Pacific Association for Computational Linguistics

AUTOMATIC THESAURUS CONSTRUCTION USING WORD CLUSTERING

MINORU SASAKI†AND HIROYUKI SHINNOU‡
†*Department of Computer and Information Sciences,*
Faculty of Engineering, Ibaraki University,
4-12-1 Nakanarusawa, Hitachi, Ibaraki, 316-8511, JAPAN
E-mail: sasaki@cis.ibaraki.ac.jp
‡*Department of Systems Engineering,*
Faculty of Engineering, Ibaraki University,
4-12-1 Nakanarusawa, Hitachi, Ibaraki, 316-8511, JAPAN
E-mail: shinnou@dse.ibaraki.ac.jp

In this paper, we propose a new clustering algorithm for large scale document size to construct the thesaurus automatically in aid of summarization. The existing word-clustering systems use various similarity and clustering algorithm based on the context of the information retrieval. In case of the clustering using term-document matrix, the distribution of the index word represents the frequency of the word appearance in a certain contents of a document. Therefore, semantic relation between these words in the document is not so strong. As a result, the words which appear frequently in the contents tend to be gathered for one cluster. To construct a cluster set in which semantic relation between these words is contained, we show a word clustering using a pair of words with cooccurrence relation automatically. We further show that our clustering is effective for word sense disambiguation in comparison with using term-document matrix.

Key words: Word sense disambiguation, Word clustering, Thesaurus, Vector space model, Latent semantic indexing

1. INTRODUCTION

A user usually applies natural language to express a own query. Using a search engine such as Lycos and google on the Internet, the user represents queries which consists of a few words. If the user has some knowledge of the words typically used to describe a particular topic, queries can be represented exactly. However, the user does not come up with the topic words, it is difficult to represent queries with the content to search. In an information retrieval (IR) system, without considering lexical and semantic ambiguity such as paraphrase representation, documents containing the input words are retrieved.

In a document summarization (DS) system, as well as the IR system, the system typically summarizes a document based on word frequency contained in a document and the location of a word in the document. Specially, a word which occurs frequently in the document has important element of the document. In this case, if the same word appears continuously in some sentences in the document, it is easy to understand the importance the word has within the document. However, one word generally has multiple meanings and is able to be represented as the other words so it is difficult to find the topic word.

To reduce response time for retrieval operation with IR system and output a summarization result from a document with a high degree of accuracy, the use of thesaurus is a possible approach to expand and to focus the queries. A thesaurus is a compilation of words and phrases showing synonymous and hierarchical relationship and dependencies, is often used for the support of retrieval approaches[Tokunaga1999]. To apply the thesaurus in such the cases, it is necessary to construct the dictionary of the form which can be treated by the computer. Additionally, when a thesaurus is used for the support of IR and DS system, a complicated layered structure is not so required and it is possible to support these tasks with a thesaurus that has simpler layered structure. Therefore, it is necessary to construct the

thesaurus of the exclusive use for the purpose of these objects.

In this paper, we propose a new word clustering algorithm for a large scale set of words with the aim of automatic construction for word sense disambiguation. In previous research on word clustering, many algorithm based on vector space model are proposed [Kawamae et al.2001]. In this model, the distribution of an index word which appears in documents can be represented statistically as a vector. However, it is hard to express the semantic relation between the words in a document, and words common to a certain topic tend to gather in a cluster outputted as a result. In our research, to construct some clusters, each cluster has a semantic commonality, we extract word pairs which has a cooccurrence relation and express word statistics.

2. WORD CLUSTERING

2.1. Extraction of Cooccurrence Relations

To perform word clustering, word pairs with cooccurrence relation, which are extracted from a large scale set of documents, are used as features. Then, we extract two patterns, "noun + no + noun" structure and "noun + noun" structure such as compound noun. For example, we extract these Japanese phrases, "SEKAI no HEIWA (peace of the world)" and "FUKUGOU MEISHI".

In this extraction, we make some exception to extract cooccurence relation more accurately. First, when adverb and adjective words exist between the noun words, such words are ignored. Second, our experience takes the noun words into consideration so that pronoun is also ignored. Furthermore, hiragana katakana words and the words which mean days and time such as "ICHI-GATSU(JANUARY)" are also ignored and intend to apply for our system in the future.

Word-pairs are extracted using morphological analysis system "Chasen" and the patterns described above. From these word-pairs, cooccurence relation matrix is obtained through assignment of term weight statistically. Then, the problem is to use which of words in the clustering. For this problem, we consider that the qualified word is more significant than qualifier. In our research, we specify target word in the clustering as the qualified word. To express the target word numerically, the word is represented as a word vector whose elements are statistics of the qualifiers.

When each target word is represented by a vector, the elements of a word vector d are assigned two-part values $w_{ij} = L_{ij} \times G_i$. In the experiments, the factor L_{ij} is a local weight that reflects the weight of term i qualifying target word j and the factor G_i is a global weight that reflects the overall value of the target word i for the entire word pairs as follows:

$$L_{ij} = \begin{cases} 1 + \log f_{ij} & (f_{ij} > 0) \\ 0 & (f_{ij} = 0) \end{cases} \tag{1}$$

$$G_i = 1 + \sum_{j=1}^{n} \frac{\frac{f_{ij}}{F_i} \log \frac{f_{ij}}{F_i}}{\log n} \tag{2}$$

where n is the number of target words in the collection, f_{ij} is the frequency of the i-th target word modified by the j-th qualifier, and F_i is the frequency of the i-th target word throughout the entire word pairs[Chicholm and Kolda1998].

2.2. Latent Semantic Analysis of Cooccurrence Relation Matrix

Performing the process of indexing in the previous section, we obtain a cooccurrence relation matrix A from the n target words and the m qualifiers. And we suppose that the matrix A is a sparse $m \times n$ with $rank(A) = r$. The singular value decomposition(SVD) of A can be represented as follows:

$$A = U\Sigma V^T, \tag{3}$$

where $U = (u_1, \cdots, u_m)$ is an $m \times n$ orthogonal matrix and $V = (v_1, \cdots, v_n)$ is an $n \times n$ orthogonal matrix.

$$U^T U = V^T V = I_n \tag{4}$$

$\Sigma = diag(\sigma_1, \sigma_2, \cdots, \sigma_n)$ is an $n \times n$ diagonal matrix with real and non-negative numbers in descending order:

$$\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_r \geq \sigma_{r+1} = \cdots = \sigma_n = 0. \tag{5}$$

Generally, The rank of a matrix is equal to the number of non-zero singular values. We can take the singular vectors corresponding to the k largest singular values as an basis of the subspace, then we obtain a rank-k approximation A_k. This dimension reduction method is called Latent Semantic Indexing(LSI).

We use the SVD for the cooccurrence relation matrix A to compute the reduced rank approximated matrix A_k. This A_k contains effective cooccurrence information in the target word space, the target word is associated with others semantically. For example, we consider two qualifier words "management" and "economic". If these qualifier words cooccur with various target words such as "environment" frequently, there is a close relation between these qualifier words. Therefore, Even if the qualifier word "management" does not exist in a word vector which is composed of the qualifier word "economic" and some qualifier words, the weight for the word "management" increases by using the SVD.

2.3. Word Clustering Algorithm

As a method to compute the word clusters, we apply the spherical k-means algorithm [Dhillon and Modha1999], which produce disjoint clusters and unit length centroid vectors, to a high-dimensional and sparse document set. We present a summary of the Spherical k-means algorithm. For instance, the following algorithm partitions the word vectors $\mathbf{x}_1, \mathbf{x}_2, \cdots, \mathbf{x}_N$ into k clusters $\pi_1^*, \pi_2^*, \cdots, \pi_k^*$ that maximize the objective function D defined as:

$$D = \sum_{j=1}^{s} \sum_{\mathbf{x}_i \in \pi_j} \mathbf{x}_i^T \mathbf{c}_j \tag{6}$$

When the similarity of this algorithm is calculated, the similarity of two word vector $\mathbf{x}_i$ and $\mathbf{x}_j$ is represented as the following inner product w_{ij}:

$$w_{ij} = \mathbf{x}_i \cdot \mathbf{x}_j. \tag{7}$$

In the case of the cooccurence relation matrix, the similarity matrix D can be represented as $D = A^T A$. Then the Singular Value Decomposition(SVD) of the matrix A is represented as $A = U\Sigma V^T$ so the matrix D can be represented using the SVD of A as follows [Bellegarda et al.1996][Kita1999]:

$$\begin{aligned} D &= (U\Sigma V^T)^T (U\Sigma V^T) \\ &= (V\Sigma U^T)(U\Sigma V^T) \\ &= (\Sigma V^T)^T (\Sigma V^T). \end{aligned} \tag{8}$$

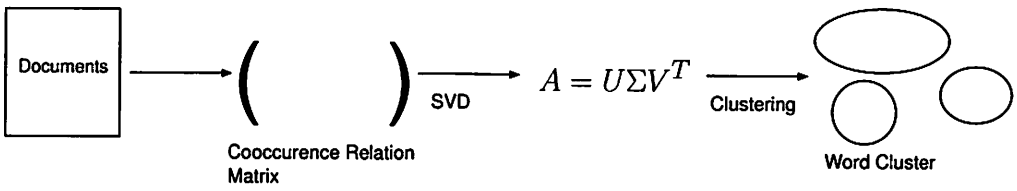

FIGURE 1. brief process of our word clustering algorithm

This expression means the similarity w_{ij} is equal to inner product of the i-th column vector and the j-th column vector in ΣV^T. If we take the k-largest singular values of Σ, we can obtain rank-k approximation of A. Therefore, this approximation brings about an effect to remove noise and reduction of calculation time for the similarity.

1. Partition the word vectors arbitrarily into the given s clusters $\{\pi_j^{(0)}\}_{j=1}^s$. Let $\{\mathbf{c}_j^{(0)}\}_{j=1}^s$ indicate the centroid vectors derived from the clusters.
2. For each word vector $\mathbf{x}_i (1 \leq i \leq N)$, find the centroid vector that maximizes the cosine similarity for $\mathbf{x}_i$. Consequently, the word vectors are partitioned again into the new clusters $\{\pi_j^{(t+1)}\}_{j=1}^s$ from the centroid vectors $\{\mathbf{c}_j^{(t)}\}_{j=1}^s$

$$\pi_j^{(t+1)} = \{\mathbf{x}_i : \mathbf{x}_i^T \mathbf{c}_j^{(t)} \geq \mathbf{x}_i^T \mathbf{c}_l^{(t)}\} \quad (1 \leq l \leq N,\ 1 \leq j \leq s) \tag{9}$$

   where $\pi_j^{(t+1)}$ is the set of vectors which are closest to the centroid vectors $\{\mathbf{c}_j^{(t)}\}_{j=1}^s$.
3. Normalize the length of the new centroid vectors

$$\mathbf{c}_j^{(t+1)} = \frac{\mathbf{m}_j^{(t+1)}}{||\mathbf{m}_j^{(t+1)}||}, \quad (1 \leq j \leq s), \tag{10}$$

   where $\mathbf{m}_j^{(t+1)}$ indicates the centroid vectors of the documents contained in the cluster $\pi_j^{(t+1)}$.
4. Calculate the value of the objective function $D^{(t+1)}$ and the difference between the $D^{(t+1)}$ and the old value of the objective function $D^{(t)}$. If this difference is not more than 1 as follows:

$$\|D^{(t)} - D^{(t+1)}\| \leq 1, \tag{11}$$

   then, set $\pi_j^* = \pi_j^{(t+1)}$ and set $\mathbf{c}_j^* = \mathbf{c}_j^{(t+1)}$ $(1 \leq j \leq s)$ and exit. Otherwise, the algorithm increments the variable t and return to step 2.

3. APPLICATION TO WORD SENSE DISAMBIGUATION

In this section, we apply word clusters obtained by our method to word sense disambiguation problems. In this experiment, we use 50 verbs applied to the Japanese dictionary task

on the SENSEVAL 2 contest[Kurohashi and Shirai2001]. We apply the naive Bayes learner and classifier to learn a training data and estimate probability of clusters from some feature sets. The precision is calculated for each verb using this probabilistic model.

3.1. Definition of feature set

We apply the feature set, which is defined in [Shinnou and Sasaki2002], to the estimation of probability. A thesaurus used in our research is not "Bunrui-Goi-Hyo"(a set of semantic principles), but word cluster obtained by word clustering. So we use the cluster number as the feature of concept instead of thesaurus ID number. In this paper, we use the following six features(e1 to e6) for the target word w:

e1: word in front of w
e2: word behind w
e3: two content words in front of w
e4: two content words behind w
e5: cluster number of the words in e3
e6: cluster number of the words in e4

From the feature sets, we use the naive Bayes to learn a training data and estimate probability of clusters. The naive Bayes classifier is the method to compute a conditional probability $P(f|c)$ of a feature f occurring given a class c. In our experiments, this probability using smoothing algorithm is estimated by the following,

$$P(f|c) = \frac{1 + \sum_{d \in D_c} N(f,d)}{|F| + \sum_{m=1}^{|F|} \sum_{d \in D_c} N(f_m,d)} \tag{12}$$

where D_c is the set of training data contained in the cluster c, d is an element of D_c, F is the set of all features, f_m is an element in F and $N(f_m,d)$ is the number of f_m contained in the instance d.

3.2. Experiments

The data used in our experiment is Mainichi newspaper articles for 1994. For these newspaper articles, we remove all tags and carry out the morphological analysis using "Cha-Sen" to extract word-pairs as shown in the previous section. For these word-pairs, we calculate a weight of the modifier statistically from the frequency to obtain cooccurrence relation matrix. It follows that we obtain 55,597 of the target words to be clustered. Calculating the cooccurrence relation matrix, we compute the SVD of this matrix to decompose to the triplet matrices. Word clustering is performed using the spherical k-means algorithm for the decomposed matrix.

Finally, we evaluate the obtained clusters using the Japanese dictionary task. Then, the number of clusters is specified as 1000, 3000, 5000 respectively. For feature sets obtained by the training data used in this task, we apply naive Bayes method to estimate the probability $P(c|f)$, where C is a set of classes and f is a feature set. If the training phase finishes, the precision is computed on this model using test data.

To evaluate the efficiency of these word clusters, we also make clusters using a term-document matrix and compute precision by the same clustering algorithm. In this experiment, the number of clusters is also 1000, 3000, 5000 respectively. Additionally, to compare with our precision, we experiment in the case of using the feature set without the cluster

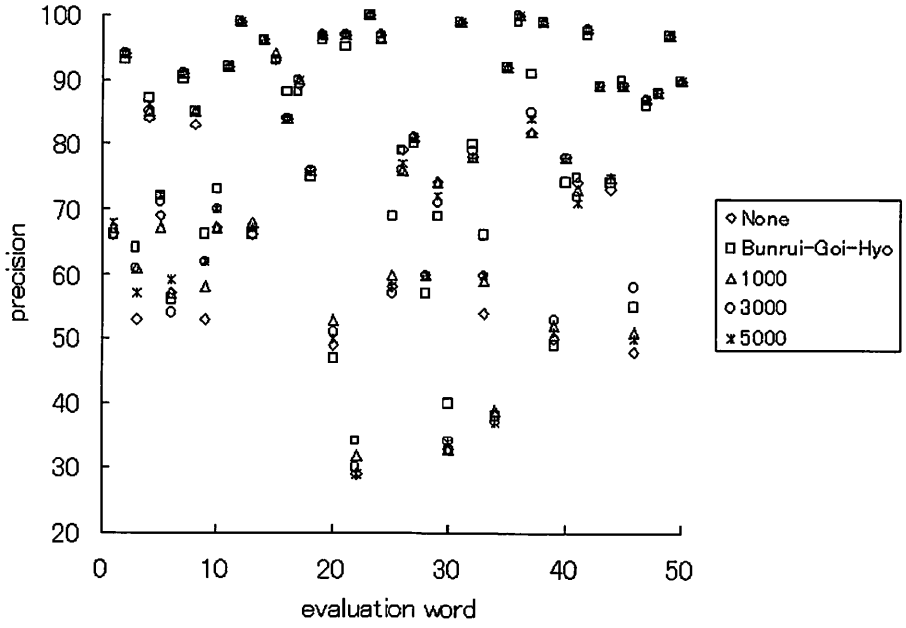

FIGURE 2. precision for the target words

The number of words	1000	3000	5000
Increase	21	17	20
Decrease	16	17	16

TABLE 1. the number of words for which precision is increased and decreased

number and using the feature set with the semantic class in the "Bunrui-Goi-Hyo" respectively.

To compute the precision, we use two scoring methods as evaluation criteria according to the SENSEVAL-2 workshop[Kurohashi and Shirai2001]. our results are based on accuracy using fine-grained scoring method. Fine-grained scoring is defined as the ratio of word sense which conforms the correct answer perfectly. For the model using the "Bunrui-Goi-Hyo", since a word sense is defined with a layered structure in this thesaurus, results are based on accuracy using mixed-grained scoring method. Mixed-grained scoring is the method which gives partial points according to the layered structure of the word sense.

Dimension	1000	3000	5000	None	Bunrui-Goi-Hyo
Precision	0.78105	0.782	0.781	0.7738	0.78785

TABLE 2. Average precision in the case of using the cooccurrence relation matrix

Dimension	1000	3000	5000
Precision	0.7806	0.7828	0.7804

TABLE 3. Average precision in the case of using the term-document matrix

Figure 2 shows the result of our experiment. In this figure, horizontal axis corresponds to the 50 verbs in the Japanese dictionary task and vertical axis represents the precision for each verb. About words for which precision is comparatively high, there was little change in the precision even if the cluster information is used. However, for the words with low precision, there was some change in the precision using the word cluster or the "Bunrui-Goi-Hyo". Table 1 shows the increase and decrease of the precision from the case where "Bunrui-Goi-Hyo" is used. In this table, precision was increased in some of the number of words for using our cluster compared with the use of the "Bunrui-Goi-Hyo". Therefore, the efficiency using the word clustering is higher than using the "Bunrui-Goi-Hyo".

Table 2 and 3 show the average precision for the word clustering using cooccurence relation matrix, the "Bunrui-Goi-Hyo" and term-document matrix respectively. In the table 2, the average precision was 78.2% using 3,000 clusters and 78.1% using 1,000 and 5,000 clusters. these results show the precision using word clustering was less than the average precision using the "Bunrui-Goi-Hyo". However, the precision using word clustering was higher than the precision without clustering so using word cluster has some benefit for the word sense disambiguation.

Comparing with the average precision using the term-document matrix, The average precision using both the word clustering and the term-document matrix brought the almost same result in 3,000 clusters. However, in 1,000 and 5,000 clusters, the performance using the word clustering was a little better than using the term-document matrix. Therefore, using the word clustering have a beneficial effect on the word sense disambiguation compared with using the term-document matrix.

4. CONCLUSION

In this paper, we propose a new word clustering algorithm for a large scale set of words with the aim of automatic construction for word sense disambiguation. We give experimental results of evaluation of the obtained clusters using the Japanese dictionary task. In comparison with using the term-document matrix, we found that using the word clustering have a beneficial effect on the word sense disambiguation. In comparison with using the "Bunrui-

Goi-Hyo", we found that precision was increased in some of the number of words for using our cluster. Further work could involve analyzing the other clustering algorithm to obtain the better word clusters.

REFERENCES

[Bellegarda et al.1996] J. R. Bellegarda, J. W. Butzberger, and Y.-L. Chow. 1996. A novel word clustering algorithm based on latent sema ntic indexing. In *Proceedings of the 1996 IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP-96)*, pages 172–175.

[Chicholm and Kolda1998] E. Chicholm and T. G. Kolda. 1998. New term weighting formulas for the vector space method in inf ormation retrieval. Technical report, Oak Ridge National Laboratory, Oak Ridge, Tennessee.

[Dhillon and Modha1999] I. S. Dhillon and D. S. Modha. 1999. Concept decompositions for large sparse text data usin g clustering. Technical report, IBM Almaden Research Center.

[Kawamae et al.2001] Noriaki Kawamae, Terumasa Aoki, and Hiroshi Yasuda. 2001. The word clustering based on statistical model. in technical report of nlp, IPSJ.

[Kita1999] Kenji Kita. 1999. *Probabilistic Language Model.* University of Tokyo Press.

[Kurohashi and Shirai2001] Sadao Kurohashi and Kiyoaki Shirai. 2001. Senseval-2 japanese task. in technical report of nlc, IEICE.

[Shinnou and Sasaki2002] Hiroyuki Shinnou and Minoru Sasaki. 2002. Fast method of word sense disambiguation using information retrieval technique. in technical report of nlp, IPSJ.

[Tokunaga1999] Takenobu Tokunaga. 1999. *Information Retrieval and Natural Language Processing.* University of Tokyo Press.