

Automatic Thesaurus Construction based on Grammatical Relations

Tokunaga Takenobu
Dept. of Computer Science
Tokyo Institute of Technology
2-12-1 Ôokayama Meguro
Tokyo 152 Japan

Iwayama Makoto
Advanced Research Lab.
Hitachi Ltd.
2520 Akanuma Hatoyama
Hiki 350-03 Japan

Tanaka Hozumi
Dept. of Computer Science
Tokyo Institute of Technology
2-12-1 Ôokayama Meguro
Tokyo 152 Japan

Abstract

We propose a method to build thesauri on the basis of grammatical relations. The proposed method constructs thesauri by using a hierarchical clustering algorithm. An important point in this paper is the claim that thesauri in order to be efficient need to take (surface) case information into account. We refer to the thesauri as "*relation-based thesaurus* (RBT)." In the experiment, four RBTs of Japanese nouns were constructed from 26,023 verb-noun co-occurrences, and each RBT was evaluated by objective criteria. The experiment has shown that the RBTs have better properties for selectional restriction of case frames than conventional ones.

1 Introduction

For most natural language processing (NLP) systems, thesauri are one of the basic ingredients. In particular, coupled with case frames, they are useful to guide correct analysis [Allen, 1988]. In the example-based frameworks, thesauri are also used to compensate for insufficient example data [Sato and Nagao, 1990; Nagao and Kurohashi, 1992]. *Roget's International Thesaurus* [Chapman, 1984], *Bunruigoihyô* [Hayashi, 1966] and *WordNet* [Miller *et al.*, 1993] are typical thesauri which have been used in the past NLP research. All of them are handcrafted, machine-readable and have fairly broad coverage. However, since these thesauri are originally compiled for human use, they are not always suitable for natural language processing by computers. Their classification of words is sometimes too coarse and does not provide sufficient distinctions between words.

One of the reasons for this is that these thesauri aim for broad coverage, rather than for dealing with a particular domain. Experience has shown that restricting the target domain appropriately is the key to building successful NLP systems. This fact has been discussed by researchers working on "sublanguage" [Grishman and Sterling, 1992; Sekine *et al.*, 1992] or "register" [Halliday and Hassan, 1985; Biber, 1993]. Another problem with handcrafted thesauri is the fact that their classification is based on the intuition of lexicographers, with their

criteria of classification not being always clear. Furthermore, crafting thesauri by hand is very expensive even in restricted domains.

Therefore, building thesauri automatically from corpora has received a large attention in recent years [Hirschman *et al.*, 1975; Hindle, 1990; Hatzivassiloglou and McKeown, 1993; Pereira *et al.*, 1993]. These attempts basically take the following steps [Charniak, 1993].

- (1) extract co-occurrences
- (2) define similarities (distance) between words on the basis of co-occurrence data
- (3) cluster words on the basis of similarity

At each step, we have several options. In this paper, we will focus on step (1): the properties of co-occurrences. As for step (2) and (3), we will use the method proposed by Iwayama and Tokunaga [Iwayama and Tokunaga, 1995], which is briefly described in section 3.

Co-occurrences are usually gathered on the basis of some relations such as predicate-argument, modifier-modified, adjacency, or mixture of them. For example, Hindle used verb-subject and verb-object relations to classify nouns [Hindle, 1990]. Hirschman *et al.* also used verb-subject and verb-object relations, as well as prepositions and adjective-noun relations [Hirschman *et al.*, 1975]. Hatzivassiloglou and McKeown suggested to use as many relations as possible in order to classify adjectives [Hatzivassiloglou and McKeown, 1993].

All these attempts assume a distribution hypothesis, that is, words appearing in a similar context are similar, hence they should be classified into the same class [Grishman *et al.*, 1986; Hindle, 1990]. As far as we concerned, we consider co-occurrences of words as a kind of context. The more specific the context is, the more precise our classification will be. In this respect, we should use as specific relations as possible in order to obtain better thesauri. Unlike previous research on this topic, we suggest to build a thesaurus for each grammatical relation. In particular, we will use surface cases. Therefore we would have a thesaurus for each surface case. This is what we call "*relation-based thesaurus* (RBT)."

Another aspect that seems to be lacking in the past research is an objective evaluation of the automatically built thesauri. All the previous attempts except [Pereira

et al., 1993] evaluate their results on the basis of subjective criteria: to what extent is the result consistent with human intuition. In this paper, we propose an objective evaluation method for automatically built thesauri.

In the following, we will introduce relation-based thesauri (section 2) and describe the clustering algorithm (section 3). Section 4 describes an experiment in which we compared with relation-based thesauri to conventional ones. Finally, section 5 concludes the paper and gives some future research directions.

2 Relation-based thesauri

This paper focuses on building thesauri of nouns based on verb-noun relations. Following the research mentioned in the previous section, co-occurrence data is represented by tuples as shown in the left column of figure 1, where n_i and v_j denote nouns and verbs respectively, while r_k denotes grammatical relations such as *subject*, *object* and so forth.

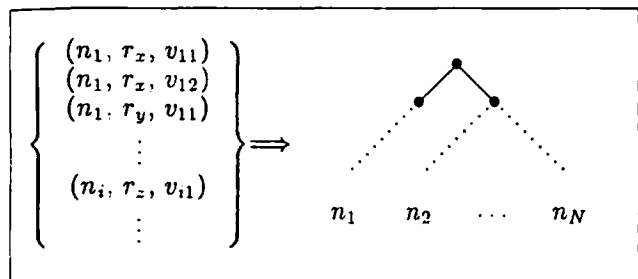


Fig. 1 Thesaurus construction from tuples

Past research has not focused on using grammatical relations (r_k). For example, Hindle used *subject* and *object* relations but did not distinguish between them when calculating the distance between nouns [Hindle, 1990]. Hirschman et al. used other grammatical relations than *subject* and *object* in order to build word classes. Actually they used various relations simultaneously [Hirschman et al., 1975]. On the other hand, Pereira et al. used only the *object* relation [Pereira et al., 1993]. Unlike all these attempts, we will focus on difference of relations and propose to build a thesaurus for each relation. This approach is based on the fact that a noun behaves differently depending on its grammatical role. Take the following examples.

- John studied English at the university.
- Mary worked till late at her office.
- The university stated that they would raise the tuition fee.
- The mayor stated that he would raise taxes.

With regard to taking a *locative* role (derived from "at" phrase in (a) and (b)), "university" and "office" behave similarly, hence they would be classified into the same word class. On the other hand, with regard to being a *subject* of verb "state" (in (c) and (d)), "university" behaves like "mayor." With this respect, "university" and "mayor" would be classified into the same class. It should be noted that the transitivity does not always hold beyond the relations. In the above example, it is questionable if we could classify "office" and "mayor"

into the same class. The bases of the similarity between "university" and "office", and that between "university" and "mayor" are different.

In conventional thesauri, "university" and "mayor" would be placed in the different classes: "university" would be some kind of ORGANIZATION and "mayor" some kind of HUMAN. However, they could be put in the same class, namely as being a subject of a certain set of verbs.

Figure 2 shows our approach, while figure 1 illustrates the conventional ones. The tuples are divided into the subsets with respect to their *relation*. A thesaurus is built from each set of these tuples.

3 Hierarchical Bayesian Clustering

We adopt a hierarchical clustering algorithm that attempts to maximize the Bayesian posterior probability at each step of merge. This algorithm has been introduced by Iwayama and Tokunaga [Iwayama and Tokunaga, 1995] and is referred to as *Hierarchical Bayesian Clustering* (HBC). In this section, we briefly review the outline of the algorithm.

Given a set of training data D , HBC constructs the set of clusters C that has the locally maximum value of the posterior probability $P(C|D)$. This is a general form of the well-known Maximum Likelihood estimation, estimating the most likely model (i.e., set of clusters) for a given set of training data.

Input:

$D = \{d_1, d_2, \dots, d_N\}$: a collection of N data;

Initialize:

$C_0 = \{c_1, c_2, \dots, c_N\}$: a set of clusters;

$c_i = \{d_i\}$ for $1 \leq i \leq N$

calculate $SC(c_i)$ for $1 \leq i \leq N$

calculate $SC(c_i \cup c_j)$ for $1 \leq i < j \leq N$

for $k = 1$ to $N - 1$ do

$(c_x, c_y) = \arg \max_{c_x, c_y} \frac{SC(c_x \cup c_y)}{SC(c_x)SC(c_y)}$

$C_k = C_{k-1} - \{c_x, c_y\} + \{c_x \cup c_y\}$

calculate $SC(c_x \cup c_z)$ for all $c_z \in C_k$ where $z \neq x$

Function $SC(c)$

return $\prod_{d \in c} P(d|c)$

Fig. 3 Hierarchical Bayesian Clustering

Like most agglomerative clustering algorithms [Cormack, 1971; Anderberg, 1973; Griffiths et al., 1984; Willett, 1988], HBC constructs a cluster hierarchy (also called "dendrogram") from bottom up by merging two clusters at a time. At the beginning (the bottom level in a dendrogram), each datum belongs to a cluster whose only member is the datum itself. For every pair of clusters, HBC calculates the posterior probability after merging the pair, selecting the pair with the highest probability. To see the details of this merge process, consider a merge step $k+1$ ($0 \leq k \leq N-1$). By the step $k+$

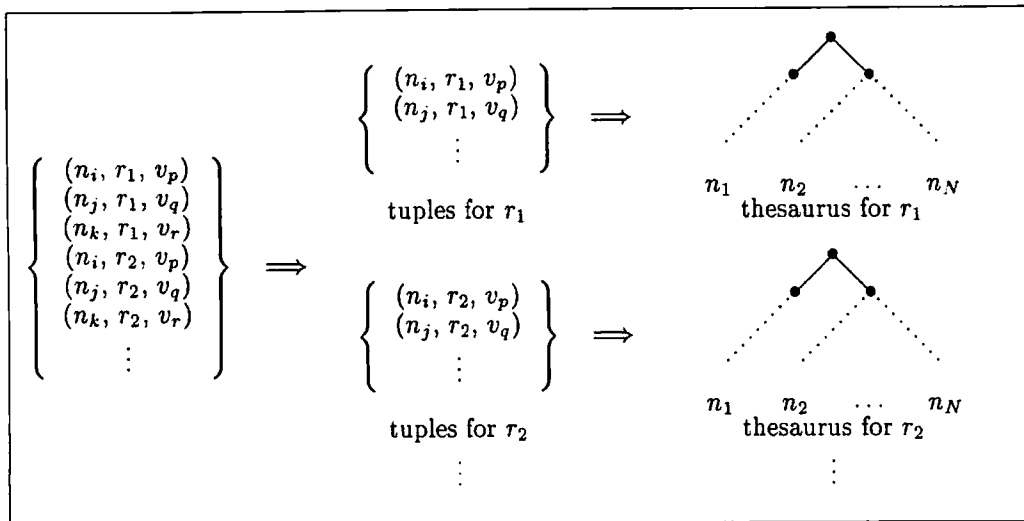


Fig. 2 Case-based thesauri construction

1, a data collection $D = \{d_1, d_2, \dots, d_N\}$ has been partitioned into a set of clusters $C_k = \{c_1, c_2, \dots, c_{N-k}\}$. That is, each datum $d_i \in D$ belongs to a cluster $c_j \in C_k$ and the clusters being mutually exclusive. The overall posterior probability at this point becomes

$$\begin{aligned}
 P(C_k|D) &= \prod_{c \in C_k} \prod_{d \in c} P(c|d) \\
 &= \prod_{c \in C_k} \prod_{d \in c} \frac{P(d|c)P(c)}{P(d)} \\
 &= \frac{\prod_{c \in C_k} P(c)^{|c|}}{P(D)} \prod_{c \in C_k} \prod_{d \in c} P(d|c) \\
 &= \frac{PC(C_k)}{P(D)} \prod_{c \in C_k} SC(c) \quad (1)
 \end{aligned}$$

Here $PC(C_k)$ corresponds to the prior probability that N random data are classified into a set of clusters C_k . This probability is defined as follows:

$$PC(C_k) = \prod_{c \in C_k} P(c)^{|c|}. \quad (2)$$

$SC(c)$ defines the probability that all data in a cluster c are produced from the cluster and is defined as

$$SC(c) = \prod_{d \in c} P(d|c). \quad (3)$$

When the algorithm would merge two clusters $c_x, c_y \in C_k$, the set of clusters C_k is updated as follows:

$$C_{k+1} = C_k - \{c_x, c_y\} + \{c_x \cup c_y\}. \quad (4)$$

After the merge, the posterior probability is inductively updated as follows:

$$P(C_{k+1}|D) = \frac{PC(C_{k+1})}{PC(C_k)} \frac{SC(c_x \cup c_y)}{SC(c_x)SC(c_y)} P(C_k|D). \quad (5)$$

Note that this updating is local and can be done efficiently, because all we have to recalculate since the previous step is the probability for the merged new cluster, that is, $SC(c_x \cup c_y)$. The factor of $\frac{PC(C_{k+1})}{PC(C_k)}$ can be neglected for maximization of $P(C|D)$, since the factor would reduce to a constant regardless of the merged pair. See [Iwayama and Tokunaga, 1995] for further discussion.

For a collection of N data, merge takes place $N - 1$ times, and the last merge produces a single cluster containing the entire set of data. Figure 3 shows the HBC algorithm.

Our current concern is clustering nouns based on the relations they have with verbs. In order to apply HBC to clustering nouns, we need to calculate the elemental probability $P(d|c)$ that a cluster c actually contains its member noun d . We follow Iwayama and Tokunaga [Iwayama and Tokunaga, 1994] in order to calculate this probability.

[Iwayama and Tokunaga, 1994] discusses clustering of documents where each document is represented as a set of terms. In our case, we make clusters of nouns, each one of them being represented as a set of verbs co-occurring with this particular noun. A cluster c being a set of nouns, c is also represented as a set of verbs that all the members of c co-occur with. Consider an event $V = v$ where a randomly extracted verb V from a set of verbs is equal to v . Conditioning $P(d|c)$ on each possible event gives

$$P(d|c) = \sum_v P(d|c, V = v)P(V = v|c). \quad (6)$$

If we assume conditional independence between c and d given $V = v$, we obtain

$$P(d|c) = \sum_v P(d|V = v)P(V = v|c). \quad (7)$$

Using Bayes' theorem, this becomes

$$P(d|c) = P(d) \sum_v \frac{P(V = v|d)P(V = v|c)}{P(V = v)}. \quad (8)$$

Since each $P(d)$ appears in every estimation of $P(C|D)$ only once, this can be excluded for maximization purpose. Other probabilities, $P(V = v|d)$, $P(V = v|c)$, and $P(V = v)$ are estimated from given data by using the simplest estimation as below.

- $P(V = v|d)$: relative frequency of a verb v co-occurring with a noun d
- $P(V = v|c)$: relative frequency of a verb v co-occurring with nouns in cluster c
- $P(V = v)$: relative frequency of a verb v appearing in the whole training data

4 Evaluation

This section describes an experiment to evaluate RBTs compared with a thesaurus constructed without considering grammatical relations.

4.1 Data and preprocessing

The data we used for evaluation is a subset of the EDR collocation dictionary of Japanese [EDR, 1994]. This dictionary contains 1,159,144 tuples of words with various relations. The tuples are extracted from newspaper articles and magazines. The words in the tuples are tagged with concept identifiers, which are the pointers to the EDR concept dictionary. This dictionary describes thus collocations of word senses. This is a nice feature for clustering words, because we can avoid the problems caused by polysemy [Fukumoto and Tsujii, 1994].

From the dictionary, we extracted the tuples that fulfilled the following three conditions:

- describing verb and noun relations
- the surface case of the nouns are either "ga" (*nom*), "wo" (*acc*), "ni" (*dat/loc*) or "de" (*inst/loc*)¹
- both verb and noun are tagged with concept identifiers, that is, words are semantically disambiguated

We excluded the tuples in which the surface cases changed because of the passive or causative constructions. As a result, we obtained 199,574 tuples. Due to the scarceness of the data and the limitation of our computational resources, we chose 100 nouns on the basis of their frequencies and used only those tuples containing them. These 100 nouns were used for clustering. Table 1 shows the number of tuples which include these 100 nouns for each surface case.

Table 1 Number of tuples

surface case	No. of tuples
ga	5,993
wo	9,810
ni	6,441
de	3,779
total	26,023

¹In this paper we deal only with these four relations.

We conducted 2-fold cross validation with this data, namely, one half of the data was used as training set for building clusters and the other half was held out as test data, and vice versa. Since we are considering these four surface cases, we built four RBTs from the training data, and one thesaurus from all the training data without taking into account surface cases. We refer to the last one as "relation-neglected thesaurus (RNT)." The RNT is the baseline of the RBTs.

4.2 Evaluation method

The thesauri are evaluated by the following procedure. For a each verb in the test data, a set of nouns that co-occur with the verb is associated with the verb. This set of nouns is referred to as "answer set" of the verb. We use a threshold of the number of nouns in an answer set. Only the verbs that have more nouns than the threshold in their answer set are used as test cases. In the experiment, the threshold was set to 10. The number "10" does not have any special meaning, it is simply chosen as a compromise between accuracy and reliability of the evaluation. Greater threshold decreases the number of test cases, therefore it degrades the reliability of the evaluation. On the other hand, lower threshold spuriously decreases the accuracy of each test case.

Each verb has an answer set for each surface case, thus we have four test set of verbs corresponding to each surface case. As we can see from the algorithm described in section 3, the HBC algorithm generates a binary tree (dendrogram), in which each leaf is a noun. We traverse the tree from top to bottom for each verb in the test data, and at each node we calculate recall and precision from the answer set of the verb and the set of nouns under the current node. We have an option at each non-terminal node. The child node which dominates more nouns that are in the answer set is chosen. Figure 4 is an example of such a tree traversal.

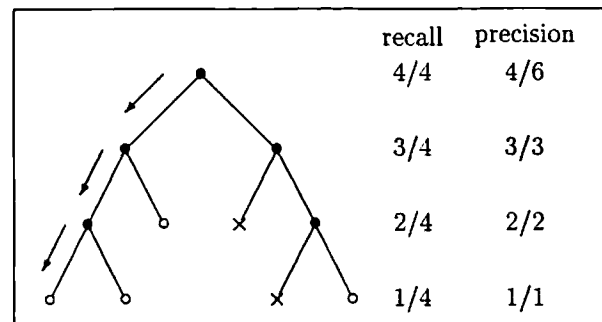


Fig. 4 Traverse of the thesaurus

In figure 4, "o" denotes a noun that is included in the answer set of the verb, while "x" denotes a noun that is not. We call the former *correct noun* and the latter *incorrect noun*. Recall and precision at each node are calculated as follows.

$$Recall = \frac{\text{number of correct nouns under the current node}}{\text{number of nouns in the answer set of the verb}} \quad (9)$$

$$Precision = \frac{\text{number of correct nouns under the current node}}{\text{number of the nouns under the current node}} \quad (10)$$

In the above example, the answer set of the verb includes four (correct) nouns. Recall and precision of this verb are calculated as shown in the right column of figure 4. As we move down the tree, the recall decreases monotonically, since the number of the nouns dominated by the current node decrease. On the other hand, the precision increases as we move down the tree. If we aggregate the nouns having a similar tendency to co-occur with verbs, the recall will remain at the high level. Therefore, we can evaluate the quality of the thesaurus in terms of the recall-precision curve. For example, suppose we use the thesaurus for the constraints of selectional restriction. For this purpose, we also need case frames of verbs in which a node, or a set of nodes of the thesaurus is described as the case fillers². If the thesaurus has the desirable property described above, the number of nodes to be described as a case filler would decrease. This is precisely what we want. Because, unlike the example-based framework, one of the motivations of using thesaurus is to minimize the description of knowledge. In the example-based framework, all the individual words that co-occur with a verb would be described as case fillers [Kurohashi and Nagao, 1993].

4.3 Result and discussion

For all combinations of the four test sets corresponding to each surface case and the five thesauri (four RBTs and one RNT), the recall-precision curves were calculated. As mentioned before, recall and precision have mutual exclusive properties. In order to summarize their balance, we used a *breakeven point* which is defined as the point at which the recall and precision become equal on a recall-precision curve [Lewis, 1992]. The greater breakeven value means the better the recall-precision curves. For each test case, the breakeven point was calculated by linear interpolation, and for each combination of the test set and the thesaurus, the mean average of breakeven points was also calculated. Table 2 summarizes the mean breakeven points of every combination.

Table 2 Breakeven point [%]

	Test set			
	<i>ga</i>	<i>wo</i>	<i>ni</i>	<i>de</i>
RBT/ <i>ga</i>	37.45	30.88	31.55	28.56
RBT/ <i>wo</i>	33.98	36.79	32.23	29.91
RBT/ <i>ni</i>	31.47	30.96	37.40	33.82
RBT/ <i>de</i>	29.51	28.19	31.53	38.06
RNT	35.38	35.04	36.04	31.67

Table 2 shows that for all surface cases, the RBT marks the best breakeven value with the test set of the

² Assigning thesaurus nodes to a case filler is also an important issue and several attempts have been made [Grishman *et al.*, 1986; Grishman and Sterling, 1992]. The automatic method for acquiring case frames should be discussed together with the automatic thesaurus construction. However, this issue is beyond the scope of this paper. We are currently working on a paper that deals with this problem.

corresponding surface case: the diagonal values in the table are the best in the columns. They are also superior to RNTs. This result supports our claim that we would be able to obtain better thesauri by considering surface cases.

The breakeven values in the table are all very poor in the absolute sense. The main reason for this is that we derived the answer set only from the co-occurrence data. There might be many nouns that would actually be a case filler of a verb but do not belong to the answer set of the verb. In order to solve this problem, we need to check manually which noun can really be the case filler of the verb for all nouns in the thesaurus. However, this is time consuming and introduces subjective criteria, therefore we used only the observed data. Thus the values in table 2 should be interpreted in the relative sense not in the absolute one.

As for the surface cases "*wo*" and "*ni*", the difference between RBT and RNT is not really significant. The reason for this is that for these two surface cases, the distribution of noun frequencies in the tuples for RBT is very similar to that for RNT. In other words, many frequently occurring nouns in the tuples of these two surface cases do not appear in the tuples of other surface cases. Note that the tuples used for creating each RBT is a proper subset of the tuples used for creating the RNT. We would not suffer from this problem, if more data were available, or we chose the target nouns based on the frequency in the tuples of each surface case. In the latter case, however, we would be able to compare a RBT to the RNT, but not to the other RBTs. Because the set of nouns to be clustered would be different for each surface case.

5 Concluding remarks

We have proposed to build thesauri on the basis of grammatical relation. We have conducted a preliminary experiment with 26,023 tuples of verb, noun and surface cases of Japanese. The results are quite promising. We have also proposed a method that allows to evaluate thesauri objectively.

We started from the assumption that surface cases are independent from each other. However, such an assumption is questionable. We also need to evaluate RBTs in the context of real world settings, such as parsing [Grishman *et al.*, 1986]. For this purpose, we need case frames whose case fillers are described in terms of the RBT nodes. We should explore methods that can automatically acquire case frames [Grishman *et al.*, 1986; Grishman and Sterling, 1992] as well.

Furthermore, we have used EDR collocation dictionary, in which the words are already semantically disambiguated. Obviously we can not expect to find such pure data if we work on large scale. Last but not least we have to evaluate the quality of RBT that are built from raw data (text).

References

- [ACL, 1993] ACL '93. *Proceedings of the 31st Annual Meeting of the Association for Computational Linguistics*, 1993.
- [Allen, 1988] J. Allen. *Natural Language Understanding*. The Benjamin/Cummings Publishing Company, Inc., 1988.
- [Anderberg, 1973] M. R. Anderberg. *Cluster Analysis for Applications*. Academic Press, 1973.
- [Biber, 1993] D. Biber. Using register-diversified corpora for general language studies. *Computational Linguistics*, 19(2):219-241, 6 1993.
- [Chapman, 1984] L. R. Chapman. *Roget's International Thesaurus (Fourth Edition)*. Harper & Row, 1984.
- [Charniak, 1993] E. Charniak. *Statistical Language Learning*. the MIT Press, 1993.
- [COL, 1992] COLING '92. *Proceedings of the 14th International Conference on Computational Linguistics*, 1992.
- [Cormack, 1971] R. M. Cormack. A review of classification. *Journal of the Royal Statistical Society*, 134:321-367, 1971.
- [EDR, 1994] EDR. Collocation dictionary. Technical Report TR-043, Japan Electronic Dictionary Research Institute, 1994.
- [Fukumoto and Tsujii, 1994] F. Fukumoto and J. Tsujii. Automatic recognition of verbal polysemy. In *Proceedings of the 14th International Conference on Computational Linguistics*, volume 2, pages 762-768. COLING '94, 1994.
- [Griffiths et al., 1984] A. Griffiths, L. A. Robinson, and P. Willett. Hierarchic agglomerative clustering methods for automatic document classification. *Journal of Documentation*, 40(3):175-205, 1984.
- [Grishman and Sterling, 1992] R. Grishman and J. Sterling. Acquisition of selection patterns. In *Proceedings of the 14th International Conference on Computational Linguistics* [1992], pages 658-664.
- [Grishman et al., 1986] R. Grishman, L. Hirschman, and N. T. Nhan. Discovery procedures for sublanguage selectional patterns: Initial experiments. *Computational Linguistics*, 12(3):205-215, 7 1986.
- [Halliday and Hassan, 1985] M. A. K. Halliday and R. Hassan. *Language, context, and text: Aspects of language in a social-semiotic perspective*. Deakin University Press, 1985.
- [Hatzivassiloglou and McKeown, 1993] V. Hatzivassiloglou and K. R. McKeown. Towards the automatic identification of adjectival according to meaning. In *Proceedings of the 31st Annual Meeting of the Association for Computational Linguistics* [1993], pages 172-182.
- [Hayashi, 1966] O. Hayashi. *Bunruigoihyô*. Syueisyuppan, 1966.
- [Hindle, 1990] D. Hindle. Noun classification from predicate-argument structures. In *Proceedings of the 28th Annual Meeting of the Association for Computational Linguistics*, pages 268-275. ACL '90, 6 1990.
- [Hirschman et al., 1975] L. Hirschman, R. Grishman, and N. Sager. Grammatically-based automatic word class formation. *Information Processing & Management*, 11:39-57, 1975.
- [Iwayama and Tokunaga, 1994] M. Iwayama and T. Tokunaga. 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 (ANLP '94)*, pages 162-167. 1994.
- [Iwayama and Tokunaga, 1995] M. Iwayama and T. Tokunaga. Hierarchical bayesian clustering for automatic text classification. In *Proceedings of the International Joint Conference on Artificial Intelligence (to appear)* 1995.
- [Kurohashi and Nagao, 1993] S. Kurohashi and M. Nagao. Structural disambiguation in Japanese by evaluating case structures based on examples in a case frame dictionary. In *Proceedings of the International Workshop on Parsing Technologies*, pages 111-122. IWPT '93, 1993.
- [Lewis, 1992] D. D. Lewis. An evaluation of phrasal and clustered representations of a text categorization task. In *Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 37-50, 1992.
- [Miller et al., 1993] G. A. Miller, R. Bechwith, C. Fellbaum, D. Gross, K. Miller, and R. Teng. Five papers on WordNet. Technical Report CSL Report 43, Cognitive Science Laboratory, Princeton University, 1993. Revised version.
- [Nagao and Kurohashi, 1992] M. Nagao and S. Kurohashi. Dynamic programming method for analyzing conjunctive structures in Japanese. In *Proceedings of the 14th International Conference on Computational Linguistics* [1992], pages 170-176.
- [Pereira et al., 1993] F. Pereira, N. Tishby, and L. Lee. Distributional clustering of english words. In *Proceedings of the 31st Annual Meeting of the Association for Computational Linguistics* [1993], pages 183-190.
- [Sato and Nagao, 1990] S. Sato and M. Nagao. Toward memory-based translation. In *Proceedings of the 13th International Conference on Computational Linguistics*, volume 3, pages 247-252. COLING '90, 1990.
- [Sekine et al., 1992] S. Sekine, J. J. Carroll, S. Ananiadou, and J. Tsujii. Automatic learning for semantic collocation. In *Proceedings of the 3rd Conference on Applied Natural Language Processing (ANLP '92)*, pages 104-110. 1992.
- [Willett, 1988] P. Willett. Recent trends in hierarchic document clustering: A critical review. *Information Processing & Management*, 24(5):577-597, 1988.