BMC Bioinformatics

BioMed Central

Proceedings

Open Access

A coherent graph-based semantic clustering and summarization approach for biomedical literature and a new summarization evaluation method

Illhoi Yoo*[1], Xiaohua Hu[2] and Il-Yeol Song[2]

Address: [1]Department of Health Management and Informatics, School of Medicine, University of Missouri-Columbia, USA and [2]College of Information Science and Technology, Drexel University, USA

Email: Illhoi Yoo* - yooil@health.missouri.edu; Xiaohua Hu - thu@cis.drexel.edu; Il-Yeol Song - song@drexel.edu

* Corresponding author

from First International Workshop on Text Mining in Bioinformatics (TMBio) 2006
Arlington, VA, USA. 10 November 2006

Published: 27 November 2007

BMC Bioinformatics 2007, **8**(Suppl 9):S4 doi:10.1186/1471-2105-8-S9-S4

This article is available from: http://www.biomedcentral.com/1471-2105/8/S9/S4

Abstract

Background: A huge amount of biomedical textual information has been produced and collected in MEDLINE for decades. In order to easily utilize biomedical information in the free text, document clustering and text summarization together are used as a solution for text information overload problem. In this paper, we introduce a coherent graph-based semantic clustering and summarization approach for biomedical literature.

Results: Our extensive experimental results show the approach shows 45% cluster quality improvement and 72% clustering reliability improvement, in terms of misclassification index, over Bisecting K-means as a leading document clustering approach. In addition, our approach provides concise but rich text summary in key concepts and sentences.

Conclusion: Our coherent biomedical literature clustering and summarization approach that takes advantage of ontology-enriched graphical representations significantly improves the quality of document clusters and understandability of documents through summaries.

Background

A huge amount of textual information has been produced and collected in text databases or digital libraries for decades because the most natural form to store information is text. For example, MEDLINE, the largest biomedical bibliographic text database, has more than 16 million articles and more than 10,000 articles are weekly added to MEDLINE. Figure 1 shows the exploding volume of biomedical literature in MEDLINE over the past 57 years, which makes it difficult to locate and manage the public biomedical information.

In order to tackle this pressing text information overload problem, document clustering and text summarization together have been used as a solution. This is because document clustering enables us to group similar text information and then text summarization provides condensed text information for the similar text by extracting the most important text content from a similar document set or a document cluster. For this reason, document clustering and text summarization can be used for important components of information retrieval system.

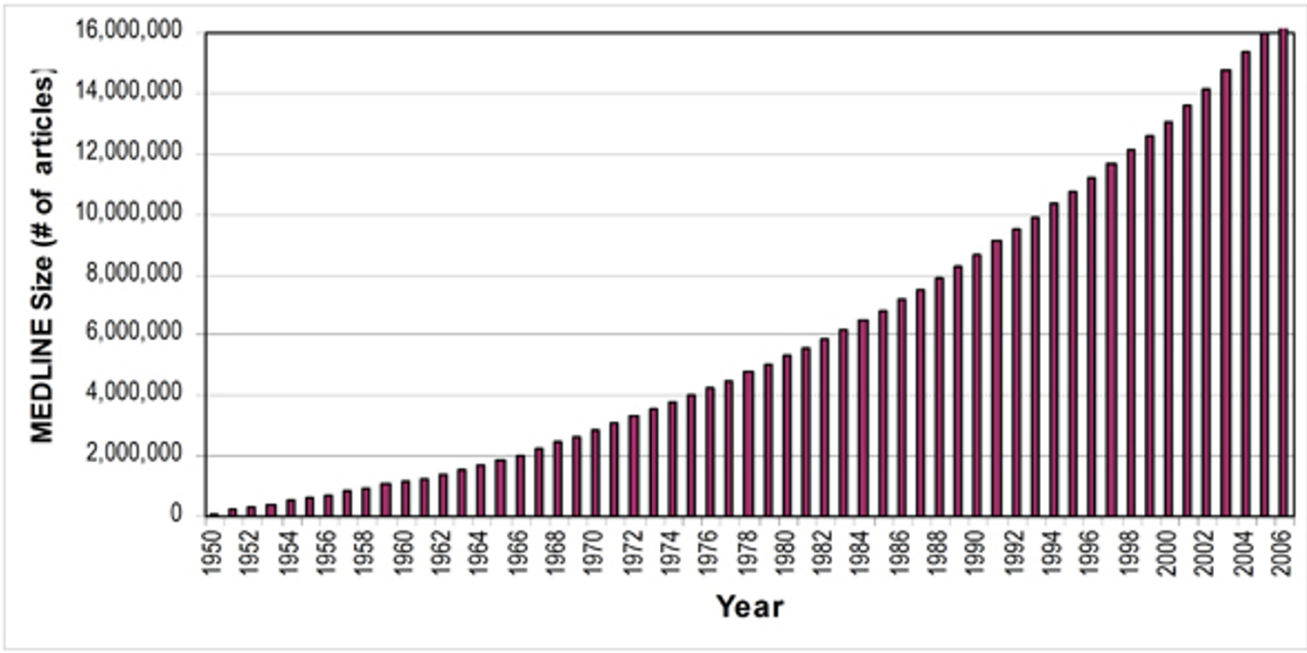

Figure 1
The Exploding Number of MEDLINE Articles over Years. The data was retrieved from PubMed [1] using "dp" keyword, which stands for Data of Publication.

Traditional document clustering and text summarization approaches, however, have four major problems. First, traditional approaches are based on the vector space model. The use of vector space representation for documents causes two major limitations. The first limitation is the vector space model assumes all the dimensions in the space to be independent. In other words, the model assumes that words/terms are mutually independent in documents. However, most words/terms in a document are related to each other. This is a fundamental problem of the vector space model on document representation. For example, consider the word set, {Vehicle, Car, Motor, Automobile, Auto, Ford}; they are not independent but are closely related. The second limitation is that text processing in a high dimensional space significantly hampers its similarity detection for objects (here, documents) because distances between every pair of objects tend to be the same regardless of data distributions and distance functions [2]. Thus, it may dramatically decrease clustering performance.

Second, traditional document clustering and text summarization approaches do not consider semantically related words/terms (e.g., synonyms or hyper/hyponyms). For instance, they treat {Cancer, Tumor, Neoplasm, Malignancy} as different terms even though all these words have very similar meaning. This problem may lead to a very low relevance score for relevant documents because the documents do not always contain the same forms of words/terms. In fact, the problem comes intrinsically from the fact that traditional document clustering approaches do not "perceive" objects nor "understand" what the objects "mean".

Third, traditional clustering approaches cannot provide an explanation of why a document is grouped into one of document clusters [3] because they pursue a similarity-based mechanism on clustering, which does not produce any models or rules for document clusters. Another reason is that they involve a very high dimensional vector space representation for a document, which does not allow users to interpret the representation.

Lastly, on vector representations of documents based on the bag-of-words model, they tend to use all the words/terms in the documents after removing the stop-words. This leads to thousands of dimensions in the vector representation of documents; this is called the "*Curse of Dimensionality*". In addition, it is well known that only a very small number of words/terms in documents have distinguishable power on clustering documents and become the key elements of text summaries. Those words/terms are normally the concepts in the domain related to the documents.

These four traditional problems have motivated this study. In this paper, we introduce a coherent biomedical literature clustering and summarization approach. The coherence of document clustering and text summarization is required because a set of documents are usually multiple-topics. For this reason text summarization does not yield high-quality summary without document clustering. On the other hand, document clustering is not very useful for users to understand a set of documents if the explanation for document categorization or the summaries for each document cluster is not provided. In other words, document clustering and text summarization are complementary. This is the primary motivation for the coherent approach of document clustering and text summarization.

The key of the approach is the use of the graphical representation method for text using a biomedical ontology. The graphical representation method represents a set of documents or sentences as an ontology-enriched scale-free graph. This ontology-enriched graphical representation method provides a very natural way to portray the contents of documents, provides *document representation independence* (to be discussed in Section 3), and guarantees better scalability on text mining than the traditional vector space model.

The ontology-enriched graph (i.e., the corpus-level graphical representation of documents) is clustered under the consideration of the power law distribution of terms in documents to identify document cluster models as semantic chunks capturing the semantic relationships among the terms in document clusters. These document cluster models are used for assigning documents to clusters to group semantically similar documents in accordance with the similarity between each document and document cluster models. For each document cluster, text summarization is performed by constructing Text Semantic Interaction Network (TSIN) using the semantic relationships in the document cluster model. TSIN is constructed based on the semantic similarities among selected sentences that depend on the edit distances between their ontology-enriched graphical representations. Significant text contents by considering their centrality in the network become the summary.

The primary contribution of this paper is we introduce a coherent biomedical literature clustering and summarization approach that takes advantage of ontology-enriched graphical representations of documents. Our approach significantly improves the quality of document clusters and understandability of documents through summaries for each document cluster.

Results

Document sets

In order to measure the effectiveness of the proposed approach, **C**lustering and **Su**mmarization with **Gra**phical **R**epresentation for documents (CSUGAR), we conducted extensive experiments on public MEDLINE abstracts. For the extensive experiments, first we collected document sets related to various diseases from MEDLINE. We use "MajorTopic" tag along with the disease-related MeSH terms as queries to MEDLINE. Table 1 shows the base document sets retrieved from MEDLINE. After retrieving the base data sets, we generate various document combinations whose numbers of classes are 2 to 9 (as shown in Table 2) by randomly mixing the document sets. The document sets used for generating the combinations are later used as answer keys on the document clustering performance measure.

Evaluation method

Document clustering

In general, clustering systems have been evaluated in three ways. First, document clustering systems can be assessed based on user studies whose main purpose is to measure the user's satisfaction with the output of the systems. Second, the objective functions of clustering algorithms have been used to evaluate the algorithms. This method is normally used when the classes are unknown. Finally, clustering algorithms can be evaluated by comparing clustering output with known classes as answer keys. There have been a number of comparison metrics (see [4] for details).

Table 1: The Document Sets and Their Sizes

Document Sets	ID	No. of Docs
Gout	Gt	642
Chickenpox	Ghk	1,083
Raynaud Disease	RD	1,153
Insomnia	Ins	1,352
Jaundice	Jn	1,486
Hepatitis B	Hpt	1,815
Hay Fever	HF	2,632
Kidney Calculi	KS	3,071
Impotence	Imp	3,092
Age-related Macular Degeneration	AMD	3,277
Migraine	Mg	4,174
Otitis	Ot	5,233
Osteoporosis	Ost	8,754
Osteoarthritis	OA	8,987
Parkinson Disease	Pk	9,933
Alzheimer Disease	Alz	18,033
Diabetes Type 2	Diab	18,726
AIDS	AIDS	19,671
Depressive Disorder	Dep	19,926
Prostatic Neoplasm	Pros	23,639
Coronary Heart Disease	CHD	53,664
Breast Neoplasm	Bre	56,075

Table 2: List of Test Corpora Generated from the Base Data Sets

Corpus Name	Corpus ID	No. of Docs
2_Mg-Alz	C2.1	22 K
2_Ot-AMD	C2.2	9 K
2_Bre-CHD	C2.3	110 K
3_AMD-Mg-Ot	C3.1	28 K
3_OA-Ost-Pk	C3.2	13 K
3_Pros-Bre-CHD	C3.3	132 K
4_Alz-AMD-Ot-Ost	C4.1	35 K
4_Ost-AMD-Mg-Ot	C4.2	76 K
4_Dep-AIDS-Alz-Diab	C4.3	21 K
5_Alz-AMD-Mg-Ost-Ot	C5.1	55 K
5_HF-KS-Imp-AMD-Mg	C5.2	39 K
5_AIDS-Alz-AMD-Ot-Ost	C5.3	16 K
6_AMD-Mg-Ot-OA-Ost-Pk	C6.1	40 K
6_Ins-Jn-Hpt-HF-KS-Imp	C6.2	13 K
6_Pros-Ost-Alz-AIDS-Dep-Diab	C6.3	109 K
7_Chk-Jd-Hpt-HF-KS-AMD-Mg	C7.1	20 K
7_Jn-Hpt-HF-KS-Imp-AMD-Mg	C7.2	18 K
7_Ost-Pk-Alz-AIDS-Dep-Diab-Pros	C7.3	119 K
8_Hpt-HF-AMD-Mg-Ot-OA-Ost-Pk	C8.1	14 K
8_Mg-Gt-Chk-Jn-Hpt-HF-KS-AMD	C8.2	18 K
8_OA-Ost-Pk-Alz-AIDS-Dep-Diab-Pros	C8.3	128 K
9_Mg-Gt-Chk-RD-Jn-Hpt-HF-KS-AMD	C9.1	19 K
9_Mg-Chk-Ins-Jn-Hpt-HF-KS-Imp-AMD	C9.2	22 K
9_Ot-OA-Ost-Pk-Alz-AIDS-Dep-Diab-Pros	C9.3	133 K

Among them we use misclassification index (MI) [5], F-measure, and cluster purity as clustering evaluation metrics.

MI is the ratio of the number of misclassified objects to the size of the whole data set; thus, MI with 0% means the perfect clustering. For example, MI is calculated as follows under the situation shown in the Table 3. Note that the total number of objects in classes is the same as the number of objects in clusters.

$$MI = \frac{\#\text{ of misclassified objects}}{\text{total }\#\text{ of objects}} = \frac{3}{100} = 3\%$$

F-measure is a measure that combines the recall and the precision from information retrieval. When F-measure is used as a clustering quality measure, each cluster is treated as the retrieved documents for a query and each class is regarded as an ideal query result. Larsen and Aone [6] defined overall clustering F-measure as the weighted average of all values for the F-measure as given by the following: for class i and cluster j

$F = \sum_i \frac{n_i}{n} \max\{F(i,j)\}$, where the max function is over all clusters, n is the number of documents, and

$$F(i,j) = \frac{2 \times \text{Recall}(i,j) \times \text{Precision}(i,j)}{\text{Recall}(i,j) + \text{Precision}(i,j)}$$

However, this formula is sometimes problematic; if a cluster has the majority (or even all) of objects, more than a class are matched with only such a cluster for calculating F-measure and some clusters are not matched with any classes (meaning that those clusters are not evaluated in F-measure). Thus, we exclude matched clusters on the process of the *max* function. In consequence, a class is matched with only a cluster that yields the maximum F-measure.

The cluster purity indicates the percentage of the dominant class members in the given cluster; the percentage is nothing more than the maximum precision over the classes. For measuring the overall clustering purity, we use the weighted average purity as shown below (for class i

Table 3: Sample Classes and Clustering Output. Each number in the table is the number of objects in its class or cluster

Classes	20	50	30
Clusters	20	53	27
	No misclassified objects	3 objects misclassified	No misclassified objects

and cluster j). Like F-measure, we eliminate matched clusters on the process of the *max* function.

$Purity = \sum_j \frac{n_j}{n} \max_i \{\text{Precision}(i, j)\}$, where n is the number of documents

Text summarization

Text summarization has often been evaluated by comparing system-generated summary with human-generated summary as "gold standards". However, this evaluation method has two problems. First, human-generated summary is not always available for every domain; there are *de facto* standard data sets with summaries called Document Understanding Conferences (DUC) for summarization approaches but these data sets are not fit for our evaluation because we apply our method to biomedical domain using biomedical ontology. Second, as Salton [7] and Nomoto and Matsumoto [8] indicated, human-generated summaries vary so that they are not really reliable and may not be used as "gold standards".

To this end, we introduce a new summarization evaluation method. This evaluation method judges the quality of summary in terms of the performance of document clustering for original documents excluding summary. Thus, for high-quality summary its document clustering performance is subjected to be poor; the higher summary quality, the lower document clustering performance for a set of documents excluding its summary sentences.

Experiment results

Document clustering

Because the full detailed experiment results are too big to be depicted in this paper, we average the clustering evaluation metric values and show the standard deviations (σ) for them to indicate how consistent a clustering approach yields document clusters (simply, the reliability of each approach). The σ would be a very important document clustering evaluation factor because document clustering is performed in the circumstance where the information about documents is unknown. Table 4 summarizes the statistical information about clustering results. From the table, we notice the following observations:

• CSUGAR outperforms the nine document clustering methods.

• CSUGAR has the most stable clustering performance regardless of test corpora, while CLUTO Bisecting K-means and K-means do not always show stable clustering performance.

• Hierarchical approaches have a serious scalability problem.

• STC and the original Bisecting K-means have a scalability problem.

• MeSH Ontology improves the clustering solutions of STC.

Unexpectedly, the original BiSecting K-means [9] shows poor performance. Unlike the studies [10] and [11], our experiment results indicate the original BiSecting K-means is even worse than K-means. On the other hand, such a result is also found in [12]. This contradiction leads us to deem that the clustering results of BiSecting K-means and K-means heavily depend on document sets used.

We observe that CSUGAR has the best performance, yields the most stable clustering results and scales very well. More specifically, CSUGAR shows 45% cluster quality improvement and 72% clustering reliability improvement, in terms of MI, over Bisecting K-means with the best parameters. There are three reasons to support the results. First, CSUGAR uses an ontology-enriched graphical representation that still retains the semantic relationship information about the core concepts of the documents. Second, CSUGAR uses document cluster models that capture the core semantic relationship for each document

Table 4: Summary of Overall Experiment Results on MEDLINE Document Sets

	STC		K-means	Original Bisecting K-means [25]	CLUTO Bisecting K-means		CSUGAR
	word strings	concept strings			Largest	LOS	
MI	μ: 0.429	μ: 0.359	μ: 0.128	μ: 0.395	μ: 0.161	μ: 0.096	μ: **0.053**
	σ: 0.238	σ: 0.149	σ: 0.148	σ: 0.193	σ: 0.139	σ: 0.112	σ: **0.031**
Purity	μ: 0.601	μ: 0.731	μ: 0.932	μ: 0.666	μ: 0.918	μ: 0.944	μ: **0.947**
	σ: 0.214	σ: 0.098	σ: 0.080	σ: 0.154	σ: 0.064	σ: 0.056	σ: **0.030**
F-measure	μ: 0.499	μ: 0.512	μ: 0.828	μ: 0.532	μ: 0.780	μ: 0.880	μ: **0.926**
	σ: 0.285	σ: 0.198	σ: 0.206	σ: 0.236	σ: 0.180	σ: 0.139	σ: **0.062**

LOS: selecting the cluster (to be bisected) with the least overall similarity and Largest: selecting the largest cluster to be bisected. MI: the smaller, the better clustering quality. Purity and F-measure: the bigger, the better clustering quality.

cluster to categorize documents. Third, as the number of documents to be processed increase, a corpus-level graphical representation at most linearly expands or keeps its size with only some changes on edge weights, while a vector space representation (i.e. document*word matrix) at least linearly grows or increases by $n*t$, where n is the number of documents and t is the number of distinct terms in documents. In addition to the superiority of CSUGAR over traditional document clustering approaches, one should notice that only CSUGAR supply a meaningful explanation for document clustering as well as the summaries of each document cluster through generated document cluster models (as shown in Figure 2). This could be critical for users to understand clustering results and documents as a whole because document clustering is performed in the circumstance where the information about documents is unknown.

Text summarization

Figure 3 shows the comparison of degree centrality approach and mutual refinement (MR) centrality approach for four sample datasets due to the page limitation. Table 5 shows document clustering performance decrease as summary compression ratio increase for each summarization method; MI is used for clustering evaluation metric. In contrast to our expectation, MR centrality does not show better performance than degree centrality except 5% summary compression ratio even if MR spends extra time to refine node ranking process; the complexity of MR is $O(n(n - 1)/2)$. However, several studies have observed that the degree centrality is a decent but fast method to measure the centrality of node in a graph. For example, Erkan and Radev [13] found degree centrality and LexRank (simplified PageRank algorithm) yield similar output quality in text summarization and Wu et al [14] also showed degree centrality produce similar output quality to betweenness centrality in their scale-free network study.

We include two sample text summarization outputs for Alzheimer Disease and Osteoarthritis document clusters in Additional file 1. This summary consists of document cluster model that would be regarded as the semantic chunk of the document cluster, and top seven summary sentences.

Conclusion

In this paper, we introduce a coherent biomedical literature clustering and summarization approach. Document clustering and text summarization should be integrated

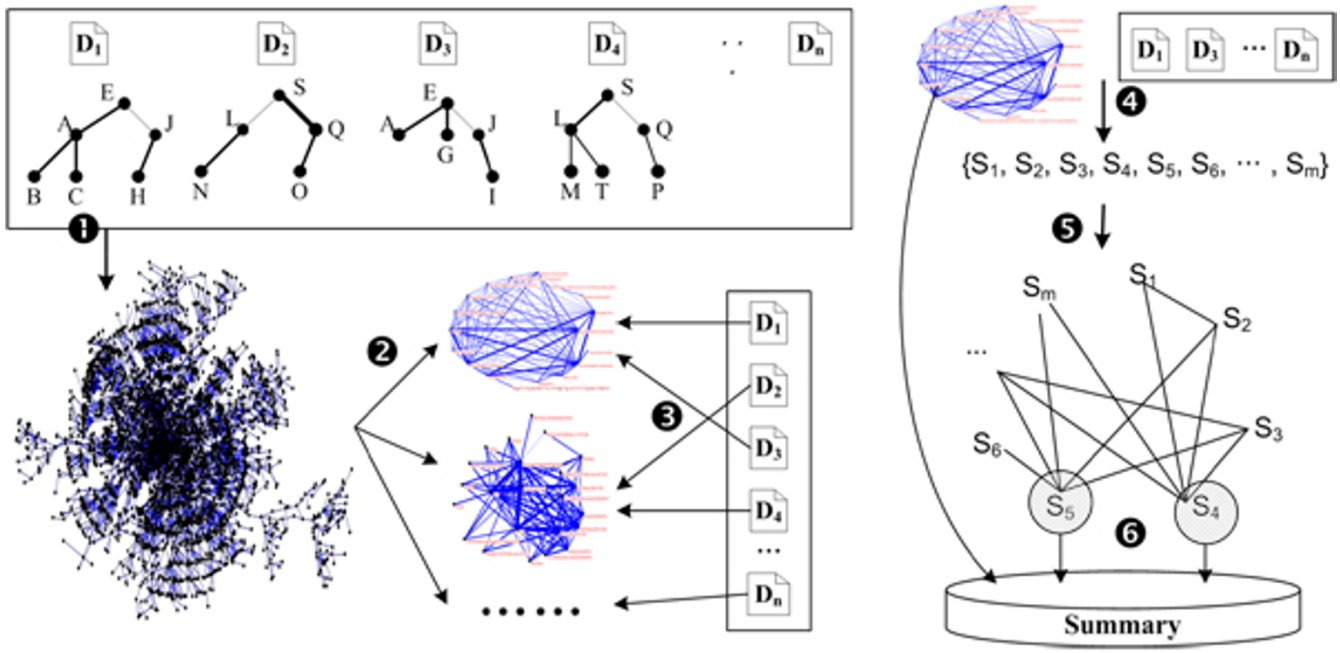

Figure 2

The Dataflow of the CSUGAR. ❶ making an ontology-enriched graphical representation for documents. ❷ graph clustering for a graphical representation of documents. ❸ assigning documents to clusters based on the document cluster models. ❹ making ontology-enriched graphical representations for each sentence. ❺ constructing Text Semantic Interaction Network (TSIN). ❻ selecting significant text contents for summary

Table 5: Document Clustering Performance Decrease as Summary Compression Ratio Increase for Each Summarization Method

Summary compression Ratio/Summarization Method	5%	10%	20%
Degree centrality	0.4%	2.0%	6.0%
Mutual Refinement centrality	0.7%	2.0%	6.0%

into a coherent framework. There are two reasons to support this claim: First, a set of documents are usually multiple-topics and thus text summarization does not yield high-quality summary without document clustering. Second, document clustering is not very useful for users to understand a set of documents if the explanation for document categorization or the summaries for each document cluster is not provided. Simply, document clustering and text summarization are complementary each other. This is the primary motivation for the coherent approach of document clustering and text summarization.

We used the graphical representation method to represent documents using a biomedical ontology for document clustering and text summarization. The graphical representation method represents a set of documents or sentences as an ontology-enriched scale-free graph. This ontology-enriched graphical representation method provides a very natural way to portray the contents of documents, provides *document representation independence*, and guarantees better scalability on text mining than the traditional vector space model. Our approach significantly improves the quality of document clusters and under-

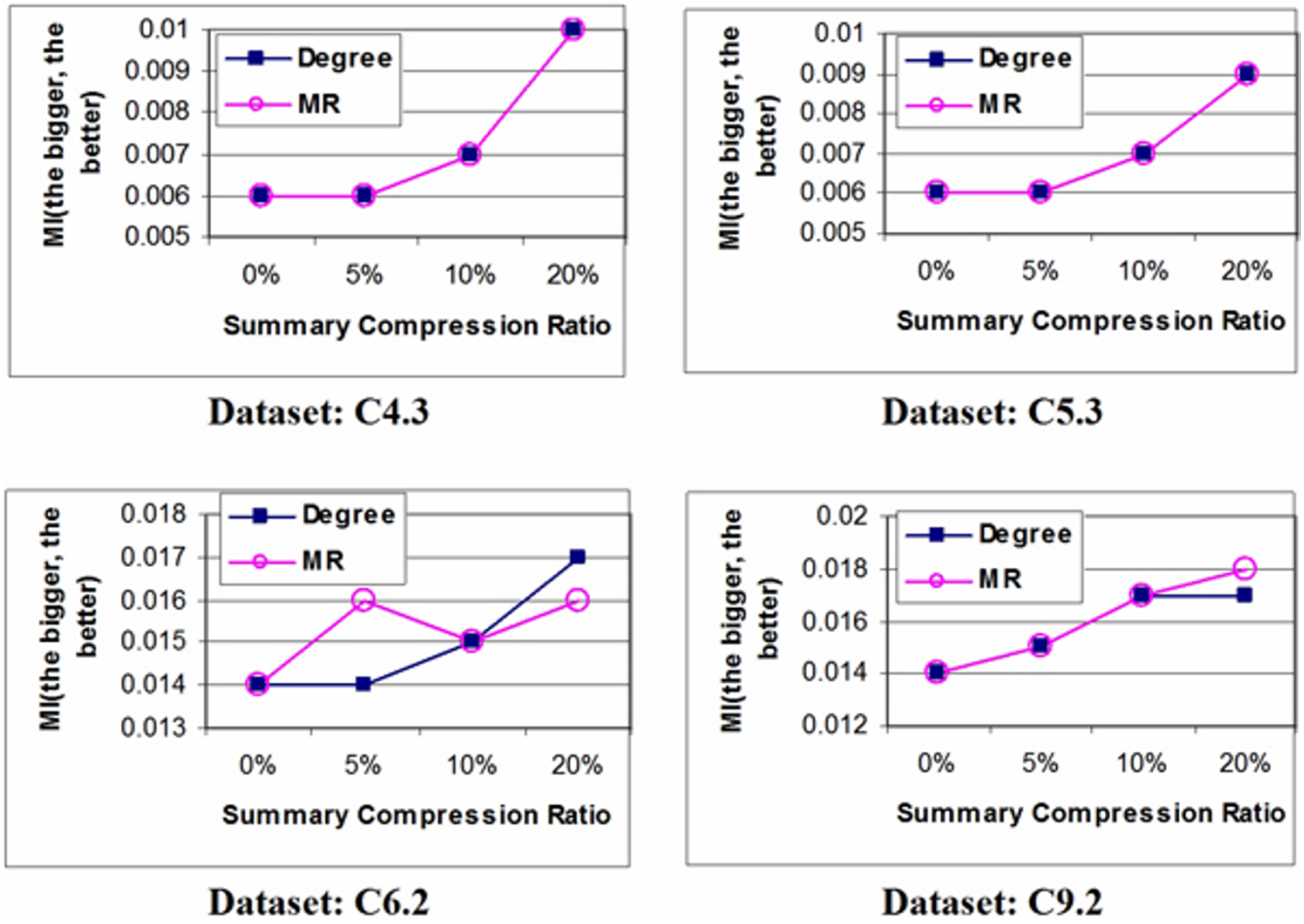

Figure 3
Comparison of Degree Centrality Approach and Mutual Refinement Centrality Approach for Four Sample Datasets.

standability of documents through summaries for each document cluster.

Methods

We present a novel coherent document clustering and summarization approach, called *C*lustering and *SU*mmarization with *Gr*Aphical *R*epresentation for documents (CSUGAR). Before discussing CSUGAR in detail we first discuss MeSH ontology due to its importance in our approach.

Medical Subject Headings (MeSH) ontology

Medical Subject Headings (MeSH) [1] mainly consists of the controlled vocabulary and a MeSH Tree. The controlled vocabulary contains several different types of terms, such as Descriptor, Qualifiers, Publication Types, Geographics, and Entry terms. Among them, Descriptors and Entry terms are used in this research because only they can be extracted from documents. 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 Neoplasm", "Neoplasm, Benign"}. MeSH descriptors are organized in a MeSH Tree, which can be seen as the MeSH Concept Hierarchy. In the MeSH Tree there are 15 categories (e.g. category A for anatomic terms), and each category is further divided into subcategories. For each subcategory, corresponding descriptors are hierarchically arranged from most general to most specific. In addition to its ontology role, MeSH descriptors have been used to index MEDLINE articles. For this purpose, about 10 to 20 MeSH terms are manually assigned to each article (after reading full papers). On the assignment of MeSH terms to articles, about 3 to 5 MeSH terms are set as "MajorTopics" that primarily represent an article.

Clustering and summarization with graphical representation (CSUGAR)

The proposed approach consists of two components, document clustering and text summarization as shown in Figure 2. Each step is discussed in detail below; see the circled numbers in Figure 2. Note the steps 1 to 3 correspond to document clustering and the steps 4 to 6 correspond to text summarization.

Step1: Ontology-enriched graphical representation for documents through concept mapping

All document clustering methods must first convert documents into a proper format. Since we recognize documents as a set of concepts that have their complex internal semantic relationships, we represent each document as a graph structure using the MeSH ontology. The primarily motivations behind the graphical representation of documents are the following. First, the graphical representation of documents is a very natural way to portray the contents of documents because the semantic relationship information about the concepts in documents remains on the representation while the vector space representation loses all the information. Second, the graphical representation method provides *document representation independence*. This means that the graphical representation of a document does not affect other representations. In the vector space representation, the addition of a single document usually requires the changes of every document representation. Third, the graphical representation guarantees better scalability than vector space model. Because a document representation is an actual data structure on text processing, its size should be as small as possible for better scalability. As the number of documents to be processed increases, a corpus-level graphical representation at most linearly expands or keeps its size with only some changes on edge weights, while a vector space representation (i.e. document*word matrix) at least linearly grows or increases by $n*t$ where n is the number of documents and t is the number of distinct terms in documents.

We represent the graph as a triple $G = (V, E, w)$, where V is a set of vertices that represent MeSH Descriptors, E is a set of edges that indicate the relationships between vertices, and w is a set of edge weights that are assigned according to the strength of the edge relationships. The relationships are derived from both the concept hierarchy in the MeSH ontology and the concept dependencies over documents. All the details are discussed below.

The whole procedure takes the following three steps: concept mapping, construction of individual graphical representations with both mapped concepts and their higher-level concepts, and integration of individual graphical representations. First, it maps terms in each document into MeSH concepts. In order to reduce unnecessary searches rather than searching all Entry terms in each document, it selects 1 to 3-gram words as the candidates of MeSH Entry terms after removing stop words from each document. After matching the candidates with MeSH Entry terms, it replaces Entry terms with Descriptor terms to unite the synonyms or the related terms to descriptors. Then, it filters out some MeSH concepts that are too general (e.g. HUMAN) or too common over MEDLINE articles (e.g. ENGLISH ABSTRACT). We assume that those terms do not have distinguishable power on clustering documents.

In the second step, it extends the detected MeSH concepts by incorporating higher-level (i.e. more general) concepts in the MeSH Tree on a graphical representation. The main purpose of the concept extension is to make the graphical representation richer in terms of meaning. The primary

benefit of the concept extension is to help users recognize similar topics. For example, a migraine document may involve the following concepts {"VASCULAR HEADACHES", "CEREBROVASCULAR DISORDERS", "BRAIN DISEASES", "CENTRAL NERVOUS SYSTEM DISEASES"} using MIGRAINE concept in the document through its concept extension, and these extended concepts may link the document to any vascular headache documents. For each step of the concept extension, an edge consisting of a concept and its higher-level concept is drawn in the graph. For such new edges, weights are assigned based on their extension lengths. This is based on the fact that as the layers of the concept hierarchy go up, concepts become more general and less similar than concepts at lower levels. In this way, as concept-extensions are made from a base concept, the weights of the new edges by the concept-extensions decrease. The mechanism can be explained with the taxonomic similarity [15] or the set similarity (i.e. $\frac{|\alpha \cap \beta|}{|\alpha \cup \beta|} = \frac{|\beta|}{|\alpha|}$, where α is a set of all the parent concepts of a concept plus the concept and β is a set of all the parent concepts of its immediate higher-level concept and plus the concept).

Figure 4 illustrates this second step. Based on the MeSH Tree, Descriptor terms (e.g. {B, C, H} for the document D_1) of each document are extended with their higher-level concepts (e.g., {*A*, *E*, *J*} in Figure 4; our approach involves higher-level concepts up to before the 15 category subroots of the MeSH Tree. The mechanism of edge weights is simple. The weight of edge B-A, for example, is $\frac{|\{B,A,E\} \cap \{A,E\}|}{|\{B,A,E\} \cup \{A,E\}|} = \frac{2}{3}$. For identical edges (e.g., A-E and Q-S), their weights add up. For example, the weight of edge A-E is $2 \times \frac{|\{A,E\} \cap \{E\}|}{|\{A,E\} \cup \{E\}|} = 1$. Note that the thickness of the edges in the graphical representations indicates the edge weights; the thicker the heavier weight.

In the third step, it merges the individual graphs generated from each document, into a corpus-level graph. In this step it further enriches the graph by reflecting concept dependence, which implies the necessary co-occurrence of concepts in documents, on the graph. This is based on the fact that co-occurrence concepts imply some semantic associations that the ontology cannot contain. The remaining problem for co-occurrence concepts is how to set the co-occurrence threshold; term pairs whose co-occurrence counts equal or bigger than the value are considered as co-occurrence terms. Because the threshold value fairly depends on documents or queries to retrieve documents, we develop a simple algorithm to detect a reasonable threshold value instead of just setting a fixed value. This algorithm tries to find a bisecting point in one-dimensional data. It first sorts the data, takes the two end objects (i.e. the minimum and the maximum) as centroids, and then assigns the remaining objects to the two centroids based on the distances between each remaining object and a centroid. After each assignment of the objects, the centroids are updated. 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. On the graph integration, edge weights add up for the identical edges.

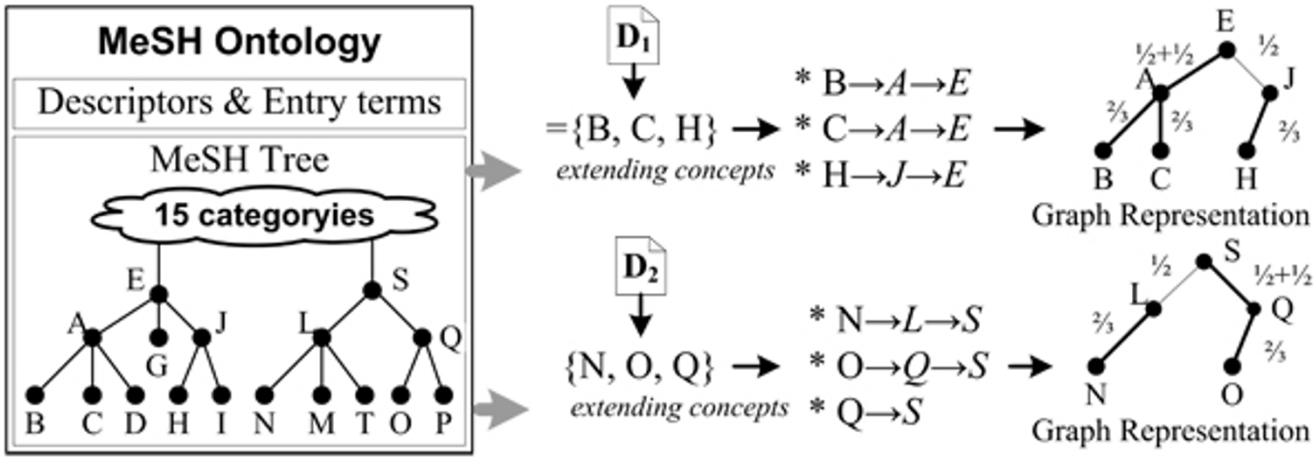

Figure 4
Individual Graphical Representations for Each Document.

Figure 5 shows the third step. The corpus-level graph is made by merging the individual graphs and by reflecting co-occurrence concepts as new edges. Note that the integrated graph in the Figure 6 is based on only the four documents (D_1 to D_4) and two co-occurrence concepts from the whole document set (D_1 to D_n). Figure 6 shows a real graph that is a typical scale-free network.

Additionally, Figure 5 presents one of the advantages of our approach. Although D_1 and D_3 documents, or D_2 and D_4 documents do not share any common concepts (thus, traditional approaches do not recognize any similarity between those documents), when the documents are represented in graphs, their graphs can have some common vertices (e.g., {A, E, J} for D_1 and D_3 documents, and {L, S, Q} for D_2 and D_4 documents). Thus, D_1 and D_3 documents, and D_2 and D_4 documents are regarded as similar to each other. This is because our document representation method involves higher-level concepts relating semantically similar documents that do not share common terms.

Step2: Graph clustering for a graphical representation of documents

A number of phenomena or systems, such as the Internet [2] have been modeled as networks or graphs. Traditionally those networks were interpreted with Erdos & Rényi's random graph theory, where nodes are randomly distributed and two nodes are connected randomly and uniformly (i.e. Gaussian distribution) [16]. However, researchers have observed that a variety of networks such as those mentioned above, deviate from the random graph theory [17] in that a few most connected nodes are connected to a high fraction of all nodes (there are a few *hub* nodes). However, these *hub* nodes cannot be explained with the traditional random graph theory. Recently, Barabasi and Albert introduced the scale-free network [2]. The scale-free network can explain the *hub* nodes with high degrees because its degree distribution decays as a power law, $P(k) \sim k^{-\gamma}$, where $P(k)$ is the probability that a vertex interacts with k other vertices and γ is the degree exponent [2].

Recently, Ferrer-Cancho and Solé have observed that the graph connecting words in English text follows a scale-free network [3]. Thus, the graphical representation of documents belongs to a highly heterogeneous family of scale-free networks. Our Scale Free Graph Clustering (SFGC) algorithm is based on the scale-free nature (i.e. the existence of a few hub vertices (concepts) in the graphical representation). SFGC starts detecting k hub vertex sets (HVSs) as the centroids of k graph clusters and then assigns the remaining vertices to graph clusters based on the relationships between the remaining objects and k hub vertex sets.

Before we describe SFGC in detail, we define the following terms.

▪ Hub vertices: a set of the most heavily-connected vertices in each graph cluster in terms of both the degrees of vertices and the weights of the edges connected to vertices due to the weighted graph.

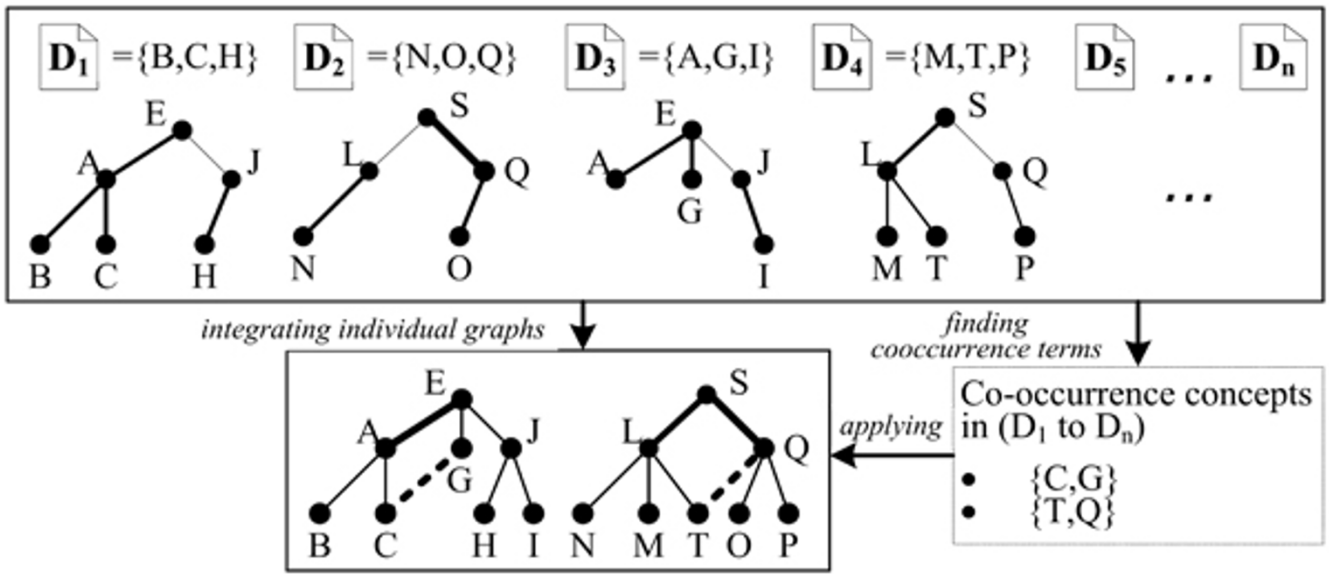

Figure 5
Integration of Individual Graphs.

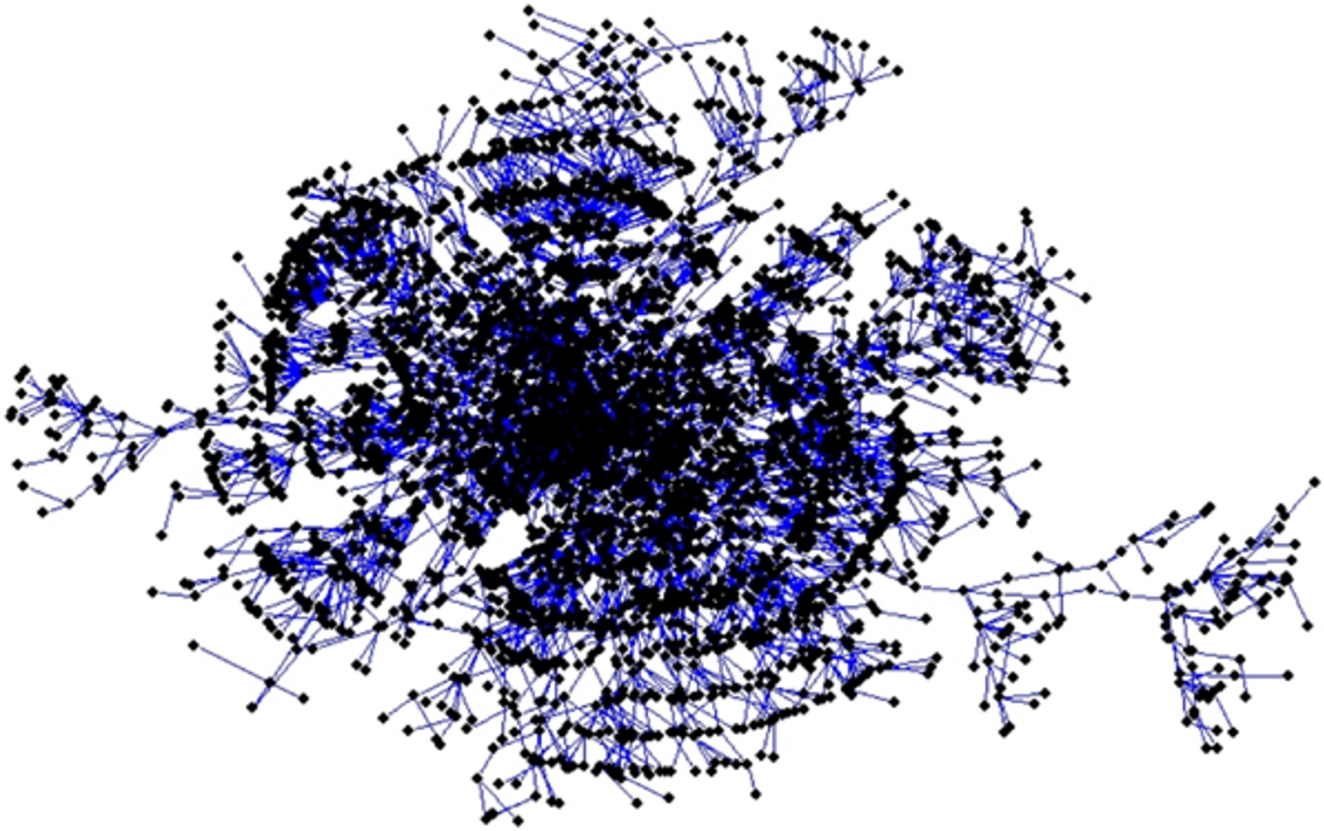

Figure 6
A Graphical Representation of a Document Set as a Scale-Free Network. This graph is from a test corpus that consists of 21,977 documents and has 9 classes.

▪ A graph cluster: a set of vertices that have stronger relationships with the hub vertices of the corresponding cluster than those of other clusters.

▪ A centroid: a set of hub vertices, not a single vertex because we assume a single term as a representative of a document cluster may have its dispositions so that the term may not have strong relationships with other key terms of the corresponding cluster. This complies with the scale-free network theory where centroids are a set of vertices that have high degrees.

Detecting k *hub vertex sets as cluster centroids*

The main process of the SFGC is to detect k hub vertex sets (HVS) as the centroids of k graph clusters. HVS is a set of vertices with high degrees in a scale-free network. Because HVSs are the cluster centroids, we might consider betweenness-based methods such as Betweenness Centrality [18] to measure the centrality of vertices in a graph; see [19] for the latest comprehensive review. However, those methods lead to quadratic or cubic running times [19] so that they are not appropriate for very large graphs. A recent scale-free network study [20] reports that Betweenness Centrality (BC) yields better experiment results to find cluster centroids than random sampling, degree ranking, and well-known HITS but degree ranking is comparable with BC. If we consider the complexities of BC ($O(|V|^2)$) and degree ranking ($O(|V|)$) in very large graphs, degree ranking should be selected. Unlike [20] that considers only the degrees (i.e. counting edges connected to vertices), we consider edge weights for a weighted graph. To this end, we introduce the salient scores of vertices that are obtained from the sum of the weights of the edges connected to vertices. The salience of a vertex is mathematically rendered as follows.

$$Salience(v_i) = \sum_{e_j \in \{e_j | e_j \text{ having } v_i\}} \text{weight of } e_j$$

In order to set highly salient vertices as HVS first, the vertices are sorted in the descending order based on their salient scores.

Within the top *n* vertices SFGC iteratively searches a vertex that has a strong relationship with any vertices in each HVS because we assume all the vertices in a HVS are strongly related to each other. If a vertex has multiple relationships with more than a HVS, the HVS that has the strongest relationship with the vertex is selected. After assigning a vertex, the vertex will not be used for HVS detection anymore.

Sometimes, HVSs are semantically similar enough to be merged because a document set (or a document cluster) may have multiple but semantically related topics. In order to measure the similarity between HVSs, we calculate an intra-edge weight sum (as similarity) of each of the two HVSs and an inter-edge weight sum between the HVSs. This mechanism is based on the fact that a "good" graph cluster should have both the maximum intra-cluster similarity and the minimum inter-cluster similarity. Thus, if an inter-edge weight sum is equal to or bigger than any of intra-edge weight sums, the corresponding two HVSs are merged. If this happened, SFGC tries to seek a new HVS.

Assigning remaining vertices to k *graph clusters*

Each of the remaining vertices (i.e. non-HVS) is (re)assigned to the graph cluster to which the vertex is the most similar. The similarity is based on the relationships between the vertex and each of the *k* HVSs. The degree of being strong in relationships is measured in the sum of edge weights. In this way *k* graph clusters are populated with the remaining vertices. In order to refine the graph clusters it iteratively reassigns vertices to the clusters with the update of *k* HVSs from their graph clusters just like K-means that updates *k* cluster centroids at each iteration to improve cluster quality. During the updates of HVSs, it uses the bisecting technique, used for co-occurrence threshold, to select new HVS from the vertices in each graph cluster based on their salient scores. In other words, the technique separates the vertices in each graph cluster into two vertex groups (i.e. HVS and non-HVS). Using the new HVSs, the vertices are reallocated to the most similar cluster. These iterations continue until no changes are made on clusters or stop at a certain iteration.

Finally, SFGC generates both graph clusters and HVSs as models. Figure 7 shows two sample HVSs generated from the graph in Figure 6. The significances of the graphic document cluster models are that (1) each model captures the core semantic relationship information about document clusters and provides the intrinsic meanings of them in a simple form; (2) this facilitates the interpretation of each cluster in terms of the key descriptors and could support the effective information retrieval.

Step3: Model-based document assignment

In this section, we explain how to assign each document to document clusters. In order to decide which document belongs to which document cluster, CSUGAR matches the graphical representation of each document with each of the graph clusters as models. Here, we might adopt graph similarity mechanisms, such as edit distance (the minimum number of primitive operations for structural modifications on a graph). However, these mechanisms are not appropriate for this task because individual document graphs and graph clusters are too different in terms of the number of vertices and edges. As an alternative to graph similarity mechanisms we take a vote mechanism. This mechanism is based on the classification (HVS or non-

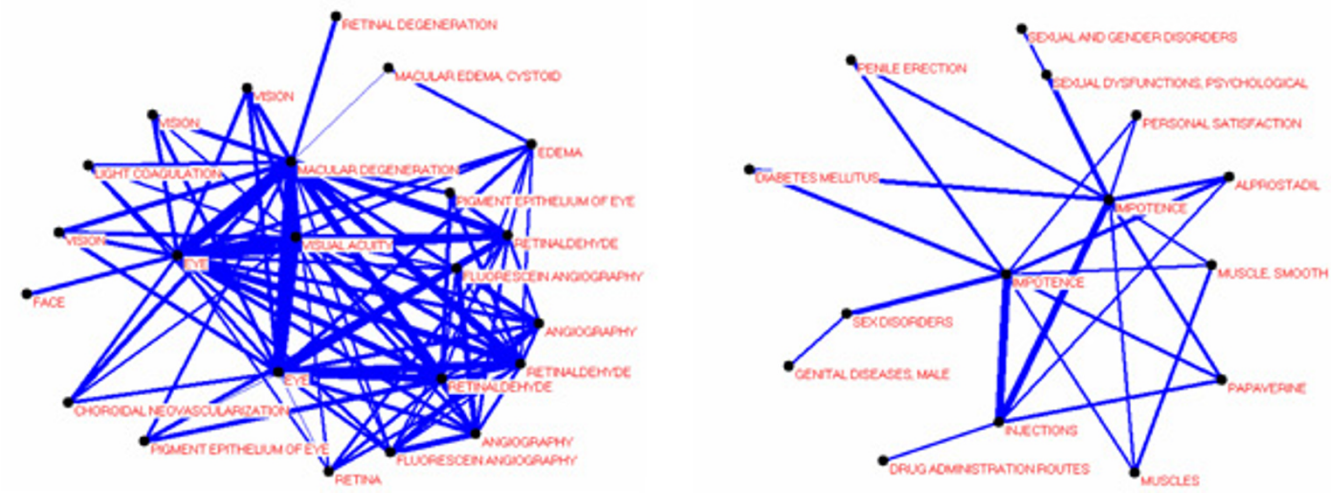

Figure 7
Two Sample Graphical Document Cluster Models from the Corpus-Level Graphical Representation in Figure 5.

HVS) of the vertices in the graph clusters according to their salient scores. This classification leads to different votes. To this end, each vertex of each individual document graph casts two different numbers of votes for document clusters based on whether the vertex belongs to HVS or non-HVS. Each document is assigned to the document cluster that has the majority of votes in the document clusters.

The next three steps correspond to text summarization. Text summarization is to condense information in a set of documents into a concise text. This text summarization problem has been addressed by selecting and ordering sentences in documents based on a salient score mechanism. We address the problem by analyzing the semantic interaction of sentences (as summary elements). This semantic structure of sentences is called Text Semantic Interaction Network (TSIN), where vertices are sentences. We select sentences (vertices in the network) as summary elements based on degree centrality. Unlike traditional approaches, we do not use linguistic features for summarization for MEDLINE abstracts since they usually consist of only single paragraphs.

Step4: Making ontology-enriched graphical representations for each sentence

The first step of the graphical representation for sentences is basically the same as the graphical representation method for documents except concept extension and individual graph integration. In this step the concepts in sentences are extended using the relationships in relevant document cluster models rather than the entire concept hierarchy. In other words, we extend concepts within relevant semantic field.

Step5: Constructing text semantic interaction network (TSIN)

Text summarization problem has been addressed by selecting and ordering sentences (or phrases) based on various salient score mechanisms. Thus, the key process of text summarization is how to select "salient" sentences (or paragraphs in some approaches) as summary elements. We assume that the sentences becoming summary have the strong semantic relationships with other sentences because summary sentences cover the main points of a set of documents and comprise a condensed version of the set. In order to represent the semantic relationship among sentences, we construct Text Semantic Interaction Network (TSIN), where vertices are sentences, edges are the semantic relationship between them, and edge weights indicate the degree of the relationships.

In order to deal with the semantic relationships between sentences and calculate the similarities (as edge weight in the network) between them, we use edit distance between the graphical representations of sentences. The edit distance between G1 and G2 is defined as the minimum number of structural modification required to become G1 into G2, where structural modification is one of vertex insertion, vertex deletion, and vertex update. For example, the edit distance between the two graphical representations of D_1 and D_2 in Figure 8 is 5.

Step6: Selecting significant text contents for summary

A number of approaches have been introduced to identify "important" nodes (vertices) in networks (or graphs) for decades. These approaches are normally categorized into degree centrality based approaches and between centrality based approaches. The degree centrality based approaches assume that nodes that have more relationships with others are more likely to be regarded as important in the network because they can directly relate to more other nodes. In other words, the more relationships the nodes in the network have, the more important they are. The betweenness centrality based approaches views a node as being in a favored position to the extent that the node falls on the geodesic paths between other pairs of nodes in the network [21]. In other words, the more nodes rely on a node to make connections with other nodes, the more important the node is.

These two approaches have their own advantages and disadvantages. Betweenness centrality based approaches yield better experiment results for small graphs to find cluster centroids than other relevant approaches, while they require cubic time complexity so that they are not appropriate for very large graphs. Degree centrality based approaches have been criticized because they only take into account the immediate relationships for each node while they require the linear time complexity and provide comparable output quality with betweenness centrality based approaches.

Because betweenness centrality cannot be applied to very large graphs due to its cubic time complexity, we adopt a well-known hyperlink ranking algorithm, Hypertext Induced Topic Search (HITS) [10], as a centrality measure in a graph. HITS was introduced by Kleinberg in 1998 [10]. HITS algorithm begins with the searching for user's query. The search result, consisting of relevant web pages, is defined as *Root Set*. Then, the *Root Set* is expanded to *Base Set* by adding two kinds of web pages; in-coming pages that have hyperlinks to the *Root Set* pages and outcoming pages that are hyperlinked from the *Root Set* pages.

After the input data set is collected, authority and hub scores are calculated for each web page. The authority score of a page is based on the hyperlinks "to" the page while the hub score is based on the links "from" the page. The calculation is based on the following observation:

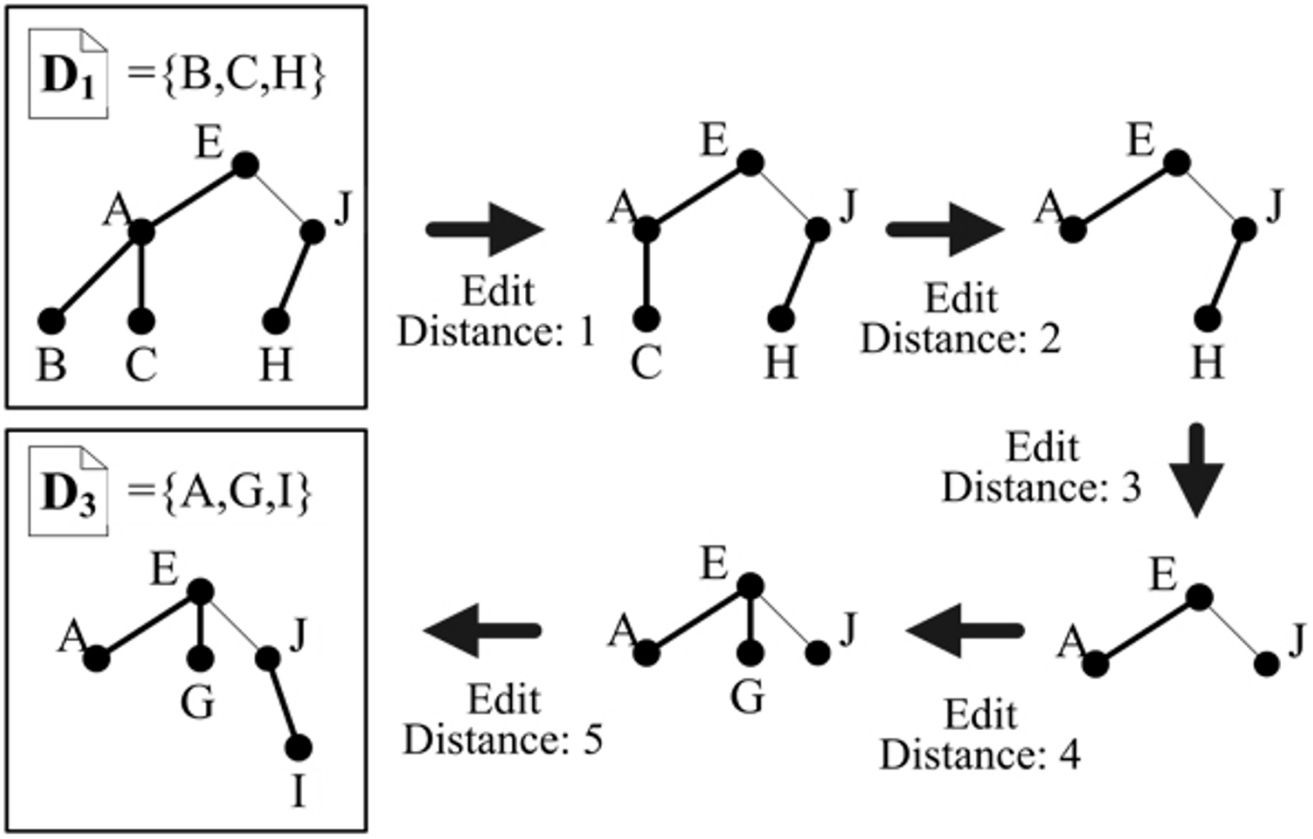

Figure 8
Edit Distance between Two Graphical Representations of D_1 and D_2.

• If a page has a good authority score, it is meant that many pages that have hyperlinks to the page have good hub scores.

• If a page has a good hub score, the page can give good authority scores to the pages that are hyperlinked by the page.

As they indicate, authority scores mutually reinforce hub scores. Based on these intuitions, for page i, authority score ($A(P_i)$) and hub scores ($H(p_i)$) are mathematically rendered as.

$$A(p_i) = \sum_{p_j \in \{p_j | Link(p_j \to p_i)\}} H(p_j)$$
$$H(p_i) = \sum_{p_j \in \{p_j | Link(p_i \to p_j)\}} A(p_j)$$

where, $Link(p_j \to p_i)$ implies page j (p_j) has a hyperlink to p_i,

These two iterative operations are performed for each web page; the authority score of each web page is updated with the sum of the hub scores of the web pages that are linked to the page and the hub score of each web page is updated with the sum of the authority scores of the web pages that link to the page. After these two operations are done in each web page, the authority and hub scores are normalized:

$$A(p_i) = \frac{A(p_i)}{\sqrt{\sum_i A(p_i)}} \quad H(p_i) = \frac{H(p_i)}{\sqrt{\sum_i H(p_i)}}$$

However, TSIN graph unlike hyperlinked web is an undirected graph so that we can unify authority score and hub score into node centrality ($C(N_i)$ for node i), which is mathematically rendered as

$$C(N_i) = \frac{C(N_i)}{\sqrt{\sum_i C(N_i)}}, \quad C(N_i) = \sum_{N_j \in \{N_j | Neighbor(N_j, N_i)\}} C(N_j)$$

where, $Neighbor\ (N_i, N_j)$ indicates nodes i and j are directly connected each other.

We call this simplified HITS as Mutual Refinement (MR) centrality here since the node centrality is recursively mutually refined. Because the node centrality mutually depends on one another, we provide each node with its degree centrality as an initial value. We will apply MR centrality as well as the degree centrality to measure the centrality of sentences in TSIN.

Competing interests

The authors declare that they have no competing interests.

Authors' contributions

IY conceived the initial idea, developed the methodology, implemented the CSUGAR and performed all the experiments on MEDLINE data sets. XH refined the proposed approach and supervised the implementation and the experiments. IYS provided suggestions in data analysis and improved and finalized the writing. All authors read and approved the final version of the manuscript.

Additional material

Additional file 1

Document Clustering Performance Decrease as Summary Compression Ratio Increase for Each Summarization Method
Click here for file
[http://www.biomedcentral.com/content/supplementary/1471-2105-8-S9-S4-S1.pdf]

Acknowledgements

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).

This article has been published as part of *BMC Bioinformatics* Volume 8 Supplement 9, 2007: First International Workshop on Text Mining in Bioinformatics (TMBio) 2006. The full contents of the supplement are available online at http://www.biomedcentral.com/1471-2105/8?issue=S9.

References

1. [http://www.ncbi.nlm.nih.gov].
2. Barabasi AL, Albert R: **Emergence of scaling in random networks.** *Science* 1999, **286:**509.
3. Ferrer-Cancho R, Solé RV: **The small world of human language.** *Proceedings of the Royal Society of London* 2001, **268:**2261-2266.
4. Ghosh J: **Scalable clustering methods for data mining.** In *Handbook of data mining* Edited by: *Ye N. Lawrence Erlbaum*; 2003.
5. Zeng Y, Tang J, Garcia-Frias J, Gao GR: **An Adaptive Meta-Clustering Approach: Combining The Information From Different Clustering Results.** *IEEE Computer Society Bioinformatics Conference (CSB2002)* 2002:276-287.
6. Hearst MA, Pedersen JO: **Reexamining the cluster hypothesis: Scatter/Gather on retrieval results.** *Proceedings of SIGIR-96* 1996:76-84.
7. Salton S, Singhal A, Mitra M, Buckley C: **Automatic text structuring and summarization.** In *Advances in Automatic Text Summarization* Edited by: *Mani I, Maybury MT. The MIT Press*; 1999:342-355.
8. Nomato T, Matsumoto Y: **Data Reliability and Its Effects on Automatic Abstracting.** *Proceedings of the 5th Workshop on Very Large Corpora: Beijing/Hong Kong, China* 1997.
9. Steinbach M, Karypis G, Kumar V: **A Comparison of Document Clustering Techniques.** In *Technical Report #00-034 University of Minnesota*; 2000.
10. Kleinberg J: **Authoritative Sources in a Hyperlinked Environment.** *Journal of the ACM* 1999, **46:**604-632.
11. Beil F, Ester M, Xu X: **Frequent Term-Based Text Clustering.** *Proceedings of 8th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining: 23–26 July 2002; Edmonton, Canada* 2002:436-442.
12. Pantel P, Lin D: **Document clustering with committees.** *Proceedings of the 2002 ACM SIGMOD International Conference on Management of data* 2002:199-206.
13. Erkan G, Radev D: **LexRank: Graph-based Lexical Centrality as Salience in Text Summarization.** *Journal of Artificial Intelligence Research* 2004, **22:**457-479.
14. Wu A, Garland M, Han J: **Mining Scale-free Networks using Geodesic Clustering.** *Proceedings of 10th ACM SIGKDD: 22–25 August 2004; Seattle, USA* 2004:436-442.
15. Rada R, Mili H, Bicknell E, Blettner M: **Development and application of a metric on semantic nets.** *IEEE Transactions on Systems, Man and Cybernetics* 1989:17-30.
16. Erdos P, Rényi A: **On the Evolution of Random Graphs.** *Publ Math Inst Hungar Acad Sci* 1960, **5:**17-61.
17. Amaral LAN, Scala A, Barthélémy M, Stanley HE: *Proc Nat Ac Sci* 2000, **97:**11149-11152.
18. Newman MEJ: **Fast algorithm for detecting community structure in networks.** *Phys Rev E* 2004, **69:**066133.
19. Newman MEJ: **Detecting community structure in networks.** *Eur Phys J B* 2004, **38:**321-330.
20. Wu A, Garland M, Han J: **Mining Scale-free Networks using Geodesic Clustering.** *Proceedings of 10th ACM SIGKDD:22–25 August 2004; Seattle* 2004:436-442.
21. Hanneman RA, Riddle M: **Introduction to social network methods [online].** 2005 [http://faculty.ucr.edu/~hanneman/]. *University of California*