

Clustering Large Collection of Biomedical Literature based on Ontology-enriched Bipartite Graph Representation and Mutual Refinement Strategy*

Illhoi Yoo, and Xiaohua Hu

College of Information Science and Technology, Drexel University, Philadelphia, PA, 19104,
USA
iy28@drexel.edu, thu@cis.drexel.edu

Abstract. In this paper we introduce a novel document clustering approach that solves some major problems of traditional document clustering approaches. Instead of depending on traditional vector space model, this approach represents a set of documents as bipartite graphs using domain knowledge in ontology. In this representation, the concepts of the documents are classified according to their relationships with documents that are reflected on the bipartite graph. Using the concept groups, documents are clustered based on the concepts' contribution to each document. Through the mutual-refinement relationship with concept groups and document groups, the two groups are recursively refined. Our experimental results on MEDLINE articles show that our approach outperforms two leading document clustering algorithms: BiSecting K-means and CLUTO. In addition to its decent performance, our approach provides a meaningful explanation for each document cluster by identifying its most contributing concepts, thus helps users to understand and interpret documents and clustering results.

1 Introduction

Document clustering was initially investigated for improving information retrieval (IR) performance (i.e. precision and recall) because similar documents grouped by document clustering tend to be relevant to the same user queries [1] [2]. However, because document clustering was too slow or infeasible for very large document sets in early days, it was not widely used in IR systems [3]. As faster clustering algorithms have been introduced and those have been adopted in document clustering, document clustering has been recently used to facilitate nearest-neighbor search [4], to support an interactive document browsing paradigm [3] [5] [6], and to construct hierarchical topic structures [7]. Thus, as

* This research work is supported in part from the NSF Career grant (NSF IIS 0448023). NSF CCF 0514679 and the PA Dept of Health Tobacco Settlement Formula Grant (#240205, 240196).

information grows exponentially, document clustering plays a more important role for IR and text mining communities.

However, traditional document clustering approaches have four main problems. First, when the approaches represent documents based on the bag of word model, they use all words/terms in documents. As Wang et al pointed out [8], only a small number of words/terms in documents have distinguishable power on clustering documents. Words/terms with distinguishable power are normally the concepts in the domain related to the documents. Second, the approaches do not consider semantically related words/terms (e.g. synonyms or hyper/hyponyms). For instance, they treat {Cancer, Tumor, Neoplasm, Malignancy} as the different terms even though all these words have similar meaning. Third, the approaches cannot provide an explanation of why a document is grouped into one of document clusters [9] because they pursue similarity-based mechanism on clustering, which does not produce any models or rules for document clusters. Lastly, the approaches are based on vector space model. The use of vector space representation on document clustering causes the two main problems. The first problem is that the vector space model assumes all the dimensions on the space are considered independently. In other words, the model assumes that words/terms are mutually independent in a document. However, most words/terms in a document are related to each other. The second problem is that clustering in high dimensional space significantly hampers the similarity detection for objects (here, documents) because the distance between every pair of objects tends to the same regardless of data distributions and distance functions [10]. Thus, it dramatically decreases clustering performance.

These problems have motivated this study. In this paper, we introduce a novel document clustering approach that solves all the four problems stated above. The rest of the paper is organized as follows. Section 2 surveys the related work. In section 3, we propose a novel graph-based document clustering approach that uses domain knowledge in ontology. An extensive experimental evaluation on MEDLINE articles is conducted and the results are reported in section 4. Finally, we conclude the paper with the three main contributions and future work.

2 Related Work

Many document clustering approaches have been developed for several decades. Most of document clustering approaches are based on vector space representation and apply various clustering algorithms to the representation. To this end, the approaches can be categorized according to what kind of clustering algorithms are used. Thus, we classify the approaches into hierarchical and partitional [11].

Hierarchical agglomerative clustering algorithms were used for document clustering. The algorithms successively merge the most similar objects based on the pairwise distances between objects until a termination condition holds. Thus, the algorithms can be

classified by the way they pick the pair of objects for calculating the similarity measure; for example, single-link, complete-link, and average-link. Partitional clustering algorithms (especially K-means) are the most widely-used algorithms in document clustering [12]. Most of the algorithms first randomly select k centroids and then decompose the objects into k disjoint groups through iteratively relocating objects based on the similarity between the centroids and the objects. The clusters become optimal in terms of certain criterion functions.

There are some hybrid document clustering approaches that combine hierarchical and partitional clustering algorithms. For instance, Buckshot [3] is basically K-means but Buckshot uses average-link to set cluster centroids with the assumption that hierarchical clustering algorithms provide superior clustering quality to K-means. In order to create cluster centroids, Buckshot first picks \sqrt{kn} objects randomly and then uses average-link algorithm; to make the overall complexity linear, Buckshot selects \sqrt{kn} objects. However, as Larsen & Aone [13] pointed out that using hierarchical algorithm for centroids does not significantly improve the overall clustering quality, compared with the random selection of centroids.

Recently, Hotho et al. introduced the semantic document clustering approach that uses background knowledge [9]. The authors apply ontology during the construction of vector space representation by mapping terms in documents to ontology concepts and then aggregating concepts based on the concept hierarchy, which is called concept selection and aggregation (COSA). As a result of COSA, they resolve a synonym problem and introduce more general concepts on vector space to easily identify related topics [9]. Because they cannot reduce the dimensionality (i.e. the document features) on vector space, it still suffers from “*Curse of Dimensionality*”. In addition, COSA cannot reflect the relationships among the concepts on vector space due to the limitation of vector space model.

3 The Proposed Approach: COBRA

We present a novel approach for Clustering Ontology-enriched Bipartite Graph Representation, called COBRA. The proposed approach consists of three main steps: (1) bipartite graph representation for documents through concept mapping, (2) initial clustering by combining co-occurrence concepts based on their semantic similarities on concept hierarchy and document subsets that share co-occurrence concepts, and (3) mutual refinement strategy for concept groups and document clusters. Before discussing these three main components in detail we first briefly discuss Medical Subject Headings (MeSH) as a biomedical ontology due to its importance in our approach.

Medical Subject Headings (MeSH), published by the National Library of Medicine in 1954, mainly consists of the controlled vocabulary and MeSH Tree. The controlled vocabulary contains several different types of terms. Among them Descriptor and Entry terms are used in this research because only they can be used for graph representation.

Descriptor terms are main concepts or main headings. Entry terms are the synonyms or the related terms to descriptors. For example, “Neoplasms” as a descriptor has the following entry terms {“Cancer”, “Cancers”, “Neoplasm”, “Tumors”, “Tumor”, “Benign Neoplasms”, “Neoplasms, Benign”, “Benign Neoplasm”, “Neoplasm, Benign”}. MeSH descriptors are organized in MeSH Tree, which can be seen as MeSH Concept Hierarchy. In MeSH Tree there are 15 categories (e.g. category A for anatomic terms) and each category is further divided into subcategory. For each subcategory, corresponding descriptors are hierarchically arranged from most general to most specific. In fact, because descriptors normally appear in more than one place in the tree, they are represented in a graph rather than a tree. In addition to its ontology role, MeSH descriptors were originally used to index MEDLINE articles. For this purpose around 10 to 20 MeSH terms are manually assigned to each article (after reading full papers). On the assignment of MeSH terms to articles around 3 to 5 MeSH terms are set as “MajorTopic” which primarily represent an article.

3.1 Bipartite Graphical Representation for Documents through Concept Mapping

Every document clustering method first needs to convert documents into proper format (e.g. document*term matrix). Since we recognize documents as a set of concepts that have their complex internal semantic relationships and assume that documents could be clustered based on what concepts they contain, we represent a set of documents as a bipartite graph to indicate the relationships between concepts and documents on the graph.

This procedure takes the following three steps: concept mapping in documents, detection of co-occurrence concepts, and construction of bipartite graph representations with co-occurrence concepts. Firstly, it maps terms in each document into MeSH concepts. In order to reduce unnecessary search for MeSH concepts, it removes stop words from each document and generates three gram-words as the candidates of MeSH Entry terms. After matching the candidates with Entry terms it replaces Entry terms with Descriptor terms, which is called *concept aggregation*. Then it filters out some MeSH terms that are too general (e.g. HUMAN, WOMEN or MEN) or too common over MEDLINE articles (e.g. ENGLISH ABSTRACT or DOUBLE-BLIND METHOD); see [14] for details. We assume that those terms do not have distinguishable power on clustering documents.

In the second step, it finds out co-occurrence concepts from sets of concept pairs in each document based on the number of times they appear in documents. Co-occurrence terms have long been used in document retrieval systems to identify indexing terms during query expansion [15] [16]. We use co-occurrence concepts instead of concepts because co-occurrence concepts contain some semantic associations between concepts and thus they are regarded more important than single concept.

The remaining problem for co-occurrence concepts is how to set the threshold value for co-occurrence counts; concept pairs whose co-occurrence counts equal or bigger than the value are considered as co-occurrence concepts. Because the threshold value fairly

depends on documents or query to retrieve documents, we develop a simple algorithm to detect reasonable threshold value instead of just setting a fixed value. This algorithm tries to find bisecting point in one-dimensional data. It first sorts the data, takes as centroids the two end objects, and then assigns the remaining objects to the two centroids based on the distances with dynamic centroids update; because the data (co-occurrence counts) was already sorted, it does not need any iteration like other partitioning clustering algorithms. After obtaining the threshold value co-occurrence concepts are mirrored as edges on the graph and their co-occurrence counts are used as edge weights.

In the third step, it constructs a bipartite graph. Given the graph $G = (V_D + V_{CC}, E)$, V_D indicates a set of documents, V_{CC} represents a set of co-occurrence concepts in documents and E indicates the relationships between two vertices. Weights can be optionally specified on edges. In that case one should provide a sophisticated weight scheme to measure the contribution of concepts to each document. However, such a weight scheme may not be appropriate especially for small size of documents, such as Medline abstracts. In addition, the scheme requires $|V_D| * |V_C|$ complexity. Thus, we draw an unweighted bipartite graph.

3.2 Initial Clustering by Combining Co-Occurrence Concepts

Here, COBRA generates initial clusters for the next step by combining co-occurrence concepts. Since similar documents share the same or semantically similar co-occurrence concepts, COBRA combines co-occurrence concepts and then cluster documents based on their similarities to k co-occurrence concept groups. On combining them there are two ways to measure the similarity between co-occurrence concepts: their semantic similarity on the concept hierarchy (sim_{cc}) and the overlap of their document sets (sim_{doc}). We integrate the two measures with weights. Given two co-occurrence concepts (CC_i & CC_j), the similarity is defined as ($\lambda=0.5$ in the experiments)

$sim(CC_i, CC_j) = \lambda \cdot sim_{cc}(CC_i, CC_j) + (1 - \lambda) \cdot sim_{doc}(CC_i, CC_j)$, with $\lambda \in [0, 1]$ as weights

The semantic similarity between two co-occurrence concepts (CC_i & CC_j) on concept hierarchy (sim_{cc}) is the average similarity of four concept pairs. C^p indicates the set of parent concepts of C concept on the concept hierarchy. sim_{doc} is built on the information theoretic based measure [17]. It is defined as the ratio between the amount of information needed to state the commonality of co-occurrence concepts and the information needed to fully describe what the co-occurrence concepts are in terms of the number of relevant documents.

$$sim_{cc}(CC_i, CC_j) = \frac{\sum_{C_i \in CC_i, C_j \in CC_j} \frac{C_i^p \cap C_j^p}{C_i^p \cup C_j^p}}{|CC_i| + |CC_j|} \quad sim_{doc}(CC_i, CC_j) = \frac{|docs_{CC_i} \cap docs_{CC_j}|}{|docs_{CC_i} \cup docs_{CC_j}|}$$

, where $docs_{CC_i}$ implies a set of documents that contain CC_i co-occurrence concept.

Based on average-link clustering algorithm that uses the integrated similarity function, COBRA combines co-occurrence concepts until we get k co-occurrence concept groups. For initial document clusters COBRA links each document to k co-occurrence concept groups based on its similarity to k groups. This similarity is simply measured by the number of times co-occurrence concepts in each document appear in each of k groups. A document is assigned to the most similar co-occurrence concept group. For example, suppose there are two co-occurrence concept groups ($CCG_1=\{CC_1, CC_2, CC_3\}$, $CCG_2=\{CC_4, CC_5\}$) and a document has CC_2, CC_3 , and CC_5 . Then, the document is assigned to CCG_1 .

3.3 Mutual Refinement Strategy for Document Clustering

Through the procedures above COBRA generates initial clusters. However, this clustering cannot correct erroneous decisions like hierarchical clustering methods. In other words, once clustering procedures are performed, the clustering results are never refined further even if the procedures are based on local optimization.

In this procedure COBRA “purifies” the initial document clusters by mutually refining k co-occurrence concept groups and k document clusters. The basic idea of the mutual refinement strategy for document clustering is the followings.

- A co-occurrence concept should be linked to the document cluster to which the co-occurrence concept makes the best contribution.
- A document cluster should be related to co-occurrence concepts that make significant contributions to the document cluster.

For this mutual refinement strategy we draw another bipartite graph. Given the graph $G = (V_{DC}+V_{CC}, E)$, V_{DC} indicates a set of (k) document clusters, V_{CC} represents a set of co-occurrence concepts in documents and E indicates the relationships between two vertices. We specify weights on edges so that we measure the contribution of co-occurrence concepts to each document cluster. This contribution is defined as the ratio between the amount of information needed to state the co-occurrence concepts in a document cluster and the total information in the document cluster in terms of the number of documents.

$$cntrb(CC_i, DC_k) = \frac{Size(docs_{DC_k}^{CC_i})}{Size(DC_k)}$$

, where $Size$ function returns the number of relevant documents, $docs_{DC_k}^{CC_i}$ indicates a set of documents with co-occurrence concept (CC_i) in the document cluster (DC_k).

After each refinement, using k new co-occurrence concept groups, each document is reassigned to the proper document cluster in the same way used for generating initial clusters. This mutual refinement iteration continues until no further changes occur on the document clusters.

4 Experimental Evaluation

In order to measure the performance of COBRA, we conduct experiments on public MEDLINE documents (abstracts). For the experiments first we collect several abstract sets about various diseases from PubMed. Specifically, we use “MajorTopic” tag along with the disease MeSH terms as queries to PubMed (see Section 3 for the tag in detail). Table 1 shows each document set and its size. After retrieving the data sets, we generate various document combinations whose numbers of classes are 2 to 10 using the document sets. Each document set used for the combinations is later used as an answer key on the performance measure.

Table 1. Document Sets

Document Sets	# of Docs	Document Sets	# of Docs
Gout	642	Otitis	5,233
Chickenpox	1,083	Osteoporosis	8,754
Raynaud Disease	1,153	Osteoarthritis	8,987
Insomnia	1,352	Parkinson Disease	9,933
Jaundice	1,486	Alzheimer Disease	18,033
Hepatitis B	1,815	Diabetes Type2	18,726
Hay Fever	2,632	AIDS	19,671
Kidney Calculi	3,071	Depressive Disorder	19,926
Impotence	3,092	Prostatic Neoplasms	23,639
AMD	3,277	Coronary Heart Disease	53,664
Migraine	4,174	Breast Neoplasms	56,075

There are a number of clustering evaluation methods. Among them we use misclassification index (MI) [18] as a measure of cluster quality since MI intuitively shows the overall quality of generated clusters. MI is the ratio of the number of misclassified objects to the size of the whole data set [18]; thus, 0% MI means the perfect clustering.

We evaluate our approach to see how much COBRA provides better clustering results compared with two leading document clustering approaches, and to check if the mutual refinement strategy is able to improve clustering quality.

4.1 Comparison of CORBRA, BiSecting K-means and CLUTO

We apply COBRA to MEDLINE articles to compare its performance with two leading document clustering approaches BiSecting K-means and CLUTO’s vcluster (<http://www-users.cs.umn.edu/~karypis/cluto>). Two recent document clustering studies showed BiSecting K-means outperforms traditional hierarchical clustering method and K-means on various document sets from TREC, Reuters, WebACE, etc, [12] [19]. A recent comparative study showed CLUTO’s vcluster outperforms several model-based document clustering algorithms [20]; none of studies have compared the two approaches.

For the experiments we generated the various document collections using document sets in Table 1. These corpora include very large corpus sets (Cx.3 as Corpus ID in Figure 1) whose size are more than 50k; most document clustering studies [13][19][20][21] used at most 8.3k to 20k size corpora for their experiments. Figure 1 shows MI results (smaller is better) for the three approaches. Table 2 shows averages of MIs as overall clustering performance index and standard deviation of MIs as the clustering performance consistency index for the approaches. These experiment results indicate that COBRA outperforms BiSecting K-means and CLUTO. As Table 2 shows, COBRA consistently produces better clustering results for various corpus sets. CLUTO yields more or less comparable clustering results with COBRA. But sometimes (for C2.2, C4.1, C6.1, C10.2, C3.3, & C10.3) CLUTO outputs poor clusters. We believe that a prestigious document clustering should consistently produce high-quality clustering results for various document sets.

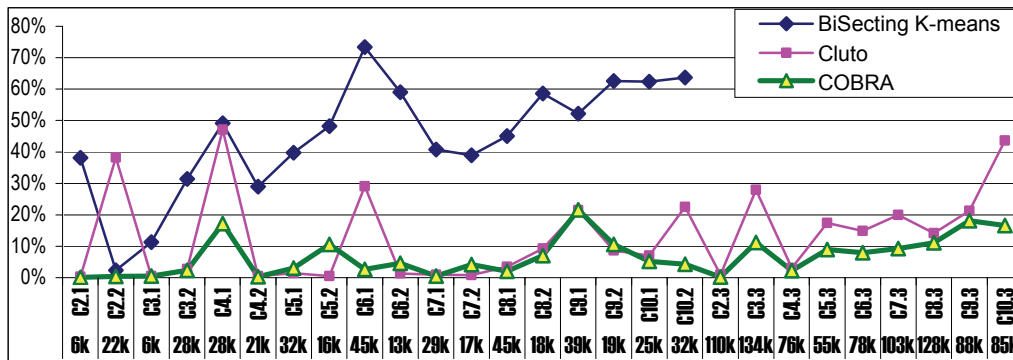


Fig. 1. Comparison of MI for BiSecting K-means, CLUTO, and COBRA (MI on X-axis and Corpus ID and Corpus Size on Y axis); Cx.y, where x indicates k , and y is a sequence number. BiSecting K-means failed to cluster the corpora whose size are more than 45k. Because BiSecting K-means produces different results every time due to its random initialization, BiSecting K-means is run ten times and the average values of MIs are used for the comparison.

Table 2. Simple Statistical Analysis of Experiment Results

	Average of MIs	Standard Deviation of MIs
BiSecting K-means	44.77%	0.18%
CLUTO	13.30%	0.14%
COBRA	6.78%	0.06%

4.2 Evaluation of Mutual Refinement Strategy on Document Clustering

We evaluate mutual refinement strategy (MRS) to check if MRS is able to improve overall clustering quality. For this evaluation we measured MIs before and after MRS process. Table 3 shows MI improvement through mutual refinement strategy (MRS). We notice

that MRS significantly improves the performance of COBRA. We also observe that, without this iterative MRS, COBRA still yields comparable performance with CLUTO.

Table 3: MI Improvements through Mutual Refinement Strategy (MRS)

Corpus ID	Before MRS	After MRS	MI Improvements	Corpus ID	Before MRS	After MRS	MI Improvements
C2.1	0.15%	0.15%	0.00%	C6.3	13.06%	7.99%	38.82%
C2.2	6.70%	0.41%	93.88%	C7.1	2.50%	0.52%	79.20%
C2.3	0.12%	0.16%	-33.33%	C7.2	5.46%	4.21%	22.89%
C3.1	0.61%	0.51%	16.39%	C7.3	7.23%	9.27%	-28.22%
C3.2	3.66%	2.36%	35.52%	C8.1	2.68%	2.00%	25.37%
C3.3	23.07%	11.24%	51.28%	C8.2	10.40%	7.04%	32.31%
C4.1	17.16%	17.18%	-0.12%	C8.3	15.59%	11.15%	28.48%
C4.2	0.95%	0.35%	63.16%	C9.1	28.15%	21.60%	23.27%
C4.3	1.93%	2.29%	-18.65%	C9.2	12.11%	10.58%	12.63%
C5.1	27.52%	3.05%	88.92%	C9.3	29.19%	18.15%	37.82%
C5.2	24.96%	10.61%	57.49%	C10.1	6.42%	5.17%	19.47%
C5.3	25.65%	8.93%	65.19%	C10.2	18.09%	4.29%	76.29%
C6.1	6.52%	2.60%	60.12%	C10.3	13.64%	16.57%	-21.48%
C6.2	13.21%	4.58%	65.33%	AVG	11.73%	6.78%	33.04%

5 Conclusions

In this paper, we mainly discussed how ontology is incorporated into document clustering procedures and how ontology-enriched bipartite graph representation and mutual refinement strategy improves the document clustering results. The main contributions of this paper are fourfold. First, COBRA becomes a new leading document clustering approach in terms of performance. Second, we introduce a new way of the use of domain knowledge in ontology on document clustering without depending on vector space model. Third, COBRA provides a meaningful explanation for each document cluster by identifying its most contributing co-occurrence concepts. Fourth, we introduce mutual refinement strategy to improve clustering quality. The strategy can be applied to virtually every document clustering approach.

References

1. van Rijsbergen, C. J. (1979). Information Retrieval, 2nd edition, London: Butterworth. (<http://www.dcs.gla.ac.uk/Keith/Preface.html>)

2. Willett, P. (1988). Recent trends in hierarchical document clustering: A critical review. *Information Processing & Management*, Vol. 24, No. 5, pp. 577-597.
3. Cutting, D., Karger, D., Pedersen, J. and Tukey, J. (1992). Scatter/Gather: A Cluster-based Approach to Browsing Large Document Collections, *SIGIR '92*, pp. 318-329.
4. Buckley, C. and Lewit, A. F. 1985. Optimization of inverted vector searches. In *Proceedings of SIGIR-85*. pp. 97-110.
5. Hearst, M. A. and Pedersen, J. O. 1996. Reexamining the cluster hypothesis: Scatter/Gather on retrieval results. In *Proceedings of SIGIR-96*. pp. 76-84. Zurich, Switzerland.
6. Zamir O., Etzioni O.: Web Document Clustering: A Feasibility Demonstration, *Proc. ACM SIGIR 98*, 1998, pp. 46-54.
7. Koller, D. and Sahami, M. 1997. Hierarchically classifying documents using very few words. In *Proceedings of ICML-97*. pp. 170-176. Nashville, TN.
8. Bill B. Wang, R.I. (Bob) McKay, Hussein A. Abbass, Michael Barlow. Learning Text Classifier using the Domain Concept Hierarchy. In *Proceedings of International Conference on Communications, Circuits and Systems 2002*, China.
9. Hotho, A., Maedche A., and Staab S. (2002). Text Clustering Based on Good Aggregations. *Künstliche Intelligenz (KI)*, 16(4), p. 48-54
10. Beyer, K., Goldstein, J., Ramakrishnan, R., & Shaft, U. (1999). When is nearest neighbor meaningful?. *Proceedings of 7th International Conference on Database Theory*, pp 217-235.
11. Kaufman L., and Rousseeuw P.J. 1990. "Finding Groups in Data: an Introduction to Cluster Analysis". John Wiley & Sons.
12. Steinbach, M., Karypis, G., and Kumar, V. (2000). A Comparison of Document Clustering Techniques. Technical Report #00-034. Department of Computer Science and Engineering, University of Minnesota.
13. Bjornar Larsen and Chinatsu Aone, Fast and Effective Text Mining Using Linear-time Document Clustering, *KDD-99*, San Diego, California, 1999.
14. Hu X., Mining Novel Connections from Large Online Digital Library Using Biomedical Ontologies, *Library Management Journal*, 26(4/5), 2005, pp. 261-270.
15. Harper, D.J., and van Rijsbergen, C. J. (1978). Evaluation of feedback in document retrieval using co-occurrence data. *Journal of Documentation*, 34, 189-216
16. Van Rijsbergen, C.J., Harper, D.J. and Porter, M.F. (1981). The selection of good search terms. *Information Processing and Management*, 17, 77-91.
17. D. Lin. An information-theoretic definition of similarity. In *Proceedings of the Fifteenth International Conference on Machine Learning*, 1998, 296-304.
18. Zeng, Y., Tang, J., Garcia-Frias, J. and Gao, G.R. (2002): An Adaptive Meta-Clustering Approach: Combining The Information From Different Clustering Results, *CSB2002 IEEE Computer Society Bioinformatics Conference Proceedings* 276-287.
19. F. Beil, M. Ester and X. Xu: "Frequent Term-Based Text Clustering", 8th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, July 23-26, 2002, Edmonton, Alberta, Canada
20. Zhong, S., & Ghosh, J. (2003). A comparative study of generative models for document clustering. *Proceedings of the workshop on Clustering High Dimensional Data and Its Applications in SIAM Data Mining Conference*.
21. Patrick Pantel, Dekang Lin: Document clustering with committees. *SIGIR 2002*: 199-206
22. Jinze Liu, Wei Wang, and Jiong Yang: A framework for ontology-driven subspace clustering, *Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining*, 2004, pp. 623-628.