A Simplicial Complex, a Hypergraph, Structure in the Latent Semantic Space of Document Clustering

Tsau Young ('T. Y.') Lin[a], I-Jen Chiang[b]

[a] *Department of Computer Science*
San Jose State University
One Washington Square
San Jose, CA 95192-0249

[b] *Graduate Institute of Medical Informatics*
Taipei Medical University
205 Wu-Hsien Street
Taipei, Taiwan 110

Abstract

This paper presents a novel approach to document clustering based on some geometric structure in *Combinatorial Topology*. Given a set of documents, the set of associations among frequently co-occurring terms in documents forms naturally a simplicial complex. Our general thesis is *each connected component* of this simplicial complex represents a concept in the collection. Based on these concepts, documents can be clustered into meaningful classes. However, in this paper, we attack a softer notion, instead of connected components, we use maximal simplexes of highest dimension as representative of connected components, the concept so defined is called maximal primitive concepts.

Experiments with three different data sets from web pages and medical literature have shown that the proposed unsupervised clustering approach performs significantly better than traditional clustering algorithms, such as *k-means*, *AutoClass* and *Hierarchical Clustering* (HAC). This abstract geometric model seems have captured the latent semantic structure of documents.

Key words: document clustering, association rules, topology, hierarchical clustering, simplicial complex

Email addresses: tylin@cs.sjsu.edu (Tsau Young ('T. Y.') Lin), ijchiang@tmu.edu.tw (I-Jen Chiang).

1 Introduction

Internet is an information ocean. Automatic tools are needed to help users find, filter, and extract the desired information. Search engines have become indispensable tools for gathering web pages and documents that are relevant to a user's query. Unfortunately, inconsistent, uninteresting and disorganized search results are often returned. Without conceptual categorization, issues like *polysemy*, *phrases* and *term dependency* impose limitations on search technology [22]. The goal of this paper is to improve the current state. Search results can be improved by proper organization based on categories, subjects, and contents.

How to organize the information ocean? Roughly speaking, we will organize the information by decomposed (triangulated, partitioned, granulated) the latent semantic space of documents into a simplicial complex (in combinatorial topology), which could be viewed a special form of hypergraphs. Note that the notion of simplicial complexes is actually predated that of hypergraphs about half a century, even though the latter notion is more familiar to modern computer scientists.

A good search engine needs to discriminate whether a piece of information is relevant to users' queries within a short time. In the current sate of art, to extract full semantic from a document automatically is Impossible. Given that multiple concepts can be simultaneously defined in a single Web page, and it is hard to limit the number of concept categories in a collection of Web pages. So some unsupervised clustering methods probably are best strategy. So we are proposing a technique, which is based on the triangulation of the latent semantic space of documents into a simplicial complex, to classify or cluster Web documents.

Clustering the documents by the associations (high frequent itemsets) of key terms that can be identified in a collection of documents naturally form a *simplicial complex* in combinatorial topology [41]. For example, the association that consists of "wall" and "street" denotes some financial notions that have meaning beyond the two nodes, "wall" and " street". This is similar to the notion of open segment (v_0, v_1)), in which two end points represent one dimensional geometric object that have meaning beyond the two 0-dimensional end points. In general, an r-association represents some semantic generated by a set of r keywords, may have more semantics or even have nothing to do with the individual keywords. The *Apriori property* of such associations is reflected exactly in the mathematical structure of simplicial complex in combinatorial topology (Section 4). We could regard such a structure as a triangulation (partition, granulation) of the space of latent semantics of web pages; The thesis of this paper is that a connected component of the simplicial complex of term-associations represents a CONCEPT in the conceptual structure of the latent semantic space of web pages. Based on the conceptual structure, the documents can be clustered. In this research, we investigate the strength of such a geometric view against traditional approaches, such as *k-means*, *AutoClass* [9] and *Hierarchical Clustering* (HAC) algorithms. The experimental results indicates that our approach is significantly stronger than classical approaches the performance.

In what follows, we start by reviewing related work on Web document clustering in section 2. Section 3 defines the association rules in a collection of documents and illustrates the way to compute the *support* and *confidence* of each association rule. The concepts and definitions of *latent semantic networks* based on geometric forms for the frequent itemsets generated by association rules are given in section 4. Section 5 presents the clustering algorithm for clustering a complex of concepts in a *latent semantic network* into several concrete concepts, each of which represents a concept in the document collection. Documents can then be clustered based on the primitive concepts identified by this algorithm. Experimental results from three different data sets are described in Section 6; followed by the conclusion.

2 Related Work

Most search engines provide instant gratification in response to user queries [8,31,36,42], however, they provide little guarantee on precision, even for detailed queries. There has been much research on developing more intelligent tools for information retrieval, such as machine learning [40], text mining and intelligent Web agents (see an earlier survey by Mladenic [33]).

Document clustering has been considered as one of the most crucial techniques for dealing with the diverse and large amount of information present on the World Wide Web- information ocean. In particular, clustering is used to discover latent concepts in a collection of Web documents, which is inherently useful in organizing, summarizing, disambiguating, and searching through large document collections [25].

Numerous document clustering methods have been proposed based on probabilistic models, distance and similarity measures [16], or other techniques, such as SOM [24]. A document is often represented as a feature vector, which can be viewed as a point in the multi-dimensional space. Many methods, including *k-means* [30], hierarchical clustering [20] and nearest-neighbor clustering [29] etc., select a set of key terms or phrases to organize the feature vectors corresponding to different documents. Suffix-tree clustering [44], a phrase-based approach, formed document clusters depending on the similarity between documents.

When the number of features selected from each document is too large, methods for extracting the salient features are taken. However, the residual dimension can still be very large, moreover the quality of the resulting clusters tends to decrease due to the loss of relevant features. Common frameworks for reducing the dimension of the feature space are principle component analysis [21], independent component analysis [19], and latent semantic indexing [3,5]. Furthermore, in the presence of noise in the data, feature extraction may result in degradation of clustering quality [6]. In that paper, association rule hypergraph partition was first proposed in [6] to transform documents into a transactional database form, and then apply hypergraph partitioning [23] to find the item clusters.

Hierarchical clustering algorithms have been proposed in an early paper by Willett [43]. Cutting et al. introduced partition-based clustering algorithms for document clustering [11]. Buckshot and fractionation were developed in [27]. Greedy heuristic methods are used in the hierarchical frequent term-based clustering algorithm [4] to perform hierarchical document clustering by using frequent itemsets. We should note here that frequent itemsets are also referred to as associations (undirected association rules).

3 Undirected Term-Associations

The notion of association rules was introduced by Agrawal et al. [1] and has been demonstrated to be useful in several domains [7,10],such as retail sales transaction database. In the theory two standard measures, called *support* and *confidence*, are often used. We will be concerned more on the frequency than direction of rules. Our focus will be on the support; a set of items that meets the support is often referred to as frequent itemsets; we will call them *associations* (undirected association rules) as to emphasize more on their meaning than the phenomena of frequency.

The frequency distribution of a word or phrase in a document collection is quite different from the item frequency distribution in a retail sales transaction database. In [28], we have shown that isomorphic relations have isomorphic associations. Documents are amorphous. Isomorphic essentially means identical. A single key word does not carry much information about a document, yet a huge amount of key words may nearly identify the document uniquely. So finding all associations in a collection of textual documents presents a great interest and challenge.

Traditional text mining generally consists of the analysis of a text document by extracting key words, phrases, concepts, etc. and representing in an intermediate form refined from the original text in that manner for further analysis with data mining techniques (e.g., to determine associations of concepts, key phrases, names, addresses, product names, etc.). Feldman and his colleagues [12,13,15] proposed the *KDT* and *FACT* system to discover association rules based on keywords labeling the documents, the background knowledge of keywords and relationships between them. This is not an effective approach, because a substantially large amount of background knowledge is required. Therefore, an automated approach that documents are labeled by the rules learned from labeled documents are adopted [26]. However, several association rules are constructed by a compound word (such as "Wall" and "Street" often co-occur) [37]. Feldman et al. [12,14] further proposed term extraction modules to generate association rules by selected key words. Nevertheless, a system without the needs of human labeling is desirable. Holt and Chung [18] addressed Multipass-Apriori and Multipass-DHP algorithms to efficiently find association rules in text by modified the Apriori algorithm [2] and the DHP algorithm [35] respectively. However, these methods did not consider about the word distribution in a document, that is, identify the importance of a word in a document.

According to the trivial definition of distance measure in this space, no matter what kind of a method is, some common words are more frequent in a document than other words. Simple frequency of the occurrence of words is not adequate, as some documents are larger than others. Furthermore, some words may occur frequently across documents. In most cases, words appeared in a few documents tend to most "important." Techniques such as TFIDF [39] have been proposed directly to deal with some of these problems. The TFIDF value is the weight of term in each document. While considering relevant documents to a search query, if the TFIDF value of a term is large, then it will pull more weight than terms with lesser TFIDF values.

3.1 Feature Extraction

A general framework for text mining consists of two phases. The first phase, *feature extraction*, is to extract key terms from a collection of "indexed" documents; as a second step various methods such as association rules algorithms may be applied to determine relations between features.

While performing association analyses on a collection of documents, all documents should be indexed and stored in an intermediate form. Document indexing is originated from the task of assigning terms to documents for retrieval or extraction purposes. In early approach, an indexing model was developed based on the assumption that a document should be assigned those terms that are used by queries to retrieve the relavent document [32], [17]. The weighted indexing is the weighting of the index terms with respect to the document with this model given a theoretical justification in terms of probabilities. The most simple and sophisticated weighted schema which is most common used in information retrieval or information extraction is TFIDF indexing, i.e., tf × idf indexing [39,38], where tf denotes term frequency that appears in the document and idf denotes inverse document frequency where document frequency is the number of documents which contain the term. It takes effect on the commonly used word a relatively small tf × idf value. Moffat and Zobel [34] pointed out that tf × idf function demonstrates: (1) rare terms are no less important than frequent terms in according to their idf values; (2) multiple appearances of a term in a document are no less important than single appearances in according to their tf values. The tf × idf implies the significance of a term in a document, which can be defined as follows.

Definition 1 *Let T_r denote a collection of documents. The significance of a term t_i in a document d_j in T_r is its TFIDF value calculated by the function* $\mathrm{tfidf(t_i, d_j)}$*, which is equivalent to the value* $\mathrm{tf(t_i, d_j)} \times \mathrm{idf(t_i, d_j)}$. *It can be calculated as*

$$\mathrm{tfidf(t_i, d_j) = tf(t_i, d_j) \log \frac{|T_r|}{|T_r(t_i)|}}$$

where $|T_r(t_i)|$ denotes the number of documents in T_r in which t_i occurs at least once, and

$$\mathrm{tf(t_i, d_j)} = \begin{cases} 1 + \log(N(t_i, d_j)) \textit{ if } N(t_i, d_j) > 0 \\ \\ 0 \qquad\qquad\qquad\qquad \textit{otherwise} \end{cases}$$

where $N(t_i, d_j)$ denotes the frequency of terms t_i occurs in document d_j by counting all its nonstop words.

To prevent the value of $|T_r(t_i)|$ to be zero, *Laplace Adjustment* is taken to add an observed count.

TFIDF values are often organized into the following matrix form: Let a document d_j in T_r be represented as a vector $V_j = <$ $\mathrm{tfidf(t_1, d_j)}$, $\mathrm{tfidf(t_2, d_j)}$, $\cdots$, $\mathrm{tfidf(t_n, d_j)}$ $>$ and therefore T_r be represented as a matrix $M_r = <V_1$, V_2, $\cdots$, V_I, $\cdots >^T$. Most previous works [12,13,15] proposed to finding the association rules or partitioning the association rules into clusters [6] from M_r. However, there are often more than thousands of terms in a document and some terms may appear only in a few documents of a collection. The document matrix M_r

is large and sparse. It becomes computationally hard to find the independent sets of association rules for automatic clustering of the documents.

3.2 Measures on Undirected Term-Associations

We observed that the direction of key terms is irrelevant information for the purpose of document clustering. So we ignore the *confidence* and consider only the *support*. In other words, we consider the structure of the *undirected* associations of key terms; we believe the set of key terms that co-occur reflects the essential information, the rule directions of the key terms are inessential, at least in the present stage of investigation. Let t_A and t_B be two terms. The *support* is defined for a collection of documents as follows.

Definition 2 *The significance of undirected associations of term t_A and term t_B in a collection is:*

$$\mathrm{tfidf}(\mathrm{t_A}, \mathrm{t_B}, \mathrm{T_r}) = \frac{1}{|\mathrm{T_r}|} \sum_{\mathrm{i}=0}^{|\mathrm{T_r}|} \mathrm{tfidf}(\mathrm{t_A}, \mathrm{t_B}, \mathrm{d_i})$$

where

$$\mathrm{tfidf}(\mathrm{t_A}, \mathrm{t_B}, \mathrm{d_i}) = \mathrm{tf}(\mathrm{t_A}, \mathrm{t_B}, \mathrm{d_i}) \log \frac{|\mathrm{T_r}|}{|\mathrm{T_r}(\mathrm{t_A}, \mathrm{t_B})|}$$

and $|T_r(t_A, t_B)|$ define number of documents contained both term t_A and term t_B.

The term frequency $\mathrm{tf}(\mathrm{t_A}, \mathrm{t_B}, \mathrm{d_i})$ of both term t_A and t_B can be calculated as follows.

Definition 3 $$\mathrm{tf}(\mathrm{t_A}, \mathrm{t_B}, \mathrm{d_j}) = \begin{cases} 1 + \log(\min\{N(t_A, d_j), N(t_B, d_j)\}) \\ \textit{if } N(t_A, d_j) > 0 \textit{ and } N(t_B, d_j) > 0 \\ \\ 0 \\ \textit{otherwise.} \end{cases}$$

A minimal support θ is imposed to filter out the terms that their TFIDF values are small. It helps us to eliminate the most common terms in a collection and the nonspecific terms in a document.

Let t_A and t_B be two terms. The *support* and *confidence* defined in the document matrix M_r is as follows.

Definition 4 Support *denotes to be the ratio of the number of documents in T_r that contains both term t_A and term t_B, that is,*

$$\mathrm{Support(t_A, t_B) = \frac{|T_r(t_A, t_B)|}{|T_r|}}$$

where $|T_r(t_A, t_B)|$ is the number of the documents that contains both t_A and t_B.

Definition 5 *The* confidence *is obtained from* tfidf *of both t_A and t_B, which denotes the* score *of documents that contains t_A and also contain t_B within a fixed distance.*

$$\mathrm{Confidence(t_A, t_B) = P(t_B|t_A) = \frac{tfidf(t_A, t_B, T_r)}{tfidf(t_A, T_r)}}$$

where

$$\mathrm{tfidf(t_A, T_r) = \frac{1}{|T_r|}\sum_{i=0}^{T_r} tfidf(t_A, d_i)}$$

and

$$\mathrm{tfidf(t_A, t_B, T_r) = \frac{1}{|T_r|}\sum T_{ri=0} tfidf(t_A, t_B, d_i)}$$

where

$$\mathrm{tfidf(t_A, t_B, d_i) = tf(t_A, t_B, d_i)\log\frac{|T_r|}{|T_r(t_A, t_B|)}}$$

where $|T_r(t_A, t_B|$ is number of documents contained both term t_A and term t_B and

$$\mathrm{tf(t_A, t_B, d_j)} = \begin{cases} 1 + \log(\min\{N(t_A, d_j), N(t_B, d_j)\}) \\ \textit{if } N(t_A, d_j) > 0 \textit{ and } N(t_B, d_j) > 0 \\ 0 \\ \textit{otherwise.} \end{cases}$$

The terms with lower *confidences* than a given threshold, i.e., *minimum confidence*, from the origin matrix M_r are filtered to be the condensed matrix $\hat{M}_r$. There are a lot of algorithms developed for discovery association rules discussed in the previous section, such as *Apriori* [1], have been used to discover association rules in $\hat{M}_r$. The discovered association rules can then be taken as these clusters of items. ms.

4 Geometric Representations of Term-Associations

The goal of this section is to model the internal CONCEPTS that are hidden in a collection of documents. We observe that (1) term-term inter-relationships represent and carry the intrinsic semantics or CONCEPTS hidden in a collection of documents, and (2) the co-occurred term associations, will be called term-associations, represent the term-term inter-relationships. So key to model the hidden semantics or CONCEPTS in a set of documents is lied in modeling the term-associations. Somewhat a surprise, the mathematical structure of term-associations is a known geometric/topological subject, called simplicial complex.

So a natural way to represent the latent semantic in a set of documents is to use geometric and topologic notions that capture the totality of thoughts expressed in this collection of documents.

4.1 Combinatorial Topology

Let us introduce and define some basic notions in combinatorial topology. The central notion is n-simplex.

Definition 6 *A n-simplex is a set of independent abstract vertices $[v_0, \ldots, v_{n+1}]$. A r-face of a n-simplex $[v_0, \ldots, v_{n+1}]$ is a r-simplex $[v_{j_0}, \ldots, v_{j_{r+1}}]$ whose vertices are a subset of $\{\ v_0, \ldots, v_{n+1}\ \}$ with cardinality $r+1$.*

Geometrically 0-simplex is a vertex; 1-simplex is an open segment (v_0, v_1) that does not include its end points ; 2-simplex is an open triangle (v_0, v_1, v_2) that does not include its edges and vertices; 3-simplex is an open tetrahedron (v_0, v_1, v_2, v_3) that does not includes all the boundaries. For each simplex, all its proper faces (boundaries) are not included. An n-simplex is the high dimensional analogy of those low dimensional simplexes (segment, triangle, and tetrahedron)in n-space. Geometrically, an n-simplex uniquely determines a set of $n+1$ linearly independent vertices, and vice versa. An n-simplex is the smallest convex set in a Euclidean space R^n that contains $n+1$ points v_0

. . ., v_n that do not lie in a hyperplane of dimension less than n. For example, there is the standard n-simplex

$$\delta^n = \{(t_0, t_1, \ldots t_{n+1}) \in R^{n+1} \mid \sum_i t_i = 1, t_i \geq 0\}$$

The convex hull of any m vertices of the n-simplex is called an m-face. The 0-faces are the vertices, the 1-faces are the edges, 2-faces are the triangles, and the single n-face is the whole n-simplex itself. Formally,

Definition 7 *A simplicial complex C is a finite set of simplexes that satisfies the following two conditions:*

- *Any set consisting of one vertex is a simplex.*
- *Any face of a simplex from a complex is also in this complex.*

The vertices of the complex v_0, v_1, $\cdots$, v_n is the union of all vertices of those simplexes ([41], pp. 108).

If the maximal dimension of the constituting simplexes is n then the complex is called n-complex.

Note that, any set of $n + 1$ objects can be viewed as a set of abstract vertices, to stress this abstractness, some times we refer to such a simplex a combinatorial n-simplex. The corresponding notion of combinatorial n-complex can be defined by (combinatorial) r-simplexes. Now, by regarding the key terms, as defined by high TFIDT values, as abstract vertices, an association of $n+1$ key terms, called $n + 1$-association, is a combinatorial n-simplex: A 2-association is an open 1-simplex. An open 1-simplex ("wall", "street") represents a financial notion that includes some semantics that is well beyond the two vertices, "wall" and "street." A $(n+1)$-association is a combinatorial n-simplex of keywords that often carries some deep semantics that are well beyond the "union" of its vertices, or faces individually.

We need much more precise notions. A (n, r)-skeleton (denoted by S^n_r) of n-complex is a n-complex, in which all k-simplexes($k \leq r$) have been removed. Two simplexes in a complex are said to be *directly connected* if the intersection of them is a nonempty face. Two simplexes in a complex are said to be *connected* if there is a finite sequence of directly connected simplexes connecting them. For any non-empty two simplexes A, B are said to be *r-connected* if there exits a sequence of k-simplexes $A = S_0, S_1, \ldots, S_m = B$ such that S_j and S_{j+1} has an h-common face for $j = 0, 1, 2, \ldots, m - 1$; where $r \leq h \leq k \leq n$.

The maximal r-connected subcomplex is called a *r-connected component.* Note that a r-connected component implies there does not exist any r-connected component that is the superset of it. A maximal r-connected sub-complexes of n-complex is called r-connected component. A maximal r-connected component of n-complex is called connected component, if $r = 0$.

4.2 The Geometry of Term-Associations

In the last section, we have observed that a $n+1$-association is an abstract n-simplex, in fact, the set of all associations has more structures. In this section, we will investigate the mathematical structures of term-associations. First let us recall the notion of hypergraph:

Definition 8 *A* hypergraph $G = (V, E)$ *contains two distinct sets where* V *is a finite set of abstract vertices, and* $E = \{e_1, e_2, \cdots, e_m\}$ *is a nonempty family of subsets from* V*, in which each subset is called a* hyperedge*.*

It is obvious that the set of association can be interpreted as a hypergraph: The key terms are the vertices, the term-associations are hyperedges. Likewise, a simplicial complex is a hypergraph: the set of vertices is V, and the set of simplexes is E. However, both term-associations and simplicial complex has more structures. A simplicial complex satisfies further conditions that are specified in last section. Simplicial complex is a very special kind of hypergraphs. Actually the differences are deeper and intrinsic:

- A hypergraph theory targets on the graph theoretical structure of vertices that are connected by hyperedges.
- A simplicial complex (combinatorial topology) targets on the geometrical or topological structure of the spaces (polyhedron) that are supported by simplicial complex.

Note that the Apriori conditions on term-associations meet the conditions of the simplicial complex: an 1-association is the 0-simplex, and a "subset" of an association is an association of shorter lengths. So the notion of simplicial complex is a natural view of term-associations. We will *take this view.*

In our application each vertex is a key term, a simplex is a term-association of maximal length. The open 1-simplex (Wall, Street) represents a concept in financial business. The 0-simplex (Network) might represent many different CONCEPTS, however, while it is combined with some other terms would denote further semantic CONCEPTS. For example, the following 1-simplexes (Computer, Network), (Traffic, Network), (Neural, Network), (Communication, Network), and etc., express further and richer semantic than their individual 0-simplexes. Of course, the 1-simplex (Neural, Network) is not conspicuous than the 2-simplexes (Artificial Neural Network) and (Biology, Neural, Network).

A collection of documents may carry a set of distinct CONCEPTS. Each concept, we believe, is carried by a connected component of the complex of term-associations. Here is our belief and our thesis:

- An IDEA (in the forms of complex of term-associations) may consist many CONCEPTS (in the form of connected components) that consists of PRIMITIVE CONCEPTS (in the form of maximal simplexes). The maximal simplexes of highest dimension is called MAXMIAL PRIMITIVE CONCEPT. A simplex is said to be a maximal if no other simplex in the complex is a superset of it. The geometric dimension represents the degree of preciseness or depth of the latent semantics that are represented by term-associations.

Example 1 *In Figure 4, we have an idea that consist of twelve terms that organized in the forms of* 3*-complex, denoted by* S^3*. Simplex*(a,b,c,d) *and Simplex*(w,x,y,z) *are two maximal simplexes of* 3*, the highest dimension. Let us consider* S_1^3*. It is the leftover from the removal of all* 0*-simplexes from* S^3*:*

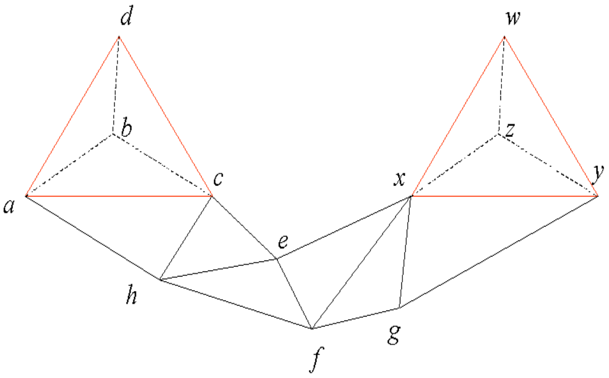


Fig. 1. A complex with twelve vertexes.

- Simplex(a, b, c, d) *and its ten faces:*
  - · Simplex(a, b, c)
  - · Simplex(a, b, d)
  - · Simplex(a, c, d)
  - · Simplex(b, c, d)
  - · Simplex(a, b)
  - · Simplex(a, c)
  - · Simplex(b, c)
  - · Simplex(a, d)
  - · Simplex(b, d)
  - · Simplex(c, d)
- Simplex(a, c, h) *and its three faces:*
  - · Simplex(a, c)
  - · Simplex(a, h)
  - · Simplex(c, h)

- $\mathrm{Simplex(c, h, e)}$ *and its three faces:*
  - · $\mathrm{Simplex(c, h)}$
  - · $\mathrm{Simplex(h, e)}$
  - · $\mathrm{Simplex(c, e)}$
- $\mathrm{Simplex(e, h, f)}$ *and its three faces:*
  - · $\mathrm{Simplex(e, h)}$
  - · $\mathrm{Simplex(h, f)}$
  - · $\mathrm{Simplex(e, f)}$
- $\mathrm{Simplex(e, f, x)}$ *and its three faces:*
  - · $\mathrm{Simplex(e, f)}$
  - · $\mathrm{Simplex(e, x)}$
  - · $\mathrm{Simplex(f, x)}$
- $\mathrm{Simplex(f, g, x)}$ *and its three faces:*
  - · $\mathrm{Simplex(f, g)}$
  - · $\mathrm{Simplex(g, x)}$
  - · $\mathrm{Simplex(f, x)}$
- $\mathrm{Simplex(g, x, y)}$ *and its three faces:*
  - · $\mathrm{Simplex(g, x)}$
  - · $\mathrm{Simplex(g, y)}$
  - · $\mathrm{Simplex(x, y)}$
- $\mathrm{Simplex(w, x, y, z)}$ *and its ten faces:*
  - · $\mathrm{Simplex(w, x, y)}$
  - · $\mathrm{Simplex(w, x, z)}$
  - · $\mathrm{Simplex(w, y, z)}$
  - · $\mathrm{Simplex(x, y, z)}$
  - · $\mathrm{Simplex(w, x)}$
  - · $\mathrm{Simplex(w, y)}$
  - · $\mathrm{Simplex(w, z)}$
  - · $\mathrm{Simplex(x, y)}$
  - · $\mathrm{Simplex(x, z)}$
  - · $\mathrm{Simplex(y, z)}$

Note that $\mathrm{Simplex(a, c)}$, $\mathrm{Simplex(c, h)}$, $\mathrm{Simplex(h, e)}$, $\mathrm{Simplex(e, f)}$, $\mathrm{Simplex(f, x)}$, $\mathrm{Simplex(g, x)}$*, and* $\mathrm{Simplex(x, y)}$ *all have common faces. So they generate a connected path from* $\mathrm{Simplex(a, b, c, d)}$ *to* $\mathrm{Simplex(w, x, y, z)}$*, and sub-paths. Therefore the S^3_1 complex is connected. This assertion also implies that S^3 is connected. Hence the IDEA consists of a single CONCEPT(please note the technical meaning of the IDEA and CONCEPT given above). Next, let us consider the* $(3, 2)$*-skeleton S^3_2, by removing all* 0*-simplexes and* 1*-simplexes from S^3:*

- $\mathrm{Simplex(a, b, c, d)}$ *and its four faces:*
  - · $\mathrm{Simplex(a, b, c)}$
  - · $\mathrm{Simplex(a, b, d)}$
  - · $\mathrm{Simplex(a, c, d)}$

  - $\cdot$ $\mathrm{Simplex(b, c, d)}$
- $\mathrm{Simplex(a, c, h)}$
- $\mathrm{Simplex(c, h, e)}$
- $\mathrm{Simplex(e, h, f)}$
- $\mathrm{Simplex(e, f, x)}$
- $\mathrm{Simplex(f, g, x)}$
- $\mathrm{Simplex(g, x, y)}$
- $\mathrm{Simplex(w, x, y, z)}$ *and its four faces:*
  - $\cdot$ $\mathrm{Simplex(w, x, y)}$
  - $\cdot$ $\mathrm{Simplex(w, x, z)}$
  - $\cdot$ $\mathrm{Simplex(w, y, z)}$
  - $\cdot$ $\mathrm{Simplex(x, y, z)}$

There are no common faces between any two simplexes, so S^3_2 has eight connected components, or eight CONCEPTS. For S^3_3, it consists of two non-connected 3*-simplexes or two MAXIMAL PRIMITIVE CONCEPTS.*

A complex, connected component or simplex of a skeleton represent a more technically refined IDEA, CONCEPT or PRIMITVE CONCEPT. If a maximal connected component of a skeleton contains only one simplex, this component is said to organize a primitive concept.

Definition 9 *A set of maximal connected components is said to be* independent *if there are no common faces between any two maximal connected components.*

4.3 Layered Clustering

From a collection of documents, a complex of term-associations can be generated. Based on such complex, document can be clustered in layer fashions.

In this section, we will first examine the intuitive meaning of such complex. As seen in Example 1, connected components in S^n_k are contained in S^n_r, where $k \geq r$. Based on that, the goal of this paper is to define the layered clustering based on the dimension hierarchies of primitive CONCEPTS.

Example 2 *Figure 2 is* 2*-complex composed of the term set V={t_A, t_B, t_C} in a collection of documents. It is a close* 2*-simplex; we recall here that a closed simplex is a complex that consists of one simplex and all its faces. In the skeleton S^2_1, all* 0*-simplexes are ignored, i.e., the terms depicted in dash lines. The simplex set S =*{$\mathrm{Simplex}^2_1$, $\mathrm{Simplex}^1_2$, $\mathrm{Simplex}^1_3$, $\mathrm{Simplex}^1_4$} *is the closed* 2*-simplex that consists of one* 2*-simplex and three 1-faces,* $\mathrm{Simplex}^1_2$, $\mathrm{Simplex}^1_3$ *and* $\mathrm{Simplex}^1_4$ *(*0*-faces are ignored). These r-simplexes (*$0 \leq r \leq 2$*) represents frequent itemsets (term-associations) from V, where W =*{$w_{A,B}$,

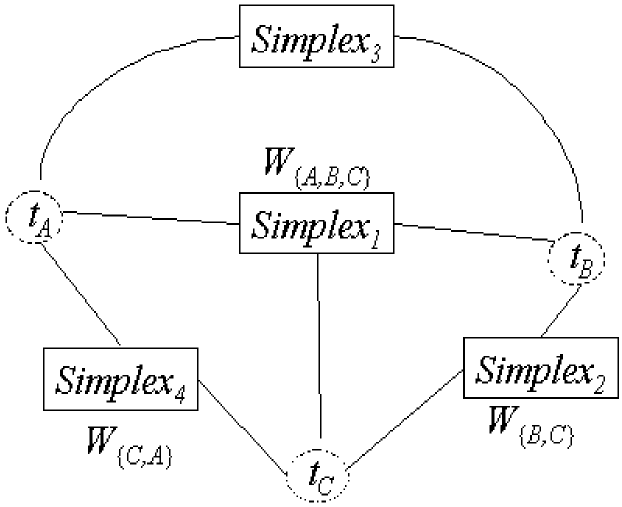


Fig. 2. This figure illustrates the skeleton S_1^3 of Example 2. It is composed from three key terms $\{t_A,\ t_B,\ t_C\}$ of a collection of documents, where each simplex is identified by its tfidf value and all 0-simplexes have been removed (the nodes are drawn by using dash circles). Note that Simplex$_1$ has dimension 2, we draw its incidences with three vertices, but skip the incidences with three 1-simplexes.

$w_{C,A}$, $w_{B,C}$, $w_{A,B,C}\}$ denote their corresponding supports. The lines connecting Simplex$_1$ *and three vertices represent the incidences of* 2*-simplex and* 0*-simplex; the incidences with* 1*-simplexes are not shown to avoid overcrowding the figure.*

One of the geometric property of simplicial complex is all faces of a simplex, that is in the complex, has to be in the complex:

Property 1 *A simplex has $\binom{n+1}{i+1}$ i-faces ($i \leq n$), where $\binom{n}{k}$ is a binomial coefficient. This is the* apriori *condition of association rules.*

So, as we have observed previously, that in a complex of term-associations, the set of 0-simplexes (vertices) represents all frequent 1-itemsets, 1-simplexes frequent 2-itemsets and 2-simplexes frequent 3-itemsets, and so on.

According to Example 1, it is obvious that simplexes within the higher level skeleton S_r^n is contained in the lower level skeleton S_k^n within the same n-complex, $r \geq k$. Figure 3 shows the hierarchy, each skeleton is represented as a layer. For the purpose of simplicity, we skip the middle layer, namely, S_r^n, $0 \leq r < 3$, are not shown.

By considering different skeletons, we can draw distinct layer of CONCEPTS:

(1) In full complex $S = S_0^n$, this example only has one CONCEPT (one connected component).
(2) In S_1^n, this complex still has only one CONCEPT.
(3) In S_2^n, this complex has eight CONCEPTS.

(4) in S_3^n, this complex has two CONCEPTS; they are two MAXIMAL PRIMITVE CONCEPTS.

For each choice, say S_2^n, we have, in this case, eight CONCEPTS to label the documents (or clustered the documents). A document is labeled $CONCEPT_k$, if the document has high TFITD values on the term-associations that defines $CONCEPT_k$. By consider different cases, we have layered clusters. In fact, we even could consider a very coarse clustering that is, we consider only the MAXIMAL PRIMITVE CONCEPTS; this is the case of S_3^n. For the purpose of illustrating the methodology, we have focused on this "over simplified" one.

In general, the simplexes at the lower layers could have common faces between them. Therefore, to use all layers of CONCEPTS at the same time will produce vague discrimination as shown in Figure 4, in which an overlapped CONCEPTS induced by (lower dimensional) common faces could exist. As seen in the skeleton S_1^3, the maximal connected components generated from simplex Simplex(a, b, c, d) and simplex Simplex(a, c, h) have a common face Simplex(a, c) that makes some documents not able to properly discriminated in accordance with the generated association rules from term a and term c, so are the other maximal connected components in the skeleton. Because of the intersection produced by such faces, a proper way is to ignore the lower the skeleton as much as application can tolerate.

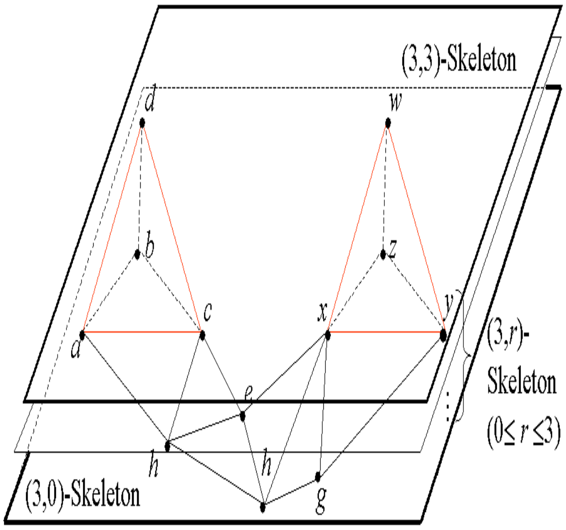


Fig. 3. This figure illustrates the layer structures of Example 1. The top layer is skeleton (3, 3)-Skeleton that has two distinct CONCEPTS Simplex(a, b, c, d) and Simplex(w, x, y, z). The middle layer (3, 2)-Skeleton has 8 CONCEPTS; it is not illustrate here. The layer (3, 1)-Skeleton is skipped. The bottom layer (3, 0-Skeleton contains only one connected component; it is shown in the figure.

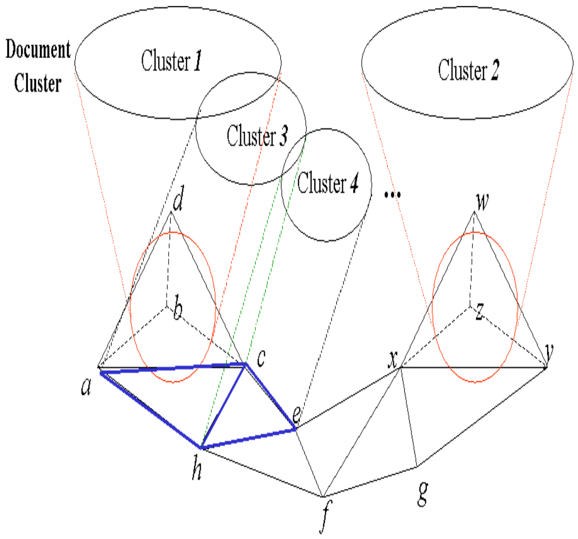


Fig. 4. Each cluster of documents is identified by a maximal connected component. Some clusters may overlap with other cluster because of the common face between them; this phenomenon is illustrated here. To handle such a situation properly, we need to ignore the lower dimensional simplexes. By so doing the overlapping will disappear (not shown).

5 Finding Maximal Connected Components

We can visualize that the latent semantic of a collection of documents is a space triangulated/partitioned/granulated by term-associations (simplexes). The space contains CONCEPTS, PRIMITIVE CONCEPTS. We have observed that combinatorial geometry is an effective theory for modeling the latent semantics space of a huge variety of high-dimensional data, such as document collection, or bioinformatics data. The algorithms for finding all CONCEPTS, i.e., maximal connected components in the complex of term-associations will be introduced below; In fact, we will focus on "over simplified" version, namely, on the complex S_n^n. In other words, maximal PRIMITVE CONCEPTS (highest dimension).

5.1 Incidence Matrices

First, we need some geometric notations.

Definition 10 *In a simplicial topology, let $\mathcal{V}$ be the set of (individual) key terms in a collection of documents, i.e., 0-simplices, and $\mathcal{E}$ be the set of all r-simplices, where $r \geq 0$. If* Simplex$_{\mathrm{A}}$ *is in $\mathcal{E}$, its support is defined as* $w_($Simplex$_{\mathrm{A}})$*, i.e., the* tfidf *of the simplex* Simplex$_{\mathrm{A}}$ *of term-association.*

The *incident matrix* and the *weighted incident matrix* of a complex can be defined as follows; here we are more interested in the case $\mathrm{Simplex_i}$ is a 0-face.

Definition 11 *The* $n \times m$ incident matrix $A = (a_{ij})$ *associated to a complex is defined as*

$$a_{ij} = \begin{cases} 1 \ \textit{if} \ \mathrm{Simplex_i is a face of Simplex_j} \\ 0 \ \textit{otherwise.} \end{cases}$$

The corresponding weight incident matrix $A' = (a'_{ij})$ *is*

$$a'_{ij} = \begin{cases} w_{ij} \ \textit{if} \ \mathrm{Simplex_i is a face of Simplex_j} \\ 0 \quad \textit{otherwise} \end{cases}$$

where the weight w_{ij} *denotes the* support *of a term-association.*

Example 3 *As seen in Example 2, the* 2*-simplex is the set* $\{t_A, t_B, t_C\}$*, which is also the maximal connected component that represents a concept in a document collection. Based on the Venn diagram of this complex, the incident matrix I and the weighted incident matrix* I_W *of the simplexes can be constructed. For clarity, we only illustrate the incidences between the key terms (*0*-simplexes) and term-associations (r-simplexes, r=1,2) as follows.*

$$I = \begin{pmatrix} 1\,0\,1\,1 \\ 1\,1\,1\,0 \\ 1\,1\,0\,1 \end{pmatrix}.$$

$$I_W = \begin{pmatrix} w_{A,B,C} & 0 & w_{A,B} & w_{C,A} \\ w_{A,B,C} & w_{B,C} & w_{A,B} & 0 \\ w_{A,B,C} & w_{B,C} & 0 & w_{C,A} \end{pmatrix}.$$

Each row represents the incidence of a vertex with r-simplexes. Each column corresponds to the incidence of a fixed simplex and all vertices.

5.2 Algorithm

As we already known, a r-simplex is a $(r+1)$-term-association (frequent $(r+1)$-itemset). Documents can be clustered based maximal simplexes of highest dimension (MAXIMAL PRIMITIVE CONCEPTS), namely, the longest associations. Note that documents clustered by MXAIMAL PRIMITVE CONCEPTS contains common lower dimensional faces (shorter associations, in particular 0-simplexes); this is consequence of apriori property. In this sense, the methodology provides a soft approach; we allow lower dimensional overlapped CONCEPTS exist within different clusters. Considering Example 4, two maximal 2-simplexes in the skeleton S_3^3 produce two MAXIMAL PRIMITIVE CONCEPTS with common 0-face.

Example 4 *As shown in Figure 5, one component is organized by the simplex* $\mathrm{Simplex}_1 = \{t_A, t_B, t_C\}$*, the other is generated by the simplex* $\mathrm{Simplex}_5 = \{t_C, t_D, t_E\}$*. The incident matrix is: (5 vertices × 8 simplexes)*

$$\begin{pmatrix} 1&1&0&1&0&0&0&0 \\ 1&1&1&0&0&0&0&0 \\ 1&0&1&1&1&1&1&0 \\ 0&0&0&0&1&1&0&1 \\ 0&0&0&0&1&0&1&1 \end{pmatrix}.$$

Both simplexes share a common concept 0*-simplex* $\{t_C\}$*, which is an* 1*-item frequent itemset* $\{t_C\}$*.*

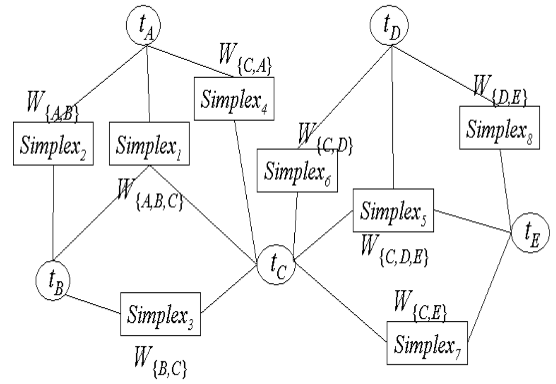


Fig. 5. The complex is composed of two maximal 2-simplexes $\mathrm{Simplex}_1 = \mathrm{Simplex}(\mathrm{t_A}, \mathrm{t_B}, \mathrm{t_C})$ and $\mathrm{Simplex}_5 = \mathrm{Simplex}(\mathrm{t_C}, \mathrm{t_D}, \mathrm{t_E})$. Both of them contain a common face $\mathrm{Simplex}(\mathrm{t_C})$ that produces an undiscriminating concept region.

Since the intersection of connected components has lower dimensions. It is convenient for us to design an efficient algorithm for documents clustering in a skeleton by skeleton fashion. The algorithm for finding all maximal connected components in a skeleton is listed as follows.

Require: $\mathcal{V} = \{t_1, t_2, \cdots, t_n\}$ be the vertex set of all reserved terms in a collection of documents.

```
Ensure: $\mathcal{S}$ is the set of all maximal connected components.
   Let $\theta$ be a given minimal support.
   $\mathcal{S} \Leftarrow \emptyset$
   Let $S_0 = \{e_i | e_i = \{t_i\} \forall t_i \in V\}$ be the 0-simplex set.
   $i \Leftarrow 0$
   while $S_i \neq \emptyset$ do
      while for all vertex $t_j \in V$ do
         $S_{(i+1)} \Leftarrow \emptyset$ be the $(i+1)$-simplex set.
         while for all element $e \in S_i$ do
            if $e' = e \bigcup \{t_j\}$ with $t_j \notin e$ whose support is no less than $\theta$ then
               add $e'$ in $S_{(i+1)}$
               remove $e$ from $S_i$
            end if
         end while
      end while
      $\mathcal{S} \Leftarrow \mathcal{S} \bigcup S_i$
      $i \Leftarrow (i+1)$
   end while
```

Use our notation S_i is a skeleton of S_0^i. It is clear, one can get S_m^n for any n and m. A simplex will be constructed by including all those co-occurring terms whose support is bigger than or equal to a given minimal support θ. An external vertex will be added into a simplex if the produced support is no less than θ.

The documents can be decomposed into several categories based on the MAXIMAL PRIMITIVE CONCEPTS (correspond to a maximal simplex of highest dimension). If a document contains a MAXIMAL PRIMITIVE CONCEPT, it means that document highly equates to such concept, thereby, by the *apriori* property, all the sub-associations in the concept is also contained in this document. The document can be classified into the category identified with such a concept. A document often consists of more than one MAXIMAL PRIMITIVE CONCEPTS, in this case it can be classified into multi-categories. In the following sections, the algorithm is abbreviated to MPCC (Maximal Primitive Concepts Clustering).

6 Experimental Results

As for text search systems and document categorization systems, experimental results are conducted to evaluate the clustering algorithm, rather than analytic statements.

6.1 Data Sets

Three kinds of datasets are experiments are taken in our study. The first dataset is Web pages collected from Boley et al.[6]. 98 Web pages in four broad categories: business and finance, electronic communication and networking, labor and manufacturing are selected for the experiments. Each category is also divided into four subcategories.

The second dataset is 848 electronic medical literature abstracts collected from *PubMed*. All those abstracts are collected by searching from the keywords of *cancer*, *metastasis*, *gene* and *colon*. Our purpose is to discriminate all articles in according to which organs a cancer spreads from the primary tumor. In our study, we neglect the primary tumor is occurred in colon or from the other organs. A few organs are selected for this study, such as, liver, breast, lung, brain, prostate, stomach, pancreas, and lymph.

The third dataset is 305 electronic medical literatures collected from the journals, *Transfusion*, *Transfusion Medicine*, *Transfusion Science*, *Journal of Pediatrics* and *Archives of Diseases in Childhood Fetal and Neonatal Edition*. Those articles are selected by searching from keywords, *transfusion*, *newborn*, *fetal* and *pediatrics*. The MeSH categories have the use of evaluating the effectiveness of our algorithm.

The second and the third datasets are a homogeneous topic. They both denote a similar concept hierarchy. It is best for us to make validities on the concepts generated from our method by human experts.

6.2 Evaluation Criteria

The experimental evaluation of document clustering approaches usually measures their *effectiveness* rather than their *efficiency* [40], in the other word, the ability of an approach to make a *right* categorization.

Table 1
The contingency table for category c_i.

Category		Clustering Results	
c_i		**YES**	**NO**
Expert	**YES**	TP_i	FN_i
Judgment	**NO**	FP_i	TN_i

Considering the contingency table for a category (Table 1), *recall*, *precision*, and F_β are three measures of the effectiveness of a clustering method. Precision and recall with respect to a category is defined as follows respectively.

$$\mathrm{Precision_i} = \frac{\mathrm{TP_i}}{\mathrm{TP_i} + \mathrm{FP_i}}$$

$$\mathrm{Recall_i} = \frac{\mathrm{TP_i}}{\mathrm{TP_i} + \mathrm{FN_i}}$$

The F_β measure combined with precision and recall has introduced by van Rijsbergen in 1979 as the following formula:

$$F_\beta = \frac{(\beta^2 + 1) \times \mathrm{Precision_i} \times \mathrm{Recall_i}}{\beta^2 \times \mathrm{Precision_i} + \mathrm{Recall_i}}$$

In this paper, we use F_1 measure obtained when β equals 1 that means precision and recall are equal weight to evaluate the performance of clustering. Because many categories that will be generated and because of the comparison reasons, the overall precision and recall are calculated as the average of all precisions and recalls belonging to ever categories, respectively. F_1 is calculated as the mean of individual results. It is a macroaverage among categories.

6.3 Results

Table 2
The first dataset is compared with four algorithms, MPCC, PDDP, k-means and AutoClass.

Method	MPCC	PDDP	k_means	AutoClass	HCA
Precision	68.3%	65.6%	56.7%	34.2%	35%
Recall	74.2%	68.4%	34.9%	23.6%	22.5%
F_1 measure	0.727	0.67	0.432	0.279	0.274

Table 2 demonstrates the results of the first experiment. The result of the algorithm, PDDP [6], is under consideration by all non-stop words, that is, the F1 database in their paper, with 16 clusters. The result of our algorithm, MPCC, is under consideration by all non-stop words with the minimal support, 0.15.

The PDDP algorithm hierarchically splits the data into two subsets, and derives a linear discriminant function from them based on the principal direction (i.e., principal component analysis). With sparse and high dimensional datasets, principal component analyses often hurt the results of classification, which induces a high false positive rate and false negative rate. The hyperedges generated by PDDP is based on the average of the confidences of the itemsets with the same items. It is unfair that a possible concept would be withdrawn if a very small confidence of an itemset is existed from an implication direction.

As seen in Figure 6, 47 clusters, i.e. MAXIMAL PRITITIVE CONCEPTS (maximal connected components of top skeleton), has been generated by MPCC. It is larger than the original 16 clusters. After performing on decreasing the minimal support value to be 0.1, the number of clusters reduces to be 23 and its precision, recall, and F_1, become 63.7%, 77.3%, 0.698 respectively. The higher the minimal support value is, the lower the number of co-occurred terms in a complex. Figure 7 demonstrates the performance on the first dataset of MPCC.

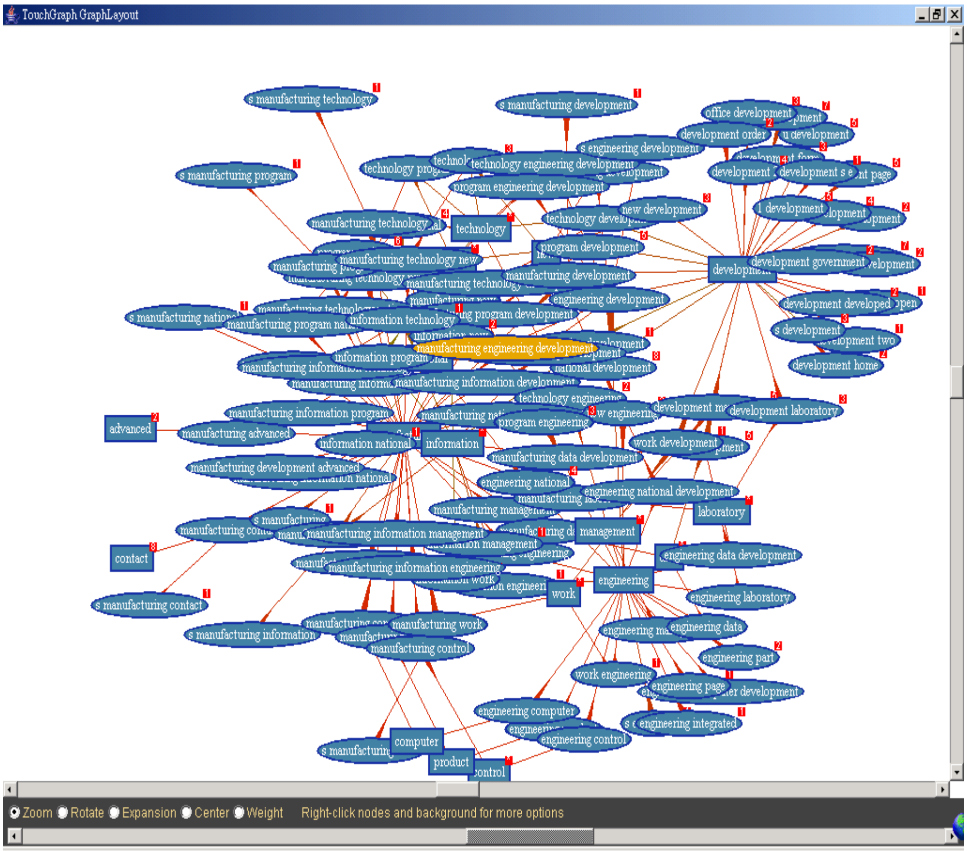


Fig. 6. The complex generated from the first dataset by using MPCC.

The effectiveness of the second dataset is shown in Figure 8. The use of fourteen organ related words are selected for clustering those abstracts. Figure 9 demonstrates the generated hypergraph associated with a minimal support, 0.05.

The MeSH categories (22 categories) have been taken to evaluate the effectiveness of MPCC on each individual category of the third dataset. Document clustering is based on the MeSH terms related to “Transfusion” and “Pediatrics”. The effectiveness of all categories is shown in Figure 10. The MeSH

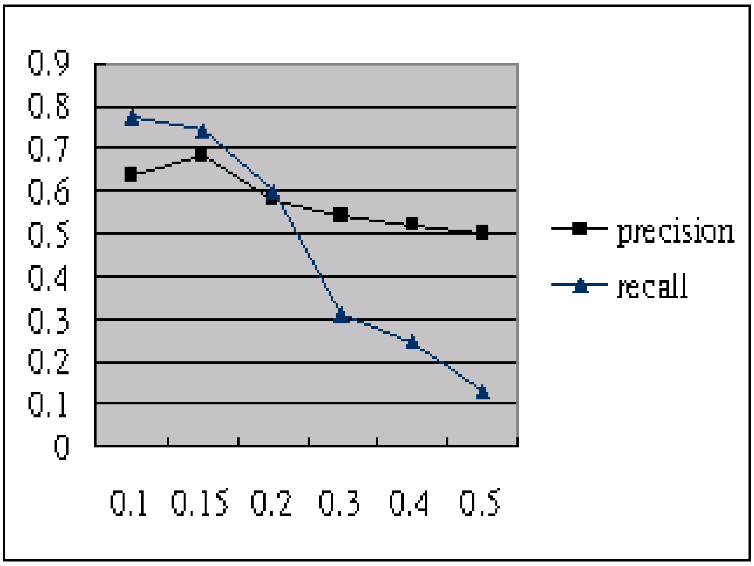


Fig. 7. The effectiveness of MPCC on the first dataset.

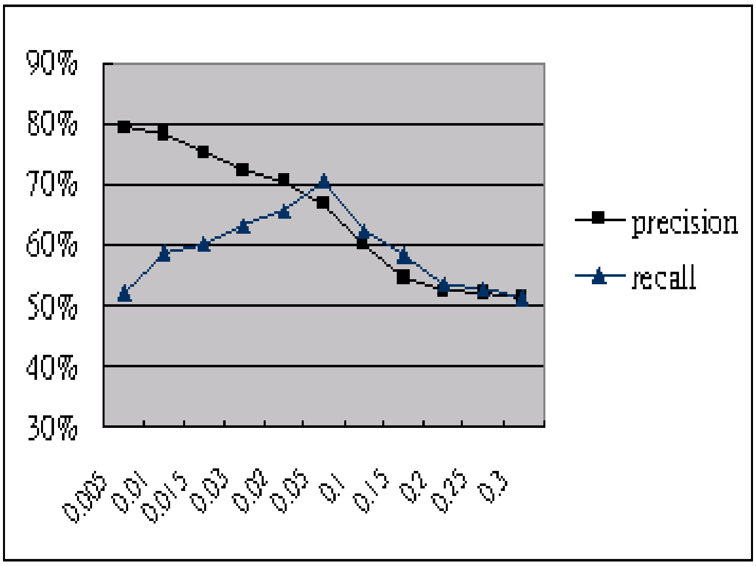
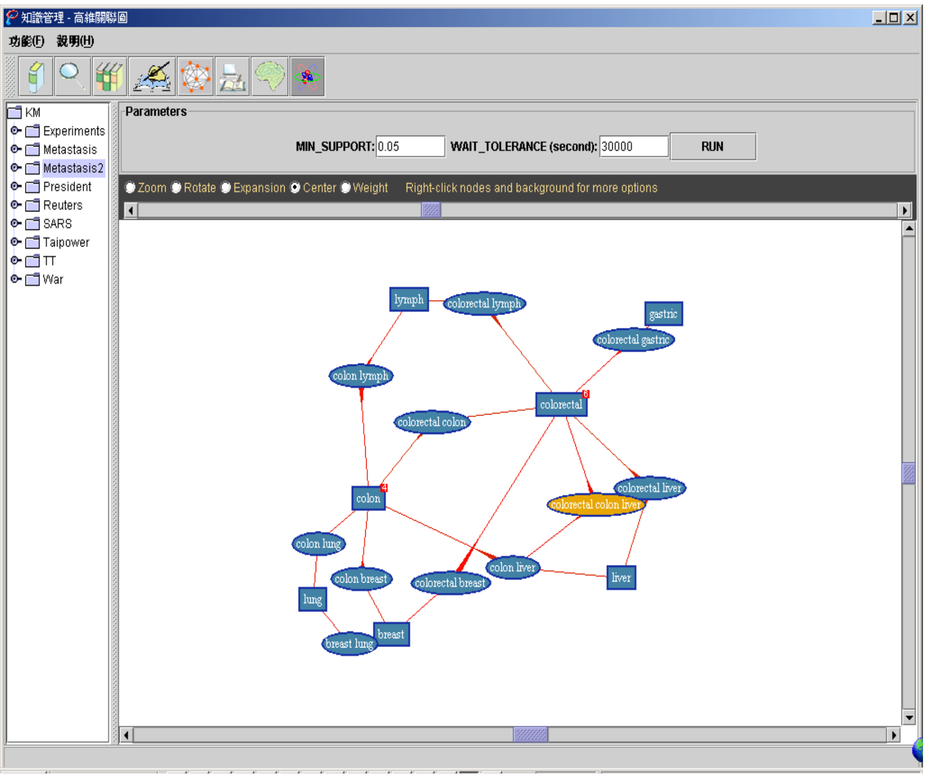


Fig. 8. The effectiveness of MPCC on the second dataset.

categories are a hierarchical structure that some categories are the subcategories of the other categories. Many concept categories are shared with the same terminologies that induces a high false negative rate by MPCC on document clustering. In this dataset documents are not uniform distributed in all categories, some categories only contain a few documents that makes their latent concepts restricted by a few terms, for example, the *Anemia* and the *Surgery* categories whose precision are both below 70%.


知識管理 - 高維關聯圖

功能(F) 說明(H)

KM
- Experiments
- Metastasis
- Metastasis2
- President
- Reuters
- SARS
- Taipower
- TT
- War

Parameters

MIN_SUPPORT: 0.05 WAIT_TOLERANCE (second): 30000 RUN

○ Zoom ○ Rotate ○ Expansion ● Center ○ Weight Right-click nodes and background for more options

Fig. 9. The complex generated from the second dataset with minimal support, 0.05.

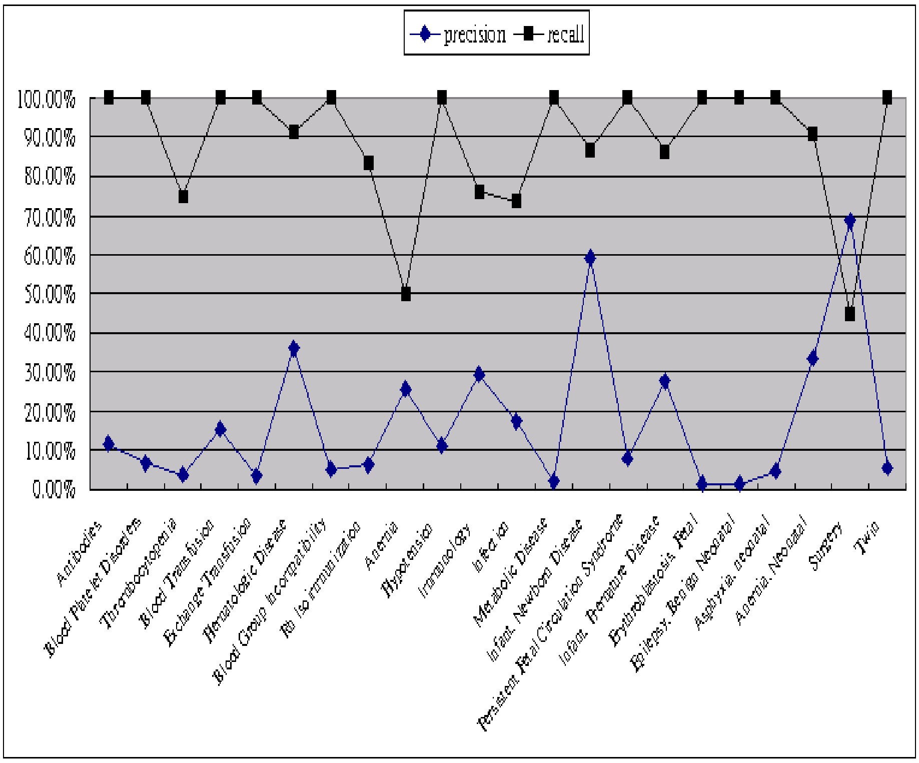


Fig. 10. The effectiveness of MPCC of the third experiment with minimal support, 0.02.

7 Conclusion

Polysemy, *phrases* and *term dependency* are the limitations of search technology [22]. A single term is not able to identify a latent concept in a document, for instance, the term “Network” associated with the term “Computer”, “Traffic”, or “Neural” denotes different concepts. To discriminate term associations no doubt is concrete way to distinguish one category from the others. A group of solid term associations can clearly identify a concept. Most methods, such as *k-means*, *HCA*, *AutoClass* or *PDDP* classify or cluster documents from the represented matrix of a set of documents. It seems inefficient and complicated to discover all term associations from such a high-dimensional and sparse matrix. The term-associations (frequently co-occurring terms) of a given collection of documents, form a simplicial complex. The complex can be decomposed into connected components at various levels (in various level of skeletons). We believe each such a connected component properly identify a concept in a collection of documents.

The paper presents a novel view based on finding maximal connected components for document clustering. An agglomerative method for finding geometric maximal connected components without the use of distance function is proposed. An maximal r-simplexes of highest dimensions can represent a MAXIMAL PRIMITVE CONCEPT in a collection of documents. We can effectively discover such a maximal simplexes of highest dimension and use them to cluster the collection of documents. Comparing with some traditional methods, such as *k-means*, *AutoClass* and *Hierarchical Clustering* (HAC), and the partition-based hypergraph algorithm, *PDDP*, our algorithm demonstrates its superior performance on three datasets. The paper illustrates that geometric complexes are effective models for automatic document clustering.

References

[1] R. Agrawal, T. Imielinski, and A. Swami. Mining association rules between sets of items in large databases. In *Proceedings of the 1993 International Conference on Management of Data (SIGMOD 93)*, pages 207–216, May 1993.

[2] R. Agrawal and R. Srikant. Fast algorithms for mining association rules. In *Proceedings of the 20th VLDB Conference*, 1994.

[3] T. W. Anderson. On estimation of parameters in latent structure analysis. *Psychometrika*, 19:1–10, 1954.

[4] F. Beil, M. Ester, and X. Xu. Frequent term-based text clustering. In *Proceedings of 8th Int. Conf. on Knowledge Discovery and Data Mining (KDD 2002)*, Edmonton, Alberta, Canada, 2002.

[5] M. W. Berry. Large scale sparse singular value computations. *International Journal of Supercomputer Applications*, 6(1):13–49, 1992.

[6] D. Boley, M. Gini, R. Gross, E.-H. Han, K. Hastings, G. Karypis, V. Kumar, B. Mobasher, and J. Moore. Document categorization and query generation on the world wide web using webace. *Artificial Intelligence Review*, 13(5-6):365–391, 1999.

[7] S. Brin, R. Motwani, J. Ullman, and S. Tsur. Dynamic itemset counting and implication rules for market basket data. In *Proceedings of ACM SIGMOD International Conference on Management of Data*, pages 255–264, 1997.

[8] S. Brin and L. Page. The anatomy of a large-scale hypertextual web search engine. In *Proceedings of the Seventh International WWW Conference (WWW 98)*, Brisbane, Australia, 1998.

[9] P. Cheeseman and J. Stutz. Baysian classification (autoclass): Theory and results. In U.M. Fayyad, G. Piatetsky-Shapiro, P. Smith, and R. Uthurusamy, editors, *Advances in Knowledge Discovery and Data Mining*, pages 153–180. AAAI/MIT Press, 1996.

[10] M. S. Chen, J. Han, and P. S. Yu. Data mining: An overview from a database perspective. *IEEE Transaction on Knowledge and Data Engineering*, 8(6):866–883, 1996.

[11] D. R. Cutting, D. R. Karger, J. O. Pedersen, and J. W. Tukey. Scatter/gather: A cluster-based approach to browsing large document collections. In *Proceedings of the Fifteenth Annual International ACM SIGIR Conference*, pages 318–329, 1992.

[12] R. Feldman, Y. Aumann, A. Amir, W. Klósgen, and A. Zilberstien. Text mining at the term level. In *Proceedings of 3rd International Conference on Knowledge Discovery, KDD-97*, pages 167–172, Newport Beach, CA, 1998.

[13] R. Feldman, I. Dagan, and W. Klósgen. Efficient algorithms for mining and manipulating associations in texts. In *Cybernetics and Systems, The 13th European Meeting on Cybernetics and Research*, volume II, Vienna, Austria, April 1996.

[14] R. Feldman, M. Fresko, H. Hirsh, Y. Aumann, O. Liphstat, Y. Schler, and M. Rajman. Knowledge management: A text mining approach. In *Proceedings of 2nd International Conference on Practical Aspects of Knowledge Management*, pages 29–30, Basel, Switzerland, 1998.

[15] R. Feldman and H. Hirsh. Mining associations in text in the presence of background knowledge. In *Proceedings of 3rd International Conference on Knowledge Discovery*, 1996.

[16] W. B. Frakes and R. Baeza-Yates. *Information Retrieval Data Structures and Algorithms.* Prentice Hall, Englewood Cliffs, NJ, 1992.

[17] N. Fuhr and C. Buckley. A probabilistic learning approach for document indexing. 9(3):223–248, 1991.

[18] J. D. Holt and S. M. Chung. Efficient mining of association rules in text databases. In *Proceedings of CIKM*, Kansas City, MO, 1999.

[19] A. Hyvarinen, J. Karhunen, and E. Oja. *Independent Component Analysis.* John Wiley & Son, 2001.

[20] A. K. Jain and R. C. Dubes. *Algorithms for Clustering Data.* Prentice Hall, 1988.

[21] I. T. Jolliffe. *Principle Component Analysis.* Spring-Verlag, New York, 1986.

[22] A. Joshi and Z. Jiang. Retriever: Improving web search engine results using clustering. In A. Gangopadhyay, editor, *Managing Business with Electronic Commerce: Issues and Trends*, chapter 4. World Scientific, 2001.

[23] G. Karypis, R. Aggarwal, V. Kumar, and S. Shekhar. Multilevel hypergraph partition application in vlsi domain. *Proceedings ACM/IEEE Design Automation Conference*, 8:381–389, 1997.

[24] T. Kohonen. *Self-Organization Maps.* Springer-Verlag, Berlin Heidelberg, 1995.

[25] R. Kosala and H. Blockeel. Web mining research: A survey. *SIGKDD Explorations*, 2(1):1–15, 2000.

[26] B. Lent, R. Agrawal, and R. Srikant. Discovering trends in text databases. In *Proceedings of 3rd International Conference on Knowledge Discovery, KDD-97*, pages 227–230, Newport Beach, CA, 1997.

[27] K. I. Lin and H. Chen. Automatic information discovery from the invisible web. In *Proceedings of the The International Conference on Information Technology: Coding and Computing (ITCC'02), Special Session on Web and Hypermedia Systems*, 2002.

[28] T. Y. Lin. Attribute (feature) completion - the theory of attributes from data mining prospect. In *Proceedings of 2002 IEEE International Conference on Data Mining (ICDM)*, pages 282–289, Maebashi, Japan, 2002.

[29] S.Y. Lu and K.S. Fu. A sentence-to-sentence clustering procedure for pattern analysis. *IEEE Transactions on Systems, Man and Cybernetics*, 8:381–389, 1978.

[30] J. MacQueen. Isome methods for classification and analysis of multivariate observations. In *Proceedings of the fifth Berkeley symposium on mathematical statistics and probability*, volume 1, pages 281–297, Berkeley, CA, 1967. University of California Press.

[31] M. Marchiori. The quest for correct information on the web: Hyper search engines. In *Proc. The Sixth International WWW Conference (WWW 97)*, Santa Clara, CA, 1997.

[32] M. E. Maron and J. K. Kuhns. On relevance, probabilistic indexing, and information retrieval. *Journal of ACM*, 7:216–244, 1960.

[33] D. Mladenic. Text-learning and related intelligent agents: A survey. *IEEE Intelligent Systems*, pages 44–54, 1999.

[34] A. Moffat and J. Zobel. Compression and fast indexing for multi-gigabit text databases. *Australian Computing Journal*, 26(1):19, 1994.

[35] J. S. Park, M. S. Chen, and P. S. Yu. Using a hash-based method with transaction trimming for mining association rules. *IEEE Transaction on Knowledge and Data Engineering*, 9(5):813–825, 1997.

[36] B. Pinkerton. Finding what people want: Experiences with the webcrawler. In *Proc. The Second International WWW Conference*, Chicag, IL, 1994.

[37] M. Rajman and R. Besanon. Text mining: Natural language techniques and text mining applications. In *Proceedings of seventh IFIP* 2.6 *Working Conference on Database Semantics (DS-7)*, Leysin, Switzerland, 1997.

[38] G. Salton and C. Buckley. Term weighting approaches in automatic text retrieval. *Information Processing and Management*, 24(5):513–523, 1960.

[39] G. Salton and M. J. McGill. *Introduction to Modern Information Retrieval.* McGraw-Hill, 1983.

[40] F. Sebastiani. Machine learning in automated text categorization. *ACM Computing Surveys*, pages 1–47, 2002.

[41] E. Spanier. *Algebric Topology.* McGraw-Hill Book Company, New York, NY, 1966.

[42] R. Weiss, B. Velez, M. A. Sheldon, C. Manprempre, P. Szilagyi, A. Duda, and D. K. Gifford. Hypursuit: A hierarchical network search engine that exploits content-link hypertext clustering. In *Proceedings of the 7th ACM Conference on Hypertext*, New York, NY, 1996.

[43] P. Willett. Extraction of knowledge from databases. *Information processing and management*, 24:577–597, 1988.

[44] O. Zamir and O. Etzioni. Web document clustering: a feasibility demonstration. In *Proceedings of 19th international ACM SIGIR conference on research and development in information retrieval (SIGIR 98)*, pages 46–54, 1998.