International Journal of Advance Research in Science and Engineering
Volume No.08, Issue No.05, May 2019
www.ijarse.com
IJARSE
ISSN: 2319-8354

Big Data Analytics: Predicting Academic Course Preference Using Hadoop Inspired Map Reduce

Prajwal M G, A Ananda Shankar

School Of Computing And Information Technology, REVA UNIVERSITY (Bengaluru, India)

Abstract:

With the emergence of new technologies, new academic trends introduced into Educational system which results in large data which is unregulated and it is also challenge for students to prefer to those academic courses which are helpful in their industrial training and increases their career prospects. Another challenge is to convert the unregulated data into structured and meaningful information there is need of Data Mining Tools. Hadoop Distributed File System is used to hold large amount of data. The Files are stored in a redundant fashion across multiple machines which ensure their endurance to failure and parallel applications. Knowledge extracted using Map Reduce will be helpful indecision making for students to determine courses chosen for industrial trainings. In this paper, we are deriving preferable courses for pursuing training for students based on course combinations. Here, using HDFS, tasks run over Map Reduce and output is obtained after aggregation of results

1. Introduction

Data mining is one of the most prominent areas in modern technologies for retrieving meaningful information from huge amount of unstructured and distributed data using parallel processing of data. There is huge advantage to Educational sector of following Data Mining Techniques to analyze data input from students, feedbacks, latest academic trends etc which helps in providing quality education and decision-making approach for students to increase their career prospects and right selection of courses for industrial trainings to fulfill the skill gap pertains between primary education and industry hiring students. Data Mining has great impact in academic systems where education is weighed as primary input for societal progress.

Big data is the emerging field of data mining. It is a term for datasets that are so large or complex that traditional data processing application software is incompetent to deal with them. Big data includes gathering of data for storage and analysis purpose which gain control over operations like searching, sharing, visualization of data, query processing, updation and maintain privacy of information. In Big data, here is extremely large dataset that is analyzed computationally to reveal patterns, trends and associations. It deals with unstructured data which may include MS Office files, PDF, Text etc whereas structured data may be the relational data. Hadoop is one technique of big data and answer to problems related to handling of unstructured and massive data. Hadoop is an open-source programming paradigm which performs parallel processing of applications on clusters. Big Data approach can help colleges, institutions, universities to get a comprehensive aspect about the students. It helps in answering questions related to the learning behaviors, better understanding and curriculum trends, and future course selection for students which helps to create captivating learning experiences for

students. The problem of enormously large size of dataset can be solved using Map Reduce Techniques .Map Reduce jobs run over Hadoop Clusters by splitting the big data into small chunks and process the data by running it parallel on distributed clusters.

The paper is arranged as follows. Section II laid stress on various approaches followed by authors for solving problems related to large and unstructured data using Big Data Techniques. Section III presents data description i.e., input data and the desired output using Map Reduce programming model for running jobs using Hadoop File System for the proposed work are discussed in section IV. The results from proposed work are reviewed in section V and lastly, the work is concluded in section VI.

2. LITERATURE SURVEY

2.1 Predicting Student Performance Using Map Reduce

Data mining and machine learning depend on classification which is the most essential and important task. Many experiments are performed on Student datasets using multiple classifiers and feature selection techniques. Many of them show good classification accuracy. The existing work proposes to apply data mining techniques to predict Students dropout and failure. But this work doesn't support the huge amount of data. It also takes more time to complete the classification process. So the time complexity is high. To improve the accuracy and reduce the time complexity, the Map Reduce concept is introduced. In this work, the deadline constraint is also introduced. Based on this, an extensional Map Reduce Task Scheduling algorithm for Deadline constraints (MTSD) is proposed. It allows user to specify a job's (classification process in data mining) deadline and tries to make the job to be finished before the deadline. Finally, the proposed system has higher classification accuracy even in the big data and it also reduced the time complexity.

2.2Map Reduce as a Programming Model for Association Rules Algorithm on Hadoop

As association rules widely used, it needs to study many problems, one of which is the generally larger and multi-dimensional datasets, and the rapid growth of the amount of data. Single processor's memory and CPU resources are very limited, which makes the algorithm performance inefficient. Recently the development of network and distributed technology makes cloud computing a reality in the implementation of association rules algorithm. In this paper we describe the improved Apriori algorithm based on Map Reduce mode, which can handle massive datasets with a large number of nodes on Hadoop platform.

2.3 Implementation of Hadoop Operations for Big Data Processing in Educational Institutions

Education plays an important role in maintaining the economic growth of a country. The objective of this paper is to focus on the impact of cloud computing on educational institutions by using latest big data technology to provide quality education. Our educational systems have a large amount of data. Big Data is defined as massive sets of data that is so large or so complex that it is very difficult to process by using conventional applications

and software technologies. This has resulted in the penetration of Big Data technologies and tools into education, to process the large amount of data involved. In this paper we discuss what Cloud and Hadoop is, and its types, operations and services offered. Hence it has an advantage which will surely help the students when used in an appropriate way.

2.4 Data Mining in Education: Data Classification and Decision Tree Approach

Educational organizations are one of the important parts of our society and playing a vital role for growth and development of any nation. Data Mining is an emerging technique with the help of this one can efficiently learn with historical data and use that knowledge for predicting future behavior of concern areas. Growth of current education system is surely enhanced if data mining has been adopted as a futuristic strategic management tool. The Data Mining tool is able to facilitate better resource utilization in terms of student performance, course development and finally the development of nation's education related standards. In this paper a student data from a community college database has been taken and various classification approaches have been performed and a comparative analysis has been done. In this research work Support Vector Machines (SVM) are established as a best classifier with maximum accuracy and minimum root mean square error (RMSE). The study also includes a comparative analysis of all Support Vector Machine Kernel types and in this the Radial Basis Kernel is identified as a best choice for Support Vector Machine. A Decision tree approach is proposed which may be taken as an important basis of selection of student during any course program. The paper is aimed to develop a faith on Data Mining techniques so that present education and business system may adopt this as a strategic management tool.

2.5Apriori-Map/Reduce Algorithm

Map/Reduce algorithm has received highlights as cloud computing services with Hadoop frame works were provided. Thus, there have been many approaches to convert many sequential algorithms to the corresponding Map/Reduce algorithms. The paper presents Map/Reduce algorithm of the legacy Apriori algorithm that has been popular to collect the item sets frequently occurred in order to compose Association Rule in Data Mining. Theoretically, it shows that the proposed algorithm provides high performance computing depending on the number of Map and Reduce nodes.

3. Existing System:

Educational system which results in large data which is unregulated and it is also challenge for students to prefer to those academic courses. we are deriving preferable courses for pursuing training for students based on course combinations. Here, using HDFS, tasks run over Map Reduce and output is obtained after aggregation of results.

Disadvantages

- Less performance
- Processing unstructured data is very difficult.

4. Proposed System

Using Map Reduce, the application can be scaled to run over multiple machines in a cluster and for that Hadoop cluster is used. The Map Reduce Framework consists of Map and Reduce Functions with single Resource Manager which acts as a master and one Node manager which acts as slave per cluster node. The input dataset is fed into the mapper and after passing through shuffle phase, reducer displays the output after aggregating the tuples obtained from mapper and are in the form of <key, value> pair.

Advantages:

- More Performance
- improved to perform huge number of data.

5. Modules:

5.1 Data collection :

In the data collection phase the sensor sensing the data will be collected the data generated will be like twitter events generated data the Course name, time and year in which the events occurred.

5.2 Data processing:

In the data processing phase the data will be processed and send the data to Course Prediction network

5.3 Communication:

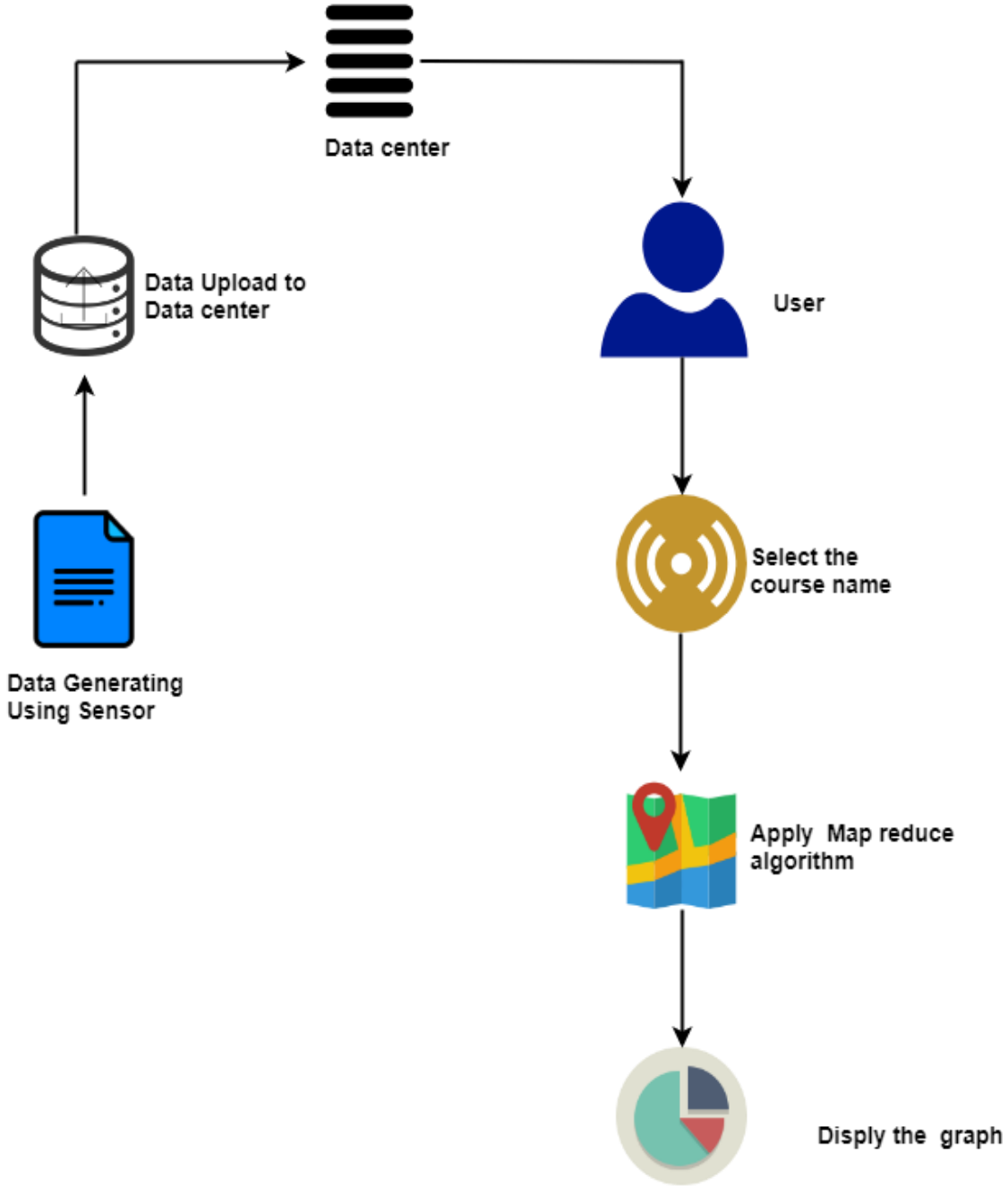
The communication occurred between the sensor and the Course Prediction network will be established through the socket communication. The user once login to the system search the data by hash keyword i.e.. Course name in which the particular event occurred and stores the count and display graph by using map-reduce program.

5.4 Data storage:

In the data storage phase Course Prediction network the data will be stored.

System Architecture

International Journal of Advance Research in Science and Engineering
Volume No.08, Issue No.05, May 2019
www.ijarse.com
IJARSE
ISSN: 2319-8354



6. SYSTEM DESIGN

System design is the process of defining the architecture, components, modules, interfaces and data for a system to satisfy specified requirements. One could see it as the application of systems theory to product development. There is some overlap with the disciplines of systems analysis, systems architecture and systems engineering. If the broader topic of product development "blends the perspective of marketing, design, and manufacturing into a single approach to product development," then design is the act of taking the marketing

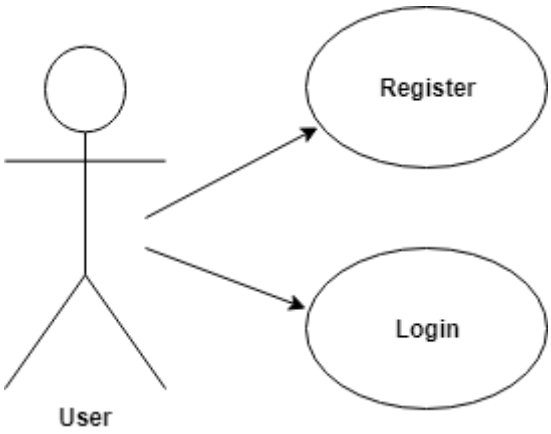
International Journal of Advance Research in Science and Engineering
Volume No.08, Issue No.05, May 2019
www.ijarse.com
IJARSE
ISSN: 2319-8354

information and creating the design of the product to be manufactured. Systems design is therefore the process of defining and developing systems to satisfy specified requirements of the user.

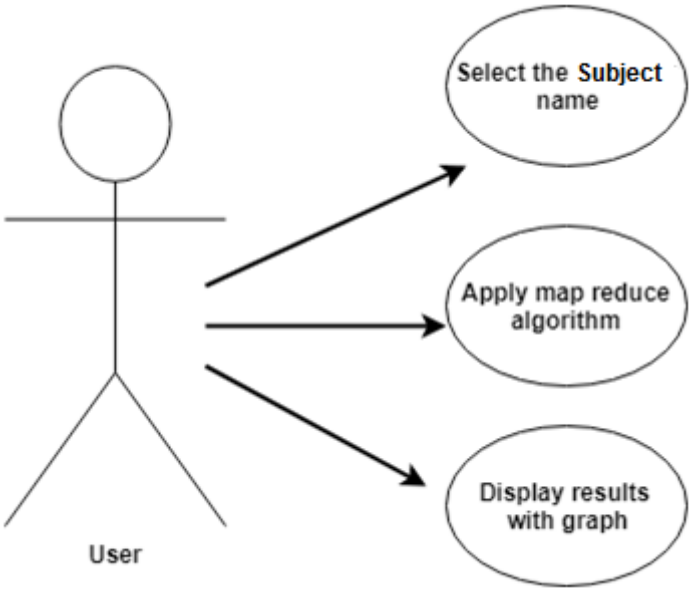
7 FIGURES

7.1 Use Case Diagram:

Use case 1



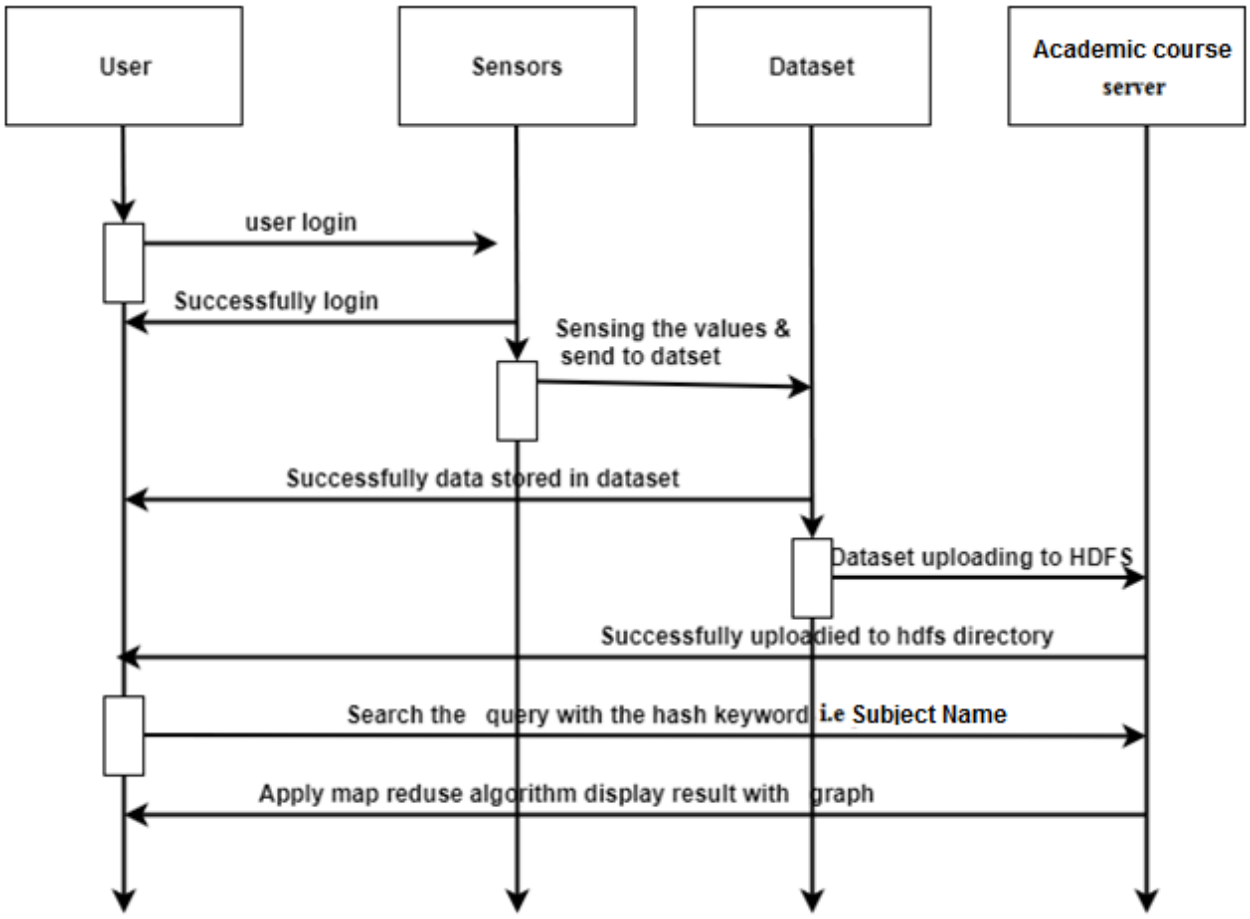
Use case 2



International Journal of Advance Research in Science and Engineering
Volume No.08, Issue No.05, May 2019
www.ijarse.com
IJARSE
ISSN: 2319-8354

7.2 Sequence Diagram

A sequence diagram in a UML is a kind of interaction diagram that shows how processes operate with one another and in what order. It is a construct of a Message Sequence Chart. A sequence diagram shows object interactions arranged in time sequence. It depicts the objects and classes involved in the scenario and the sequence of messages exchanged between the objects needed to carry out the functionality of the scenario. Sequence diagrams typically are associated with use case realizations in the Logical View of the system under development.



7.3 Data Flow Diagram

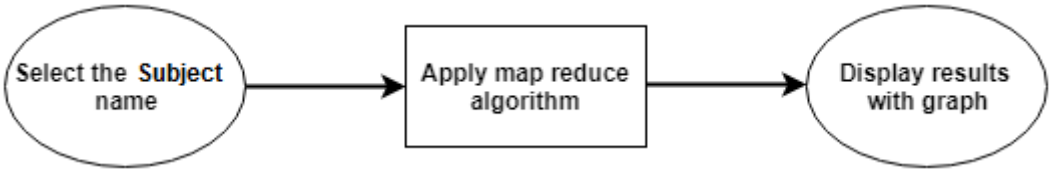
A data flow diagram is a graphical representation of the "flow" of data through an information system, modeling its process aspects. Often they are a preliminary step used to create an overview of the system which can later be elaborated. DFDs can also be used for the visualization of data processing (structured design).

International Journal of Advance Research in Science and Engineering
Volume No.08, Issue No.05, May 2019
www.ijarse.com
IJARSE
ISSN: 2319-8354

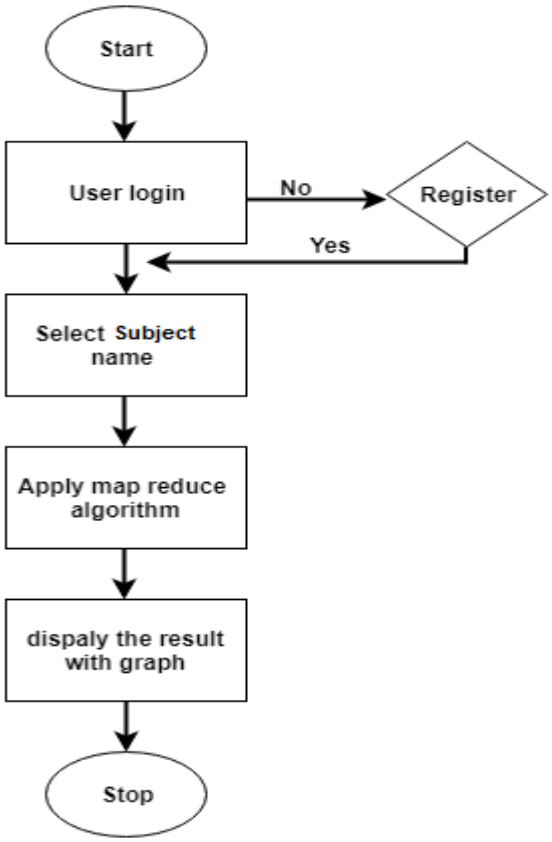
It is a simple graphical formalism that can be used to represent a system in terms of the input data to the system, various processing carried out on these data, and the output data is generated by the system.

DFD 1

DFD2

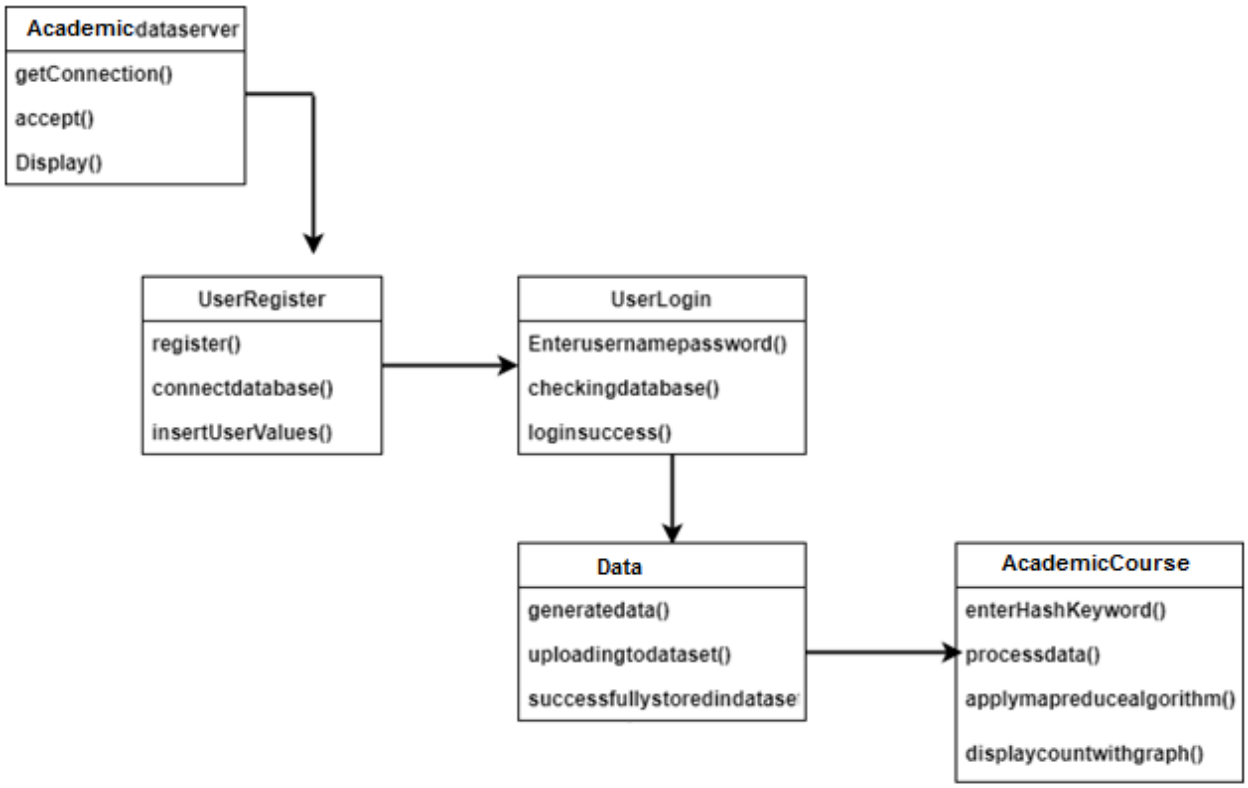


7.4 Flow Chart



International Journal of Advance Research in Science and Engineering
Volume No.08, Issue No.05, May 2019
www.ijarse.com
IJARSE
ISSN: 2319-8354

7.5 Class Diagram:



8 SYSTEM REQUIREMENT SPECIFICATION

To be used efficiently, all computer software needs certain hardware components or other software resources to be present on a computer. These prerequisites are known as (computer) system requirements and are often used as a guideline as opposed to an absolute rule. Most software defines two sets of system requirements: minimum and recommended. With increasing demand for higher processing power and resources in newer versions of software, system requirements tend to increase over time. Industry analysts suggest that this trend plays a bigger part in driving upgrades to existing computer systems than technological advancements.

8.1 Non functional requirements

Non functional requirements are the functions offered by the system. It includes time constraints and constraints on the development process and standards. The non functional requirements are as follows:

- **Speed:** The system should process the given input into output within appropriate time.
- **Ease of use:** The software should be user friendly. Then the customers can use easily, so it doesn't require much training time.

- **Reliability:** The rate of failures should be less then only the system is more reliable
- **Portability**: It should be easy to implement in any system.

8.2 Specific Requirements

The specific requirements are:

- **User Interfaces:** The external users are the clients. All the clients can use this software for indexing and searching.
- **Hardware Interfaces:** The external hardware interface used for indexing and searching is personal computers of the clients. The PC's may be laptops with wireless LAN as the internet connections provided will be wireless.
- **Software Interfaces:** The Operating Systems can be any version of Windows.
- **Performance Requirements:** The PC's used must be atleast Pentium 4 machines so that they can give optimum performance of the product.

9 Software requirements

Software requirements deal with defining software resource requirements and prerequisites that need to be installed on a computer to provide optimal functioning of an application.

These requirements or prerequisites are generally not included in the software installation package and need to be installed separately before the software is installed.

- Java1.4 or higher
  - Java Swing – front end
  - Networking-Socket programming
- Windows 98 or higher-Operating System

10 Hardware requirements

The most common set of requirements defined by any operating system or software application is the physical computer resources, also known as hardware, A hardware requirements list is often accompanied by a hardware compatibility list, especially in case of operating systems. An HCL lists tested, compatible, and sometimes incompatible hardware devices for a particular operating system or application. The following sub-sections discuss the various aspects of hardware requirements.

All computer operating systems are designed for a particular computer architecture. Most software applications are limited to particular operating systems running on particular architectures. Although architecture-independent operating systems and applications exist, most need to be recompiled to run on a new architecture.

The power of the central processing unit (CPU) is a fundamental system requirement for any software. Most software running on x86 architecture define processing power as the model and the clock speed of the CPU. Many other features of a CPU that influence its speed and power, like bus speed,

cache, and MIPS are often ignored. This definition of power is often erroneous, as AMDAthlon and IntelPentium CPUs at similar clock speed often have different throughput speeds.

- 10GB HDD(min)
- 128 MB RAM(min)
- Pentium P4 Processor 2.8Ghz(min)

11 Overview of technologies

The technologies used in TARF is described as below:

11.1History of Java

Java language was developed by James Gosling and his team at sun Microsystems and released formally in 1995. Its former name is oak. Java Development Kit 1.0 was released in 1996 to popularize java and is freely available on Internet.

11.2Overview of Java

Java is loosely based on c++ syntax, and is meant to be Object-Oriented Structure of java is midway between an interpreted and a compiled language. The java compiler into ByteCodes, which are secure and portable across different platforms, compiles Java programs. These byte codes are essentially instructions encapsulated in single type, to what is known as java virtual machine (JVM), which resides in standard browser. JVM is available for almost all OS. JVM converts these byte codes into machine specific instructions at runtime. Java is actually a platform consisting of three components:

- Java programming language.
- Java library of classes and interfaces.
- Java Virtual Machine

11.3Features of Java

- Java is a simple language. It does not make use of pointers, function overloading etc,.
- Java is object-oriented language and supports encapsulation, inheritance, Polymorphism and dynamic binding, but does not support multiple inheritance.
- Everything in java is an object except some primitive data types.
- Java is portable.
- It is an architecture neutral that is java programs once compiled can be executed on any machine that is enabled.
- Java is distributed in its approach and used for Internet programming.
- Java is robust, secured, high performing and dynamic in nature.
- Java supports multithreading. Therefore different parts of the program can be executed at the same time.

11.4Packages

One of the most innovative features of java is packages. The packages both a naming and a visibility control mechanism we can define classes inside a package that are not accessible by code outside the package. It can define the class members that are only exposed to the other members of the same package. Java uses file system directories to store packages. For example the .class files for any classes you declare to be part of My Package must be stored in the directory called My Package remember that cases significant and directory name must match the package name exactly.

A package hierarchy must be reflected in the file system of your java development system. For example the package declared as -package java.awt.image; needs to be stored in java\awt\image in a windows environment.

11.5Java.lang package

The java package, java.lang contains fundamental classes and interfaces closely tied to the language and run time system which includes the root classes that form the class hierarchy, types tied to the language definition, basic exceptions, math functions, threading, security functions as well as some information on the underlying native system.

11.6Java.util

Data structures that aggregate objects are the focus of the Java.util package included in the packet is the collections API and organized data structure hierarchy influenced heavily by design pattern consideration.

11.7Java .security

It provides the classes and interfaces for security framework. It includes classes that implement an easily configurable, fine grained access control security architecture. The packages also supports a generation and storage of cryptographic public key pairs. Finally this package provides classes that support signed/guarded objects and secure random number generation.

11.8Swings

Swing is a widget toolkit for Java. It's a part of sun Microsystems Java foundation classes-API for providing graphical user interface for Java programs. Swing was developed to provide a more sophisticated set of GUI components than the earlier abstract window toolkit. Swings provide a native look and feel that emulates look and feel of several look and feel unrelated to the underlying platform. Swings introduced a mechanism that allows the look and feel of every component in an application to be altered without making substantial changes to the application code. The introduction of support for a plugable look and feel allows swing components to emulate for the appearance of native components while still retaining the benefits of platform independence. The above feature also makes it easy to make an application written in swing look very different from native programs if desired.

Look and feel

In software design look and feel is used in respect of GUI and comprises of its design, including elements such as colors, shapes, layout and typefaces(the "LOOK") as well as the behavior of dynamic elements

such as button, boxes and menus(the "FEEL"). The term look and feel is used in reference to both software and websites.

12 Hadoop :

What is Big Data

Data which are very large in size is called Big Data. Normally we work on data of size MB(WordDoc ,Excel) or maximum GB(Movies, Codes) but data in Peta bytes i.e. 10^15 byte size is called Big Data. It is stated that almost 90% of today's data has been generated in the past 3 years.

Sources of Big Data

These data come from many sources like

- **Social networking sites:** Facebook, Google, LinkedIn all these sites generates huge amount of data on a day to day basis as they have billions of users worldwide.
- **E-commerce site:** Sites like Amazon, Flipkart, Alibaba generates huge amount of logs from which users buying trends can be traced.
- **Weather Station:** All the weather station and satellite gives very huge data which are stored and manipulated to forecast weather.
- **Telecom company:** Telecom giants like Airtel, Vodafone study the user trends and accordingly publish their plans and for this they store the data of its million users.
- **Share Market:** Stock exchange across the world generates huge amount of data through its daily transaction.

3V's of Big Data

1. **Velocity:** The data is increasing at a very fast rate. It is estimated that the volume of data will double in every 2 years.
2. **Variety:** Now a days data are not stored in rows and column. Data is structured as well as unstructured. Log file, CCTV footage is unstructured data. Data which can be saved in tables are structured data like the transaction data of the bank.
3. **Volume:** The amount of data which we deal with is of very large size of Peta bytes.

Use case

An e-commerce site XYZ (having 100 million users) wants to offer a gift voucher of 100$ to its top 10 customers who have spent the most in the previous year. Moreover, they want to find the buying trend of these customers so that company can suggest more items related to them.

Issues

Huge amount of unstructured data which needs to be stored, processed and analyzed.

Solution

Storage: This huge amount of data, Hadoop uses HDFS (Hadoop Distributed File System) which uses commodity hardware to form clusters and store data in a distributed fashion. It works on Write once, read many times principle.

Processing: Map Reduce paradigm is applied to data distributed over network to find the required output.

Analyze: Pig, Hive can be used to analyze the data.

Cost: Hadoop is open source so the cost is no more an issue.

What is Hadoop

Hadoop is an open source framework from Apache and is used to store process and analyze data which are very huge in volume. Hadoop is written in Java and is not OLAP (online analytical processing). It is used for batch/offline processing.It is being used by Facebook, Yahoo, Google, Twitter, LinkedIn and many more. Moreover it can be scaled up just by adding nodes in the cluster.

Modules of Hadoop

1. **HDFS:** Hadoop Distributed File System. Google published its paper GFS and on the basis of that HDFS was developed. It states that the files will be broken into blocks and stored in nodes over the distributed architecture.
2. **Yarn:** Yet another Resource Negotiator is used for job scheduling and manage the cluster.
3. **Map Reduce:** This is a framework which helps Java programs to do the parallel computation on data using key value pair. The Map task takes input data and converts it into a data set which can be computed in Key value pair. The output of Map task is consumed by reduce task and then the out of reducer gives the desired result.
4. **Hadoop Common:** These Java libraries are used to start Hadoop and are used by other Hadoop modules.

Advantages of Hadoop

- **Fast:** In HDFS the data distributed over the cluster and are mapped which helps in faster retrieval. Even the tools to process the data are often on the same servers, thus reducing the processing time. It is able to process terabytes of data in minutes and Peta bytes in hours.
- **Scalable:** Hadoop cluster can be extended by just adding nodes in the cluster.
- **Cost Effective:** Hadoop is open source and uses commodity hardware to store data so it really cost effective as compared to traditional relational database management system.
- **Resilient to failure:** HDFS has the property with which it can replicate data over the network, so if one node is down or some other network failure happens, then Hadoop takes the other copy of data and use it. Normally, data are replicated thrice but the replication factor is configurable.

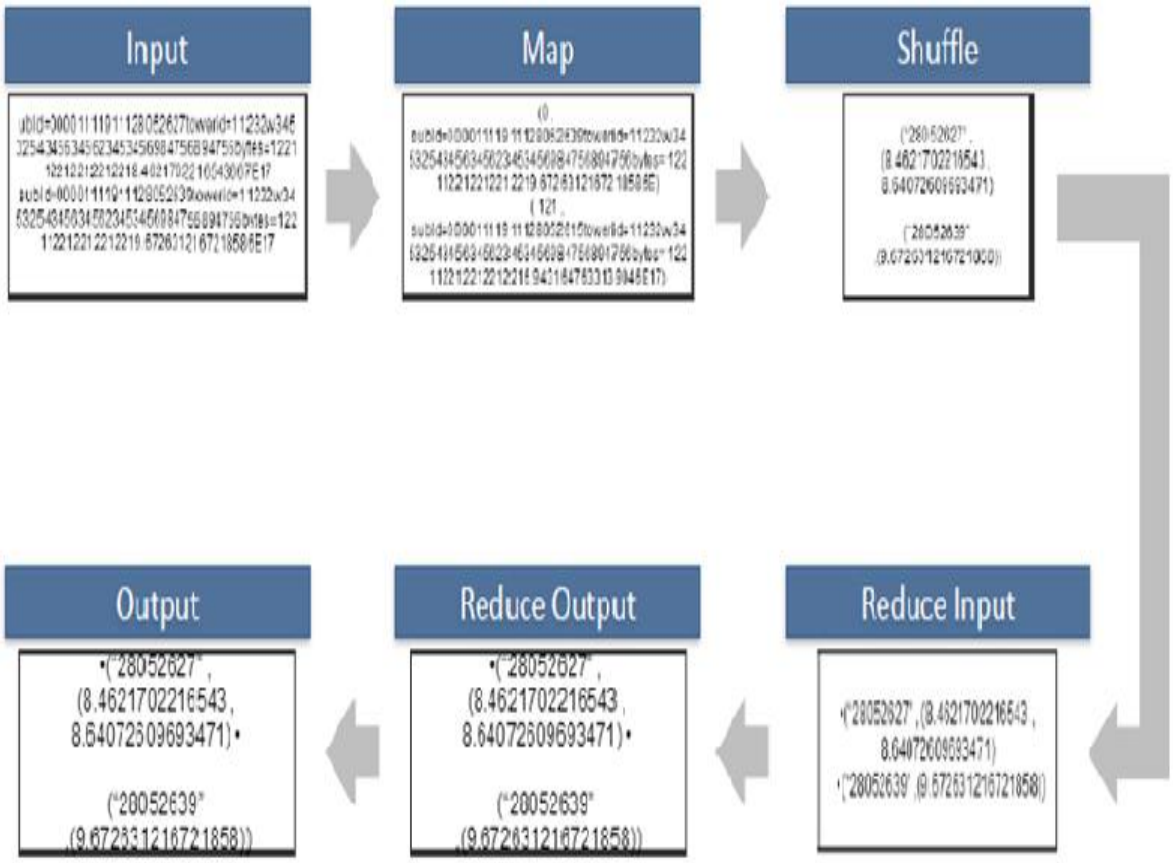
International Journal of Advance Research in Science and Engineering
Volume No.08, Issue No.05, May 2019
www.ijarse.com
IJARSE
ISSN: 2319-8354

MapReduce

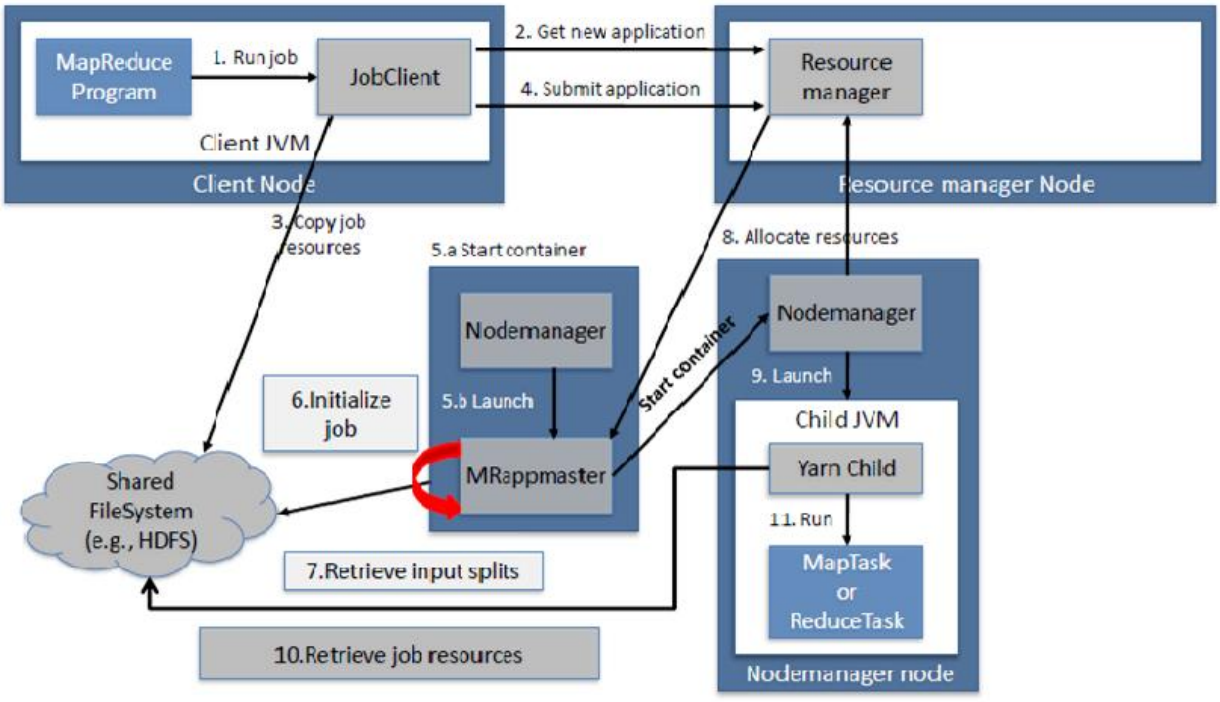
To take the advantage of parallel processing of Hadoop, the query must be in MapReduce form. The MapReduce is a paradigm which has two phases, the mapper phase and the reducer phase. In the Mapper the input is given in the form of key value pair. The output of the mapper is fed to the reducer as input. The reducer runs only after the mapper is over. The reducer too takes input in key value format and the output of reducer is final output.

Steps in Map Reduce

- Map takes a data in the form of pairs and returns a list of <key, value> pairs. The keys will not be unique in this case.
- Using the output of Map, sort and shuffle are applied by the Hadoop architecture. This sort and shuffle acts on these list of <key, value> pairs and sends out unique keys and a list of values associated with this unique key <key, list(values)>.
- Output of sort and shuffle will be sent to reducer phase. Reducer will perform a defined function on list of values for unique keys and Finaloutput will<key, value> will be stored/displayed.



International Journal of Advance Research in Science and Engineering
Volume No.08, Issue No.05, May 2019
www.ijarse.com
IJARSE
ISSN: 2319-8354



How Many Maps

The size of data to be processed decides the number of maps required. For example, we have 1000 MB data and block size is 64 MB then we need 16 mappers.

Sort and Shuffle

The sort and shuffle occur on the output of mapper and before the reducer.When the mapper task is complete, the results are sorted by key, partitioned if there are multiple reducers, and then written to disk.Using the input from each mapper <k2,v2> , we collect all the values for each unique key k2. This output from the shuffle phase in the form of <k2,list(v2)> is sent as input to reducer phase.

13 TESTING

Testing is a critical element which assures quality and effectiveness of the proposed system in (satisfying) meeting its objectives. Testing is done at various stages in the System designing and implementation process with an objective of developing an transparent, flexible and secured system. Testing is an integral part of software development. Testing process, in a way certifies, whether the product, that is developed, complies with the standards, that it was designed to. Testing process involves building of test cases, against which, the product has to be tested.

13.1Test objectives

- Testing is a process of executing a program with the intent of finding an error.
- A good case is one that has a high probability of finding an undiscovered error.
- A successful test is one that uncovers a yet undiscovered error. If testing is conducted successfully (according to the objectives) it will uncover errors in the software. Testing can't show the absences of defects are present. It can only show that software defects are present.

13.2Testing principles

Before applying methods to design effective test cases, a software engineer must understand the basic principle that guides software testing. All the tests should be traceable to customer requirements.

13.3Testing design

Any engineering product can be tested in one of two ways:

13.3.1 White box Testing

This testing is also called as glass box testing. Inthis testing, by knowing the specified function that a product has been designed to perform test can be conducted that demonstrates each function is fully operation at the same time searching for errors in each function.

it is a test case design method that uses the control structure of the procedural design to derive test cases.

13.3.2 Black box Testing

Inthis testing by knowing the internal operation of a product, tests can be conducted to ensure that "all gears mesh", that is the internal operation performs according to specification and all internal components have been adequately exercised. It fundamentally focuses on the functional requirements of the software.

The steps involved in black box test case design are:

- Graph based testing methods
- Equivalence partitioning
- Boundary value analysis
- Comparison testing

13.4Testing strategies

A software testing strategy provides a road map for the software developer. Testing is a set of activities that can be planned in advanced and conducted systematically. For this reason a template for software testing a set of steps into which we can place specific test case design methods should be defined for software engineering process.

Any software testing strategy should have the following characteristics:

a. Testing begins at the module level and works outward toward the integration of the entire computer based system.
b. Different testing techniques are appropriate at different points in time.
c. The developer of the software and an independent test group conducts testing.

d. Testing and debugging are different activities but debugging must be accommodated in any testing strategy.

Levels of Testing

Testing can be done in different levels of SDLC. They are:

Unit Testing

The first level of testing is called unit testing. Unit testing verifies on the smallest unit of software designs-the module. The unit test is always white box oriented. In this, different modules are tested against the specifications produced during design for the modules. Unit testing is essentially for verification of the code produced during the coding phase, and hence the goal is to test the internal logic of the modules. It is typically done by the programmer of the module. Due to its close association with coding, the coding phase is frequently called "coding and unit testing." The unit test can be conducted in parallel for multiple modules.

The Test cases in unit testing are as follows:

Test Cases

Table I: Unit Test Case 1

Test Case ID	Unit Test Case 1
Description	Academic Course network working properly
Input	Academic Course network
Expected output	Academic Course network receiving data
Actual Result/Remarks	Got the expected output
Passed(?)	Yes

Table II: Unit Test Case 2

Test Case ID	Unit Test Case 2
Description	Sensor sending data
Input	Data
Expected output	Data sent successfully
Actual Result/Remarks	Got the expected output
Passed(?)	Yes

Table III: Unit Test Case 3

Test Case ID	Unit Test Case 3
Description	User registration
Input	User register with credentials
Expected output	Successful registration
Actual Result/Remarks	Got the expected output
Passed(?)	Yes

Table III: Unit Test Case 4

Test Case ID	Unit Test Case 3
Description	User request for data to Academic Course network
Input	Sending request
Expected output	Successfully receiving data
Actual Result/Remarks	Got the expected output
Passed(?)	Yes

14 Conclusion:

The Map Reduce approach is used for running jobs over HDFS. Using Map Reduce, the application can be scaled to run over multiple machines in a cluster and for that Hadoop cluster is used. The Map Reduce Framework consists of Map and Reduce Functions with single Resource Manager which acts as a master and one Node manager which acts as slave per cluster node. The input dataset is fed into the mapper and after passing through shuffle phase, reducer displays the output after aggregating the tuples obtained from mapper and are in the form of <key, value> pair. The dataset shows that the students have opted for multiple course combinations for industrial trainings and the data becomes unstructured as well confusing for students to optfor course for trainings. The results in this paper shows that large volume of course combinations in the form of input dataset after passed through mapper function in Map Reduce Framework which runs the job in parallel on a single node cluster using HDFS, it converts the data into individual tuples and the meaningful data obtained from Reducer function classifies the data of course combinations opted more by students and strengthens the decision-making of students as well institutions to prefer demanding course for industrial trainings. Apart from it, the predicted result from Hadoop programming framework which is the emerging field of Data Mining also helps Management to stress over these courses in their curriculum to improve student skills and increases employment chances for them.

Reference

[1] Sonali Agarwal, G. N. Pandey, M. D. Tiwari, "Data Mining inEducation: Data Classification and Decision Tree Approach", International Journal of e-Education, e-Business, e-Management and e-Learning, Vol. 2, No. 2, April 2012.

[2] Jongwook Woo, "Apriori -Map/Reduce Algorithm." Proceedings ofthe International Conference on Parallel and Distributed Processing Techniques and Applications (PDPTA). The Steering Committee of The World Congress in Computer Science, Computer Engineering and Applied Computing (World Comp), 2012.

[3] Xin YueYang, Zhen Liu, Yan Fu, "Map Reduce as a Programming Model for Association Rules Algorithm on Hadoop", Information Sciences and Interaction Sciences (ICIS), 2010 3rd International Conference on, pp. 99-102. IEEE, 2010.

[4] Katrina Sin, Loganathan Muthu," Application of Big Data in Education Data Mining and Learning Analytics – A Literature Review", ICTACT Journal on Soft Computing, ISSN: 2229-6956 (online), Vol5, Issue 4,July 2015.

[5] B.Manjulatha,Ambica Venna,K.Soumya," Implementation of Hadoop Operations for Big Data Processing in Educational Institutions", International Journal of Innovative Research in Computer and Communication Engineering, ISSN(Online) : 2320-9801,Vol. 4,Issue 4, April 2016.

[6] N.Tajunisha, M.Anjali, "Predicting Student Performance UsingMapReduce",IJECS,Vol.4,Issue 1, Jan 2015, p. 9971-9976.

[7] Shankar M.Patil, Praveen Kumar, "Data Mining Model for Effective Data Analysis of Higher Education Students UsingMapReduce",IJERMT,ISSN:2278-9359,Vol.6,Issue 4,April 2017.

[8] Madhavi Vaidya,"Parallel Processing of cluster by MapReduce",IJDPS,Vol.3,No.1,2012.

[9] Jeffrey Dean, Sanjay Ghemawat, " MapReduce: Simplified Data Processing on Large Clusters", Google, Inc, OSDI 2010.

[10] Harshawardhan S. Bhosale, Devendra P. Gadekar, " A Review Paper on Big Data and Hadoop",IJSRP,ISSN:2250-3153,Vol 4,Issue 10,Oct2014.

MapReduce

MapReduce is a framework using which we can write applications to process huge amounts of data, in parallel, on large clusters of commodity hardware in a reliable manner.

What is MapReduce?

MapReduce is a processing technique and a program model for distributed computing based on java. The MapReduce algorithm contains two important tasks, namely Map and Reduce. Map takes a set of data and converts it into another set of data, where individual elements are broken down into tuples (key/value pairs). Secondly, reduce task, which takes the output from a map as an input and combines those data tuples into a smaller set of tuples. As the sequence of the name MapReduce implies, the reduce task is always performed after the map job.

The major advantage of MapReduce is that it is easy to scale data processing over multiple computing nodes. Under the MapReduce model, the data processing primitives are called mappers and reducers. Decomposing a data processing application into mappers and reducers is sometimes nontrivial. But, once we write an application in the MapReduce form, scaling the application to run over hundreds, thousands, or even tens of thousands of machines in a cluster is merely a configuration change. This simple scalability is what has attracted many programmers to use the MapReduce model.

The Algorithm

Generally MapReduce paradigm is based on sending the computer to where the data resides!

MapReduce program executes in three stages, namely map stage, shuffle stage, and reduce stage.

Map stage : The map or mapper's job is to process the input data. Generally the input data is in the form of file or directory and is stored in the Hadoop file system (HDFS). The input file is passed to the mapper function line by line. The mapper processes the data and creates several small chunks of data.

Reduce stage : This stage is the combination of the Shuffle stage and the Reduce stage. The Reducer's job is to process the data that comes from the mapper. After processing, it produces a new set of output, which will be stored in the HDFS.

During a MapReduce job, Hadoop sends the Map and Reduce tasks to the appropriate servers in the cluster.

The framework manages all the details of data-passing such as issuing tasks, verifying task completion, and copying data around the cluster between the nodes.

Most of the computing takes place on nodes with data on local disks that reduces the network traffic.

After completion of the given tasks, the cluster collects and reduces the data to form an appropriate result, and sends it back to the Hadoop server.

MapReduce Algorithm

Inputs and Outputs (Java Perspective)

The MapReduce framework operates on <key, value> pairs, that is, the framework views the input to the job as a set of <key, value> pairs and produces a set of <key, value> pairs as the output of the job, conceivably of different types.

The key and the value classes should be in serialized manner by the framework and hence, need to implement the Writable interface. Additionally, the key classes have to implement the Writable-Comparable interface to facilitate sorting by the framework. Input and Output types of a MapReduce job: (Input) <k1, v1> -> map -><k2, v2>-> reduce -><k3, v3>(Output).

Input	Output	
Map	<k1, v1>	list (<k2, v2>)
Reduce	<k2, list(v2)>	list (<k3, v3>)

Terminology

PayLoad - Applications implement the Map and the Reduce functions, and form the core of the job.

Mapper - Mapper maps the input key/value pairs to a set of intermediate key/value pair.

NamedNode - Node that manages the Hadoop Distributed File System (HDFS).

DataNode - Node where data is presented in advance before any processing takes place.

MasterNode - Node where JobTracker runs and which accepts job requests from clients.

SlaveNode - Node where Map and Reduce program runs.

JobTracker - Schedules jobs and tracks the assign jobs to Task tracker.

Task Tracker - Tracks the task and reports status to JobTracker.

Job - A program is an execution of a Mapper and Reducer across a dataset.

Task - An execution of a Mapper or a Reducer on a slice of data.

Task Attempt - A particular instance of an attempt to execute a task on a SlaveNode.

Example Scenario

Given below is the data regarding the electrical consumption of an organization. It contains the monthly electrical consumption and the annual average for various years.

Jan	Feb	Mar	Apr	May	Jun	Jul	Aug	Sep	Oct	Nov	Dec	Avg
1979	23	23	2	43	24	25	26	26	26	26	25	26
1980	26	27	28	28	28	30	31	31	31	30	30	29
1981	31	32	32	32	33	34	35	36	36	34	34	34
1984	39	38	39	39	39	41	42	43	40	39	38	40
1985	38	39	39	39	39	41	41	41	00	40	39	45

If the above data is given as input, we have to write applications to process it and produce results such as finding the year of maximum usage, year of minimum usage, and so on. This is a walkover for the programmers with finite number of records. They will simply write the logic to produce the required output, and pass the data to the application written.

But, think of the data representing the electrical consumption of all the largescale industries of a particular state, since its formation.

When we write applications to process such bulk data,

They will take a lot of time to execute.

There will be a heavy network traffic when we move data from source to network server and so on.