PERFORMANCE OPTIMIZATION OF HADOOP

Dissertation submitted in partial fulfilment of the requirements for the

Degree of

MASTER OF TECHNOLOGY

in

COMPUTER SCIENCE AND ENGINEERING

By

CHETAN SHARMA

11311389

Supervisor

Ms. Guneet Deol

School of Computer Science and Engineering

Lovely Professional University

Phagwara, Punjab (India)

November, 2017

ABSTRACT

With the increase in amount of data, need of a strong data storage system which can store large amount of data with an effective way arises. Hadoop architecture comes out to fulfil this need in which not only we can store large amount of data but we can process it also with the help of thousands of nodes working together. Basically, our research is based upon proposing a technique that can identify slow nodes into the cluster as a single node can reduce the overall performance of cluster.

DECLARATION

I hereby declare that the research work reported in the Dissertation II in partial fulfilment of the requirement for the award of Degree for Master of Technology in Computer Science and Engineering at Lovely Professional University, Phagwara, Punjab is an authentic work carried out under supervision of my research supervisor Ms. Guneet Deol. I have not submitted this work elsewhere for any degree or diploma.

I understand that the work presented herewith is in direct compliance with Lovely Professional University's Policy on plagiarism, intellectual property rights, and highest standards of moral and ethical conduct. Therefore, to the best of my knowledge, the content of this dissertation represents authentic and honest research effort conducted, in its entirety, by me. I am fully responsible for the contents of my dissertation work.

Chetan Sharma

11311389

CERTIFICATE

This is to certify that the work reported in the M.Tech Dissertation on Big Data (Hadoop Performance Optimization), submitted by Chetan Sharma at Lovely Professional University, Phagwara, India is a bonafide record of his original work carried out under my supervision. This work has not been submitted elsewhere for any other degree.

Signature of Supervisor
Name: Guneet Deol
Date: 30/11/2017

ACKNOWLEDGEMENT

Gratitude cannot be seen or expressed. It can only be felt in heart and is beyond description. Often, words are inadequate to serve as a model of expression of one’s feeling, specially the sense of indebtedness and gratitude to all those who help us in our duty.

It is of immense pleasure and profound privilege to express our gratitude and indebtedness along with sincere thanks to our mentor Ms. Guneet Deol, for her invaluable guidance, motivation and encouragement in spite of her busy schedule.

I am grateful to our Lovely Professional University for me with an opportunity to undertake this research topic in this university and providing all the facilities.

Finally, we would like to thank our parents and our family members for their constant support. We whole heartedly thank them all for their encouragement and support all the way from home from their hearts. We dedicate all our success to each one of them

TABLE OF CONTENT

TABLE OF FIGURES

CHAPTER 1
INTRODUCTION

DATA data everywhere, today in the silicon age where each and every machine is getting its electronic versions and everything is getting digital which leads to a new kind of explosion, DATA explosion.

International Data Corporation(IDC) Estimates the size of the "Digital Universe" at 0.18 Zetta Bytes in 2006 and forecasted a tenfold growth by 2020 to 40 ZB.

A research statement by Avendus Capital guesses that the IT market for big data in India floated around $1.15 billion as 2015. This contributed to one fifth of India's KPO marketplace worth $5.6 billion.

India has 600 information examination firms with 100 new companies setup in 2015 alone. Taking into account this request would require a vast supply of information researchers. As indicated by Team Lease Services - a staffing arrangements organization - by 2020, India will confront a request supply hole of 2,00,00 information investigation experts. The supply is short even in the US work showcase with just 40 out of 100 positions for information researchers being filled. (NDTV.com)

From where this data is coming from

- Individuals transfer recordings, take pictures on their phones, content companions, refresh their Facebook status, leave remarks around the web, tap on advertisements etc.
- Machines, too, are producing and possessing this increasing data.

This flood of data is coming from many sources. Consider the following:

- The New York Stock Exchange produces nearby one terabyte of novel trade data per day.
- Facebook hosts roughly 10 billion photos, taking up one petabyte of storage.
- More important, the measure of information produced by machines will be considerably more prominent than that created by People. Ex: Machine logs, sensor networks.
- Facebook: over 500TB of data per day.
- eBay: over more than 500TB of data per day.

1.1 How Is it becoming a problem?

Consider an amount of 1GB of data that is to be processed. The data are stored in a relational database in your PC and PC has no problem handling this load, but then the respective organization starts growing very swiftly, and that data grows to 10GB, and then 100GB. And you start to reach the limits of your current desktop computer. So, you scale-up by investing in a larger computer, and you are then OK for a few more months. When your data grows to 10TB, and then 100TB.And you are fast approaching the boundaries of that computer. Furthermore, now we have to nourish system with unstructured information coming from sources like Facebook, Twitter, RFID readers, sensors, and so on. We need to originate information mutually from the relational data and the unstructured data, and wants this information as soon as possible.

1.2 Failure of Traditional Large-Scale System!!!!!!!!!

Traditionally, computation is processor bond. For decades, the main push was to surge the computing ability of a single node i.e. eager processor, more RAM But it can be done in a limit only and it costs much also

Let us take an example of conventional computation to process a file of 1 TB approx. 1000 GB stored in a SATA hard disk with a speed of 100MBpS with 4 I/O chunks.

So, it will access data as:
100 MB in 1 sec

1000 MB => 1 GB (approx.) in 10 secs

Therefore, 250 GB in 2500 sec. s

Because of 4 I/O chunks it will access 250*4=1000 GB i.e. 1 TB in 2500 secs approx. 41 minutes. Now imagine the scenario in which we have to process PB'S of data with conventional computing systems and this just for the access of the data now imagine to processing of data will take time

Now consider same scenario with Big Data architecture.
Just split the data to different nodes which takes 100 MB of data as each on the same commodity hardware.
So, the current Equation will be:
Time will reduce to 10 times i.e. 2500/10 =25 secs each node and now imagine the scenario where millions of nodes are working together.

1.3 Problem of Dirty Data

Most of the real-world data sets starts off dirty because in the increase in technology which leads towards lot of dirty and uncertain data. The various fields where we get dirty data are, Finance, Healthcare, retail, hospitality and even in education we receive a large dirty data. So by cleaning dirty data we can predict the weather or forecast the Sensex in business in order to gain profits and accurate predictions in every above case. As the data get bigger than the number of things goes wrong is also increase. So, it become harder to look on entire data to find and remove duplicity from the data. To Increase accuracy of uncertain data we need to clean the data that is process only those data which is necessary to increase the response time and decrease the latency of the database. Data cleaning is an iterating process in which first we need to collect the data and then. After detect the data which is need to be clean the data may be invariant, incomplete, inconsistent, sparse data, duplicate fields, missing data, and like JSON (Java Script Object Notation) documents field might not have the structured fields we expected, or we Might have number or data which is out of range for example the age can’t not be negative (age is always positive and maximum it would be 100 in case of humans) etc. The case with missing data may contains valuable information for the others variable so ignoring these cases means loss of information which leads to decrease the variance of the data. So, we need to clean data in such a way that the important information cannot be lost cleaning the data we need to analyse the data and the present report on the data. So, to find duplicate data we use database queries or in big data we use mapper to find duplicate entries. So, when we are done with mapping we need to clean the duplicate data. To clean that data, we have normalizations in database and in big data we have Reducer to reduce duplicate data. Map reducer performs two different tasks. The first is Map and second is Reduce. Map takes the input data set and processes it to produce the key value pairs and then Reducer comes into work it takes the key value pairs which is the output of Map and the Reducer job is to take those key value pairs and then combines or aggregate them to produce the final result. The Reducer performs its job only when Map is done with its job. This helps to optimize the data and provide only the data as output which is accurate, consistent and beneficial for the prediction. Then after cleaning the data we need to analyse the data that what type of data is needed or what is not. If your analysis is not correct then there will be no mean of cleaning data because we clean data to predict the correct output so wrong analysis leads to wrong decision as well as wrong predictions which makes your business in loss. So, when you are done with correct analysis then you can proceed towards the next step that making report on the analysis and then you can predict your business easily and accurately. So, because of data cleaning user can take decision seriously on the bases of data, with logic as well as with analysis and with this user can easily set goal for small success.

1.4 Problem with Distributed Systems

- Programming for conventional distributed systems is little bit hectic.
- Finite bandwidth is available.
- Data exchange requires synchronization.
- It is tough to pact with fractional failures of the system.

Normally, information for a distributed framework is put away on a SAN. At processing time, information is replicated to the process hubs Fine for generally constrained measures of information, Moreover There is no data locality concept in SAN every time you need to bring the data through n/w to compute node to process.

The conventional information handling model has information stored in 'storage cluster', information is duplicated over to different groups of nodes for processing and outcomes are composed back to previous locations. This model doesn't exactly suitable for Big Data. Since replicating out such a great amount of information out to process group may be excessively tedious or unimaginable. So, what is the appropriate response?

1.5 What is the Solution for this problem??

Need leads to new invention. In today’s Era, more than 80% of new inventions are to satisfy the business needs like virtualization, Cloud computing as so Fourth **BIG DATA**

1.5.1 Hadoop: Originally, it was developed by Doug Cutting, who was known for development of Apache Lucene and Apache Nutch and after some time, in 2003 Google launched a paper on Google File System as a solution to large amount of data and in 2004 a paper on MapReduce. After that, near about 2006 Cutting joined Yahoo and dedicated himself in the development of Hadoop, meanwhile due to popularity and success of Hadoop other companies like Facebook, Yahoo etc. also started using Hadoop. Yahoo made an cluster of node with a huge number of 10,000. Due to interconnectivity of large number of node into a single cluster, a new problem arises i.e. straggling nodes.

1.6 Straggler: When a node is done with more than 95% of work, suddenly its performance started taking a dip, due to which performance of whole cluster gets affected. Nodes whose performance deteriorates is called straggler node.

1.6.1 Solution: One possible solution of this problem is to detect the slow processes and identify the straggler and try to reschedule the remaining process on to some another node.

CHAPTER 2

SCOPE OF THE STUDY

Main idea behind this research work, is to find out the slow processes from the cluster and later identify the straggler from the cluster and to pass the outcome to suitable technique which can reschedule it to some another node in order to raise the whole reliability of the cluster.

In this work, a brief discussion of each present technique to identify straggler will be done and their advantages, scope, approach followed and limitation will be discussed. A new approach will be proposed and a practical implementation will be performed and results will be analyzed.

At last results of proposed work and previous techniques will be compared and it will be concluded whether it is an improvement over previous technique or not and if so up to how much extent. Future work and scope of further study will also be taken into account.

CHAPTER 3

OBJECTIVE

The ultimate aim is to Optimize the performance of a Hadoop cluster as an end outcome while there are some objective that have to be achieved in order to do so. These objectives are:

- To spot the slow processes due to which overall performance of the cluster is getting affected.
- To identify the Straggler node from the cluster.
- Developing an algorithm which are capable of implementing upper two objectives with better performance than the predecessor algorithms.

CHAPTER 4

LITERATURE SURVEY

1. **Understandable Big Data: A survey**

   Authors: Cheikh Kacfah Emani, Nadine Cullot, Christophe Nicolle

   Description: In this paper, a complete definition of big data with characteristics is given and it also explained about a common myth regarding the Big Data that how a layman with a little knowledge in bigdata takes Size of the Data i.e. Volume as the only parameter for the characterization of Big Data and Explains other characteristics(5 V's) like Data in motion(Velocity), Data in Highlight(Value),Data in Doubt(Veracity) and Data in many Forms(Variety).By looking into these 5 V's the objective of Big data seems to be to extract the hidden meanings or patterns from the huge chunk of large variety of information by allowing its high velocity. It also explains how virtue of Management of data is changed.

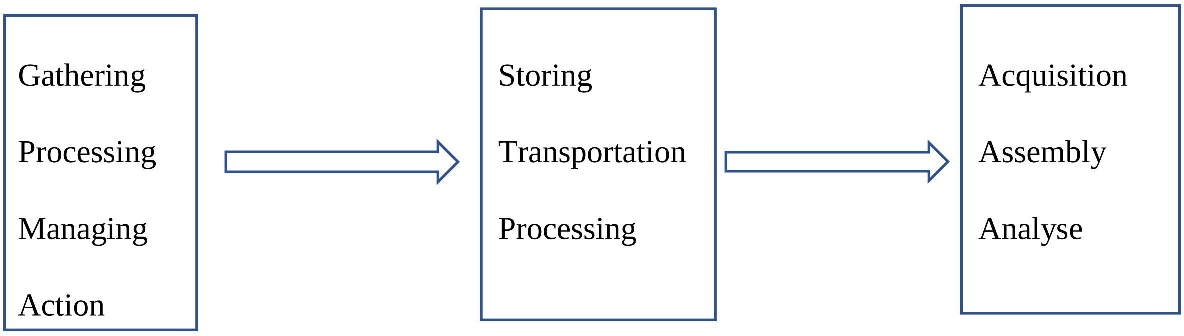

Fig. 2 Big Data Management

This tells that to handle the Big data an Infrastructure must be linear scalable, capable to handle data with different formats with high throughput, tolerable to faults, self-recoverable, with a high degree of parallelism and processing of distributed data.

It also Explains the Visualization techniques like Tag cloud (various Text Formats such as font size, colour.), History Flow (evolution of a documentation of particular topic by different authors. The horizontal axis is for time and the vertical axis for the names of the authors)

It also tells the seeking of the company from Big Data as Customer's Digital Footprint on the Digital world and other is data generated by the machines.

Terms used:

- Names Entity Resolution: Extraction of events from the Different Digital foot prints of the customers using various markers like #, @ etc.
- Coreference resolution: concluding expressions w.r.t the similar entity.
- Ontology: Fitting the data into a particular format by getting the data processed through some tools.

  Ontology matching is big challenge in the domain of big data which possesses challenges like matcher of selection ,combination and tuner, user involvement.

2. **MapReduce: Simplified Data Processing on Large Clusters**

   Authors: Jeffrey Dean and Sanjay Ghemawat

   Description: In this paper, both authors working in Google while writing the paper, explains about how MapReduce works inside Google and its different functionality. According to them," MapReduce is a technically iterative programming model and an associated execution for information handling or processing and producing large dataset results that might be proved useful for real world use"

   Factors that leads to the development of MapReduce: Messy Parallelization Architecture, Fault Tolerable, Distribution of Data and Load Balancing.

   Terms Used:

- **Map**: It is a user written Algorithm which takes an input pair and gives a set of in*termediate* key/value pairs as an output. The MapReduce library does the grouping of all intermediate values together having equal intermediate key *I* and reduce function fetches these groups for further processing.
- **Reduce**: It is an user written Algorithm takes an intermediate key *I* and a set of values for that key. It merges these values together to produce a reduced set of output. Typically, binary output value has been given out per reduce invocation.
- **JobTracker**: The JobTracker always present on 'master node'. Submission of MapReduce jobs to the JobTracker is done by client and the JobTracker further allots Map and Reduce tasks to other different nodes on the rack.
- **TaskTracker: S**oftware daemon named TaskTracker responsible for running on nodes. The TaskTracker is actually instantiate the Map or Reduce task, and reports back the progress to the JobTracker.

- **Straggler**: A situation when node is at verge of completing its task including backup and the speed drops suddenly which lengthens the time of execution. In this case, Master node assigns rest of the task to the other idle node and current node is assigned with another task.

Hardware Used: Dual-Processor X86

- Linux as OS
- 4-8GB of memory/ machine.
- 1 gigabit/second of network bandwidth,
  A Computing Cluster contains 1000's if Machines

Processing Mechanisms: Map invocation is splited into M pieces among Distributed multiple machines and similarly Reduce invocation is distributed in R pieces and all works parallel to each other.

i. Program first ruptures the input files into *M* pieces of classically 16-64MB per piece.
ii. Using replica of the program, the master assigns the tasks to M map tasks and *R* reduce tasks to allocate. Sluggish nodes are chosen by master and jobs are allocated.
iii. A worker analyses the matter of the consistent input split to parses key/value pairs and the intermediate key/value pairs formed by the map function are buffered in memory.
iv. Locations of these resultant pairs thrown back to the master with respect to local disk who again forward to the reduce workers.
v. Grouping of intermediate keys with same keys are done by reducer via sorting. The sorted arrangement is required in light of the fact that regularly a wide range of keys map to the same reduce task. In the event that the measure of transitional information is too vast to fit in memory, an outer sort is utilized.
vi. Reducer repeats among all the intermediate value and an exclusive intermediate value is bump into and final output is printed back to the final output file for the reduce partition
vii. After completion masters revokes the MapReduce call to return back to the user code

Fault Tolerance: The master revokes each labourer occasionally. On the off chance that no reaction is gotten from a labourer in a specific measure of time, the master denotes the worker as fizzled. Any map work asks finished by the specialist are retuned back to their underlying inactive state and in this way, end up plainly qualified for booking on different laborers. So also, any map work or reduce task in advance on a fizzled specialist is likewise reset to sit out of gear and ends up noticeably qualified for rearranged. Finished map assignments are re-executed on a disappointment on the grounds that their yield is put away on the nearby disk(s) of the fizzled machine and is consequently out of reach. Finished task don't should be re-executed since their yield is put away in a worldwide document framework. There are certain refinements majorly based upon user custom requests. This Architecture is easy to use because it abstracts the details of parallelization, fault tolerance, locality optimization, and load balancing and it is very useful for accommodating large clusters having 1000 of machines.

3. **The Hadoop Distributed File System**

Authors: Konstantin Shvachko, Hairong Kuang, Sanjay Radia, Robert Chansler

Description: In this paper, both authors working in Google while writing the paper, explains about how HDFS works inside Yahoo and its different functionality. According to them, "The Hadoop Distributed File System (HDFS) is intended to store very great data sets steadfastly, and to stream those data sets at high bandwidth to user applications**.**"

The most basic part of HDFS is splitting of data into different data blocks and then distributing it to many parallel running nodes and scaling the computation capacity, storage capacity and I/O bandwidth simply through commodity Hardware.

Current Hadoop Cluster Configuration of Yahoo!:

- A span of 25000 servers
- 25 PB of Data in storage
- Largest cluster having 3500 servers

HDFS Architecture: It stores file system meta data and user data onto two different location and rather than using RAID's for storage it just replicates the data onto multiple data nodes for reliability.

Term Used:

- NameNode: These node is the master node which stores the meta data like permissions modifications, access times etc. Clients request first goes to name node to access the location of data blocks and similarly for writing data.
- DataNode: These are the slave nodes and their function is just to store data into it and replying to the NameNode onto a ping via handshake mechanisms.
- Image: The data of InNode and the list of Data Blocks of each file comprising meta data is called Image.
- Checkpoint: Persistent record of image stored on the local file system is called the checkpoint.
- Journal: Modification log generated and stored inside the name node is called Journal

- HandShake: It is process executed during establishment of connection between name node and data node to verify ID and software version of DataNode.
- SnapShot: It is created immediately before the software upgradation so that any bug comes during upgradation can be removed by simply rolling back to the previous state.

File I/O Operations: After a user gets a lease on particular file to write data on it no other write from any other user will be entertained while concurrent multiple reads can be performed on a file up to last commit performed. No data manipulation can be done once the file is closed by user or hard limit expired.

HDFS uses a google replica placement policy for creating replicas because smaller the distance between to replicas greater utilization of bandwidth will be and the basic policy is:

- No Data node must not have more than single replica of any data block.
- No rack must not have more than two replicas of the same data block, taking care of sufficient amount of racks on the cluster.

A balancer is accommodated to keep a track that data must be placed uniformly on a datanode without violating the replication policies.

Limitations

- Unavailability of Hadoop cluster when NameNode is Down.
- Scalability of NameNode.
- Size of NameNode limits the addressing Scheme
- NameNode becomes Unresponsive during maximum memory usage
- Sharing of multiple namespaces, a single cluster is under study due to high cost.

4. **The rise of “big data” on cloud computing: Review and open research issues**

Authors: Ibrahim Abaker Targio Hashem, Ibrar Yaqoob, Nor Badrul Anuar, Salimah Mokhtar, Abdullah Gani, Samee Ullah Khan

Description: In this paper, Authors have discussed about Cloud Computing , Big data and their different aspects which relates them with each other along with their classification based upon several parameters. Authors also discussed about the different present cloud platforms and gave the comparison between them and similar comparison for big data initiative by different companies. Authors have studied about the different case studies of the companies which will help to understand the scope and importance of the Big data and cloud computing.

Cloud Computing: A robust tool for complex and large computations which gave significance of virtual resources, parallel processing, security, and scalable data storage is termed as Cloud computing.

Causes behind cloud computing:

- Lack of Hardware
- To reduce Capex
- Massive increase in data

Cloud computing: This is a technical model for allowing ubiquitous, easily available and dynamic network excess to number of configured sources that can be quickly provisioned and released with minimum management efforts.

Core business can be focused without worrying about infrastructure, flexibility and resource availability.

Bigdata: Although, there is no defined and clear definition is present for the big data but it can be characterized as the "another age of advances and models, intended to financially extricate an incentive from vast volumes of a wide assortment of information, by empowering the high speed catch, revelation, as well as examination" in simple words it is the amount of data just beyond traditional technology's capability to store, manage, and process efficiently

Causes behind the rise of Big data are:

- Numerous amount of data
- Relation databases are not able to manage it
- Rapid generation and processing of data

Big data classification depends upon five aspects

- Data sources: Web and social, machine, sensor, transactions, IOT
- Content format: Structured, semi-structured, unstructured
- Data stores: Document oriented, column oriented, graph based, key value
- Data staging: cleaning, normalization, transform
- Data processing: Batch, real time

Relation among Big Data and Cloud Computing: User can access mundane computing to process distributed queries with multiple data sets under given time through big data while cloud computing provides computation facilities and service models. However, non-data availability can be expensive because of wrong and costly decisions. Studies are going on

- Data acquisition
- Real time intrusion detection system
- Big data management analysis
- To encounter smarter fraud
- Uploading data into cloud from different geographical locations

Some research project case studies on big data:

- **Swift key**: It gives personalized predictions and corrections for text screen typing for which company collects and analysed terabytes of data for many users. It uses AS3 and AEC2 to manage multiple TB of data
- **343 Industries**: For creation of halo game developer analyses player preferences and online tournaments using windows azure HD in site service depends on apache Hadoop big data framework.
- **Redbus** : To manage tens of thousands of bus schedules and ten thousand routes, it uses google query, big query and google data processing infrastructure.

HDFS: It has two types of nodes, name node called master node that registers attributes like access time, modification, permission and disc space quota while data node called slave have data blocks.

Map reduce by google is a simplified programming model for processing big amount of data assets.

Hive Hbase Mahout pig zoo keeper are similar technologies.

CHALLENGES

- **Scalability**: scaling of infrastructure with increasing of data
- **Availability** : 24x7 is resource acquisition availability based on authorisation
- **Data integrity** : authorize person can only access the data
- **Transformation** : Data must be processed in such a way that it can be analysed
- **Data Quality** : multiple data sources with multiple data varieties decreases a quality of data. Data cleaning ensures the data quality.

- **Heterogeneity**: Variety of data sources leads to heterogenous data. Cloud gives facility to store data in multiple formats.
- **Privacy** : It is one of the serious issues of big data to secure the information. Incryption algorithm are used for privacy.
- **Legal/regulatory issues** : Diverse countries have dissimilar laws regarding cloud and big data. Specific law and regulations must be established.
- **Governance**: Data governance bodies controls the transparencies, accountability and laws.

Rain Cloud**:** Involved collaboration between single clouds to provide nearby resources in a need.

Data present is in a big amount and increasing exponentially. Cloud and big data can mutually resolve this issue with clouds serves a service models.

5. **A Survey on Big Data Analytics: Challenges, Open Research Issues and Tools**

Authors: Debi Prasanna Acharjya, Kauser Ahmed P

Description: Modern Information system and digital technologies like IOT, cloud computing generates huge repositories of data. In this paper, Authors have discussed about the possible impact of big data contests, open research issues, and various tools associated with it. Big data is building upon Third platform industry which mostly denoting to big data, cloud computing, IOT, and community commercials and is robust impetus to next generation of industry.

Data Warehouse are used to store data of large data sets and data mining is used to mine the hidden patterns out of it which proves that the complex concept of big data will lead to understanding essential physiognomies and creation of intricate formats in big data and knowledge abstraction

Challenges in Big Data

- **Data Storage and Analysis:** Information openness must be on the best need; notwithstanding, existing calculations may not generally react in a satisfactory time when managing these high dimensional information. Late advancements, for example, Hadoop and MapReduce make gathers vast measure of semi organized and unstructured information in a sensible measure of time
- **Knowledge Discovery and Computational Complexities:** Accessible tools may not be good to progression these high end data for obtaining meaningful information and handle

inconsistencies and additionally It might be hard to set up an exhaustive numerical framework that is comprehensively appropriate to Big Data.

- **Scalability and Visualization of Data :** The goal of envisioning information is to display them all the more satisfactorily utilizing a few procedures of diagram hypothesis, However, current enormous information perception devices generally have poor exhibitions in functionalities, versatility, and reaction in time.
- **Information Security :** Different safety efforts that huge information applications confront are size of system, wide range of gadgets, ongoing security checking, and absence of interruption framework The real test is to build up a multi-level security, protection safeguarded information demonstrate for huge information.

OPEN RESEARCH ISSUES IN BIG DATA ANALYTICS :

- **IoT for Big Data Analytics :** Much the same as the web, Internet of Things empowers the gadgets to exist in a horde of spots and encourages applications extending from minor to the critical. IoT gadget creates persistent floods of information and the analysts can create devices to separate significant data from these information utilizing machine learning procedures. IoT Knowledge Exploration System can be spoken to as:

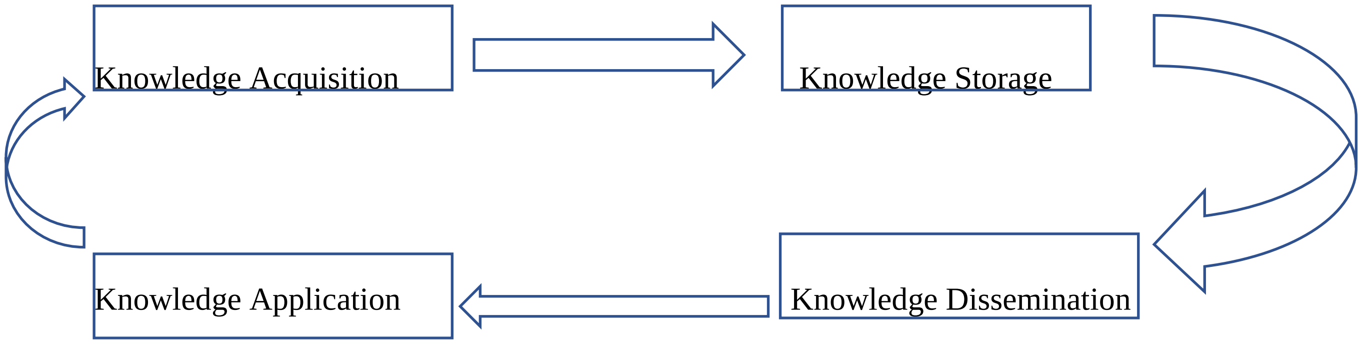

Fig. 3 IOT Knowledge Exploration System

- **Cloud Computing for Big Data Analytics:** The utilization of virtual PCs is known as distributed computing which has been a standout amongst the most strong enormous information strategy. Enormous information application utilizing distributed computing should bolster information scientific and improvement. distributed computing should likewise empower scaling of devices from virtual advancements into new advances like start, R, and different sorts of enormous information handling methods.

TOOLS FOR BIG DATA PROCESSING:

- **Apache Hadoop and MapReduce:** The most settled programming stage for enormous information investigation is Apache Hadoop and MapReduce. It comprises of Hadoop bit, MapReduce, Hadoop conveyed document framework (HDFS).
- **Apache Mahout :** Apache mahout expects to give versatile and business machine learning methods for huge scale and insightful information examination applications. Center calculations of mahout including grouping, order, design mining, relapse, dimensionality decrease, transformative calculations, and clump construct community oriented sifting keep running with respect to best of Hadoop stage through guide lessen system.
- **Apache Spark :** Apache stark is an open source enormous information preparing structure worked for speed handling, and refined examination. Notwithstanding map decrease operations, it bolsters SQL questions, spilling information, machine learning, and diagram information preparing. Configuring these issues cannot be done by a layman user but a scientific study needs to be done as big data's potential is still in under discovery.

6. Mobile Healthcare Information Management utilizing Cloud Computing and Android OS

Authors: Charalampos Doukas Thomas Pliakas, Ilias Maglogiannis

Description: In this paper, author has explained a successful implementation of Mobile health care system using a cloud service and an Android System enabling e-healthcare data storage, updation and retrieval.

Main goals of Mobile Health care system are

- The accessibility of e-health applications and medical history and knowledge anywhere
- Invisibility of computing

Mobile Healthcare System Services are

- Location-based medical services
- Emergency response management
- Personalized monitoring and pervasive access

Why Cloud Computing?

From several studies, it has been that the constrained access to quiet related data amid basic leadership and the insufficient correspondence among tolerant care colleagues are proximal reasons for medicinal blunders in human services. By using Cloud Computing we will provide the basic features of cloud in health care like On-demand self-service, Broad network access, Resource pooling, Rapid elasticity etc.

This work is majorly done be seeking a need of transferring medical data to the health care system for better evaluation but not for the data management and interoperability issues which will be handled by cloud mostly.

Why Google Android OS?

The main reason behind this is that the stage is versatile to bigger and conventional advanced mobile phone designs. It bolsters an awesome assortment of sound, video and still picture arrange, making it appropriate for showing therapeutic substance.

Why Amazon S3 Cloud Services?

The principle purpose behind choosing the particular Cloud Computing stage is that it is a business benefit entrenched and utilized effectively in a few applications.

Features of Health Care System are

- Image viewing support
- Appropriate user verification and data encryption

Medical Data of an individual is indeed very much private and handling such a data is a critical thing, although author have done some steps for such scenarios but still it needs some more work to be done.

7. **Can Sensors Collect Big Data? An Energy Efficient Big Data Gathering Algorithm for WSN**

Authors: Shalli Rani, Syed Hassan Ahmed, Rajneesh Talwar, Jyoteesh Malhotra

Description: There two main sources of data which leads to development of Big Data:

- Digital footprints by people
- Machines and Sensors

So, one of the key providers for big data that can make the significant amount of data is Distributed Wireless Sensors Network. In this paper, many energy efficient Big Data collection algorithms for same technology are explained for data collection purpose. Although, for machines it is very much important to collect their data at real time, but it is also challenge among data scientists that how to spread the real time information to the data centre after its flock from the situation. There are many Energy efficient algorithm are introduced like LEACH, PEGASIS etc.

Objective is to lower down the distance between to nodes as energy exhaustion is directly proportional to the travelled path.

Solution is that data must be transmitted by means of transfer hubs chose in the course from source hub to the BS or by means of some most brief way directing calculation. Here WSN machines are in static state and no Environmental factor is disturbing them.

Clustering Problem:

- How to distribute the nodes into clusters
- What is the best number of clusters

A transmission calculation to shape the groups by basic recipe and to encourage the multi-jump directing by means of adjusting load on the hubs when they will play out the part of hand-off hubs is use to diminish the transmitting separation of the most distant hubs.

Architecture: The separation of the hubs is registered by limitation techniques in light of RSSI. CHs stores the information, until the point that they get any flag from BS to transmit the information. The information is transmitted in multi-jump method. In proposed procedure, we accept the system sent over thick and vast zone, for example, for fringe reconnaissance, condition observing. The quantity of hubs in each group is arbitrary and even after irregular dissemination each bunch has 99% same number of hubs. RNs are processed with the assistance of RSSI and area of the hubs.

The transmission process takes place as:

i. Data is gathered at the CHs at customary interims.

ii. Data is transmitted to the BS, when some adjustment in the past information is seen (else it is disposed of.

iii. In case the cushion space scopes to its limit level, at that point an examination is made at that phase to discover the adjustment in information and choice to transmit or dispose of the information is taken.

iv. If whenever the level of vitality of CHs or CCOs is discovered not as much as the edge, at that point their decision is again encouraged by the calling the CH or CCO race calculation

In this paper, Experiment is performed on expansive number on hub and is finished utilizing MATLAB. Add up to separate in multi-jump correspondence is higher than single bounce information transmission. Future work will focus on lessening the aggregate separation for information transmission and will consider alternate parameters like flag to clamor proportion and bit mistake rate.

8. **Data Cleaning and Query Answering with Matching Dependencies and Matching Functions**

   Authors: Leopoldo Bertossi, Solmaz Kolahi, Laks V.S. Lakshmanan

   Description: This paper explains about the Matching Dependencies, which are the explanatory principles for information cleaning and element determination, Matching Dependencies are utilized to definitively indicate the distinguishing proof (or coordinating) of certain quality esteems in sets of database tuples when some similitude conditions on different esteems are fulfilled. This paper also explains about the clean query answering which means it improves the data that as of now existed in the grimy occurrence and made it more informatics, critical and reliable. It is done by taking the best lower bound (glb) and minimum upper bound (lub) of answers of the question over numerous spotless examples, rather than taking the set-theoretic intersection and union. In this paper, various notations like clean answer query notation and monotonic query notation are explained. clean response to question postured to the grimy database was defined as a couple shaped by a lower and an upper bound regarding data content for the inquiry answers. In this unique circumstance, I examined the thought of monotone inquiry w.r.t. the mastery request and how to unwind a question into a monotone one that gives more educational answer than the first one

---

9. **Answering Queries using Humans, Algorithms and Databases**

   Authors: Aditya Parameswaran, Neoklis Polyzotis

   Description: In this paper, researcher approach view is Crowed Sourcing Services as alternative database where results are calculated by human calculators, here, the use of human knowledge so as to understand assignments that are generally straightforward for humans– distinguishing and perceiving ideas in pictures, dialect or discourse, positioning, rundown and naming, to name a few– yet are famously difficult for calculations. Freebase.com has gathered more than 2 million human reactions for assignments identified with information mining, purifying and curation. Several start-up's, such as Crowd Flower, u-Test and Micro task, have built up a plan of action on task administration for big business. The exploration group has created libraries that give primitives to make and oversee human calculation assignments and in this manner empower programmable access to swarm sourcing administrations. In this, cloud sourcing services i.e. PaaS, h-query, 1) fuzzy image map 2) aggregation 3) sorting + selection. In this we use re-captcha to distinguish the images and cities and blur texts by getting majority answers, so to do this we involve humans, this involvement of humans helps to answering the queries or prediction and the answer is selected by getting majority answers another approach for answering the query is Algorithm, in this we have fixed algorithms or sequence of commands which helps you to give desired output from the given data and this leads to certain answers only. This also consider as have built up a plan of action on task administration for big business. The exploration group has created libraries that give primitives to make and oversee human calculation assignments and in this manner, empower programmable access to swarm sourcing administrations.

10. Data Cleaning: Overview and Emerging Challenges:

Authors: Xu Chu, Ihab F. Ilyas Sanjay Krishnan, Jiannan Wang

Description: This paper explains about the challenges in data analytics, as well as the present scientific categorization of the information cleaning writing in which they feature the current enthusiasm for procedures that utilization requirements, principles, or examples to identify mistakes, which we call subjective information cleaning with this they also explain about the limitations with illustrative examples. First, they explain about the qualitative error detection techniques that is "What to Detect", "How to Detect" and "Where to Detect" In What to Detect the researcher of the paper explains about that what type of error to be captured or what type of language is used to describe the constraints of legal data instance. The type of error contains the duplicate records because the duplicate records are considered as the violation of integrity of the data.

11. Outsourcing to the cloud: data security and privacy risks

Authors: Peter Brudenall, Bridget Treacy and Purdey Castle

Description: In this paper, I have read about the outsourcing of data into cloud computing architecture and how different vendors provide this service to their customers. This paper also explains about the different European Union Laws and Key points that a customer must ensure while outsourcing its data to cloud. It explains about how correct information about the kind of data an organization possess can help it to choose an appropriate model, although according to EU Data Protection Law, client have a full right to Access, Store or Modify data in any condition and anytime with consideration to security of data. However, organisation doesn't pay their attention to it but most of the time it becomes very crucial. Large International Vendors are more concerned and trust worthy in this scenario than the Local ones. According to EU Laws, an organization is fully responsible for the data of their customers not the cloud vendor and in case something happens to the client's data, client can take action against organization. While migration of data to cloud, it becomes more vulnerable so EU Laws tells that an organization cannot store it's client's data outside the EEA except from certain conditions. Data must be available to client 24*7 but some of the vendors hesitates to give an assured guarantee for this.

Some essential steps for opting Cloud:

- Be assured that your organization with current type of data need a Cloud Architecture.
- Examine the Safeguards to ensure the cloud vendors approach to serve your data.
- Study the proper Laws, regulation and related authorities to prevent any legal threats.
- Proper agreements must be there for ensuring rights and security standard.
- Encryption Technique while transferring data to cloud.
- Seeking control over the location of data inside cloud and what are the disaster management techniques of the vendor

12. Research on the security technology of big data information

Authors: Hong Zhu**,** Zheng Xu, Yingzhen Huang

Description: In this paper, I have studied about the various security concerns that raising these days to dawn of a new technology called big data. Because to Big Data technology is capable enough to find the hidden meanings out of the big amount of raw data available from different sources. It enables to forecast and monitored the different nodes independent of the fact that it is a living entity or not and then present it in the form of visual charts to show the inherent relationships. This gave rise to huge security risk as large amount of the data becomes vulnerable to attack. In this era, when data is wealth, big data is also give rise to big security risks. It has been discussed that how an equipment doesn't matter computer hardware or not can be vulnerable and how its safety is related to data safety. As commodity hardware is used which is in large scale i.e. large number of controllers and hard disk or network related material which makes it risky to be compromised not only to the outside attacks but also to the system failures and other inner attacks so a proper backup and redundancy techniques must be there. Prevention of electromagnetic information leakage is a new challenge to it. The basic concerns are:

- Highly scaled architecture is more vulnerable.
- Real Time security is a major concern.
- Wealth of data attracts countless hackers.

Steps that can be taken:

- Tradition security Tech must be improved.
- Proper criterion for key management and authentication process.
- Establishment of reliable protocols.
- One-way data control flow.
- System must be intelligent to detect compromised modules.
- Photoelectric magnetic field shielding technology.

Availability will definitely lead to vulnerability same as more demand of data will lead to more security concerns.

13. Survey of Research on Information Security in Big Data

Authors: Zhang Hongjun, Hao Wenning, He Dengchao, Mao Yuxing

Description: In this paper, I have read about the Big Data Information Security, their conclusions and different technological solutions to it. In this era, due data analytics and data science our vendors know our habits like google know what we are searching, Facebook know what we are doing. Big data is not only cause of personal leaks but also to the sensitive data stored on to it. I also read about the different domains and their respective loop holes to threat like in phase of Data Acquisition data become more prone to attacks as data is in huge amount so the chance of data leakage, tampering and forgery. In-spite of strict access control and privacy management there is still some data leakage in technology like NoSQL. Techniques like data mining are not hampering personal privacy of user but also there is a threat of hidden malwares or Trojans. Any failure to handle data may lead to large data leakage. Causes behind this are: Lack of Standard Regulation Authorities, Lack of standard and reliable authentications for cloud and its computing service. To avoid any security breach techniques like network separation, access control, authentication, encryption.

14. Hive – A Petabyte Scale Data Warehouse Using Hadoop

Authors: Ashish Thusoo, Joydeep Sen Sarma, Namit Jain, Zheng Shao, Prasad Chakka, Ning Zhang, Suresh Antony, Hao Liu and Raghotham Murthy

Description: In this paper, I have studied about the Hive which is an open source Hadoop based program, actually it is just an amendment to the Hadoop i.e. limitation of Hadoop are avoided in Hive. Main problems are that Hadoop is not much user friendly as user have to write its own map-reduce programs and it also does not support the conventional SQL like languages. Hive structures the data as according to the conventional concepts like tables, rows, column and also new complex concepts like map, struct, list are included i.e. why it is very close to the SQL. Data can be stored to the HiveQL as it is without any transformation which saves time where HiveQL is query language of Hive which is very close to the SQL consist of all conventional data types and concepts like joins, clauses etc. although insertion is little bit different. Concepts like external and internal tables and SerDe increases its robustness and show its advancement from Hadoop. Hive supports multiple file format in respect to the user and user can declare its own file format also. MetaStore is the most important component for Hive as it gives all necessary things that make it different from other parallel technologies which stores the all meta data about schemas, tables, partition etc. using RDBMS using an open source layer called Data-nucleus. Query compiler serves the role of compiler for hive programs and acts as tradition compiler. While execution engine executes the queries. Facebook uses hive because Hadoop have a problem in tuning the processes in time and resource scheduling. Unlike from these systems hive is provides a system catalogue that perseveres metadata about tables inside the system. Hive is able to improve the performance by 20 % over Hadoop.

15. Hive: Distributed Agents for Networking Things

Authors: Nelson Minar, Matthew Gray, Oliver Roup, Raffi Krikorian, and Pattie Maes

Description: This paper is focused majorly upon three things: Distributed Systems, Hive and Things That Think. Here, author is discussing about a system in which everything doesn't matters electronic or not thinks for the sake of human favour by using distributed systems and Hive. Basic terms that are used are: Cell which is actually an individual system or node. Shadow which is the software drivers that are present inside each cell and Agents which are actually the processes which is to be executed. The main features of Agents are: Mobility i.e. can migrate from cell to another, autonomous, proactive, interactive etc. User interface is of graphical form i.e. each resource is depicted in the form of icon, user is just need to draw an arrow in order to connect them. Locating resource is a complete responsibility of cell and also protecting from outside threat is also taken car by host. Agent interaction is completer ad hoc and can only be done for a certain time say till completion of process etc. Inside Hive interaction can be done in two way i.e. either by syntactically or semantically. Mobility is main feature as at any circumstance agent has the capability to migrate to a different cell. There are certain limitations to this system i.e. Options available for distributed architecture in java are limited. Synchronous communication is not that much successful in these types of system as each cell is independent. Communication is based upon Connectionless theory. New kind of device needs to train system for its own acceptability. Due to code mobility, versioning is new challenge which is emerging. Open nature of hive makes the system unpredictable. Although, Hive proved itself a powerful partner for distributed system but still there is certain room for the advancement.

16. ESAMR: An Enhanced Self-Adaptive MapReduce Scheduling Algorithm

Authors: Xiaoyu Sun, Chen He and Ying Lu

Description: This paper is majorly focused on the problem of straggler in Hadoop cluster while running Map and Reduce jobs on the different nodes. When a particular node is about to finish it's task, then suddenly it's performance degrades with respects to processing speed which adversely affects the overall performance of Hadoop cluster. The node which suffers from this problem is called as straggler. This paper proposes an algorithm to detect the slow processes into the cluster named as ESAMR which is the extension of previous work SAMR i.e. self-adaptive MapReduce. The major difference between these two algorithms is approach of heterogeneity i.e. while SAMR consider only hardware heterogeneity of different nodes that can lead to different time of completion of task whereas ESAMR takes account of heterogeneity of software's on different nodes i.e. same task can take different amount of time on different node as data included in between them might be different. ESAMR. ESAMR supports the idea of SAMR of having historical information as an parameter but additionally it uses K-means clustering to categorize this historical data. It then uses its own algorithm to identify the slow task into the cluster. Identification of slow task is done by generating stage weights which is more accurate in case of ESAMR than SAMR.

17. HPMR: Prefetching and Pre-shuffling in shared MapReduce Computation Environment

Authors: Sangwon Seo1, Ingook Jang1, Kyungchang Woo2, Inkyo Kim3, Jin-Soo Kim4, and Seungryoul Maeng1

Description: In this paper, Hadoop performance is increased by adding two more phases into the whole scenario and theses phases are Prefetching and Pre-Shuffling. For the implementation of this whole new idea an engine is created which is named HPMR i.e. High-Performance MapReduce whose plugin can be linked two Hadoop (in this case Hadoop 0.18.3 is used). The main objective of this paper is to fully utilize the advantage of data locality and to reduce network overhead. Algorithms for both schemes are proposed for both Map tasks and Reduce tasks separately. One more algorithm is proposed in this paper separately for detecting the slow task in order to manage the problem of straggler. This algorithm is motivated from LATE, so it is name as D-LATE. This overall concept is implemented and proved to be very useful as performances comes out to be better in case of HPMR.

18. SAMR: A Self-Adaptive MapReduce scheduling algorithm in Heterogeneous Environment

Authors: Q. Chen, D. Zhang, M. Guo, Q. Deng, and S. Guo

Description: In this paper, an attempt to propose a new algorithm to identify the stragglers into a Hadoop clusters is made. Straggler is a node into a cluster whose processing task is almost done but at the end of processing its performance reduced, which affects the overall performance of the cluster. SAMR is proposed to overcome the limitations of the previous algorithm such as classical Hadoop based algorithm and Longest Approximate Time to End (LATE) algorithm which works in a static manner and changes its progress score in a monotonic way which is not applicable for all jobs. SAMR introduces the concept of hardware heterogeneity and historic data which becomes more accurate in detecting or predicting the slow task and identifying straggler.

19. Longest Approximate Time to End Scheduling Algorithm in Hadoop Environment

Authors: R. Thanga selvi and R. Aruna

Description: In this paper, Scheduling of various processes on various nodes into cluster is discussed and a new algorithm is proposed for the sake to increase the overall productivity of the cluster. Various previous scheduling algorithm like First In First Out(FIFO), Round-Robin (RR), Fair scheduling algorithm, Capacity scheduling algorithm are discussed and then LATE is proposed with its three basic principles i.e.

i. Prioritizing the task available
ii. Selection of node to run
iii. Cap speculative task to avoid thrashing

Experimentally it is proven that, LATE is more accurate than any other previous schemes in every case with any means and capable of increasing the performance.

But still there are some limitation of LATE:

i. It works in static manner.
ii. It changes the progress rate in monotonic way.
iii. It works in same way for all processes even if they have heterogenous way.

CHAPTER 4

RESEARCH METHODOLOGY

As aim of the research is to optimize the performance of Hadoop cluster, so Hadoop architecture will be used. Some extra related technology might be used if needed like plug-in of HPMR (High Performance MapReduce) etc.

Brief Idea: Main aim of this research work is to find the straggler node from a cluster and reschedule its respective task in process to another node. There are many algorithms already present which are proposed earlier like LATE, SAMR, ESAMR, D-LATE. In this research work, we will make a hybrid of two different approaches which are Adaptive MapReduce and D-LATE algorithm.

Methodology Used: Series of steps are being performed in order to achieve the ultimate goal. Every individual step is as important as any other step.

- **Deciding Broad Area:** This is the first towards deciding the problem to work upon. Big Data, IOT, Database Security, Data cleaning are some of broad areas that are currently available in Database domain.
- **Literature Survey:** A number of paper have been reviewed in order to accumulate the knowledge which is necessary to complete the work. Initially from the university library and other related sources after that other sources like internet libraries are considered.
- **Deciding Problem Area:** In this phase, Broader area is narrow down to some specific area in which there is room to find the problem on which work can be done. In this paper, our problem area is Hadoop architecture more precisely performance optimization of Hadoop.
- **Declaring Problem Statement and Base Paper:** Declaration of problem statement is one of the most important task i.e. to be done and sometimes it becomes a bottleneck for researcher.
- **Deciding Tools and Technology to be used:** After declaring problem is to finalizing the technologies and tools that is to be used which will make it feasible to reach to the ultimate solution of the problem.

- **Installation of Infrastructure:** Necessary software and tools is to be installed in this phase which will be used in implementation part. In this case, Hadoop architecture is the most important.
- **Dry Run of work proposed:** It is also necessary to dry run the proposed work i.e. to implement the algorithm (if any) on paper, which will give a glimpse to researcher that how the work is going to be happen.
- **Implementation:** It is an iterative work in which all the proposed work is to be done practically. Every time certain step fails, little change can be made and work is executed and it is repeated till the objective is accomplished.
- **Documentation:** Proper representation of work needs proper documentation. Documentation must have Initial work, related work, source of idea, details of implementation, graphs, figures, facts, results etc.

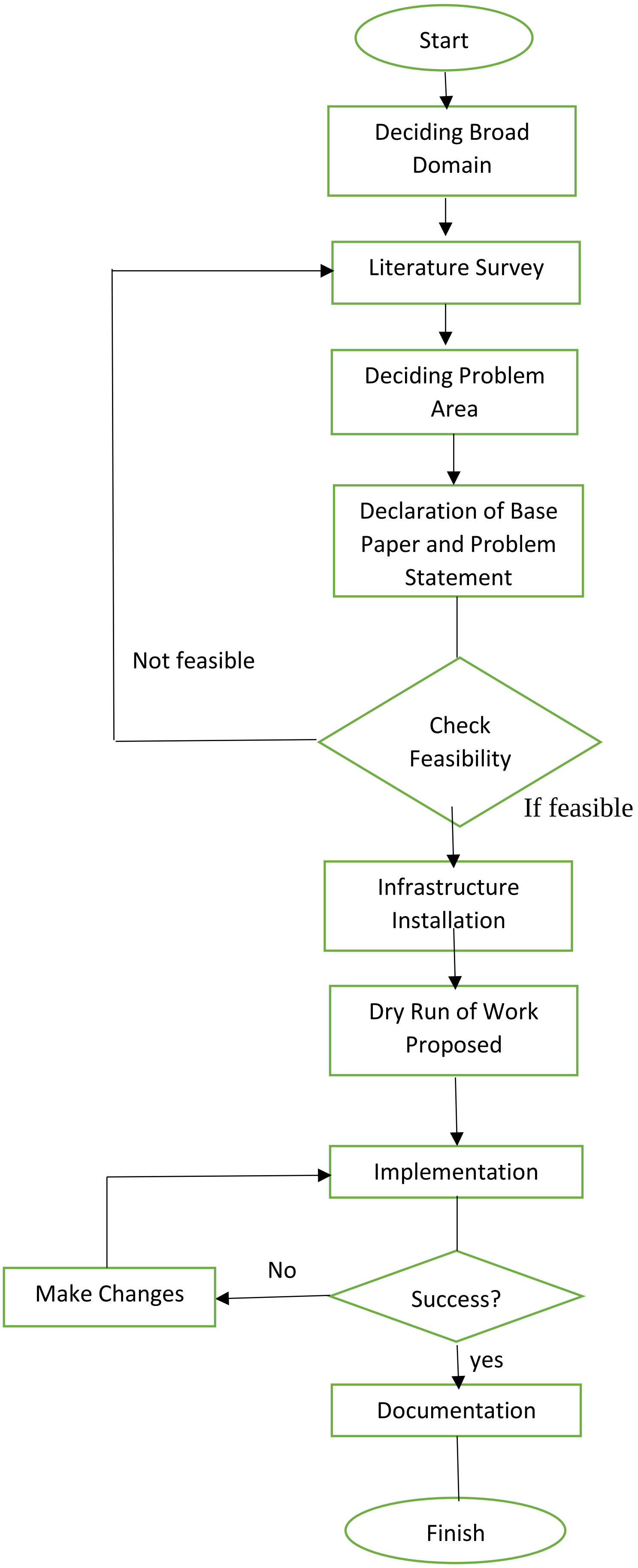

Fig. 1 Flowchart of Proposed Methodology

CHAPTER 6
EXPECTED OUTCOME

The expected outcome of this research work is to achieve the ultimate goal i.e. to identify the slow processes and then encounter straggler nodes form cluster by proposing a algorithm which might be a hybrid of previously proposed algorithms in order to combine their benefits and exclude limitations.

CHAPTER 7

SUMMARY AND CONCLUSION

Problem of straggler can be solved if we reschedule the process on to some other node into cluster but first straggler need to be identified from cluster, as single straggler can reduce performance of whole cluster. In this paper, we focus on just to encounter slow processes and then identifying which will later be passed to some technique which is capable of rescheduling of process to some other node.

REFERENCES

[1] Understandable Big Data: A survey
Authors: Cheikh Kacfah Emani, Nadine Cullot, Christophe Nicolle

[2] MapReduce: Simplified Data Processing on Large Clusters
Authors: Jeffrey Dean and Sanjay Ghemawat

[3] The Hadoop Distributed File System
Authors: Konstantin Shvachko, Hairong Kuang, Sanjay Radia, Robert Chansler

[4] The rise of "big data" on cloud computing: Review and open research issues
Authors: Ibrahim Abaker Targio Hashem, Ibrar Yaqoob, Nor Badrul Anuar, Salimah Mokhtar, Abdullah Gani, Samee Ullah Khan

[5] A Survey on Big Data Analytics: Challenges, Open Research Issues and Tools
Authors: Debi Prasanna Acharjya, Kauser Ahmed P

[6] Mobile Healthcare Information Management utilizing Cloud Computing and Android OS
Authors: Charalampos Doukas Thomas Pliakas, Ilias Maglogiannis

[7] Can Sensors Collect Big Data? An Energy Efficient Big Data Gathering Algorithm for WSN
Authors: Shalli Rani, Syed Hassan Ahmed, Rajneesh Talwar, Jyoteesh Malhotra

[8] Data Cleaning and Query Answering with Matching Dependencies and Matching Functions
Authors: Leopoldo Bertossi, Solmaz Kolahi, Laks V.S. Lakshmanan

[9] Answering Queries using Humans, Algorithms and Databases
Authors: Aditya Parameswaran, Neoklis Polyzotis

[10] Data Cleaning: Overview and Emerging Challenges:
Authors: Xu Chu, Ihab F. Ilyas Sanjay Krishnan, Jiannan Wang

[11] Outsourcing to the cloud: data security and privacy risks
Authors: Peter Brudenall, Bridget Treacy and Purdey Castle

[12] Research on the security technology of big data information
Authors: Hong Zhu, Zheng Xu, Yingzhen Huang

[13] Survey of Research on Information Security in Big Data
Authors: Zhang Hongjun, Hao Wenning, He Dengchao, Mao Yuxing

[14] Hive – A Petabyte Scale Data Warehouse Using Hadoop
Authors: Ashish Thusoo, Joydeep Sen Sarma, Namit Jain, Zheng Shao, Prasad Chakka, Ning Zhang, Suresh Antony, Hao Liu and Raghotham Murthy

[15] Hive: Distributed Agents for Networking Things
Authors: Nelson Minar, Matthew Gray, Oliver Roup, Raffi Krikorian, and Pattie Maes

[16] ESAMR: An Enhanced Self-Adaptive MapReduce Scheduling Algorithm
Authors: Xiaoyu Sun, Chen He and Ying Lu

[17] HPMR: Prefetching and Pre-shuffling in shared MapReduce Computation Environment
Authors: Sangwon Seo1, Ingook Jang1, Kyungchang Woo2, Inkyo Kim3, Jin-Soo Kim4 and Seungryoul Maeng1

[18] SAMR: A Self-Adaptive MapReduce scheduling algorithm in Heterogeneous Environment
Authors: Q. Chen, D. Zhang, M. Guo, Q. Deng, and S. Guo

[19] Longest Approximate Time to End Scheduling Algorithm in Hadoop Environment
Authors: R. Thanga selvi and R. Aruna

[20] http://www.hadooptpoint.org

[21] https://www.wisdomjobs.com