2011 2nd International Conference on Networking and Information Technology
IPCSIT vol.17 (2011) © (2011) IACSIT Press, Singapore

Dynamic Construction of Hierarchical Thesaurus using Co-occurrence Information

Kazuhiro Morita, Shuto Arai, Hiroya Kitagawa, Masao Fuketa and Jun-ichi Aoe

Dept. of Information Science and Intelligent Systems , The University of Tokushima, Tokushima, Japan
e-mail: kam@is.tokushima-u.ac.jp

Abstract—The thesaurus which is one of the important knowledge in the natural language processing is manually made in general. Therefore, high accuracy is obtained. However, it is difficult to update it growing of the scale to take huge time by the hand work frequently. This paper aims to construct a hierarchical large-scale thesaurus by using clustering based on the co-occurrence information among words. The thesaurus of about 60,000 words was made by using the co-occurrence information of about 16 million extracted from the Google N-gram. When the node was extracted from the made thesaurus at random and accuracy had been requested, it was understood about 82%, and to be able to make it even in a large-scale thesaurus by high accuracy.

Keywords-thesaurus; co-occurrence; clustering; hierarchy; similarity measurement;

1. Introduction

The thesaurus which is one of the important knowledge in the natural language processing is applied to a document classification using semantic contents [1], a query expansion in information retrieval, a word sense disambiguation [2], and so on.

Because the thesaurus such as WordNet [3] is manually made in general, high accuracy is obtained. However, it is difficult to update it growing of the scale to take huge time by manual operation frequently.

On the other hand, various researches that automatically construct the thesaurus are performed. For instance, the generating method using the neural network algorithm [4], clustering techniques using the co-occurrence relation [5,6,7], and so on are proposed.

However, it is difficult to divide an unknown word into a new classification node. Moreover, a minute change like the addition of an unknown word needs a big cost because of static construction.

This paper aims to construct a hierarchical large-scale thesaurus by using clustering based on the co-occurrence information among words, and proposes the clustering algorithm for paying attention to a minute and frequent change such as adding an unknown word.

In the experiment to evaluate the effectiveness of the proposal technique, the accuracy of clustering is confirmed, and the thesaurus of a large scale is constructed using co-occurrence information on about 16 million extracted from the Google N-gram [8].

2. Co-Occurrence Information and Clustering

2.1 Co-occurrence Information

The co-occurrence is that words appear at the same time in the document.

In this paper, the co-occurrence information is defined between a registered word X and a co-occurrence word Y attended with co-occurrence frequency f as (X, Y, f).

2.2 Attribute Vector

The attribute vector is generated using co-occurrence information for clustering.

The attribute vector includes the word vector made as features of each registered words, and the node vector registered as a feature of each node of the thesaurus tree.

1) Word Vector

The word vector is composed of co-occurrence words and co-occurrence frequencies.

Word vector W_X of a word X that co-occurs with co-occurrence words Y_i of t times can be shown as follows:

$$W_X = \sum_{i=1}^{t} f_i V_{Y_i} \tag{1}$$

where, f_i is co-occurrence frequency, and V_{Y_i} is a linearly independent unit vector corresponding to Y_i. Thereafter, V_{Y_i} is omitted for simplifying.

The sum and the difference between arbitrary word vector W_A and W_B are defined respectively as follows.

$$W_A + W_B = \sum_{i=1}^{t'} (f_{A_i} + f_{B_i}) \tag{2}$$

$$W_A - W_B = \sum_{i=1}^{t'} (f_{A_i} - f_{B_i}) \tag{3}$$

Where t' is number of differences of co-occurrence words of A and B. f_{A_i} and f_{B_i} are each co-occurrence frequency.

2) Node Vector

The node vector shows the feature of each node of the thesaurus tree, and it is composed by word vectors of all registered words that belong to the node.

At this time, node vector G with the registered word of s times is as follows.

$$G = \sum_{j=1}^{s} W_j \tag{4}$$

2.3 Similarity measurement

Distance $D(A, B)$ of word A and B is used as a similarity measurement for clustering of the word.

The cosine measure, the Dice coefficient, the Pearson product-moment correlation coefficient, and the Jaccard coefficient are proposed as a similarity measurement used in a basic clustering technology.

1) Cosine Measure

The cosine measure is a basic similarity used by a lot of clustering algorithms, and can be derived from the definitional equation of inner product of vectors.

Cosine measure $D_C(A, B)$ can be shown as follows.

$$D_C(A,B) = \frac{\sum_{i=1}^{t'} (f_{A_i} \cdot f_{B_i})}{\sqrt{\sum_{i=1}^{t'} f_{A_i}{}^2}\sqrt{\sum_{i=1}^{t'} f_{B_i}{}^2}} \tag{5}$$

2) Dice Coefficient

To reduce the problem of the cosine, the Dice coefficient is given to normalization.

Dice coefficient $D_D(A, B)$ can be shown as follows.

$$D_D(A,B) = \frac{2\sum_{i=1}^{t'} (f_{A_i} \cdot f_{B_i})}{\sum_{i=1}^{t'} f_{A_i}{}^2 + \sum_{i=1}^{t'} f_{B_i}{}^2} \tag{6}$$

3) Pearson Product-moment Correlation Coefficient

The correlation coefficient is a statistics index that shows the correlation (similarity) between two probability variables.

In general, the correlation coefficient indicates Pearson product-moment correlation coefficient.

Correlation coefficient $D_P(A, B)$ can be shown as follows:

$$D_P(A,B) = \frac{\sum_{i=1}^{t'}(f_{A_i} - \bar{f}_A)(f_{B_i} - \bar{f}_B)}{\sqrt{\sum_{i=1}^{t'}(f_{A_i} - \bar{f}_A)^2}\sqrt{\sum_{i=1}^{t'}(f_{B_i} - \bar{f}_B)^2}} \tag{7}$$

where $\bar{f}_A$ and $\bar{f}_B$ are arithmetic means of data f_{A_i}, f_{B_i} respectively.

4) Jaccard Coefficient

The identity frequency of two sets is originally shown, and it is used also for the similarity computation of the vector similarly.

Jaccard coefficient $D_J(A, B)$ can be shown as follows.

$$D_J(A,B) = \frac{\sum_{i=1}^{t'}(f_{A_i} \cdot f_{B_i})}{\sum_{i=1}^{t'} f_{A_i}{}^2 + \sum_{i=1}^{t'} f_{B_i}{}^2 + \sum_{i=1}^{t'}(f_{A_i} \cdot f_{B_i})} \tag{8}$$

2.4 Kullback-Leibler Divergence

General distance $D(A,B)$ used by clustering has symmetry, that is $D(A, B) = D(B, A)$. However, because these judge the similarity only by the distance, the hierarchy that is important in the construction of the thesaurus cannot be considered.

On the other hand, the technique that uses the Kullback-Leibler divergence as a similarity measurement that derives the hierarchy of the word is researched [9]. Bessho et al. shows that the Kullback-Leibler divergence ties strongly the word related to the high rank and the subordinate position compared with the distance of the Euclid.

The Kullback-Leibler divergence has no symmetry, and the value that $D(A, B)$ and $D(B, A)$ are different is taken. As a result, direction of the high rank and the subordinate position and the similarity can be derived between A and B.

Then, the Kullback-Leibler divergence is introduced into the proposal technique in this research.

The Kullback-Leibler divergence calculates the similarity by deriving the expectation of entropy loss between A and B using probability distribution.

When relative value F_{A_i} and F_{B_i} are shown as $F_{A_i} = f_{A_i} \Big/ \sum_{i=1}^{t'} f_{A_i}$ and $F_{B_i} = f_{B_i} \Big/ \sum_{i=1}^{t'} f_{B_i}$ respectively, Kullback-Leibler divergence $KL(A, B)$ can be shown as follows:

$$KL(A,B) = \sum_{i=1}^{t'} F_{A_i} \log \frac{F_{A_i}}{F_{B_i}} \tag{9}$$

however the following are exceptionally defined.

$$F_{A_i} \log \frac{F_{A_i}}{F_{B_i}} = \begin{cases} 0 & \text{if } F_{A_i} = 0 \\ F_{A_i}(\log F_{A_i} + C) & \text{if } F_{B_i} = 0 \end{cases} \tag{10}$$

Constant C is assumed finite value though it becomes infinity at $F_{B_i} = 0$.

$KL(A,B)$ becomes small if A is a broader term of B because there are a similarity of the Kullback-Leibler divergence near 0.

3. Algorithm of Classification and Hierarchy

3.1 System Overview

Fig. 1 shows the system overview chart in this research.

This system divides roughly into two processing shown as follows.

1) *Registration of Co-occurrence Information*

In registration processing of co-occurrence information, if the registered word has been inserted in the thesaurus tree, it removes from the thesaurus tree once.

Afterwards, information is registered in the collocation dictionary by new or the addition.

This processing is repeated until the input of co-occurrence information ends.

2) *Thesaurus Tree Updating*

In the thesaurus tree dictionary, information on registered words included in each node, a node vector, and link information of the node are stored.

For registration to the dictionary, the word vector is made for the registered word, and the similar node to the registered word is searched in the thesaurus tree. At this time, the measurement and the direction of similarity between the word vector and the node vector are calculated by the Kullback-Leibler divergence.

First, the node with the highest similarity as a similar node candidate is decided from among child nodes of *Root* for the Kullback-Leibler divergence. In the same way, the similarity is calculated for child nodes of a similar node candidate, and the node with the highest similarity is chosen as the similar node.

Next, whether the node is newly added or the registered word is included to the similar node is decided by using another similarity measurement.

This processing is repeated until all registered words are added to the thesaurus tree.

3.2 Procedure to Register Co-occurrence Information

1) *Removing of Word Vector*

When word vector W_X is removed from node vector G_n of node n that belongs registered word X, node vector $G_n{}'$ as updated G_n is shown by the following expressions.

$$G_n' = G_n - W_X \tag{11}$$

In case of G_n=W_X, node n is deleted because of $G_n' = \phi$, and child nodes of n are connected to the parent node of n as shown in Fig. 2.

2) *Storing to Collocation Dictionary*

Co-occurrence information (X, Y, f) is stored to the collocation dictionary using the Link Trie Structure [10].

In this case, if (X, Y, f') has already been stored, it is updated to (X, Y, f+f').

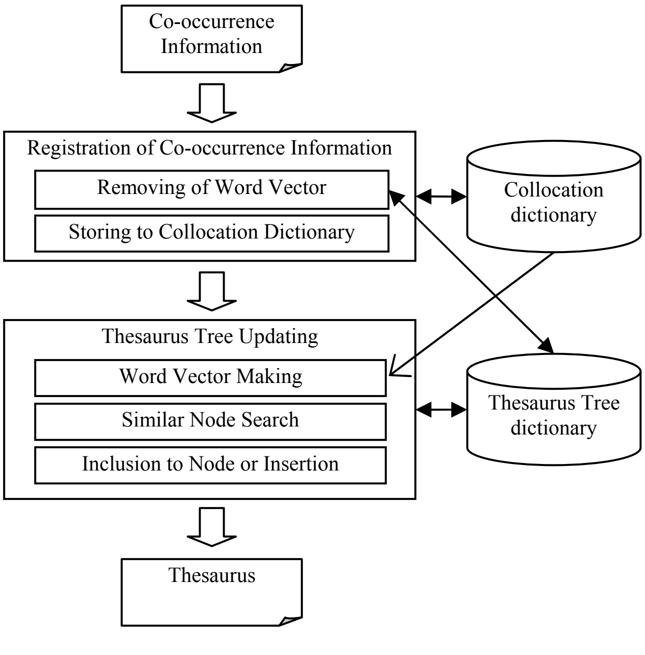
Co-occurrence Information
Registration of Co-occurrence Information
Removing of Word Vector
Storing to Collocation Dictionary
Collocation dictionary
Thesaurus Tree Updating
Word Vector Making
Similar Node Search
Inclusion to Node or Insertion
Thesaurus Tree dictionary
Thesaurus

Figure 1. System Overview Chart.

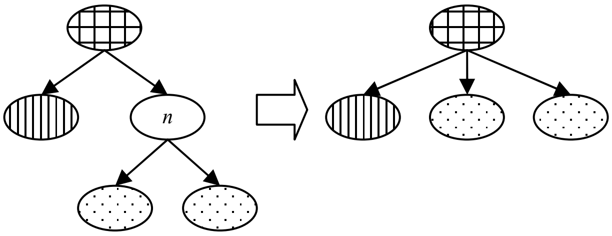

Figure 2. Example of Deleting Node n.

3.3 Procedure to Update Thesaurus Tree

1) *Word Vector Making*

Set $K=\{(X, Y_i, f_i) \mid i=1,2\ldots s\ldots t, f_i>f_{i+1}\}$ of co-occurrence information with registered word X is acquired referring to the collocation dictionary.

Word vector $W_X = \sum_{i=1}^{s} f_i$ is made from K.

2) *Similar Node Search*

Simirarity measurement *KLsim* and Simirarity direction *KLdir* using Kullback-Leibler divergence *KL* is defined respectively as follows.

$$KLsim(G_n, W_x) = \min(KL(G_n, W_X), KL(W_X, G_n)) \tag{12}$$

$$KLdir(G_n, W_X) = \begin{cases} -1 & \text{if } KL(G_n, W_X) \le KL(W_X, G_n) \\ 1 & \text{if } KL(G_n, W_X) > KL(W_X, G_n) \end{cases} \tag{13}$$

However, the threshold α is set, and it is judged that there is no similarity between G_n and W_X if $KLsim(G_n,W_X)>\alpha$. The direction of the high rank becomes $KLdir(G_n,W_X)<0$ for node n, and the direction of the subordinate position becomes $KLdir(G_n,W_X)>0$.

Let S_n be a set of node n and all child nodes of n. The following steps search the similar node.

Step1 n is initialized with *Root*.

Step2 Node vector G_i ($i \in S_n$) are made for each nodes of S_n. The node i which similarity $KLsim(G_i,W_X)$ is minimized, and its direction $KLdir(G_i,W_X)$ are requested.

Step3 Step2 is repeated until becoming $i=n$.

3) *Inclusion to Node or Insertion*

Whether registered word X is contained to the similar node n or becomes a new node is decided by using distance $D(G_n,W_X)$.

If $D(G_n, W_X) \le \beta$, which is the threshold, X is included among a set R_n of registered words that belongs to node n, that is, $R_n' = R_n \cup \{X\}$. And G_n is renewed as $G_n' = G_n + W_X$.

If $D(G_n, W_X) > \beta$, new node m is added as $R_m=\{X\}$ and $G_m=W_X$.

Node m is inserted between node n and its parent node p when $KLdir(G_n,W_X)<0$. Conversely, if $KLdir(G_n,W_X)>0$, m is added as a child node of n.

Fig. 3 shows examples of the insertion process.

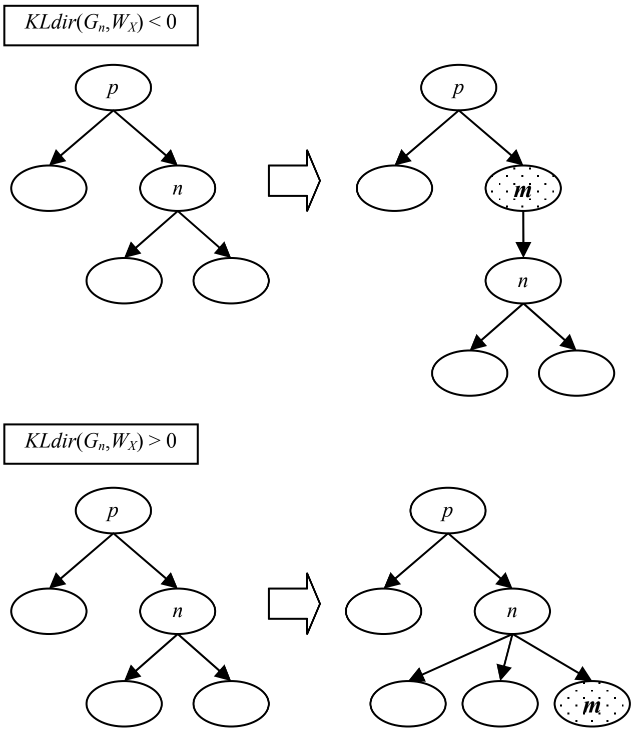

Figure 3. Example of the Insertion Process.

4. Experiments adn Evaluation

The thesaurus was made by using co-occurrence information extracted from the Google N-gram [8] to evaluate the effectiveness of the proposal technique, and the preliminary experiments compared with the k-means method [11] ware done.

Moreover, the construction time of a large-scale thesaurus was measured, and the evaluation experiment that confirmed the accuracy of clustering could be done.

4.1 Preliminary Experiment

In the preliminary experiment, the thesaurus was made by using co-occurrence information on about 240,000, and each value of the threshold α and β ware decided.

By using the decided threshold, how the high rank and the subordinate position relation ware constructed was confirmed in comparison with the k-means method.

Fig. 4 shows the result of deciding the threshold α.

F-measure of 0.799, recall of 0.735, precision of 0.877 that was the highest result was indicated at threshold α=65.

Next, the result of deciding the threshold β is shown in Fig. 5. The highest F-measure of 0.852 was indicated at threshold β=0.03 using the Dice coefficient.

The experimental result of using the k-means method showed F-measure of 0.758, recall of 0.821, precision of 0.703. The proposal technique became F-measure that was about 10% higher than the k-means method.

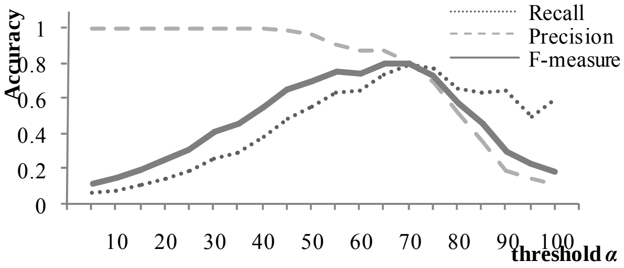

Figure 4. Change of Accuracy according to the threshold α.

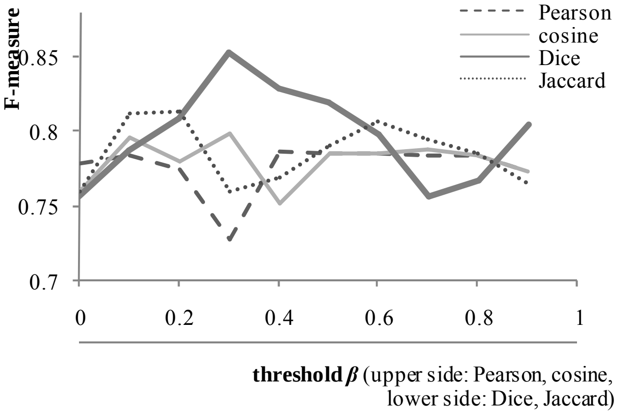

Figure 5. Change of F-measure according to the threshold β.

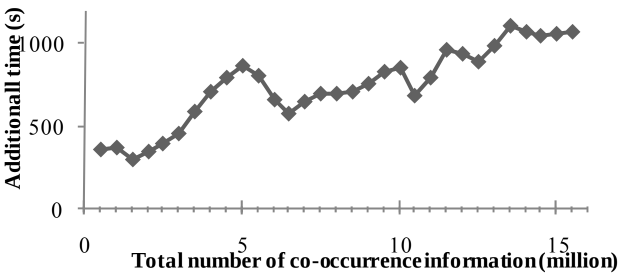

Figure 6. Additional Time per 500,000 Co-occurrence Information.

4.2 Evaluation Experiment

500,000 co-occurrence information is gradually added to the thesaurus tree made by the preliminary experiment.

Fig. 6 shows the result of measuring this additional time. The average per one addition time was 1.48ms, and an excellent result was obtained.

Finally, co-occurrence information on 16,605,816 totals was added and the thesaurus tree dictionary was constructed. 1,998 subtrees from *Root* were made.

Some subtrees are pulled out from this constructed system at random, and the result of confirming precision is shown in Table 1.

The node number in the table shows the child node number of *Root*, and the number of registered words contains the subtree below the child node. Total Precision became 80% or more and good results. It has been understood to be able to make it from this even in a large-scale thesaurus by high accuracy.

5. Conclusion

This paper has presented the technique to construct a hierarchical large-scale thesaurus using clustering based on the co-occurrence information of the word.

In the proposed new clustering algorithm, the Kullback-Leibler divergence to judge the high rank and the subordinate position relation has been introduced. Besides, the thesaurus tree can be updated in each node for a minute change like the addition of an unknown word.

In the experiment to evaluate the effectiveness of the proposal technique, the accuracy of clustering was confirmed, and the construction time was measured.

When the thesaurus was made by using co-occurrence information on about 240,000 extracted from the Google N-gram to confirm the accuracy of clustering, the accuracy of the proposal technique became 85% which was about 10% higher than accuracy of the k-means method.

Moreover, when the thesaurus of about 60,000 words was made by using co-occurrence information on about 16 million extracted from the Google N-gram and accuracy was confirmed, an effective result of about 82% was obtained.

In the future, it will aim at the construction of a more useful thesaurus by doing the experiment that uses a lot of words that exist in the high rank and the subordinate position relation repeatedly.

TABLE I. PRECISION OF SUBTREES PULLED OUT

Node No.	Number of registered words	Number of correct words	Number of incorrect words	Precision (%)
3	441	387	54	87.75
7941	138	126	12	91.30
3334	304	225	79	74.01
8	476	440	36	92.43
17929	60	54	6	90.00
24588	30	13	17	43.33
1038	481	346	135	71.93
2320	208	133	75	63.94
10776	116	97	19	83.62
18202	44	31	13	70.45
38427	62	61	1	98.38
33054	32	30	2	93.75
22829	36	32	4	88.88
21045	583	513	70	87.99
Total	3011	2488	523	82.63

6. References

[1] Pu Wang, Jian Hu, Hua-Jun Zeng and Zheng Chen, "Using Wikipedia knowledge to improve text classification", Knowledge and Information Systems, Volume 19, Number 3, pp.265-281, 2008.

[2] F. J. Pinto, A. F. Martinez and C. F. Perez-Sanjulian, "Joining automatic query expansion based on thesaurus and word sense disambiguation using WordNet", International Journal of Computer Applications in Technology, Vol.33, No.4, pp.271-279, 2008.

[3] Christiane Fellbaum, WordNet: An Electronic Lexical Database, Cambridge, MA: MIT Press, 1998.

[4] V. J. Hodge, and J. Austin, "Hierarchical word clustering—Automatic thesaurus generation". Neurocomputing, Vol.48, No.1-4, pp.819–846, 2002.

[5] H. Chen, B. R. Schatz, T. Yim and D. Fye, "Automatic Thesaurus Generation for an Electronic Community System", Journal of the American Society for Information Science, Vol.46, No.3, pp.175-193, 1995.

[6] H. Schutze and J.O. Pedersen, "A Cooccurrence based Thesaurus and Two Applications to Information Retrieval", International Journal of Information Processing and Management, Vol.33, No3, pp.307-318, 1997.

[7] K. Morita, E-S. Atlam, M. Fuketa, K. Tsuda, M. Oono and J. Aoe, "Word classification and hierarchy using co-occurrence word information", Information Processing & Management, Vol.40, No.6, pp.957-972, 2004.

[8] Taku Kudo, Hideto Kazawa, "Web Japanese N-gram Version 1", published by Gengo Shigen Kyokai, 2007.

[9] K. Bessho, T. Uchiyama, R. Kataoka, "Extraction of Hierarchical Relations among Words Based on Co-occurrences between Words and Semantic Attributes", IEIC Technical Report (Institute of Electronics, Information and Communication Engineers), VOL.106, NO.518, pp.31-36, 2007.

[10] K. Morita, M. Koyama, M. Fuketa and J. Aoe, "A Link Trie Structure of Storing Multiple Attribute Relationships for Natural Language Dictionaries", Computer Mathematics, Vol.72, No.4, pp.463-476, 1999.

[11] J. B. MacQueen, "Some methods for classification and analysis of multivariate observations", Proceedings of the Fifth Symposium on Math, Statistics, and Probability, pp.281-297, 1967.