A Private Cloud Document Management System with Document Clustering Algorithm

Jiajia Miao[1,2]
1) Institute of Command Automation, PLA University of Science and Technology, Nanjing, China
2) Key Laboratory of C4ISR Technology, National University of Defense Technology, Changsha, China
jjmiao@ieee.org

Zhongjun Fan
613#, School of Computer, National University of Defense Technology, Changsha, China
fang_zhongjun@163.com

Guoyou Chen
Institute of Command Automation, PLA University of Science and Technology, Nanjing, China
goyalchen@163.com

Handong Mao
Key Laboratory of C4ISR Technology, National University of Defense Technology, Changsha, China
handmao2005@126.com

Le Wang
Key Laboratory of C4ISR Technology, National University of Defense Technology, Changsha, China
cc_alan@163.com

***Abstract*—Recently, more and more enterprises use virtualization technology and cloud computing technology to improve the level of information management. Private cloud document management system from the lab to practical application. We launched a private cloud file management system is characterized by the automatic cluster of files, so as to achieve the automated management of the text block. Document clustering has been extensively studied, because it is an effective solution, the organization of a large number of files. In order to overcome the main challenges that the current document clustering a huge number of documents, high dimensional process and comprehensible cluster, we propose a hybrid algorithm based on the top-k frequent itemsets and K-Means. The experimental results show the efficiency and effectiveness of the algorithm is superior to the other two representative clustering algorithm on two public data sets. Our algorithm can be further improved in the future parallel implementation, based on semantic representation and similarity measurement.**

Keywords- private cloud; document management; document clustering; frequent term sets

I. Introduction

For any organization, extending information access can be a challenge. Yet it’s often the key to increasing profitability, maximizing productivity and reducing operating costs. PCDM is a powerful content repository designed to help achieve that goal and give access to content quickly, effectively and securely. It combines sophisticated yet intuitive document management with flexible and accessible collaboration to accelerate the flow of information throughout your organization.

Designed for departments and small to mid-sized businesses PCDM helps you capture, manage, distribute and work with documents in diverse working environments, whether you have 10 or 500 employees. With PCDM, you start with a feature-rich system that perfectly meets your needs.

The document clustering is an effective solution for organizing a large number of pages, such as the grouping of a large number of pages returned by the search engine, the natural file classification and browsing [3] [4]. In addition, the cluster analysis method can also be used in other text mining applications [5]. The theme document clustering has been extensively studied in the literature [6], in text processing, and the most commonly used method is the so-called vector space model, the model file is converted into a high-dimensional vector size. The major challenge in document clustering is included in the following three areas [7]:

- Very high-dimensional data (~ 10,000): It requires the ability to deal with the sparse data space dimensionality reduction method.
- Very large size of the database (in particular the World Wide Web): Therefore, the clustering algorithm is very effective, scalable large-scale database.
- Understandable of the cluster description: The description of the cluster must be understandable; clustering can be used to guide the user to browse.

K-means [9] (or its variants) is a good choice, because it is effective to deal with these challenges, better than other well-known quality and simple clustering algorithm [10]. However, long-term deal with k-means the complete frequency vector high-dimensional vector space, and can not find any understandable description of cluster. To make matters worse, the selection of the initial partition of the k-means is very sensitive. So, using the k-means still need optimal initial partition of the (local) to get better efficiency, large capacity corpus.

Often long text clustering [7] [8] In order to deal with these challenges, is another preferred method, because the description of it is understandable cluster cluster network files and significantly reduce the dimensionality of data. The frequent term basis the efficiency of the method is excellent,

This paper is supported by the Natural Science Foundation of Jiangsu Province BK2010131, and Foundation of PLA University of Science and Technology 20110208.

even a very large database. However, this method is not better than the other algorithms (variants of the k-means) clustering quality [1].

This is a natural idea hybrid clustering method for the synthesis of these two algorithms. The hybrid approach first frequent term-based method to generate initial cluster, and easy-to-understand description, and then use the k-means to improve the preliminary results and optimal clustering. The MFI the K-means [1] According to this idea is a good example.

MFI K-means maximum frequent itemsets generated k-means initial partition. If a person is not a superset of the frequent itemsets is frequent, it is defined as a maximum frequent itemsets. Divided into an initial partition all the files, one of the largest frequent item sets all the terms. Longer file (short) more opportunities, the common maximal frequent itemsets frequent term of office as the initial partition like large files, leading to slow the combined effect of unsteadiness With the file length. To resolve this issue, the MFI had to cut the K-means file more than 300 words. Therefore, the MFI of the K-means method [1] has encountered an obstacle, with the processing of the various lengths of the Web document.

Inspired by the concept of integration, we propose a top-k frequent terms and the k-means algorithm based on a simple hybrid document clustering algorithm (SHDC). SHDC to eliminate restrictions on the length of the file, and to simplify the process to generate the initial partition. The experiment conducted in a real public data sets show that, the SHDC has outperformed effectiveness and efficiency compared to the K-means in the same baseline of MFI.

II. PCDM Architecture

Private Cloud Server: The foundation of this document management solution provides a central location to manage multiple types of content, including scanned documents, electronic documents, e-mail, CAD files and multimedia audio and video files.

Private Cloud Client: An easy to use tool to give users access to the content repository. The client can be used to perform all indexing, searching, distribution and editing actions. It also provides a document preview when browsing search results.

Web Client: Remote access is easy with the web client interface. It is useful for remote offices, travelling employees as well as a customer and vendor portal. This client provides most of the document management features available in the rich client.

III. The Hybrid algorithm

The top-k frequent itemsets of the SHDC algorithms found the use parTFI algorithm [2]. Documentation set, which includes all aspects of a common set of terms, be regarded as a cluster candidates and their average value as the initial k-means cluster seed polymerization similar file. The final optimal clustering by the k-means on the basis of, on the basis of the set of terms of the top-k frequent to generating initial cluster returns.

This usually refers to the following terms.

- The coverage of a term set F, denoted by $cov(F)$, is a set of documents containing all terms of F. The document D supports F if $D \in cov(S)$.
- The coverage of each frequent term set F can be regarded as a cluster (candidate) C. The cluster description of C is defined as the content of F.
- A cluster mean is a vector whose elements are generated respectively by the average weight of corresponding dimensions of other points in this cluster.

A. Finding top-k frequent term sets

The top of a parallel k frequent itemsets mining algorithm, called parTFI find the top-k set of commonly used terms. parTFI introduced in our previous work [2]. Of top-k frequent set of terms are defined as follows:

Found in the process of top-k frequent itemsets adaptively selected file distribution between the probability of frequent itemsets. In addition, there are limitations frequent itemsets, frequent itemsets of length less than the length of the threshold will not be included in the final results. Therefore, this approach does not take into account the term set with a very high support, but rarely. In this way, a very short document will not disturb the aggregation effect.

The final results, parTFI algorithm to generate k frequent itemsets, which have the greatest support, that is, the maximum number of files, which contain these Terms. In other words, k frequent itemsets and picture coverage main topics of the data set. Clustering results will not be subject to interference with long file, length frequent itemsets [1] Compare excessive pursuit algorithm, it avoids the shortcomings tend to be longer document.

B. Producing Initial Clusters

SHDC found that the top-k frequent itemsets use parTFI. Then, for each of the top-k frequent itemsets, all the files that contain the term set is divided into an initial cluster (candidate). After that, SHDC calculated k initial clusters (candidate), and an average of each of the device is used for all documents into k clusters (true initial clusters) by seed polymerization. Finally, the return on the basis of more refined k-means clustering initial cluster.

Need of special note here. Initial cluster k (candidate), respectively corresponding to the k top-k frequent itemsets coverage, and may not include all of the files in the database, you can overlay. They only means of k-means is used to calculate the initial requirements, covering all the files and do not overlap is not mandatory. Real initial clustering, which is polymerized by k seeds truly meet the above rules. This simplification, avoiding overlap eliminate time-consuming process [7], [8] and pretreatment exit simple, rapid large corpus, this is very important.

Ref [2] most commercial database systems to provide full-text indexing, Oracle Text [11], which is the top-k frequent itemsets parTFI the excavation. Oracle text, it is a valid file, which contains the specified words in a very large database using the following SQL statement:

SELECT document FROM table

WHERE CONTAINS (document, ‘Dog and Cat’) > 0;

So it is easy to get all documents which contain some frequent term set by Oracle Text. For l-th initial cluster (candidate), the j-th dimension $t_j^{(l)}$ of mean can be calculated according to formula (1).

$$t_j^{(l)} = 1/n_l \cdot \sum_{d_i \in \mathrm{cov}(F_l)} t_{ij} \qquad (1)$$

Where $cov(F_l)$ denotes the coverage of frequent term set F_l, t_{ij} the j-th dimension of d_i which supports F_l, and n_l the cardinality of $cov(F_l)$. All vectors of documents have been preprocessed according TF-IDF.

The document in database is grouped into l-th cluster if the similarity between this document and l-th mean $m^{(l)}$ is the biggest one among all clusters (or candidates). The similarity is defined as formula (2).

$$sim(d, m^{(l)}) = \sum \frac{d \cdot m^{(l)}}{|d| \cdot |m^{(l)}|} \qquad (2)$$

C. Refining the Clustering

Will perform the refining process of the K-means, which is generated according to the method described in the above sections of the initial clustering. The k-means iterations, until there are no changes in the file dependent relationship. Return the best k-means clustering.

The SHDC official described as follows. We assume that m n terms in files and databases. For the SHDC algorithm in the following summary:

- K frequent item sets, which we used in the production of the initial k-means clustering. This method eliminates the prefer a longer document, the disadvantage exists in [1].
- The method, which generates the initial k-means partition, avoiding process on the high-dimensional space. In addition, the method is simple enough, and therefore, its efficiency is high. The efficiency is a very important criterion value of the algorithm, bulky corpus. In a subsequent experiment to verify the efficiency of the algorithm.
- Clustering results is effective k-means algorithm top-k frequent itemsets provides understandable description of k clusters.

IV. Experimental Evaluations

L. Zhuang, MFI verify the validity of the K-means [1] by comparing two representative k-means algorithm (furthest k-means and random initial k-means). MFI K-means code we can not get the evaluation and impartial consideration, we do not develop the source code of the self-advocates. We make the K-means of our method and MFI indirect comparison by comparison SHDC with [1] using the same two representative algorithm.

A. Evaluation Criteria

Our use of two cluster validation methods that outline coefficient (SC) [12] and normalized mutual information (NMI) [13], the public text data sets to evaluate the performance of clustering. These two authentication methods are independent of the number of clusters K.

Suppose that there are m documents $D = \{d_1, d_2, \cdots, d_m\}$, which belong to l classes with no overlap. $\overline{C} = \{\overline{C_1}, \overline{C_2}, \cdots, \overline{C_k}\}$ defines a clustering result.

B. Text Datasets

The main idea behind the experiment is the comparison between the standard text corpus and manually predefined classification has been the subject of clustering results. The predefined classification only exist in some of the text corpus.

The following two data sets are used: (1) Reuters -21,578 corpus, which is common for text clustering. It includes a prior document classification, and its wide range of realistic network file. (2) 20 newsgroups data, which is a collection of 20,000 messages from 20 different Usenet newsgroups, 1000 messages from each collection.

Reuters -21,578, we only select a theme, and have done a Reuters article, is the only designated. In addition, category, there are less than three document file will be discarded. Therefore, we production Reuters -2157812000 file is divided into 82 categories, including a new version. Empty file is deleted, NG20 datasets, there are a total of 19,949 documents and 43,586 terms.

C. Experimental Setting

The HP the unit 4 the Itanium II1.4G processor and 48 GB of memory as the hardware platform. Bow kit mentioned above pre-processing of the original data set, delete the title, stopped, and the three files in this sentence is less than or greater than the average number of documents in each category. Then, we create two data sets, data set is loaded into the Oracle Text create a full-text index of all the terms of the data set. The definition of the structure of the table, such as Table 1:

D. Experimental Results

For all these three algorithms which are evaluated in the experiments, we use a maximum number of iterations of 10 (to make a fair comparison). Each experiment runs ten times, and at each time, we start from a different random initialization. The averages and standard deviations of the NMI are reported in tables. The SC and running time results are shown by figures. We set different varying ranges of k for different datasets in order to consider the real number of categories in them. The min_length of parTFI is set to 10 for all the following listed results.

NG20 and Reuters -21 578 different categories. From the results, we can see that the SHDC better benefits than other variants of k-means, k-means and random initial partition of the poorest, which will lead to sensitive k-means initial partition. The k-means to provide the best conditions, can achieve better results than those of random initial partition.

In order to study the effect of k, we use different K experimental NG20 and RTR -21,578. If in the space limitations, only the main point is plotted in Figure 2. NMI, the best result is close to the true number of classes, respectively, is similar to the results of the two sets of data.

Figure 1 irradiation, the average performance for the running time of 10 iterations. We show only the main points on two data sets k.

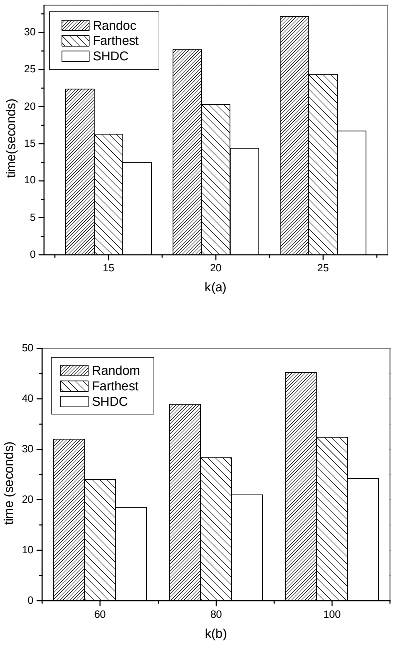

Figure 1. Comparing the average time performance on NG20(a) and Reuters-21578(b)

We can see that SHDC can accommodate a variety of clusters. Its running time a little more, when faced with the rise of the cluster. This characteristic facilitates the handling of the large capacity scalability corpus. In addition, the heuristic initial cluster, SHDC consume less running time than the other two algorithms. K-means furthest better than other algorithms from random initial partition. This result can be explained as follows: the more optimal initial partition fewer files are re-assigned to the cluster, wherein the retrenches of running time to re-calculate the similarities between the device and file cluster means and unchanged.

Besides, we have studied the effect of min_length for clustering effectiveness and efficiency. We found that the clustering quality is better when min_length is about 10. The quality would decline when the value of min_length beyond or below 10. From current first step analysis, the reason is the impact of min_length over parTFI. Furthermore, this impact would also interface the running time of SHDC.

REFERENCES

[1] Ling Zhuang, Honghua Dai.: A Maximal Frequent Itemset Approach for Web Document Clustering. Proceedings of the Fourth International Conference on Computer and Information Technology (CIT'04)

[2] Wang Yongheng, Jia Yan and Yang Shuqiang.: Parallel Mining of Top-K Frequent Items in Very Large Text Database. WAIM (2005)

[3] Oren Zamir, Oren Etzioni.: Web document clustering: A feasibility demonstration. In Melbourne, Australia Proceedings of SIGIR' 98.

[4] Oren Zamir, Oren Etzioni 99 Grouper.: A Dynamic Clustering Interface to Web SearchResults. In Proceedings of the 8th WWW Conference, Toronto Canada, 1999

[5] Jiawei Han, Micheline Kamber.: Data Mining: Concepts and Techniques, Second Edition.: Morgan Kaufmann Press, 2006..

[6] Andreas Hotho, Andreas Nürnberger, and Gerhard Paaß.: A Brief Survey of Text Mining. LDV FORUM – Vol. 20 – 2005, p.19-62

[7] Beil F., Ester M., Xu X.: Frequent Term-Based Text Clustering, Proc. 8th Int. Conf. on Knowledge Discovery and Data Mining (KDD '2002), Edmonton, Alberta, Canada, 2002.

[8] Benjamin C. M. Fung, Ke Wang, Martin Ester.: Hierarchical Document Clustering using Frequent Itemsets. SDM 2003.

[9] J.B. MacQueen. Some methods for classification and analysis of multivariate observations. In Proceeding of the 5th Berkeley symposium in mathematics and probability, 1967.

[10] Steinbach M., Karypis G., Kumar V.: A Comparison of Document Clustering Techniques, Proc. TextMining Workshop, KDD 2000, 2000.

[11] Oracle Text technical white paper. Oracle Corporation, January 2004.

[12] Andreas Hotho, Alexander Maedche, Steffen Staab.: Ontology-based Text Document Clustering. KI 16(4) (2002) 48-54

[13] Strehl, A., & Ghosh, J. (2002). Cluster ensembles - a knowledge reuse framework for combining partitions. Journal of Machine Learning Research, 3, 583–617.