Stock Price Prediction Using Distributive Computing

Thesis
Submitted in partial fulfilment of the requirements for the degree of

MASTER OF TECHNOLOGY
in
COMPUTER SCIENCE

by
Ravita Meena
(Roll N0.: 2K16/CSE/10)

Under the supervision of:
DR.RAJNI JINDAL

DEPARTMENT OF COMPUTER SCIENCE
DELHI TECHNOLOGICAL UNIVERSITY
NEW DELHI - 110042

June 2018

DECLARATION

I hereby *declare* that the Report of the Post Graduate Project Work entitled "**Stock Price Prediction Using Distributive Computing** ", which is being submitted to **Delhi Technological University, Delhi**, in partial fulfillment for the requirements of the award of degree of **Master of Technology** in **Computer Science** in the **Department of Computer Science**, is a *bonafide report of the work carried out by me.* The material contained in this Report has not been submitted at any other University or Institution for the award of any degree.

Place: DTU, Delhi
Date: 13-June-2018

Ravita Meena
(Roll. No.: 2K16/CSE/10)

CERTIFICATE

This is to certify that the Post Graduate Project Work Report entitled **”Stock Price Prediction Using Distributive Computing”** submitted by **Ravita Meena** , (Roll Number: 2K16/CSE/10) as the record of the work carried out by her, is accepted as the Post Graduate Project Work Report submission in partial fulfilment of the requirements for the award of degree of **Master of Technology** in the **Computer Science** in the **Department of Computer Science** at **Delhi Technological University, Delhi** during the academic year 2016-2018.

Place: DTU Delhi
Date: 12-June-2018

Signature of the Guide

Name of the Guide: Dr Rajni Jindal
Designation of the Guide: Head of Department
Name and Address of Organization:DTU,Delhi

ACKNOWLEDGEMENT

The success of a project requires help and contribution from numerous individuals and the organization. Writing the report of this project work gives me an opportunity to express my gratitude to everyone who has helped in shaping up the final outcome of the project.

I express my heartfelt gratitude to my project guide **Dr.Rajni Jindal** for giving me an opportunity to do my major project work under her guidance. Her constant support and encouragement has made me realize that it is the process of learning which weighs more than the end result. I am highly indebted to the panel faculties during all the progress evaluations for their guidance, constant supervision and for motivating me to complete my work. They helped me throughout by giving new ideas, providing necessary information and pushing me forward to complete the work.

I express my deep gratitude to **Prof. Dr. Rajni Jindal**, Head of the Department of Computer Science, Delhi Technological University, Delhi for her constant co-operation, support and for providing necessary facilities throughout the M. Tech. programme.

I also reveal my thanks to all my classmates and my family for constant support.

Ravita Meena

Abstract

The stock market, equity market or share market is the aggregation of buyers and sellers (a loose network of economic transactions , not a physical facility or a discrete entity) of stocks (also called shares); these may include securities listed on a stock exchange as well as those only traded privately. In today's world Stock Market is a great factor in determining the state of the economy of a country. The movement of share market determines the movement of economy. Hence everyone tries to determine the movement of market on a particular day.

Here we present a methodology of finding the important stocks for day to day traders with reduced time and considerable accuracy. The data sets being used are a subset of published stock data by the Bombay Stock Exchange(BSE) and National Stock Exchange(NSE) in the past one year. We collected the sentiment score of a stock. Sentiment score is a score which gives us an idea about how people feel about buying a particular stock. We used www.moneycontrol.com for finding the sentiment score associated with a particular stock, which was further used for training the neural network model. The neural network would work as a non-linear binary classifier that would predict the progress of the company using the data for the present day.

Due to the large amount of data required and quicker real-time feedbacks to be supplied, we will be using Apache Spark. Apache Spark is an open source cluster computing framework for implementing distributed computing. In this framework a bigger job is distributed among a group of workers so as to reduce the actual time required for the job. With the help of Apache Spark and neural network, we would be able to find a list of companies listed on the stock market that have the tendency to go up in the day in a very short period of time.

Contents

List of Figures

List of Tables

Chapter 1

INTRODUCTION

With the rapid growth in the world economy the importance of stock market is increasing day by day. People are investing more and more money in stock market each day with the aim of getting better returns at the end. In such an environment where a lot depends on the stock market, forecasting the values of stock in a short period of time seems to be a good idea.

1.1 Importance Of Stock Price Prediction

A stock market, equity market or share market is the aggregation of buyers and sellers (a loose network of economic transactions, not a physical facility or discrete entity) of stocks (also called shares); these may include securities listed on a stock exchange as well as those only traded privately. Forecasting of stock prices means predicting what would be the value of a particular stock on a particular day. successful prediction of these value yields great profit to the individual or the organization . In today's world the impact of stock market is global. The movement of stock market determines the movement of large multinational as well as world economies. For example the Wall Street Crash of 1971 led to the Great Depression of 1971.

Hence a large amount of money, time and efforts are being invested in predicting the stock price so as to get good returns and profit. Various researchers have come up with different methodologies for determining the stock prices using previous day data. Some considered simple factors while others considered a set of factors for determining the stock prices.

Simple data mining techniques like K-means as well as complex data mining techniques like Artificial Neural Network has been used to accurately find the stock prices for various companies.

1.2 Challenges Of Stock Price Prediction

There are various challenges faced by the researchers when trying to correctly predict the stock prices for a large number of companies. Some of them have been listed down

- Determining Input Factors

  The most important challenge in determining the stock price for companies is to determine what should be the input for the prediction algorithm. Some researchers believed a single input like closing price is sufficient while others believed that an set of input like closing price along with open price, high price and low price for the day should be considered. Some researchers believed that even human sentiments play an important role and thus should be considered along with various other input factor.

- Determining Learning Algorithm

  A large variety of learning algorithm is available today. The learning algorithm vary from simple and linear algorithm like K-Means to complex and non linear algorithm like Artificial Neural Network.So we have to choose any suitable learning algorithm or a hybrid of 2-3 algorithm , depending on our input factor as to get the best accuracy.

- Non-Linear Or Heterogeneous Nature Of Market

  Another factor with respect to forecasting stock values is the non-linear or heterogeneous nature of the market. According to "Efficient Market Hypothesis (EMH) " in an efficient market, stock market prices fully reflect available information about the market and its constituents and thus any opportunity of earning excess profit ceases to exist. So it is ascertain that no system is expected to outperform the market predictably and consistently. Hence it is quite difficult to create a model that will fully fit the stock market data that is provided to it . even if it fits the model accurately there are significant chances that it would predict wrongly for a new input.

- Large Time Consumption

  Another important challenge in predict the stock prices is the amount of time taken for learning and prediction. Many efforts have been taken to correctly predict the market values but they all take a huge amount of time. More complex the algorithm more the time taken for training and prediction .But since the stock market is an ever changing environment hence time also plays a crucial factor in this. The market Hence efforts should be taken to reduce the time required for this entire process. This would give a huge advantage to the day to day traders as they could decide which shares are of more importance at the start of the day only. Hence we would be parallelizing the task of training the model in order to reduce the time for this.

1.3 Sentiment Analysis

Sentiment analysis is the branch of computer science that uses natural language processing and text analysis for the purpose of determining opinions or attitude of the speaker with respect to a particular topic. People Sentiment also play an important role in determining the value of stock on a particular day. People opinion towards a particular company has huge impact on its stock prices.For example, the news of death of Steve Jobs, created a shock among the Apple Inc. investors and hence the stock price of Apple took a steep dive the very next day as most of the people associated the success of Apple Inc. with Steve Jobs. Hence if a large majority of people tend to sell the stock of a particular company then there is a chance that the prices might go down. Hence the sentiment part related with stock market prediction can't be ignored. We have considered human sentiments as one of the factors in determining volatile stocks for the day

1.4 Artificial Neural Network

There are a wide variety of learning algorithm available now. But we have used Artificial Neural Network in our project.An artificial neuron network (ANN) is a computational model based on the structure and functions of biological neural networks. Information that flows through the network affects the structure of the ANN because a neural network changes - or learns, in a sense - based on that input and output.Artificial neural network are considered nonlinear statistical data modeling tools where the complex relationships between inputs and outputs are modeled or patterns are found.
The entire neural network is defined into 3 layers namely

- Input layer
  The input layer is only responsible for taking the input from the user and providing it to the next layer of neuron i.e hidden layer for computation.There can be only one input layer in the entire neural network and it can either contain 1 or more than 1 neuron depending on the number of inputs being considered. The input layer is always the first layer of the neural network

- Hidden layer
  The layer of neurons present between the input layer and the output layer is known as hidden layer. The hidden layer is responsible for providing the non-linear relationship to the neural network due to the presence of activation function in it. All the computation are done in this layer only. There can be any number of hidden layers each having any number of neurons in it

- Output layer
  the output layer is responsible for providing the output either to the user or back to the neural network for further computation. Error for weight updation are also calculated in this layer only.There can be only one output layer in the entire neural network and it can contain either one or more than one output neurons depending on our problem statement.

The advantages of using artificial neural network over other algorithm are-

- Varying Number Of Input

  since we know that the number of inputs play a very important role in the predicting stock prices hence we choose a learning algorithm that can take one or more than one input factors. In comparison to other model like ARIMA which can take only 1 input, the neural network is capable of having any number of input variable.

- Detecting Non-Linear relationship

  Since we don't know the exact nature of relationship present between the input variable and output , hence we want an algorithm that can work for both linear and non linear relationship. Hence in comparison to other learning algorithm like K-Means which can detect only linear relationship, neural network can work both both scenario and can determine the nature of relationship on its own.

Hence due to the following reasons we have used artificial neural network in our project work. The artificial neural network is used to create a model that can train from the historic data and the provide us with a output.

1.5 Distributive Framework

One of the major challenge in determining the stock prices is the huge amount of time taken for the same. lets suppose even if the training and prediction of stock price of a single company takes 1 second then even if have only 8000 companies then it would take more than 2 hours to calculate the prices for all of them. But since the stock market is an ever changing environment hence a lot would have been changed in these 2 hours and hence these results might be of no use.

Hence there was a need of a proper distributive framework that could reduce the time taken for determining the stock prices for all the companies by breaking the larger jobs into smaller jobs that could be executed parallely. Apache Spark is one of the distributive framework that is being put into use now-a-days. Apache Spark is an open source cluster computing framework. In this framework a bigger job is distributed among a group of PC's so as to reduce the actual time required for the job.

Figure 1.1 shows the basic architecture of spark consisting of 1 driver node and 2 worker node. The driver node is also known as the master node and the worker node is also known as the slaves. There can be only 1 master node and as many slave nodes as you like. The master mode is responsible for distributing the job to the worker node and doesn't actually perform the job while the worker node takes the job performs it and provides the result back to worker node for further calculation. The executor inside the worker node is actually responsible for performing the task provided to the worker node. The executor is also responsible for task parallelism. For example if we are able to spawn three executor on a single worker node then we can run three small task parallely on the worker node. The number of executor decide the parallelism inside the worker node and the number of worker node decide the parallelism on the overall scale. Since the worker node doesn't have any executor hence it can't actually do any job.

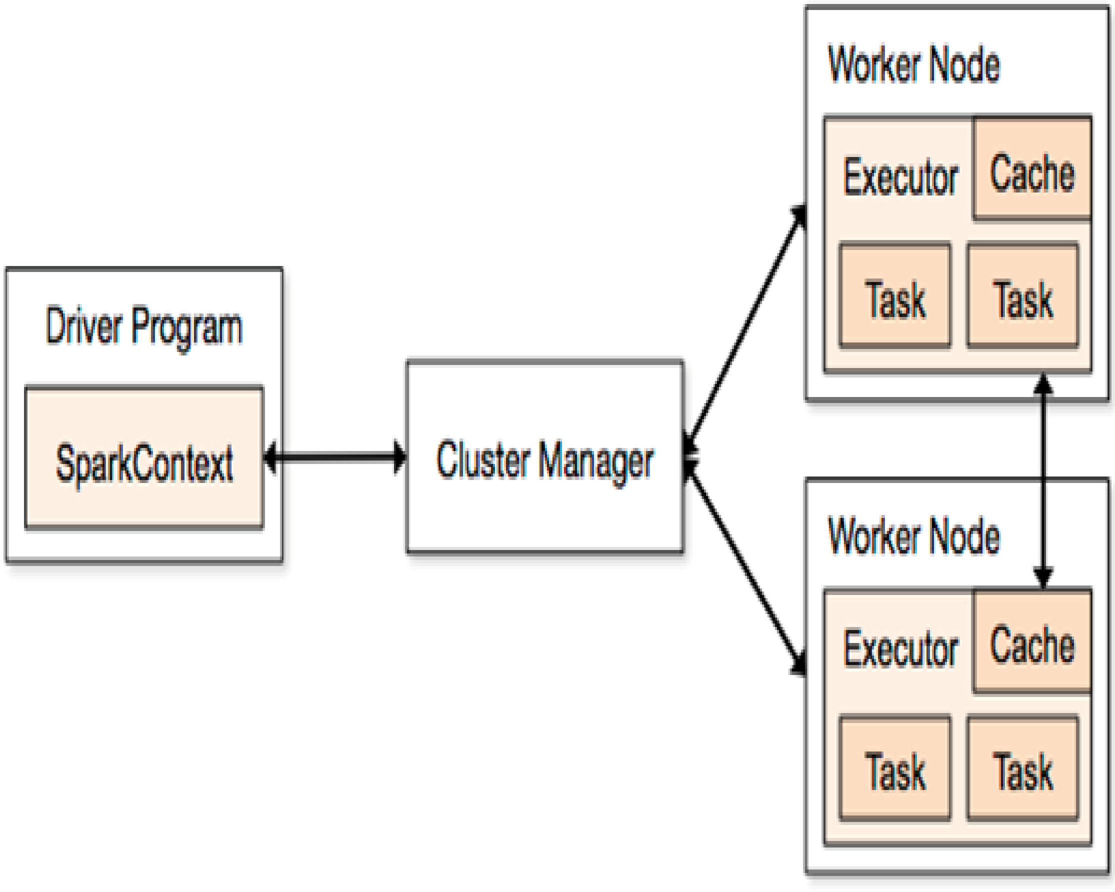


Figure 1.1: Spark Architecture[14]

The cluster manager is a resource manager program that is responsible for managing the resources of the slave node so as to reduce the time as much as possible. The various resource manager available for Apache Spark are

- Spark own resource manager available in spark standalone mode
- Yarn Resource Manager
- Mesos Resource Manager

The main difference between these resource manager is how actually they spawn the executors on the worker node. The maximum number of executors that can be spawned on a worker node using Spark's resource manager depends on the number of processors in the system while in Yarn and Mesos it depends on the total RAM availabe to the system.For example if we have 64 GB Ram then we can spawn a maximum of 8 executors on each system (8GB each) using Spark's resource manager while we can spawn even 64 executor(1GB each) using Yarn resource manager. The master node is always responsible for taking the job from the user and providing the output back to the user.

There are some other distributive computing framework available in the market like Hadoop, Apache Storm and many more. But Apache Spark has some significant advantages over the other. Some of them have been listed below

- In Memory Processing

  Apache spark uses in memory processing i.e it keeps the intermediate computation in the main memory only. Other computing framework like Hadoop keeps on writing the intermediate results to the secondary memory and keeps fetching the result from there only. Since Apache Spark doesn't involve read and write from the secondary memory which is generally time consuming hence due to this fact it is generally faster than other distributive framework.

- Support For Large Number Of Languages

  Apache Spark provides support for large number of languages for example Java, Scala and Python whereas Hadoop provides support for only one language i.e Java

- Availability Of Data Science Tools

  Apache Spark provides support for a large number of pre built machine learning libraries and algorithms which is not available in Hadoop. Moreover it provides support for various R algorithm.

- Availability Of Master Node

  There was no central point of control in Hadoop but in Apache Spark with the presence of Master node there is a central point of control i.e the master node is responsible for distributing the jobs.

The entire thesis work is divided into literature review, methodology, experimental result and analysis, conclusion and future work followed by references.

Chapter 2

LITERATURE REVIEW

2.1 Related Work

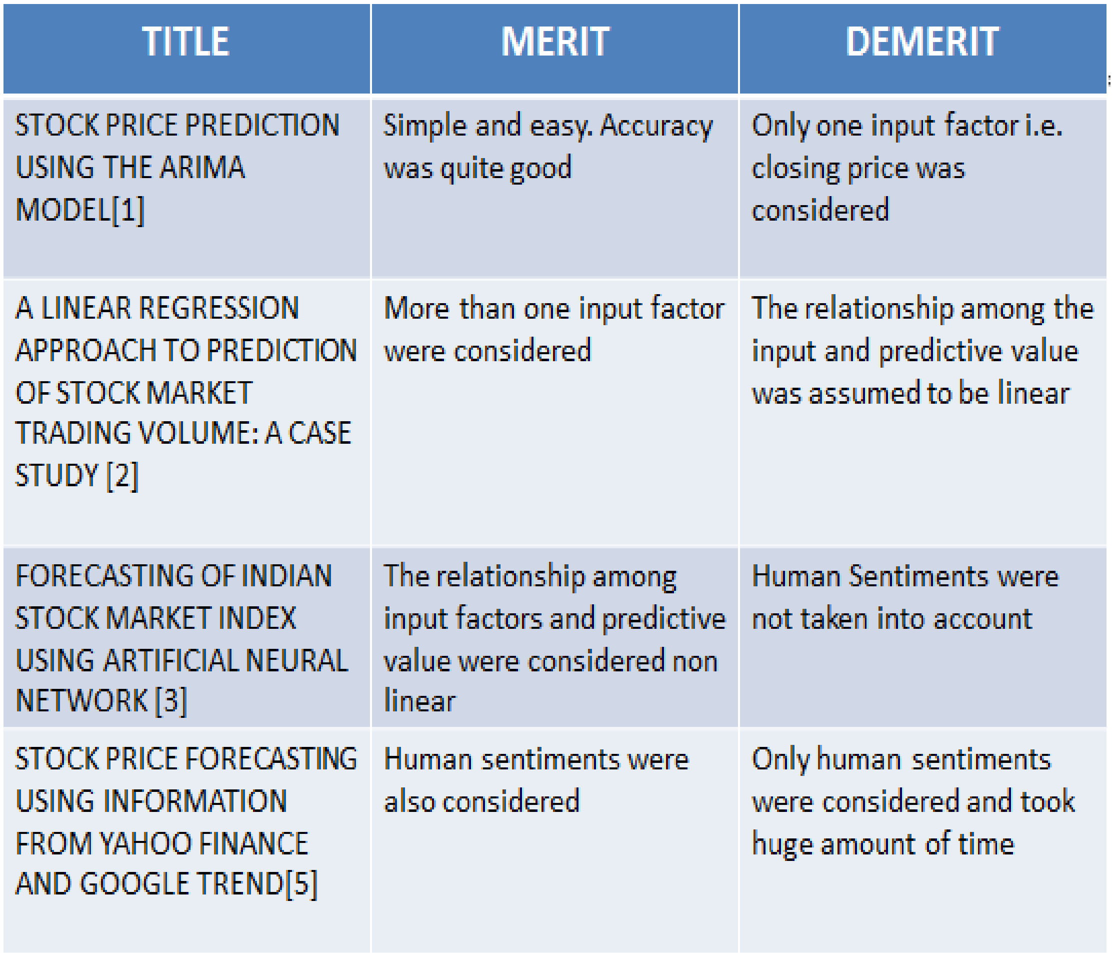
A lot of work has been done for forecasting the nature of market using various techniques.Some of them used simple linear model where they assumed linear relationship among the factors while some assumed a non-linear relationship between them. The later work even took into consideration the sentiments attached with a particular stock.

Initially most of the research about stock market prediction was done using autoregressive integrated moving average(ARIMA) model for price determination[1]. ARIMA is a generalized case of Autoregressive moving average model which is generally used for predicting the stock prices. The model used simple regression algorithm and simple input for finding the points in future. The drawback of this was that it generally considered only single input factor

Later other linear regression model were used that considered more than one input factor for price determination[2]. But the drawback remained that they still considered the relationship among the input factors are linear only. the disadvantage was that even if the model was fitted properly it gave significant error for new set of inputs

Next with the advancement of Deep learning techniques Artificial Neural Network were used in the area of stock market prediction[3]. the advantage was that it could consider even non-linear relationship among the input factors. But still one of the important factors i.e human sentiments were not considered over here. human sentiments could also

be an important factor in finding the movement of market as it is only people that are responsible for selling and buying shares. Hence a method was proposed[5] which took google news and yahoo finance data into consideration.
Still the time required for performing this methodology over a large amount of stock is significant. even if each stock takes 10 seconds for execution, the total time for 500 stocks would be around 83 minutes which is quite significant for day to day traders as the trend of market might change in an entire one hour. Hence in this project we aim to find the significant stocks for day to day traders using sentiment analysis and Artificial Neuarl Network in an appropriate period of time



TITLE	MERIT	DEMERIT
STOCK PRICE PREDICTION USING THE ARIMA MODEL[1]	Simple and easy. Accuracy was quite good	Only one input factor i.e. closing price was considered
A LINEAR REGRESSION APPROACH TO PREDICTION OF STOCK MARKET TRADING VOLUME: A CASE STUDY [2]	More than one input factor were considered	The relationship among the input and predictive value was assumed to be linear
FORECASTING OF INDIAN STOCK MARKET INDEX USING ARTIFICIAL NEURAL NETWORK [3]	The relationship among input factors and predictive value were considered non linear	Human Sentiments were not taken into account
STOCK PRICE FORECASTING USING INFORMATION FROM YAHOO FINANCE AND GOOGLE TREND[5]	Human sentiments were also considered	Only human sentiments were considered and took huge amount of time

Figure 2.1: Literature Survey

The Fig 2.1 is a comparison study between various kinds of approaches followed by researchers. The first column in the table represents the title of the paper, the second represents the advantage that particular approach had and the last one represents the disadvantage of that particular method. the first entry is an example where the researchers used the ARIMA approach, the second one is an example of linear regression technique, the third one is an example of Artificial Neural Network and the last one uses Sentiment Analysis technique

2.2 Outcome Of Literature Survey

The following are the outcome of the literature survey that we have gone through -

- Many approaches are followed for calculating the semantic similarity between documents like ARIMA, linear regression, Neural Network
- The computation time required for forecasting the important stock is quite large.
- Not much work has been done towards parallelizing the entire approach of forecasting the important stocks.

2.3 Motivation

Most of the researchers on stock market prediction have either used linear model like ARIMA , non-linear model like artificial neural network or have used sentiment analysis for forecasting the values of stock. only some of them have considered both sentiment and machine learning techniques for forecasting stock. still the time required to gather information for a large amount of stock is great. But for day to day traders ”Time Is Money”.

Hence in the project we aim to reduce the time required for determining the important stocks for the day using both sentiment analysis and neural network within an appropriate amount of time

2.4 Problem Statement

Our aim is to reduce the computational time required for finding the volatile stocks for day to day trading

2.5 Research Objectives

- To implement sequential approach for finding volatile stocks for a day using artificial neural network
- To implement parallel approach for finding the volatile stocks for a day using artificial neural network
- To analyze the model for accuracy

Chapter 3

METHODOLOGY AND FRAMEWORK

3.1 Methodology

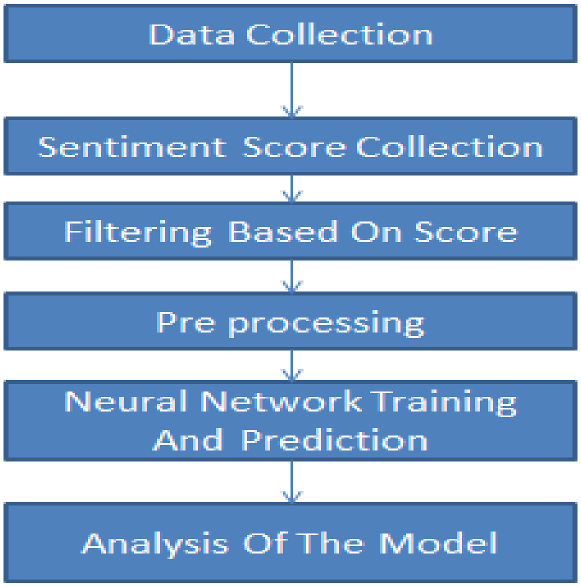


Figure 3.1: Overall flow of control

The following steps define the overall flow of work-

- In the initial step we collect the data for all the companies listed on moneycontrol.com
- In the next step we collect sentiment score for all the companies listed on moneycontrol.com for that particular day
- In the next step we do some filtering based on the sentiment score that has been calculated in the last stage.
- Next we do some preprocessing of the data collected in the step 1
- Next this preprocessed data is provided as an input to the neural network for training and testing purpose
- In the final stage we analyse the model based on various parameters.

3.1.1 Data Collection

This is the first phase of our methodology. In this phase we would be collecting the previous days data for all the companies that are listed on the www.moneycontrol.com. This website would be used for finding the opinion of people for a particular stock on a particular day. These data would be later used as an input to the neural network part. The data that would be collected are-

- Open Price-

  Open price is defined as the opening price i.e the price at which the market opened for that particular stock for that particular day
- High Price-

  High Price is defined as the highest value for that particular stock on that particular day.
- Low Price-

  Low Price is defined as the lowest value for that particular stock on that particular day. High and low price are used to calculate VARIATION which is defined as a measure of how much the value of the stock varied for that particular day. Mathematically it is high price - low price

- Close Price-

  Close price is defined as the closing price i.e the price at which the market closed for that particular stock for that particular day.

- Class Label-

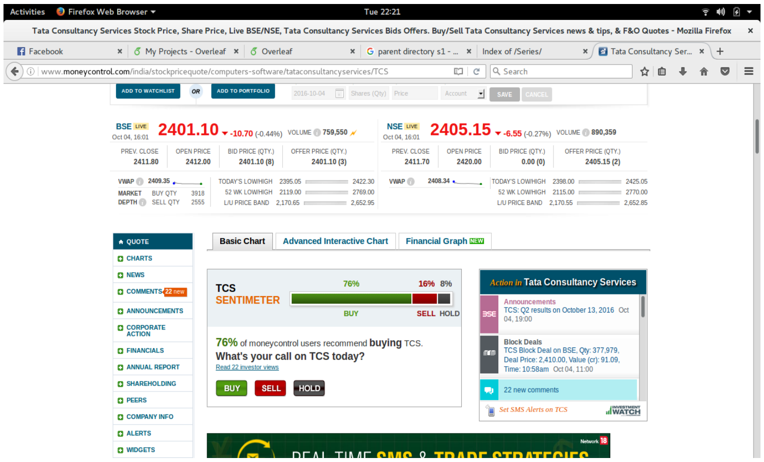
  Class label would represent whether the movement of market for that particular stock i.e whether the stock market went up or down for that particular stock. It would be a binary valued data where 0 represent a negative movement i.e the market went down and 1 represent a positive movement i.e the market went up for that particular stock

3.1.2 Sentiment Score Collection

The sentiment score is a score which gives us an idea about how people feel about buying a particular stock.We would be using www.moneycontrol.com for finding the sentiment score associated with a particular stock. www.moneycontrol.com provide a feature where people can vote whether they want to buy the stock, sell the stock or hold the stock for that particular day. The sentimeter which is available for each stock takes into consideration all the votes and then displays this data in form or a colored bar where different color represent different choices made by the people. Green represents the % of people that tend to buy the stock, red represents the % of people that tend to sell the stock and grey represents the % of people that tend to hold the stock for that day. The sentiment score is a score which gives us an idea about how people feel about buying a particular stock. It is based on the sentimeter which is available on the moneycontrol site which gives an idea about public opinion about a stock. The sentiment score would be an integer value between 0 and 100. A value of x would mean that x % of people have voted in favour of buying the stock.For finding the value we crawl through the particular url for that company and then get the value.

3.1.3 Filtering Based On Sentiment Score

After getting the sentiment score for all the companies the next phase would be filtering of these companies based on the sentiment score.Since the sentiment score represents the view of people about a particular stock hence if a company is having a low sentiment score then people are less likely to invest in that company. For example if a company has a sentiment score of only 10 then it means that only 10 % of people are interested in buying the stocks of that company. The rest either wants to sell or hold it. Hence the stock prices for that company are expected to either go down or remain constant. Hence there would be no point in making investment in these companies.Hence we can neglect companies having a sentiment score less than a predefined threshold as it wont be beneficial to us. For our experiment purpose we have defined the threshold to be 75 %.It means that We would be taking only those companies that have a sentiment score which is greater than 75 i.e 75 percent of the people are thinking about buying the stock on that particular day. The fig 3.1 depicts how an actual sentimeter of moneycontrol.com looks like



Tata Consultancy Services Stock Price, Share Price, Live BSE/NSE, Tata Consultancy Services Bids Offers. Buy/Sell Tata Consultancy Services news & tips, & F&O Quotes - Mozilla Firefox

www.moneycontrol.com/india/stockpricequote/computers-software/tataconsultancyservices/TCS

BSE LIVE Oct 04, 16:01	2401.10 -10.70 (-0.44%)	VOLUME 759,550	
PREV. CLOSE	OPEN PRICE	BID PRICE (QTY.)	OFFER PRICE (QTY.)
2411.80	2412.00	2401.10 (8)	2401.10 (3)

VWAP 2409.35
MARKET DEPTH BUY QTY 3918 SELL QTY 2555
TODAY'S LOW/HIGH 2395.05 2422.30
52 WK LOW/HIGH 2119.00 2769.00
L/U PRICE BAND 2,170.65 2,652.95

NSE LIVE Oct 04, 16:01	2405.15 -6.55 (-0.27%)	VOLUME 890,359	
PREV. CLOSE	OPEN PRICE	BID PRICE (QTY.)	OFFER PRICE (QTY.)
2411.70	2420.00	0.00 (0)	2405.15 (2)

VWAP 2408.34
TODAY'S LOW/HIGH 2398.00 2425.05
52 WK LOW/HIGH 2115.00 2770.00
L/U PRICE BAND 2,170.55 2,652.85

TCS SENTIMETER
76% BUY 16% SELL 8% HOLD

76% of moneycontrol users recommend buying TCS.
What's your call on TCS today?
Read 22 investor views

Action in Tata Consultancy Services

Announcements
TCS: Q2 results on October 13, 2016 Oct 04, 19:00

Block Deals
TCS Block Deal on BSE, Qty: 377,979, Deal Price: 2,410.00, Value (cr): 91.09, Time: 10:58am Oct 04, 11:00

22 new comments

Set SMS Alerts on TCS

Figure 3.2: Sentimeter[7]

3.1.4 Preprocessing Of Data

The data that has been collected in the previous stage is quite diverse and hence can't be provided directly as an input to the neural network. Hence we preprocess the data before providing it as an input to the neural network. The preprocessing involves

- Conversion Of Opening Prices Into Categorical Data-

  Since the opening price is a continous data i.e it can take any value hence we would convert it into categorical data so as to increase the accuracy of neural network.

  The first step towards converting it into categorical data would be converting the given data to a scale of [0,1]. For this we would be using Min Max scaler.The Min Max scaler is defined by the following equation

  $$X = \frac{X - X_{min}}{X_{max} - X_{min}} \tag{3.1}$$

  Here X would represent the current value and X_{min} and X_{max} represents the minimum and maximum value in that category respectively. After converting it into a range of [0,1] we would divide this range into equal interval as to represent various categories like very low open price i.e from 0 to 0.2 represented by 1, low open price i.e from 0.2 to 0.4 represented by 2, high open price etc. Now even this categorical data cant be provided as an input to the neural network because there is a kind of ordering between the categories i.e 2 is more important than 1 , 3 is more important than 2 and so on.

  Hence we use one hot encoding method to convert the categorical data so generated into a vector that can be provided as an input to the neural network for training and classification purpose. In one hot encoding method we create a vector of size equal to the number of categorical data and we keep only one of them as 1 and rest of them as 0. The one which is kept 1 represent the category. For example if we have 3 category and the data belongs to the 3^{rd} category then the equivalent one hot encoding represenation would be like [0,0,1].

- Converting Variation To Categorical Data-

  Variation of a stock is defined as how much it has varied during a day. Mathematically variation is defined as

  $$Var(STOCK_X) = high_{price}(STOCK_X) - low_{price}(STOCK_X) \tag{3.2}$$

  Since this variation is again a continous data i.e it can take any value. Hence we apply the same steps as in last section i.e Min Max scaling followed by converting into category and one hot encoding to convert this continous data into categorical data.Fig 3.3 represent a diagram of how we convert continous data into categorical data.

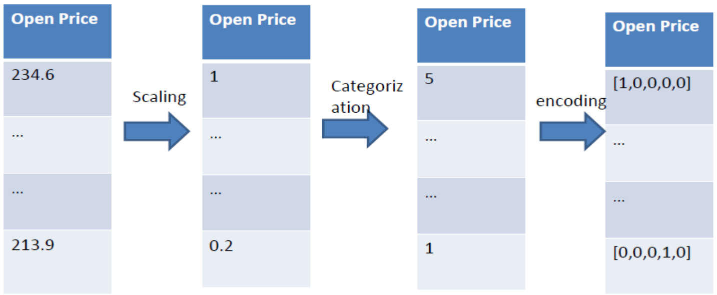


Figure 3.3: Preprocessing Of Open Price Data

3.1.5 Artificial Neural Network

After selecting the companies having a good sentiment score i.e sentiment score greater than a threshold , the next phase would be predicting whether these companies are likely to go up or down. Hence in this phase we would be would be training the neural network model using the data from the first phase only for the companies that were selected from the last phase. The neural network would work as a non-linear binary classifier that would predict the movement of the company using the data for the present day.

Fig. 3.4 represents the basic structure of a neural network consisting of a single input layer , single hidden layer and single output layer each containing some number of neurons. The neurons are responsible for taking the input and providing some output based on the activation function used. The different activation function that can be used are tanh, step function, logistic and many more. The choice of activation function depends on the problem

There are 2 main types of neural network

- Feed Forward Neural Network-

  The feed forward is a neural network in which the output goes only in the forward direction. There is no loop between the input and output. At each point of time the

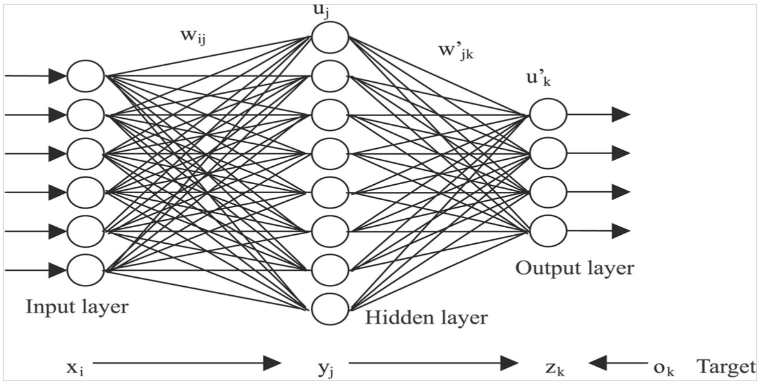


Figure 3.4: Structure Of Neural Network [13]

output depends only at the input at that time and not the output of the previous stage. Fig 3.4 depicts a simple feed forward neural network.

- Recurrent Neural Network-

  The recurrent neural network are used of time series analysis. In this type of neural network the output of the previous stage is provided as an input to the next stage. Hence there is a loop from output to input. At each point of time the output depends both on the input at that time and the output of the previous stage.

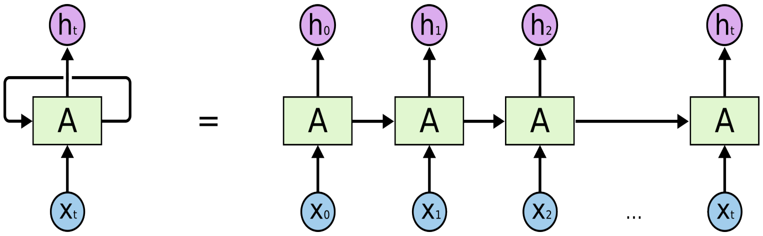


Figure 3.5: Structure Of Recurrent Neural Network [13]

  Fig. 3.5 represents a simple recurrent neural network. From the Fig we can clearly see that there is a loop from the output layer to the input layer. Hence the output at any particular time depends on both.

Backpropagation algorithm is used for weight updation after each iteration. We have used Feed Forward Neural Network with back propagation algorithm in our experiment. We have used this because we simply want the relationship between the various input parameters. For example if 2 years ago the opening price was 200 and the market went down and again today the opening price is 200 then the recurrent neural network would give 0 as output as it remembers the previous output. But the possibility of both 1 and 0 are their. Hence we have used simple neural network to determine the relationship between input parameters.

3.1.6 Analysis Of Model

After we have created the neural network we analyse the model for accuracy. We change various parameters of the neural network so as to see its impact on the accuracy of the neural network.

- Learning Rate-

  Learning rate in a neural network is used to control the change in the weight and bias after every iteration. Learning rate determines how much the weight should differ in the next iteration.

- Tolerance Value-

  Tolerance is defined as value which is used to control the termination condition of the neural network. If the difference in error in 2 consecutive iteration if less than the tolerance value than convergence is considered to be reached and hence the training stops .

- Number OF Hidden Layers-

  The neuron layer present between the input layer and the output layer is know as hidden layer in the neural network

In this stage we vary the value of various parameters of neural network so as to get the optimum set of parameters which would give us the best result. for example low activation function as well low tolerance value leads to better accuracy.

3.1.7 Apache Spark

Apache Spark is an open source cluster computing framework. In this framework a bigger job is distributed among a group of PC's so as to reduce the actual time required for the job. In our work we would be using Apache Spark to parallelise our approach. We have used Apache Spark at two different places in our approach.

- Sentiment Score Collection-

  The entire process of calculating the importance score for each stock is sequential in nature and can't be parallelised. But the importance score of 1 company is independent of the other. Hence the process of calculating the importance score for each company can be parallelised and run in parallel i.e. the importance score for more than 1 company would be calculated at any point of time.

- Neural Network Training And Testing-

  Since the historic data for each of the company is independent of one another i.e the historic data of company X doesn't depend on the historic data of company Y , hence the neural network for 2 different companies can be trained independently.

Apache Spark works entirely on the map reduce architecture. Fig 3.6 represents the basic map reduce architecture of Apache Spark. First the data values are mapped, then reduced and finally collected by the master node. The significance of each stage has been described below.

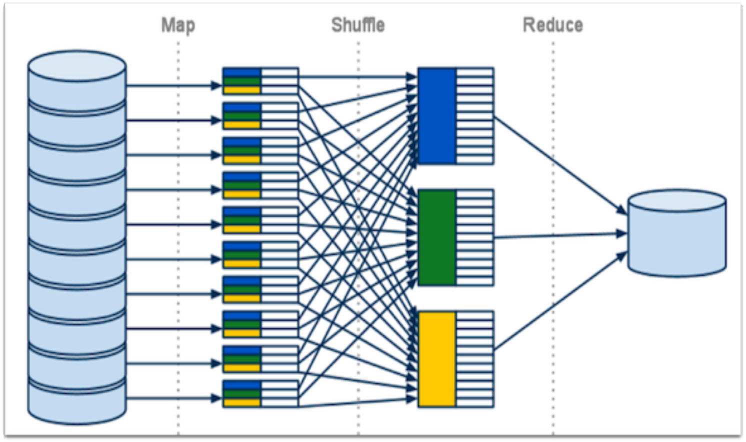


Figure 3.6: Map Reduce Architecture[12]

Mapper Function

The mapper function in Apache Spark is responsible for creating a key-value pair. The same key different value pair are then reduced using the reducer function. The mapper function in the semantic part is responsible for creating company name as key and its semantic score as the value.

During the neural network part the mapper function is responsible for creating key value pair were the name of the company would be the key and the value would be the class label predicted for that day by neural network These values would be later reduced using the reducer function

Reducer Function

The reducer function are responsible for reducing the same key different value pair using some function. We would be using the reducer function to filter values greater than 75 during the sentiment stage and to filter values having class label 1 during the neural network part.

Collect Function

The collect function in the Map-Reduce Architecture is responsible for collecting output from different worker nodes. After the execution of the mapper and reducer part the output is stored at the worker node only. So we need to fetch the results back from each of the worker node back to the master node and then constitute them together. This job is done by the collect part of the Map-Reduce Architecture

3.2 Spark Installation Manual

3.2.1 Requirement

The system should have linux operating system and Java 7 or higher installed

3.2.2 Steps to install Spark

- Step 1

  Download spark latest version from spark official website i.e. *https* : *//spark.apache.org/downloads.html* Directly download from the Download Spark link on the page

- Step 2

  You will get a .tgz file after download.Extract it and keep it anywhere you like. The extarcted folder is the actual spark working directory

- Step 3

  Repeat this steps on as many computers as you like to create more and more slave nodes. Keep 1 of the computers as master and rest as slave

3.2.3 Steps To Start Master

- Step 1

  Go to the computer that you have thought to make as master and find its ipaddress using ifconfig command

- Step 2

  Go to the working directory of spark and in there go to conf folder

- Step 3

  create a spark-env.sh file and add `SPARK_MASTER_HOST` =ur-ip-address If you already have the file just open it and edit the `SPARK_MASTER_HOST` as ur ip address. If it is not there just add it

- Step 4

  Open terminal and move to the spark working directory using cd command. enter the command ./sbin/start-master.sh If the master has successfully started no error would be displayed.

- Step 4

  You can check whether the master is up or not by typing the url localhost:8080. If the master is up it would be displayed on this url

3.2.4 Steps To Start Slave

- Step 1

  Go to computer which you have determined to use as slave.

- Step 2

  Open terminal and move to the spark working directory using cd command. enter the command ./sbin/start-slave.sh spark://master-ip-address:7077. wait for some time for the slave to get connected.

- Step 3

  If no error was reported then it means the slave was connected properly to the master. To confirm it view the slave on master localhost using url localhost:8080. If the ip address of slave is shown in worker then it means it is connected.

- Step 4

  Flow the same steps for all the computer that you want to make as slave node.

3.2.5 Steps To Run Program On Spark

- Step 1

  To run any program on spark cluster the program must be present in spark working directory of the master node. So before executing the program move it to the master node's spark directory.

- Step 2

  Open the program in any editor and change the master of the program in the line where we create the spark context using SparkConf() function. Just add .setMaster("Master-ip-address") after SparkConf() function.

- Step 3

  open terminal and move to the spark working directory using cd command.

- Step 3

  Type the command ./bin/spark-submit filenamewithextension spark://master-ip-address:7077 for starting the execution of program on spark cluster. After entering this command the program will begin execution

Chapter 4

EXPERIMENTAL RESULTS AND ANALYSIS

4.1 Work Done

Following work has been done in the direction of completion of project

- Finding volatile stocks using artificial neural network using sequential algorithm
- Finding volatile stocks using artificial neural network using parallel algorithm
- Changing the various parameter of the neural network so as to get an optimal set of parameters for maximum accuracy

4.2 Experimental Setup

We implemented our algorithm using a ubuntu virtual system having 5.3 mb RAM and 8 processors. We installed Apache Spark version 2.0.0 which has pre-built libraries for Hadoop on each of the system. Spark Standalone cluster mode is used for creating cluster of computer. The cluster contained 1 master node and various number of slave nodes. For our experiment we have used cluster having 1, 2 and 3 worker node.
Then we scrapped www.moneycontrol.com for the url of all the companies that have been listed on the site using a python script.A total of 7769 unique url were collected form this

website each belonging to a different company. These companies are either listed on NSE i.e. National Stock Exchange or BSE i.e. Bombay Stock Exchange or both. All the urls were stored in a file along with the name of the company to which they actually belong.

4.3 Historical Data

The historical data consists of company data for the past 1 year. The data that we have collected consist of

- Open Price
- High Price
- Low Price
- Close Price

The time duration under consideration is from 1st January 2016 to 31st December 2016. All the data is stored in csv format with the file name consisting of the name of the company and its unique ID (as provided by moneycontrol website). This data is later read for neural network training.

4.4 Results

4.4.1 Sentiment Score Collection

We ran our algorithm for sentiment score collection for all the 7796 companies listed on NSE and BSE. We first ran the algorithm sequentially and then on a Spark Cluster having 1, 2 and 3 worker node. We also ran the algorithm locally on a single computer using spark. Since running the algorithm only once doesn't provide us much accurate result hence we ran the algorithm 10 times and the average of time taken by 10 execution was noted down. The time taken by all these different spark cluster configuration has been listed in Table 4.1

Table 4.1: Time Taken By Different Spark Cluster Configuration

Spark cluster configuration	Sequential	Local	1 worker	2 worker	3 worker
Time taken	100	50	32.5	18.3	9.2

The Figure 4.1 depicts the graphical representation of the same data. The color of bar represents the technique that have been used i.e sequential method, running locally on spark cluster, running with 1,2 or 3 worker node on spark cluster. The height of the bar represents the time taken by that different spark cluster configuration. The time taken is in minutes It is quite clear from the graph that we can decrease the time taken by spark cluster by increasing the number of worker node in spark cluster. Morever for decreasing the amount of time, we also increased the number of cores on each worker node. In spark standalone mode each core is responsible for execution of 1 RDD block independently. For example if we have 20 rdd partitions and 8 cores then 8 rdd partition would be executed parallely.

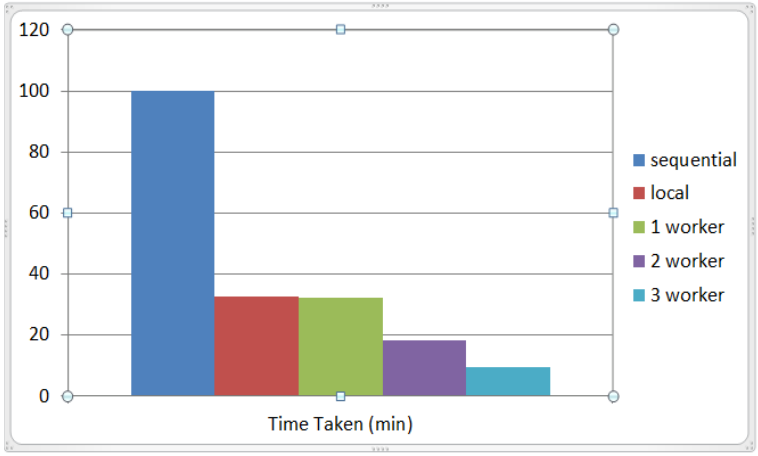


Figure 4.1: Time taken Vs Different Spark Cluster Configuration Used

In our case we have used 8 cores per worker node and we have divided our rdd externally in 200 partitions. Hence the framework is capable of executing $3 * 8 = 24$ partition at any point of time.

Moreover we can see that the time taken by executing the job locally on spark and with one worker node is same. This is due to the fact that whenever we create a cluster with a master then the master is only responsible for distributing the job among the worker node. Apart from this and managing the worker nodes the master node doesn't have any other function. So when we have 1 worker node then we have the computational resources of only 1 computer which is same as running the job locally using spark on a single computer. Hence the time taken by single node cluster and running locally on single computer are almost the same.

4.4.2 Neural Network Training And Prediction

We implemented the parallel approach of training and testing of neural network model for each individual selected stock using spark standalone cluster mode having 1 master and 1,2 and 3 slave nodes. We even ran the algorithm sequentially and locally on spark and noted the time taken. Since executing the algorithm single time wont give much accurate result hence the algorithm was ran 10 times and the average of time taken was noted. The table 4.2 shows the time taken to train the neural network and predict the output class for present day data for various techniques. The techniques include sequential approach, parallel approach running on a single machine as a local job, parallel approach running on a single worker cluster, parallel approach running on a 2 worker cluster and parallel approach running on a 3 worker cluster.

Table 4.2: Time Taken By Different Spark Cluster Configuration

Spark Cluster Configuration	Sequential	Local	1 worker	2 worker	3 worker
Time taken (min)	27	9.2	9.3	5.3	4.5

Fig 4.2 shows a graphical representation of the analysis. The height of the bar represents the time taken in minutes and color of the bar represent the approach used like sequential approach, local job approach and so on. From the graph it is quite clear that we can decrease time taken for neural network training and prediction by moving from sequential

to parallel and then further by adding more and more worker node in the parallel approach.

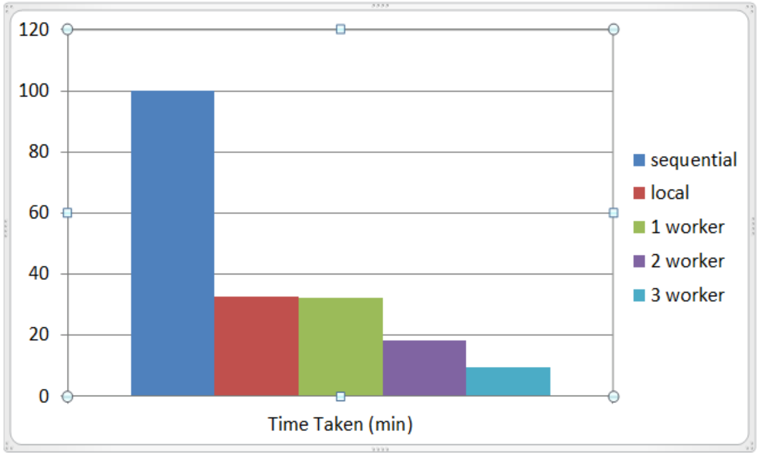


Figure 4.2: Time taken Vs Different Spark Cluster Configuration Used For Neural Network Training

Similiar to the sentiment score calculation part we decreased the time further by increasing the number of cores in each individual worker. Since each individual worker has 8 cores hence the spark cluster is able to run 3*8 i.e. 24 parallel jobs at a particular point of time. Again the time taken on running the job locally on spark and with 1 worker node are almost the same due to the fact mention in section 4.4.1

4.4.3 Analysis Of Model

After the neural network has been created we varied the various parameters of the neural network like learning rate, tolerance value as to see its impact on the accuracy of the neural network The accuracy of each individual companies neural network is defined as the total number of instances correctly classified by it out of the total instances provided for training. Moreover we have 10 fold cross validation for predicting the accuracy of each individual company's neural network. The overall accuracy of the entire neural network model is defined as the average of all the individual neural network accuracy. For example if we have 5 companies with accuracy as 60,65,70,75 and 80 then the average of all these accuracies i.e 70 is defined as the overall accuracy for neural network model The idea behind varying the various parameters of the neural network was to get view its impact on the overall accuracy of the neural network so as to get an optimal set of parameters for which the accuracy of the neural network model would be highest

Learning Rate Analysis

Learning rate in a neural network is used to control the change in the weight and bias after every iteration. Learning rate determines how much the weight should differ in the next iteration. The values of learning vary from a maximum of 1 to a minimum of 0. If the learning rate is high then there would be huge difference in the weight of two consecutive iterations and low learning rate means small changes .We varied the learning rate from 0.1 to 0.0000001 .Table 4.3 shows the variation of the overall accuracy of the neural network model with learning rate.

Table 4.3: Accuracy Vs Learning Rate

Accuracy (%)	82.95786	83.00	83.41532	83.19748	66.66761	49.32989
Learning Rate Value	0.1	0.01	0.001	0.0001	0.00001	0.000001

The Figure 4.2 shows a graphical representation of the analysis. The y-axis represents the accuracy and the x-axis represents the value of learning rate. From the graph it is quite clear that we can increase the accuracy by decreasing the learning rate initially. This is due to the fact that the changes in weight would be smaller and hence we have a higher probability of reaching the global maximum value. But if we decrease the learning rate after a certain value keeping the maximum iterations as constant than the accuracy decrease. This is due to the fact that if the changes in weight are small and the maximum iteration are constant then the neural network would stop training as soon as the maximum iteration runs out. since the changes in weight were small so even iteration ranging upto 10000 would have changed the weight very less. Hence the accuracy of the model decreases

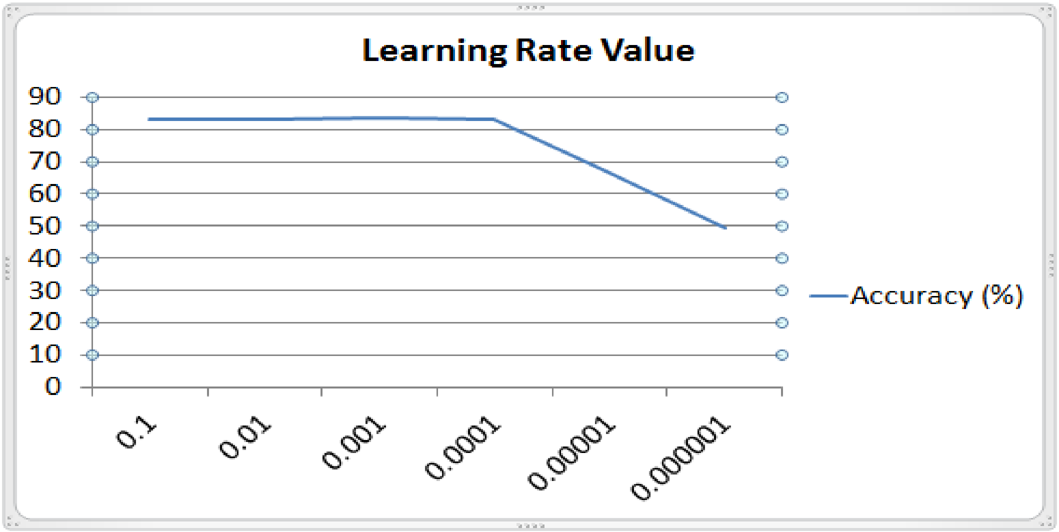


Figure 4.3: Accuracy Vs Learning Rate

Tolerance Value Analysis

Tolerance is defined as value which is used to control the termination condition of the neural network. If the difference in error in 2 consecutive iteration if less than the tolerance value than convergence is considered to be reached and hence the training stops . Higher value of tolerance means the training terminate early while lower value of tolerance means the training would terminate after some time we varied the tolerance from a maximum of 0.1 to a minimum of 0.000001. Table 4.3 shows the variation of overall accuracy of the neural network with changes in the value of tolerance.

Table 4.4: Accuracy Vs Tolerance

Accuracy (%)	49.67094	69.28688	83.17555	83.36953	83.03570	82.76023
Tolerance Value	0.1	0.01	0.001	0.0001	0.00001	0.000001

Fig 4.3 shows a graphical representation of the analysis. The Y-axis represents the accuracy and the x-axis represents the value of tolerance. Initially the accuracy is low since the tolerance value is high and hence the neural network terminates after small

number of iterations and hence the model is not properly trained.If we decrease the the tolerance value then the neural network is trained for more and more iteration and hence the accuracy of the neural network model increase After the certain value if we keep the number of iteration constant then the accuracy becomes constant. This is due to the fact that in this scenario the maximum number of iteration becomes the terminating condition and hence if the maximum iteration is constant then the accuracy would also remain constant.

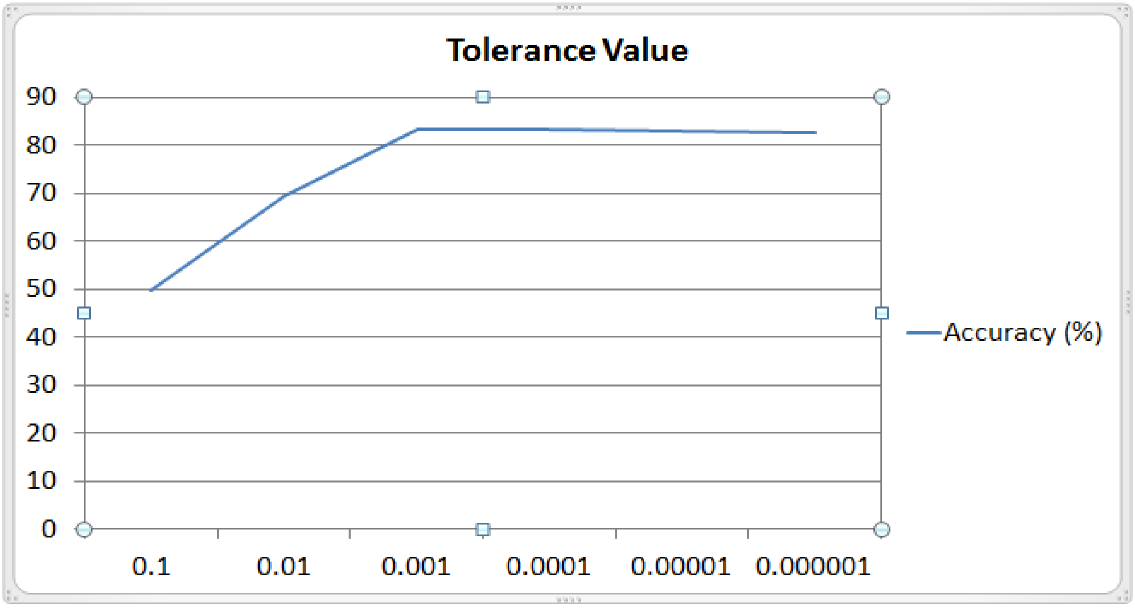


Figure 4.4: Accuracy Vs Tolerance Value

Hidden Layer Analysis

Hidden Layer in a neural network are defined as the layers of neuron present between the input and the output layer.These layers are responsible for determining the non-linear relationship between the input and output layer. If there is no hidden layer then the neural network will not be able to detect non-linear relationship A higher value of hidden layer means there are more number of neuron layer between the input and the output layer and a lower value of hidden layer means the opposite.We varied the number of hidden layers from a minimum of 1 to a maximum of 5. Table 4.4 shows the impact of number of hidden layers on the time taken to train the network.

Table 4.5: Number Of Hidden Layers Vs Time Taken

Hidden layer Size	1	2	3	4	5
Time Taken (min)	1.6	1.9	2.4	2.6	3.4

Figure 4.4 represents the graphical representation of the variation in time taken for training the network with the number of hidden layers in the neural network. The x-axis represents the number of hidden layers while the y axis represents the time taken in minutes. From the graph it is quite clear that if we increase the number of hidden layer then the time taken to train the neural network increase. It is due to the fact that if we increase the number of hidden layers then the number of neurons between the input layer and the output layer would increase. An increased number of neurons would mean more weight adjustment have to be performed. Since each weight updation would take some time hence more updation would increase the total time taken by the neural network for training.

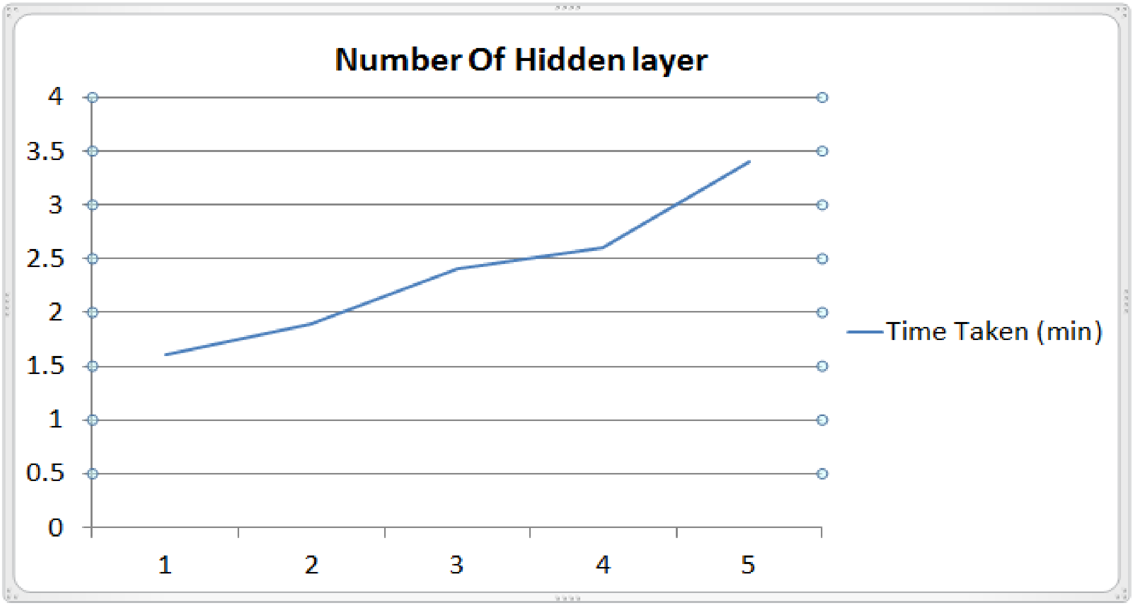


Figure 4.5: Hidden Layer Size Vs Time Taken

The table 4.5 shows the impact of the number of hidden layers on the overall accuracy of the neural network. We have varied the number of hidden layers from a minimum of 1 to a maximum of 5. 10 neurons were used in each hidden layer

Table 4.6: Number Of Hidden Layers Vs Accuracy

Hidden layer Size	1	2	3	4	5
Accuracy (%)	82.67	83.22	83.32	83.33	83.36

Figure 4.5 shows the graphical representation of the table 4.5. The y axis represents the accuracy in % and the x-axis represents the size of the hidden layer. From the graph it is quite clear that even on increasing the size of the hidden layer there is no significant increase in the accuracy of the overall model. This is because the one or two hidden layer are enough for determining the non-linear relationship since they both would have activation function neccessary for the determining non-linear relationship. More hidden layer wont add much to the accuracy but would increase the time

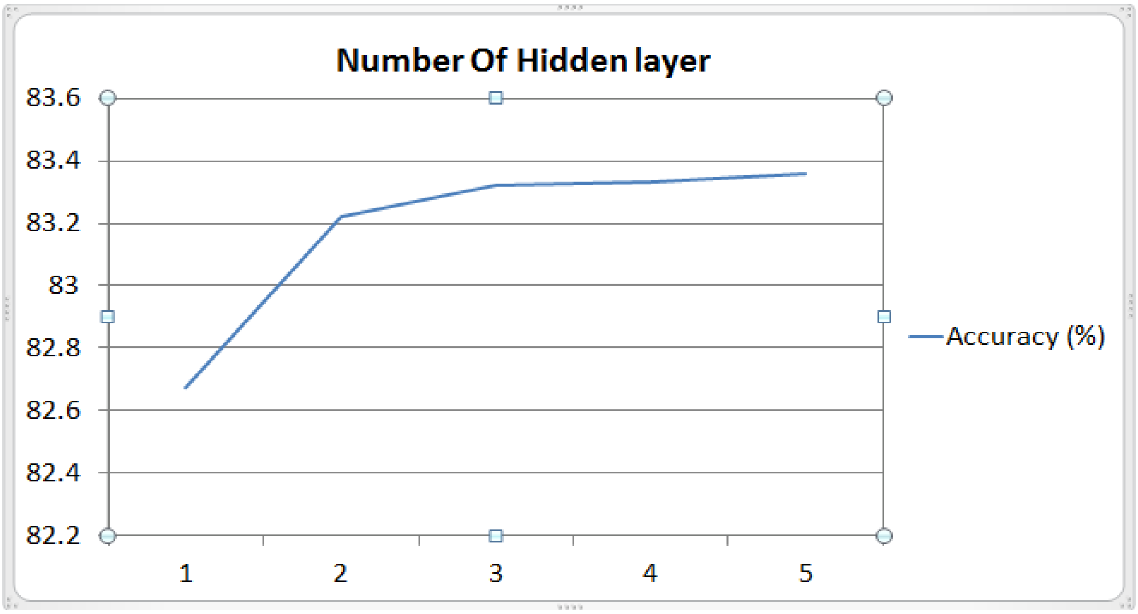


Figure 4.6: Accuracy Vs Hidden Layer Size Vs Time Taken

4.4.4 Accuracy Of Model

Since we are training the neural network model hence they must have an accuracy that defines how many instances among the testing part were correctly classified by them. For example if we have 10 instances in our testing set and we are able to correctly classify 7 among then the accuracy of individual neural network is 70The accuracy of the neural network is the average accuracy of all the instances for neural network that have been trained. For example if i have 5 companies and they have individual neural network accuracy as 70,75,80,85 and 65 then the average accuracy of neural network is 75
The overall accuracy of the model is defined as the number of stocks correctly predicted by the model.For example if i have 10 stocks and i was able to correctly predict the movement of 8 stocks then the accuracy of the model is defined to be 80
The fig 4.6 depicts the average accuracy of the neural network and the overall accuracy

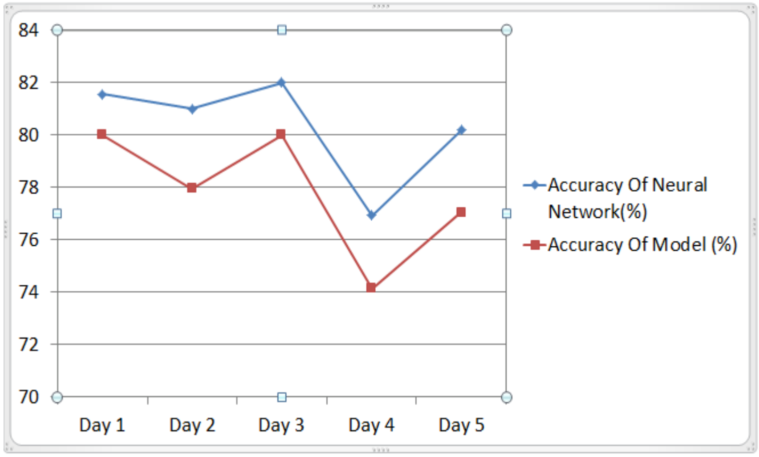


Figure 4.7: Neural Network accuracy and Model accuracy for 1 week

of the model for a duration of 1 week. From the figure it is quite clear that if the accuracy of model decrease as the average accuracy of the neural network decreases. The reason for such a behaviour is that low average accuracy of neural network means that they were not trained properly. Since the neural network were not trained properly hence they are not able to correctly predict the output class and hence the accuracy of the model decreases.

Chapter 5

CONCLUSION

5.1 Conclusion And Future Work

From this work we can conclude that using Apache Spark and Map Reduce architecture we can reduce the computational time required for getting the sentiment score for all the companies. From the analysis of the neural network part it is quite clear that we can increase the accuracy of the neural network by decreasing the value of learning rate and increasing the value of tolerance upto a certain limit. If we keep on decreasing the learning rate more then the accuracy would go down and if we keep on increasing the tolerance value after that then the accuracy would remain constant but time taken would increase. Moreover since there is no significant increase in the accuracy by increasing the number of hidd5en layers hence we can keep it 1 or 2. In the future we can use any other machine learning algorithm for training and prediction part and compare its accuracy with the existing model

REFERENCES

[1] P. Pai and C. Lin, “A hybrid ARIMA and support vector machines model in stock price prediction”, Omega vol. 33, no. 6, Dec., pp. 497-505, 2005.

[2] Gharehchopogh, Farhad Soleimanian, Tahmineh Haddadi Bonab, and Seyyed Reza Khaze. ”A linear regression approach to prediction of stock market trading volume: a case study.” International Journal of Managing Value and Supply Chains, vol. 4, no. 3, Sep., pp. 25-30, 2013.

[3] Majumder, Manna, and M. D. Hussian. ”Forecasting of Indian stock market index using artificial neural network.” Available : www. nse-india. com/content/research/FinalPaper206. pdf (2007). [Accessed: Oct. 12, 2017].

[4] Abhyankar, A., Copeland, L. S., Wong, W. (1997). ”Uncovering nonlinear structure in real-time stockmarket indexes: The S-P 500, the DAX, the Nikkei 225, and the FTSE-100” Journal of Business - Economic Statistics, vol 15., no. 1, Jan., pp. 1-14, 1997.

[5] Xu, S. Y. “Stock Price Forecasting Using Information from Yahoo Finance and Google Trend”. UC Brekley, 2014.

[6] Kar, Abhishek. ”Stock Prediction using Artificial Neural Networks.” Dept. of Computer Science and Engineering, IIT Kanpur.

[7] Igiri Chinwe Peace, Anyama Oscar Uzoma ”Effect of Learning Rate on Artificial Neural Network in Machine Learning” , International Journal of Engineering Research & Technology (IJERT) , vol. 4, no. 3, Feb., pp. 359-363, 2015

[8] Urmi Jadhav, Ashwija Shetty ”Effect of varying neurons in the hidden layer of neural network for simple character recognition ”, International Journal on Recent and Innovation Trends in Computing and Communication, vol. 4, no. 6, Jun. , pp.266 - 269, 2016.

[9] Khaw, John FC, B. S. Lim, and Lennie EN Lim. ”Optimal design of neural networks using the Taguchi method.”Journal of Neurocomputing, vol. 7, no. 3, Apr., pp. 225-245, 1995.

[10] *http : //spark.apache.org/examples.html* [11-June-2018]

[11] *https : //www.youtube.com/channel/UCkw4JCwteGrDHIsyIIKo4tQ* [11-june-2018]

[12] *http : //www.moneycontrol.com/* [11-June-2018]

[13] *http : //craw.tk/mapreduce/* [11-June-2018]

[14] *https : //www.extremetech.com* [11-June-2018]

[15] *http : //spark.apache.org/architecture.html* [11-June-2018]