

# A Review on Meta Information Based Text Data Clustering.

Ms. Mrunal V. Upasani<sup>1</sup>, Prof. Rucha C. Samant<sup>2</sup>

<sup>1</sup>Student, Computer Engg. Department, GES's R. H. Sapat COE Nashik  
Pune University, India

<sup>2</sup>Professor, Computer Engg. Department, GES's R. H. Sapat COE Nashik  
Pune University, India

**Abstract:** Clustering is a widely studied data mining problem in the text domains. This problem finds numerous applications in classification, visualization, document organization, collaborative filtering and indexing. This paper provides a detailed survey of the problem of text clustering. It will discuss the key methods used for text clustering, and their relative advantages. The problem of classification has also been widely studied in the database, data mining, and information retrieval community's. The paper will also provide a survey of a wide variety of text classification algorithms. In most cases, the data is not available only in the purely text form. Large amount of side-information is available along with the text documents. This side-information may be of different kinds, such as the links in the document or webpage, user-access behaviour from web logs or links or other non-textual attributes which are embedded into the text document. Such unstructured attributes may contain a tremendous amount of information for clustering purposes. Therefore, this paper will use an approach which carefully ascertains the coherence of the clustering characteristics of the side information with that of the text content. This will help in improving the quality of the clustering effects of both the text data and side information. The paper then shows how to extend the clustering approach to the classification problem using the side information of the text documents.

**Keywords:** Classification, Clustering, Data mining, Side information, Text Mining.

## I. INTRODUCTION

Clustering is the process in which data objects are organized into a set of disjoint classes called clusters. Objects that are in the same cluster are similar among themselves and dissimilar to the objects belonging to other clusters. The problem of clustering has been studied widely in the database and statistics literature in the context of a wide variety of data mining tasks. The clustering problem is defined as the process of finding groups for similar objects in the data. This similarity between the objects is measured by the use of a similarity function. Clustering is very useful in the text domain, where the objects to be clusters are of different sizes like documents, paragraphs, sentences or terms. Clustering is useful for organizing documents for improving retrieval and support browsing.

Document clustering has been investigated for use in a number of different areas of text mining and information retrieval. Document clustering has also been used to automatically generate hierarchical clusters of documents.

The problem of text clustering arises in the context of many application domains such as the social networks, web and other digital collections. The rapid increase in amount of text data in the context of these large online collections has led to an interest in creating scalable and effective mining algorithms. A large amount of work has been done in recent years on the problem of clustering in text collections. The survey of text clustering algorithms is given in [4].

In many application domains, a large amount of side information is also associated along with the documents. Because text documents typically occur in the variety of applications in which there may be a large amount of other kinds of database attributes or meta-information which may be useful to the clustering process. Examples of such side-information are discussed below.

The application in which the system tracks user access behaviour of web documents. The access behaviour of use may be captured in the form of web logs. For every document, the meta-information corresponds to the browsing behaviour of the different users. These logs can be used to enhance the quality of the mining process which is more meaningful to the user, and also application-sensitive. Because the logs can often pick up subtle correlations in content, which cannot be picked up by the raw text alone.

Many text documents contain links in between them, which can also be treated as attributes. These links contain a lot of useful information for mining purposes.

Many web documents have meta-data associated with them which correspond to different kinds of attributes such as the provenance or other information like ownership, location, or even temporal information about the origin of the document. In a number of network and user-sharing applications, documents which are associated with user-tags, may also be quite informative.

Such side-information can be useful in improving the quality of the clustering process [1]. The primary goal of this paper is to study the clustering of data in which auxiliary information is available with text.

Such scenarios are very common in a wide variety of data domains. Therefore, the paper extend the clustering approach to the problem of classification, which provides superior results because of the incorporation of side information

The problem of classification has been widely studied in the data mining, information retrieval and database communities with applications in a number of diverse domains, such as target marketing, diagnosis in medical field, filtering of news groups, and document organization. Text classification finds applications in a wide variety of domains in text mining like filtering and organization of news, document organization and retrieval, sentimental analysis, Email classification and spam filtering etc.

Goal of this paper is to show that the advantages of using side-information extend beyond a pure clustering task, and can provide competitive advantages for a wider variety of problem scenarios.

The above section discusses the introduction of the clustering, classification and some examples of Meta information available with the documents. Section II describes the literature survey of classification and clustering. Sections III formalize the problem of text clustering with side information. Sections IV discuss how to extend these clustering techniques to the classification problem. Section V contains the conclusion.

**II. LITERATURE SURVEY**

A tremendous amount of work has been done in recent years on the problem of clustering in text collections in the database and information retrieval communities.

The Survey of Text Clustering Algorithms is studied in [3] [9]. There are many types of Distance-based Clustering Algorithms like Agglomerative and Hierarchical Clustering Algorithm, Distance-based Partitioning Algorithms like k-means clustering, A Hybrid Approach for clustering like scatter/gather technique etc.

In Distance-based Clustering Algorithms there is a use of similarity function which measures the closeness between the text objects takes place. The most well-known similarity function which is used commonly in the text domain is the cosine similarity function. These similarity functions can be used in conjunction with a wide variety of traditional clustering algorithms like Agglomerative and Hierarchical Clustering Algorithms.

The general concept of agglomerative clustering is to successively merge documents into clusters based on their similarity with one another. Almost all the hierarchical clustering algorithms successively merge groups based on the best pairwise similarity between these groups of documents. The main differences between these classes of methods are in terms of how this pairwise similarity is computed between the different groups of documents. Conceptually, the process of agglomerating documents into successively higher levels of clusters creates a cluster hierarchy for which the leaf nodes correspond to individual documents, and the internal nodes correspond to the merged groups of clusters. When two groups are merged, a new node is created in this tree corresponding to this larger merged group. The two children of this node correspond to the two groups of documents which have been merged to it

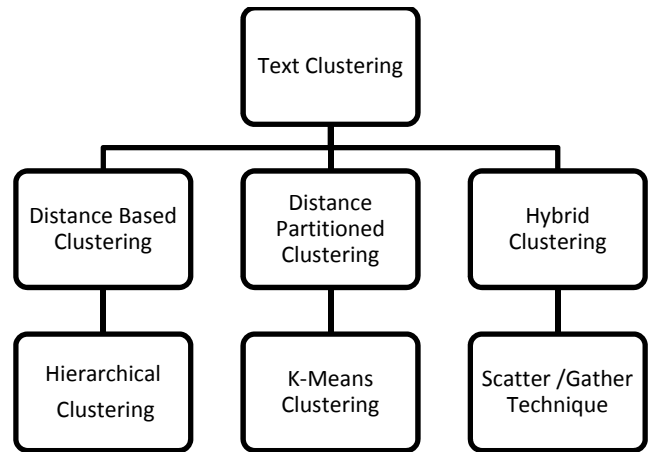


Fig 1. Types of Text Clustering Techniques

A hierarchical clustering algorithm called CURE that is more robust to outliers, and identifies clusters having non-spherical shapes and wide variances in size is given in [7]. Another hierarchical Clustering algorithm i.e. ROCK: A Robust Clustering Algorithm for Categorical Attributes for data with Boolean and categorical attributes is studied in [8]. It employs links and not distances for merging clusters. [11] Presents the hierarchical data clustering method BIRCH (Balanced Iterative Reducing and Clustering using Hierarchies) and it demonstrates that it is especially suitable for very large databases.

The next method of document clustering is Distance based Partitioning algorithm. Partitioning algorithms are widely used in the database literature in order to efficiently create clusters of objects. K-means clustering algorithm is a partitioning algorithm [3]. It uses a set of k representatives, around which the clusters are built. In particular, K-means uses the notion of a centroid, which is the mean or median point of a group of points. Note that a centroid almost never corresponds to an actual data point. The simplest form of the k-means approach is to start off with a set of k seeds from the original corpus, and assign documents to these seeds on the basis of closest similarity. In the next iteration, the centroid of the assigned points to each seed is used to replace the seed in the last iteration. In other words, the new seed is defined, so that it is a better central point for this cluster. This approach is continued until convergence. Simultaneous Clustering and Dynamic Keyword Weighting for Text Documents takes place in [6]. It uses the approach to extend K-means algorithm, that in addition to partitioning the dataset into a given number of clusters, also finds the optimal set of feature weights for each clusters. [12] Combines an efficient online spherical k-means (OSKM) algorithm with an existing scalable clustering strategy to achieve fast and adaptive clustering of text streams. The OSKM algorithm modifies the spherical k-means (SPKM) algorithm, using online update (for cluster centroids) based on the well-known Winner-Take-All competitive learning.

The third type of document clustering is the Hybrid Technique. Scatter-gather technique is the hybrid clustering technique [3] [5]. A classic example of this is the Scatter/Gather method, which provides a systematic browsing technique with the use of clustered organization of the document collection. Initially the system scatters the collection into a small number of document groups, or clusters, and presents short summaries of them to the user. Based on these summaries, the user selects one or more of the groups for further study. The selected groups are gathered together to form a sub collection. The system then applies clustering again to scatter the new sub collection into a small number of document groups, which are again presented to the user. With each successive iteration the groups become smaller, and therefore more detailed. Ultimately, when the groups become small enough, this process bottoms out by enumerating individual documents. The scatter-gather approach can be used for organized browsing of large document collections, because it creates a natural hierarchy of similar documents.

However, all of these methods are designed for the case of pure text data, and do not work for cases in which the text-data is combined with other forms of data. Some limited work has been done on clustering text in the context of network-based linkage information like graph mining and algorithms of graph mining in [2] [10].

A wide variety of techniques have been designed for text classification in [4]. There are different techniques for classification of the data such as Probabilistic and Naive Bayes Classifiers. Probabilistic classifiers are designed to use an implicit mixture model for generation of the underlying documents. This mixture model typically assumes that each class is a component of the mixture. Each mixture component is essentially a generative model, which provides the probability of sampling a particular term for that component or class.

Next type of classifier is Decision Tree Classifiers. Decision tree is essentially a hierarchical decomposition of the (training) data space, in which a predicate or a condition on the attribute value is used in order to divide the data space hierarchically. The division of the data space is performed recursively in the decision tree, until the leaf nodes contain a certain minimum number of records, or some conditions on class purity. The majority class labelling the leaf node is used for the purposes of classification.

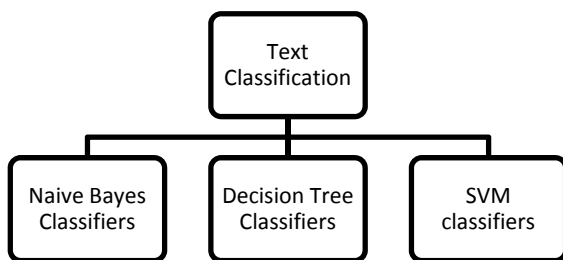


Fig 2. Types of Text Classification Techniques

Another type of classifier is SVM classifiers, the main principle of SVM is to determine separators in the search space which can best separate the different classes. SVM training algorithm builds a model that assigns new examples into one category or the other, making it a non-probabilistic binary linear classifier. SVMs can efficiently perform a non-linear classification using what is called the kernel trick, implicitly mapping their inputs into high-dimensional feature spaces.

All this work is not applicable to the case of general side information attributes. This paper will provide a first approach to using other kinds of attributes in conjunction with text clustering. It will show the advantages of using such an approach over pure text-based clustering. Such an approach is especially useful, when the auxiliary information is highly informative, and provides effective guidance in creating more coherent clusters. It will also extend the method to the problem of text classification.

### III. CLUSTERING TECHNIQUE FOR THE SIDE INFORMATION

The focus of this paper is to show the advantages of using side-information for mining text data extend beyond a pure clustering task which provides competitive advantages for a wider variety of problem scenarios.

The algorithm used for clustering of side information is COATES Algorithm, which corresponds to the fact that it is a COntent and Auxiliary attribute based TExt cluStering algorithm. The algorithm requires two phases

#### A. Initialization

It is a lightweight initialization phase in which a standard text clustering approach is used without any side-information. For this purpose, it uses the k-means clustering algorithm. The reason that this algorithm is used, it is a simple algorithm which can quickly and efficiently provide a reasonable initial starting point. The partitioning and the centroids created by the clusters formed in the first phase provide an initial starting point for the second phase. The first phase is based on text information only, not the auxiliary information.

#### B. Main Phase

This phase starts off with these initial groups, and iteratively reconstructs these clusters with the use of both the text content and the auxiliary information. Alternating iterations which use the text content and auxiliary attribute information in order to improve the quality of the clustering are performed in this step. These iterations are content iterations and auxiliary iterations respectively. The combination of the content iteration and auxiliary iteration is referred to as a major iteration. Each major iteration contains two minor iterations, which corresponding to the auxiliary and text-based methods respectively.

### IV. CLASSIFICATION BASED ON CLUSTERING OF SIDE INFORMATION

In this section, discussion of how to extend the approach to classification is done. This paper will extend the earlier clustering approach in order to incorporate

supervision, and it creates a model which summarizes the class distribution in the data in terms of the clusters. Then, it will show how to use the summarized model for effective classification.

For extension of classification to the problem of clustering, COLT algorithm is used for classification of side information which refers to the fact that it is a Content and auxiliary attribute-based Text classification algorithm. This algorithm uses a supervised clustering approach in order to partition the data into different clusters. This partitioning is then used for the purposes of classification.

The algorithm works in 3 steps.

#### A. Feature Selection

In the first step, it uses feature selection to remove the attributes, which are not related to the class label. It is performed both for the text attributes and the auxiliary attributes.

#### B. Initialization

In this step, it uses a supervised k means approach in order to perform the initialization, with the use of purely text content. The class memberships of the records in each cluster are pure for the case of supervised initialization. Thus, the k-means clustering algorithm is modified, so that each cluster only contains records of a particular class.

#### C. Cluster-Training Model Construction

In this phase, a combination of the text and side-information is used for the purposes of creating a cluster-based model. As in the case of initialization, the purity of the clusters is maintained during this phase.

## V. CONCLUSION

This paper gives the brief introduction about the broad field of document clustering and classification. The techniques which are used for clustering like k-means, hierarchical etc. and classifications like naïve bayes, SVM etc. are discussed. This paper also presented methods for mining text data with the use of side-information. Many forms of text databases contain a large amount of side-information or meta-information, which may be used in order to improve the clustering process. In order to design the clustering method, combination of an iterative partitioning technique with a probability estimation process which computes the importance of different kinds of side-

information takes place. COATES and COLT approach can greatly enhance the quality of text clustering and classification, while maintaining a high level of efficiency.

## ACKNOWLEDGMENT

We are glad to express our sentiments of gratitude to all who rendered their valuable guidance to us. We would like to express our appreciation and thanks to Prof. Dr. P. C. Kulkarni, Principal, G. E. S. R. H. Sapat College of Engg., Nashik. We are also thankful to Prof. N. V. Alone, Head of Department, Computer Engg., G. E. S. R. H. Sapat College of Engg., Nashik. We thank the anonymous reviewers for their comments.

## REFERENCES

- [1] Charu C. Aggarwal, Fellow, IEEE, Yuchen Zhao, and Philip S. Yu, Fellow, IEEE, "On the Use of Side Information for Mining Text Data", IEEE Transactions On Knowledge And Data Engineering, Vol. 26, No. 6, June 2014
- [2] C. C. Aggarwal and H. Wang, "Managing and Mining Graph Data", New York, NY, USA: Springer, 2010.
- [3] C. C. Aggarwal and C.-X. Zhai, "Mining Text Data", New York, NY, USA: Springer, 2012.
- [4] C. C. Aggarwal and C.-X. Zhai, "A survey of text classification algorithms", in Mining Text Data. New York, NY, USA: Springer, 2012.
- [5] D. Cutting, D. Karger, J. Pedersen, and J. Tukey, "Scatter/Gather: A cluster-based approach to browsing large document collections", in Proc. ACM SIGIR Conf., New York, NY, USA, 1992, pp. 318-329.
- [6] H. Frigui and O. Nasraoui, "Simultaneous clustering and dynamic keyword weighting for text documents," in Survey of Text Mining, M. Berry, Ed. New York, NY, USA: Springer, 2004, pp. 45-70.
- [7] S. Guha, R. Rastogi, and K. Shim, "CURE: An efficient clustering algorithm for large databases," in Proc. ACM SIGMOD Conf., New York, NY, USA, 1998, pp. 73-84.
- [8] S. Guha, R. Rastogi, and K. Shim, "ROCK: A robust clustering algorithm for categorical attributes," *Inf. Syst.*, vol. 25, no. 5, pp. 345-366, 2000.
- [9] M. Steinbach, G. Karypis, and V. Kumar, "A comparison of document clustering techniques," in Proc. Text Mining Workshop KDD, 2000, pp. 109-110.
- [10] Y. Sun, J. Han, J. Gao, and Y. Yu, "iTopicModel: Information network integrated topic modelling," in Proc. ICDM Conf., Miami, FL, USA, 2009, pp. 493-502.
- [11] T. Zhang, R. Ramakrishnan, and M. Livny, "BIRCH: An efficient data clustering method for very large databases," in Proc. ACM SIGMOD Conf., New York, NY, USA, 1996, pp. 103-114.
- [12] S. Zhong, "Efficient streaming text clustering," *Neural Netw.*, vol. 18, no. 5-6, pp. 790-798, 2005.