Clustering Co-occurrence Graph based on Transitivity

Kumiko TANAKA-Ishii
Electrotechnical Laboratory
1-1-4 Umezono, Tsukuba, Ibaragi 305 JAPAN.
kumiko@etl.go.jp

Hideya IWASAKI
Tokyo Univ. of Agriculture and Technology
2-24-16 Naka-cho, Koganei, Tokyo 184 JAPAN.
iwasaki@ipl.ei.tuat.ac.jp

Abstract

Word co-occurrences form a graph, regarding words as nodes and co-occurrence relations as branches. Thus, a co-occurrence graph can be constructed by co-occurrence relations in a corpus. This paper discusses a clustering method of the co-occurrence graph, the decomposition of the graph, from a graph-theoretical viewpoint. Since one of the applications for the clustering results is the ambiguity resolution, each output cluster is expected to have no ambiguity and be specialized in a single topic. We observed that a graph has no ambiguity if its branches representing co-occurrence relations are transitive. An algorithm to extract such graphs are proposed and its uniqueness of the output is discussed. The effectiveness of our method is examined by an experiment using co-occurrence graph obtained from a 30M bytes corpus.

1 Introduction

Clustering is the operation to group words by some criterion. Thesauri and synonym dictionaries are some of its manual examples. Automatic outputs can be used not only to revise them, but also to aid ambiguity resolution, an essential problem in natural language processing. For instance, the meaning of an ambiguous word can be decided by examining the cluster it belongs to. Furthermore, clusters grouped according to topics have many application areas such as automatic document classification. The input in this paper is the word co-occurrence graph obtained from corpus. The output is its subgraphs with the condition that each subgraph is specialized in a topic.

Many automatic clustering methods have been already proposed. Most of them are based on the statistical similarity between two words. Our approach is different; it is graph theoretical. We tried to find out the special structure in linguistic graph.

Having a huge co-occurrence graph obtained from a corpus, we first tried to decompose it to analyze its graph structure using graph theoretical tools, such as maximum strongly connected components, or biconnected components. Although both tools decompose a graph into tightly connected subgraphs, these trials resulted in vain. The question arose; what must be taken into account to decompose the co-occurrence graph? The answer is the ambiguity. Furthermore, we reached to the conclusion that the ambiguity can be explained in terms of intransitivity. This feature is developed into an algorithm for clustering.

This paper is organized as follows. The following chapter describes the relationship between the transitivity in the graph and the ambiguity resolution. Chapter 3 shows the relationships between clustering and transitivity. Chapter 4 proposes and discusses an algorithm for clustering. Related work is resumed in Chapter 5. Our method is examined in Chapter 6 by some experiments.

2 Word Ambiguity and Transitivity

Two words are said to co-occur when they frequently appear close to each other within texts. Regarding words as nodes and co-occurring re-

lations as branches, a graph can be constructed from a given corpus. We define such a graph as **co-occurrence graph**.

When a portion of a corpus specializes in a topic, we can still extract a co-occurrence graph from the portion. A general corpus, such as newspaper corpus, contains many corpus portions, each specializing in one topic. Therefore, the whole co-occurrence graph obtained from a general corpus contains subgraphs, each specializing in one topic. Our question is to extract such subgraphs of topics from a co-occurrence graph.

We denote V as the set of nodes (words), E as the set of branches (co-occurrence relations), $G=< V, E >$ as an input graph and $|V|$ as the number of nodes. English words referred as examples will be written in this font.

2.1 Transitivity in Co-occurrence Relation

The most basic mathematical laws discussed about relations between elements in a set are reflective, symmetric and transitive laws. Having $a, b, c \in V$ and R as a relation, they can be described as follows:

Reflective aRa.
Symmetric $aRb \rightarrow bRa$.
Transitive $aRb,\ bRc \rightarrow aRc$.

Let V be word set and R be co-occurrence relation. When each property holds for R, words a, b and c can be explained as follows from the linguistic viewpoint:

Reflective Word a co-occurs with itself.

Symmetric Co-occurrence relation does not depend on the occurrence order.

Transitive Word b does not have two-sided meanings (ambiguity). For instance, doctor, which has both medical and academic meanings, co-occurs with nurse within a medical topic, and co-occurs with professor within an academic topic. However, nurse and professor do not co-occur, so the transitivity between nurse, doctor and professor does not hold.

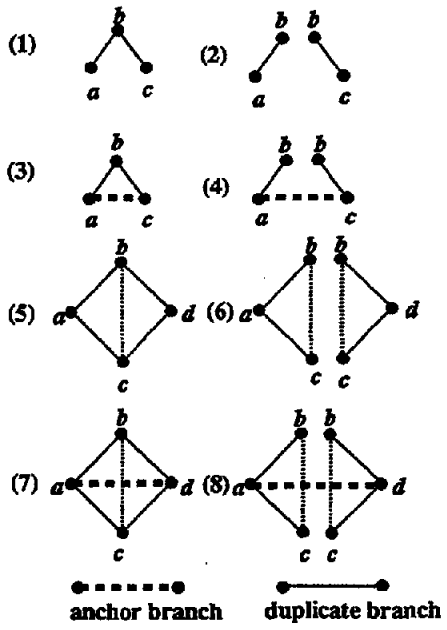

Figure 1: Graph Decomposition

Our request is to extract subgraphs each of which focuses on one topic with no ambiguity. Therefore, we perform clustering by extracting subgraphs whose branches form transitive co-occurrence relations.

3 Transitivity and Clustering

3.1 Decomposition and Duplication

The simplest case is a graph of three nodes. Figure 1-(1) is a graph in which the transitivity does not hold. For example, when b is doctor, a is nurse and c is professor, nurse and professor do not co-occur due to the node b's two-sided meanings. Therefore, as in Figure 1-(2), we "duplicate" b so that each duplicated node corresponds only to a single meaning. Then the ambiguity within b is resolved and the entire graph is divided into two subgraphs, the academic one and the medical one. To sum up, when the transitivity does not hold, a graph can be decomposed by duplicating the ambiguous node.

On the other hand, when the transitivity holds among three nodes (Figure 1-(3)), the

graph cannot be decomposed by duplication of b (Figure 1-(4)). This can be explained that the graph does not have the ambiguity.

We extend the above into the case of four nodes(Figure 1-(5)). Here, transitivity does not hold in a-b-d because there is no branch between a-d. When b–c is duplicated, the graph can be decomposed into two subgraphs (Figure 1-(6)) in which the transitivity holds. On the contrary, Figure 1-(7) cannot be decomposed by duplicating b–c due to the branch a–d (Figure 1-(8)); this shows that b–c is not ambiguous. Note that Figure 1-(7) is a complete graph of 4 nodes. We define **duplicate branch** as a branch to be duplicated for graph decomposition (such as b–c) and **anchor branch** as a branch which inhibit graph decomposition by duplication (such as a–d).

In general, when a graph could not be separated by duplicating its subgraph, then the subgraph is regarded not to have ambiguity. Therefore, ideal clustering is to decompose graphs into subgraphs which cannot be decomposed further by duplication. Unfortunately, this constraint is too strict because such a graph is restricted to a complete graph. In addition, extracting complete graphs within a given graph is NP-complete. Therefore we discuss in the following how to loosen the constraint.

3.2 Transitive Graph

There are two methods to loosen the constraint.

The first is to decrease the number of anchor branches. In the complete graph of more than 5 nodes, several anchor branches exist for each duplicate branch (Figure 2-(1)). However, only one anchor branch is sufficient to inhibit the decomposition. The less the number of anchor branch is, the looser the constraint is.

This intuitively corresponds to loosen the sharpness of the focus of the topic in the resulting cluster. For instance, two words pneumonia and cancer do not always co-occur, but they do co-occur with words as doctor, nurse and hospital forming the core of medical topics.

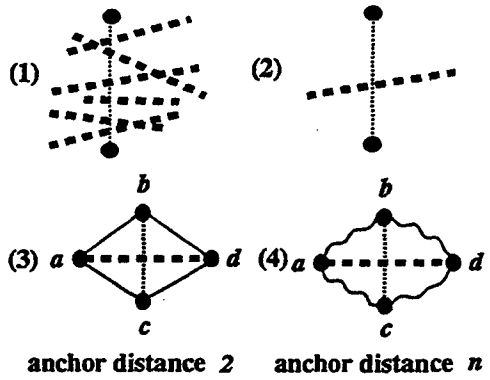

Figure 2: Loose Constraint for Graph Decomposition

Pneumonia will be included into a cluster if it is connected with these three words even if it is not connected with cancer. If cancer is also connected with these three words, both cancer and pneumonia, the different subtopical words within a medical topic are included in a cluster.

The second is to loosen the transitivity itself. It was defined in Section 2.1 within three nodes. We may prepare a loose transitivity as follows:

$$v_1 R v_2, \cdots, v_{n-1} R v_n \rightarrow v_1 R v_n$$

We define **anchor distance** as the maximum distance of the minimum distances of a–b–d and a–c–d. For example, when minimum distance of a–b–d is 4 and that of a–c–d is 6 then the anchor distance is 6. The tightest constraint is when anchor distance is 2 as in Figure 2-(3). This also blurs the topic focus of a cluster. In the example of pneumonia, the word will be included if it is connected directly with at least one of the words among doctor, hospital, nurse, and cancer, and connected indirectly with the others.

For $m, n \leq |V| - 1$, G is called (m,n)-**transitive graph** when

> for all $e \in E$, there are m anchor branches $e' \in E$ of anchor distance $\leq n$.

(m,n)-transitive graphs can be extracted as the subgraphs of the input graph. Figure 3 shows a map of (m,n)-transitive graphs. The axis of ordinates describes the number of anchor branches (m). The axis of abscissas de-

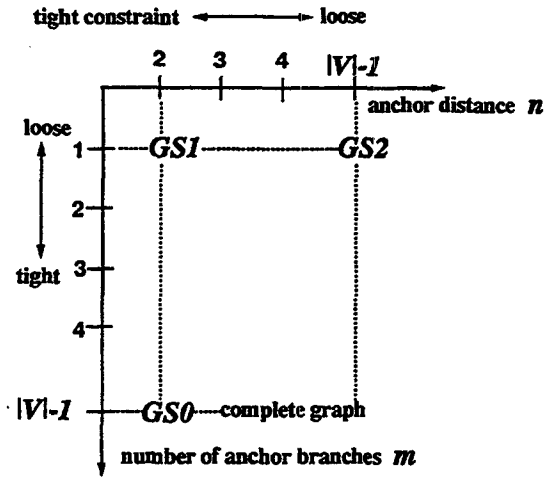

Figure 3: (m, n)-Transitive Graph

scribes the anchor distance (n). The constraint is loose when m is small and n is large. For the same input, (m_1, n)–transitive graphs are included in (m_2, n)–transitive graphs when $m_1 < m_2$, and (m, n_1)–transitive graphs are included in (m, n_2)–transitive graphs when $n_2 < n_1$.

$GS2$ in Figure 3 is the clusters obtained under the loosest constraint: m is the maximum and n is the minimum. In $GS2$, all ambiguity of a branch and nodes at its ends are resolved. $GS0$ are the transitive graphs of the tightest constraint. All transitive graphs on the horizontal line including $GS0$ are complete.

4 Extraction of Transitive Graphs

So far, we did not explain how to detect the duplicate and anchor branches, given a graph. An algorithm for clustering can be top-down or bottom-up. The former gives clusters by decomposing the input graph by detecting duplicate branches.

Although we have explained our clustering method top-down up to now, we propose our clustering method as bottom-up. We obtain clusters by accumulating adjacent nodes so that every branch has anchor branches and the resulting clusters include no duplicate branch. Thus, in the bottom-up method, we need *not* detect duplicate branches. This is convenient, because the condition for anchor and duplicate branches is denoted by local relationships among nodes.

The branches in the input graph are assumed to be all symmetric. In this section, we use terms **clusters** as our output and **subgraphs** as their candidates.

4.1 Definition of Clusters

We extract $GS1$, the $(1, 2)$–transitive graph.

A subgraph A including a branch e in the input graph can be extracted as follows:

Step 1. Put a triangle graph including e into A.

Step 2. Take a branch e' in A and a node v which makes a triangle with e' (Figure 4). If the following condition is satisfied, put v into A.

> There exists a node $v' \in G$ (input graph) whose distance from e' is 1, and it is connected to v with a branch. Here, the branch v'–v is the anchor branch so that e' is hindered to be the duplicate branch in the resulting cluster.

Additionally, put every branch between $v'' \in G$ and v into A.

Step 3. Repeat Step 2 until A cannot be extended.

Performing the above procedure starting from every branch in the input graph, we obtain many subgraphs. Considering the inclusion relation between subgraphs, they constitute a partial order (Figure 5). We define clusters as maximal subgraphs in this partial order chain. They are subgraphs not included in any other subgraphs. The uniqueness of the clusters for an input is self-evident.

4.2 Algorithm for Clustering

In the previous section, the procedure to obtain subgraphs should begin from every branch in the input. However, it is sufficient to calculate as follows.

Step 0. $i = 0$

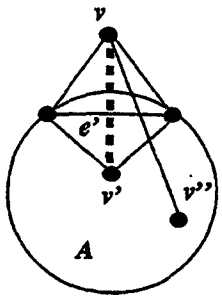

Figure 4: Extraction of (1,2)–Transitive Graph

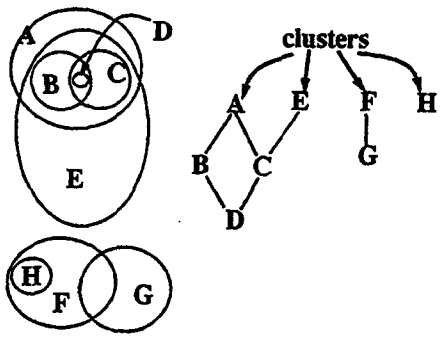

Figure 5: Subgraphs and their Partial Order

Step 1. Choose a branch $e \in G$ not included in $G_0, \cdots, G_{i-1}$. If no e is found, go to Step 5. $G_i = < \emptyset, \emptyset >$. Put a triangle graph including e into G_i.

Step 2 and 3. Extend G_i using Step 2 and 3 of the previous section.

Step 4. Set $i = i + 1$ and goto Step 1.

Step 5. Examine every pair of subgraphs (A, B), and if A includes B, then drop B. The remaining subgraphs are defined as clusters.

A maximal subgraph cannot be missed. Its starting branch is encountered without fail in the above algorithm. If it is encountered as the starting branch in Step 1, the maximal subgraph is obtained. If it is captured into a subgraph and becomes e' in Step 2, the subgraph extends to the size of maximal subgraph; if it gets larger, the subgraph contradicts being maximal as the result of the last section.

The algorithm halts since the input graph is finite, and the output is unique for an input.

5 Related Work

[Li and Abe, 1996] compared clustering methods proposed so far [Hindle, 1990] [Brown *et al.*, 1992] [Pereira *et al.*, 1993] [Tokunaga *et al.*, 1995][Li and Abe, 1996].

Most of them are so-called *hard clustering*: each word is included only in one cluster. We do not follow the trend, from the sense that our objective is the extraction of clusters of topics. It is natural that an ambiguous word should be included in different clusters.

[Pereira *et al.*, 1993] adopts *soft clustering*. They measured co-occurrence between nouns and verbs, and clustered nouns of the same distribution of verbs.

[Fukumoto and Tsujii, 1994]'s work has common motivation with us: the ambiguity should be resolved for clustering. They clustered verbs using the gravity of multivariate analysis.

[Sugihara, 1995]'s approach has a common point in that it focuses on graph structure for clustering and tries to structurize the input graph, a bipartite graph of words and concepts (such as food, fruit etc.). His clustering method is so called Dulmage-Mendelsohn decomposition in graph theory. The output naturally gives a partial order of clusters which can be compared with conventional thesauri.

Our input is not bipartite. In the beginning, we tried to decompose input graph into maximum strongly connected components to obtain graphs of topics from the observation that nodes in a cycles are strongly related[1]. However, subgraphs about different topics is merged into the same cluster by two ambiguous words which bridge these two subgraphs(Figure 6). Next, we observed that articulation nodes are ambiguous, so we performed decomposition into biconnected components. In this case when several biconnected components are connected in a ring, articulation nodes could not be detected (Figure 7). The observation that there are no co-occurrence relationship between

[1][Tokunaga and Tanaka, 1990] discusses on extraction of cycles formed by translation relations from bilingual dictionary.

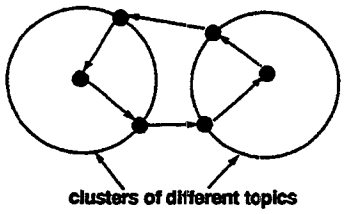

Figure 6: Problematic Structure in MSCC Clustering

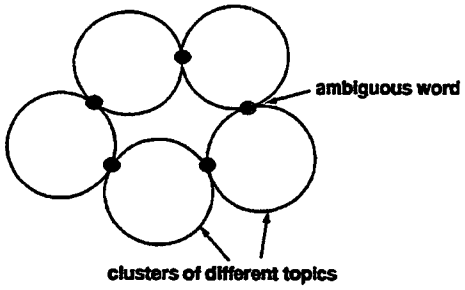

Figure 7: Problematic Structure in Biconnected Component Clustering

two biconnected graph across the articulation node was the start point of this paper.

6 Experiments

6.1 Procedure of Clustering

First, we make the input graph from a 30M bytes of Wall Street Journal. Co-occurrences of nouns and verbs are extracted by a morphological analyzer. We defined that a word co-occurs with 5 words ahead of the word within a sentence. Co-occurrence degree is measured by mutual information[Church and Hanks, 1990]. We set a certain threshold to the values to extract the input graph.

The number of resulting clusters depends on the input graph as follows:

1 One huge connected subgraph.
2 Several medium sized subgraphs.
3 Many small sized subgraphs.

When the threshold value is too high, the output is 3. On the contrary, when it is too low, then the output becomes 1. Both 1 and 3 are not interesting, because 1 is a graph including all topics, and 3 generates graphs of too small topics to check the global trend of topics in the input. Therefore, we varied threshold from 1.0 up to 7.0 by 0.5 steps to make the input graph, applied our algorithm to each input in order to detect the best threshold.

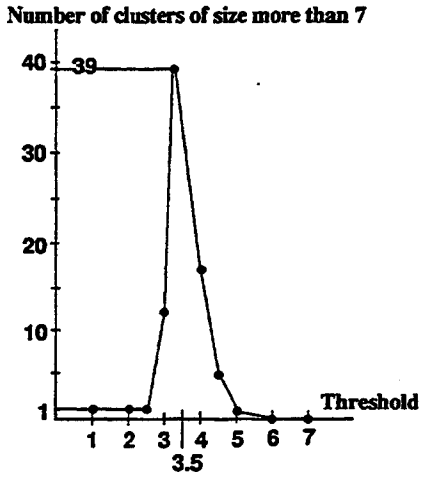

Figure 8: Threshold and the Output

The result is shown in Figure 8. The number of clusters whose sizes are more than 7 is plotted against the threshold value. When the threshold is 3.5, such clusters were most numerous. In 39 clusters, there were 727 different words out of 15347 in the input graph.

6.2 Evaluation of Clusters

In Appendix, 39 clusters are shown their contents sorted by size. Words judged inappropriate in each cluster are attached "†". Words undecidable being suitable in their clusters are put "‡".

All 39 clusters are attached four items as follows:

- Subjectively judged topic
- Cluster size (CS): number of words in a cluster
- Error rate (ER): rate of inappropriate words (attached "†")
- Uncertainty rate (UR): rate of uncertain words (attached "†" and "‡")

The average of the above items were CS=20,

ER=10.3%, UR= 14.7%; hence, the ambiguity was removed from clusters up to 85% on average. The number of the cluster whose topic was inestimable is only 1. The estimation of topic becomes difficult with two factors, CS and UR. When CS is too small, even when UR is 0.0, the cluster itself lacks in information. When UR is high, it is natural that the topic becomes inestimable.

6.3 Evaluation of Words Contained in More than Two Clusters

The number of words belonging to more than two clusters amounts to 57. They are classified as follows (numbers in parenthesis are cluster number in Appendix):

1. A word with different meanings (10 words).

cell	(1, 15)	ice	(3, 8)
panel	(1, 7)	treat	(1, 3)

2. A word with the same meaning but used in different contexts (32 words).

children	(6, 24)	music	(12,23)
star	(9,12,14)	brand	(3,22)

3. A word with the same meaning in the same context (7 words).
4. Others (One of the words is uncertain, or its cluster's context is not estimated. 6 words)

Words of class 1 is the ambiguous words. Cell in Cluster 1 means cells of tissue, whereas that in Cluster 15 means battery. Ice in Cluster 3 means ice for cooling beverage, whereas that in Cluster 8 means ice to skate on.

According to our objectives to obtain subgraphs of topics, words in class 2 is quite important to be duplicated. For instance, star in Cluster 9 is a sport player star, that in Cluster 12 is a singer star and that in Cluster 14 is a movie star. If star were not duplicated, the three different topics would be merged into a single subgraph. The same situation is observed for children: it would merge topics of childbirth and education into a graph if it was not duplicated. We are apt to pay attention only to the words of class 1, but that of class 2 plays an important role in clustering.

Words in class 3 is not ambiguous: they should connect two subgraphs into one (see Section7).

6.4 Cluster Hierarchy

An output subgraph of higher threshold is included as that of lower threshold. With this inclusion relation, the clusters form a hierarchy(Figure 9). A part of the hierarchy is shown below:

Threshold 3.75

A admission college scholarship
B admission applicant college
C campus children classroom college education enroll faculty grade math parent scholarship school student sugar teach teacher tuition tutor university voucher
D birth child children couple marriage marry parent wedlock woman
E birth infant weight
F birth boy marry

Threshold 3.5 Cluster 6,24 in Appendix.

Threshold 3.25

G admission applicant baby birth boy campus century child children classroom college couple daughter education endowment enroll enrollment establishment faculty father gift girl god grade homework husband infant ivy kid live love man marriage marry math mother oxford parent professor psychologist scholar scholarship school son student study sugar taught teach teacher teaching toy tuition tutor university voucher wed wedlock woman

At threshold 3.75, the origins of **education (Cluster 6)** and **childbirth(Cluster 24)** clusters are already formed. Among **education**, there are subtopics on scholarship school and school entrance. They are merged into Cluster 6 when the threshold is lowered to 3.5. Cluster 24 is also formed from Clusters D,E,F of threshold 3.75. Then Cluster 6 and 24 are merged into Cluster G when the threshold is

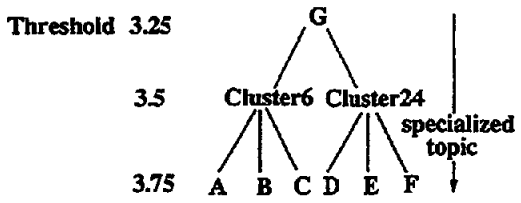

Figure 9: Cluster Hierarchy

lowered again to 3.25. The clusters' hierarchical relationships are shown in Figure 9.

We may see that the topic is more specialized when the threshold is high. Clusters which are merged between threshold 3.75 and 3.5 were those within a topic (A,B,C or D,E,F), but topics of different clusters are merged at 3.25 (Clusters 6 and 24). Thus, the lower the threshold is, the more the cluster contains ambiguity. The reason is that the two words in different topics do not co-occur.

7 Discussion

The best threshold differs in topics. Some examples are:

Economic topic: Although Wall Street Journal articles have economic tendency, clusters of economic topic cannot be found in the clusters with more than 7 words of threshold 3.5. They appear in clusters at threshold 3.0 as follows:

- accountant audit bracket deduction filer income offset tax taxpayer
- convert conversion debenture debt holder outstand prefer redeem redemption repay tidewater

The threshold should be lower for this topic.

Medical topic: Cluster 1 have too many words. Despite of a medical cluster, potato appears in the cluster. At the threshold 3.0, Cluster 3 is completely merged with Cluster 1. The appearance of potato shows the sign that the merge of two different topic has already begun. Therefore, a higher threshold is preferred.

Trial topic: Several clusters exist on trial in Appendix. They should form a cluster with relatively lower threshold.

Consequently, one of the most important future work is to integrate two stages, the first stage of making input graph with the static threshold, and the second stage of clustering, into a single stage with dynamic threshold.

8 Conclusion

We have discussed a method to cluster a co-occurrence graph obtained from a corpus, from a graph-theoretical viewpoint. A graph has no ambiguity if its branches, co-occurrence relations are transitive. This graph theoretical approach using graph is characteristic and it differs from the conventional clustering method. We proposed an algorithm to extract subgraphs whose branches are transitive co-occurrence relations and discussed its features. The effectiveness of our method was examined using the 30M corpus.

References

[Li and Abe, 1996] H. Li and N. Abe. Clustering Words with the MDL Principle. In *Proceedings of the 15th International Conference on Computational Linguistics,* vol1. pp.4–10, 1996.

[Sugihara, 1995] K. Sugihara. *A Graph–Theoretical Method For Monitoring Concept Formation.* Pattern Recognition, Vol.28, No.11, pp. 1635–1643, 1995.

[Tokunaga *et al.*, 1995] T. Tokunaga, M. Iwayama and H. Tanaka. Automatic Thesaurus Construction based on Grammatical Relations. In *Proceedings of the International Joint Conference on Artificial Intelligence '95*, 1995.

[Fukumoto and Tsujii, 1994] F. Fukumoto and J. Tsujii. Automatic Recognition of Verbal Polysemy. In *Proceedings of the 14th International Conference on Computational Linguistics,* vol.2, pp.762–768, 1994.

[Pereira *et al.*, 1993] F. Pereira, N. Tishby and L. Lee. Distributional Clustering of En-

glish Words. In *Proceedings of the 31st ACL*, pp. 183–190, 1993.

[Brown *et al.*, 1992] P. Brown, V. Pietra, et al. Class-based n-gram Models of Natural Language. *Computational Linguistics*, 18 (4), pp. 283–298, 1992.

[Tokunaga and Tanaka, 1990] Tokunaga, T. and Tanaka, H. The Automatic Extraction of Conceptual Items from Bilingual Dictionaries. *PRICAI*, 1990.

[Hindle, 1990] D. Hindle. Noun Classification from Predicate — Argument Structures. In *Proceedings of the 28th ACL*, pp.168–175, 1990.

[Church and Hanks, 1990] K. W. Church and P. Hanks. Word Association Norms, Mutual Information, and Lexicography. *Computational Linguistics*, vol. 16 (1), pp. 22–29, 1990.

Appendix

The triples are (CS, ER, UR) (See Section 6.2).

Cluster 1: medicine (105, 4.8%, 5.7%)
afflict cancer disease gene researcher cell hepatitis lung patient bacterium protein repair scientist therapy mice tissue virus fibrositis implant mouse vaccine antigen nicotine treat infect receptor switch† blood enzyme inject insulin molecule pill aid heart infection transplant prostate stroke symptom transmit treatment cure depression diagnose kidney trial† bone chemotherapy dose marrow ulcer cause stomach doctor breast disorder prescribe blocker‡ clot eradicate medication sample brain nerve potato† donate plasma patch arthritis drug lithium placebo version† test laboratory protease prevention syndrome panel schizophrenia substance tumor psychotherapy sclerosis suffer die pain physician lab urine rat radiation animal cholesterol pharmacist pharmacy collagen internist hospital prescription medicine care referral†nonprescription

Cluster 2: transportation (101, 12.6%,14.6%)
air surveillance missile† plane fighter† radar aircraft patriot†transport airline jet tank boat pilot serb† trainer airliner crash engine flight fly aviation passenger bus carrier bump airport hub saudi airplane delivery surface‡ machinist attendant wing airway shuttle nonstop route delta fare hanoi† walkout mechanic transportation haul denver railroad courier rail freight mile truck shipment diesel pickup minivan gasoline fuel sport† built† midsize† assembly heat inventory detroit vehicle model oil petroleum car auto styling† emission‡ brake cylinder sedan equip antilock omega airbag luxury bag driver jaguar dodge† accident pump motor rover wheel neon† ford cherokee† rear front dealership†explorer plant build shreveport

Cluster 3: meal (60, 8.3%, 11.7%)
alcohol beer brew label liquor miller beverage taste bottle brewer brewery ale ice brand vodka coca cola wine can drink milk coke diet fruit juice dinner draft dairy tea nestle supermarket grocer toronto† bottler crown† flavor lithuania† atlanta† pepper cranberry infection‡ soup cheese cow hormone meat refrigerate calorie category spray† eat ocean‡ meal chicken variety treat food sauce beef restaurant

Cluster 4: agriculture (32, 3.1%, 15.6%)
acre flood corn farmer grain harvest agriculture hog soybean wheat bushel crop depress† exporter feed livestock cotton rain africa† cattle farm meat brazil‡ july rice shipment forecast‡ bale frost season flake‡ sugar

Cluster 5: Near East Asia (32, 0.0%, 0.0%)
arab qatar saudi egypt kuwait israel king peace arabia bahrain gaza emirate oil jew jordan lebanon occupy fighter jerusalem syria palestine barrel settler massacre territory kill strip cabinet liberation jericho

Cluster 6: education (31, 3.2%, 12.9%)
admission scholarship school college applicant student university education professor taught teach tuition classroom faculty girl† teacher enroll enrollment grade homework math parent campus psychologist‡ scholar ivy voucher‡ children tutor sugar† endowment

Cluster 7: politics (25, 24.0, 40.0%)
bidden† senate senator democrat nomination bipartisan vermont‡ chafe† dole inman† maine committee subcommittee confirmation rhode† bob† columnist‡ judiciary‡ ranking chairmen panel chairman hearing oversight† oregon‡

Cluster 8: winter sports (24, 4.2%, 12.5%)
alpine race ski cup norway winter medal skate skier boat‡ cross kilometer sport silver snow tommy† men‡ athlete skater boot ice meter speed track

Cluster 9: ball games (24, 16.7%, 16.7%)
baseball sports basketball fan football league liberty† team pro soccer televise coach stadium star fox† cowboy† bowl conference† franchise game hockey player tournament college

Cluster 10: computer network (18, 5.6%, 5.6%)
bulletin user prodigy† computer message internet mail pager laptop facsimile machine desktop modem notebook dell subscription communicate subscriber

Cluster 11: Asia (17, 23.5%, 23.5%)
asia singapore thailand indonesia malaysia philippines vietnam taiwan korea burma china interbank†neighbor visitor† pakistan status† human† beijing

Cluster 12: song (17, 0.0%, 0.0%)
album pop sing song music radio singer jackson artist recording disk band channel jazz star copy concert

Cluster 13: Soviet Union (17, 0.0%, 5.9%)
arsenal russia soviet ukraine weapon dismantle moscow baltic ruble bloc warhead kiev parliament reformer cabinet uranium quit‡
Cluster 14: movie (17, 5.9%, 5.9%)
academy hollywood film picture studio movie polygram actor library video turner lansing†screen script actress star theater
Cluster 15: satellite broadcast (17, 5.9%, 11.8%)
affiliate station broadcast fox‡ network radio television clutter† antenna satellite channel beam signal transmit format dish cell
Cluster 16: software (16, 0.0%, 0.0%)
apple version window mac macintosh user application software unix lotus spreadsheet developer copyright excel feature word
Cluster 17: Europe (15, 0.0%, 0.0%)
austria belgium france italy netherlands sweden finland spain britain denmark holland switzerland norway kingdom membership
Cluster 18: petrolum (14, 14.3%, 28.6%)
alberta‡ energy calgary† gas oil pipeline feet barrel basin‡ refine chevron† exploration petroleum condensate
Cluster 19: art (13, 15.4%, 15.4%)
art works exhibit exhibition paint collection gallery museum photograph landscape artist avenue† floor†
Cluster 20: Korea (13, 15.4%, 23.1%)
artillery tank weapon missile korea pyongyang regime seoul iraq† patriot inspector scud‡ stance†
Cluster 21: Balkan Peninsula (12, 0.0%, 0.0%)
artillery serb strike weapon muslim bomb troop peace peacekeeper croat negotiator withdraw
Cluster 22: cigarette (11, 0.0%, 0.0%)
addict smoker cigarette nicotine smoke camel morris tobacco ban antismoke brand
Cluster 23: multimedia (11, 0.0%, 0.0%)
audio text video disk cassette library multimedia tape music videocassette entertainment
Cluster 24: childbirth (11, 0.0%, 0.0%)
birth wedlock child marriage marry mother parent woman couple children infant
Cluster 25: calamity (11, 9.1%, 9.1%)
crop relief disaster flood earthquake riot† hurricane quake rebuilding victim francisco
Cluster 26: goods (10, 10.0%, 10.0%)
apparel retailer store furnishing mart merchandise warehouse outlet mass† chain
Cluster 27: trial (10, 20.0%, 20.0%)
appeal reinstate ruling court upheld circuit† arbitration judge overturn† arbitrator
Cluster 28: trash (10, 50.0%, 50.0%)
cleanup site† superfund† dump pollute insurer† waste trash ferris† glut†
Cluster 29: ? (10, 100.0%, 100.0%)
arkansas† rock† thrift† guaranty† madison† supervise† failure† limitation† surrounding† investigation†
Cluster 30: earthquake (9, 22.2%, 22.2%)
aftershock quake damage earthquake repair epicenter freeway† homeowner† inspect
Cluster 31: real estate (9, 0.0%, 11.1%)
apartment bedroom rent avenue manhattan tower tenant subsidize‡ landlord
Cluster 32: trial (9, 0.0%, 0.0%)
convict prison sentence jail parole conviction fine offender plead
Cluster 33: trial (9, 0.0%, 0.0%)
award damage jury plaintiff verdict defendant case juror convict
Cluster 34: winter resort (9, 0.0%, 0.0%)
crest resort ski skier mountain snow valley slope sky
Cluster 35: trial (8, 0.0%, 0.0%)
bureau probe subpoena investigation prosecutor counsel inquiry suicide
Cluster 36: telephone network (8, 0.0%, 0.0%)
cable wire fiber phone transmission voice fax microwave
Cluster 37: bond (8, 0.0%, 75.0%)
bondholder reorganization‡ chapter‡ creditor reorganize‡ petition‡ proceeding‡ filing‡
Cluster 38: book (8, 12.5%, 12.5%)
audio† publisher title book bestseller bookstore reference copy
Cluster 39 guerrilla (8, 12.5%, 12.5%) guerrilla syria negotiate peace rebel uprising negotiator mexico†