The Word Clustering Method for Lexical Knowledge Acquisition from Domain-Specific Documents

Takahiro SAITO† **Isamu WATANABE**† **Kunio MATSUI**†
Akira TERADA‡ **Takashi SAITO**‡
† Fujitsu Laboratories Ltd, 4-1-1, Kamikodanaka, Nakahara-ku, Kawasaki, Kanagawa 211-8588, Japan
‡ Japan Airlines International Co.,Ltd, Terminal1, 3-3-2, Haneda Airport, Ota-ku, Tokyo 144-0041, Japan

ABSTRACT

In this paper, we introduce a new similarity measure between words, and a graph-based word clustering method using this similarity measure. Our similarity measure is a quantification of the "mutual substitutability" of two words, and our graph-based word clustering method is composed of two steps. The first step is a building pairs of terms whose similarity measures are high into the connected graphs, and the second step is a division of the connected graphs by estimating the density of their edges.

Here we report on the results of experiments in which we compared our method with existing techniques. In these experiments, we attempted to acquire the lexical knowledge from aviation incident reports. To conclude, we show that our similarity measure is more suitable for this purpose than the cosine measure, a popular similarity measure, and show that our clustering method creates more meaningful clusters than the k-means clustering method, a popular clustering method.

Keywords: clustering, graph, similarity-measure, mutual-substitutability

1 INTRODUCTION

The Aviation Safety Reporting System(ASRS) is a large scale database that contains more than 200,000 reports of aviation incidents. These reports are analyzed to reduce the likelihood of subsequent aviation accidents. However, to do this task manually is time consuming and tiresome. It is required to analyze them by computer. But this analysis is difficult due to the numerous abbreviations and technical terms in the reports. For appropriate analysis of documents such as ASRS, it is necessary to have knowledge about the lexicons in the target document set(e.g. the words 'SMT', 'SMA', 'LTT' all mean aircraft in ASRS). This knowledge is dependent upon the target documents, and therefore it is very useful to acquire this automatically. In order to obtain lexical knowledge from documents, we introduce a new word clustering method which can generate proper word clusters from the set of words contained in the target documents.

2 METHODS

There are two key ideas in our method: a high-accuracy similarity measure by which we can get a pair of words belonging to the same word class, and a graph-based clustering method that builds proper word clusters from pairs of words.

2.1 Similarity Measure

We adopt the "mutual substitutability" measure(we call it "S-measure", for short) as a similarity measure, and here we introduce some relevant notation:

- X and Y are terms which are either a word or a compound word.
- $Phrs(X)$ is the set of phrases(word sequences in corpus, for example, sentence, clause, noun phrase, etc.) containing a term X.
- $N(X)$ is the number of phrases in $Phrs(X)$.
- $Phrs(X, Y)$ is the subset of $Phrs(X)$, such that for any phrase p in $Phrs(X, Y)$, there exsists a phrase where Y replaces X in $Phrs(Y)$
- $N(X, Y)$ is the number of phrases in $Phrs(X, Y)$

Using this notation, $S(A, B)$, the S-measure between the term A and the term B, is defined as follows:

$$S(A, B) = N(A, B)^2/(N(A) * N(B))$$

For example, if the phrase "the action of the *pilot*" and the phrase "the action of the *plt*" are in input phrases, the former belongs to $Phrs$("*pilot*") and $Phrs$("*pilot*", "*plt*"), the latter belongs to $Phrs$("*plt*") and $Phrs(\text{"}plt\text{"}, \text{"}pilot\text{"})$. And if $Phrs(\text{"}pilot\text{"})$ consists of ten phrases, and $Phrs(\text{"}plt\text{"})$ consists of this phrase only, $S(\text{"}pilot\text{"}, \text{"}plt\text{"}) = 1^2/(10 * 1) = 0.1$.

We calculate this similarity measure between the terms in the given set of terms, and extract the high-score pairs of terms as a pair of terms, called "relation", belonging to the same term class.

However, even if the relation is correct(both terms in the relation belong to the same term class), it is not always valuable, for example, a pair of articles. Additionally, there may be wrong relations with a high S-measure, for example, a pair that comprises an article and an adjective. Therefore, we adopt filtering rules based on a Part-of-Speech(POS). For example, we remove pairs of terms that have a different POS; detailed filtering rules adopted in our experiment will be described later. By this filtering procedure before or after the selection of high-score pairs, we can eliminate obstructive relations.

2.2 Term Clustering

Our graph-based method is composed of two steps. The first step is a building the relations into connected graphs. The second step is a division of the connected graphs by removing some edges. The procedure of the first step is well known; we can apply the 'Depth-First Search(DFS)' algorithm for this step without modification. For the second step, we describe the procedure that we used in our method.

For a detailed explanation of this, we define the following.

- r is the relation in the graph g.
- $R(c)$ is the set of r whose similarity measure are higher than that of relation c.
- $g(R)$ is the graph composed of the set of relations R.
- $N_S(g)$ is the number of connected subgraphs in graph g.
- $C(g)$ is a set of 'connectors', and the definition of a 'connector' is that:

  The relation c in graph g is a 'connector' $\Longleftrightarrow N_S(g(R(c) \bigcup \{c\})) < N_S(g(R(c)))$

  Let the two connected subgraphs of $g(R(c))$ connected by c be $I_1(c), I_2(c)$.

  Figure 1 is an example of a 'connector', which illustrates the above definitions.

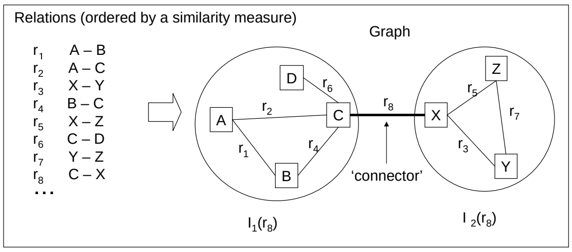

Figure 1: Definition of 'connector'

  In Figure 1, $r_i (i = 1, 2, ..8)$ are relations, and they are ordered by a similarity measure(higher order). That is, the similarity measure of relation r_i is higher than that of r_j if $i < j$. When we add the relations to the graph in this order, we find that the relation r_8 connects the two connected subgraphs $I_1(r_8), I_2(r_8)$, and the number of connected subgraphs decrease. Thus, r_8 is a 'connector'.

- $Splt(g, c)$ is the graph composed of the set of relations in the graph g except the relations connecting $I_1(c)$ and $I_2(c)$ where c is a 'connector'.
- $d(g)$ is the density of the edges of the connected graph g, that is, $2 * e(g)/(n * (n - 1))$, where $e(g)$ is the number of the edges in graph g, and n is the number of nodes in graph g.
- $avrd(g)$ is the average density of edges of graph g, that is, $\sum_i d(g_i)/N_S(g)$, where g_i is the connected subgraph in the graph g.

Let g be a connected subgraph obtained by the first step, if g is not complete, the following procedure is applied to g.

1. Getting $C(g)$
2. Getting $c_m \in C(g)$, such that

   $c_m = argMax_{c \in C(g)} \{avrd(Splt(g, c))\}$
3. If $avrd(Splt(g, c_m))$ is higher than $d(g)$,

   dividing g into $Splt(g, c_m)$.

This is recursively applied to all the connected subgraphs. Finally, we obtain a hierarchical structure of term clusters.

A concrete example of this procedure is described. Table 1 shows relations ordered by a similarity measure(higher order), and the numbers of connected subgraphs made with the relation and upper relations. This table shows that r_7 and r_{10} are 'connectors', which decrease the number of connected subgraphs.

Figure 2 shows the graph of these relations(Graph 0), and the division of Graph 0 by each 'connector'. For example, when we add the 'connector' relation r_7 to the graph, it connects the graph of r_5 and the graph of r_3, r_6. Furthermore, the relation r_{11} and the relation r_{12} also connect these two graphs. Thus, we remove the relations r_7, r_{11}, r_{12} from Graph 0, and get the Graph 1, and calculate the average density of edges. Similarly, we get Graph 2 in the case of 'connector' r_{10}, and calculate the average density of edges of the graph. The density of edges of Graph 0 is 0.33(=12/36), and the average densities of Graphs 1 and 2 are 0.69(=(8/21+1/1)/2) and 0.63(=(4/6+6/10)/2), respectively. Hence Graph 0 is divided to give Graph 1.

Table 1: Examples of Relations

ID	Relation	Number of Connected Subgraphs
r_1	A - B	1
r_2	A - C	1
r_3	X - Y	2
r_4	B - C	2
r_5	P - Q	3
r_6	X - Z	3
r_7	Q - X	2(connector)
r_8	C - D	2
r_9	Y - Z	2
r_{10}	C - X	1(connector)
r_{11}	B - Q	1
r_{12}	P - Y	1

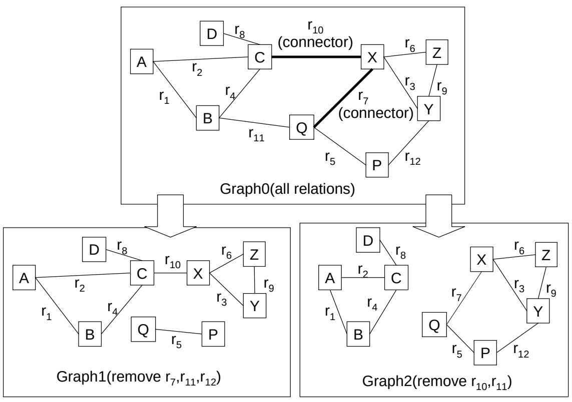

Figure 2: Example of Division of Connected Graph

3 EXPERIMENTS

In this section, we report two experiments. The first experiment is an evaluation of the clustering result achieved by our method. The second experiment is a comparison of our method and an existing method, comparing the clustering results from the same terms set.

3.1 Evaluation

For input data, we adopt base noun phrases and all their sub word sequences. These base noun phrases are extracted by 'fnTbl'[1], and we adopt the phrases which are extracted from more than one sentence in 80391 ASRS reports; we calculate all the pairs of terms and then apply the following filtering rules to the relations.

1. Eliminate pairs of terms in which the either or both of the last words of the terms are not nouns.
2. Eliminate pairs of terms in which the difference between the terms is one word of each terms, where these words are articles or possessive pronouns or adjectives, for example, the pair of 'the frequency' and 'our frequency' is eliminated by this filter.
3. Eliminate the pair of terms A and B if either $N(A) \leq 10$ or $N(B) \leq 10$.

By these rules, we can get 1147213 relations(2381 terms) whose similarity measures are not zero.
Finally, we select the best 300 pairs of nouns or compound nouns after filtering, and from these we make clusters.

The result is that 47 connected graphs(that is, 47 clusters) with 192 terms are built in the first step by our term clustering method. That is, we can get lexical knowledges about the 8%(=192/2381) of all the high-frequency nouns or compound nouns which may comprise a certain type of a word class.

Examples of these clusters are shown in Table 2.
There are only three clusters that contain terms that belong to obviously different word classes(Cluster 4, Cluster 16 and Cluster 17). The biggest cluster(Cluster 1) contains 49 terms that mean location or airport name. However, there is another cluster composed of two terms that also mean location(Cluster 2). Additionally, there are four clusters of the terms which mean aircraft(Clusters 3,4,5, and 6, although Cluster 4 contains an error), and two clusters of the terms

Table 2: Examples of Clusters

Cluster	Terms in the Cluster
1	CLT, BALTIMORE, PHENIX ...(49 terms)
2	WICHITA, OHAMA
3	SMT, LTT, MDT, LGT, AIRCRAFT
4	MR, CGA
5	ENGINE SMA, ENGINE AIRCRAFT
6	INTRUDER, ATR
7	W, S
8	SE, NE, NW
9	MODE CONTROL PANEL MCP
10	CTRLINE, CENTER LINE
11	MIKE, MICROPHONE
12	OCCURRENCE, EVENT
13	S/O, F/E, FE, ENGINEER
14	HSI, ADI, DG
15	R BASE, L BASE, BASE LEG, R DOWNWIND, L DOWNWIND
16	PLEASURE, PHOTO
17	VEHICLES, CREWS

which mean direction(Clusters 7, and 8). The other clusters do not need to be united, and are assumed to be clusters of synonyms or terms which belong to the same meaning class(Clusters 9,10,11,12 and 13[1],14[2],15[3]).

The second step of our clustering method is a division of the connected graphs by removing some edges. In this experiment, two divisions occurred. The first involves Cluster 1, which is devided into eight sub-clusters, but without any meaningful hierarchy. The other division relates to Cluster 15, which can be split into two sub-clusters: one composed of the terms 'R BASE', 'L BASE' and 'BASE LEG', and the other composed of the terms 'R DOWNWIND' and 'L DOWNWIND'. This division seems meaningful.

3.2 Comparison

To compare the clustering methods easily, we used only the pairs of terms listed below. These terms are selected from the set of high-frequency words in the ASRS reports.

- Group 1(AIRCRAFT): CPR,SMA,AIRPLANE,MLG,SMT, MLT,ACFT,LGT,LTT,COMMUTER
- Group 2(HUMAN): PAX,MGR,S/O,FO,PLT,COPLT, CAPT,SUPVR,FE,CREW
- Group 3(LOCATION): SAN,LAX,ORD,JFK,OAK,PHX, CLT,PHL,BWI,DTW

The 135 pairs of terms which belong to the same group are regarded as correct.

1. *Comparison of Similarity Measures*

   The most popular similarity measure between the terms seems to be the cosine measure(C-measure, in short). We compare the accuracy of the C-measure with our S-measure.

   The C-measure between the terms A and B is defined by the inner product between A and B, and the norms of A and B. To calculate these values, we should express each term as a vector(usually called a "feature vector"). According to [2], the meaning of an English term is strongly bounded by the one or two morphemes just before it. Therefore, we express a target term as the vector whose components are the one and two morphemes just before it.

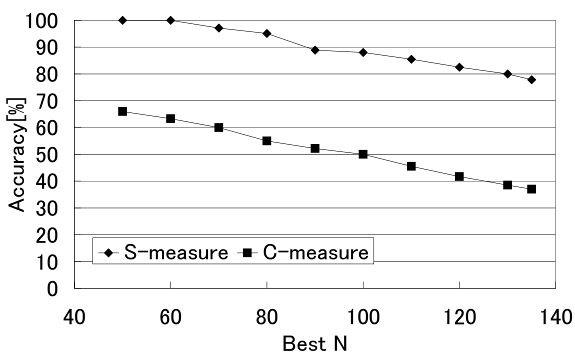

Figure 3: Accuracy of Best n Relations

Figure 3 is a result of the accuracy of best n relations estimated by the S-measure and the C-measure. Based on this figure, we can say followings;

[1]Terms in this cluster refer to people.
[2]Terms in this cluster refer to the instruments in aircraft.
[3]Terms in this cluster refer to the traffic pattern.

- S-measure is more accurate than C-measure.

  The accuracy of the S-measure appears to be higher than C-measure, clearly.

- S-measure is an appropriate similarity measure.

  The accuracy of S-measure is a monotone decreasing function at the macro level, a necessary characteristic for such a measure, and one also seen in the C-measure.

2. *Comparison of Clustering Methods*

   There are various clustering methods. They are classified into two types: hierarchical, and non-hierarchical. Our method belongs to the hierarchical clustering method. However, the evaluation of a hierarchical structure is more difficult than the evaluation of a non-hierarchical structure, and our method cannot assure that the hierarchy of clusters corresponds to the hierarchy of meanings. Therefore, we compare our method and the k-means clustering method which is a representative method of the non-hierarchical clustering method.

   The k-means clustering method is a method of clustering the points allocated in a vector space. In contrast, our method handles pairwise relations. To overcome this, we express a target term as a vector whose components are the similarity measure between the two target words(let the similarity measure of itself be 1, and the similarity measure of the terms which are not in the input relations be 0).

   We use the following scores introduced in [3] for the estimation.

   - Rand Statistic :
     $RS = (SS + DD)/(SS + SD + DS + DD)$
   - Jaccard Coefficient:
     $JC = SS/(SS + SD + DS)$
   - Folkes and Mallows Index:
     $FM = \sqrt{SS/(SS + SD) * SS/(SS + DS)}$

   Where,

   - SS: the number of the pairs of terms that belong to the same cluster and to the same group
   - SD: the number of the pairs of terms that belong to the same cluster and to different groups
   - DS: the number of the pairs of terms that belong to different clusters and to the same group
   - DD: the number of the pairs of terms that belong to different clusters and to different groups.

   In this experiment, we use the best 135 relations for clustering. In the case of the k-means method, we should specify a parameter that defines the number of clusters. The results of clustering stand on this parameter(we write this number "NoC" in the tables), and then, we adopt the number of clusters made by our method, as well as three, which is the number of term groups defined above as this parameter. In addition, the result of k-means clustering depends on the initial seeds, with these estimate values being averages of 10,000 tests with initial seeds selected at random.

   We want to estimate the clustering methods independently of the accuracy of the similarity measure. For that purpose, we perform two experiments, one adopting the S-measure as a similarity measure(table 3), and one adopting the C-measure as a similarity measure(table 4). In the first case, the best 135 relations contain all terms in the test word set(30 terms), but in the latter case, the best 135 relations contain only 22 terms.

   Tables 3 and 4 show that our method is better than the k-means method regardless of the similarity measures.

Table 3: Estimation of Clustering(S-measure)

Procedure	NoC	RS	JC	FM
Our procedure	7	0.862	0.556	0.745
k-means(k=7)	7	0.831	0.494	0.680
k-means(k=3)	3	0.781	0.544	0.698

Table 4: Estimation of Clustering(C-measure)

Procedure	NoC	RS	JC	FM
Our procedure	3	0.762	0.476	0.645
k-means	3	0.591	0.299	0.462

3.3 Discussion

Figure 4 shows the result of clustering, which is estimated in the comparative experiment, for the case where our method and the S-measure are used. In this figure, the group number of each term is added after the target term, for example, 'AIRPLANE(1)' means that the term 'AIRCRAFT' is a member of Group 1("AIRCRAFT" group).

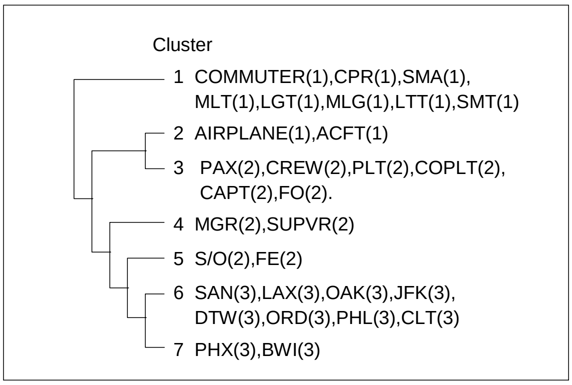

Figure 4: Result of Clustering(S-measure)

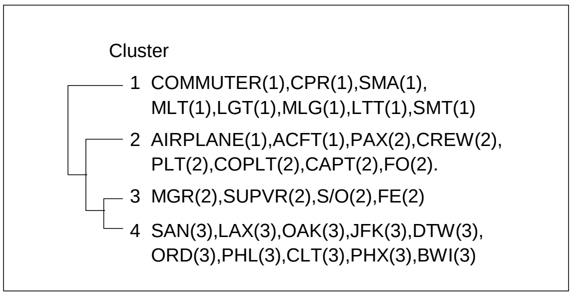

Figure 5: Result with the Restriction

This result shows that an unnecessary division occurred(the division of Cluster 6 and Cluster 7), and that small clusters composed of only two terms are generated(Cluster 2, 4, 5), but all clusters are pure. If we add a restriction to the division, making the small graph composed of only two nodes for the purpose of reducing the unnecessary divisions, the result of clustering is shown in Figure 5. In this case, Cluster 2 in Figure 5 contains the different group terms, but the three estimate values defined above are improved across the board(RS is 0.880, JC is 0.646, FM is 0.790). However, this restriction cannot assure this improvement for all cases, particularly as the impact of this restriction seems to depend on the set of target terms.

Additionally, the result of clustering depends on the number of relations adding to the graph. Table 5 is a results of the three estimate values defined above when the number of relations(n) is changed.

Table 5: Estimation of Clustering of Best n Relations

n	NoC	RS	JC	FM
100	7	0.862	0.556	0.745
135	7	0.862	0.556	0.745
200	4	0.880	0.646	0.790
250	5	0.908	0.704	0.839
300	5	0.926	0.763	0.873
350	4	0.889	0.682	0.813
400	3	0.799	0.553	0.726

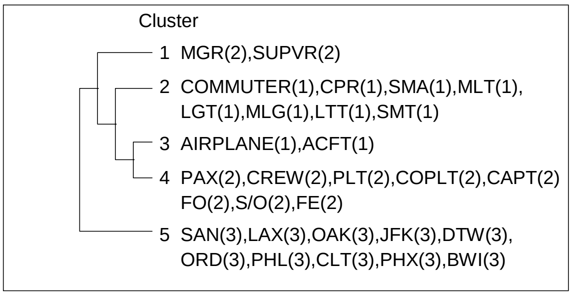

Figure 6: Result of Best Case

In the best case($n = 300$), the accuracy of relations is only 0.433. Although this result cannot be generalized, it shows that we can adopt more relations than we expected. The reason for this appears to be that our similarity measure is a proper method, and that our clustering method aggressively eliminates doubtful relations[4]. The result of the best case is described in Figure 6. Comparing this with Figure 4, we find that the unnecessary divisions are well controlled. Thus, increasing the number of adopting relations seems to have a similar effect to the restriction on the division that generates a small cluster. However, even in the best case, we cannot altogether conquer the unnecessary divisions, but the division of Cluster 1 and Cluster 4 is meaningful. In addition, it is difficult to find meaning in the division of Cluster 2 and Cluster 3, but we can say that the terms of Cluster 2 mean a particular type of aircraft and

[4] In contrast, this causes the unnecessary division.

the terms of Cluster 3 mean aircraft, but they do not identify a particular type of aircraft.

4 FUTURE WORK

There are some problems to be solved in our method. In this section, we explain these problems and suggest some possible solutions.

1. *Problem of The Filtering*

   Text in ASRS seems more difficult to analyze than other text such as that in a newspaper, and there are more errors associated with POS tagging. The POS of a word is assumed to be that which appears most frequently with that word in the input texts. For example, the word 'evasive' appears 2816 times in the input data in our experiments, and it is analyzed as an adjective 2805 times and as other POS 11 times, hence this word is treated as an adjective. In another example, the term 'evasive maneuver' appears 270 times, with the word 'maneuver' within this term analyzed as a noun 268 times, and hence the term 'evasive maneuver' is treated as a noun.

   However, this procedure may eliminate valuable relations between the terms composed of one word, for example, the relation between the term 'rting' and 'routing', because the term 'rting' is not treated as a noun but as a verb.

   To solve this problem, we should change the method used for determining a term's POS. When we calculate the similarity measure between 'rting' and 'routing', the POS of 'rting' is not fixed by the most frequent POS of 'rting' in input text, but rather the most frequent POS in $Phrs$('rting'). In this example, the term 'rting' appears 364 times in ASRS reports, and it is analyzed as verb 230 times, thus, we regard it as verb. However there are 25 phrases in $Phrs$('rting'), and it is analyzed as a noun in most of these phrases, thus, we can regard it as noun.

2. *Problem of Hard Clustering*

   Our clustering method is a hard clustering method, where each term must be an element of only one cluster. Where a term is a word that has multiple meanings, a multisense word, our hard clustering method may delete the correct relations associated with the term. To examine this in further detail, let us look at the connected graph G, where the subgraphs of G are G_1 and G_2 are connected by only the one collect relation between the term A which is a multisense word and B, where A is a node of G_1, B is a node of G_2. If G is divided into G_1 and G_2, the correct relation between A and B is eliminated.

   To solve this problem, we may alter our method of making subgraphs. In the example above, we make the subgraphs G_1 and $\tilde{G_2}$ from G, where $\tilde{G_2}$ is a graph G_2 and node A. However, this procedure needs to have a means of detecting multisense words, which seems very difficult.

3. *Problem of Parameter Setting*

   We have shown that the results of clustering depend on the number of relations adopted for generating clusters. However, we have not developed a method for determining the way to find the best value of this parameter.

   To solve this problem, We should develop innovative techniques, but this seems to be a very difficult to overcome.

5 RELATED WORK

Graph-based clustering methods have been proposed for a variety of applications, for example, data mining in bioinformatics[4], or pattern recognition[5], and in document or word clustering.

Taking the example of document clustering, [6] presented an idea of modeling the document collection as a bipartite graph between documents and words, and a spectral algorithm which simultaneously yields a clustering of document and words using this model.

Graph-based clustering methods handle the clustering problem as a graph partitioning problem. For example, [7] reported a method which generates Japanese and English bilingual keyword clusters. This method produces a graph from the bilingual keywords pairs extracted from the bilingual corpus first, to detect a possible correspondence error in analyzing the graph, and to partition the two nodes of the possible error using 'the minimum edge cut' method. The idea of this study is very similar to ours, but this method cannot be applied to our data, because this method relies on the fact that the keyword in the bilingual keyword pair is a translation of the other keyword.

Other methods for word clustering have also been proposed due to their popularity for use in lexical knowledge acquisition. In these methods, various semantic similarity measures between terms have been proposed. Typically, they can be classified into two types: one type is based on the taxonomical relationships, and the other is based on distributional evidence. The former estimates the similarity with respect to hierarchically structed lexical resources. For our purpose, we were unable to use this approach as we could not use such resources. In this latter approach, the semantic similarity measure is often defined as the distance between two vectors. To calculate this, each target term T_i is characterized by a vector $C(T_i)$[5]. In our experiment, we adopted the "n-gram" approach to make $C(T_i)$ from T_i. In this approach, $C(T_i)$ can be regarded as the distribution of words which appear near the target term in the corpus. Our similarity measure looks something like this approach on the basis that the words that appear near the target term are focused on. As another example, [8] proposed the measure based on "verb - object " relations found in a corpus. In this paper, the target noun is characterized by the distribution of verbs for which it serves as direct object in the corpus. This measure seems well suited to our method, but it requires syntax analysis. In our particular case, in which we must analyze text containing many abbreviations and technical terms, its accuracy may be problematic.

6 CONCLUSION

We have introduced a new method for word clustering. The first step of our procedure is to extract pairs of words which belong to the same word class using our similarity measure. The second step of our procedure is to build the pairs of words into the connected graphs and to divide these into subgraphs depending on the estimation of edges in the subgraphs; We have shown that our method gives plausible results.

REFERENCES

[1] G.Ngai, R.Florian: Transformation-Based Learning in Fast Lane, Proceedings of NAACL-2001, pp 40-47,2001

[2] P.Broun et al.: Word-Sense Disambiguation using Statistical Methods, Proceedings of ACL-91, pp 264-270,1991

[3] M.Halkidi, M.Vazirgiannis: An Introduction to Quality Assessment in Data Mining, ECML/PKDD-2002 Tutorial, 2002

[4] H.Kawaji, Y.Takenaka, H.Matsuda: Graph-based clustering for finding distant relationships in a large set of protein sequences, Bioinformatics, Vol.20 No.2, pp243-252, 2004

[5] E.R.Hancock, M.Vento: Graph Based Representations in Pattern Recognition, Lecture Notes in Computer Science 2726, 2003

[6] I.S.Dhillon: Co-clustering documents and word using Bipartite Spectral Graph Partitioning, SIGKDD-01, pp 269-274, 2001

[7] A.Aizawa, K.Kageura: An Approach to the Automatic Generation of Multilingual Keyword Clusters, Proceedings of COMPTERM98, pp 8-14,1998

[8] F.Pereira, N.Tishby: Distributional Similarity, Phase Transitions and Hierarchical Clustering, Working Notes, Fall Symposium Series, AAAI, pp108-112, 1992

[5] This vector is often called a "characteristic vector" or a "feature vector".