Textual BigData Analysis by Record-aware Partial Compression Schema using Hadoop

[1]Manasa Goudashivannavar, [2]Dr. ChetanaPrakash

[1]Department of Computer Science and Engineering,
[2]Head of the Department, MCA
Bapuji Institute of Engineering and Technology
Davangere, India

Abstract:**As the amount of data is increasing due to the high usage of social media and Internet it is becoming a critical task in analyzing and storage of this semi-structured and unstructured data. The extraction of useful information from these data is widely acknowledged with consequent logical and business exploitation. Continuous increase in the production of data pushes the data analytic platform to their limits. So the effective means for storing the large quantity of data is by using different Compression techniques which is used to reduce data size which has been employed by using many emerging data analytic platforms. The primary motivation behind using compression techniques is to decrease the storage space and transmission cost over the network. Since general purpose compression strategies try to accomplish high level compression proportions which utilizes information change procedures and logical information. The proposed work includes the techniques for more efficient analysis of textual data using record-aware compression schemes which is appropriate for Hadoop platforms and it will be evaluated for number of standard MapReduce tasks by using a collection of private and public datasets.**

Index Terms—RaPC, Hadoop, BigData, MapReduce

I.INTRODUCTION

A significant measure of data is available as content on the web. With the help of current enormous information examination, distributed computing is turning into an exceptionally fundamental piece of basic leadership forms in banking, healthcare, education, government sectors, open service, and security segments. Because of the steady increment in the information volume, analytic stages, for example, Hadoop and Spark, are under consistent weight, working at the cutoff points of calculation control, organize data transmission, and capacity limit. Right now, there are a few ways to deal with information compression, for example, Run-Length En-coding, Variable-Length Codes, Dictionary-Based, Quantization, and Transforms.Propelled executions of these diverse compression plans have extremely successful outcomes as far as compression proportion. These calculations separate data at square level or carelessness content requesting data. In this manner, they are damaging to the implying that the information, unique substance, its organization and structure are mixed, and the information can't be utilized for examination without decompression. In this work, we present a Record-mindful, Partial Compression plot for huge literary information examination utilizing Hadoop. The method of reasoning behind the RaPC is that any significant strings, e.g., English words, are just important to people. A string and its changed frame (code) don't have any effect to machines, as long as the machines procedure them similarly.

BigData is a term that alludes to specific informational collections or mixes of informational collections whose intricacy, size and rate of development make them hard to be caught, prepared, overseen or broke down by traditional instruments and innovations, for example, social databases and perception bundles or work area insights, inside the time important to make them helpful.

Apache Hadoop is an open-source programming structure used for appropriated limit and storage of information files of enormous data using the MapReduce programming paradigm. It includes PC bundles worked from product hardware.It ought to likewise be noticed that a lot of information being gathered is semi-organized or unstructured , basically in content organization. The forceful speed of information development and the sheer volume of information exhibit real difficulties to all parts of huge information investigation from systematic stages to adaptable calculations. So as to manage the difficulties brought by enormous information, there is exceptionally dynamic continuous advancement of new apparatuses, calculations, computational structures, expository stages and arrangement techniques. Current enormous information investigation improvement centers around scaling up/out explanatory stages and utilizing estimation calculation. Another enhancement, reciprocal to those different advancements is lessen the extent of the source information through compression, and bolster the immediate utilization of the packed information in examination. On current systematic stages, for example, Hadoop, the motivation behind compression is to diminish information estimate keeping in mind the end goal to spare storage room utilizing the Random aware Partial Compression (RaPC).

1) Project objectives

The main objective of this system is to propose Record-aware Partial Compression (RaPC) scheme suitable for textual big data analysis in Hadoop.

- To make the compacted information straight forwardly consumable by MapReduce programs utilizing compression schemas.
- To maximally reduce the data size and minimizing data loading by using compression schema.
- To achieve performance, scalability and reliability by using MapReduce framework.

II. LITERATURE SURVEY

Tian XIA1 et al. [1] in 2008 have assisted the work on Mining of SMS Messages based on Map-Reducing techniques which will be present in Large-Scale. This work proposes a mining approach based on Map-Reduce parallel framework: Firstly, original dataset is pre-processed and grouped by the senders' mobile numbers. Secondly, transformation to regroup the dataset by the short content keys, and then extract the popular messages according to the count of different senders which have the same key. Furthermore, it propose a sentence similarity computation method and a novel Forward Merging and K-Neighbor Checking algorithm to merge the similar messages semantically.

GarhanAttebury et al. [2] in 2009 worked on how Hadoop Distributed File System used for the Grid. This work proposed a newly emerged file system called Hadoop distributed file system (HDFS), is deployed and tested within the Open Science Grid (OSG) middleware stack. Efforts have been taken to integrate HDFS with other Grid tools to build a complete service framework for the Storage Element (SE). Scalability tests show that sustained high inter-DataNode data transfer can be achieved for the cluster fully loaded with data-processing jobs. The WAN transfer to HDFS supported by BeStMan and tuned GridFTPservers shows large scalability and robustness of the system.

Amrit Pal et al. [8] in 2014 described the Analyzing Memory Utilization which includes the Experimental Approach on BigData using Hadoop Cluster. The Apache Hadoop project provides better tools and techniques to handle this huge amount of data. A Hadoop distributed file system for storage and the MapReduce techniques for processing this data can be used. In this proposed system we presented our experimental work done on big data using the Hadoop distributed file system and the MapReduce. We have analyzed the variable like amount of time spend by the maps and the reduce, different memory usages by the Mappers and the reducers.

III. SYSTEM ARCHITECTURE

System architecture is the unique model that characterizes the structure, conduct, and more properties of a framework. Preprocessed data sets will be loaded here as Input dataset. The data will be sent for RaPC L1 compression where it will be compressed by using compression algorithm. The compressed data will be sent for RaPC L2 compression where it will be where it will be further compressed. Map Reduce is a programming paradigm and an associated implementation for processing and generating big data sets with a parallel, distributed algorithm on a cluster. Then the file is uploaded to HDFS.To get the original data it will be decompressed. The below figure depicts the architecture diagram of the textual data is compressed by using 2 layer RaPC compression scheme which is suitable for analysis of the textual data in Hadoop.

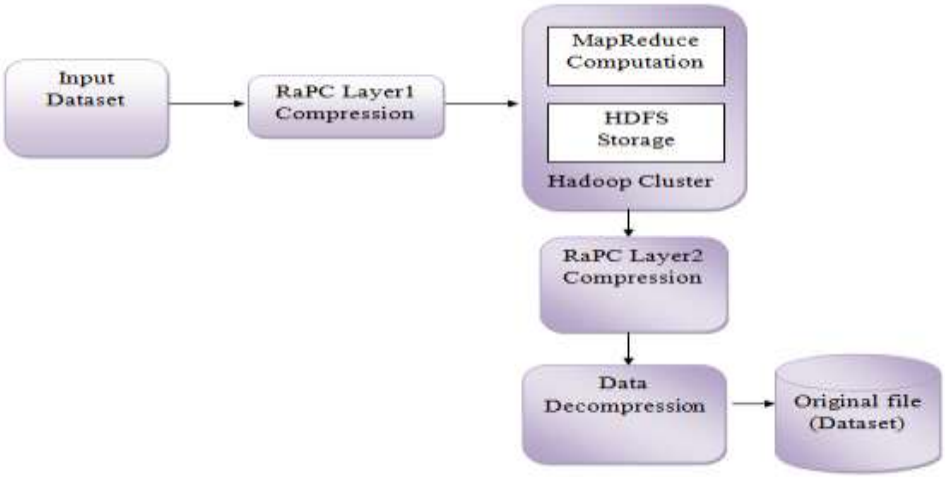

Figure 1 : Architecture Diagram for RaPC

1) Hadoop Distributed File System

A hadoop cluster is the one which involves a single master node and multiple slave or worker nodes. It provides a software framework for distributed storage and processing of big data using the MapReduce programming model. The master node consists of a Job Tracker, Task Tracker, Data Node, and Name Node. A slave or worker node acts as both a Data Node and Task Tracker, though it is possible to have data-only and compute-only worker nodes.

2) MapReduce Framework

MapReduceis a programming computation which is used by hadoop to run the applications, where the data is dealt with in parallel on different CPU center points. The term MapReduce truly implies the going with two one of a kind errands that Hadoop programs perform:

The Map Task: This is the primary errand, which takes input data and converts it into a game plan of data, where particular segments are isolated into tuples (key/regard sets).

The Reduce Task: This errand takes the yield from a guide undertaking as data and merges those data tuples into a tinier plan of tuples. Usually both the information and the yield are secured in a record structure. The MapReduce framework involves a lone expert JobTracker and one slave TaskTracker per gather center point. The slaves TaskTracker execute the errands as composed by the expert and give task status information to the pro discontinuously.

3) Compression algorithm

Step 1: Load the Pre-processed data.

Step 2: Read each line of dataset file using IntWritable().

Step 3: Upload dataset to HDFS and run using hadoop.

Step 4: Create the output file and compress the file using indexing using two layer compression.

Step 5: Store the compressed data in output file.

Step 6: Decompress the data to get the original file.

4) MapReduce algorithm

Step 1: Given an input file to process.

Step 2: Read the data set.

Step 3: Map tasks read each record from input and maps input key value pairs.

Step 4: Reduce task receives the output produced after map processing and then performs parallel operation on the list of values against each key. It then emits output key value pairs by using word count method.

IV. COMPRESSION SCHEMA

RaPC is an embedded two layer compression scheme, in which each layer is particularly designed for the corresponding stage of data processing. The overall design goal of this compression scheme is to maximally reduce the data size for minimizing data loading time.

RaPC Layer-1 Compression

The RaPC Layer-1 encoding is a byte-oriented partial compression scheme. RaPC-L1 only compresses the informational contents. Here we use compression algorithm in which process of reducing the size of a data file is often referred to as data compression. The compression schemes have been evaluated for a number of standard MapReduce analysis tasks using a collection of public and private real-world datasets.

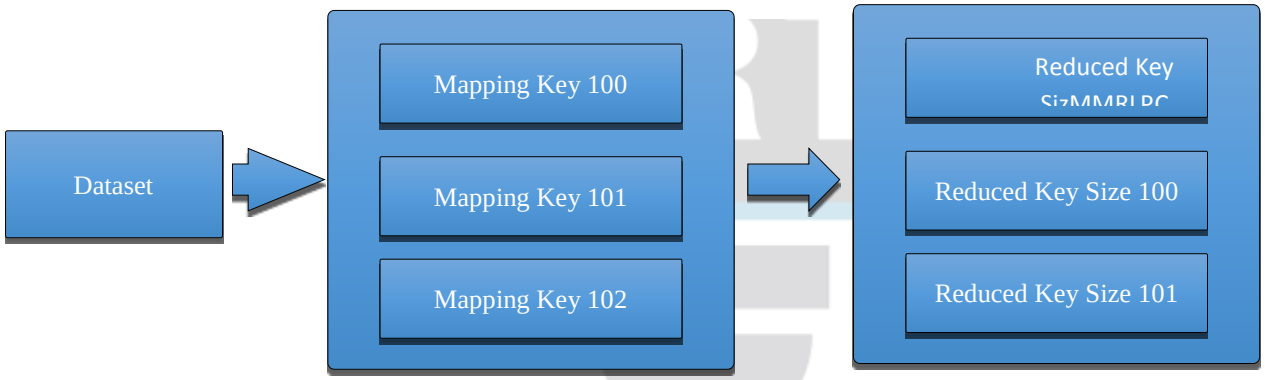

Figure 2: RaPC Layer1 Compression

RaPC Layer-2 Compression

The RaPC Layer-2 used to compress the RaPC-L1 compressed data. The purpose of RaPC-L2 is to maximally reduce data size as much as possible. The RaPC-L2 compression is based on the Deflate algorithm which is block-based. It consists of dictionary method that is LZ77 and a statistical method called Huffman Coding.

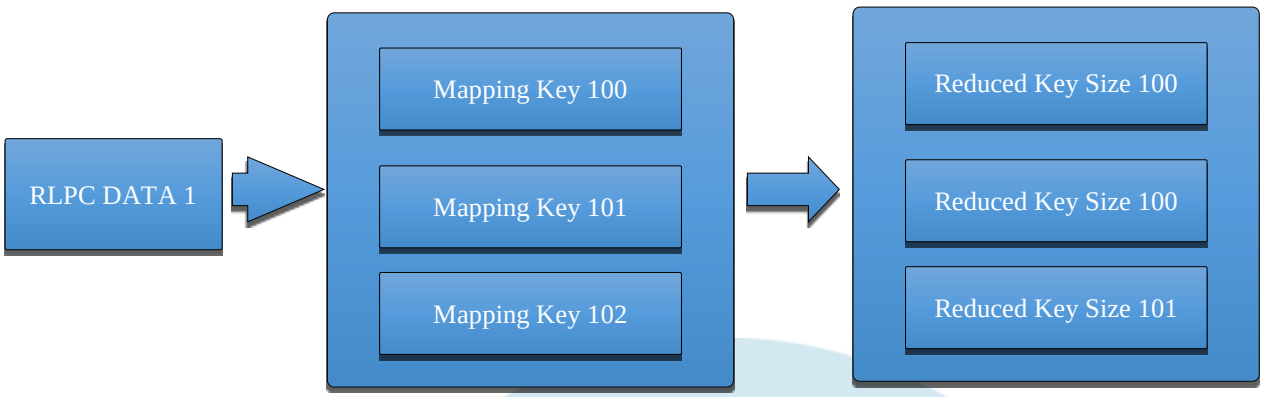

Figure 3: RaPC Layer2 Compression

V. CONCLUSION AND FUTURE SCOPE

This section summarizes and concludes that RaPC is effective for analysis of big data. Based on the fact that text-based files often contain lengthy and repetitive strings, RaPC reduces the size of data by replacing those strings with shorter codes by using compression algorithm. We apply RaPC to various Map Reduce jobs and demonstrate performance gains, storage space relaxation, ease of use and applicability. Additionally RaPC is particularly useful for repetitive analysis for large textual resources, such as social media, web pages, and server logs.

FUTURE SCOPE

The future scope of our proposed system will be:

- At present the proposed system is using only the text based data compression so it can be enhanced to compress the images, videos etc.

- Right now we are using the single dataset for compression, but it can be improved to use the real time data for compression and storage.

- Compression and MapReduce can be improved by applying the enhanced algorithms.

REFERENCES

[1] Large-Scale SMS Messages Mining Based on Map-Reduce" 2008 International Symposium on Computational Intelligence and Design.
[2]Hadoop Distributed File System for the Grid" 2009 IEEE Nuclear Science Symposium Conference Record.
[3] Characterization of Hadoop Jobs using Unsupervised Learning" 2nd IEEE International Conference on Cloud Computing Technology and Science.
[4] Twitter Analytics: A Big Data Management Perspective.
[5] Applying Big Data in Higher Education.
[6]Hadoop Acceleration in an OpenFlow-based cluster" 2012 SC Companion: High Performance Computing, Networking Storage and Analysis.
[7] Dynamic Data Partitioning and Virtual Machine Mapping: Efficient Data Intensive Computation" 2013 IEEE International Conference on Cloud Computing Technology and Science.
[8] An Experimental Approach Towards Big Data for Analyzing Memory Utilization on a Hadoop cluster using HDFS and MapReduce." 2014 First InternationalConference on Networks & Soft Computing.
[9] High Performance Analytics of Bigdata with Dynamic and Optimized Hadoop Cluster" 2016 International Conference on Advanced Communication Control and Computing Technologies (ICACCCT)
[10] O. Goonetilleke, T. Sellis, X. Zhang, and S. Sathe, "Twitter analytics: A big data management perspective," SIGKDD Explor. Newsl., vol. 16, no. 1, pp. 11–20, Sep. 2014.