Automatic Clustering of Web News Based on Topics-Discovery

Hsi-Cheng Chang
Department of Information Management,
Hwa Hsia Institute of Technology
E-mail:hcchang@cc.hwh.edu.tw

Jeen-Fong Lin
Department of International Trade,
Taipei College of Maritime Technology
E-mail:jflin@mail.tcmt.edu.tw

Abstract

This paper proposes a new method for the unsupervised clustering of large and high-dimensional sets of textual data. The system begins with the topics-discovery process, which determines the *k* groups of document with maximal intra-group similarity and well scattered throughout the similarity space of the text collection. These *k* document groups are regarded as the *central topics* of the entire document collection. Then an intelligent feature selection algorithm is applied to deriving the features, called as *topic keywords*, that are the most suitable representation of the topics. Finally, all documents in the collection are clustered into *k* clusters according to the topic keywords. This method provides advantages of a very efficient clustering operation and involves no humanly predefined thresholds, which mean that no expert intervention is required. The experimental results indicate that this approach generated higher quality of cluster than many well-known document clustering algorithms.

Keywords: *Document clustering, feature selection, topic identification,, keyword clustering.*

1. Introduction

The explosive growth of the Internet has led to the establishment of a great deal of E-commerce sites, academic digital libraries and news web sites, etc.. The Internet has gradually become a major source of information and a medium of communication of people. It changes the way in which people live and work, meanwhile, the excessive information on the Internet also causes the serious problem of information overflow. Retrieving information effectively from the Internet is a great challenge.

These problems can be solved partially by clustering documents according to their topics and main contents. Therefore, some topic directory-based search engines were established. Clustering of data in a high-dimension space has extensive applications to many areas. However, most of traditional well-known clustering algorithms become computationally expensive and yield poor clustering results when the dataset clustered is large and the feature dimensions of the data elements are high. Furthermore, such clustering methods, including newly developed clustering methods, all apply some predefined thresholds to reduce the number of dimensions of features or eliminating the outliers of the dataset. Unfortunately, the thresholds differ from the datasets and are set according to the knowledge of experts or experiments.

This study develops a fully automatic unsupervised document clustering method for highly accurate document clustering and simple computing, without the need for any training data to be prepared or any humanly pre-specified threshold to be applied. The clustering process is divided into two stages to cluster the given collection of texts

accurately. The first stage applies a general distance measure and a Min-Max-Greedy clustering algorithm to generating *k* groups of document, which have maximal intra-group coherence and are well scattered over the entire similarity space of the text collection, called as *central-topics*. A central-topic is a small subset of documents that their similarities are greater than a value of maximum overall similarity of the text collection. In the second stage, an intelligent feature selection algorithm, called *feature projecting*, derives the most representative features of the central-topics. Then, all documents in the text collection are clustered into *k* text clusters according to the topic keywords of the *k* central-topics.

2. Related Work

Document clustering has been addressed for application in various areas of information retrieval and text mining. Document clustering was initially considered to improve the precision or recall rate of information retrieval systems [1,2,3,4]. More recently, document clustering technique has been developed for organizing retrieved results returned by a search engine in response to a user's query [5] or in browsing a collection of documents [6,7]. Document clustering has also been used to generate automatically hierarchical clusters of documents to increase the convenience of reading a large collection of documents [8]. A somewhat different approach finds natural clusters in an already existing document taxonomy [9], and then uses these clusters to generate an effective document classifier for new documents [10,11]. All of these document-clustering methods can be divided into two main groups - *supervised* and *unsupervised*. In supervised clustering, classifying knowledge will be obtained from domain experts or learned automatically using training documents. Acquiring knowledge from domain experts is time-consuming and knowledge may be incomplete, which will require complicated models and theories to be applied. In contrast, classifying knowledge automatically learned from training documents can be used efficiently, but its accuracy is constrained by the learning model and training data employed. Manually classifying documents is time-consuming and expensive, and so is unfeasible for handling the huge number of documents on the Internet. It also suffers from a bottleneck in the manual classification of newly collected documents. Unsupervised clustering does not depend on the preparation of training data. Clustering knowledge is obtained from the collection of documents. Some clustering techniques have been proposed for unsupervised document clustering, including the main ones - *Agglomerative clustering* and *partition clustering*. In the text document domain, the Scatter/Gather system [6,7], a document browsing system based on clustering, uses a hybrid method that involves both *K*-means and agglomerative hierarchical clustering algorithm.

These clustering algorithms begin either by estimating pairwise distances among documents or by measuring distortion between a document and a class centroid. Very often, the choice of the distance or distortion function is sensitive to the particular representing features, which may not accurately reflect the relevant structure of the high dimensional of documents. All of these methods share an important problem in that the high dimensionality of the feature space yields high computational complexity and need for space. This is because the native feature space consists of the unique terms in documents, which may number tens or hundreds of thousands of terms for even a moderately sized collection of texts.

Reducing the computational complexity

of keyword comparison in documents clustering and increasing the clustering accuracy are the main goals of this research. Keyword-clustering method generates new and size-reduced feature spaces by joining association keywords into groups, which greatly reduce the computational complexity of comparing keywords in clustering documents. Moreover, we endeavor to avoid applying any predefined thresholds in the clustering process, since the thresholds will differ from the dataset and may be determined by domain expert or experiment. And it doesn't make the clustering algorithm automatic for any new document collection. All in all, this study focuses on automatic unsupervised document clustering to achieve the following objectives: (1) to develop a method that does not require domain-dependent background information, predefined document categories or a given list of topics; (2) to determine the thresholds by the dataset itself instead of the predefined thresholds in the clustering process; (3) to yield highly accurate clustering results but with low computational complexity, and (4) the method should estimate the number of clusters in the collection.

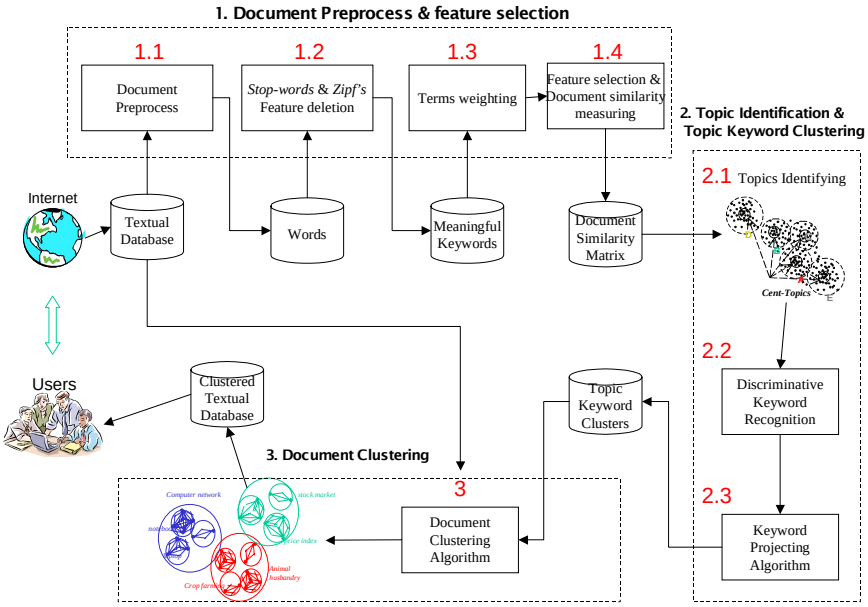

Fig. 1 The workflow and major components of the document clustering system

3. The Proposed Clustering Method

Given a collection of unlabeled documents, which are collected from some web sites, an attempt is made to identify clusters that are strongly related to the actual topics in the document collection. This task is especially difficult to complete in practice since no labeled examples of the topics are provided. The authors' earlier experimental studies [12,13] found that document clustering based on measures of document-document similarity not only has high computational complexity but also frequently yields poor performance, because not all of the words in the documents are discriminatory or characteristic. The main shortcoming of this clustering method and others as well are that they treat all the features in the feature set of the document collection equally, even though some of these features are discriminative. In several

document corpora discriminative words occur less frequently than non-discriminative words. When non-discriminative words dominate a document, clustering the document by the above methods may cause its misplacement. Accordingly, this study used a keyword-cluster-based method to generate the document clusters. It differs from traditional clustering methods, such as agglomerative clustering and the k-mean algorithm, which are document-based. Figure 1 shows the major components and workflow in the document clustering system of this paper. The implementation of the clustering algorithm is divided into three main phases: *document preprocessing*, *identifying central-topics & topic keyword-clusters* and *clustering documents using keyword clusters*. The following section details each phase.

3.1 Preprocessing and Vector Space Modeling of Documents

The system is designed to handle bi-lingual documents in both English and/or Chinese. In order to allow content-based clustering of documents we need to obtain a representation of their content. Hence, during the preprocessing in the system, a *Term Parser* unit is used to partition the paragraphs into sentences and extracts terms from sentences. Each extracted English term is not stemmed, while a sentence in Chinese, which is a character-based language, must be segmented into meaningful multi-character terms. Here the CKIP (Chinese Knowledge Information Processing) Chinese word segmentation program is used to process Chinese text and determine the part-of-speech of each term. An examination of various corpora of data also indicates that documents related to the same topic/event usually share several name entities and word-pairs, including the names of people, organizations, locations, and others, as well as many technical phrases. For example, ”wireless (無線)”, ”local (區域)”, ”network (網路)” are ordinary words in documents that discuss network application, and ”local network”, ” wireless network” as well as ”wireless local network” are the phrases that appear most often. The phrases are more meaningful and more suitable for using to represent the documents than ordinary words. Based on these observations, each document is represented using a rich set of features that includes salient phrases and all of the unique words. Hence, a phrase identification program, a statistical technique and a DHP algorithm [14] are adopted and used to identify meaningful phrases in the document collection. Then, some feature selection metrics are applied to eliminating meaningless terms that are used as function words in a sentence and cannot be used to express the subject of the documents. These are useless for document clustering. The feature selection metrics used to discard the meaningless terms are as follows.

1. All function terms, which are those in a sentence used as modifiers, like adjectives, prepositions and pronouns, are deleted based on part-of-speech information.
2. A *Zipf's* law-based [15] eliminator of terms is applied to removing terms that appear fewer than three times. Terms that appear in over two thirds of the documents in the collection are also eliminated, because they are too common to be useful for clustering.

These feature-selecting rules can substantially reduce the features in the documents, which will be shown in the experimental results.

After the text is preprocessed and the simple feature selection method is applied, the term parser counts the term frequency (*tf*) and calculates the term weight of each term using the $tf_i \times idf_i$ formula. Then, all of the documents are represented using a vector-space model [3,4]. In this model, all

terms are represented as (word, weight) pairs. In its simplest form, each document is represented as $d=\{w_1, w_2, \ldots, w_n\}$. The weights w_i of the terms are estimated as $tf_i \times idf_i$. Here *term-frequency* tf_i is the frequency of the i th term in the document, and idf_i is the *inverse document frequency* in the collection of documents. The motivation that underlies this weighting is that terms that appear often in many documents have limited discriminatory power, and so should be de-emphasized. It is designed to increase the discriminating capacity of high-frequency terms that occur in only a few documents, and can be calculated as $idf_i = \log_2(N/n_i)$. Where N is the total number of documents in the collection, and n_i is the number of documents that contain the term i. Therefore, words that arise in more documents are assigned lower weights. Finally, the weight of each document vector is normalized to unit length, such that $\left|\sum_{i=1}^{n}(tf_i \times idf_i)^2\right|^{1/2}$ to account for the fact that documents have different lengths. The rest of this study assumes that the vector representation d of each document has been weighted using $tf \times idf$ and normalized to be of unit length, according to the equation (1).

$$w_i = \frac{tf_i \times idf_i}{\sqrt{\sum_{i=1}^{n}(tf_i \times idf)^2}} \qquad (1)$$

In the vector-space model, the similarity between two documents d_x and d_y is commonly measured using the cosine function [3], given by

$$S_{xy} = cos(d_x, d_y) = \frac{\sum_k (w_{xk} \times w_{yk})}{\sqrt{\sum_k w_{xk}^2 \times \sum_k w_{yk}^2}} \qquad (2)$$

Since the document vectors are normalized to unit length, the above formula is simplified to be $cos(d_x, d_y) = d_x \cdot d_y$. Calculating the similarity between any two documents d_x and d_y in the collection yields an n by n document-document similarity matrix.

3.2 Identifying Topics Using the Min-Max-Greedy Agglomerative Clustering Algorithm

When the document similarity matrix has been generated in the preprocessing stage, a Min-Max-Greedy Agglomerative Clustering algorithm, as depicted in Fig. 2, is used to generate a number of document groups that with maximal intra-group coherence and well scattered throughout the dataset as central-topics in the document collection. A Greedy Agglomerative Clustering Algorithm is a common clustering technique for grouping items by measured similarity. A standard greedy agglomerative clustering system is composed of a set of items as inputs and a mean of computing the distance between any pair of these items. These items are then grouped into clusters by combining each closest pair of clusters until the number of clusters has been reduced to the target number. Here, the standard implementation of the greedy agglomerative clustering algorithm is modified and the *maximum overall pairwise similarity* of the text collection is used as a threshold for terminating the grouping process. The main intention that underlies this modification is to prevent the process from becoming stuck at a local maximum and to guarantee that topic groups are thoroughly scattered throughout the dataset.

As the algorithm in Fig. 2 and Fig. 3 indicate, a pair of items with maximum similarity is first obtained from all items in the document collection as a group. Then the item with the minimum average distance to all elements in the group is chosen and merged into the group. The grouping procedure is repeated until the mean similarity of the group is less than the *maximum overall pairwise similarity* of the document collection.

Algorithm Min-Max-Greedy agglomerative clustering
{*Set of data points*: *S*}
{*Topic clusters*: *C*}
{*Number of medoids: k* }
{d(.,.) is the *cosine* distance function }
{The maximum overall similarity of a data collection: S_{max} }
{The minimum number can be formed a cluster: *N* }
begin
M = {m_i, m_j} //{ m_i,m_j is a point-pair of *S* that with minimum distance}
AS = average distance between each data point in medoid *M*
while(*AS* > S_{max}) **{**
for each $x \in \{S\text{-}M\}$
dist(*x*) = d(*x*, *M*)
dist(x_i) = **min**{dist(x_i) | $x_i \in \{S\text{-}M\}$ }
$M = M \cup \{x_i\}$ // {data point *i* with the minimum dist(x_i,*M*) }
AS = average distance between each point in medoid *M*
compute the virtual- center of medoid *M*
}
if((sizeof *M*) > *N*.)
$C = C \cup M$
do {
{choose a point-pair as a new medoid, which will be as far away as possible from the previous medoids }
let m_i ,$m_j \in \{S\text{-}C\text{-}M\}$ be s.t. dist($m_i m_j$) = **max**{dist(*x*) | $x \in \{S\text{-}C\text{-}M\}$ }
$M = \{m_i, m_j\}$
AS = average distance between each data point in medoid *M*
while(*AS* > S_{max}) **{**
for each $x \in \{S\text{-}M\text{-}C\}$
dist(*x*) = d(*x*, *M*)
dist(x_i) = **min**{dist(x_i) | $x_i \in \{S\text{-}M\text{-}C\}$ }
$M = M \cup \{x_i\}$ // data point *i* with minimum dist(*M*, x_i)
AS = average distance between each data point in medoid *M*
compute the virtual-center of medoid *M*
}
if(sizeof *M* > *N*.)
$C = C \cup M$
} while (no new topic cluster formed)
end

Fig. 2 The Min-Max-Greedy Agglomerative clustering algorithm

The maximum overall pairwise similarity of a document collection is given by equation 3. In the absence of external information, including class labels, the cohesion of the data collection can be used as a measure of the distribution of a dataset.

$$S_{\max} = \frac{1}{N} \sum_{d_x, d_y \in D} \text{Max}(S_{d_x, d_y}) \qquad (3)$$

where *N* is the total number of documents and $Max(S_{d_x,d_y})$ is the most pairwise similarity of each document to others in the document collection. When a grouping process is terminated, based on the groups found previously, another pair of documents with maximum similarity to all of the remaining documents but with maximum distance to the groups found previously are selected and established as the elements of a new group. This grouping process will be repeated until no new group is generated.

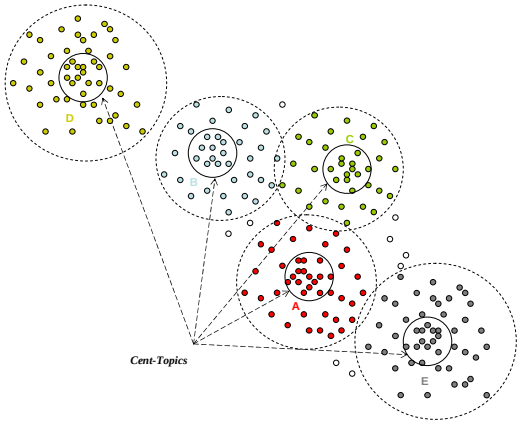

Fig. 3: Min-Max-Greedy Agglomerative Data Clustering

The numbers of documents in the finally identified groups may differ. Too many or too less of the documents in a group both will influence the identification of the representing keywords of the group. So the document groups must be refined. The average number of documents in the found groups is calculated as $\alpha = \sum_{i=1}^{n} |G_i| / n$, where n is the number of the groups and $|G_i|$ is the number of documents in group i. If a group with the number has more than α documents, then some documents that are less similar to the other documents in the group are removed. On the other hand, our previous researches and experiments [12,13] reveal that if the number of documents is under ten, then the group will not include a rich body of keywords that can be used objectively to identify the topic of the group. Such a group is poor and will be discarded. Finally, all of the groups thus derived will have maximal internal similarity and will be thoroughly scattered throughout the collection. These groups will be used as the *central-topics* in the collection. Discriminatory keywords are then obtained to represent the groups and topic keyword groups are used to group all of the documents in the collection.

3.3 Finding Discriminative Keywords for Topics Using a Keyword Projecting Algorithm

However, in almost all general document clustering methods, the computational complexity of the clustering algorithm increases exponentially with the size of the feature space. In this study, a keyword-based algorithm is used to cluster documents automatically by removing the words that do not discriminate among topics to reduce its computational complexity. In the document-preprocessing step, it eliminates many function words from the documents, but still many undiscriminating words remain. They not only increase the computational complexity of the method but also reduce the precision of clustering. So in this step, a keyword-projecting algorithm is applied to identifying effectively the discriminative keywords of each *central topic*.

After the central-topic groups of documents have been identified, the weights of the features in each group are recalculated. Some discriminative feature metrics, which compare the frequencies of occurrence of words inside a group to those outside the group, are introduced to determine whether a word is discriminative in a group or not. Three local properties are considered to obtain the relative weights of features among groups and are used to identify discriminative words:

1. *Term frequency* (*tf*): the word with high term frequency in each document in a group is more possible to be a discriminative word;
2. *Document frequency* (*df*): the word with high document frequency in a group tends to be a discriminative word;
3. *Inverse cluster frequency* (*icf*): the word with high frequency of occurrence in group *i* but a low frequency occurrence outside the group is highly discriminative for group *i*.

Based on these properties, the weights of the keywords in each group are determined by the equation (4) as follow:

$$w_c(t_k) = tf \times df \times icf = \sum_{N_d} (tf_k \times \log \frac{n_{df}(t_k)}{N_{df}(t_k)} \times \log \frac{N_{cf}(t_k)}{n_{cf}(t_k)}) \quad (4)$$

where $w_c(t_k)$ is the weight of word t_k in group C, n_{df} is the document frequency of word t_k in group C, N_{df} is the total number of document in group C, n_{cf} is the frequency of word t_k that presents in group C, and N_{cf} is the total number of groups of the central topics. The numbers of documents in each central-topic group are different, so the weights of the keywords are normalized according to the equation $W_c(t_k) = w_c(t_k) \Big/ \sum_{t_i} w_c(t_i)$.

All of the keywords in each group are weighted using the above equation and sorted in a descending order of weight. After the weights of the keywords in the central topics are determined, the keywords that can be used to represent the topic of the groups must be found. In practice, for a given dataset, the performance of the group will degrade rather than improve if the number of selected features above some threshold is applied. That is, in most cases, the additional information that is lost by discarding some features is compensated by more accurate mapping in a low dimensional space. As Fig. 4 shows, the best performance may occur at low feature dimension. Several studies [8, 13] have demonstrated this fact.

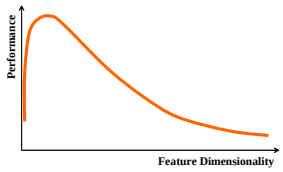

Fig.4 Relation of features and precision of cluster result

The authors' earlier work [12,13] also verified this claim. In the earlier experiments, we selected 5-40 keywords from each document in a document collection (ten topics, each including 50 documents) and applied the clustering algorithm based on keyword clusters to cluster the documents in the collection. Table 1 and figure 5 show the experimental results. It shows that using 10-25 features to represent a document yields the best clustering results.

Table1: The relation of features selected and clustering precision rate

Topics of testing corpus	Reduce weight (減肥),Traffic accident (車禍), Typhoon (颱風), Cellular phone (手機), Movies (電影), Pop music (流行音樂), Broadcast (廣播), Broadband (寬頻), Liquid crystal (液晶), Environment protection (環保)							
Number of terms selected from each document	5	10	15	20	25	30	35	40
Total different terms	125	293	560	911	1298	1738	2181	2632
Precision rate	0.948	0.972	0.930	0.902	0.948	0.80	0.702	0.461

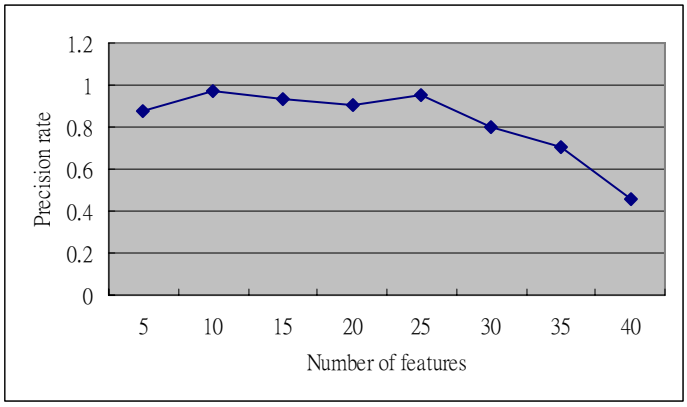

Fig. 5: The clustering accuracy of employing different number of features

In this paper, a keyword-projecting algorithm, presented in figure 5, is applied to identifying the discriminative features of each central-topic group. The algorithm is executed by three phases - an *initialization* phase, a *re-cluster* phase, and an *evaluation* phase. Some notations used to describe the algorithm are defined. Let $C = \{c_1,\ldots,c_k\}$ be the k central-topic groups, and $F = \{F_1,\ldots, F_k\}$ be the features in each central-topic group. Let $Mc=\{m_1,\ldots,m_k\}$ be the set of medoids used for clustering in a clustering iteration, and $m_i = \{f_1,\ldots, f_m\}$ be the set of features in a medoids. The distance between two points is given by $dist(x_i, x_j) = cos(x_i \cdot x_j)$. The objective of the algorithm is to determine the best set of keywords for representing the topics of the central-topic groups.

Algorithm keyword_projecting
{ K is the number of clusters}
{ C is the clusters}
{ D is all of the texts in the clusters}
{ F_i is the set of features associated with cluster C_i }
{ Mc is the set of medoids used for clustering in current iteration}
{ M_b is the best set of medoids found so far}
{ N is the final set of medoids returned along with associated dimensions}
{ A, B are constant integers}

begin
{1. Initialization Phase}
$F = \{F_1, , \ldots, F_k\}$ the feature sets of each cluster and sorted by its weight on the decrease
repeat
Mc = from $\{F_1,\ldots, F_k\}$ respectively pick a feature in order and form the k medoids $\{M_1,\ldots, M_k\}$
{ 2. Re-cluster Phase}
for each $x \in D$
dist(x_i) = d(x, M_i) //{$M_i \in Mc$ }
dist(x_i) = **min**{dist(x_i) | $M_i \in Mc$ }
$C_i = C_i \cup \{x\}$
{3. Evaluation Phase}
Re-evaluate the precision of clustering result of $\{C_1, , \ldots, C_k\}$
If(C_i with higher precision than previous clustering result)
Assign $(M_b)_i = M_i$
until (M_b is found) // i.e. the clustering result with maximum precision rate
end

Fig. 5 The keyword projecting Algorithm

The algorithm uses a greedy technique to find the best keyword sets of the central-topic groups. Initially, the keywords of each central-topic group have already been weighted and sorted in the decreasing order of weight, and the algorithm picks features from $F = \{F_1,\ldots, F_k\}$ sequentially to generate the mediods Mc$=\{m_1,\ldots,m_k\}$ of the groups. Then, all of the documents in the original central topics are re-clustered according to these mediods. Each document is assigned to the mediod to which it is the most similar as measured by $cos(d \cdot m_i)$, as shown in equation (2). After re-clustering has been completed, the new clusters are compared with the original central-topic groups to evaluate the re-clustering results. The processes is iterated until the algorithm yields the feature sets of all central-topics that can be reformed or are closest to the original central-topic groups. Then, these feature sets $M_{best}=\{m_1,\ldots,m_k\}$are the most discriminative one to represent the central-topics.

3.4 Document Clustering Using Topic Centroids

Given the topic sets and their corresponding vector representations, the vector of the topic is defined as *T*. Analogous to the corresponding similarity measurement of documents, the similarity between a document *d* and a vector of topic *T* is determined by the $cos(d,T) = d \cdot T \big/ (|d|^{1/2} \times |T|^{1/2})$ *cosine* measure. After the representative keywords of the central-topics are all identified, all of the documents in the document collection are clustered using the keyword clusters, associated with the best set of medoids $M_{best}=\{m_1,\ldots,m_k\}$. Each document will be assigned to a central-topic with a maximum document-to-centroid similarity.

4. Evaluating the Automatic Clustering Algorithm

This section describes the testing phase of the clustering process. All of the documents used were collected from Internet web sites. The test corpora were clustered by using the proposed system and four well-known document clustering methods, and the results were compared. The experiment results showed that the proposed system outperforms the well-known document clustering methods.

4.1 Evaluation method

Two main measures can be used to evaluate the quality of a data clustering method [16]. One compares different sets of clusters by determining overall similarity from the similarity between pairs of documents in a cluster instead of referring to external knowledge. This type of measure is called as an *internal quality* measure. The other evaluates the clustering technique by comparing the generated groups to known clusters. This type of measure is called as an *external quality* measure. The performance and relative ranking of different clustering algorithms vary greatly according to the measure used. However, if one clustering algorithm outperforms others on many of these measures, then some confidence can be gained that this clustering algorithm may be better than others in the context of interest.

In this paper, two test corpora were prepared and the results of the proposed clustering method were compared to those obtained by using well-known document to document similarity-based clustering methods - *single-link*, *complete-link* as well as *average-link* agglomerative hierarchical clustering methods and the *K-mean* of partition clustering method. Clusters are compared to the original manually grouped clusters of documents to estimate the precision of clustering. The clustering precision rate is

$$precision\ rate = \frac{The\ number\ of\ documents\ found\ by\ a\ clustering\ method\ and\ belonging\ to\ the\ correct\ \ cluster}{The\ number\ of\ documents\ in\ the\ cluster}$$

4.2 Experimental Results

Due to the lack of the benchmark corpus for Chinese text clustering, in our experiments, we collected the testing data from some famous web sites, including "*Yahoo!*"(http://tw.yahoo.com), "*Yam*" (http://www.yam.com/), "*Chinatimes News*" (http://news.chinatimes.com/), and "*et news*" (http://www.ettoday.com/), and others in Taiwan. Some keywords, covering twenty topics listed in Table 2 were entered as queries and 150-200 documents in each topic were gathered. All of these documents were selected and divided into three test corpora, corpus 1 with ten topics and each containing 50 documents for a total of 500 gathered documents; corpus 2 with 20 topics, each with 100 documents for a total of 2000 documents; corpus 3 with 20 topics also, each with 130-170 documents for a total of 3126 documents, and in this corpus a number of documents which contents involve multi-topics are included. Table 2 presents the parameters. From this table, the number of words in the corpora is very large. The performance is very bad when only using stoplist and *Zipf's* law to remove the function words in the documents. The stoplist and *Zipf's* law are incompetent to filter out all of the meaningless words in the documents, so eliminating the undiscriminating words in the corpora using the part-of-speech of words is important.

Table 2: The parameters of the two test corpora

	Corpus1	**Corpus2**	**Corpus 3**
Topics	wireless communication(無線通訊) cellular phone(手機,行動電話) digital camera(數位相機) PDA(掌上電子產品) Fixed Network(固網) IC design(IC 設計製造) LCD display(LCD 顯示器) Printer(印表機) Memory(記憶體) Notebook(筆記型電腦)	**Topics of Corpus1 +** main-board(主機板,電路板) hard disk(硬碟) CD-ROM(光碟片) Scanner(掃描器) environment protection(環保) SARS (SARS 肺炎) credit card(信用卡) music(音樂) educational reforms(教改) finance(金融)	
Total number of documents	500	2000	3126
Total number of words	278327	1098351	1721966
Average length of document	557	549	551
Number of words (nouns, verbs) remained	53185	198741	243935
Total different words	6857	9846	11774
Percentage of word removed	80.89%	81.90%	85.83%

These test corpora are clustered using the proposed system and the four well-known document-document similarity-based clustering methods. The clustering results were then compared. Table 2 indicates that although the proposed feature selection method eliminates many undiscriminating words, 6857-11774 different terms remained in the test corpora. The numbers of words in each document are still very high. Using these terms directly to cluster documents is not only highly computationally complex but also yields unfavorable result. Hence, in our clustering

approach, the keyword-projecting algorithm is applied to extracting the most discriminative keywords to represent the central topics and to form the keyword clusters. Table 3 shows the experimental results generated by using the different clustering algorithms for the corpora.

Table3: The clustering precision rate of different technique

Clustering Method	Corpus1	Corpus2	Corpus3
Single-link	0.119	0.083	0.011
Complete-link	0.346	0.278	0.225
Average-link	0.352	0.304	0.236
K-Mean	0.588	0.504	0.431
Proposed method	**0.672**	**0.553**	**0.487**

The experimental results indicate that the proposed clustering approach outperforms the four well-known document-document similarity-based clustering methods. For corpus 3, the clustering accuracy of the four well-known document similarity-based clustering method declines quickly, especially the single-link clustering algorithm, since a number of documents whose contents involve multi-topics are included.

In the other experiment, we want to observe the influence between the number of topics and the clustering accuracy. Hence, the collected documents were grouped into four test corpora with 5, 10, 15 and 20 topics, each of with 120 documents. Clustering was then performed by using the five clustering algorithms. Fig. 6 shows the experimental results.

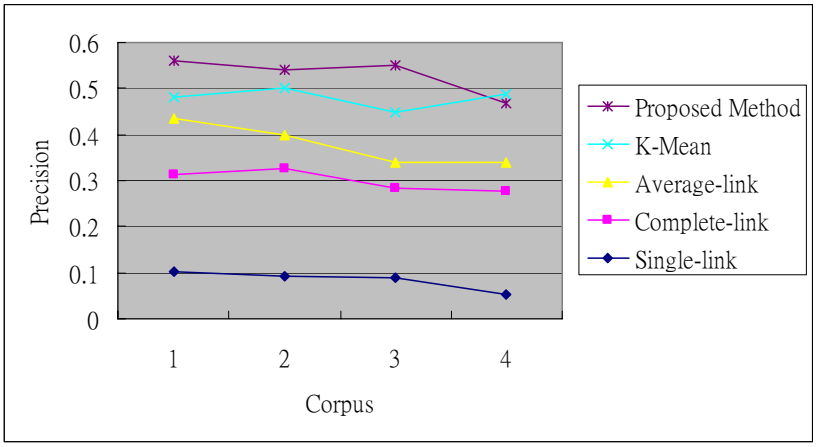

Fig. 6 Precision rate of five clustering methods in each corpus

From Fig. 6, the accuracy of the proposed clustering method keeps above 0.46, but the accuracy of the agglomerative hierarchical clustering algorithm declines with the number of documents and topics.

Summarizing the experimental results, the clustering accuracy of the single-link algorithm is always unsatisfactory in every case. The performances of agglomerative hierarchical clustering algorithm as well as partition techniques, as typified by K-Mean, are related with the contents of the data collection. When more documents are included, indicating that the contents involve multi-topics, the accuracy of clustering results will decline more quickly. The proposed method, which is based on topic identification, is not only efficient but also obtains high accuracy of clustering results.

5. Conclusion

Although numerous interesting document clustering methods have been extensively studied for many years, accurately clustering documents without

using domain-dependent background information, predefined document categories or a given list of topics remains difficult. Moreover, the high computational complexity and predefined thresholds in other clustering methods make them neither efficient nor automatic. Reducing the heavy computational load and increasing the precision of the unsupervised clustering of documents are important problems.

This study presented a novel method, based on the concept of central-topics identification and keyword-clusters recognition to solve these problems satisfactorily. The proposed clustering method provides three main advantages. Firstly, using the topic-identification and keyword-projecting algorithm, only the most discriminative and meaningful keywords are used, which greatly reduce the computational complexity; secondly, the system offers more accuracy of document clustering, and thirdly, no humanly predefined thresholds are involved so the system can automatically cluster newly collected documents.

Reference

[1] C. J. van Rijsbergen, (1989), "*Information Retrieval*", Buttersworth, London, second edition.

[2] Gerald Kowalski, "*Information Retrieval Systems – Theory and Implementation*", Kluwer Academic Publishers, 1997.

[3] G. Salton, "Automatic Text Processing: The Transaction, Analysis, and Retrieval of Information by Computer," *Addison-Wesley*, 1989.

[4] R. Baeza-Yates and B. Ribeiro-Neto, "Modern Information Retrieval," *Addison Wesley Longman Limited*, 1999.

[5] Oren Zamir, Oren Etzioni, Omid Madani, Richard M. Karp, "*Fast and Intuitive Clustering of Web Documents*", KDD '97, Pages 287-290, 1997.

[6] Douglass R. Cutting, David R. Karger, Jan O. Pedersen, and John W. Tukey, *Scatter/Gather: A Cluster-based Approach to Browsing Large Document Collections*, SIGIR '92, Pages 318 – 329, 1992.

[7] Hearst, M. A. and Pedersen, J. O. 1996. Reexamining the cluster hypothesis: Scatter/Gather on retrieval results. *ACM SIGIR-96*. pp. 76–84. Zurich, Switzerland.

[8] Daphe Koller and Mehran Sahami, *Hierarchically classifying documents using very few words*, Proceedings of the 14th International Conference on Machine Learning (ML), Nashville, Tennessee, July 1997, Pages 170-178.

[9] Charu C. Aggarwal, Stephen C. Gates and Philip S. Yu, *On the merits of building categorization systems by supervised clustering*, Proceedings of the fifth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Pages 352 – 356, 1999.

[10] Yu-Sheng Lai and Chung-Hsien Wu, "Meaningful Term Extraction and Discriminative Term Selection in Text Categorization via Unknown-Word Methodology", ACM Transactions on Asian Language Information Processing, Vol. 1, No. 1, March 2002, Pages 34-64.

[11]Shian-Hua Lin, Meng-Chang Chen, Jan-Ming Ho and Yueh-Ming Huang, "ACIRD: Intelligent Internet Document Organization and Retrieval", IEEE Transactions on Knowledge and Data Engineering, Vol. 14, No. 3, May/June 2002, Pages 599-613.

[12] Hsi-Cheng Chang, Chiun-Chieh Hsu, Chi-Kai Chan, "Automatic Document Clustering Based on Keyword Clusters Using Partitions of Weighted Digraphs",

accepted for publication in the International Journal of Computer System Science & Engineering, November 2003.

[13] Hsi-Cheng Chang, Chiun-Chieh Hsu, Yi-Wen Deng, “Automatic Document Clustering Based on Keyword Clusters Using Partitions of Weighted Undirected Graph”, Proceeding of 2003 Symposium on Digital Life and Internet Technologies, September 2003.

[14] J.S. Park, M. S. Chen, and P.S. Yu, “Using a Hashing-Based Method with Transaction Trimming for Mining Association Rules,” IEEE Transactions on Knowledge and Data Engineering, vol. 9, no. 5, September/October 1997.

[15] H. P. Zipf, “Human behavior and the principle of least effort”, Addison-Wesley, Cambridge, Massachusetts, 1994.

[16] Michael Steinbach, George Karypis and Vipin Kumar, “A Comparison of Document Clustering Techniques”, http://citeseer.nj.nec.com/steinbach00comparison.html