International Journal of Advanced Technology in Engineering and Science
Vol. No.3, Special Issue No. 01, November 2015
www.ijates.com
ijates
ISSN 2348 - 7550

EFFICIENT CLINICAL DATA ANALYSIS USING INTERNET OF THINGS

Poornimathi Krishnan[1], Jeyalakshmi Jeyabalan[2], Sreesubha Soundarrajan[3], Sindhuja M[4], Anitha Jaikumar[5]

[1,2,3,4,5]*Department of Information Technology,*

Rajalakshmi Engineering College, Thandalam, Chennai, (India)

ABSTRACT

Analyzing abundant healthcare data is a computationally intensive task and combining with standard clinical data adds additional layers of complexity. There are various pervasive healthcare techniques available but still advancements are always open in this area with respect to improvement of efficiency in terms of cost, precision and speed. The proposed healthcare monitoring system is designed to gather and share patient information directly with healthcare monitoring system, making it possible to collect, record and analyze new data streams faster in order to accurately describe the health and fitness of the patients. The analysis part is done on the monitoring centre using big data which takes care of large scale distributed processing. The speed is taken care by means of including a distributed cache mechanism. The proposed system design includes two type of cache memory, one is local centralized cache, and other is distributed cache memory. These cache memories are used for reducing the recompilation time. Thus improvisation is provided in speed and time of computation and decision making.

Index Terms: IoT, Pervasive Healthcare, Big Data Analytics, Caching.

I. INTRODUCTION

Internet of Things (IoT), uses several devices and shares information directly with each device and the cloud, making it possible to collect, record and analyze new data streams faster and more accurately.

Healthcare data is of many types which includes the following types Structured EHR Data, Unstructured Clinical Notes, Medical Imaging Data, Genetic Data, Epidemiology & Behavioral data. Analyzing this multi faceted data can be cumbersome. More advanced search and analytics techniques and methodologies are required to meet the performance constraints like time. Healthcare is a more critical area where time delay can be more sensitive and costly in terms of human lives. So using such a technology which meets the time criteria is mandatory. Internet of Things(IoT) and Big data Analytics is a boon for Healthcare data analytics wherein data analytics and visualization is possible.

Internet of Things (IoT) is the network of physical objects or "things" embedded with electronics, software, sensors and connectivity to enable it to achieve greater value and service by exchanging data with the manufacturer, operator and/or other connected devices based on the infrastructure of International Telecommunication Union's Global Standards Initiative. Internet of Things connect physically and remotely by

individuals, for both public sector and private sector, in the sense of a computer network grid, of a created electrical device that is in place, with economic benefit and potential usefulness. Each thing is uniquely identifiable through its embedded computing system but is able to interoperate within the existing Internet infrastructure.

Things, in the IoT, can refer to a wide variety of devices such as heart monitoring implants, biochip transponders on farm animals, electric clams in coastal waters, automobiles with built-in sensors, or field operation devices that assist fire-fighters in search and rescue, These devices collect useful data with the help of various existing technologies and then autonomously flow the data between other devices. The data collected from these devices is analyzed using Big Data.

Big data Analytics is serving to be a boon in clinical data analysis providing way for different variety, veracity, volume and velocity of data. The Hadoop framework provides a distributed cluster framework for analysis. The speed of computation can be made faster and even faster with the help of the Distributed cache mechanisms. The paper provides the related work in Chapter 2, system description and architecture in chapter 3 and conclusion in Chapter 4.

II. RELATED WORK

The literature survey has been carried out to explore the previous works done in the relevant areas namely: Big data, Hadoop, Map reduce, Applications of map reduce, Cache. Demchenko et Al [3] discusses a nature of Big Data that may originate from different scientific, industry and social activity domains and proposed big data definition that includes data models and structures, data analytics, infrastructure and security. The big data architecture addresses all aspects of the big data components and analyses the requirements to provide suggestion on how the components can address the big data issue. The following literature provide the existing systems in HDFS and Map Reduce.

Zibin Zheng, Jieming Zhu and Michael R.Lyu[7] provides an overview of the big data and Big Data-as-a-Service. First, various types of service-generated big data are exploited to enhance system performance. Then Big Data-as-a-Service, including Big Data Infrastructure-as-a-Service, Big Data Platform-as-a-Service, and Big Data Analytics Software-as-a-Service is employed to provide common big data related services.

Meenakshi Shrivatava et al [12] presented the feature of the HDFS which aims to provide storage to large files so that the read and write performance for such not degraded.Kurasami.S.et Al (2013) [9] provided the ability to automatically replicate previous data and significantly reduce the cost of ownership of petabyte scale data storage over alternative solution and also described the enhancement to hadoop through the HDFS as it stores file.

Nitin sawant, Himanshu shah [13] describes the big data application architecture for managing the information captured from markets to gain a competitive advantage when using traditional data analytical methods. It can be mainly applicable to field which need less expensive computing and storage power to analyze complex scenarios and models involving images, videos, text and other data.

J.Dean, et Al [5] described the performance evaluation using realistic workloads gives cluster operator new ways to identify workload specific choice on map reduce. Kyoung Ha Lee, et al[8] described a novel map reduce

runtime built using the Microsoft azure cloud infrastructure service proposed a model which increase the performance of the fully elastic model by comparing it with an semi elastic model.

Jinwoo lee, et Al [6] tries to solve big data problems so proposed the mechanism of the two-layer HDFS data prefetching. The experimental results show that the Hadoop platform which offers data prefetching mechanism can improve 60% of whole performance on data prefetching. Shaochun Wu, Guobing Zou [14] proposed resource allocation approaches to minimizing the mean end-to-end delay of customer jobs or services under the constraints of the energy consumption and the availability of MapReduce clusters. Kaiqi Xiong, Yuxiong He [10] presented the framework which insisted three Stages and seven Layers to divide Big Data application into modular blocks. The system provided mechanism which enable organizations to better manage and architect a very large Big Data application.

Anirban Mukherjee, Joydip Datta [1] described that big Data analytics requires internet scale scalability: over hundreds of compute nodes with attached storage by comparing VERITAS Cluster File System (SF-CFS) with Hadoop Distributed File System (HDFS) using popular Map-reduce benchmarks like Terasort, DFS-IO and Gridmix on top of Apache Hadoop.

Sivaram, et Al [16] describes the hadoop mapReduce paradigm for distributing a task across multiple nodes in hadoop and also big data evolution ,the future of big data based on gartners Hype cycle.

The following literature explain the advantages of using a Distributed cache.

K. Senthilkumar, K. Satheeshkumar [11] described Multi Intelligent caching is one such mechanism in which the cache distributed over redis servers and this redis server (single place to store all cached data) serves client request. This mechanism helps in improving the performance, reducing access latency and increasing the throughput. In order to enhance and improve the performance of MapReduce, the new design of HDFS is proposed by introducing multi intelligent caching concept with redis server.

Yaxiong Zhao, Jie Wu and Cong Liu [18] presented that, in Dache, tasks submit their intermediate results to the cache manager. A task queries the cache manager before executing the actual computing work. A novel cache description scheme and a cache request and reply protocols are designed.

Ganesh ananthanarayanan et al [3] presented a design of a proactive fetching and caching mechanism based on Memcached and integrate it with Hadoop, Hadoop with Memcached servers which would store meta-data about objects cached at datanodes. The blocks to be cached are decided on the basis of two level greedy approaches.

Wang jing, et al [17] proposed a hibernate cache technique which consists of primary cache, secondary cache, the query cache,etc., for different application. The framework used Ehcache to cache to cache the data the secondary cache and the query cache during the development of the application. The secondary cache can play a negative impact on the application if the query data quantity is relatively large.

Sang-Deok Yoon, et al [15] proposed a technique for calculating the I/O performance of file system in various distributed parallel clusters that can critically affect content-aware application performance. Sang also presented a new architecture for Local caching system which adjusts content locally between local file system and hadoop distributed file system by using proper cache algorithm and managing content queue , location and showed that local caching system would improve the performance of content-aware application sharply.

III. SYSTEM DESCRIPTION AND ARCHITECTURE

The proposed system is explained as below with fig 1 which presents the architecture.

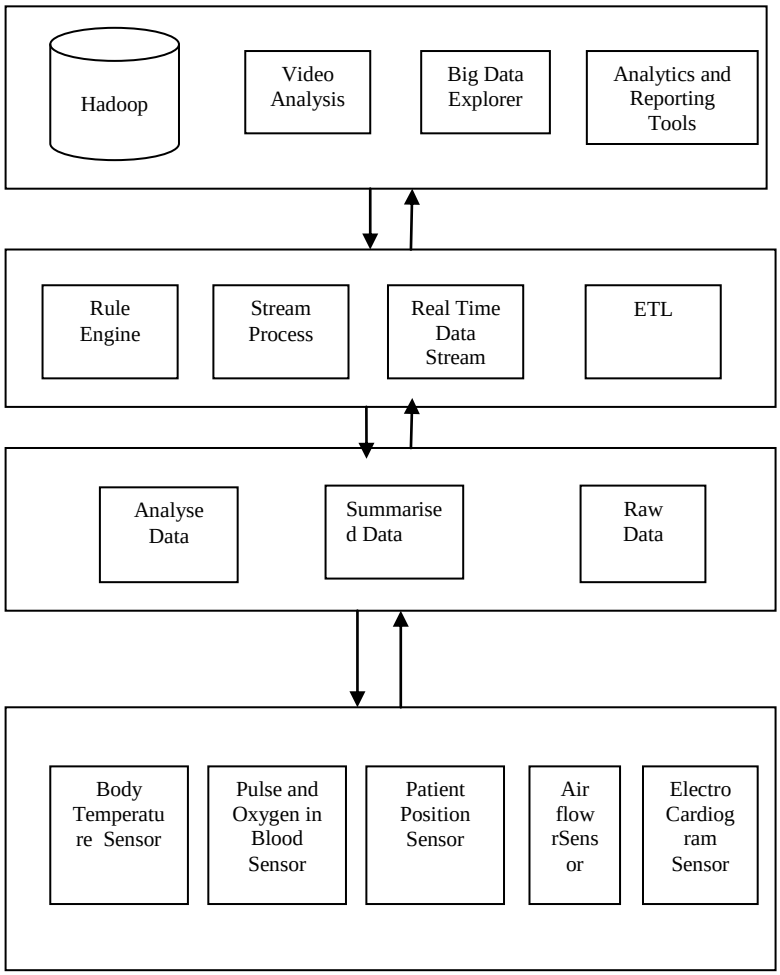

Fig 1 : System Architecture

The system works with IoT basically for smart access and collection of distributed data. The Big Data Analytics framework gathers and analyses the information from the data like Body temperature, pulse and oxygen in blood, patients position, airflow and electro cardiogram provided data. The process is explained as below.

Apache's hadoop is an efficient software tool to access and handle large data set. Hadoop follows the master/slave architecture decoupling system metadata and application data where metadata is stored on dedicated server Name Node and application data on Data Nodes. The Hadoop Distributed File System (HDFS) is a distributed file system designed to run on commodity hardware. It has many similarities with existing distributed file systems. However, the differences from other distributed file systems are significant. HDFS is highly fault-tolerant and is designed to be deployed on low-cost hardware. It provides high throughput access to application data and is suitable for applications that have large data sets. It is part of the Apache Hadoop Core project. Map Reduce works by breaking the processing into two phases: the map phase and the reduce phase. Each phase has key-value pairs as input and output, the types of which may be chosen by the programmer. The programmer also specifies two functions: the map function and the reduce function. The input to our map phase

is the raw data. We choose a text input format that gives us each line in the dataset as a text value. The key is the offset of the beginning of the line from the beginning of the file.

The purpose of the cache is to duplicate frequently accessed or important data in such a way that it can be recalled with the least effort and delay. The trade-off in distributed caching is network latency and serialization. Distributed memory object caching system is lightweight, scalable, and fast. It speeds up applications by alleviating database, file system and computational load. It saves the time and money by allowing using the existing server memory more efficiently. Do cache offers a simple plug-and-play customization options which allow to use own serialization and logging, allowing for over 70,000 adds per second, 100,000 bulk gets per second and 150,000 removes per second. The various challenges are Capture, storage, Search, sharing, transfer, analysis, visualization.

All the local caches can be coordinated by the distributed cache server .For application data, a distributed cache keeps a copy of a subset of the data in the database and it is also temporary in nature. It is work by divide and conquer algorithm, where problem is divided in to many sub problems. These sub problems are processed by Map Reduce framework. The data from the user are processed by Map Reduce. The input will first split into various key/value pair and each key/value pair will be assigned to each map. The intermediate result that produced as a result of map task is cached in both local node and in distributed cache server. The distributed cache server does not contain the actual data. Instead of that it contains only the metadata. Then the reducer will perform the Reduce task .Number of Reduce task will always lower than the map task. The metadata will be in the form of key/value pair. The Distributed cache server has faster retrieval and helps to access the cached item from any data node. Name node(master) is the central coordinator that coordinates all the data node(slaves). The name node updates the mapping of a cached block to the data node. Data node provides the report about the local cached block to the Name node and distributed cache server. Data nodes are responsible for data replication across multiple nodes. Keeping the unused data item for a long time will result in waste of resource. An eviction policy called ARC(adaptive replacement cache)is used to evict the unused data item from the cache.

IoT is proposed to be implemented using Arduino micro controller and Hadoop using Apache Hortonworks framework.

IV. CONCLUSION

Execution time is still an issue for delivering large amount of data . Existing hadoop system does not have any feature to reduce time for recompilation. a hadoop distributed system for large data processing, is proposed in this paper which has two type of cache memory one is local centralized cache, and other is distributed cache memory. It is proposed that using these cache memories considerable amount of recompilation time can be reduced.

REFERENCES

[1] Anirban Mukherjee, Joydip Datta, Raghavendra Jorapur, Ravi Singhvi, Saurav Haloi, Wasim Akram "Shared Disk Big Data Analytics with Apache Hadoop" on 2012 IEEE

[2] Avita Katal, Mohammad Wazid, R H Goudar "Big Data: Issues, Challenges, Tools and Good Practices" ©2013 IEEE.

[3] Demchenko.Y,de Laat.C, Membrey.P "Defining architecture components of the Big data Ecosystem" on Collabaration Technologies and System,2014 International conference,IEEE.

[4] Ganesh Ananthanarayanan, Ali Ghodsi, Andrew Wang, Drubha borthakur, Srikanth Kandula, Scott Shenker and Ion Stoica, "PACMAN: Coordinated Memory Caching For Parallel Jobs" In NSDI, 2012.

[5] Firat Tekiner1, John A. Keane "Big Data Framework" 2013 International Conference on Systems, Man, and Cybernetics, IEEE.

[6] Jinwoo Lee , SyKyoung Kim "Study for performance improvement of parallel process according to analysis of Hadoop" on Information Science and Service Science and Data Mining (ISSDM), 2012 6th International Conference 2012.

[7] J.Dean and S.Ghemawat," MapReduce: Simplified Data Processing on Large Clusters" Commun, of ACM, Vol.51, no.1, pp.107-113, 2008.

[8] Jieming Zhu ; Yu Kang ; Zibin Zheng ; Lyu, M.R 2013 "services generated big data and big data as a service:An overview" proceedings at the international congress on IEEE international conference.

[9] Kyong-Ha Lee, Hyunsik Choi, Bongki Moon "Parallel Data Processing with MapReduce: A Survey" on SIGMOD Record, December 2011 (Vol. 40, No. 4).

[10] Kurazumi.S; TsumuraT. ; Saito,S. ; Matsuo,H. "Dynamic processing slots scheduling for i/o intensive jobs for hadoop mapreduce job"on Networking and Computing (ICNC), 2012 Third International Conference on 2013.

[11] Kaiqi Xiong; Yuxiong He" Power-efficient resource allocation in MapReduce clusters" Integrated Network Management (IM 2013), 2013 IFIP/IEEE International Symposium.

[12] K.Senthil Kumar,K.Satheesh Kumar,S.Chandrasekaren,"Performance Enhancement of data Processing using Multiple Intelligent Cache in Hadoop", IJIET,Vol.4,Issue 1,June 2014.

[13] Meenakshi Shrivatava, Dr.Hans- peter Bischof,"Hadoop-Collaborative Caching in Real Time HDFS", Google, 2013.

[14] Nitin sawant, Himanshu shah, "Big Data Application Architecture ", pp no 9-28,2013.

[15] Shaochun Wu ; Guobing Zou ; Honghao Zhu ; Xiang Shuai ; Liang Chen ; Bofeng Zhang "The Dynamically Efficient Mechanism of HDFS Data Prefetching" Green Computing and Communications (GreenCom), 2013 IEEE and Internet of Things (iThings/CPSCom), IEEE International Conference on and IEEE Cyber, Physical and Social Computing on 2013.

[16] Sang-Deok Yoon, In-Yang jung,Ki-Hung Kim "Improving Hdfs performance using local caching system" on 2nd international conference,2013.

[17] Sivaraman.E, Manichachezian.R "High performance and fault tolerant Distributed File System for Big Data Storage and Processing Using Hadoop" Intelligent Computing Applications, International conference on 2014.

[18] Wing jin, Rui fan " The Research of hibernate cache technique and application of Ehcache component" IEEE 3rd international conference on 2011.

[19] Yaxiong Zhao, Jiu Wu, "Dache: A Data Aware Caching For Big Data Applications Using the MapReduce Framework", Vol.19, No 1, February 2014.