ISSN : 2347 - 8446 (Online)
ISSN : 2347 - 9817 (Print)

Big Data Analysis Using Distributed Approach on Weather Forecasting Data

[I]Prof. Amit Palve, [II]Ajit Patil
[I]Professor, [II]Student
[I,II]Dept. of Computer Engineering SITRC, SITRC, Nashik, Maharashtra, India

Abstract
Data than it seems at first and extracting the useful information in an efficient manner leads a system toward major computational challenges, such as to analyze, aggregate, and store, where data are remotely collected. Keeping in view the above mentioned factors, there is a need for designing a system architecture that welcomes both real-time, as well as offline data processing. Big data is data whose characteristics force us to look beyond the traditional methods that are prevalent at the time. Online news, micro-blogs, search queries are just a few examples of these continuous streams of user activities. Evolving data streams methods are becoming a low-cost, green methodology for real time online prediction and analysis. Heterogeneity, scale, timeliness, complexity, and privacy problems with Big Data impede progress at all phases of the pipeline that can create value from data. The problems start right away during data acquisition, when the data tsunami requires us to make decisions, currently in an ad hoc manner, about what data to keep and what to discard, and how to store what we keep reliably with the right metadata

Keywords
Big data, ETL, Data processing unit, RSDU, DADU, real-time.

I. Introduction

Today over the past five years, the authors and many others at Google have implemented hundreds of special-purpose computations that process large amounts of raw data, such as crawled documents, web request logs, etc., to compute various kinds of derived data, such as inverted indices, various representations of the graph structure of web documents, summaries of the number of pages crawled per host, the set of most frequent queries in a given day, etc. Most such computations are conceptually straightforward. However, the input data is usually large and the computations have to be distributed across hundreds or thousands of machines in order to finish in a reasonable amount of time. The issues of how to parallelize the computation, distribute the data, and handle failures conspire to obscure the original simple computation with large amounts of complex code to deal with these issues.

As a reaction to this complexity, the designed a new abstraction that allows us to express the simple computations so trying to perform but hides the messy details of parallelization, fault tolerance, data distribution and load balancing in a library. I realized that most of our computations involved applying a map operation to each logical record our input in order to compute a set of intermediate key/value pairs, and then applying a reduce operation to all the values that shared the same key, in order to combine the derived data appropriately. Our use of a functional model with user specified map and reduce operations allows us to parallelize large computations easily and to use re-execution as the primary mechanism for fault tolerance.

Data stream real time analytics are needed to manage the data currently generated, at an ever increasing rate, from such applications as: sensor networks, measurements in network monitoring and traffic management, log records or clickstreams in web exploring, manufacturing processes, call detail records, email, Blogging, twitter posts and others. In fact, all data generated can be considered as streaming data or as a snapshot of streaming data, since it is obtained from an interval of time. In the data stream model, data arrive at high speed, and algorithms that process them must do so under very strict constraints of space and time. Consequently, data streams pose several challenges for data mining algorithm design. First, algorithms must make use of limited resources (time and memory). Second, they must deal with data whose nature or distribution changes over time. I need to deal with resources in an efficient and low-cost way. Green computing is the study and practice of using computing resources efficiently. A main approach to green computing is based on algorithmic efficiency. In data stream mining, I was interested in three main dimensions: Accuracy, Amount of space (computer memory) necessary. The time required to learn from training examples and to predict.

II. Literature Survey

A. Adamu Galadima describes a brief look at the Arduino microcontroller and some of its applications and how it can be used in learning. Arduino is an open source microcontroller used in electronic prototyping. Arduino hardware and its components shall be looked at. Software and the Environment that Arduino runs on are both looked at too. Some applications will be taken as examples that can help make learning Arduino more interesting. This can be used as a major way to encourage students and others to learn more about electronics and programming.

B. Jeffrey Cohen present data parallel algorithms for sophisticated statistical techniques, with a focus on density methods. Finally, he reacts on database system features that enable agile design and flexible algorithm development using both SQL and Map Reduce interfaces over a variety of storage mechanisms.

C. Brian Dolan present the design philosophy, techniques and experience providing MAD analytics for one of the world's largest advertising networks at Fox Audience Network, using the Green plum parallel database system. We describe database design methodologies that support the agile working style of analysts in these settings.

D. R. P. Singh explain why a cloud-based solution is required, describe our prototype implementation, and report on some example applications we have implemented that demonstrate personal data ownership, control, and analytics. He address these issues by designing and implementing a cloud-based architecture that provides consumers with fast access and

fine-grained control over their usage data, as well as the ability To analyze this data with algorithms of their choosing, including third party applications that analyze that data in a privacy preserving fashion.

E. Jeffrey Dean describes the basic programming model and gives several examples. Many real world tasks are expressible in these models. Implementation of Map Reduce runs on a large cluster of commodity machines and is highly scalable: a typical Map Reduce computation processes many terabytes of data on thousands of machines. Programmers and the system easy to use: hundreds of Map Reduce programs have been implemented and upwards of one thousand Map Reduce jobs are executed on Google's clusters every day.

F. Panagiotis D. Diamantoulakis implements the Big Data Analytics for Dynamic Energy Management in Smart Grids. The smart electricity grid enables a two-way flow of power and data between suppliers and consumers in order to facilitate the power flow optimization in terms of economic efficiency, reliability and sustainability. This infrastructure permits the consumers and the micro energy producers to take a more active role in the electricity market and the dynamic energy management (DEM). The most important challenge in a smart grid (SG) is how to take advantage of the user's participation in order to reduce the cost of power.

G. L. Aniello explores the idea of a framework leveraging multiple data sources to improve protection capabilities of CIs. Challenges and opportunities are discussed along three main research directions: i) use of distinct and heterogeneous data sources, ii) monitoring with adaptive granularity, and iii) attack modeling and runtime combination of multiple data analysis techniques.

III. Methodology

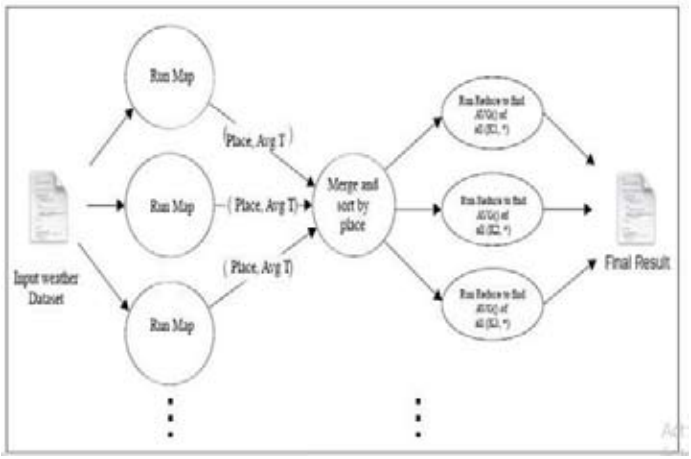

Fig. 1 : Proposed System

Proposed System is divided into four main parts like

1. Input weather dataset
2. Run Map
3. Run map reduce

In proposed system I have divided the system in four parts in the first part we have to give the weather dataset as input. Then run the map reduces in second stage after map reduces we have merge and sort the result by place then we run the map reduce to get final results.

IV. Architecture

The term Big Data covers diverse technologies same as cloud computing. The input of Big Data comes from social networks, Web servers, satellite imagery, sensory data, banking transactions, etc. Regardless of very recent emergence of Big Data architecture in scientific applications, numerous efforts toward Big Data analytics architecture can already be found in the literature. Among numerous others, the proposed remote sensing Big Data architecture to analyze the Big Data in an efficient manner as shown in Remote sensing Big Data architecture, the delineates n number of satellites that obtain the earth observatory Big Data images with sensors or conventional cameras through which sceneries are recorded using radiations. Special techniques are applied to process and interpret remote sensing imagery for the purpose of producing conventional maps, thematic maps, resource surveys, etc.

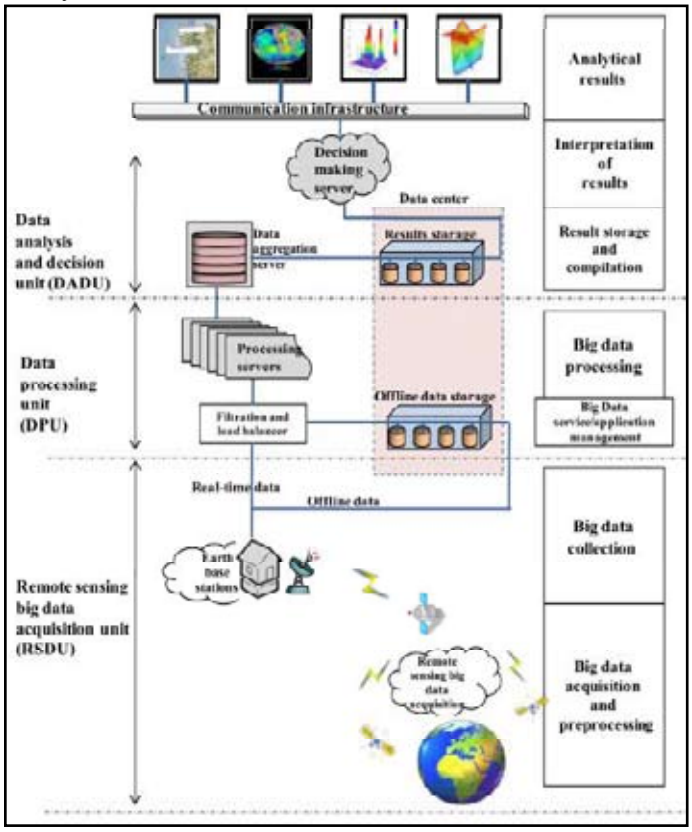

Fig. 2 : Remote sensing Big Data architecture

We have divided remote sensing Big Data architecture into three parts

1. Remote sensing data acquisition unit(RSDU)
2. Data Processing Unit (DPU)
3. Data analysis and decision unit (DADU).

The functionalities and working of the said parts are described as below.

A. Remote Sensing Big Data Acquisition Unit

Remote sensing promotes the expansion of earth observatory system as cost-effective parallel data acquisition system to satisfy specific computational requirements. The Earth and Space Science Society originally approved this solution as the standard for parallel processing in this particular context [2]. As satellite instruments for Earth observation integrated more sophisticated qualifications for improved Big Data acquisition, soon it was

ISSN : 2347 - 8446 (Online)
ISSN : 2347 - 9817 (Print)

recognized that traditional data processing technologies could not provide sufficient power for processing such kind of data. Therefore, the need for parallel processing of the massive volume of data was required, which could efficiently analyze the Big Data. For that reason, the proposed RSDU is introduced in the remote sensing Big Data architecture that gathers the data from various satellites around the globe.

B. Data Processing Unit

In data processing unit (DPU), the filtration and load balancer server have two basic responsibilities, such as filtration of data and load balancing of processing power. Filtration identifies the useful data for analysis since it only allows useful information, whereas the rest of the data are blocked and are discarded. Hence, it results in enhancing the performance of the whole proposed system. Apparently, the load balancing part of the server provides the facility of dividing the whole filtered data into parts and assign them to various processing servers. The filtration and load-balancing algorithm varies from analysis to analysis; e.g., if there is only a need for analysis of sea wave and temperature data, the measurement of these described data is filtered out, and is segmented into parts.

C. Data Analysis and Decision Unit

DADU contains three major portions, such as aggregation and compilation server, results storage server(s), and decision making server. When the results are ready for compilation, the processing servers in DPU send the partial results to the aggregation and compilation server, since the aggregated results are not in organized and compiled form. Therefore, there is a need to aggregate the related results and organized them into a proper form for further processing and to store them. In the proposed architecture, aggregation and compilation server is supported by various algorithms that compile, organize, store, and transmit the results. Again, the algorithm varies from requirement to requirement and depends on the analysis needs. Aggregation server stores the compiled and organized results into the results storage with the intention that any server can use it as it can process at any time. The aggregation server also sends the same copy of that result to the decision making server to process that result for making decision. The decision-making server is supported by the decision algorithms, which inquire different things from the result, and then make various decisions (e.g., in our analysis, we analyze land, sea, and ice, whereas other finding such as are, storms, Tsunami, earthquake can also be found). The decision algorithm must be strong and correct enough that efficiently produce results to discover hidden things and make decisions. The decision part of the architecture is significant since any small error in decision-making can degrade the efficiency of the whole analysis. DADU finally displays or broadcasts the decisions, so that any application can utilize those decisions at real time to make their development. The applications can be any business software, general purpose community software, or other social networks that need those findings.

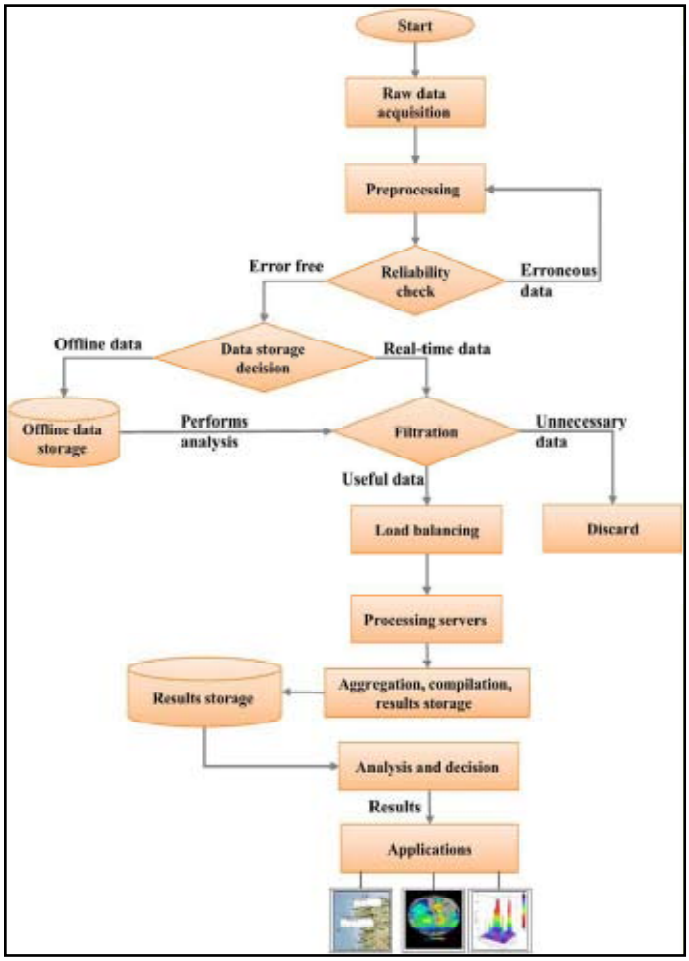

Fig. 3 : Remote sensing earth observatory image.

V. Dataset

We are performing experiments to decide the best tools among Distributed computing using Pig and Hive Queries. The previous architecture efficiently processed and analyzed real time and offline remote sensing big data for decision making.

1. To propose architecture for big data that comes from the real time remote sensing application.
2. To propose architecture this efficiently processed and analyzed real-time and off-line remote sensing Big Data for decision-making.
3. To proposed architecture to make it compatible for Big Data analysis for all Applications, e.g. sensors and social networking.

Steps

1. Collecting Raw Data
2. Data gathering at offline storage.
3. Preprocessing operation perform on gathered data.
4. Data aggregation process can be done by using the cluster.
5. Prediction process contains the collaborative filtering.
6. Result storage module displays the possibilities of the weather forecasting.

Table 1 : Dataset

Sr. No.	*Number of Attribute*	*Data Type*	*Number of Rows*
1.	28	Numeric	Ten Lack

ftp://ftp.ncdc.noaa.gov/pub/data/uscrn/products/daily01

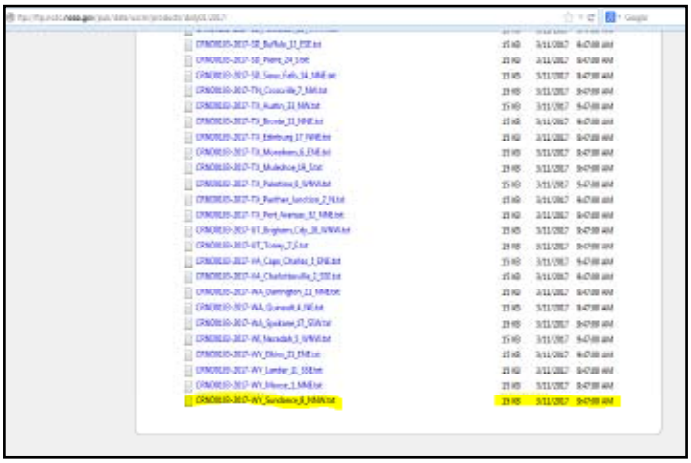

Fig. 4 : Sample Dataset

VI. Results And Comparison

The Proposed architecture is implemented in Java using eclipse 7.0 version. As a part of Data preprocessing, Distributed file system is implemented using HADOOP-1.8.0 for storing large amount of data and stored in different nodes. To implement Data Analysis unit, Map functions are designed using Java language by taking large data which was distributed in different nodes as input. In Hadoop, Map function takes the data set column offset as a key and the value in column as a parameter. Since Hadoop Map Reduce cannot directly process Id, the whole product data are converted into sequence file to be processed using Map Reduce. Following table shows result analysis of preprocessing algorithm on different data size.

Table II : Comparison between algorithms

Dataset	*Base Algorithm*	*Hadoop Algorithm*
Austin_33_NW	4256 Ms	2660 Ms
Brigham_City_28_WNW	3824 Ms	2390 Ms
Cape_Charles_5_ENE	3664 Ms	2290 Ms
Charlottesville_2_SSE	3872 Ms	2420 Ms
Darrington_21_NNE	3648 Ms	2280 Ms
Dataset	Base Algorithm	Hadoop Algorithm
Austin_33_NW	4256 Ms	2660 Ms

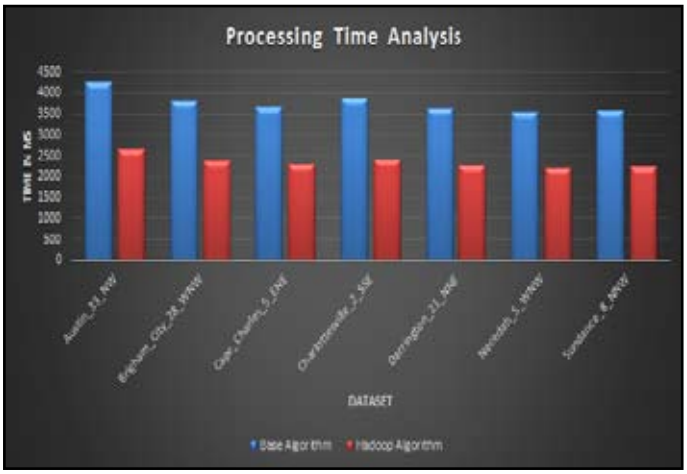

Fig. 5 : Results and Comparison

For future work, propose approach can be extend to make it compatible for Big Data analysis for all applications, e.g., sensors and social networking.

Conclusion

Finally, the system will sense the current environmental status. System gives the indication of the weather forecasting. The algorithms for each unit and subunits are used to analyze remote sensing data sets, which helps in better understanding of land and sea area. This architecture welcomes researchers and organizations for any type of remote sensory Big Data analysis by developing algorithms for each level of the architecture depending on their analysis requirement. The better analysis of the large volumes of data that are becoming available, there is the potential for making faster advances in many scientific disciplines and improving the portability and success of many enterprises. Here, only begun to see its potential to collect, organize, and process data in all walks of life. A modest investment by the federal government could greatly accelerate its development and deployment.

References

[1] Muhammad Mazhar Ullah Rathore, Anand Paul , "Real-Time Big Data Analytical Architecture for Remote Sensing Application", IEEE Journal of Selected Topics In Applied Earth Observations And Remote Sensing, Vol. 8, No. 10, October 2015.

[2] R. A. Dunge, "A Survey on Big Data in Real.",vol No. 2, issue No4,IJRITCC, 2015.

[3] D. Agrawal, S. Das, and A. E. Abbadi, "Big Data and cloud computing: Current state and future opportunities," in Proc. Int. Conf. Extending Database Technol. (EDBT), 2011, pp. 530533.

[4] J. Cohen, B. Dolan, M. Dunlap, J. M. Hellerstein, and C. Welton, "Mad skills: New analysis practices for Big Data," PVLDB, vol. 2, no. 2, pp. 14811492, 2009.

[5] A. Cuzzocrea, D. Sacc, and J. D. Ullman, "Big Data: A research agenda," in Proc. Int. Database Eng. Appl. Symp. (IDEAS13), Barcelona, Spain, Oct. 0911, 2013.

[6] R. A. Schowengerdt, "Remote Sensing: Models and Methods for Image Processing", 2nd ed. New York, NY, USA: Academic Press, 1997.

[7] D. A. Landgrebe, "Signal Theory Methods in Multispectral Remote Sensing", Hoboken, NJ, USA: Wiley, 2003.

[8] C.-I. Chang, "Hyperspectral Imaging: Techniques for Spectral Detection and Classification", Norwell, MA,USA, Kluwer, 2003.

[9] J. A. Richards and X. Jia, "Remote Sensing Digital Image Analysis: An Introduction," New York, NY, USA: Springer, 2006.

[10] J. Shi, J. Wu, A. Paul, L. Jiao, and M. Gong, "Change detection in synthetic aperture radar image based on fuzzy active contour models and genetic algorithms," Math. Prob. Eng., vol. 2014, 15 pp., Apr. 2014.

[11] Paul, K. Bharanitharan, and J.-F.Wang, "Region similarity based edge detection for motion estimation in H.264/ AVC," IEICE Electron. Express, vol. 7, no. 2, pp. 4752, Jan. 2010.

[12] A. Paul, K. Bharanitharan, and J.-F.Wang, "Region similarity based edge detection for motion estimation in

[13] A. Paul, J. Wu, J.-F. Yang, and J. Jeong, "Gradient-based

ISSN : 2347 - 8446 (Online)
ISSN : 2347 - 9817 (Print)

edge detection for motion estimation in H.264/AVC," IET Image Process., vol. 5, no. 4, pp. 323327, Jun. 2011.

[14] A.-C. Tsai, A. Paul, J.-C. Wang, and J.-F. Wang, "Intensity gradient technique for efficient intra prediction in H.264/ AVC," IEEE Trans. Circuits Syst. Video Technol., vol. 18, no. 5, pp. 694698, May 2008.

[15] A. Plaza et al., "Recent advances in techniques for hyperspectral image processing," Remote Sens. Environ., vol. 113, pp. 110122, 2009.

[16] S. Kalluri, Z. Zhang, J. JaJa, S. Liang, and J. Townshend, "Characteriz-ing land surface anisotropy from AVHRR data at a global scale using high performance computing," Int. J. Remote Sens., vol. 22, pp. 21712191, 2001.

Author's Profile

Prof. Amit Palve currently working as Assistant Professor in Department of Computer Engineering of Sandip Institute of Technology & Research Centre, Nashik. He has teaching experience of 8 year with the expert area of work in IOT (Internet of Thing), data analytics and image processing; wireless sensor network. He has published more than 10 research papers in reputed international journals and it's also available online.

Mr. Ajit Patil currently pursuing post-graduation degree in Department of Computer Engineering of Sandip Institute of Technology & Research Centre, Nashik. He have published 3 papers in international journals. Also attended many conference and seminars.