International Journal of Recent Engineering Research and Development (IJRERD)
ISSN: 2455-8761
www.ijrerd.com || Volume 02 – Issue 12 || December 2017 || PP. 21-25

Research on Advanced Approach of Clustering Using Cuda Architecture

Divya Khurana
M.Tech Scholar
Department of computer science
K.I.E.T, Ghaziabad, Uttar pradesh, India

Abstract: This project is based on the concept of data mining. Data mining is extraction of knowledge a large amount of data. We have used hierarchical clustering of data to reduce the space and time complexity of the search process. The entire project is divided into three modules i.e. pre-processing, clustering using BIRCH and parallelization using CUDA.
Keywords: Data Mining, CUDA, BIRCH.

1. Introduction

Data mining refers to extraction or "mining" knowledge from large amount of data. Its a relatively new penalizing field of computer science with a great potential to help organizations focus on the most important information in their vast data warehouses. It automates tasks like storing large amounts of data and to process it as well as produce information useful to businesses to take decisions efficiently. When performed on text databases, it is termed as text mining; Data mining commonly involves four classes of tasks:

1. Association rule learning(market basket analysis) – Searches relationships between variables
2. Clustering – Discovers groups and structures in the data that are in some way or another "similar", without using known structures in the data.
3. Classification – Generalizes known structure to apply to new data.
4. Regression – Attempts to find a function which models the data with the least error.

The main focus here is clustering. Cluster analysis or clustering is the assignment of a set of observations into subsets (*clusters*). The observations in the same cluster are similar in some sense and different from the observations in other clusters.
Clustering is also called data segmentation in some applications because clustering partitions large data sets into groups according to their *similarity*. The point to note is that clustering only forms a group of similar objects without saying why. For very large datasets clustering helps to form relatively small number of groups of objects in the dataset, properties of which could be further studied to find what makes objects belonging to same cluster similar and different from objects in other clusters.

2. Related Work

Here is a brief overview of all previous studies and research done on the subject. Early work started in 1971, but that is less relevant to our work. So talking about only the recent advancements in data mining and data clustering technology; Most of the work around the year 2000 was focused on efficiency aspects for online tasks rather than the effectiveness of method.

In 2005, Cui and Potok proposed a PSO based hybrid document clustering algorithm. The algorithm performs a search globally in the entire solution space. They applied the hybrid PSO + K-means clustering algorithm on the four different text document datasets in their experiments. The results illustrated those more compact clustering results were obtained using hybrid PSO+K means than by any method alone.

Jing et al. in 2007 presented a novel K-means type method, an algorithm for clustering high dimensional data objects like text in sub-spaces. The algorithm calculates weight for each dimension in each cluster and uses the weight values to identify the subsets of important dimensions that categorize different clusters.

In 2008, Sun et al. developed a novel hierarchical algorithm for document clustering. They used the cluster overlapping phenomenon to design cluster merging criteria.

Muflikhah and Baharudin in 2009 proposed a method that integrates the information retrieval method and document clustering as concept space approach.

Also based on the hierarchical clustering method it used the EM (expectation maximization) algorithm in the Gaussian mixture model to count the parameters and make the two sub-clusters combined when their overlap is the largest.

2.1. Research Gaps

Based on the study following research gaps were identified - Clustering algorithms assume all data can be accommodated in the memory at the same time, except in large datasets. The algorithm used for clustering of text should be efficient in terms of memory usage, which the general algorithms are not; use of summary data structures solves this.

The need is to have a hierarchical clustering algorithm which can form clusters at various levels of hierarchy and giving complete information as to which cluster does the document belong to at each level of hierarchy.

The limitation of hierarchical clustering algorithms is that their time complexity is very high and require large amount of memory to store the hierarchy tree after each step.The requirement is to have an algorithm which uses some form of summaries to store data in a compact space.

2.2. Statement of the Problem

The problem statement for the proposed research work is as follows:

"To perform parallelized, hierarchical clustering of unstructured text documents by using tree based compression."

The problem can be divided into the following sub-problems:

- To pre-process unstructured text documents to convert them into numeric vectors.
- To use tree based indexing, forming a hierarchical summarization of data in the main memory.
- To perform hierarchical clustering using the summary of data stored in the main memory.
- To parallelize the computations on similarity measurement to produce speed up.

3. Framework for Clustering Of Unstructured Text

The proposed work as shown in the basic framework is divided into 3 modules.

Module 1 is the preprocessor for Text taking input as unstructured text format and returning document in vector form using Vector space model combined with tf-idf weighing scheme.

Module 2 We implement hierarchical clustering algorithm. used for categorization of text.

Module 3 We parallelize our proposed algorithm on parallel computing GPUs using CUDA

MODULE 1: Text Preprocessor

It deals with cleaning of the data and preparing it for use in the algorithm. Preprocessing being a common task, several codes for it are available. But instead of using a prebuilt preprocessor we propose a simple yet effective idea to build a pre-processor for texts.

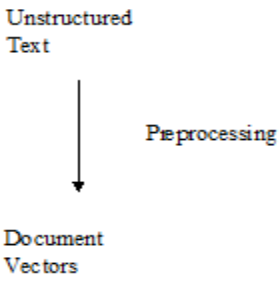

Figure1: Preprocessing.

- Starting with the basic step of tokenization the document can be read line by line, broken into tokens using punctuations and space as break points.
- Once the document is converted into a bag of words we check each word for its existence in the stop word list used by the famous SMART information retrieval system. If the word exists in the stop word list we remove it from the corpus vocabulary.
- Once the documents have been tokenized and stop words have been removed, we stem the words to their root form.
- For our work we propose to use the Porter's Stemmer, originally written by M.F. Porter as a part of larger IR project and later published as "A Program for Suffix Stripping".
- The Preprocessor stores the documents in form of a list of stemmed words as temporary files.

- These files are read by the tf-idf sub module code which stores in a dictionary form the work and their count for each document. At the same time any words which occur in a documents are a few minutes of time.
- When all the documents have been read we get a vocabulary list which represents the number of distinct words in the corpus, the number of distinct words represents the dimensionality of the vector space we will use to represent each document.

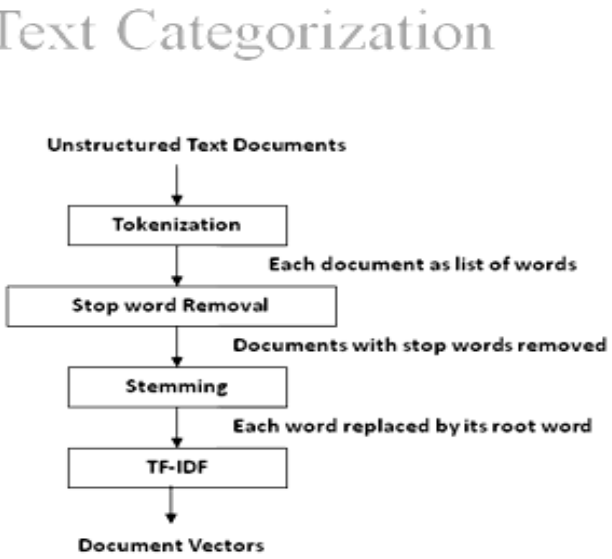

Figure2: Text Categorization Process

Tokenization: The first and foremost step in the text mining process is to get tokens in document. That is to break the document into smallest units useful for the algorithm or task to be performed. Generally tokens are relevant words because irrelevant words are removed in the step described below.

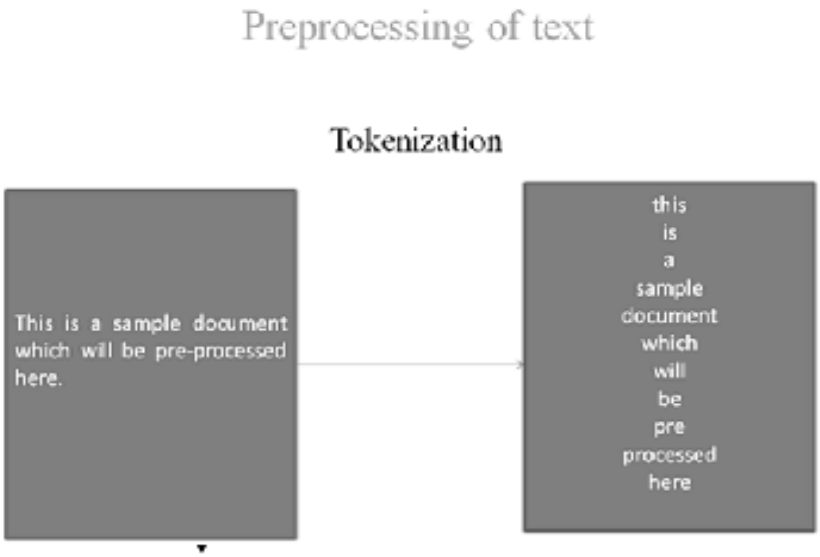

Figure3: Tokenization

Removal of Stop Words: Some common words such as prepositions or conjunctions etc appear in each document many no. of times. These terms have no discriminative power and increase the dimensionality of the Vector Space which in turn results in poorer accuracy and higher complexity of algorithm. Such terms are called Stop Words and they are removed from the document during the text preprocessing phase.

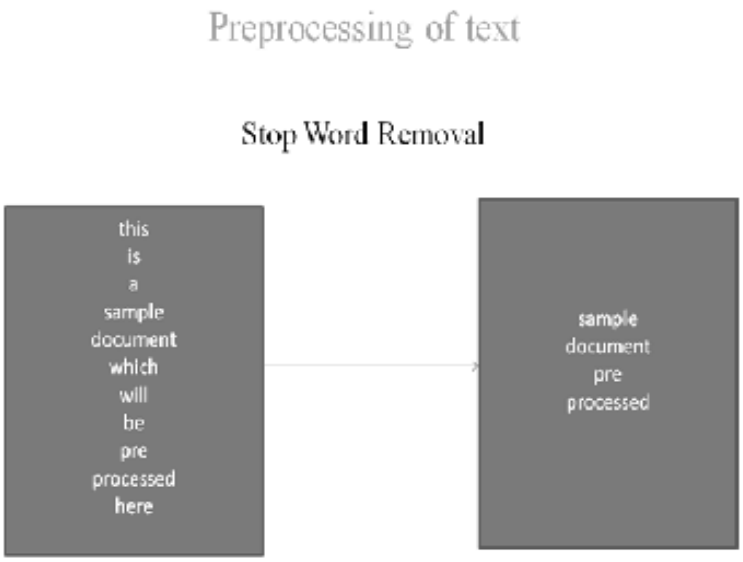

Figure 4: Stop Word Removal

Stemming: A single word can exist in various forms in document corpus but with the same meaning, that is various terms may share a common word stem. A text retrieval system needs to identify the group of words where each word is a small syntactic variant of the root. Like drug, drugged, drugs has the same root drug and can be viewed as variants of the root word drug, So all three forms should be changed to the root word. The process is called stemming.

4. Result

1. In our work we have proposed the use of a hierarchical clustering algorithm for unstructured text which uses a hybrid approach of hierarchical clustering and iterative partitioning based clustering methods. The proposed scheme produces hierarchical clusters thus finding similar document groups at various levels of hierarchy.
2. Our scheme uses a tree based compression data structure to store summaries of clusters in place of complete data. The tree thus formed is stored in memory at all points of time so the addition of new item does not require the program to read again any of the previous data values from disk. Thus even with a constraint on main memory we can work with very large sized datasets.
3. In the present work we have parallelized the computations on similarity measurement.

These calculations are performed on large sized numeric vectors, with the help of CUDA supportive GPUs these computations are performed much more quickly as compared to serial implementation. Thus the parallelization produces a significant speed up in Our solution is better in terms of both time and space complexity compared to the current methods used to cluster text. Thus our method can find use in various applications in many basic document organizing tasks like hierarchical organization of news articles, as a search engine add on for more organized results, and as a simple offline document clustering tool.

5. Future Scope

Our Method works well for data like News articles with differentiating classes, but in the presence of misc articles the accuracy is decreased. For such Fuzzy classes the method could either be combined with some other clustering method at macro clustering in place of simply finding nearest cluster.

Also there is a need to address multi class problem that is a document might belong to multiple classes at the same level of hierarchy, in our method. it is simply grouped with the most near match. But in actual there must be some threshold limits to allow documents to belong to multiple clusters.

Also we have proposed to use the normal Distance Metric for Similarity which effects the result when the articles of vastly varying lengths. The Idea of CF tree for compression could be taken a step forward to calculate summaries on the basis of vector representation of document itself.

6. Conclusions

Thus this paper is an attempt to throw light on the CUDA architecture and the BIRCH algorithm. The troubles related to unstructured text clustering with respect to time and space complexity are discussed .It can help the user to work on unstructured text or documents in the organizations of websites etc. I hope that readers will be able to get overview of parallelized clustering of unstructured data.

References:

[1]. Csorba, K.; Vajk, I.; "*Term Clustering and Confidence Measurement in document Clustering*," Computational Cybernetics, 2006. ICCC 2006. IEEE International Conference on , vol., no., pp.1-6, 20-22 Aug. 2006.Douglas E. Comer ,"*Computer Networks & Internet* ", Pearson Education Asia , 5th Edition ,2008.

[2]. NVIDIA Corporation, *NVIDIA CUDA: Compute unified device architecture*,Programming guide 2007, pp. 1–83.

[3]. J. Han and M. Kamber, *DATA MINING CONCEPTS AND TECHNIQUES*, MorganKaufmann, 2006.

[4]. Cui, X.; Potok, T.E.; Palathingal, P.; , "*Document clustering using particle swarm optimization*," Swarm Intelligence Symposium, 2005. SIS 2005. Proceedings 2005 IEEE,vol. no. , pp. 185- 191, 8-10 June 2005.

[5]. H. Sun, Z. Liu, and L. Kong, "*A Document Clustering Method Based on Hierarchical Algorithm with Model Clustering*", in Proc. AINA Workshops, 2008, pp.1229-1233.Douglas E.Comer, *Internetworking with TCP/IP*, Volume 1, 2, Forth Edition Pearson Education Asia 2006.

[6]. Na Wang; Pengyuan Wang; Baowei Zhang; , "*An improved TF-IDF weights function based on information theory*," Computer and Communication Technologies in Agriculture Engineering (CCTAE), 2010 International Conference On , vol.3, no., pp.439-441, 12-13 June 2010.

[7]. Tian Zhang, Raghu Ramakrishnan, Miron Livny; "*BIRCH: an efficient data clustering method for very large databases*", In SIGMOD '96: Proceedings of the 1996 ACM SIGMOD international conference on Management of data, pp. 103-114, 1996.

[8]. Na Wang; Pengyuan Wang; Baowei Zhang; , "*An improved TF-IDF weights function based on information theory*," Computer and Communication Technologies in Agriculture Engineering (CCTAE), 2010 International Conference On , vol.3, no., pp.439-441, 12-13 June 2010.

[9]. Lee, D.L.; Huei Chuang; Seamons, K.; , "*Document ranking and the vector-space model*," Software, IEEE , vol.14, no.2, pp.67-75, Mar/Apr 1997.

[10]. Cui, X.; Potok, T.E.; Palathingal, P.; , "*Document clustering using particle swarm optimization*," Swarm Intelligence Symposium, 2005. SIS 2005. Proceedings 2005 IEEE,vol. no. , pp. 185- 191, 8-10 June 2005

[11]. H. Sun, Z. Liu, and L. Kong, "*A Document Clustering Method Based on Hierarchical Algorithm with Model Clustering"*, in Proc. AINA Workshops, 2008, pp.1229-1233.

[12]. Liping Jing, Michael K. Ng, JoshuaZhexue Huang, "*An Entropy Weighting k-Means Algorithm Sparse Data*,"IEEE Transactions on Knowledge and Data Engineering, vol. 19, no. 8, pp. 1026-1041, June2007

[13]. Muflikhah, L.; Baharudin, B.; ,"s," Computer Technology and Development, 2009. ICCTD '09. International Conference on , vol.1, no., pp.58-62, 13-15 Nov. 2009.

[14]. Porter, M.F.: An algorithm for suffix stripping. Program, Vol. 14, No. 3, 1980.