative frequency of v_i in the whole training data, namely,

$$\hat{P}(V = v_i) = \frac{\sum_w f_r(w, v_i)}{\sum_i \sum_w f_r(w, v_i)}. \quad (6)$$

$P(c)$ is the prior probability that a randomly selected noun belongs to c. This is estimated from the relative frequency of a verb co-occurring with any noun in class c[2], namely,

$$\hat{P}(c) = \frac{\sum_{w \in c} \sum_v f_r(w, v)}{\sum_c \sum_{w \in c} \sum_v f_r(w, v)}. \quad (7)$$

4.2 Searching through the thesaurus

As is documented by the fact that we employ the probabilistic model used in document categorization, classifying words in a thesaurus is basically the same as document categorization [3]. Document categorization strategies can be summarized according to the following three types [Iwayama and Tokunaga, 1995].

- the k-nearest neighbor (k-nn) or Memory based reasoning (MBR) approach
- the category-based approach
- the cluster-based approach

The k-nn approach searches for the k documents most similar to a target document in training data, and assigns that category with the highest distribution in the k documents [Weiss and Kulikowski, 1991]. Although the k-nn approach has been promising for document categorization [Masand *et al.*, 1992], it requires significant computational resources to calculate the similarity between a target document and every document in training data.

In order to overcome the drawback of the k-nn approach, the category-based approach first makes a cluster for each category consisting of documents assigned the same category, then calculates the similarity between a target document and each of these document clusters. The number of similarity calculations can be reduced to the number of clusters (categories), saving on computational resources.

Another alternative is the cluster-based approach, which first constructs clusters from training data by using some clustering algorithm, then calculates similarities between a target document and those clusters. The main difference between category-based and cluster-based approaches resides in the cluster construction. The former uses categories which have been assigned to documents when constructing clusters, while the latter does not. In addition, clusters are structured in a tree when a hierarchical clustering algorithm is used for the latter approach. In this case, one can adopt a top-down tree search strategy for similar clusters, saving further computational overhead.

In this paper, all these approaches are evaluated for word classification, in which a target document corresponds to a target word and a document category corresponds to a thesaurus class code.

5 Experiments

In our experiments, the 23,223 nouns described in section 3 were classified in terms of the core thesaurus, BGH, using the three search strategies described in the previous section. Classification was conducted for each strategy as follows.

k-nn Each noun is considered as a singleton cluster, and the probability that a target noun is classified into each of the non-target noun clusters is calculated.

category-based 10-fold cross validation was conducted for the category-based and cluster-based strategies, in that, 23,223 nouns were randomly divided into 10 groups, and one group of nouns was used for test data while the rest was used for training. The test group was rotated 10 times, and therefore, all nouns were used as a test case. The results were averaged over these 10 trials. Each noun in the training data was categorized according to its BGH 5 digit class code, generating 544 category clusters (see Table 1). The probability of each noun in the test data being classified into each of these 544 cluster was calculated.

cluster-based In the case of the category-based approach, each noun in the training data was categorized into the leaf clusters of the BGH tree, that is,

[2] This calculation seems be counterintuitive. A more straightforward calculation would be one based on the relative frequency of words belonging to class c. However, the given estimation is necessary in order to normalize the sum of the probabilities $P(c|w)$ to one.

[3] As Uramoto mentioned, this task is also similar to word sense disambiguation except for the size of search space [Uramoto, 1996].

the 5 digit class categories[4]. For the cluster-based approach, the nouns were also categorized into the intermediate class categories, that is, the 2 to 4 digit class categories. Since we use the BGH hierarchy structure instead of constructing a cluster hierarchy from scratch, in a strict sense, this does not coincide with the cluster-based approach as described in the previous section. However, searching through the BGH tree structure in a top down manner still enables us to save greatly on computational resources.

A simple top down search, in which the cluster with the highest probability is followed at each level, allows only one path leading to a single leaf (5 digit class code). In order to take into account multiple word senses, we followed several paths at the same time. More precisely, the difference between the probability of each cluster and the highest probability value for that level was calculated, and clusters for which the difference was within a certain threshold were left as candidate paths. The threshold was set to 0.2 in this experiments.

The performance of each approach was evaluated on the basis of the number of correctly assigned class codes. Tables 2 to 4 show the results of each approach. Columns show the maximum number of class codes assigned to each target word. For example, the column "10" means that a target word is assigned to up to 10 class codes. If the correct class code is contained in these assigned codes, the test case is considered to be assigned the correct code. Rows show the distribution word numbers on the basis of occurrence frequencies in the training data. Each value in the table is the number of correct cases with its percentage in the parentheses.

Table 2 Results for the k-nn approach

freq\k	5	10	20	30	total
~ 10	1,733	2,581	3,934	4,902	12,719
	(13.6)	(20.3)	(30.9)	(38.5)	
10 ~	1,817	2,638	3,594	4,231	7,550
100	(24.1)	(34.9)	(47.6)	(56.0)	
100 ~	658	949	1,260	1,455	2,208
500	(29.8)	(43.0)	(57.1)	(65.9)	
500 ~	132	199	254	300	401
1000	(32.9)	(49.6)	(63.3)	(74.8)	
1000 ~	149	187	236	264	345
	(43.2)	(54.2)	(68.4)	(76.5)	
total	4,489	6,554	9,278	11,152	23,223
	(19.3)	(28.2)	(40.0)	(48.0)	

[4]Note that we ignore lower digits, and therefore, *leaf* means the categories formed by 5 digit class code.

Table 3 Results for the category-based approach

freq\k	5	10	20	30	total
~ 10	2,304	3,442	4,778	5,689	12,719
	(18.1)	(27.1)	(37.6)	(44.7)	
10 ~	2,527	3,458	4,449	5,025	7,550
100	(33.5)	(45.8)	(58.9)	(66.6)	
100 ~	922	1,231	1,511	1,657	2,208
500	(41.8)	(55 8)	(68.4)	(75.0)	
500 ~	204	250	298	327	401
1000	(50.9)	(62.3)	(74.3)	(81.5)	
1000 ~	181	231	264	289	345
	(52.5)	(67.0)	(76.5)	(83.8)	
total	6,138	8,612	11,300	12,987	23,223
	(26.4)	(37.1)	(48.7)	(55.9)	

Table 4 Results for the cluster-based approach

freq\k	5	10	20	30	total
~ 10	1,982	2,534	3,026	3,240	12,719
	(15.6)	(19.9)	(23.8)	(25.5)	
10 ~	2,385	3,011	3,490	3,690	7,550
100	(31.6)	(39.9)	(46.2)	(48.9)	
100 ~	887	1,077	1,205	1,264	2,208
500	(40.2)	(48.8)	(54.6)	(57.2)	
500 ~	201	227	251	259	401
1000	(50.1)	(56.6)	(62.6)	(64.6)	
1000 ~	183	209	231	239	345
	(53.0)	(60.6)	(67.0)	(69.3)	
total	5,638	7,058	8,203	8,692	23,223
	(24.3)	(30.4)	(35.3)	(37.4)	

6 Discussion

Overall, the category-based approach shows the best performance, followed by the cluster-based approach. k-nn shows the worst performance. This result contradicts past research [Iwayama and Tokunaga, 1995; Masand *et al.*, 1992]. One possible explanation for this contradiction may be that the basis of the classification for BGH and our probabilistic model is very different. In other words, co-occurrences with verbs may not have captured the classification basis of BGH very well.

The performance of k-nn is noticeably worse than that of the others for low frequent words. This may be due to data sparseness. Generalizing individual nouns by constructing clusters remedies this problem.

When k is small, namely only categories with high probabilities are assigned, the category-based and cluster-based approaches show comparable performance. When k becomes bigger, however, the category-based approach becomes superior. Since a beam search was adopted for the cluster-based approach, there was a possibility of failing to follow the correct path.

7 Related work

The goal of this paper is the same as that for Uramoto [Uramoto, 1996], that is, identifying appropriate word classes for an unknown word in terms of an existing thesaurus. The significant difference between Uramoto and our research can be summarized as follows.

- The core thesaurus is different. Uramoto used ISAMAP [Tanaka and Nisina, 1987], which contains about 4,000 words.
- We adopted a probabilistic model, which has a sounder foundation than the Uramoto's. He used several factors, such as similarity between a target word and words in each classes, class levels and so forth. These factors are combined into a score by calculating their weighted sum. The weight for each factor is determined by using held out data.
- We restricted our co-occurrence data to that included the "*WO*" postposition, which typically marks the *accusative* case, while Uramoto used several grammatical relations in tandem. There are claims that words behave differently depending on their grammatical role, and that they should therefore be classified into different word classes when the role is different [Tokunaga *et al.*, 1995]. This viewpoint should be taken into account when we construct a thesaurus from scratch. In our case, however, since we assume a core thesaurus, there is room for argument as to whether we should consider this claim. Further investigation on this point is needed.
- Our evaluation scheme is more rigid and based on a larger dataset. We conducted cross validation on nouns appearing in BGH and the judgement of correctness was done automatically, while Uramoto used unknown words as test cases and decided the correctness on a subjective basis. The number of his test cases was 250, ours is 23223. The performance of his method was reported to be from 65% to 85% in accuracy, which seems better than ours. However, it is difficult to compare these two in an absolute sense, because both the evaluation data and code assignment scheme are different. We identified class codes at the fifth level of BGH, while Uramoto searched for a set of class codes at various levels.

Nakano proposed a method of assigning a BGH class code to new words [Nakano, 1981]. His approach is very different from ours and Uramoto's. He utilized characteristics of Japanese character classes. There are three character classes used in writing Japanese, *Kanzi*, *Hiragana* and *Katakana*. A Kanzi character is an ideogram and has a distinct stand-alone meaning, to a certain extent. On the other hand, Hiragana and Katakana characters are phonograms. Nakano first constructed a Kanzi meaning dictionary from BGH by extracting words including a single Kanzi character. He defined the class code of each Kanzi character to the code of words including only that Kanzi. He then assigned class codes to new words based on this Kanzi meaning dictionary. For example, if the class codes of Kanzi K_i and K_j are $\{c_{i1}, c_{i2}\}$ and $\{c_{j1}, c_{j2}, c_{j3}\}$ respectively, then a word including K_i and K_j is assigned the codes $\{c_{i1}, c_{i2}, c_{j1}, c_{j2}, c_{j3}\}$. We applied Nakano's method on the data used in section 5[5], obtaining the accuracy of 54.6% for 17,736 words. The average number of codes assigned was 5.75. His method has several advantages over ours, such as:

- no co-occurrence data is required,
- not so much computational overhead is required.

However, there are obvious limitations, such as:

- it can not handle words not including Kanzi,
- ranking or preference of assigned codes is not obtained,
- not applicable to languages other than Japanese.

We investigated the overlap of words that were assigned correct classes for our category-based method and Nakano's method. The parameter k was set to 30 for our method. The number of words that were assigned correct classes by both methods was 5,995, which represents 46% of the words correctly classified by our method and 62% of the words correctly classified by Nakano's method. In other words, of the words correctly classified by one method, only about half can also be also classified correctly by the other method. This result suggests that these two methods are complementary to each other, rather than competitive, and that the overall performance can be improved by combining them.

8 Conclusion

This paper proposed a method for extending an existing thesaurus by classifying new words in terms of that thesaurus. We conducted experiments using the Japanese *Bunruigoihyô* thesaurus and about 420,000 co-occurrence pairs of verbs and nouns, related by the *WO* postposition. Our experiments showed that new words can be classified correctly with a maximum accuracy of more than 80% when the category-based search strategy was used.

We only used co-occurrence data including the *WO* relation (*accusative* case). However, as mentioned in comparison with Uramoto's work, the use of other relations should be investigated.

This paper focused on only 5 digit class codes. This is mainly because of the data sparseness of co-occurrence data. We would be able to classify words at deeper levels if we obtained more co-occurrence data. Another approach would be to construct a hierarchy from a set of words of each class, using a clustering algorithm.

[5]Nakano's original work used an old version of BGH, which contains 36,263 words.

References

[Basili *et al.*, 1992] Basili, R., Pazienza, M., and Velardi, P. Computational lexicons: The neat examples and the odd exemplars. In *Proceedings of third conference on Applied Natural Language Processing*, pp. 96–103.

[Chapman, 1984] Chapman, L. R. *Roget's International Thesaurus (Fourth Edition)*. Harper & Row.

[Charniak, 1993] Charniak, E. *Statistical Language Learning*. MIT Press.

[Hatzivassiloglou and McKeown, 1993] Hatzivassiloglou, V., and McKeown, K. R. Towards the automatic identification of adjectival scales: Clustering adjectives according to meaning. In *Proceedings of 31st Annual Meeting of the Association for Computational Linguistics*, pp. 172–182.

[Hayashi, 1966] Hayashi, O. *Bunruigoihyô*. Syueisyuppan. (In Japanese).

[Hindle, 1990] Hindle, D. Noun classification from predicate-argument structures. In *Proceedings of 28th Annual Meeting of the Association for Computational Linguistics*, pp. 268–275.

[Hirschman *et al.*, 1975] Hirschman, L., Grishman, R., and Sager, N. Grammatically-based automatic word class formation. *Information Processing & Management, 11*, 39–57.

[Iwayama and Tokunaga, 1994] Iwayama, M., and Tokunaga, T. A probabilistic model for text categorization: Based on a single random variable with multiple values. In *Proceedings of 4th Conference on Applied Natural Language Processing*.

[Iwayama and Tokunaga, 1995] Iwayama, M., and Tokunaga, T. Cluster-based text categorization: A comparison of category search strategies. In *Proceedings of ACM SIGIR'95*, pp. 273–280.

[Masand *et al.* 1992] Masand, B., Linoff, G., and Waltz, D. Classifying news stories using memory based reasoning. In *Proceedings of ACM SIGIR '92*, pp. 59–65.

[Miller *et al.*, 1993] Miller, G. A., Bechwith, R., Fellbaum, C., Gross, D., Miller, K., and Tengi, R. Five Papers on WordNet. Tech. rep. CSL Report 43, Cognitive Science Laboratory, Princeton University. Revised version.

[Nakano, 1981] Nakano, H. Word classification support system. *IPSJ-SIGCL, 25*.

[Pereira *et al.*, 1993] Pereira, F., Tishby, N., and Lee, L. Distributional clustering of English words. In *Proceedings of 31st Annual Meeting of the Association for Computational Linguistics*, pp. 183–190.

[Real World Computing Partnership, 1995] Real World Computing Partnership. RWC text database. `http://www.rwcp.or.jp/wswg.html`.

[Resnik, 1992] Resnik, P. A class-based approach to lexical discovery. In *Proceedings of 30th Annual Meeting of the Association for Computational Linguistics*, pp. 327–329.

[Mainichi Shimbun, 1991-1994] Mainichi Shimbun CD-ROM '91-'94.

[Tanaka and Nisina, 1987] Construction of a thesaurus based on superordinate/subordinate relations. *IPSJ-SIGNL, NL64-4*, 25–44. (In Japanese).

[Tokunaga *et al.*, 1995] Tokunaga, T., Iwayama, M., and Tanaka, H. Automatic thesaurus construction based on grammatical relations. In *Proceedings of IJCAI '95*, pp. 1308–1313.

[Uramoto, 1996] Uramoto, N. Positioning unknown words in a thesaurus by using information extracted from a corpus. In *Proceedings of COLING '96*, pp. 956–961.

[Ushioda, 1996] Ushioda, A. Hierarchical clustering of words. In *Proceedings of COLING '96*, pp. 1159–1162.

[Weiss and Kulikowski, 1991] Weiss, S. M., and Kulikowski, C. *Computer Systems That Learn*. Morgan Kaufmann.

[Yarowsky, 1992] Yarowsky, D. Word-sense disambiguation using statistical models of Roget's categories trained on large corpora. In *Proceedings of COLING '92*, Vol. 2, pp. 454–460.