International Journal of Applied Information Systems (IJAIS) – ISSN : 2249-0868
Foundation of Computer Science FCS, New York, USA
Volume 4– No.5,October 2012 – www.ijais.org

# Document Clustering: A Detailed Review

Neepa Shah
Assistant Professor, IT Department
D.J. Sanghvi College of Engineering,
Vile Parle(West), Mumbai-56

Sunita Mahajan, PhD.
Principal, Institute of Computer Science,
M.E.T., Bandra (west),
Mumbai – 50.

## ABSTRACT
Document clustering is automatic organization of documents into clusters so that documents within a cluster have high similarity in comparison to documents in other clusters. It has been studied intensively because of its wide applicability in various areas such as web mining, search engines, and information retrieval. It is measuring similarity between documents and grouping similar documents together. It provides efficient representation and visualization of the documents; thus helps in easy navigation also. In this paper, we have given overview of various document clustering methods studied and researched since last few years, starting from basic traditional methods to fuzzy based, genetic, co-clustering, heuristic oriented etc. Also, the document clustering procedure with feature selection process, applications, challenges in document clustering, similarity measures and evaluation of document clustering algorithm is explained.

## General Terms
Data Mining, Document Clustering

## Keywords
Document clustering, document clustering applications, document clustering procedure, similarity measures for document clustering, evaluation of document clustering algorithm, challenges in document clustering

## 1. INTRODUCTION
The steady and amazing progress of computer hardware technology in the last few years has led to large supplies of powerful and affordable computers, data collection equipments, and storage media. Due to this progress there is a great encouragement and motivation to the database and information industry to make a huge number of databases and information repositories; which is available for transaction management, information retrieval, and data analysis. Thus, technology advancement has provided a tremendous growth in the volume of the text documents available on the internet, digital libraries and repositories, news sources, company-wide intranets, and digitized personal information such as blog articles and emails. With the increase in the number of electronic documents, it is hard to organize, analyze and present these documents efficiently by putting manual effort [1]. These have brought challenges for the effective and efficient organization of text documents automatically [2].

Data mining is the process of extracting the implicit, previously unknown and potentially useful information from data. Document clustering, subset of data clustering, is the technique of data mining which includes concepts from the fields of information retrieval, natural language processing, and machine learning. Document clustering organizes documents into different groups called as clusters, where the documents in each cluster share some common properties according to defined similarity measure. The fast and high-quality document clustering algorithms play an important role in helping users to effectively navigate, summarize, and organize the information.

Clustering can produce either disjoint or overlapping partitions. In an overlapping partition, it is possible for a document to appear in multiple clusters [3] whereas in disjoint clustering, each document appears in exactly one cluster.

Document Clustering is different than document classification. In document classification, the classes (and their properties) are known a priori, and documents are assigned to these classes; whereas, in document clustering, the number, properties, or membership (composition) of classes is not known in advance. Thus, classification is an example of supervised machine learning and clustering that of unsupervised machine learning [3].

According to [4], document clustering is divided into two major subcategories, hard clustering and soft clustering. Soft clustering also known as overlapping clustering is again divided into partitioning, hierarchical, and frequent itemset-based clustering.

- Hard (Disjoint): Hard clustering compute the hard assignment of a document to a cluster i.e., each document is assigned to exactly one cluster; giving a set of disjoint clusters.
- Soft (Overlapping): Soft clustering compute the soft assignment i.e., each document is allowed to appear in multiple clusters; thus, generates a set of overlapping clusters. For instance, a document discussing "Natural language and Information Retrieval" will be assigned to "Natural language" and "Information Retrieval" clusters.
- Partitioning: Partitioning clustering allocate documents into a fixed number of non-empty clusters. The most well-known partitioning methods are the K-means and its variants [4]. The basic K-means method initially allocates a set of objects to a number of clusters randomly. In each iteration, the mean of each cluster is calculated and each object is reassigned to the nearest mean. It stops when there is no change for any of the clusters between successive iterations.
- Hierarchical: Hierarchical document clustering is to build dendrogram, a hierarchical tree of clusters, whose leaf nodes represent the subset of a document collection. Hierarchical Agglomerative Clustering (HAC) and Unweighted Pair Group Method with Arithmetic Mean (UPGMA) fall in this category [4].

  Hierarchical methods are classified into agglomerative methods and divisive methods. In an agglomerative method, each object forms a cluster. Two most similar clusters are combined iteratively until some termination criterion is satisfied. Thus, it follows bottom up approach. Whereas, in a divisive method top-down approach is there; i.e., from a cluster consisting of all the objects, one cluster is selected and split into smaller clusters recursively until some termination criterion is

satisfied [5]. The major decision criteria, at each step, are to find which cluster to split and how to perform the split.

The bisecting K-means, a variant of K-means, is a divisive hierarchical clustering algorithm. The algorithm recursively selects the largest cluster and uses the basic K-means algorithm to divide it into two sub-clusters until the desired number of clusters is formed [5].

Out of agglomerative and divisive, agglomerative techniques are more common [6].

In [6], UPGMA is proved to be the best among three agglomerative clustering algorithms, IST (Intra-Cluster Similarity Technique), CST (Centroid Similarity Technique), and UPGMA through experiments.

Hierarchical clustering gives better quality clustering, but is limited because of its quadratic time complexity. Whereas, partitioning methods like K-means and its variants have a linear time complexity, making it more suitable for clustering large datasets, but are thought to produce inferior clusters [6]. Also, the major problem with K-means is that it is sensitive to the selection of the initial partition and may converge to local optima [7].

- Frequent itemset-based: These methods use frequent itemsets generated by the association rule mining to cluster the documents. Also, these methods reduce the dimensionality of term features efficiently for very large datasets, thus improves the accuracy and scalability of the clustering algorithms. Another advantage of frequent-itemset based clustering method is that each cluster can be labeled by the obtained frequent itemsets shared by the documents in the same cluster [4]. These methods include Hierarchical Frequent Term-based Clustering (HFTC) [8], Hierarchical Document Clustering Using Frequent Itemsets (FIHC) [9], and Fuzzy Frequent Itemset-based Document Clustering ($F^2IDC$) [11].

  HFTC method minimizes the overlap of clusters in terms of shared documents. But the experiments of Fung et al. showed that HFTC is not scalable. For a large datasets in [9] FIHC algorithm is given where frequent itemsets derived from the association rule mining are used to construct a hierarchical topic tree for clusters. FIHC uses only the global frequent items in document vectors, which drastically reduces the dimensionality of the document set. Thus, FIHC is not only scalable, but also accurate [10]. In $F^2IDC$ [11] fuzzy association rule mining is combined with WordNet. A term hierarchy generated from WordNet is applied to discover generalized frequent itemsets as candidate cluster labels for grouping documents. The generated clusters with conceptual labels are easier to understand than clusters annotated by isolated terms for identifying the content of individual clusters.

The paper is organized as: section 2 of the paper gives various applications of document clustering followed by document clustering procedure in section 3. Similarity measures for document clustering are explained in section 4. Section 5 highlights evaluation of document clustering algorithm. Various challenges faced clustering are highlighted in section 6. Section 7 gives detailed overview of various document clustering methods studied so far. We conclude the paper in section 8.

## 2. DOCUMENT CLUSTERING APPLICATIONS

As seen in section 1, document clustering is unsupervised learning and is applied in many fields of business and science. Initially, document clustering was studied for improving the precision or recall in information retrieval systems. Document clustering has also been used to automatically generate hierarchical clusters of documents [6]. Following are few applications of document clustering [12].

- Finding Similar Documents: To find similar documents matching with the search result document. Clustering is able to discover documents that are conceptually alike compared to search-based approaches which discover documents sharing many of the same words.
- Organizing Large Document Collections: To organize large number of uncategorized documents in taxonomy identical to the one human would create for easy retrieval.
- Duplicate Content Detection: In many applications there is a need to find duplicates in a large number of documents. Clustering is employed for plagiarism detection, grouping of related news stories and to reorder search results rankings.
- Recommendation System: Here, a user is recommended articles based on the articles the user has already read. Again this is possible by clustering of the articles, and improving the quality.
- Search Optimization: Clustering helps a lot in improving the quality and efficiency of search engines as the user query can be first compared to the clusters instead of comparing it directly to the documents. Clustering is used in organizing the results returned by a search engine in response to a user's query [6]. Following this principle of cluster-based browsing by automatically organizing search results into meaningful categories are Teoma, vivisimo clustering engine, MetaCrawler, WebCrawler [13].

## 3. DOCUMENT CLUSTERING PROCEDURE

It is important to emphasize that getting from a collection of documents to a clustering of the collection, is not merely a single operation. It involves multiple stages; which generally comprise three main phases: feature extraction and selection, document representation, and clustering.

Feature extraction begins with the parsing of each document to produce a set of features and exclude a list of pre-specified stop words which are irrelevant from semantic perspective. Then representative features are selected from the set of extracted features [13]. Feature selection is an essential pre-processing method to remove noisy features. It reduces the high dimensionality of the feature space and provides better data understanding, which in turn improves the clustering result, efficiency and performance. It is widely used in supervised learning, such as text classification [14]. Thus, it is important for improving clustering efficiency and effectiveness. Commonly employed feature selection metrics are term frequency (TF), inverse document frequency (TF · IDF), and their hybrids. These are discussed further in same section. Also some improvements in traditional methods is discussed.

In the document representation phase, each document is represented by k features with the highest selection metric scores according to top-k selection methods. Document representation methods include binary (presence or absence of a feature in a document), TF (i.e., within-document term frequency), and TF.IDF.

In the final phase of document clustering, the target documents are grouped into distinct clusters on the basis of the selected features and their respective values in each document by applying clustering algorithms [13].

## 3.1 Term Frequency–Inverse Document Frequency (TF.IDF)

In most clustering algorithms, the dataset to be clustered is represented as a set of vectors $X=\{x_1, x_2, \ldots, x_n\}$, where the vector $x_i$ is called the feature vector of single object. In Vector Space Model (VSM), the content of a document is formalized as a dot in the multidimensional space and represented by a vector $d$, such as $d=\{w_1,w_2,\ldots,w_n\}$, where $w_i$ is the term weight of the term $t_i$ in one document. The term weight value represents the significance of this term in a document. To calculate the term weight, the occurrence frequency of the term within a document and in the entire set of documents is considered. The most widely used weighting scheme combines the Term Frequency with Inverse Document Frequency (TF.IDF) [7]. The term frequency gives a measure of the importance of the term within the particular document. TF.IDF is a statistical measure which presents how important a word is to a document. More frequent words in a document are more important, i.e. more indicative of the topic [15].

Let $f_{ij}$ = frequency of term i in document j

Now normalize term frequency (tf) across the entire corpus:

$tf_{ij} = f_{ij} / max\{f_{ij}\}$

The inverse document frequency is a measure of the general importance of the term. Terms that appear in many different documents are less indicative of overall topic.

Let $df_i$ = document frequency of term i
= number of documents containing term i
$idf_i$ = inverse document frequency of term i,
= log2 (N/ $df_i$)
Where N: total number of documents

A typical combined term importance indicator is TF.IDF weighting:

$w_{ij} = tf_{ij} \times idf_i = tf_{ij} \times \log2 (N/ df_i)$

## 3.2 Improvements in traditional term weighting, feature selection, and dimension reduction methods

In [16], authors have investigated several widely used unsupervised and supervised term weighting methods on benchmark data collections in combination with Support Vector Machines (SVM) and k-Nearest Neighbor (kNN) algorithms. A new simple supervised term weighting method tf:rf, to improve the terms' discriminating power for text categorization task is proposed. This proposed supervised term weighting method has consistently better performance than other term weighting methods. Also the popularly used tf:idf method has not shown a uniformly good performance in terms of different data sets.

An entropy-based feature ranking method is proposed by Dash and Liu. In the Expectation-Maximization (EM) algorithm, the Minimum Message Length criterion is derived to select the feature subset and the number of clusters. They proposed a filter method that is independent of any clustering algorithm, where feature importance is measured by the contribution to an entropy index, based on data similarity [17].

The task of selecting relevant features is a hard problem in the field of unsupervised text clustering due to the absence of class labels. In [18] authors have proposed a new mixture model named multinomial mixture model with feature selection (M3FS). In M3FS, the concept of component-dependent feature saliency to the mixture model is introduced. A feature is relevant to a certain mixture component if the feature saliency value is higher than a predefined threshold. As the feature selection process is treated as a parameter estimation problem, EM algorithm is used for estimating the model. The experiment on commonly used text datasets has shown that the M3FS method has good clustering performance and feature selection capability.

In [14], various approaches of feature selection like multitype feature co-selection for clustering (MFCC), weighted semantic features and cluster similarity using non negative matrix factorization (NMF), local feature selection for partitional hierarchical text clustering, approach based on expectation maximization and cluster validity, based on Ant Colony Optimization (swarm intelligence algorithm) are discussed.

## 3.3 Dimension Reduction

Dimension reduction for large-scale text data is attracting much attention nowadays because high dimensionality causes serious problem for the efficiency of most of the algorithms [14]. These algorithms are of two types: feature extraction and feature selection. In the feature extraction, new features are combined from their original features through algebraic transformation. Though effective, these algorithms introduce high computational overhead, making it difficult for real-world text data.

In feature selection, subsets of features are selected directly. These algorithms are widely used in real-world tasks due to their efficiency, but are based on greedy strategies rather than optimal solutions. So, a unified optimization framework is proposed in [19], which is a combined approach by integrating benefit of both these methods. This novel feature selection algorithm is called Trace-Oriented Feature Analysis (TOFA). The proposed objective function of TOFA integrates many prominent feature extraction algorithms' objective functions, such as unsupervised Principal Component Analysis (PCA) and supervised Maximum Margin Criterion (MMC), It makes TOFA applicable for both supervised and unsupervised problems. Also, by tuning a weight value, TOFA is suitable to solve semi-supervised learning problems. Experimental results on several real-world data sets validate the effectiveness and efficiency of TOFA in text data for dimensionality reduction purpose.

# 4. SIMILARITY MEASURES FOR DOCUMENT CLUSTERING

Cluster analysis methods are based on measurements of the similarity between a pair of objects. The determination of similarity between a pair of objects involve three major steps:

i) the selection of the variables to be used to characterize the objects, ii) the selection of a weighting scheme for these variables, and iii) the selection of a similarity coefficient to determine the degree of resemblance between the two attribute vectors [20]. Accurate clustering requires a precise definition of the closeness between a pair of objects, in terms of either the pair-wise similarity or distance. A variety of similarity or distance measures have been proposed and widely applied, such as cosine similarity, Jaccard correlation coefficient, Euclidean distance, and relative entropy [2].

Few widely used similarity measures are given in [2]:

- Euclidean Distance: It is a standard metric for geometrical problems. It is the ordinary distance between two points and can be easily measured with a ruler. It is also the default distance measure used in K-means algorithm.
- Cosine Similarity: The similarity of two documents corresponds to the correlation between the vectors. This is quantified as the cosine of the angle between vectors, that is, cosine similarity.
- Jaccard Coefficient: The Jaccard coefficient compares the sum weight of shared terms to the sum weight of terms that are present in either of the two documents but are not the shared terms.
- Pearson Correlation Coefficient: It is another measure of the extent to which two vectors are related.
- Averaged Kullback-Leibler Divergence: The Kullback-Leibler divergence (KL divergence), also called the relative entropy, is a widely applied measure for evaluating the differences between two probability distributions.

As many types of similarity coefficient available are for determining the degree of similarity between a pair of objects, Sneath and Sokal describe four main classes of coefficient: distance coefficients, association coefficients, probabilistic coefficients, and correlation coefficients.

- Distance coefficients: For e.g., Euclidean distance, have been used very extensively in cluster analysis, owing to their simple geometric interpretation. However, a major limitation of the Euclidean distance in the information retrieval context is that it can lead to two documents being regarded as highly similar to each other; despite they share no terms at all in common. The Euclidean distance is thus not widely used for document clustering.
- Association coefficients: These have been very widely used for document clustering. It is the number of terms common to a pair of documents having *a* and *b* terms, respectively. Here normalization becomes essential to handle documents of different sizes. Two commonly used normalized association coefficients are the Dice coefficient and the Jaccard coefficient.
- Probabilistic coefficients: Here, the main criterion for the formation of a cluster is that the documents in it have a maximal probability of being jointly corelevant to a query.
- Correlation coefficients: There do not seem to have been any reports of the use of correlation coefficients for document clustering [20].

Usually, the cosine function is used to measure the similarity between two documents, but it may not work well when the clusters are not well separated. To solve this problem, in [21], authors have applied the concepts of neighbors and link. If two documents are similar, they are called as neighbors of each other. The link between two documents is the number of their common neighbors. The neighbors and link concept involve the global information into the measurement of the closeness of two documents. This concept along with the family of k-means algorithms is proposed as: i) a new method to select initial cluster centroids using the ranks of candidate documents; ii) a new similarity measure using combination of the cosine and link functions; and iii) a new heuristic function for selecting a cluster to split using the neighbors of the cluster centroids. The experimental results on real-life data sets demonstrated that this approach can significantly improve the performance of document clustering in terms of accuracy without increasing the execution time much.

# 5. EVALUATION OF DOCUMENT CLUSTERING ALGORITHM

One of the most important issues in clusters analysis is the evaluation of the clustering results. Evaluation is the analysis of the output to understand how well it reproduces the original structure of the data [12].

The ways of evaluation are divided in two parts:

- Internal quality measure: Here, the overall similarity measure is used based on the pair wise similarity of documents and no external knowledge is used. The cohesiveness of clusters can be used as a measure of cluster similarity. One method for computing the cluster cohesiveness is the usage of the weighted similarity of the internal cluster similarity [12].
- External Quality measure: Some external knowledge for the data is required. One external measure is entropy. It provides a measure of goodness for un-nested clusters or for the clusters at one level of a hierarchical clustering. Another external measure is the F-measure which measures the effectiveness of a hierarchical clustering [6].

**Shanon's Entropy:** Entropy is used as a measure of quality of the clusters [6]**.** For each cluster, the category distribution of data is calculated first i.e. let $p_{ij}$ be the probability that a member of cluster j belongs to category i. Then the entropy of each cluster j is calculated as [12]:

$$E_j = - \sum p_{ij} \log (p_{ij})$$

The total entropy is calculated by adding the entropies of each cluster weighted by the size of each cluster:

$$E_{en} = \sum_{j=1}^{m} ((n_j * E_j) / n)$$

Where $m$ is the total number of clusters, $n_j$ is the size of $j^{th}$ cluster and $n$ is the total number of documents.

**F-measure:** This is an aggregation of precision and recall concept of information retrieval. Precision is the ratio of the number of relevant documents to the total number of documents retrieved for a query. Recall is the ratio of the number of relevant documents retrieved for a query to the total number of relevant documents in the entire collection [12]. For cluster $j$ and class $i$

$$Recall\ (i, j) = n_{ij} / n_i$$
$$Precision\ (i, j) = n_{ij} / n_j$$

where $n_{ij}$ is the number of members of class $i$ in cluster $j$, $n_j$ is the number of members of cluster $j$ and $n_i$ is the number of members of class $i$.

The F-measure of cluster $j$ and class $i$ is calculated from precision and recall as

*F(i, j) = (2 * Recall(i, j) * Precision(i, j)) / ((Precision(i, j) + Recall(i, j))*

For an entire hierarchical clustering the F-measure of any class is the maximum value it attains at any node in the tree and an overall value for the F-measure is computed by taking the weighted average of all values for the F-measure as given by the following.

$$F = \sum_{i} n_i / n \max \{F(i, j)\}$$

where the max is taken over all clusters at all levels, and *n* is the number of documents [6]. Higher value of F-measure indicates better clustering [12].

In [22], authors have shown that F-measure has bias towards hierarchical clustering algorithms so $F_{norm}$ which is the normalized version of the F-measure is proposed to solve the cluster validation problem for hierarchical clustering. Experimental results show that $F_{norm}$ is more suitable than the unnormalized F-measure in evaluating the hierarchical clustering results across datasets with different data distribution.

## 6. CHALLENGES IN DOCUMENT CLUSTERING

Document clustering is being studied from many decades but still it is far from a trivial and solved problem. The challenges are [12]:

- Selection of appropriate features of the documents.
- Selection of appropriate similarity measure
- Selection of appropriate clustering method
- Assessment of the quality of the clusters.
- Implementation of the clustering algorithm in an efficient way by making optimal use of available memory and CPU resources.
- Associate meaningful label to each final cluster [1].
- To consider semantic relationship between words like synonyms [1].

In the context of hierarchical document clustering, some other major challenges are given in [23]:

- Very high dimensionality of the data: With medium to large document collections (10,000+ documents), the number of term-document relations is millions+, and the computational complexity of the algorithm applied is thus a central factor. If the vector model is applied, the dimensionality of the resulting vector space will likewise be 10,000+ [12]. The computational complexity should be linear with respect to the number of dimensions (terms).
- The algorithms must be efficient and scalable to large data sets.
- Overlapping between document clusters should be allowed.
- The algorithms must be able to update the hierarchy when new documents arrive (or are removed).
- The clustering algorithm should be able to find number of clusters on its own.

## 7. OVERVIEW OF VARIOUS DOCUMENT CLUSTERING ALGORITHMS

The HAC method is first applied to a substantial document collection, Croft, using the single linkage method. It was 11,613 titles taken from the UKCIS document test collection. Inverted file was used to generate the similarities needed by the clustering algorithm. It avoided calculation of many zero-valued inter-document similarities. But, Harding and Willett showed that though this procedure is very efficient for short documents, it can cause increase in the running time when large documents are used. So, Willett described an improved inverted file algorithm for the calculation of inter-document similarity coefficients where each document description be processed once only for the calculation of the similarities [20].

As seen in section 1, HAC algorithms are slow when applied to large document collections as single link and group average methods take O ($n^2$) time, whereas complete link methods take O ($n^3$) time. In terms of quality, complete link algorithms perform well. So, in order to balance quality and time complexity word intersection clustering (Word-IC) method is suggested in [24]. Word-IC is a HAC algorithm which uses Global Quality Function (GQF) as heuristic which makes it faster (O ($n^2$) time complexity) and results in higher quality clustering. Also phrase intersection clustering (Phrase-IC) is proposed which treats document as a sequence of words. Phrase-IC using suffix-tree is an O (nlogn) expected time algorithm with extra space requirement of suffix tree O (n) and construction time of suffix tree O (nlogn).

In [6], a simple and efficient variant of K-means, bisecting K-means, where centroids are updated incrementally, is introduced. It produces better clusters than those produced by regular K-means. Bisecting K-means has a linear time complexity. The authors have first compared three agglomerative hierarchical techniques namely Intra-Cluster Similarity Technique (IST), Centroid Similarity Technique (CST), and UPGMA. Results show that UPGMA is the best hierarchical technique, which is then, compared with K-means and bisecting K-means. Bisecting k-means is proved to be superior to UPGMA and regular k-means. The better performance of bisecting K-means is because of production of relatively uniform size clusters.

The information bottleneck method for unsupervised document clustering is presented in [25]. Information bottleneck method says that given the empirical joint distribution of two variables, one variable is compressed. Due to this the mutual information about the other is preserved. In case of document clustering, these two variables are the set of documents and the set of words. Given a joint empirical distribution of words and documents, p (x, y), first words, Y, are clustered to obtain Y' which maximally preserve the information on the documents. The resulting joint distribution, p (X, Y'), is much less sparse and noisy. Using the same procedure now the documents are clustered, X, so that the information about the word-clusters is preserved. Experiments show that this double clustering procedure yields significantly superior performance compared to other common document distributional clustering algorithms. But the agglomerative procedure used in this work has O($X^3$) time complexity.

Various weaknesses of k-means are prior knowledge of count of clusters, quadratic time complexity when large set of documents, high dimensionality, and mean as summary of clusters. In [26], a lightweight document clustering method to operate in high dimensionality is presented. The method uses a reduced indexing i.e. only the k best keywords of each document are indexed. The number of clusters is dynamically determined, and similarity is based on nearest-neighbor distance**.** The method has been evaluated on a database of

over 50,000 customer service problem reports to demonstrate efficiency of clustering performance.

In [27], authors have experimentally evaluated nine agglomerative algorithms and six partitional algorithms. Also agglomerative algorithm based on constraining the agglomeration process using clusters obtained by partitional algorithms is introduced. The experimental results have shown that partitional methods produce better hierarchical solutions than agglomerative methods, and that the constrained agglomerative methods improved the clustering solutions obtained by agglomerative or partitional methods alone. The experimental evaluation showed that partitional clustering algorithms are well-suited for clustering large document datasets due to their relatively low computational requirements and better clustering performance.

In [8], a novel approach which uses frequent item (term) sets for text clustering is presented. Frequent sets can be efficiently discovered using association rule mining. Two algorithms for frequent term-based text clustering are presented; FTC which creates flat clustering and HFTC for hierarchical clustering. An experimental evaluation on text documents as well as on web documents demonstrates that the proposed algorithms obtain clustering of comparable quality. Furthermore, frequent term sets provide understandable description and labels.
In [9], another frequent itemsets using association rule mining is proposed. By focusing on frequent items, the dimensionality of the document set is drastically reduced. This method outperforms best existing methods in terms of both clustering accuracy and scalability. Results are compared with five reference datasets against UPGMA, Bisecting k-means and HFTC.

A novel document partitioning method based on the non-negative factorization (NMF) of the term-document matrix is presented in [28]. The method differs from the latent semantic indexing method based on the singular vector decomposition (SVD) and the related spectral clustering methods. This is because semantic space derived by NMF does not need to be orthogonal, and each document takes only non-negative values in all the latent semantic directions. These two differences bring an important benefit that each axis in the space derived by the NMF has a much more straightforward correspondence with each document cluster. Experimental evaluations show that the proposed document clustering method surpasses SVD and the eigen decomposition clustering methods in the easy and reliable clustering results and clustering accuracy.

Instead of relying on single term analysis of the document data set, such as the Vector Space Model; we might achieve more accurate document clustering using more informative features including phrases and their weights. In [29], a phrase-based document index model, the Document Index Graph, which incrementally constructs a phrase-based index of the document set, is proposed. Also, an incremental document clustering algorithm based on maximizing the tightness of clusters is given. The integration of these two models creates robust and accurate document similarity calculation.

The Particle Swarm Optimization (PSO) algorithm is a population based stochastic optimization technique. It can be used to find an optimal, or near optimal, solution. The PSO algorithm can be used to generate initial cluster centroids for the K-means, the major requirement of K-means algorithm. In [7], a hybrid PSO+K-means document clustering algorithm which performs fast clustering and can avoid being trapped in a local optimal solution is proposed. PSO+K-means is compared with PSO, K-means, and other two hybrid clustering algorithms. The results illustrate that the PSO+K-means algorithm can generate the most compact clustering results.

Simultaneous clustering of documents and words is called co-clustering. In [30], a possibilistic fuzzy co-clustering (PFCC) for automatic categorization of large document collections is presented. PFCC integrates a possibilistic document clustering technique and fuzzy word ranking. This framework brings robustness in word outliers, rich representations of co-clusters, highly descriptive document clusters, a good performance in a high-dimensional space, and a reduced sensitivity to the initialization in the possibilistic clustering. It also improves the quality of the resulting co-clusters. Experiments on several large document data sets demonstrate the effectiveness of PFCC.
One more fuzzy co-clustering technique is given in [31] which perform simultaneous fuzzy clustering of objects and features. It is known to be suitable for categorizing high-dimensional data because of its dynamic dimensionality reduction mechanism. In [31] Heuristic Fuzzy Co-clustering with the Ruspini's condition (HFCR) is given. It addresses various issues like the performance on data sets with overlapping feature clusters and the unnatural representation of feature clusters. The key idea of HFCR is the formulation of the dual-partitioning approaches for fuzzy co-clustering by adopting an efficient and practical heuristic method. Experimental results on ten large benchmark document data sets confirm the effectiveness of this new algorithm.

An efficient method based on spectra analysis of eigen values of the data set to effectively estimate the number of clusters in a given data set, which is prerequisite of many well-known algorithms such as K-Means, EM, and CLOPE is proposed in **Error! Reference source not found.**. First the relationship between a data set and its underlying spectra with theoretical and experimental results is presented. Then capability of this method for suggesting a range of k is given. Empirical results are shown to cater to this fundamental problem to enhance the clustering process for large text collections.

In Bisecting K-means (BKM), a refinement is needed to re-cluster the resulting solutions when a fraction of the dataset is left behind. In [33] a cooperative bisecting k-means (CBKM) clustering algorithm is presented which concurrently combines the results of the BKM and KM at each level of the binary hierarchical tree using cooperative and merging matrices. Experimental results show that the CBKM achieves better clustering quality than KM, BKM, and single linkage (SL) algorithms with improvement in time complexity.

In [23] two clustering algorithms called dynamic hierarchical compact and dynamic hierarchical star are presented. These methods aim to construct a cluster hierarchy, dealing with dynamic data sets. The first method creates disjoint hierarchies of clusters, and the second obtains overlapped hierarchies. The experimental results on benchmark text collections prove the effectively and efficiency of these methods to deal with dynamic datasets.

An effective Fuzzy Frequent Itemset-Based Hierarchical Clustering ($F^2$IHC) approach is presented in [9], which uses fuzzy association rule mining algorithm to improve the clustering accuracy of FIHC method. The key terms are extracted from the document set. Then, a fuzzy association rule mining algorithm for text is employed to discover a set of highly-related fuzzy frequent itemsets, Finally, these documents are clustered into a hierarchical cluster tree by referring to these candidate clusters. The experimental results show the improvement in the accuracy and quality of FIHC. Also key terms are useful as the labels of the candidate clusters.

As discussed in section 1, divisive clustering has good computational efficiency but degraded performance. Also, which cluster should be split and how to split the selected cluster are two major issues to consider. To tackle this problem, [34] proposes a new divisive clustering algorithm integrating an improved discrete PSO into a divisive clustering framework. The proposed algorithm performs better or at least comparable in terms of clustering quality and robustness. It runs much faster and is also very stable compared with the other clustering algorithms. In addition, it scales well.

In [35], authors have studied fast Self Organizing Map (SOM) clustering technology for text information. The system has two stages: offline and online. Feature extraction and semantic quantization are done offline to make text clustering more efficient. Neurons are represented as numerical vectors in high-dimension space and documents are represented as collections of important keywords. Then fast clustering techniques for online stage are proposed including how to project documents onto output layers in SOM, fast similarity computation method and the scheme of incremental clustering technology for real-time processing. The time complexity of SOM is O ( k'm*n), where k' is the number of neurons, m is the training time and n is the document number (m*n samples need to be inputted to train the network).

NMF has been widely used to generate flat clusters of text documents. It has features of preserving the local structure of original data and dimension reduction. But, the in NMF clustering results are sensitive to the initial values of the parameters. In order to overcome this drawback, the ensemble NMF for clustering biomedical documents is presented in [36]. The performance of ensemble NMF was evaluated on numerous datasets generated from the TREC Genomics track dataset and it is found that it outperforms classical clustering algorithms of bisecting K-means, and hierarchical clustering.

NMF and Matrix factorization-based techniques see only the global Euclidean geometry, whereas the local manifold geometry is not considered. A new approach to extract the document concepts which are consistent with the manifold geometry is proposed in [37]. Central to the approach is a graph model which captures the local geometry of the document sub-manifold; called as Locally Consistent Concept Factorization (LCCF). By using this graph to smooth the document-to-concept mapping, documents associated with the same concept can be well clustered. The experimental results have shown that the proposed approach provides a better representation and achieves better clustering results in terms of accuracy and mutual information.

In [38], a number of methods and tools to cluster a 7000 document inventory are discussed. The inventory which is not publicly available has research documents, influential policy documents, and policy documents. Here a full text analysis is performed on more than 300,000 pages in total on new analysis platform. To represent the results two visualization techniques are employed and compared, multi-dimensional scaling (MDS) and the derivative of self organizing map, U-Matrix. The combination of a U-matrix and an MDS map reveals information that would go unnoticed otherwise.

As seen earlier, the performance of k-means depends on the initial state of centroids and may trap in local optima. The gravitational search algorithm (GSA) is one effective method for searching problem space to find a near optimal solution. So, in [39], a hybrid data clustering algorithm combining advantages of both algorithms GSA and k-means (GSA-KM) is presented. The GSA-KM algorithm helps the k-means algorithm to escape from local optima and increases the convergence speed of the GSA algorithm. The performance of GSA-KM is compared with k-means, genetic algorithm, simulated annealing, ant colony optimization, honey bee mating optimization, particle swarm optimization and gravitational search algorithm. The experimental results have shown higher quality and the convergence speed of the proposed algorithm.

# 8. CONCLUSION

Document clustering is a fundamental operation used in unsupervised document organization, automatic topic extraction, and information retrieval. In this paper, we have explained the document clustering procedure with feature selection, various improvements for it, TF.IDF process, dimension reduction mechanisms etc. We also have highlighted on applications, challenges, similarity measures and evaluation of document clustering algorithms. We have tried to provide detailed and exhaustive overview of various document clustering methods studied and researched since last fifteen years, starting from basic traditional methods to fuzzy based, genetic, co-clustering, heuristic-oriented, NMF, etc. We feel this survey paper will be very useful in thriving research area of document clustering.

# 9. REFERENCES

[1] Rekha Baghel and Dr. Renu Dhir, "A Frequent Concepts Based Document Clustering Algorithm," *International Journal of Computer Applications*, vol. 4, No.5, pp. 0975 – 8887, Jul. 2010

[2] A. Huang, "Similarity measures for text document clustering," In *Proc. of the Sixth New Zealand Computer Science Research Student Conference NZCSRSC*, pp. 49—56, 2008.

[3] Nicholas O. Andrews and Edward A. Fox, "Recent developments in document clustering," *Technical report published by citeseer,* pp. 1-25, Oct. 2007

[4] Chun-Ling Chen, Frank S.C. Tseng, and Tyne Liang, "An integration of WordNet and fuzzy association rule mining for multi-label document clustering," *Data and Knowledge Engineering*, vol. 69, issue 11, pp. 1208-1226, Nov. 2010

[5] Yong Wang and Julia Hodges, "Document Clustering with Semantic Analysis," *In Proc. of the 39th Annual*

*Hawaii International Conference on System Sciences, HICSS 2006,* vol. 03, pp. 54.3

[6] Michael Steinbach , George Karypis, and Vipin Kumar, "A comparison of document clustering techniques," *In KDD Workshop on Text Mining,* 2002

[7] Xiaohui Cui and Thomas E. Potok, "Document Clustering Analysis Based on Hybrid PSO+K-means Algorithm," Special Issue, 2005

[8] F. Beil, M. Ester, and X. Xu, "Frequent term-based text clustering," *Proc. of Int'l Conf. on knowledge Discovery and Data Mining (KDD'02),* pp. 436–442, 2002.

[9] Benjamin C.M. Fung, Ke Wang, and Martin Ester, "Hierarchical Document Clustering Using Frequent Itemsets," In *Proc. Siam International Conference On Data Mining 2003,* SDM 2003

[10] Chun-Ling Chen, Frank S. C. Tseng, and Tyne Liang, "Mining fuzzy frequent itemsets for hierarchical document clustering," *Published in an Int'l Journal of Information Processing and Management,* vol. 46, issue 2, pp. 193-211, Mar. 2010

[11] C.L. Chen, F.S.C. Tseng, T. Liang, An integration of fuzzy association rules and WordNet for document clustering, *Proc. of the 13th Pacific-Asia Conference on Knowledge Discovery and Data Mining (PAKDD-09),* 2009, pp. 147–159.

[12] Pankaj Jajoo, "Document Clustering," Masters' Thesis, IIT Kharagpur, 2008

[13] Chih-Ping Wei, Chin-Sheng Yang, Han-Wei Hsiao, and Tsang-Hsiang Cheng, "Combining preference- and content-based approaches for improving document clustering effectiveness," *Published in Int'l Journal of Information Processing and Management,* vol. 42, issue 2, pp. 350-372, Mar. 2006

[14] MS. K.Mugunthadevi, MRS. S.C. Punitha, and Dr..M. Punithavalli, "Survey on Feature Selection in Document Clustering," *Int'l Journal on Computer Science and Engineering (IJCSE)*, vol. 3, No. 3, pp. 1240-1244, Mar 2011

[15] Yi Peng, Gang Kou, Zhengxin Chen, and Yong Shi, "Recent trends in Data Mining (DM): Document Clustering of DM Publications," *Int'l Conference on Service Systems and Service Management,* vol. 2, pp. 1653 – 1659, Oct. 2006

[16] Man Lan, Chew Lim Tan, Jian Su, and Yue Lu, "Supervised and Traditional Term Weighting Methods for Automatic Text Categorization," *IEEE Transactions on Pattern Analysis and Machine Intelligence,* vol. 31, No. 4, Apr. 2009

[17] Shen Huang, Zheng Chen, Yong Yu, and Wei-Ying Ma, "Multitype Features Coselection for Web Document Clustering," *IEEE Transactions on Knowledge and Data Engineering*, vol. 18, No. 4, Apr. 2006

[18] Minqiang Li and Liang Zhang, "Multinomial mixture model with feature selection for text clustering," *Journal of Knowledge-Based Systems*, vol. 21, issue 7, pp. 704-708, Oct. 2008

[19] Jun Yan, Ning Liu, Shuicheng Yan, Qiang Yang, Weiguo (Patrick) Fan, Wei Wei, and Zheng Chen, "Trace-Oriented Feature Analysis for Large-Scale Text Data Dimension Reduction," *IEEE Transactions on Knowledge and Data Engineering,* vol. 23, No. 7, Jul. 2011

[20] Peter Willett, "Recent Trends In Hierarchic Document Clustering: A Critical Review," *Information Processing & Management*, vol. 24, No. 5, pp. 517-597, 1988

[21] Congnan Luo, Yanjun Li, and Soon M. Chung, "Text document clustering based on neighbors," *Data and Knowledge Engineering 68,* pp. 1271–1288, 2009

[22] Junjie Wu, Hui Xiong, and Jian Chen, "Towards understanding hierarchical clustering: A data distribution perspective," *Neurocomputing 72,* pp. 2319–2330, 2009

[23] Reynaldo Gil-García and Aurora Pons-Porrata, "Dynamic hierarchical algorithms for document clustering," *Pattern Recognition Letters 31,* pp. 469–477, 2010

[24] Oren Zamir, Oren Etzioni, Omid Madani, and Richard M. Karp, "Fast and intuitive clustering of web documents citation," *In Proc. of the 3rd Int'l Conference on Knowledge Discovery and Data Mining,* 1997

[25] Noam Slonim and Naftali Tishby, "Document Clustering using Word Clusters via the Information Bottleneck Method," *In Proc. of the 23rd annual Int'l ACM SIGIR conference on Research and development in information retrieval,* pp. 208 – 215, 2000

[26] Sholom Weiss, Brian White, and Chid Apte, "Lightweight document clustering," *IBM Research Report RC-21684,* 2000

[27] Ying Zhao and George Karypis, "Evaluation of Hierarchical Clustering Algorithms for Document Datasets", *Technical Report*, Jun. 2002

[28] Wei Xu, Xin Liu, and Yihong Gong, "Document Clustering Based On Non-negative Matrix Factorization," *In Proc. of the 26th annual international ACM SIGIR conference on Research and development in information retrieval,* pp. 267-273, 2003

[29] Khaled M. Hammouda and Mohamed S. Kamel, "Efficient Phrase-Based Document Indexing for Web Document Clustering," *IEEE Transactions on Knowledge and Data Engineering,* vol. 16, No. 10, Oct. 2004

[30] William-Chandra Tjhi and Lihui Chen, "Possibilistic fuzzy co-clustering of large document collections," *Journal of Pattern Recognition,* vol. 40, issue 12, pp. 3452-3466, Dec. 2007

[31] William-Chandra Tjhi and Lihui Chen, "A heuristic-based fuzzy co-clustering algorithm for categorization of high-dimensional data," *Journal of Fuzzy Sets and Systems,* vol. 159, issue 4, pp. 371-389, Feb. 2008

[32] Wenyuan Li, Wee-Keong Ng, Ying Liu, and Kok-Leong Ong, "Enhancing the Effectiveness of Clustering with Spectra Analysis," *Journal of IEEE Transactions on Knowledge and Data Engineering,* vol. 19, issue 7, pp. 887-902, Jul. 2007

[33] R. Kashef and M.S.Kamel, "Enhanced bisecting *k*-means clustering using intermediate cooperation," *Journal of*

*Pattern Recognition,* vol. 42, issue 11, pp. 2557-2569, Nov. 2009

[34] Liang Feng, Ming-Hui Qiu, Yu-Xuan Wang, Qiao-Liang Xiang, Yin-Fei Yang, and Kai Liu, "A fast divisive clustering algorithm using an improved discrete particle swarm optimizer," *Journal of Pattern Recognition Letters*, vol.31, issue 11, pp. 1216-1225, Aug. 2010

[35] Yuan-chao Liu, Chong Wu, and Ming Liu, "Research of fast SOM clustering for text information," *An International Journal Expert Systems with Applications*, vol. 38, issue 8, pp. 9325-9333, Aug. 2011

[36] Xiaodi Huang, Xiaodong Zheng, Wei Yuan, Fei Wang, and Shanfeng Zhu, "Enhanced clustering of biomedical documents using ensemble non-negative matrix factorization," *an International Journal on Information Sciences,* vol. 181, issue 11, pp. 2293-2302, Jun. 2011

[37] Deng Cai, Xiaofei He, and Jiawei Han, "Locally Consistent Concept Factorization for Document Clustering," *IEEE Transactions on Knowledge and Data Engineering,* vol.23, no.6, pp.902-913, Jun. 2011

[38] Patrick A. De Maziere and Marc M. Van Hulle, "A clustering study of a 7000 EU document inventory using MDS and SOM," *An International Journal on Expert Systems with Applications*, vol. 38, issue 7, pp. 8835-8849, Jul. 2011

[39] Abdolreza Hatamloua, Salwani Abdullah, and Hossein Nezamabadi-pour, "A combined approach for clustering based on K-means and gravitational search algorithms," *Swarm and Evolutionary Computation*, Available online 12 Mar. 2012