International Journal of Emerging Trend in Engineering and Basic Sciences (IJEEBS)
ISSN (Online) 2349-6967
Volume 2 , Issue 1(Jan-Feb 2015), PP176-180

Survey Paper on Text Mining with Side Information

Miss Shubhangi Airekar[1], Prof Dhanshree Kulkurni[2]
[1]*Department of Computer Science, D.Y. Patil College of Engineering, Ambi, Pune, India*
[2]*Department of Information Technology, D.Y. Patil College of Engineering, Ambi, Pune, India*

Abstract :- Any text mining application may contain side information. This side information may be any links in the document, web logs which contain user access behavior, provenance information, the links for any document or any other non textual attributes which are embedded into the text document. All these attributes may contain a huge amount of information for clustering purposes. But it is difficult to count the concerned importance of this side information especially when some of the data is noisy. In that matter, it is dangerous to merge side-information into the mining process because it can upgrade the quality of the representation for the mining process or can add noise in this system. Thus, there should be a right way to do this mining process so that it will make use of side information to maximize their advantages. Therefore, it is suggested to design an efficient algorithm which makes combination of classical portioning algorithm with probabilistic models in order to create an effective clustering approach. Afterwards, extension to the classification problem is also shown.

Keywords: - Clustering, Classifier Information, Data Mining, Text mining, , Text Collection.

I. INTRODUCTION

The text clustering issue comes in many type of application domain such as the web, social networks and other digital data. The rapidly increasing amount of text data in the surrounding of this large online collection is the main reason to create efficient and scalable mining algorithms. A lot of work has been done on the issue of clustering in text collection in text collection in the database and information retrival communities. Despite the work is mainly designed fo0r the pure text clustering purpose when other kinds of attributes are absent. Here some example of such side-information is given below.

- Web logs contain Meta information which gives information related to browsing behavior of various users. We can track such web logs. Such logs can be used to improve the quality of the text mining. This is because such logs can often catch sharp interrelation in content which cannot be catched by the raw text alone.
- A lot of text documents having connections among them are also called as attributes. Such links possess a lot of useful information for mining purpose. As in the later case, such attributes may often give insights about the correlation among documents in a way which may not be easily accessible from raw context.
- Meta data which are present with many web documents may correspond to different kinds of attributes such as provenance or other information about the source of the document. Temporal information, data such as ownership, location can also be information for mining purposes. Documents with user tags also come here in case of network and user sharing application.

  Side information can be additional feature for raising the quality of the clustering process but it can be dangerous when the side information is noisy. At that time it can actually degrade the quality of the mining process. Hence a approach is used which carefully ascertains the coherence of the clustering characteristics of the side information with that of the text content. This helps in managing the clustering effects in both helpful and noisy data.

  The main approach of this paper is to determine a clustering in which the text attributes and side-information provide same indications about the nature of the underlying clusters and at the same time ignore aspects in which conflicting indications are provided.

  For achieving this goal, portioning approach is merged with probabilistic evaluation method which decides the attachment of the side-information in the clustering process. A probabilistic evaluation process on the side information uses the portioning information for the purpose of evaluating the attachment of different

While our primary goal in this paper is to study the clustering problem, we note that such an approach can also be extended in principle to other data mining problems in which auxiliary information is available with text. This is very common in very wide range of data domains. Therefore a method is proposed in this paper to extend the approach to the problem classification.

II. LITERATURE SURVEY

Paper presented by Charu C. Aggarwal, Philip S. Yu demonstrates [1] that real time clustering and segmentation of text data records is required in many applications such as news group filtering, text crawling, and document organization. The categorical data stream clustering problem also has a number of applications to the problems of customer segmentation and real time trend analysis. By making the use of a statistical summarization methodology, an online approach for clustering massive text and categorical data streams is presented here.

Paper presented by Douglass R. Cutting, David R. Karger, Jan O. Pedersen, and John W. Turkey demonstrates [2] that for information retrieval, document clustering has not been well used. There are two main categories for its objection: first, for large corporation clustering is too slow and second, that retrieval is not improved by clustering. When clustering is used in an to improve conventional search techniques then only such problems are coming. However, t clustering as an information access tool in its own right obviates these objections, and provides a powerful new access paradigm. Document clustering is presented as primary operation in document browsing technique. Fast clustering algorithms are also presented which support this interactive browsing paradigm.

Paper presented by Douglass Michael Steinbach George Karypis Vipin Kumar demonstrates [3] that results of an experimental study of some common document clustering techniques are presented here. In particular, two main approaches to document clustering, agglomerative hierarchical clustering and K-means are compared here. Hierarchical clustering is always the better quality clustering approach, but has limitation due to its quadratic time complexity. In contrast, K-means and its variants have a time complexity which is linear in the number of documents, but are thought to produce inferior clusters. Sometimes K-means and agglomerative hierarchical approaches are merged so as to “get the best of both worlds.” However, the results shows that the bisecting K-means technique is better than the standard K means approach and as good or better than the hierarchical approaches that we tested for a variety of cluster evaluation metrics. An explanation for these results that is based on an analysis of the specifics of the clustering algorithms and the nature of document data is proposed here.

Paper presented by S. Guha, R. Rastogi, and K. Shim demonstrates [4] that for discovering groups and identifying interesting distributions in the underlying data clustering is used in data mining. Traditional clustering algorithms either favor clusters with spherical shapes and similar sizes. In this paper a clustering algorithm is presented which is called CURE that is more robust to outliers, and identifies clusters having non-spherical shapes and wide variances in size. CURE achieves this by representing each cluster by a certain fixed number of points that are generated by selecting well scattered points from the cluster and then shrinking them toward the center of the cluster by a specified fraction. Having more than one representative point per cluster allows CURE to adjust well to the geometry of non-spherical shapes and the shrinking helps to dampen the effects of outliers. CURE employs a combination of random sampling and partitioning to handle large databases. A random sample drawn from the data set is first partitioned and each partition is partially clustered. The partial clusters are then clustered in a second pass to gain the desired clusters. In this paper experimental results shows that the quality of clusters produced by CURE is much better than those found by existing algorithms. Further, in this paper it is demonstrated that random sampling and partitioning enable CURE to not only outperform existing algorithms but also to scale well for large databases without sacrificing clustering quality.

Paper presented by S. Zhong demonstrates [5] that clustering data streams has been a new research topic, recently used in many real data mining applications, and has attracted a lot of research attention. However, there is not much work on clustering high-dimensional streaming text data. This paper merges an efficient online spherical k-means algorithm with an existing scalable clustering strategy to achieve fast and adaptive clustering of text streams. The OSKM algorithm modifies the spherical k-means algorithm, using online update based on the well known. Winner Take All competitive learning. It has been shown to be as efficient as SPKM, but much superior in clustering quality. The scalable clustering strategy was previously developed to deal with very large data bases that cannot fit into a limited memory and that are too expensive to read/scan multiple times. Using this method, one keeps only sufficient statistics for history data to retain (part of) the contribution of history data and to accommodate the limited memory. To make the proposed clustering algorithm adaptive to data streams, a forgetting factor is introduced here that applies exponential decay to the importance of history data. The older a set of text documents, the less weight they carry. The experimental results demonstrate the efficiency of the proposed algorithm and reveal an intuitive and an interesting fact for clustering text streams—one needs to forget to be adaptive.

III. SYSTEM ANALYSIS

A. Proposed System

1. Clustering With Side Information

Clustering text data with side information is discussed here. Let us assume that we have a corpus *S* of text documents. The total number of documents is *N*, and they are denoted by *T*1 . . . *TN*. It is assumed that the set of distinct words in the entire corpus *S* is denoted by *W*. Associated with each document *Ti*, we have a set of side attributes *Xi*. Each set of side attributes *Xi* has *d* dimensions, which are denoted by *(xi*1 . . . *xid)*. We refer to such attributes as *auxiliary* attributes. For ease in notation and analysis, we assume that each side-attribute *xid* is binary, though both numerical and categorical attributes can easily be converted to this format in a fairly straightforward way. This is because the different values of the categorical attribute can be assumed to be separate binary attributes, whereas numerical data can be discretized to binary values with the use of attribute ranges. Some examples of such side-attributes are as follows:

• In a web log analysis application, we assume that *xir* corresponds to the 0-1 variable, which indicates whether or not the *i*th document has been accessed by the *r*th user. This information can be used in order to cluster the web pages in a site in a more informative way than techniques which is based purely on the content of the documents. As in the previous case, the number of pages in a site may be large, but the number of documents accessed by a particular user may be relatively small.

• In a network application, we assume that *xir* corresponds to the 0-1 variable corresponding to whether or not the *i*th document *Ti* has a hyperlink to the *r*th page *Tr*. If desired, it can be implicitly assumed that each page links to itself in order to maximize linkage-based connectivity effects during the clustering process. Since hyperlink graphs are large and sparse, it follows that the number of such auxiliary variables are high, but only a small fraction of them take on the value of 1.

• In a document application with associated GPS or provenance information, the possible attributes may be drawn on a large number of possibilities. Such attributes will naturally satisfy the sparsity property. As noted in the examples above, such auxiliary attributes are quite sparse in many real applications. This can be a challenge from an efficiency perspective, unless the sparsity is carefully taken into account during the clustering process. Therefore, our techniques will be designed to account for such sparsity. However, it is possible to easily design our approach for non-sparse attributes, by treating the attribute values in a more symmetric way.

We note that our technique is not restricted to binary auxiliary attributes, but can also be applied to attributes of other types. When the auxiliary attributes are of other types (quantitative or categorical), they can be converted

to binary attributes with the use of a simple transformation process. For example, numerical data can be discretized into binary attributes. Even in this case, the derived binary attributes are quite sparse especially when the numerical ranges are discretized into a large number of attributes. In the case of categorical data, we can define a binary attribute for each possible categorical value. In many cases, the number of such values may be quite large. Therefore, we will design our techniques under the implicit assumption that such attributes are quite sparse. The formulation for the problem of clustering with side information is as follows:

Text Clustering with Side Information:

Given a corpus S of documents denoted by T1 . . . TN, and a set of auxiliary variables Xi associated with document Ti, determine a clustering of the documents into k clusters which are denoted by C1 . . . Ck, based on both the text content and the auxiliary variables.

We will use the auxiliary information in order to provide additional insights, which can improve the quality of clustering. In many cases, such auxiliary information may be noisy, and may not have useful information for the clustering process. Therefore, we will design our approach in order to magnify the coherence between the text content and the side-information, when this is detected. In cases, in which the text content and side-information do not show coherent behavior for the clustering process, the effects of those portions of the side-information are marginalized.

B. The COATES Algorithm

In this section, algorithm for text clustering with side-information is discussed here. It is known as *COATES* algorithm throughout the paper, which corresponds to the fact that it is a *COntent and Auxiliary attribute based Text cluStering* algorithm. The number of clusters *k* are given as input to the algorithm. As in the case of all text-clustering algorithms, it is assumed that stop-words have been removed, and stemming has been performed in order to improve the discriminatory power of the attributes.

The algorithm requires two phases:

• **Initialization:** We use a lightweight initialization phase in which a standard text clustering approach is used without any side-information. For this purpose, we use the algorithm described in [27]. The reason that this algorithm is used, because it is a simple algorithm which can quickly and efficiently provide a reasonable initial starting point. The centroids and the partitioning created by the clusters formed in the first phase provide an initial starting point for the second phase. We note that the first phase is based on text only, and does not use the auxiliary information.

• **Main Phase:** The main phase of the algorithm is executed after the first 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. This phase performs alternating iterations which use the text content and auxiliary attribute information in order to improve the quality of the clustering. We call these iterations as *content* iterations and *auxiliary iterations* respectively. The combination of the two iterations is referred to as a *major iteration*. Each major iteration thus contains *two minor iterations*, corresponding to the auxiliary and text-based methods respectively.

In the first phase clustering process is done which is based on text content. In the second phase techniques for content and auxiliary information integration are provided the first phase is simply a direct application of the text clustering algorithm. The overall approach uses alternating minor iterations of content-based and auxiliary attribute-based clustering. These phases are referred to as *content-based* and *auxiliary attribute-based* iterations respectively. The algorithm maintains a set of seed centroids, which are subsequently refined in the different iterations. In each content-based phase, we assign a document to its closest seed centroid

based on a text similarity function. The centroids for the *k* clusters created during this phase are denoted by *L*1 . . . *Lk*. Specifically, the cosine similarity function is used for assignment purposes. In each auxiliary phase, we create a probabilistic model, which relates the attribute probabilities to the cluster-membership probabilities, based on the clusters which have already been created in the most recent text-based phase. The goal of this modeling is to examine the coherence of the text clustering with the side-information attributes. Before discussing the auxiliary iteration in more detail, we will first introduce some notations and definitions which help in explaining the clustering model for combining auxiliary and text variables. We assume that the *k* clusters associated with the data are denoted by *C*1 . . . *Ck*. In order to construct a probabilistic model of membership of the data points to clusters, we assume that each auxiliary iteration has a *prior* probability of assignment of documents to clusters (based on the execution of the algorithm so far), and a *posterior* probability of assignment of documents to clusters with the use of auxiliary variables in that iteration. We denote the prior probability that the document *Ti* belongs to the cluster *Cj* by $P(Ti \in Cj)$. Once the pure-text clustering phase has been executed, the *a-priori* cluster membership probabilities of the auxiliary attributes are generated with the use of the last content-based iteration from this phase.

IV. CONCLUSION

In this paper, methods are discussed for mining text data with making use of side information. Side information may be presented in many forms of text database which are used to enhance the clustering process. Iterative portioning technique is combined with a estimation process to design the clustering method which gives the importance of different kinds of side information. This general method is used to design both clustering and classification algorithms.

REFERENCES

[1] C. C. Aggarwal and P. S. Yu, "A framework for clustering massive text and categorical data streams," in *Proc. SIAM Conf. Data Mining*, 2006, pp. 477–481.

[2] 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.

[3] M. Steinbach, G. Karypis, and V. Kumar, "A comparison of document clustering techniques," in *Proc. Text Mining Workshop KDD,*2000, pp. 109–110.

[4] 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.

[5]S.Zhong,"Efficientstreaming text clustering,"*Neurl Netw.*,vol. 18, no. 5–6, pp. 790–798, 2005.