Prediction of Natural Calamities Using Segregated Satellite Images

M.Murugesan[1],P.Baby[2]
[1] Head of Department, [2] PG Scholar,
Department of Computer Science and Engineering,
M.Kumarasamy college of Engineering, Karur, India

***Abstract* - Big data is the domain where most recent works are going on.The real time applications which handles huge amount of data can be implemented and analysed using Big data techniques. Accounting remote sensing application the satellite images are collected, Preprocessed and segregated into images of Land ,Sea, Ice. The existing architecture contains Remote Sensing Big data Acquisition Unit(RSDU), Data Processing Unit(DPU), Data Analysis and Decision Unit(DADU) supports only segregation and storage of images. Thus further analytical process on these segregational images is needed to predict the Natural calamities like Tsunami, Earthquake and Fire detection. The proposed system is reliable and fault tolerant when compared to the existing systems as it collects the data from the user to predict the interest and analyses the item to find the features. The system is also adaptive as it updates the list frequently and finds the updated interest of the user.**

***Index Terms* - Remote Sensing Big data Acquisition Unit,Data Processing Unit,Data analysis and Decision Unit.**

I. INTRODUCTION

Big data is a emerging domain in which many researches are going on. Big data domain makes impact on various fields like Business,Scientific Technologies. In particular in the field of remote sensing big data place a vital role in the prediction of natural calamities.

Big data contains the characterstics like volume,velocity,variety,complexity.In volume the data is collected from variety of sources like sensor,social media,Business transaction.In velocity the data sources to the big data storage comes at variant speed.In variety the different form of data are allowed to store. The data are like structured and unstructured,audio and video.

In complexity the data are comes from different resources which are difficult to handle,match and transform data across systems.But it necessary to connect and correlated.

II. LITERATURE SURVEY

Bhatotia use a system called In coop, which permit existing MapReduce programs, not calculated for incremental processing, to execute visibly in an incremental manner. In Incoop, calculation can respond repeatedly and professionally to modifications to their input data by reusing middle results from previous runs, and incrementally inform the output according to the modify in the input.[4].

Borkar present Pregelix, a large-scale graph analytics system that we began in 2011. Pregelix obtain a novel set-oriented, iterative dataflow approach to apply the user level Pregel programming model. It achieves so by treating the messages and vertex states in a Pregel calculation like tuples with a well-defined schema; it then employ database-style query evaluation techniques to execute the user's program.[6].

Howe provides Hadoop, a customized version of the Hadoop MapReduce framework that is planned to serve these applications. Hadoop not only extends MapReduce with programming support for iterative applications, it also considerably improves their effectiveness by making the task scheduler loop-aware and by adding various caching mechanisms.[7].

Ekanayake implements Twitter framework which is an improved MapReduce runtime with an extensive programming model that supports iterative MapReduce computations capably. It uses a publish/subscribe messaging infrastructure for communication and information transfers, and chains long running map/reduce tasks, which can be used in "organize once and use many times" approach.[10].

Ewen propose a method to combine incremental iterations, a form of work-set iterations, through parallel data flows. After presentation how to mix bulk iterations into a dataflow system and its optimizer, we current an extension to the programming model for incremental iterations[1].

III. WEB DATA ANALYZING USING MAP REDUCE APPROACH

Map Reduce is one of the core component of Apache hadoop software framework. It is a programming model that allows large number of servers in hadoop cluster. The term Map reduce contains two separate tasks.The task is Map job.It takes a set of data and convert it into an another set of data.the data are divided ito tuples(key/value pairs).the second task is to be reduce job.The input to the Reduce job is the output of the Map job and combines data tuples into a smaller set of tuples.

IV. PROPOSED SYSTEM

1. Image Data set uploads

The image data sets are uploaded.These data sets are stored in the database.

2. Preprocessing

The goal of the preprocessing is to be filter out the noise from the images.It is a nonlinear operation used in a image processing.The adaptive weight algorithm are used to remove the noise from the image.It is mainly used to remove the salt and pepper noise from the image.

A. Remote Sensing Big Data Acquisition Unit(RSDU)

Remote sensing promotes the development of earth observatory system as gainful parallel data acquisition system to satisfy specific computational rations. The proposed RSDU is introduced in the remote sensing Big Data planning that gathers the images from various satellites around the globe. For successful image data analysis, remote sensing satellite preprocesses image under many situations to integrate the image data from different sources, which not only reduces storage cost, but also improves investigation accuracy. Some relational data preprocessing techniques are image integration, image cleaning, and redundancy elimination. After preprocessing phase, the collected images are transmitted to a ground station using downlink control. This spread is directly or via relay satellite with an appropriate tracking antenna and communication link in a wireless environment. The images must be corrected in different methods to eliminate distortions caused due to the motion of the platform relation to the earth, platform attitude, earth curvature, no uniformity of illumination, variations in sensor uniqueness, etc.

The image is then transmitted to Earth Base Station for advance processing via direct communication link. We divided the images processing procedure into two steps, such as real-time Big Data processing and offline Big Data image processing. In the case of offline image processing, the Earth Base Station transmits the images to the data center for storage.

B. Data Processing Unit(DPU):

In data processing unit (DPU), the filtration and shipment balancer server have two basic responsibilities, such as filtration of data and shipment matching of processing power. Filtration identifies the useful images for analysis since it only allows useful information, whereas the rest of the images are blocked and are discarded. Hence, it results in pretty the performance of the whole proposed system. Apparently, the shipment matching part of the server provides the ability of separating the whole clean images into parts and assign them to various processing servers.

C. Data Analysis and Decision Unit (DADU):

DADU contains three major portions, such as aggregation and anthology server, results storage space server(s), and judgment making server. In the proposed architecture, aggregation and anthology server is supported by various algorithms that collect, organize, store, and transmit the results. Again, the algorithm varies from condition to condition and depends on the analysis needs. Aggregation server stores the compiled and organized results into the result's storage with the goal that any server can use it as it can process at any time. The aggregation server also sends the equal copy of that result to the managerial server to process that result for making judgment. The executive server is supported by the judgment algorithms, which inquire different things from the result, and then make various judgement (e.g., in our analysis, we analyze land, sea, and ice, whereas other finding such as fire, storms, Tsunami, earthquake can also be found). The judgment algorithm must be strong and correct enough that efficiently produce results to discover covered things and make decisions. The judgment part of the architecture is significant since any small error in managerial can degrade the efficiency of the whole analysis. DADU finally displays or broadcasts the judgment, so that any application can utilize those judgment at real time to make their development. The applications can be any business software, general idea community software, or other social networks that need those findings (i.e., managerial).

V. ALGORITHMS PARAMETERS AND VARIABLES

Following are the Parameters and variables used in the proposed algorithms. B1, B2, B3, B4, B5 . . . BN is image fixed size mass NR: Number of records in MDS (number of lines in the image). NSR: Number of example in each record (number of pixels in each line). BS: Image mass size (i.e., block size of B1, B2…BN). N: Total number of mass in the image.

i.e., $N = NR \times NSR\ BS$.

PSB: procedure sample mass

(PSB = {B1, B3, B5, B7 . . . BN−1}).

UPSB: Unprocessed sample mass

(UPSB = {B2, B4, B6,B8 . . . BN}).

XBi: Mean of sample values of mass Bi, where i = {1, 2, 3, 4 . . . N}.

Abs_Diff: Absolute difference between XBi and SDBi.

AbsDiff $= XBi - SDBi$

Maxval: brink value is set and is greater than normal range to verify how many values of the mass are deviated from the normal range.

NGmaxval: Number of values in the mass is greater than Maxval. Below are the brink variables, which are set on the basis of analysis, i.e.

∂X: Mean brink is set on the basis of analysis, which is used to compare the mean value of each mass with threshold for detecting land or any other area.

∂SD: SD brink is set on the basis of analysis, which is used to compare the SD value of each mass with threshold for detecting land or any other area.

∂Abs_diff: Absolute difference brink is set on the basis of analysis, which is used to compare absolute value of each mass with threshold for detecting land, sea, or any other area.

Filteration Loadbalancing Algorithm

Input: Satellite process data set/product
Output: filtered Image data in fixed size mass and send each mass to processing server
Steps:
1. Filter Image related data i.e. Processed data in MDS. All other redundant data will be discarded.
2. Divide the image into fixed size mass i.e. BS = 100 × 100 MDS process data ethics, row by row fashion or column by column. Each mass will be denoted by Bi where $1 \leq i \leq BS$
3. Make two samples of mass so that only half of the part is processed. i.e., PSB = {B1,B3,B5, . . . ,BN−1} and UPSB = {B2,B4,B6,B8, . . . ,BN} 4. spread UPSB directly to aggregation server without processing.
5. Assign and transmit each distinct mass(s) Bi of PSB to various processing servers in DPU.

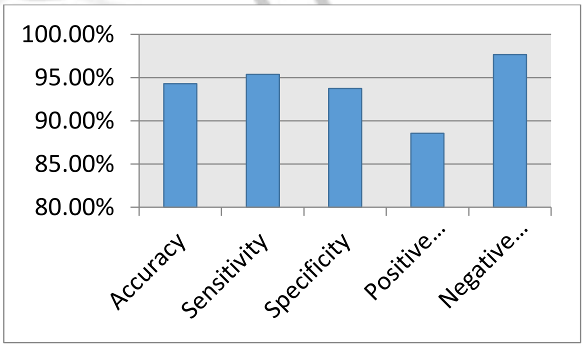
Processing and Calculation Algorithm
Input: Block Bi
Output: statistical parameters results and transmit them to aggregation server.
Steps:
1. For each Block Bi, Calculate
a. *X*Bi
b. S.DBi
c. Abs_Diff
d. NGmaxval
2. spread the results against block id and product id to the aggregation server in DADU

Aggregation and anthology Algorithm
Input: Block Bi results
Output: compile, storing and sending PSB results and UPSB mass information to managerial server.
Steps:
1. Collect Every Bi's result of PSB
2. Compile them and transmit them to managerial server.
3. Store PSB mass with results and UPSB mass without result into RBMS in result storage.

Judgment Making Algorithm
Input: PSB results and UPSB information
Output: each mass Bi with decision, land block or sea. Finally, the total image is divided into sea and land area
Rules:
Following rules are ready on the basis of land area examination
discussed in Section III for detect land mass
1. X Bi $\leq \partial \boldsymbol{X}$
2. S.DBi $\geq \partial \boldsymbol{S.D}$
3. Abs_Diff $\geq \partial \boldsymbol{Abs\_diff}$
4. NGmaxval $\leq \partial \boldsymbol{NGmaxval}$

VI. PERFORMANCE ANALYSIS

The proposed work was done on big data that starts from data mining to web mining to big data mining. We can design log analyzer using Hadoop framework. Then analyze the performance of the system using response time metrics for both existing and proposed approach and shown as graph in fig 4. To get output on input image the process of matching database image that images feature point matched in database image, message alert automatically send to particular person or in charge.



VII. CONCLUSION AND FUTURE WORK

In this project, we discuss our implementation of an earthquake prediction, Tsunami prediction, and fire detection monitoring system for managing and monitoring problems of human. The specification of our implemented project is based on Hadoop platform. The proposed building comprises three main units, such as remote sensing Big Data acquisition unit; data processing unit (DPU); and data investigation decision unit (DADU). First, RSDU acquires images from the satellite and sends these images to the bottom position, where initial processing takes place. Second, DPU plays a vital role in structural design for capable processing of real-time Big Data with providing filtration, shipment matching, and parallel processing. Third, DADU is the superior layer unit of the proposed architecture, which is responsible for collection, storage of the results, and creation of decision based on the results received from DPU. We illustrated the capability and effectiveness of the proposed method using accuracy. The results showed that the proposed method improved the accuracy up to 80% compared with various and combined classifiers. Although the prediction accuracy improvements in terms of accuracy metric achieved from the experiments sometimes was not significant, the experiment results have shown the improvement of the models for further developing suitable prediction models.

VIII. ACKNOWLEDGMENT

I wish to express my sincere thanks to Dr.S.Thilagamani M.E., PhD., Dean of Computer Engineering and Mr.M.Murugesan M.E., Head of the Department of Computer Science and Engineering and to my project Supervisor Mr.M.Murugesan M.E., Head of the Department for their guidance and timely support for me to complete this work successfully.

REFERENCES

[1] S. Ewen, K. Tzoumas, M. Kaufmann, and V. Markl. Spinning fast iterative data flows. PVLDB, 5(11):1268–1279, 2012.

[2] T. J¨org, R. Parvizi, H. Yong, and S. Dessloch. Incremental recomputations in mapreduce. In Proc. Of CloudDB ’11, 2011.

[3] R. Agrawal and R. Srikant. Fast algorithms for mining association rules in large databases. In Proc. of VLDB ’94, pages 487–499, 1994.

[4] P. Bhatotia, A. Wieder, R. Rodrigues, U. A. Acar, and R. Pasquin. Incoop: Mapreduce for incremental computations. In Proc. of SOCC ’11, 2011.

[5] S. Brin and L. Page. The anatomy of a large-scale hypertextual web search engine. Comput. Netw. ISDN Syst., 30(1-7):107–117, Apr. 1998.

[6] Y. Bu, V. Borkar, J. Jia, M. J. Carey, and T. Condie. Pregelix: Big(ger) graph analytics on a dataflow engine. PVLDB, 8(2):161–172, 2015.

[7] Y. Bu, B. Howe, M. Balazinska, and M. D. Ernst. Haloop: efficient iterative data processing on large clusters. PVLDB, 3(1-2):285–296, 2010.

[8] J. Cho and H. Garcia-Molina. The evolution of the web and implications for an incremental crawler. Technical Report 1999-22, Stanford InfoLab, 1999.

[9] J. Dean and S. Ghemawat. Mapreduce: simplified data processing on large clusters. In Proc. of OSDI ’04, 2004.

[10] J. Ekanayake, H. Li, B. Zhang, T. Gunarathne, S.-H. Bae, J. Qiu, and G. Fox. Twister: a runtime for iterative mapreduce. In Proc. of MAPREDUCE ’10, 2010.