HUFFMAN CODING WITH COMPRESSED BIG DATA IN CLOUD

[1]J.Noorul bimani, [2]T.Sheik yousuf
[1]PG Scholar, Department of Computer Science and Engineering, Mohamed Sathak Engineering College,
Ramanathapuram, India
[2]Assistant professor, Department of Computer Science and Engineering, Mohamed Sathak Engineering College,
Ramanathapuram, India.
[1]noorulbimani@gmail.com

Abstract: The data sets with multiple autonomous resources are popular in now a day. The big data plays a vital role in all the science and engineering departments such as physical, biological, biomedical science. The main goal of this project is to reduce the storage space of the cloud when storing big data into the cloud server. In this proposed system If uses the Huffman coding to effectively store the big data in the cloud server. The Huffman encoding algorithm is an optimal compression algorithm. Huffman encoding is an algorithm for the lossless compression of files based on the frequency of occurrence of a symbol in the file that is being compressed. In this proposed system the data with larger amount is compressed and it becomes a big data which is stored in a cloud server which reduces the data storage. The data is compressed and stored into the big data into the server. The proposed system improves the scalability at the end of result. That means a process to handle a growing amount of work in a capable manner.

Keywords: (Big Data, data mining, heterogeneity, autonomous sources, complex and evolving associations)

1. INTRODUCTION

Big Data concern large-volume, complex, growing data sets with multiple, autonomous sources. With the fast development of networking, data storage, and the data collection capacity, Big Data are now rapidly expanding in all science and engineering domains, including physical, biological and biomedical sciences. This paper presents a HACE theorem that characterizes the features of the Big Data revolution, and proposes a Big Data processing model, from the data mining perspective. This data-driven model involves demand-driven aggregation of information sources, mining and analysis, user interest modeling, and security and privacy considerations. We analyze the challenging issues in the data-driven model and also in the Big Data revolution.

The term 'Big Data' appeared for first time in 1998 in a Silicon Graphics (SGI) slide deck by John Mashey with the title of "Big Data and the Next Wave of Infra Stress" . Big Data mining was very relevant from the beginning, as the first book mentioning 'Big Data' is a data mining book that appeared also in 1998 by Weiss and Indrukya. However, the first academic paper with the words 'Big Data' in the title appeared a bit later in m 2000 in a paper by Diebold. The origin of the term 'Big Data' is due to the fact that we are creating a huge amount of data every day. Usama Fayyad in his invited talk at the KDD BigMine'12Workshop presented amazing data numbers about internet usage, among them the following: each day Google has more than 1 billion queries per day, Twitter has more than 250 milion tweets per day, Facebook has more than 800 million updates per day, and YouTube has more than 4 billion views per day. The data produced nowadays is estimated in the order of zettabytes, and it is growing around 40% every year. A new large source of data is going to be generated from mobile devices, and big companies as Google, Apple, Facebook, Yahoo, Twitter are starting to look carefully to this data to find useful patterns to improve user experience. Alex 'Sandy' Pentland in his 'Human Dynamics Laboratory' at MIT, is doing research in finding patterns in mobile data about what users do, and not in what people says they do .

HACE Theorem. Big Data starts with large-volume, heterogeneous, autonomous sources with distributed and decentralized control, and seeks to explore complex and evolving relationships among data. These characteristics make it an extreme challenge for discovering useful knowledge from the Big Data. In a native sense, we can imagine that a number of blind men are trying to size up a giant elephant (see Fig. 1), which will be the Big Data in this context. The goal of each blind man is to draw a picture (or conclusion) of the elephant according to the part of information he collects during the process. Because each person's view is limited to his local region, it is not surprising that the blind men will each conclude independently that the elephant “feels” like a rope, a hose, or a wall, depending on the region each of them is limited to. To make the problem even more complicated, let us assume that 1) the elephant is growing rapidly and its pose changes constantly, and 2) each blind man may have his own (possible unreliable and inaccurate) information sources that tell him about biased knowledge about the elephant (e.g., one blind man may exchange his feeling about the elephant with another blind man, where the exchanged knowledge is inherently biased). Exploring the Big Data in this scenario is equivalent to aggregating heterogeneous information from different sources (blind men) to help draw a best possible picture to reveal the genuine gesture of the elephant in a real-time fashion. Indeed, this task is not as simple as asking each blind man to describe his feelings about the elephant and then getting an expert to draw one single picture with a combined view, concerning that each individual may speak a different language (heterogeneous and diverse information sources) and they may even have privacy concerns about the messages they deliberate in the information exchange process.

Autonomous data sources with distributed and decentralized controls are a main characteristic of Big Data applications. Being autonomous, each data source is able to generate and collect information without involving (or relying on) any centralized control. This is similar to the World Wide Web (WWW) setting where each web server provides a certain amount of information and each server is able to fully function without necessarily relying on other servers. On the other hand, the enormous volumes of the data also make an application vulnerable to attacks or malfunctions, if the whole system has to rely on any centralized control unit. For major Big Data-related applications, such as Google, Flicker, Facebook, and Walmart, a large number of server farms are deployed all over the world to ensure nonstop services and quick responses for local markets. Such autonomous sources are not only the solutions of the technical designs, but also the results of the legislation and the regulation rules in different countries/ regions. For example, Asian markets of Walmart are inherently different from its North American markets in terms of seasonal promotions, top sell items, and customer behaviors. More specifically, the local government regulations also impact on the wholesale management process and result in restructured data representations and data warehouses for local markets.

While the volume of the Big Data increases, so do the complexity and the relationships underneath the data. In an early stage of data centralized information systems, the focus is on finding best feature values to represent each observation. This is similar to using a number of data fields, such as age, gender, income, education background, and so on, to characterize each individual. This type of sample feature representation inherently treats each individual as an independent entity without considering their social connections, which is one of the most important factors of the human society. Our friend circles may be formed based on the common hobbies or people are connected by biological relationships. Such social connections commonly exist not only in our daily activities, but also are very popular in cyber worlds. For example, major social network sites, such as Facebook or Twitter, are mainly characterized by social functions such as friend-connections and followers (in Twitter). The correlations between individuals inherently complicate the whole data representation and any reasoning process on the data. In the sample-feature representation, individuals are regarded similar if they share similar feature values, whereas in the sample-feature-relationship representation, two individuals can be linked together (through their social connections) even though they might share nothing in common in the feature domains at all. In a dynamic world, the features used to represent the individuals and the social ties used to represent our connections may also evolve with respect to temporal, spatial, and other factors. Such a complication is becoming part of the reality for Big Data applications, where the key is to take the complex (nonlinear, many-to-many) data relationships, along with the evolving changes, into consideration, to discover useful patterns from Big Data collections.

To show the usefulness of Big Data mining, we would like to mention the work that Global Pulse is doing using Big Data to improve life in developing countries. Global Pulse is a United Nations initiative, launched in 2009, that functions as an innovative lab, and that is based in mining Big Data for developing countries. They pursue a strategy that consists of 1) researching innovative methods and techniques for analyzing real-time digital data to detect early emerging vulnerabilities; 2) assembling free and open source technology toolkit for analyzing real-time data and sharing hypotheses; and 3) establishing an integrated, global network of Pulse Labs, to pilot the approach at country level. Global Pulse describe the main opportunities Big Data over's to developing countries in their White paper "Big Data for Development: Challenges & Opportunities": Early warning: develop fast response in time of crisis, detecting anomalies in the usage of digital media Real-time awareness: design programs and policies with a more ne-grained representation of reality Real-time feedback: check what policies and programs fails, monitoring it in real time, and using this feedback make the needed changes The Big Data mining revolution is not restricted to the industrialized world, as mobiles are spreading in developing countries as well. It is estimated than there are over billion mobile phones, and that 80% are located in developing countries.

Information sharing is an ultimate goal for all systems involving multiple parties. While the motivation for sharing is clear, a real-world concern is that Big Data applications are related to sensitive information, such as banking transactions and medical records. Simple data exchanges or transmissions do not resolve privacy concerns.

For example, knowing people's locations and their preferences, one can enable a variety of useful location-based services, but public disclosure of an individual's locations/movements over time can have serious consequences for privacy. To protect privacy, two common approaches are to 1) restrict access to the data, such as adding certification or access control to the data entries, so sensitive information is accessible by a limited group of users only, and 2) anonymize data fields such that Sensitive information cannot be pinpointed to an individual record. For the first approach, common challenges are to design secured certification or access control mechanisms, such that no sensitive information can be misconduct by unauthorized individuals. For data anonymization, the main objective is to inject randomness into the data to ensure a number of privacy goals. For example, the most common k-anonymity privacy measure is to ensure that each individual in the database must be indistinguishable from k _ 1 others. Common anonymization approaches are to use suppression, generalization, perturbation, and permutation to generate an altered version of the data, which is, in fact, some uncertain data.

One of the major benefits of the data ammonization-based information sharing approaches is that, once anonym zed, data can be freely shared across different parties without involving restrictive access controls. This naturally leads to another research area namely privacy preserving data mining , where multiple parties, each holding some sensitive data, are trying to achieve a common data mining goal without sharing any sensitive information inside the data. This privacy preserving mining goal, in practice, can be solved through two types of approaches including 1) using special communication protocols, such as Yao's protocol , to request the distributions of the whole data set, rather than requesting the actual values of each record, or 2) designing special data mining methods to derive knowledge from anonymized data (this is inherently similar to the uncertain data mining methods).

2. EXISTING SYSTEM

In this existing system the general purpose parallel program method used a weighted linear regression. It proposed a HACF (Heterogeneous Autonomous Couple x Evolving relationship) theorem. The heterogeneous was used the different collector which uses different protocols to manage system for recording. Each data is able to generate and collect information without involving any centralized control in autonomous. The value of big data was increased in the complex and evolving relationship. The existing system found the best feature from the entire feature present in the data.

The characteristics made it an extreme challenge for discovering useful knowledge from the Big Data. The existing work considered each individual as an independent entity without considering their social connections. That was one of the most important drawbacks of our existing system. The correlations between individuals inherently complicate the whole data representation and any reasoning process on the data. The most fundamental challenge for Big Data applications were to explore the large volumes of data and it requires the larger amount of storage space to store these big data It requires huge amount of storage space. It was complex and evolving in data and knowledge associations. The big data had a decentralized control.

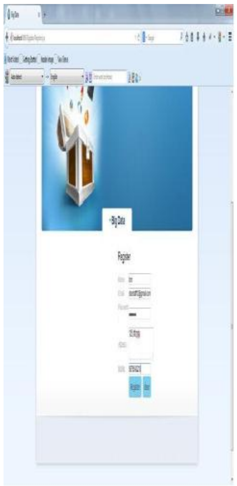
3. PROPOSED SYSTEM

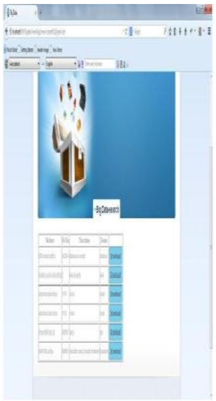
The proposed system the big data is compressed and stored it into a cloud server. In our proposed work we are going to use the Huffman coding algorithm. The Huffman coding is used as the compression technique in our proposed system. This compression technique which reduces the storage space of the big data in the cloud. The Huffman encoding is an lossless compression technique. In our proposed system there is no any data loss during the compression process.

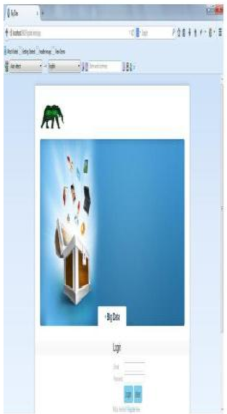
In our proposed system the data with larger amount is compressed and it becomes a big data which is stored in a cloud server which reduces the data storage. The data is compressed and stored into the big data into the server. Retrieval of stored files from the server we have to use k-means clustering algorithm. It provides more flexibility and scalability in the network. Our proposed system improves the scalability at the end of result. That means a process to handle a growing amount of work in a capable manner. Our proposed work reduces the storage space.Then the scalability of the network is also improved in our proposed work.The proposed work reduces the total cost

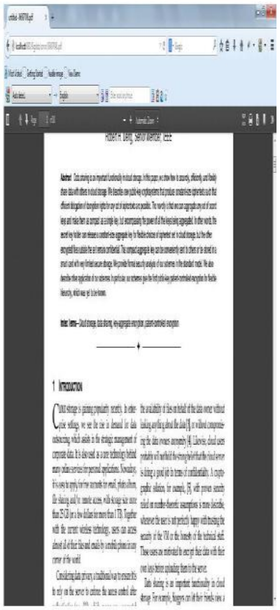
4. Result Analysis

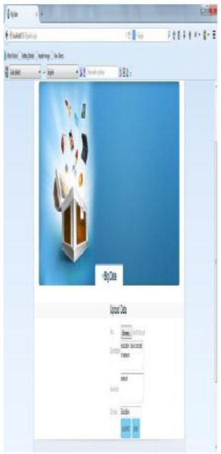
Storing of large amount of data in cloud can occupy the more memory space. It can be a cost effective process. The objective of this project is reducing the cost and efficiently using the cloud memory.In Compression technology reduces the file size and improves the storage capacity also reduce the cost. Big Data such as large amount of data could be compressed.Cloud computing is specially focused on data storage and data sharing in efficient, secure, flexible manner. So compressed data should be retrieve without losing of information. In big data cases, cloud server needs more memory space than a normal client system. The cloud server can be accessed by multiple users at a single time. So cloud must provide efficient access to each client without delaying. Memory management is one of the important features in cloud. So we design a new system for achieving reduce the cost and memory, improve the system performance by compress the stored data. Our new system provides retrieving this compressed data without loss.











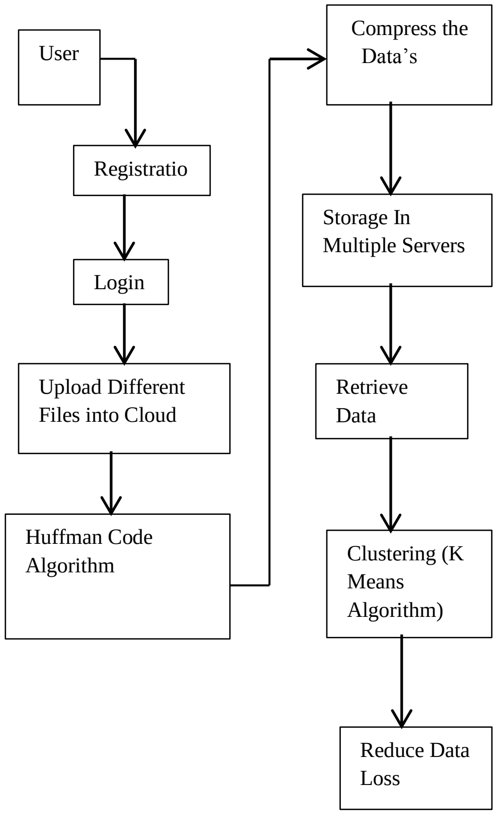
4.1 HUFFMAN CODE ALGORITHM

Huffman coding is a lossless data compression algorithm. The idea is to assign variable-length codes to input characters; lengths of the assigned codes are based on the frequencies of corresponding characters. The most frequent character gets the smallest code and the least frequent character gets the largest code. Huffman's scheme uses a table of frequency of occurrence for each symbol (or character) in the input. This table may be derived from the input itself or from data which is representative of the input. For instance, the frequency of occurrence of letters in normal English might be derived from processing a large number of text documents and then used for encoding all text documents. We then need to assign a variable-length bit string to each character that unambiguously represents that character. This means that the encoding for each character must have a unique prefix. The highest frequency letters - E and T - have two digit encodings, whereas all the others have three digit encodings. Encoding would be done with a lookup table. A divide-and-conquer approach might have us asking which characters should appear in the left and right sub trees and trying to build the tree from the top down. As with the optimal binary search tree, this will lead to an exponential time algorithm.

The decoding procedure is deceptively simple. Starting with the first bit in the stream, one then uses successive bits from the stream to determine whether to go left or right in the decoding tree. When we reach a leaf of the tree, we've decoded a character, so we place that character onto the (uncompressed) output stream. The next bit in the input stream is the first bit of the next character. If your system is continually dealing with data in which the symbols have similar frequencies of occurrence, then both encoders and decoders can use a standard encoding table/ decoding tree. However, even text data from various sources will have quite different characteristics.

4.2 K-MEANS CLUSTERING ALGORITHM

The proposed system we use k means algorithm for clustering the files fetched from the server. Here, the files are related to different domain. So retrieving files from different areas take too time for searching process according to the users queries. K-means clustering algorithm can group the related information retrieved from the server and it can be retrieved in an easy manner. This algorithm shall be provide an efficient way of retrieval process Figure


User
Registratio
Login
Upload Different Files into Cloud
Huffman Code Algorithm
Compress the Data's
Storage In Multiple Servers
Retrieve Data
Clustering (K Means Algorithm)
Reduce Data Loss

A DFD provides no information about the timing or ordering of processes, or about whether processes will operate in sequence or in parallel. It is therefore quite different from a flowchart, which shows the flow of control through an algorithm, allowing a reader to determine what operations will be performed, in what order, and under what circumstances, but not what kinds of data will be input to and output from the system, nor where the data will come from and go to, nor where the data will be stored (all of which are shown on a DFD).

5. CONCLUSION

In our proposed system we have to perform the bigdata applications. In those bigdata applications we have to use the concept of HACE. The purpose of that HACE is to choose multiple file in different are3a and also different domain. The process of big data applications contains map reduce. Here, we have to uploading the files into the cloud as in compressed manner. For compression process we have to use Huffman code compression technique. According to that compression technique the files has to e compressed. The reason for choosing this compression algorithm is to reduce the data loss at the time of compression. The compressed details are to be stored in multiple cloud servers. Next, we have to done the process of retrieval at the time of retrieval we can use the concept of map reduce. The retrieval process has to be done according to those srevers. Here for grouping the same domain files we can use k-means clustering algorithm. Using this k-means clustering algorithm we can group the retrieved files from the cloud. Finally, we have to retrieve the group of data from the cloud.

6. FUTURE ENHANCEMENT

In big data analysis is play vital role in all engineering, bio medical, science and other research area. Big data is collection of inter related data it is growing because the related information's are come from varies sources. In our proposed system, reduce the storage size by doing compression technique. The most fundamental challenge in big data application is to explore the large volume of data extract useful information for future action. We further improving the big data analysis in efficient manner by implementing the personalized concept. We know big data is a large volume of data. We just improve the retrieving of information from this big data based on user needs. One user can upload the video file in you tube, and document file in one domain, and another one type of file in various domain. By using personalized concept these all uploaded file comes under the single domain.

7. ACKNOWLEDGEMENT

Words are inadequate in offering my thanks to the almighty who has given strength and power to complete this Phase I project successfully.

I am highly indebted to Our Director Alhaj S.M.A.J. Habeeb Mohamed, Mohamed Sathak Engineering College, Kilakarai for his complete inspiration and motivation which helped me throughout the course of this project.

I feel deeply honored in expressing my sincere thanks to Our Principal Dr.J. Mohamed Jahabar B.E., M.E., Ph.D (IIT-Bombay), FIE., MISTE., MIEEE., MASME., Mohamed Sathak Engineering College, Kilakarai for making the resources available at right time and providing valuable insights leading to the successful completion of my work.I take this opportunity to express my profound gratitude and deep regards to Our Head of the Department Prof. R. Karthikeyan,M.E.,(Ph.D) Department of Computer Science and Engineering for his exemplary guidance, monitoring and constant encouragement throughout the course of this project. I am sincerely grateful to My Project Supervisor Prof.T.Sheik Yousuf, M.E(CSE)., M.Tech(IT)., (Ph.D.,), Department of Computer Science and Engineering for sharing his truthful and illuminating views on a number of issues related to project.I also extend my gratitude to My Project Coordinator Prof B. Rasina Begum,M.E.,MBA.,(Ph.D.,) Department of Computer Science and Engineering for her critical advice and guidance without which this project would not have been possible.

REFERENCES

[1] K. Su, H. Huang, X. Wu, and S. Zhang, "A Logical Framework for Identifying Quality Knowledge from Different Data Sources," Decision Support Systems, vol. 42, no. 3, pp. 1673-1683, 2006.

[2] "Twitter Blog, Dispatch from the Denver Debate," http:// blog.twitter.com/2012/10/dispatch-from-denver-debate.html, Oct. 2012.

[3] D. Wegener, M. Mock, D. Adranale, and S. Wrobel, "Toolkit-Based High-Performance Data Mining of Large Data on MapReduce Clusters," Proc. Int'l Conf. Data Mining Workshops (ICDMW '09), pp. 296-301, 2009.

[4] 5.. Wang, S.S.M. Chow, Q. Wang, K. Ren, and W. Lou, "Privacy-Preserving Public Auditing for Secure Cloud Storage" IEEE Trans.Computers, vol. 62, no. 2, pp. 362-375, Feb. 2013.

[5] X. Wu and X. Zhu, "Mining with Noise Knowledge: Error-Aware Data Mining," IEEE Trans. Systems, Man and Cybernetics, Part A, vol. 38, no. 4, pp. 917-932, July 2008.

[6] A.da Silva, R. Chiky, and G. He´brail, "A Clustering Approach forSampling Data Streams in Sensor Networks," Knowledge andInformation Systems, vol. 32, no. 1, pp. 1-23, July 2012.

[7] W. Liu and T. Wang, "Online Active Multi-Field Learning for Efficient Email Spam Filtering," Knowledge and Information Systems, vol. 33, no. 1, pp. 117-136, Oct. 2012.

[8] J. Lorch, B. Parno, J. Mickens, M. Raykova, and J. Schiffman,"Shoroud: Ensuring Private Access to Large-Scale Data in the Datacenter," Proc. 11th USENIX Conf. File and Storage Technologies (FAST '13), 2013.

[9] D. Luo, C. Ding, and H. Huang, "Parallelization with Multiplicative Algorithms for Big Data Mining," Proc. IEEE 12th Int'l Conf. Data Mining, pp. 489-498, 2012.

[10] J. Mervis, “U.S. Science Policy: Agencies Rally to Tackle Big Data,” Science, vol. 336, no. 6077, p. 22, 2012.

[11] F. Michel, “How Many Photos Are Uploaded to Flickr Every Day and Month?” http://www.flickr.com/photos/franckmichel/ 6855169886/, 2012

[12] T. Mitchell, “Mining our Reality,” Science, vol. 326, pp. 1644-1645, 2009

[13] Nature Editorial, “Community Cleverness Required,” Nature, vol. 455, no. 7209, p. 1, Sept. 2008.

[14] S. Papadimitriou and J. Sun, “Disco: Distributed Co-Clustering with Map-Reduce: A Case Study Towards Petabyte-Scale End-to-End Mining,” Proc. IEEE Eighth Int’l Conf. Data Mining (ICDM ’08), pp. 512-521, 2008.

[15] C. Ranger, R. Raghuraman, A. Penmetsa, G. Bradski, and C. Kozyrakis, “Evaluating MapReduce for Multi-Core and Multiprocessor Systems,” Proc. IEEE 13th Int’l Symp. High Performance Computer Architecture (HPCA ’07), pp. 13-24, 2007.