**M S Ramaiah Institute of Technology**

(An Autonomous Institute, Affiliated to VTU)

MSR nagar, MSRIT post, Bangalore-54

A Dissertation Report on

**Implementation of a Map reduce based context aware recommendation for HEALTH CARE**

Submitted by

**Bharatkumar M 1MS12CS017**

**Bailore Avinash 1MS12CS016**

**Raghavendra G Naik 1MS12CS081**

**Chandrashekar B 1MS13CS405**

*in partial fulfillment for the award of the degree of*

# *Bachelor of Engineering in Computer Science & Engineering*



**DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING**

**M.S.RAMAIAH INSTITUTE OF TECHNOLOGY**

**(Autonomous Institute, Affiliated to VTU)**

**BANGALORE-560054**

[www.msrit.edu](http://www.msrit.edu), **May 2015**

# Abstract

Ubiquitous healthcare systems will monitor patients as they maintain their normal everyday activities, in order to warn the patients or healthcare workers of problems as well as collecting data for trend analysis and medical research. A compression of sensing data may be a solution, but it can also cause an increment of processing time and unexpected errors in healthcare service. So here we  to solve the problem propose a massive data management and analysis solution based on Hadoop to archive better performance, scalability and fault tolerance.

#### Contents

***Declaration i***

***Acknowledgements ii***

***Abstract iii***

1 **INTRODUCTION**

* 1. General Introduction………………. **<Page No>**
  2. Statement of the Problem…………..
  3. Objectives of the project……………
  4. Project deliverables……………
  5. Current Scope………………………
  6. Future Scope……………………….

1. **PROJECT ORGANIZATION**
   1. Software Process Models
   2. Roles and Responsibilities
2. **LITERATURE SURVEY**

3.1….Introduction

3.2…Main Body

3.3 Conclusion of Survey

1. **SOFTWARE REQUIREMENT SPECIFICATIONS**

5.1 Product Overview

5.2 External Interface Requirements

5.2.1 User Interfaces

5.2.2 Hardware Interfaces

5.2.3 Software Interfaces

5.2.4 Communication Interfaces

5.3 Functional Requirements

5.3.1 Functional Requirement 1.1

:

5.3.n Functional Requirement 1.n

5.4 Software System Attributes

5.4.1 Reliability

5.4.2 Availability

5.4.3 Security

5.4.4 Portability

5.4.5 Maintainability

5.4.6 Performance

5.5 Performance Requirements

5.6 Database Requirement

5.7 Design Constraints

5.8 Other Requirements

1. **DESIGN**
   1. Introduction
   2. Architecture Design
   3. Graphical User Interface
   4. Class Diagram and Classes (represent Inheritance, Aggregation and Association)
   5. Sequence Diagram
   6. Data flow diagram
   7. Metric calculation
2. **IMPLEMENTATION**
   1. Tools Introduction
   2. Technology Introduction
   3. Overall view of the project in terms of implementation
   4. Explanation of Algorithm and how it is been implemented
   5. Information about the implementation of Modules
3. **TESTING**

**7.1** Results and Snapshots

1. **CONCLUSION & SCOPE FOR FUTURE WORK**
2. **REFERENCES**
3. **INTRODUCTION**
   1. General Introduction

In this project we are doing an overall analysis of the healthcare dataset .

* 1. Problem Statement

Implementation of a Map reduce based context aware recommendation engine for HEALTH CARE

1.3 Objectives of the Project

Monitor patients as they maintain their normal everyday activities, in order to warn the patients or healthcare workers of problems as well as collecting data for trend analysis and medical research

1.4 Project Deliverables

The project delivers the interface which gives the detailed and analyzed data of the health care as output.

1.5 Current Scope

Health care management has lot of scope in the modern era. The present hospitals has data that has and the number of patients are increasing day by day.At this moment, the code can handle the analysis part with a very good accuracy. But there are a few areas which have a lot of scope in this aspect.

1.6 Future Scope

Nowadays big data has become the buzzword in IT industry organizations. The need of analysing and processing of information has grown a lot. This paper implemented the analysing of big data Further analysis can be done to images and all types of other files based on index support.,so that industries make the data easily available to people who is using and trying accessing such type of data.

1. **PROJECT ORGANIZATION**

2.1 Software Process Models

**Agile Model**: We choose to follow it based on the project size, complexity and duration. The agile methodology is a defined process that makes it easier for a project team to deal with complexity and coherence using an incremental delivery approach. If project requirements are clearly defined and understood and expected project duration is 3 to 4 months and hence we suggest iterative and incremental development model with periodic reviews.

1. **LITERATURE SURVEY**

3.1 Introduction

The healthcare industry is producing massive amounts

of multimodal data. More and more patients and professionals

upload medical data to applications such as PHRs

( personal health records) and EMRs (electronic medical

records). Medical imaging procedures are becoming increasingly

popular each year with more than 60 million CT scans

performed per year [1]; BioScience (drug discovery, genomics,

bioassay) applications also deal with large amounts of data.

The need for parallel processing is apparent for mining these

massive multimodal data sets, which can range anywhere

from tens of gigabytes, to terabytes or even petabytes.

3.2 Main Body

Hadoop supports the MapReduce model, which was introduced by Google as a method of solving a class of petascale problems with large clusters of inexpensive machines. The model is based on two distinct steps for an application:

• Map:

An initial ingestion and transformation step, in which individual input records can be processed in parallel.

• Reduce:

An aggregation or summarization step, in which all associated records must be processed together by a single entity.

3.3 Conclusion of Survey

This project gave us hands on experience of handling and parallel processing of huge amount of data. Data collection process introduced us to java twitter streaming API. It was very interesting to gather and then aggregate the social networking data so as to extract interesting patterns and recent trends from it. We got exposure to work with prominent parallel data processing tool: Hadoop. Apache Hadoop framework is gaining significant momentum from both industry and academia as the volume of data to analyze growth rapidly.

1. **SOFTWARE REQUIREMENT SPECIFICATIONS**

4.1 Product Overview

The project involves building of a system that can be used to display the data on health care. It is implemented using Hadoop as a platform where the dataset is stored and uses Mapreduce parallel programming concept.

The final output is visualized, where output shows the details of patients and their details. This is implemented using Mapreduce.

* 1. External Interface Requirements

4.2.1 User Interfaces:

The interface will meet the following requirements to conform the required patient details It will be simple and easy to understand. Controls which allow the user to interact with the application will be clear and imply their functionality within the application.

The user interface of eclipse was simple to navigate and easy to understand.

* + 1. Hardware Interfaces:

The application will have special access to the doctors who can access the confidential information. No further hardware devices or interfaces will be required for this analysis.

* + 1. Software Interfaces:

**Inputs**

The java program takes the dataset as the input.

**Outputs**

Representation of the detailed data on the required patient or anything is the output

4.3 Functional Requirements

4.3.1 Retrieving Input:

4.3.2 Data set pre-Processing:

4.3.3 Reduced data on the patient

* + 1. Output :

4.4 Software System Attributes

4.4.1 Reliability:

The project will meet all of the functional requirements without any unexpected behavior.

4.4.2 Availability:

The project will be available on demand as it is important to all the people. The functionality of the project will not depend on any external services like internet

* + 1. Security:

The project will never disclose any personal information of patients and will collect no personal information from its own users.

* + 1. Portability:

This project is designed to run on any operating system.

* + 1. Maintainability:

The project should be written clearly and concisely. The code will be well documented. Particular care will be taken to design the project modularly to ensure that maintenance is easy.

4.5 Performance Requirements

4.5.1 Real-Time:

The project will provide up-to-date information on the patients and their diseases.

4.5.2 System Resource Consumption:

Resource consumption of this application should not reach an amount that affects the normal process of system. The application should be capable of operating in the background.

4.6 Database Requirement

The Hadoop S3 database is used to store the huge datasets.

4.7 Design Constraints

The constraints that affect the design of this project are the cost and the performance. The design of this project is cost-effective and the performance is high as parallel programming is used using HPC tool like Hadoop.

4.8 Other Requirements

Knowledge on diseases and how to use the interface to assess the information

1. **DESIGN**

5.1 Introduction

5.1.1 Number of Modules: 4

5.1.2 Modules Description:

* **Fetching and Extracting Data:**

**1. Data Preprocessing:**

Data set is pre-processed using eclipse IDE.

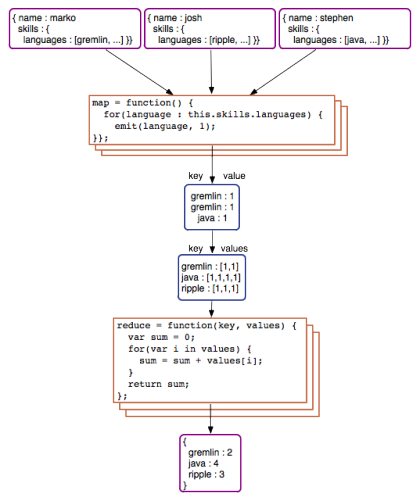
**2. Classification:**

Classification will be performed on the user input using the interface . The data analysis will provide a negative, neutral, or positive numeric value.

**3. Analysis:**

The Preprocessed data is considered for analysis. Based on the number of words that the input maps in the data which will be reduced and required output is presented

5.1.3Algorithm Design:



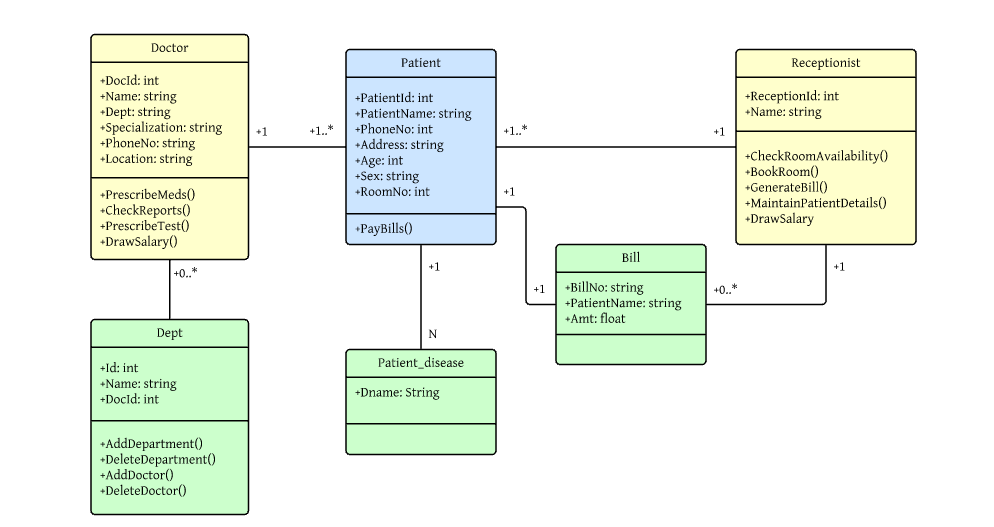
5.2 Architecture Design

****

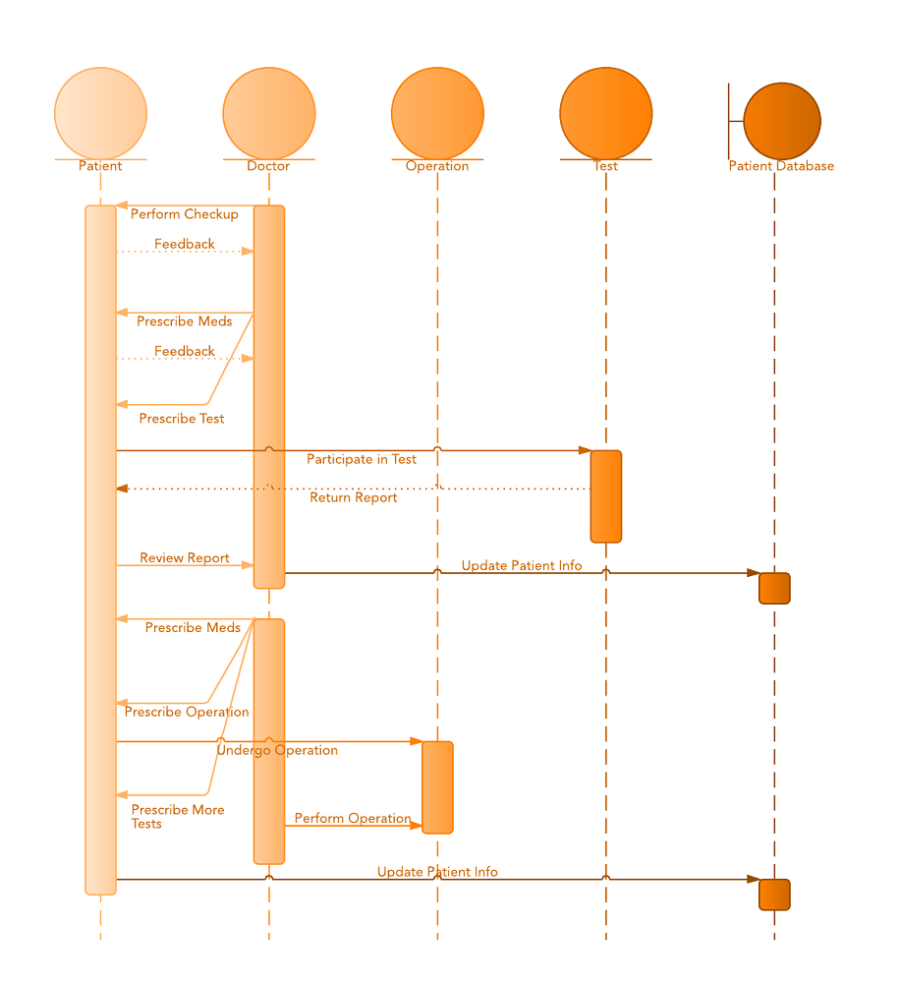
5.3 Graphical User Interface

The interface will meet the following requirements to conform to the users’ needs. It will be simple and easy to understand. Controls which allow the user to interact with the application will be clear and imply their functionality within the application. The user interface of eclipse was simple to navigate and easy to understand.

5.4 Class Diagram

****

5.5 Sequence Diagram

****

**6. IMPLEMENTATION**

6.1 Tools Used:

Hadoop, Eclipse , Java, Idle

6.2 Technology Used:

Mapreduce concept, Java

6.3 Overall view of the project in terms of implementation

The Project is implemented as follows:

* Health care Data Set Identification and Extraction
* Pre-processing of data set
  + Removing unwanted attributes
  + Eliminating stop words
  + Present the required data

6.4 Explanation of Algorithm and how it is been implemented

**Our Approach In Implementation**:

 Mapper computes a given function for each item and emits value of the function as a key and item itself as a value. Reducer obtains all items grouped by function value and process or save them. In case of inverted indexes, items are terms (words) and function is a document ID where the term was found.

i. **Stop words removal:**

The stop words like a, an , this which are not useful in performing the analysis are removed in this phase. Stop words are removed using java in eclipse platform. All the words are not considered are not considered for analysis.

ii. **Map reduce using Hadoop:**

. The MapReduce operates exclusively on < key, value>,that is the

framework views the input to the jo as a set of <key, value> pairs and

produces set of <key, value> pairs as output.

6.5 Information about the implementation of Modules

Number of Modules: 4

* Fetching and Extracting Data:
* Data Pre-processing:
* Classification:
* Analysis

**7. TESTING**

7.1 Results and Snapshots

After streaming the tweets into HDFS in real time, Naïve bayes is used in analyzing the tweets. Tweets are tagged as documents where categories are the hash tags defined in the configuration file. Later the tweets are grouped as positive, negative and neutral based on subjectivity corpus forming a dictionary of words and its polarity. The sample example shown in pie chart which is obtained by running the map reduce code in hadoop.

**GRAPHICAL REPRESENTATION OF NATURE OF USER** **TWEETS**

The sample count or the the given above chart is shown in the following table:

|  |  |
| --- | --- |
| Opinion | Count |
| Positive | 25540 |
| Negative | 20765 |
| Neutral | 76586 |

**GRAPHICAL REPRESENTATION OF NATURE OF SINGLE USER** **TWEET**

The sample count or the the given above chart is shown in the following

table:

|  |  |
| --- | --- |
| Opinion | Count |
| Positive | 313 |
| Negative | 152 |
| Neutral | 555 |

**8. CONCLUSION & SCOPE FOR**

**FUTURE WORK**

This project gave us hands on experience of handling and parallel processing of huge amount of data. Data collection process introduced us to java twitter streaming API. It was very interesting to gather and then aggregate the health data so as to extract interesting patterns and recent data from it. We got exposure to work with prominent parallel data processing tool: Hadoop. Apache Hadoop framework is gaining significant momentum from both industry and academia as the volume of data to analyze growth rapidly.

This project helped us not only to gain knowledge about installation and configuration of hadoop distributed file system but also map reduce programming model. At the end of analysis phase data visualization was performed with the help of Google Developer. Amongst the many fields of analysis, there is one field where humans have dominated the machines more than any – the ability to analyze sentiment, or sentiment analysis.

**9. REFERENCES**

* <https://archive.org/download/twitter_cikm_2010>
* <http://www.cloudera.com/content/www/enus/documentation/other/tutorial/CDH5/Hadoop-Tutorial/ht_example_4_sentiment_analysis.html>
* <http://www-nlp.stanford.edu/courses/cs224n/2009/fp/3.pdf>
* <http://www.ncbi.nlm.nih.gov/pmc/articles/PMC4224309/>
* http://www.google.com
* <https://highlyscalable.wordpress.com/2012/02/01/mapreduce-patterns/>
* <http://www.tutorialspoint.com/hadoop/hadoop_mapreduce.htm>
* <https://www.mapr.com/blog/how-write-mapreduce-program>