A report on

Stock Market Prediction and Analysis using Deep Learning and Hadoop

Project report submitted in partial fulfillment of the requirement

for the degree of Bachelor of Technology

in

Computer Science and Engineering

By

Parth Verma (161241)

Under the supervision of

Prof. Hari Singh

to

Department of Computer Science & Engineering and Information Technology Jaypee University of Information Technology,

Waknaghat,Solan,

Himacal Pradesh, 173234

CANDIDATE'S DECLARATION

I hereby declare that the work presented in this report entitled "**Stock Market Prediction and Analysis using Deep Learning and Hadoop**" in partial fulfillment of the requirements for the award of the degree of **Bachelor of Technology** in **Computer Science and Engineering/Information Technology** submitted in the department of Computer Science & Engineering and Information Technology**,** Jaypee University of Information Technology Waknaghat is an authentic record of my own work carried out over a period from **July 2019** to **May 2020** under the supervision of **Dr. Hari Singh**, Assistant Professor (Senior Grade), Jaypee Univeristy of Information Technology, Waknaghat,Solan.

The matter embodied in the report has not been submitted for the award of any other degree or diploma.

Parth Verma – (161241)

This is to certify that the above statement made by the candidate is true to the best of my knowledge.

Dr. Hari Singh,

Assistant Professor

(SeniorGrade)

Department of Computer Science & Engineering and Information Technology.

Dated: 27/May/2020

ACKNOWLEDGMENT

Any serious and lasting achievement cannot be achieved without the help, guidance and co-operation of numerous people involved in the work.

First and foremost, we would like to express my gratefulness to Prof. Dr. Samir Dev Gupta, Head Department of Computer Science & Engineering and Information Technology, Jaypee University of Information Technology for providing us the opportunity to carry out this project as our final year project. It gives us immense pleasure to express my deepest gratitude and thanks to Dr. Hari Singh, Assistant Professor (Senior Grade), Department of Computer Science & Engineering and Information Technology, for not only imparting his knowledge but also his constant supervision, advice and guidance throughout the project, without which this project wouldn't have been possible.

We would also like to thank all other department faculty at Jaypee University of Information Technology. Not only did they taught us and made us capable enough to undertake this project but were always there at the need of the hour and provided with all the help, facilities and co-operation, which was required towards the completion of our project.

A special mention to Ravi Raina Sir and Sanjeev Kumar Sir who assisted our project lab and guided us towards all the minor issues.

Last but not the least , We would like to express our thanks to our parents and family members for their support at every step of my life.

TABLE OF CONTENT

List of Abbreviations

- ML: Machine Learning
- ANN: Artificial Neural Network
- RNN: Recurrent Neural Network
- LSTM: Long Short-Term Memory
- RMSE: Root Mean Square Error
- MAE: Mean Absolute Error
- MAPE: Mean Absolute Percentage Error
- BPTT: Backpropagation Through Time
- DL: Deep Learning
- CSV: Comma Separated Values
- API: Application Program Interface
- HDFS :Hadoop Distributed File System

List of Figures

ABSTRACT

In a financially unstable market, like the stock market, it is necessary to have very precise prediction of the stock prices. Because of hundred of different factors stock prices changes within seconds, thus it is important to have a secure, reliable and precise predictions of the market trends. The successful prediction of stock's future price could return good profit. Analyzing historic trends and various other factors to predict a complex non-linear graph requires advance algorithms of machine learning and deep learning.

In this project **Stock Market Prediction using "Deep Learning and Hadoop"** we have taken initiative to analyze stock market and predict future stock prices. We did the research and literature survey and found that there are various methods to predict the share market. We studied different types of neural networks and their working. So in this project we have studied the various algorithms and many different tools used for prediction and applied that knowledge to make a deep learning model to predict the future stock prices.

Many different methodologies were adopted. It was the beginning with the collection of data from past years. Then we studied different techniques to pre process the data and then pre processed it as required by our model. We identified the features like opening price, highest price, volume etc. which affects the stock prices. Then we made the neural network model, trained it using different hyper-parameters to tune the model and then predicted the future values. Considering overall this project is a great learning experience. It is a concept that enables us to predict something where trillions of dollars are on stake.

To build the project different tools and technologies are used. Different libraries of python are used to complete this project. Panda and numpy are used to collect and pre-process the data. Keras is used to build the neural network and implement the algorithm and matplotlib is used to visualize the results.

Chapter-1

INTRODUCTION

Creating a model, predicting the stock prices and analyzing the trends of the stock market is now becoming a fascinating field for people belonging to different fields whether they may be scholars, investors or researchers. The stock market is a concept where monetary goods are exchanged among people who buy and sell it. The analysis and prediction of the financial market (i.e., stock market) has gained interest with the application of artificial intelligence and Hadoop.

The Stock Market analysis using Hadoop uses Map Reduce algorithm to make analysis of stocks in a very fast manner rather than using any other infrastructure which may consume a lot of time. The stock market varies due to the environment which is political and macro-economical.. Sample size of the data used for stock markets is real -world transactional. On one hand, a larger data size refers to a longer duration of the transaction record; On the other hand, large data increases the uncertainty of the financial environment during the testing/predicting period. In this project, we use stock data instead of daily data to reduce the probability of uncertain noise, and relatively increase the sample size within a given period of time.

In the past years, there has been extensive research on machine learning and hadoop methods for their potential in forecasting and analyzing the financial market. Neural networks have more efficiency in predicting the stock prices rather than traditional models and hadoop infrastructure is very efficient in working on huge databases and working calculations on them. They also have the ability to understand the systems that run dynamically. It is done through a reprocessing process using new data pattern and Map reduce algorithm uses mappers and reducers to efficiently work on the data.

Problem Statement

The problem we are dealing and trying to find the solution is the problem of the risk of losing the money on the stock markets. Stock market is dependent on various factors. We have many attributes of a company on stock market such as opening date, closing date, opening price, closing price, volume and adjusted close which keeps on changing and the money value of a stock goes up and down. The problem here arises that in which company's stock does an investor should invest his money such that he/she will maximize their profit.

We hear about the stock market in the news every day whenever it reaches a new high or a new low in its prices/volume. If a better algorithm could be developed to forecast the short term price of an individual stock and if some of the datasets(big data) are analyzed with an efficient algorithm and covariance is calculated then the rate of the investment by an individual can reasonably increase by winning his trust with the help of a software.

In fundamental-analysis a company's future profits are analyzed by taking two factors in account i.e., current business environment and financial performance.

In technical-analysis charts and using statistical figures are used to determine the trends in the stock market.

We are aiming to make some algorithm work for the above problem statement which will predict stock price with a low percentage of error and thus lowering the risk of losing the investor's money.

Why Stock Market Prediction and Analysis?

Stock market is collection of buyers and sellers of shares or stocks of companies which are listed on the stock market or privately traded stocks. As of 2017, size of the entire stock markets of the world was touching US$80 trillion mark.

Stock market is one of the most significant method for raising the investment for the companies allowing them to function publicly and earn a lot of profit.

Everyday billions of transactions taker place on the stock market around the world and the market doesn't depends only factors which can be characterized.

Stock market values get affected by various factors both internal and external like supply and demand, investor sentiments, company performance and other internal factors or some other external factors like interest rate, speculation, growth and GDP, socio-political factors.

Stock market values varies non-linear and can vary within seconds which makes it much more difficult to predict them precisely and efficiently.

With the advancement in technology stock market forecasting has moved from traditional methods to technological like machine learning and stock market analysis has moved from the hands of sql queries to the HIVEQL queries.

With the advancement in neural networks the accuracy of prediction has improved immensely and with the diversification of the hadoop infrastructure the analysis is made just a few seconds scenario.

The most important technique is time series prediction using recurrent neural networks (RNN) which is being used in this project to predict stock market.

The algorithm of map reduce in Hadoop used through hive is used for the analysis part of this project .

Objectives

The Objectives of this project are:

FOR PREDICTION:

- To detect the various common trends stock market prices.
- To collect and feed the data of various companies in a csv file to our machine learning algorithm in a formatted way and HDFS.
- As per the detected trends and applied algorithm of LSTM predict whether the price will go up or down to give the maximum profit.
- Design the algorithm/project to minimize the error percentage in predicting the stock market trends in the prices.

FOR ANALYSIS:

- To analyze the Stock Market.
- To collect and feed data of companies in a csv file to HDFS and then to Hive.
- The covariance is calculated by applying its formula on the data available from csv file.
- A user is able to analyze the stock market by accessing the datasets in a faster manner.

Methodology

A. PYTHON:

Python was chosen as the best choice available due to many reasons and some of them are listed below.

1. There is a big community behind python and is the most popular language of today's programming world. Whenever any error or doubt comes in our mind then it gets easily cleared by making a small trip to the Stack Overflow or Google .

2. There is an abundance of powerful tools in Python which are ready for implementation. Numpy, Tensorflow ,Kersas are easily available and well documented for the use. The problem of writing a big complex code is certainly reduced by making the use of these packages which makes the iterations in python really quick.

3. Python can also be termed as a forgiving language as it can be easily implement by just looking at some pseudo code available online or on the papers.

B. NUMPY

The scientific and higher level mathematical abstractions are wrapped in python under the python module-NUMPY . We are not able to write mathematical abstractions such as f(x) in most of the programming languages because it will be not semantically or syntactically correct thus will result producing errors in our code.But we can use numpy and use as many as mathematical abstractions that we want in our code.

The numerical work can be done efficiently in python with the help of an efficient data structure such as Numpy's array type e.g., manipulating matrices. Many numerical routines can be solved by Numpy such as we can calculate the eigen vectors using numpy.

C. TENSORFLOW

Data flow graphs based numerical calculation can be performed in python by using the Tensorflow which is on open software easily available. The mathematical operations are represented by the nodes whereas the edges of the graph represents the multidimensional data arrays (which are termed as tensors) communicated between them. The computation can be deployed to 1 or more CPU's or GPU's in a PC,server or android with the help of this flexible architecture with only one single API.

It was developed by the Google when they were conducting a research in ML and deep neural networks and mow it can be deployed in several domains other than these.

While others can run on single devices whereas TensorFlow can be run on multiple machines (CPU's or GPU's) and it is available on linux , windows, apple(mac), android and IOS.

D. KERAS

It can be referred as an API coded in Python which is an interface of neural networks and is run on the top of the Tensorflow. The reason to design/develop it was to enable a faster experimentation. As a faster experimentation is the key to successful research therefore Keras is used widely for its purpose. It speed is due to the three factors namely friendliness, modularity, and extensibility. It can run on both CPU and GPU and can work on the two networks namely - convolutional networks and recurrent networks

E. COMPILER

We have used anaconda to run our project as it has a user friendly implementation and the libraries can be accessed with an ease.

F. Hadoop

It is a framework which is used to process the huge datasets in a distributed manner. A large number of concurrent tasks or jobs can be processed and the storage is provided for any kind of data. Its basic functionality can be defined as follows:

1. The storage is provided with the help of HDFS.
2. The data is processed using Map-Reduce.
3. The tasks are divided with the help of Yarn.

G. Map-Reduce

It is an algorithm used by Hadoop to process the massive datasets in a parallel manner. It basically consists of mappers and reducers which helps in the scheduling of tasks. It helps in processing the data available in the HDFS in a faster manner.

H. HIVE

It is a software of data warehouse which provides SQL like interface to query the data available in different databases and integrate them with Hadoop Infrastructure. It is present on the Hadoop at the top.

Organization

The organization of project report is as follows:

Chapter 1 will have the basic-introduction to our system. Aim, objectives and the problem statement being worked upon are all covered in the introduction. A brief introduction about the methodologies used and the algorithm being used in our project is illustrated.

Chapter 2 will have the literature survey which include the various surveys and research being made on the LSTM algorithm ,Hive , Hadoop framework and the other modules being used in our project. This literature survey was done to find the best solution to our problem statement.

Chapter 3 will have the system design and structure thoroughly explained.

Chapter 4 covers the algorithms used for our project.

Chapter 5 covers the results and performance analysis part.

Chapter 6 will have the conclusion and future work to be made in this project to extend it.

Artificial Neural Network (ANN)

An Artificial Neural Network is a system which is based upon the human nervous system. An ANN is a model that is data driven and which we don't need to program explicitly and is capable of solving complex problems with the help of machine-learning neurons. They are able to interpret complex non-linear relationships between dependent and the independent vectors. The input and output are related to each other by

$Y=f(X^n)$

Where Y is the output vector and $X^n$is n-dimensional input vector. The f(.) is function which is unknown but is represented by various parameters of the model.

The basic architecture of any ANN consists of three layers:

- an input layer,
- a hidden layer and
- an output layer.

Weights, bias and different types of activation functions like sigmoid function are used to connect neurons in one layer to another layer

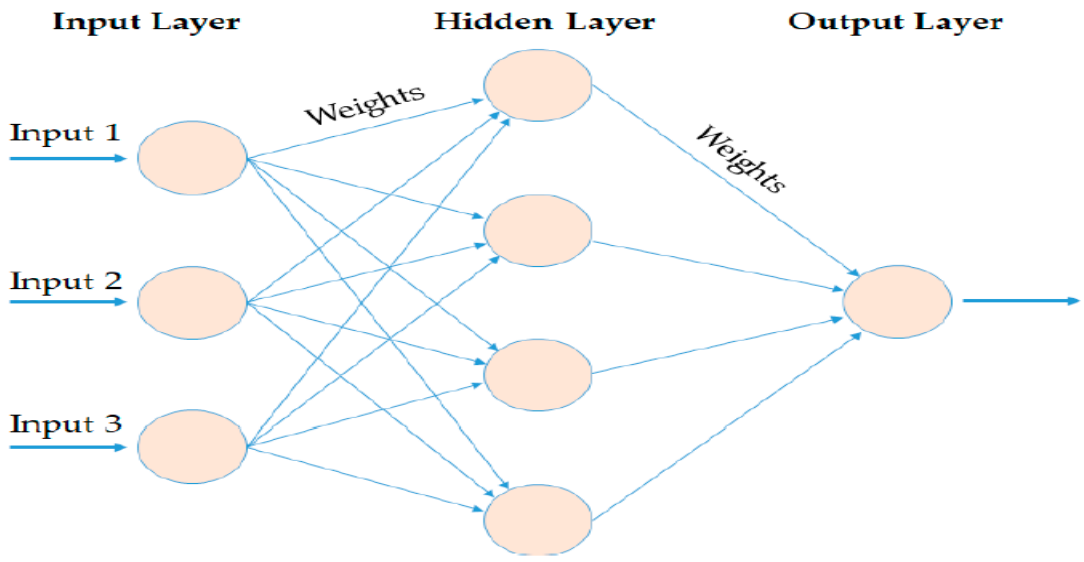


Figure 1: Simple ANN architecture

Rise to variations in ANN is given by the the difference in the no. of hidden layers and no. of neurons in each hidden layer. These parameters are selected based on data type or data size and can be determined by analyzing the training results.

Typically, in an ANN to start the algorithm random weights are assigned to all the links between the neurons. Using the inputs and the (input->hidden node) linkages activation rate of neurons of hidden layer is calculated. Similarly, using the activation rate of hidden layer neurons and weights assigned to hidden and output layer links, activation rates of output layer is found out. Error rate is calculated at the output rate and depending upon that all the weights are updated accordingly using the backpropagation algorithm from hidden layer to the input layer. This process is looped again time and again until a certain criterion is met and the weights are updated. Using these weights and and activation functions of the layers output is given by the model.

The traditional ANN model has certain limitations and it fails to solve the sequential time series problems such as timeseries stock market prediction. To overcome this problem recurrent neural network (RNN) were introduced.

Advantages of ANN

- Storing data on entire network
- Can work without complete knowledge
- Fault tolerant
- Distributed memory
- Parallel processing
- Slow corruption

Disadvantages of ANN

- Depends on hardware
- Doesn’t explain how the solution is found
- No proper network structure
- Not able to solve timeseries problems

Recurrent Neural Network (RNN)

It was first introduced in 80s. Like ANN, it also consists of the same type of layers (input ,output and hidden). But unlike ANN, RNN can have one or more hidden layers. These layers form a chain-like structure in which these layers acts as memory units to store previously processed information. Unlike feedforward networks in which the information flows in forward direction only, RNNs also consist have feedback loop allowing the network to acquire stream of inputs which means that the output of step t-1 is given back as an input into the neural network to make the output of step t more accurate. This process make the RNNs better than ANN in learning sequences.

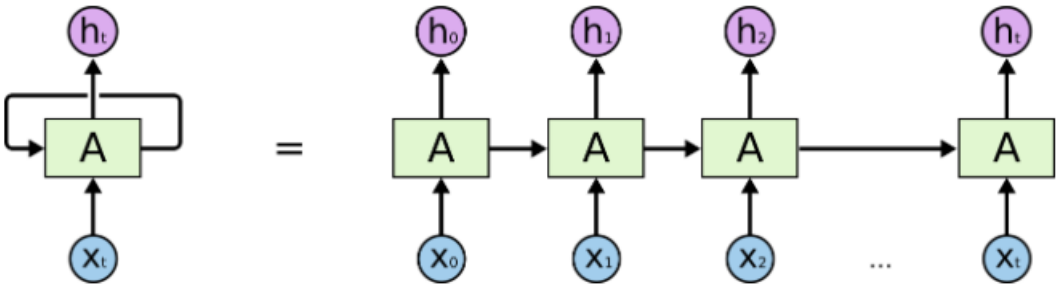


Figure 2:RNN

Figure mentioned above illustrates a simple RNN with one input, hidden and output unit and one recurrent hidden unit. The input at timestep t is denoted by X_t and similarly output at timestep at t is denoted by h_t. To train the RNN network, it uses a backpropagation algorithm which is also used in ANN to adjust the weights. But in RNN, feedback process also adds up to count which is thus known as backpropagation through time (BPTT).A backward approach is used by BPTT where it updates the weights of the links between the neurons unit by unit from output layer to the input layer making it not good enough to train the model to learn long term dependencies due to various problems it faces like gradient vanishing.

Advantages of RNN

- Can model timeseries problem
- Able to remember previous data
- RNN are able to pair with CNN to extend pixel neighborhood

Disadvantages of RNN

- Gradient exploding and vanishing
- Train a RNN model is not easy
- Cannot process large sequences of data

Long Short-Term Memory (LSTM) Neural Network

It is an improvement over regular RNN model to overcome the problems mentioned above by increasing the no. of cells. LSTM are improved version of regular RNN that can learn long term dependencies and remember necessary information required to train the model. LSTM architecture form a chain like structure. However, unlike RNN which have single neural network, the repeating module of LSTM has a structure consisting of four communicating layers which interact with each other in a particular way.

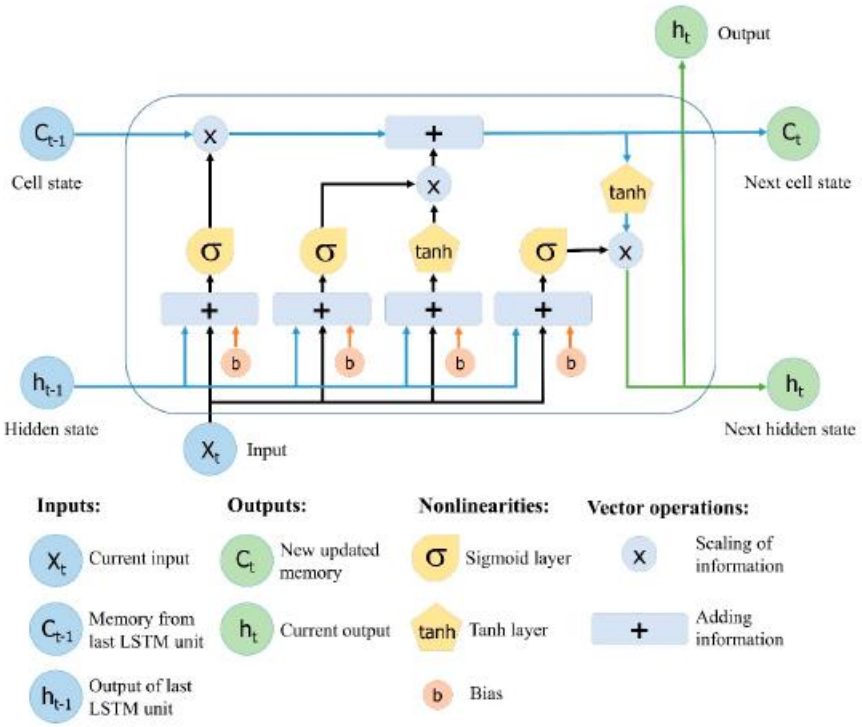


Figure 3:LSTM Cell

A LSTM model is made up of cells which acts as memory units of the network. The two cell states namely next cell state and next hidden state are fed to the cell which is connected to the next. The cell state is main component which allow the passage of data in forward direction without any change. Sigmoid gates can be used to remove data from the cell states. LSTM are able to overcome long term dependency difficulty with the help of gates which controls the process to remember the information.

To build a LSTM model the initial step is to recognize the data that is not important and will not be passed through the cell in that particular step. Sigmoid function takes care of it which takes two inputs i.e. output of last LSTM unit (h_{t-1}) at time t-1 and input (X_t) at current time t. The function of this sigmoid function is to identify which part of last unit output is to be discarded. This gate is known as the forget gate (f_t) where $f_t$is a vector which can have value from 0 to 1.

$$F_t=\sigma(W_f\,[h_{t-1},X_t] + b_f)$$

The next step is to identifying and storing (X_t) information (new input) in the cell state and cell state updation also. To carry out this step, two steps needs to be done. First the sigmoid layer and the second, tanh layer of the cell. First step is to decide whether the information of the X_t needs to be added or ignored which means it can have two values 0 or 1 and this is decided by the sigmoid layer. The second step is to give weight to the updated information based on its importance to the model which means it can have value ranging from [-1,1] and this is decided by the tanh layer of the module. The product of these two values is calculated and the new cell state is updated. The new information is and old information C_{t-1} are added to make new information/memory C_t.

$$i_t= \sigma\,(W_i\,[h_{t-1},X_t] + b_i$$

$$N_t = \tanh\,(W_n\,[h_{t-1}\,,X_t] + b_n$$

$$C_t = C_{t-1}f_t + N_t i_t$$

Where, C_{t-1} and C_t are two different cell states at time t-1 and t respectively, whereas W and b are the weights and bias of the cell states respectively.

In the last step, the output values of the model depend upon the state of the output cell (O_t) of the model. It is also a two-step process. Firstly, the part of the cell which makes to the output is decided by applying sigmoid function by the sigmoid layer of the cell. Next, the output generated by the sigmoid gate is multiplied by the output of the tanh layer from the cell state (C_t) with a value which ranges from [-1,1].

$$O_t = \sigma\,(\,W_o\,[h_{t-1}\,,\,X_t] + b_o)$$

$$h_t = O_t\tanh(\,C_t)$$

Advantages of LSTM

- Overcomes vanishing gradient problem
- Can solve long-term dependencies
- Outperforms traditional RNN

Disadvantages of LSTM

- Takes time to train
- Easy to overfit
- Difficult to implement dropout
- Different random weight initialized at the beginning can affect model

HADOOP:

Hadoop is a framework which is written in java and that is used to store large datasets of any kind in a distributed manner. The data stored is then processed using map-reduce algorithm with the help of mappers and reducers.

It is basically made up of two parts:

a)HDFS(The Hadoop distributed file system) for storage.

b)Map-Reduce algorithm for processing .

Its latest version in use is 3.2.1 that has been used in this project. There exists enormous number of packages which can be installed at its top namely,

APACHE-HIVE
APACHE-SQOOP
APACHE-HBASE
APACHE-PIG

Apache hive is used in our project for processing the database and getting the required results and,

Apache sqoop is used for the importing the csv file containing data to HDFS and Hive table.

The main advantage in Hadoop is high availability due to the fact that the Hadoop libraries are designed to find and work on error to remove it at the application layer itself.

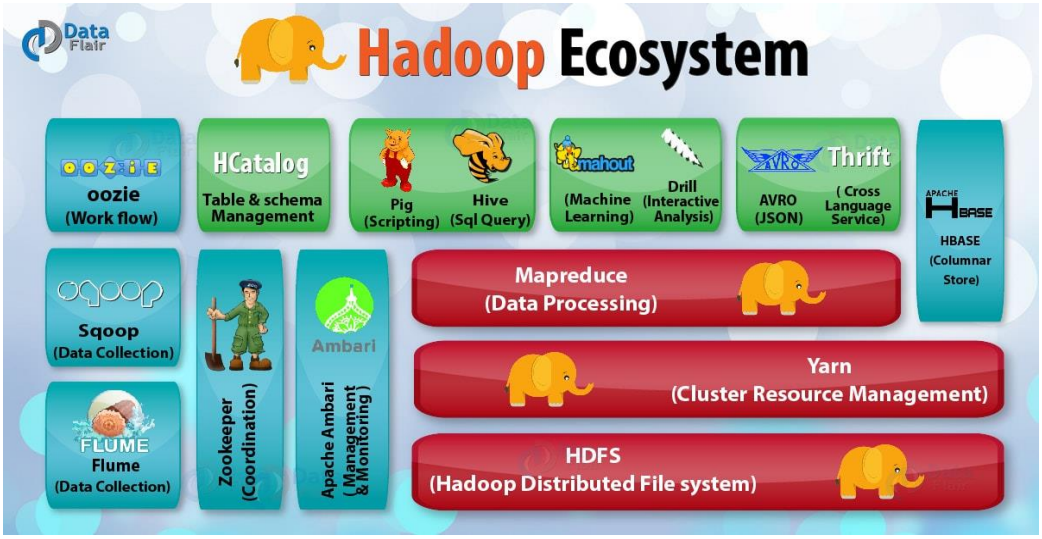


Figure 4:Hadoop Ecosystem

Hadoop Vs Traditional RDBMS:

- Hadoop can be defined as a software which use a distributed environment to solve problems on large datasets whereas RDBMS is a software which uses a relational model for creating and managing the databases .
- Hadoop is able to store any type of data such as :

    1. structured data,

    2. unstructured data and

    3. semi-structured data

    whereas RDBMS is used to store only structured data.

- Hadoop performs faster read and write in comparison with RDBMS on a large dataset.

Map-Reduce:

The Map-Reduce algorithm is used to process data in a distributed environment .It consists of two tasks, namely, map and reduce. Data in raw form is fed to map which converts it into smaller chunks also known as tuples. These tuples are given as a input to the reduce which combine these tuples further. The main advantage of map-reduce algorithm is that once we use this form to write an application then that application could be made to run of thousands of machines in a cluster.

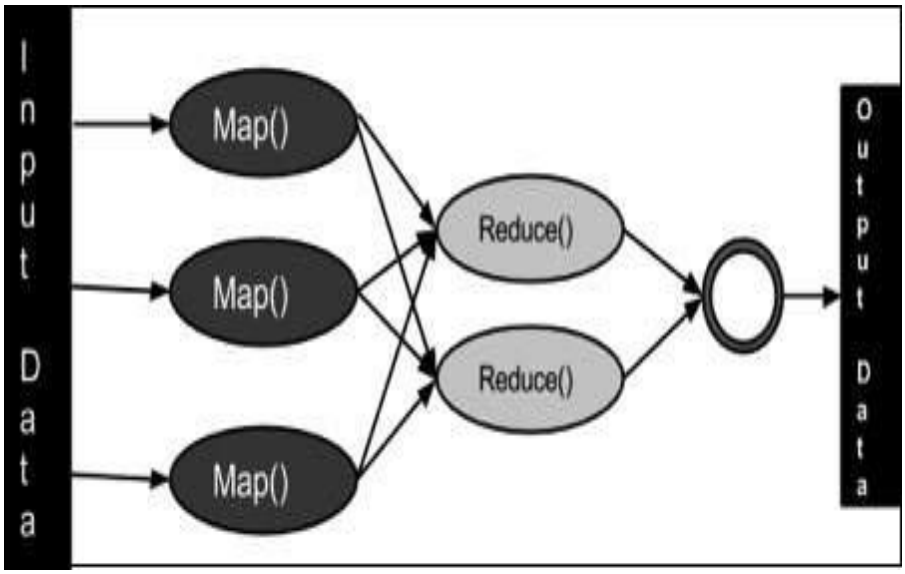


Figure 5:Map Reduce Algorithm

The input data is passed to map() function whose output is then sent as an input to the reduce().

Key-value pairs are fed as input and also obtained through output . This algorithm is the backbone of the fast and efficient processing of massive datasets on Hadoop.

Chapter-2

Literature Survey

Research paper: Performance analysis of various activation functions in generalized MLP architectures of neural networks

Author's Name: BekirKarlik and A. VehbiOlgac

Abstract:

Activation functions are used to define neuron's output considering the input /set of inputs. It is used to change the activation lavel of the node which means it converts input signal into output signal. There are various activation functions available which can be used according to the requirements of the neural network model. These activation functions are used in hidden layers and output layer of the network. The various activation functions available includes sigmoid functions, tanh function, conic section.

Bipolar-sigmoid function

$$g(x)=(1 - e^{-x}) / (1 + e^{-x})$$

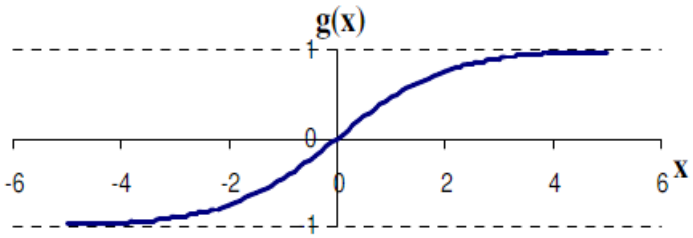


Figure 6: Sigmoid function

This function gives the output value in the range of (-1,1) and is good for networks in which backpropagation algorithm is used.

Hyperbolic Tangent function

$$\tanh(x) = (e^{x} - e^{-x}) / (e^{x} + e^{-x})$$

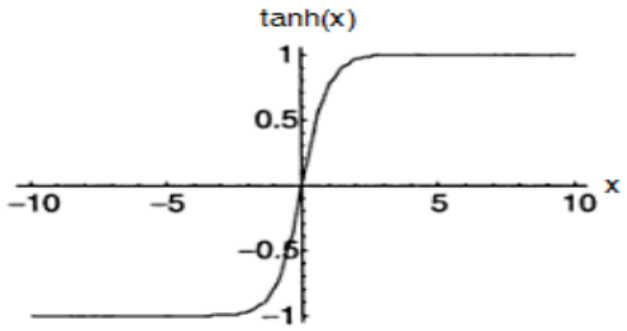


Figure 7: Hyperbolic tangent function

Hyperbolic tangent function is almost similar to sigmoid function in a way that both function output value ranges from (-1,1).

There are many other activation function available. After comparing these function it is known that function like step function are used in classification problems. ReLu function works better for recurrent neural networks whereas sigmoid function are faster for classification problems.

Research Paper: Learning Long term Dependencies with Gradient Descentis difficult

Authors: Yoshuabengio, Patrice Simard and Paolo Frasconi

Abstract:

RNs are very strong in their capability to display context and their results are often more reliable over static network. Although it would not be proper to train RNN for dependencies which are long-term. It is has been researched in the previous years that the system would not be tough enough to input noise or would not be properly trainable by gradient descent when the long-term data is required.

The gradient is either vanished or the input noise is tough for the system to handle; this conclusions have been drawn from this literature by Y.bengio.Deep feed-forward networks may face problems due to the vanishing gradient .The paper concludes that gradient becomes increasingly inefficient but not impossible to train Recurrent Networks on long term dependencies.

Research Paper: LSTM: A Search Space Odyssey

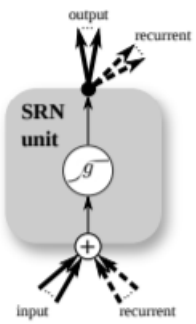
Authors: Jurgen Schmidhuber, Rupesh K. Srivastava, Klaus Greff, Jan Koutn´ık ,Bas R. Steunebrink

Abstract:

The use of variants of LSTM architecture have been depicted in this literature. The most commonly architecture of LSTM used is the vanilla LSTM which performs very good on various data.

Forget gate and the output activation function are the most important parts of LSTM and removing any of the component will depreciate the performance significantly. In order to prevent the system from destabilized learning and the propagation of unbounded cell state through the network we use the output activation function. Therefore, the GRU performs well because its cell state is bounded due to the coupling of forget gate and input

In comparison to the other methods, the neural networks would be difficult job to understand for some new practitioners. The modifications of LSTM architecture are depicted in the literature. Moreover, all the variants of the LSTM were explained in the literature with its components and with their areas of usage.



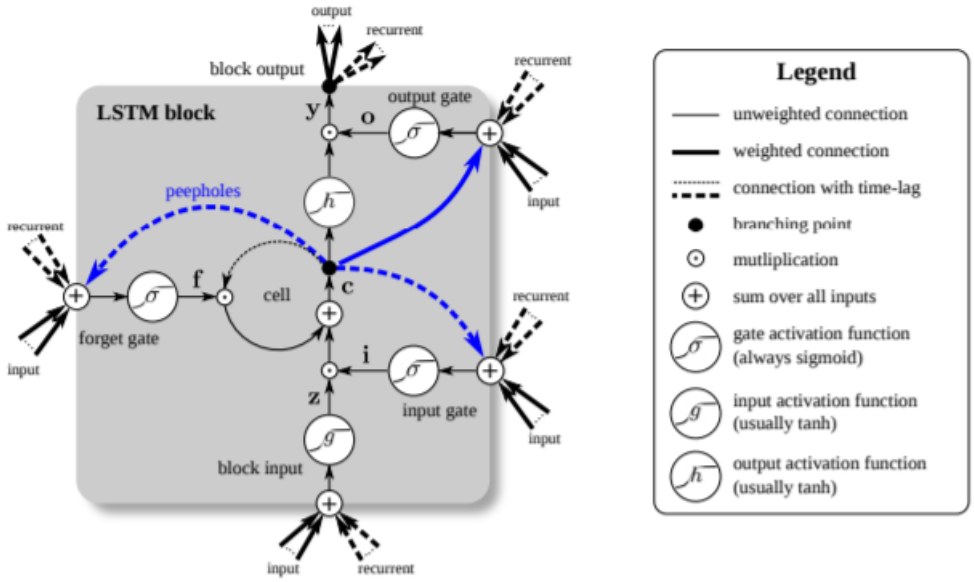


Figure 8: Simple recurrent network (SRN) unit (above) and a LSTM block (below) used in hidden layers

LSTM's advantages over other methods are explained taking the point in view the ability of the LSTM to remember things for their usage in the future. This literature helped us understanding that LSTM should be used in our project of stock market prediction because in the scenario of our project the trends of the stock market occurred in the past should be remember and thus it should be predicted that what will be the trend of the stock market in the future.

Research Paper: Dropout: a simple way to prevent neural networks from overfitting

Authors: N. Srivastava, G. Hinton, A Krizhevsky, Ilya Sutskever, R Salakhutdinov

Abstract:

Neural networks having number of hidden layers are known as deep neural networks. They are very effective ML systems. But large networks having multiple layers become difficult to use as they are slow. Overfitting is also the problem with these types of networks. Overfitting occurs when a network models the training data too good. In this, a neural network learns everything including the noise which hampers the performance of the model negatively. To overcome this problem a technique known as Dropout is used.

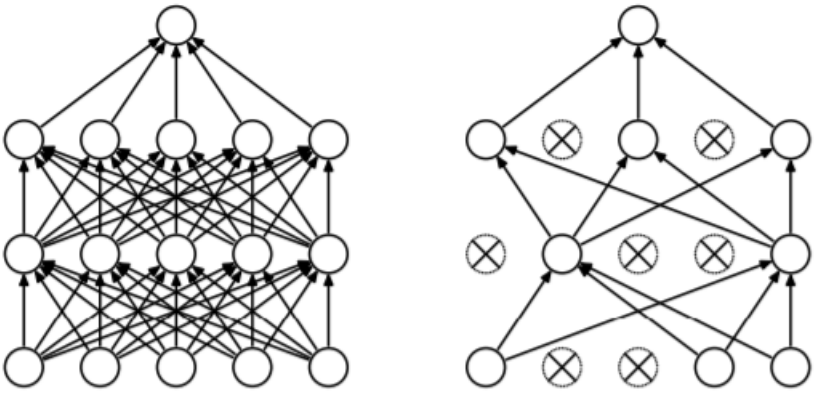


Figure 9: Dropout Neural Network

In this technique, random neurons are chosen and dropped from the Network including their connections to the other neurons. With addressing the overfitting problem it also makes combination of different neural network architecture effective.

However, during the testing of the model complete neural network without dropout is used. But unlike the same weight, the weight of the dropped neurons is adjusted by multiplying their actual weight by their probability of getting dropped. The probability can be assigned to the network or can be set 0.5 simply.

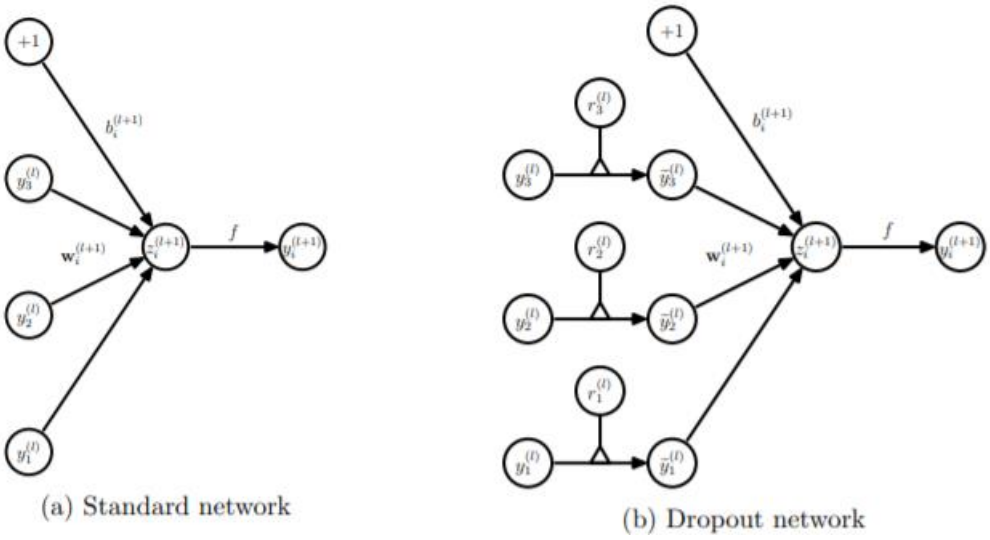


Figure 10: Basic Operation of Standard and Dropout Network

Research Paper: A Review Paper on Big Data and Hadoop

Authors: Sumit Kumari

ABSTRACT:

The three terms of Big Data are being discussed as the 3V's of Big Data which are:

1) Volume of Data: The amount of data in current world is gigantic in volumes and this fact should be put into some useful deed.
2) Variety of Data: The data available could be in structured form, semi-structured form or unstructured from. (images/texts/videos etc)
3) Velocity of Data: The data available is also increasing at an astonishing rate.

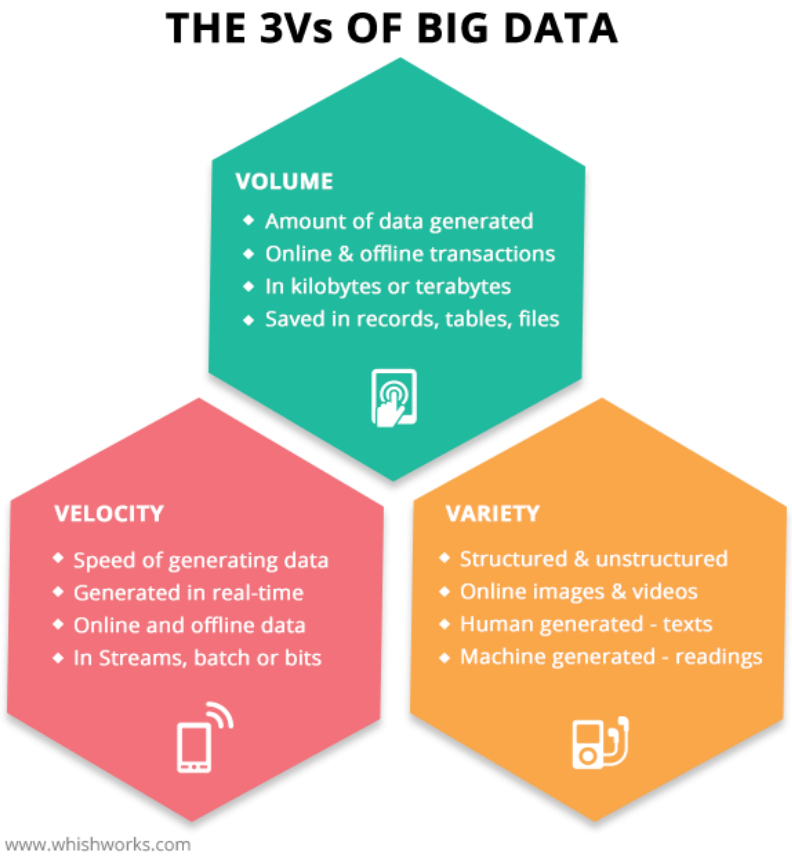


Figure 11:3Vs of Big Data

The paper further discusses the challenges related to the Big data that are:

- storage,
- processing power,
- information growth and
- data issues.

Hadoop is a solution to all the challenges and the three V's of Big Data because

- It can store a lot of data in HDFS (Storage challenge),
- It can work on all kind of data no matter of its form -Structured/Semi-structured or unstructured (Information Growth Challenge)
- and process that simultaneously in a distributed environment using Map Reduce. (Processing Power)

This research paper provided me the knowledge about big data and hadoop's components and about its architecture.

Research Paper: HADOOP CONFIGURATION AND IMPLEMENTATION IN VIRTUAL CLOUD ENVIRONMENT

Authors:Akanksha and Nasrullah

ABSTRACT:

The paper discusses about the architecture of HDFS and working of map-reduce is also discussed but the main focus of this paper is on the steps to configure Hadoop on a virtual environment which are as follows:

1) Install VMware Workstation Player.
2) Download ubuntu iso file to make a virtual machine of ubuntu on virtual machine.
3) Download Java (requirement for Hadoop installation).
4) Download Hadoop.bin.tar.gz file from internet and then install it.
5) Edit the bashrc file and set environment variables.

Hadoop is now installed on our system and we just have to run it now.

In order to start Hadoop , we have to perform two commands on the terminal which are as follows:

- start-dfs.sh
- start-yarn.sh

After running Hadoop now we can run its applications such as hive ,pig and sqoop etc by just writing their names.

This research paper helped me a lot in setting up Hadoop in ubuntu and execute the analysis of the stock market.

Research Paper: MapReduce and Its Applications, Challenges, and Architecture: a Comprehensive Review and Directions for Future Research

Authors:Seyed and Nima Jafari

Abstract:

The applications of Map-Reduce are discussed by explaining an example of word count.The architecture and working of Map Reduce has also been discussed :

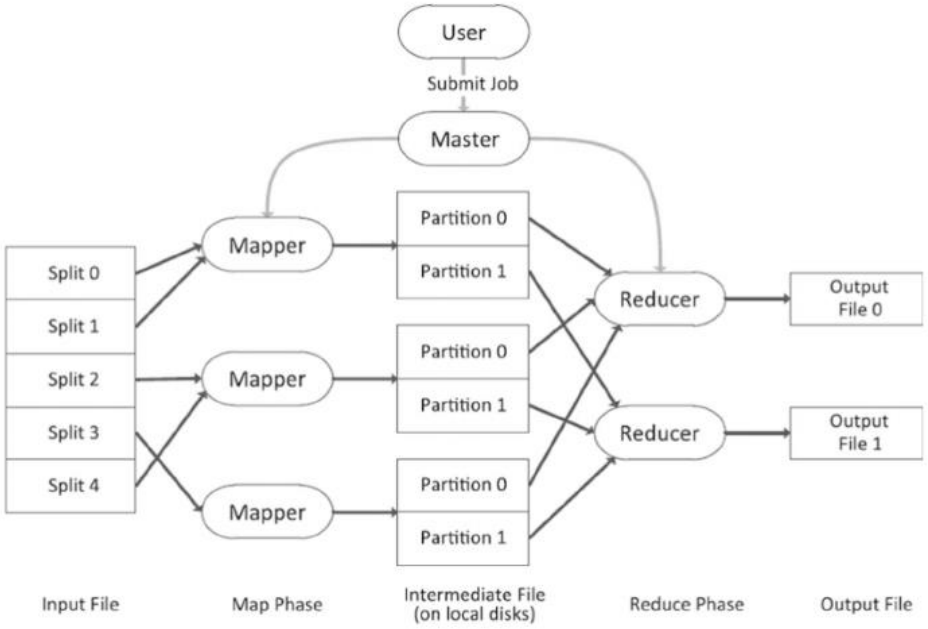


Figure 12:Partitions in Map Reduce

The input to the Mapper class is the preprocessed data. And after getting the input it performs three operations: Tokenizing input ,Mapping and Shuffle and sort and thus the partitions are made. Then the Output of Mapper Class is fed as the input to the Reducer Class which further performs the operations of searching and performing and thus the desired output is produced in no delay.

The implementation of map-reduce in google mapreduce, Hadoop,Hadoop+,GridGrain and Mars.

The applications of Map reduce were also discussed and it was found useful in distributed grep,word count,tera sort,inverted index and ranked inverted index ,Spark and Term-Vector.

Research Paper: Hive: Processing Structured data in Hadoop.

Authors: Anish Gupta, Manish Gupta

Abstract:

This research paper revolves around the functionality of hive. The data types used in hive could be primitive(int,string,float) or complex(list,structs). The architecture of hive is as follows:

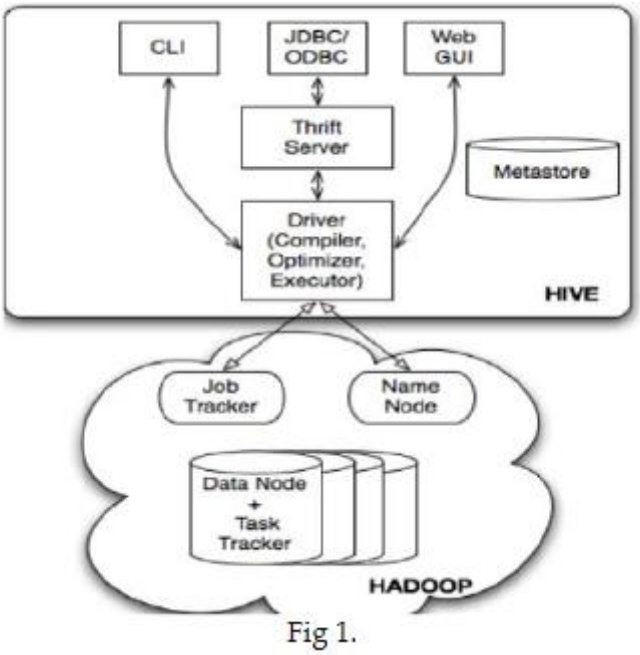


Fig 1.

Figure 13:Hive Architecture

Description of Architecture : CLI is the user interface used .Thrift Server displayed in the figure is the client API to perform Hive queries. And the driver is responsible for maintaining the lifecycle of the hive statements.

Some Commands were discussed: load ,create drop etc;

There were two types of tables :managed table and external table discussed in the paper.

The paper helped me in knowing the working of hive in Hadoop.

Chapter-3

System Development

For Prediction: Various open-source libraries like NumPy, Pandas, Matplotlib are used to import data, preprocessing the sample data and for the visualization of the results. We build the LSTM model using library keras. Keras is a python library used for development of neural networks. MSE and MAE are calculated using scikit-learn.Python is the programming language used in our project.

For Analysis: In our project the data of New York stock exchange has been used for the analysis purpose. The data is taken from the local system to HDFS using sqoop and then also the data is imported and inserted into the table of hive where the queries are acted upon on the data and various parameters are calculated.

1.1.Scenarios

In this report we use a easy and effective model for stock market prediction and analysis. The stock market characteristics used as input data include the opening price, lowest and highest price during the day, closing price of the day the volume and the adjusted close of stocks traded during the day. The model is used to predict the closing price of the next day and to analyze the stock market data to determine the particular stock which would give maximum profit. The google stock price data of last 15 years is used and the stock price data of new York stock exchange have been used.

The total sample data is further subdivided into two sets for model training and testing. 80% of the data was used to train the model and the next 20% of sample data is used to test the model for its performance and accuracy. The input data preprocessing was done as required by the model using python libraries numpy, pandas and scikit-learn. Root mean squared error (RMSE) and mean absolute error (MAE) were calculated to analyze the performance of the model and further improve the system and tune the model.

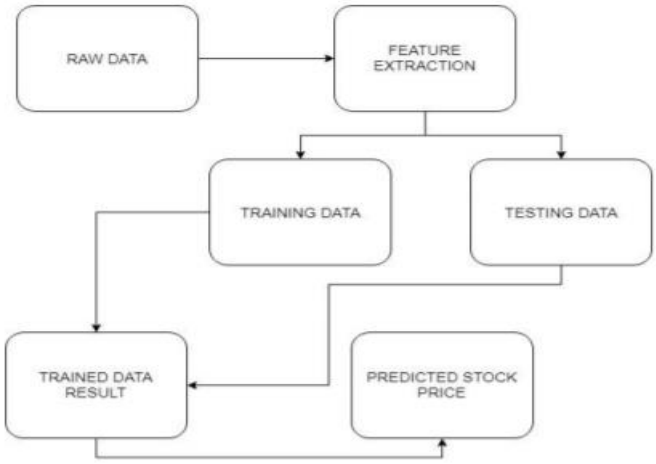


Figure 14:Data Circulation

For the analysis part of our project , the covariance of the stock price is also calculated and if the assets are moving along with each other then the stocks will not change its trend and if the assets are not moving together then the assets will not show a stable trend and hence money should not be invested in that company's stock. The analysis of the stock market in Hadoop will be done by hive by storing the csv file in the hive table using the load command.

3.2 **Model Design**

The predicted results i.e. closing stock price of the next day depends only on the sample data collected of a particular stock market. The input data values are used to update the different cells states of the LSTM model. It does not include various other factors which might affect the stock price like economic factors like inflation, interest rates etc, politics, market psychology etc. However, the model try to calibrate itself with the help of input data and analyzedata and try to predict as accurately as possible.

Different training parameters known as hyperparameters like time steps, epochs etc. were used and changed to get the optimum result. The data was formatted and made as timeseries data using different timesteps like 20, 30, 60. As the LSTM model only takes the 3D vector form, the data was processed accordingly. The input 3D vector consist of samples, timesteps and features (independent variables) and the vector has a shape n_samples, n_timesteps and n_features. The n_samples are the input data rows. The n_timesteps are past value affected by the no. of timesteps used in the model. The n_features are data columns like opening price, highest price etc.

This data was fed into the input layer having 50 units. Further, LSTM layers were added having different number of units 30, 40, 50. A dense layer as a output layer was added which gives the forecasted value. A dropout of factor 0.2 was added to prevent the problem of overfitting. Overfitting is learning of details and noise in the training dataset by the model to an extent that it affects the model performance negatively for the testing data. Another parameter in neural network model is the no. of epochs.1 epoch is considered when the all the data is moved to and fro in a neural network . To train the model and for a model to acquire the important information this entire process of forward and backwards data propagation is done thousands of time.

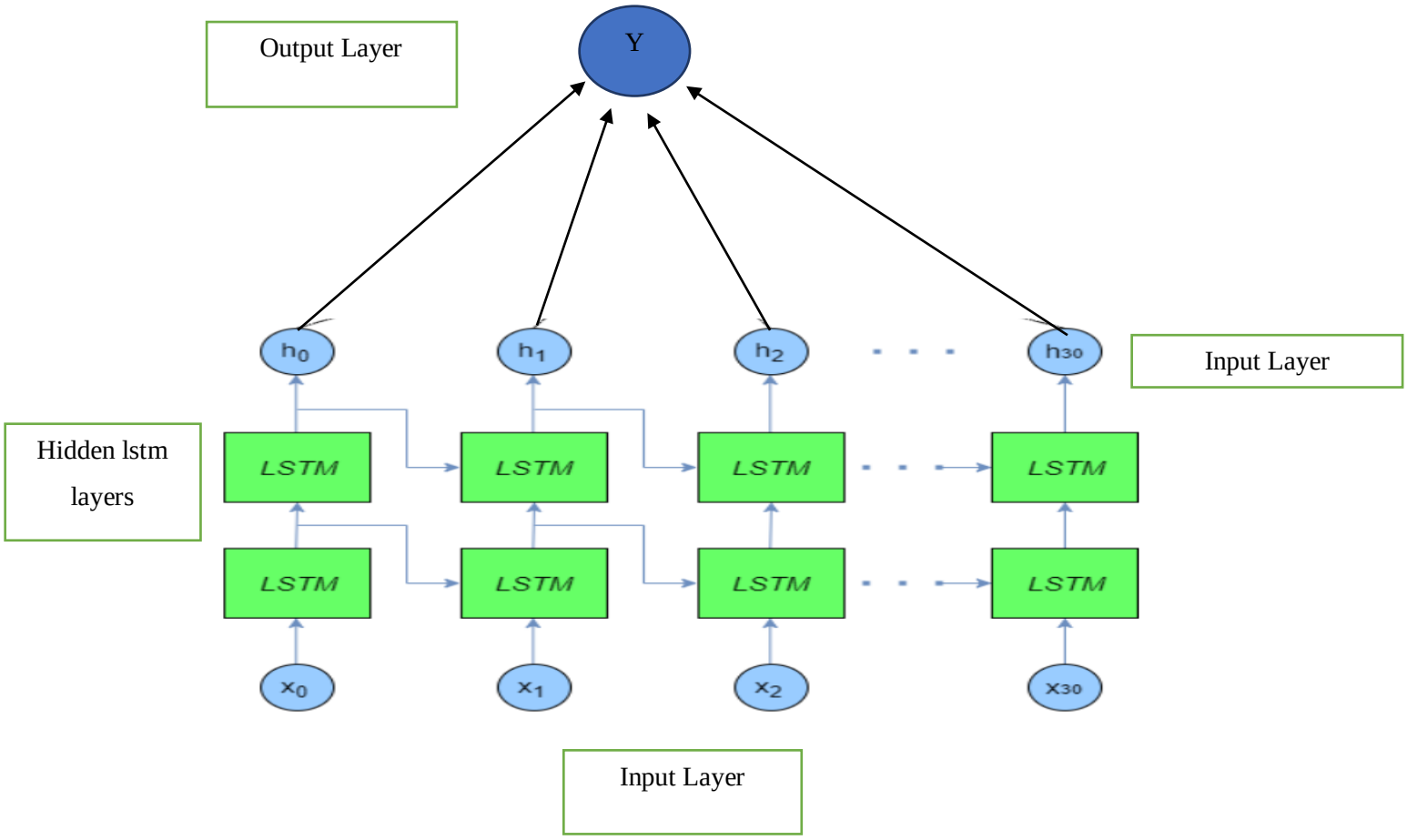


Figure 15: LSTM Architecture

Another significant parameter in the construction of neural network is optimizer. Optimization algorithms are used to minimize the error function E(x). Error function is a simple mathematical function which is based on different parameters of the model like weight and bias which keeps learning and updating as we train the model and are used to give the output of the model. There are many different optimizer algorithms like gradient descent, adagrad, adam. In our model we have used adam (Adaptive Moment Estimation) optimizer as it converges rapidly and learning rate of model is excellent. It also overcomes from the problems like high variance, vanishing learning rate which affect the loss function, faced by other optimizer algorithms.

FOR ANALYSIS PART:

To analyze the stock market data the covariance of the stocks are calculated using the formula as follows:

Average (high of a*high of b) - (Average (high of a) *Average(high of b)

If the value calculated from the above formula is found out to be positive and greater than one then the stock is a perfect choice for analyzing.

The average close of a particular stock is calculated for each month .

The average close of every stock is also calculated.

The maximum profit (open price – close price) is also calculated for each stock.

The average volume of stocks is also analyzed.

The max(high) of the stocks are also analyzed among various stocks.

After analyzing the above calculated parameters , such stock can be easily chosen by the investor which will have the maximum profit by chosing the stock having high positive covariance and greater difference between opening price and closing price.

Model Evaluation Criteria

To analyze the efficiency and performance of the model to predict stock market values RMSE, MAPE and MAE are calculated. RMSE is root mean square error and evaluate how closely true value and predicted value matches to each other.

$$\mathrm{MAE} = \frac{1}{N}\sum_{t=1}^{N}\left|d_t - y_t\right|,$$

$$\mathrm{RMSE} = \sqrt{\frac{1}{N}\sum_{t=1}^{N}\left(d_t - y_t\right)^2},$$

$$\mathrm{MAPE} = 100 \times \frac{1}{N}\sum_{t=1}^{N}\left|\frac{d_t - y_t}{d_t}\right|,$$

Where d_t is the real value and y_t is the forecasted value by the model and N is the total number of data fed to the model.

The lower the value of these evaluation criteria, better the efficiency of the model. However, if these 3 criteria lack consistency MAPE is chosen and model performance is analyzed using MAPE because out of these three MAPE is most stable criteria.

Data preprocessing

For any machine or deep learning model to be effective, data preprocessing is necessary step as the quality of data and its usefulness has a direct effect on the capability of our model to learn and adapt; hence, it is of utmost importance to preprocess the data before we train our model by feeding the data.

There are various concepts and steps in data preprocessing:

- Handling Null Values
- Standardization
- Handling Categorical Values

Handling NULL Values

In any actual and real world dataset there will always be some values missing known as null values because of various reasons. Regression, classification, classification or some other problem it doesn't matter, no model is capable of handling these NULL values on its own so we need to take care of that.

First we need to check if our dataset contains NULL values or not. In python we check that by using isnull() method.

```
df.isnull()
# Returns a boolean matrix, if the value is NaN then True otherwise
False


df.isnull().sum()
# Returns the column names along with the number of NaN values in
that particular column
```

Figure 16: isnull() method

There are different methods to take care of this. First and the easiest method is to drop the columns or rows that have NULL values. However, this is generally not a good option as while dropping these rows or columns we may loose important information which will affect the model.

Imputation

It is a process of replacing NULL values of the dataset by some other values using one of the many methods available. We can define a customized function to complete the task or use Imputer class provided by sklearn.

```
from sklearn.preprocessing import Imputer
imputer = Imputer(missing_values='NaN' ,strategy='mean')
imputer = imputer.fit(df[['C','D']])
df[['C','D']] = imputer.transform(df[['C','D']])
```

Figure 17: Imputer

Imputation can be done using following methods also:

- Replacing null values with mean or median.
- Replacing null values with the most frequent value.
- k-NN methods.
- deep learning methods.

Standardization

It is another important preprocessing step after handling the missing values. In standardization we convert the dataset values in such a way that the mean of the values is 0 and standard deviation is 1.

In our dataset there are many features which affect the output. Lets just consider two features opening price and volume of stocks traded. As volume of stocks generally will always be greater than the opening price. So if we feed these values directly into our model it will give more weightage to the volume column which is not ideal as opening price is also an important feature affecting the output value. So to get rid of this problem standardization is done.

$$z = \frac{x_i - \mu}{\sigma}$$

Figure 18: Formula of Standardization

Mean and standard deviation of the dataset is calculated and then to standardize each data point the difference of mean and value is calculated and then it was divided by standard deviation. Thus, this calculated value will replace the initial value.

Handling Categorical Variables

Categorical variables are basically the variables on which meaningful arithmetic operations can't be done. These values are discrete and not continuous eg gender etc. These variables are further categorized in:

- Ordinal categorical variables: A categorical variable is ordinal if there is some natural ordering of its possible values.
- Nominal categorical variables: If there is no natural ordering it is nominal.

Ordinal categorical variables can be handled by a method known as LabelEncoder which codes a label with a value between 0 and no. of distinct values -1.

Nominal categorical variables are handled by OneHot encoder. In this method we add n more columns to our dataset. The n here represents the no. of values nominal variable can have. The new columns will contain 1 if the value corresponds to that column otherwise 1.

Chapter-4

ALGORITHMS

Humans don't start thinking about everything from beginning but every time they come across something. They try to learn about anything using previous understanding. We don't just forget the past understanding and start from the scratch. Every previous information have some importance.

Traditional neural networks are not capable of this. Recurrent Neural Networks overcomes this problem. These networks have loops in nodes which allow them to retain information.

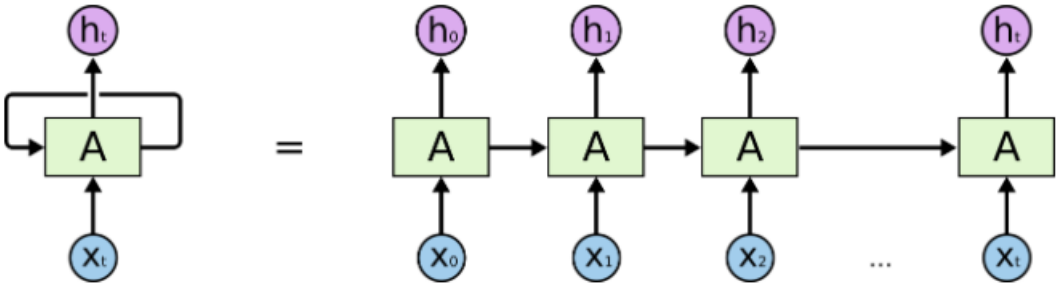


Figure 19:An unrolled recurrent neural network

In our project we have used a special type of RNN's "LSTMs" (Long Short Term Memory). As compared to standard version of RNN, LSTM performs better. They are built to get away from long-term dependencies. LSTMs have chain like architecture but unlike standard RNN which have only one neural network layer, there are four layers, which communicate with each other in a specific way.

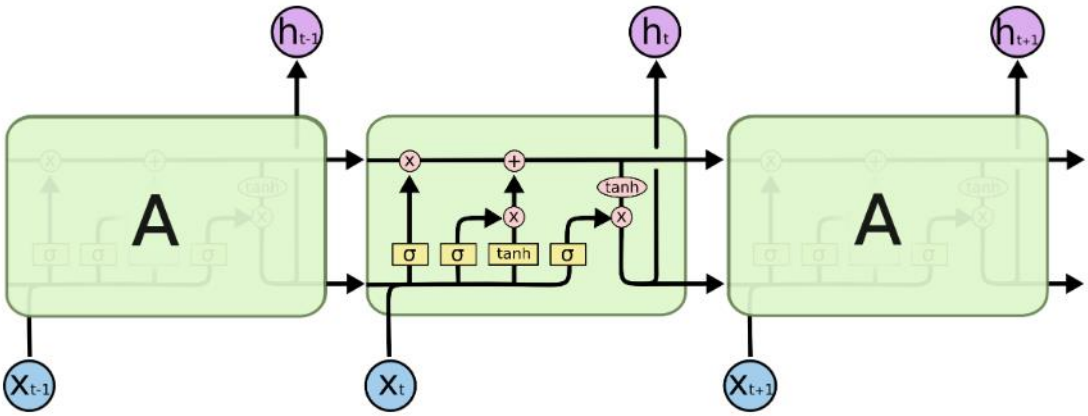


Figure 20: Repeating module in LSTM

The significant aspect of LSTM is that these are cell states. The horizontal line on top of the LSTM cell acts as conveyer belt through which information just follows the path without much variation. The LSTM does possess ability to add or delete the information in the LSTM cell with the help of various structures known as gates the cell contains.

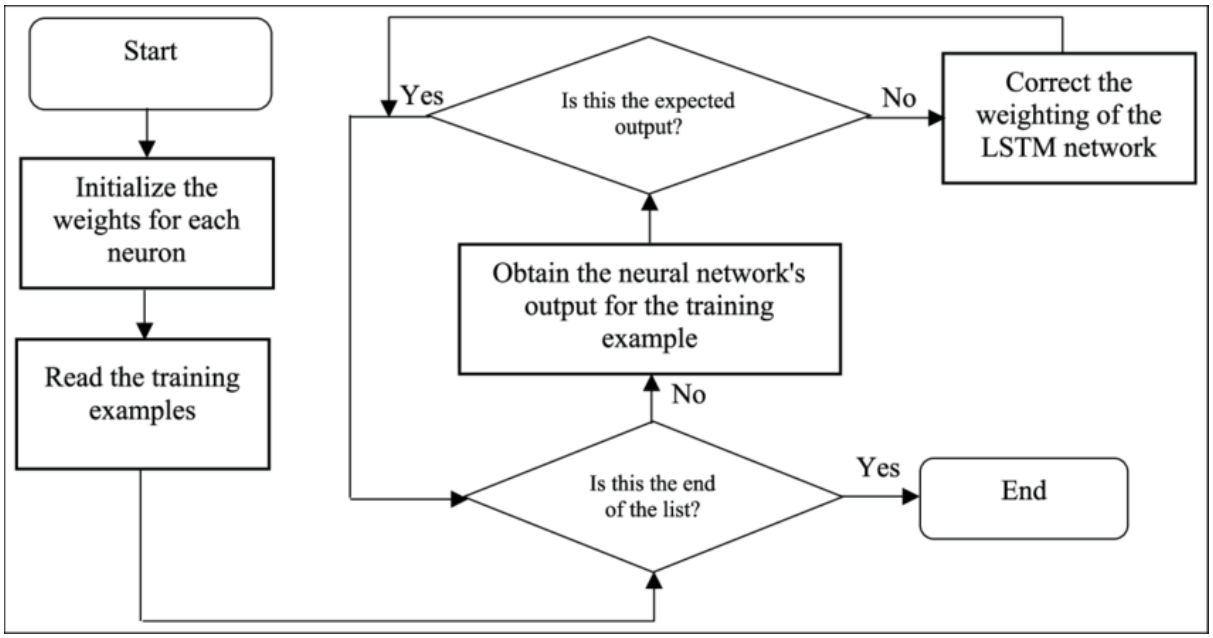


Figure 21: LSTM training Flow Chart

In our project we implement this algorithm using keras library of python. We insert the first input layer, add number of nodes and feed data i.e. independent variables. LSTM expects

input data to be in a particular 3D format of sample data, timesteps and number of features where-

Timesteps=RNN is run for these no. of timesteps i.e. how many previous data points affects the next output.

Features – These are simply the number of dimensions we feed at each time step. In our project there are 5 features namely opening, closing, highest, lowest price and volume of stocks traded.

```
# Adding the first LSTM layer and some Dropout regularisation
regressor.add(LSTM(units = 50, return_sequences=True,input_shape = (X_train.shape[1], features)))
regressor.add(Dropout(0.2))
# Adding a second LSTM layer and some Dropout regularisation
regressor.add(LSTM(units = 50,return_sequences=True))
regressor.add(Dropout(0.2))
# Adding the output layer
regressor.add(Dense(units = 1))
```

Figure 22: Implementing LSTM using keras

After that we input few more LSTM layers in our model. To avoid the overfitting of the data dropout value of 20% is used . Then a dense layer is added which is output layer and gives the predicted value.

It has various phases. First the model is trained by feeding large amount of data. After that the model is tested and based upon the results the model is improved and tuned by varying the hyperparameters like time steps, epochs, batch size etc.

For analysis part, we need to run Hadoop on our system. So first we will login a Hadoop user using the command below:

```
parth@ubuntu:~$ su - hadoopusr
Password:
hadoopusr@ubuntu:~$
```

Figure 23:Creating a Hadoop User

Then to start Hadoop, we need to start the dfs and yarn of Hadoop as follows:

```
hadoopusr@ubuntu:~$ start-dfs.sh
20/05/24 05:15:16 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-j
ava classes where applicable
Starting namenodes on [localhost]
hadoopusr@localhost's password:
localhost: starting namenode, logging to /usr/local/hadoop/logs/hadoop-hadoopusr-namenode-ubuntu.out
hadoopusr@localhost's password:
localhost: starting datanode, logging to /usr/local/hadoop/logs/hadoop-hadoopusr-datanode-ubuntu.out
Starting secondary namenodes [0.0.0.0]
hadoopusr@0.0.0.0's password:
0.0.0.0: starting secondarynamenode, logging to /usr/local/hadoop/logs/hadoop-hadoopusr-secondarynamenode-ubuntu.out
20/05/24 05:16:36 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-j
ava classes where applicable
hadoopusr@ubuntu:~$ start-yarn.sh
starting yarn daemons
starting resourcemanager, logging to /usr/local/hadoop/logs/yarn-hadoopusr-resourcemanager-ubuntu.out
hadoopusr@localhost's password:
localhost: starting nodemanager, logging to /usr/local/hadoop/logs/yarn-hadoopusr-nodemanager-ubuntu.out
hadoopusr@ubuntu:~$
```

Figure 24:Starting hadoop

As the Hadoop is started in our system now , we have to just go into the hive directory and write hive in the terminal to enter the hive platform as follows:

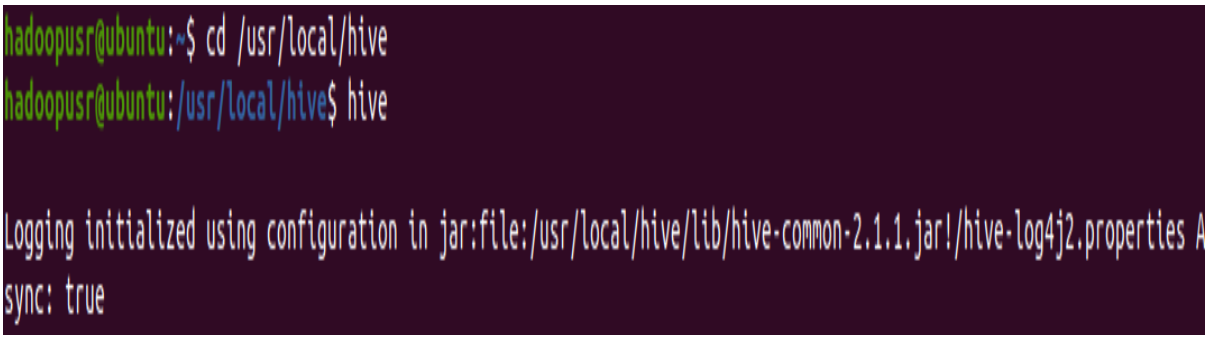


Figure 25:Running Hive

The database is created named project and table nyse is created with attributes of date, name, open , close , low , high, adjusted close price of the stocks in it. The command for csv file is given in the syntax that the row fields are separated by “,”.

We use the sqoop export command to send the file in HDFS and then to hive. Or we can directly transfer it to hive from local system using command-load. Then the covariance is calculated of the stocks stored in the hive table and the results are found on the terminal.

Chapter-5

Results and Performance Analysis

The developed prediction model was tested on the independent dataset and RMSE, MAE and MAPE values were used to analyze the performance and prediction capabilities of the developed model The 20% of initial data was used to test the model.

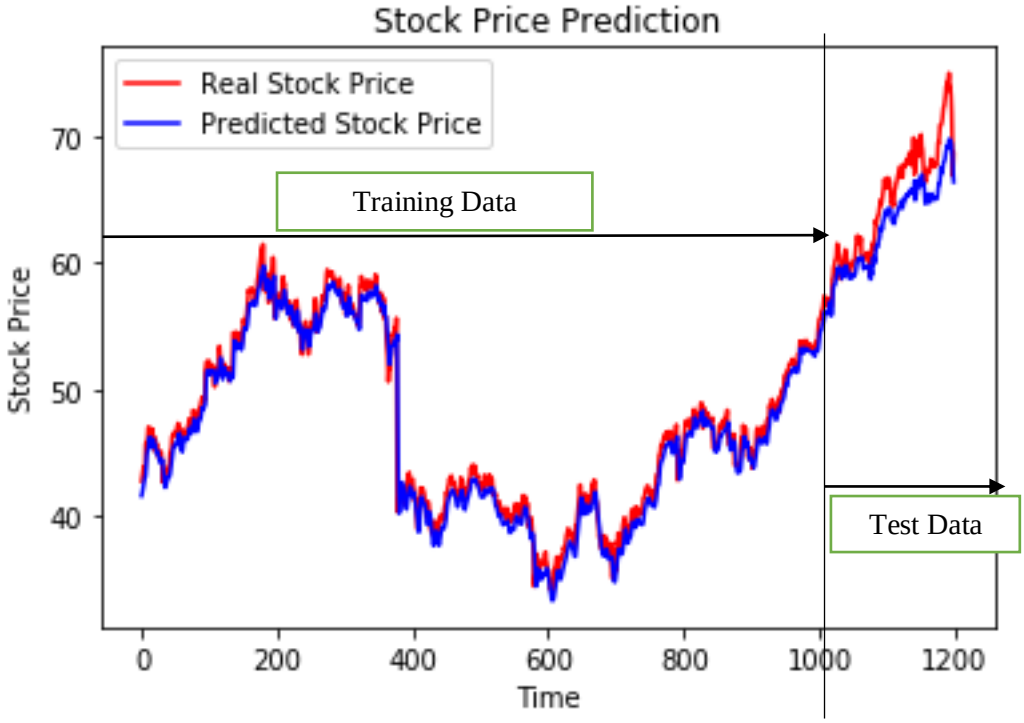


Figure 26: Predicted and Real Stock Price

The above graph shows the comparison between real stock prices and predicted stock prices by the model. As we only considered only 5 features and stock market get affected by various other factors also which is not easy to quantify, the above result predicted by model is quite good.

Criteria	Value
RMSE	62.007
MAE	62.081

The above values can further be improved by changing hyperparameters and the model can be further improved by tuning the model.

Result for Analysis:

```
Ended Job = job_202005221132_0002
OK
QRR     QTM     1       -0.13994965986395513
QRR     QTM     2       2.060000000021489E-4
QRR     QTM     3       0.0029299999999956583
QRR     QXM     1       -0.015941496598628646
QRR     QXM     2       0.005124999999964075
QRR     QXM     3       -0.01335799999998244
QTM     QXM     1       -0.003653287981855158
QTM     QXM     2       -0.026352499999998003
QTM     QXM     3       0.006057000000000201
QTM     QXM     4       0.02727107438016496
QTM     QXM     5       0.026688662131521212
QTM     QXM     6       0.05287052154195315
QTM     QXM     7       0.023126033057851103
QTM     QXM     8       0.022061224489796416
QTM     QXM     9       0.05976031746031918
QTM     QXM     10      0.0035079395085071408
QTM     QXM     11      0.018371745152354402
QTM     QXM     12      -0.0038603305785123165
Time taken: 73.616 seconds
hive>
```

Figure 27:Covariance of stocks

The stocks with high covariance can be known from the results and hence from the prediction and analysis software, we could be able to save some amount of investor's money.

Chapter-6

CONCLUSION AND FUTURE WORK:

The Long short-term memory (LSTM) architecture of deep learning field is used for prediction of stock prices unlike feed forward neural network uses feedback connections and is based on time series data thus incorporating past data impact on future. It is a great success as it was able to predict the future values with great accuracy.

Now we are planning to take this project a level up. We will now be working to predict the stock price in using the live and real time data. We will use different big data tools like apache kafka and Apache Spark to build real time data pipelines, apache spark for big data processing and different spark tools for streaming and machine learning.

REFERENCES

[1]Karlik, B., &Olgac, A. V. (2011). Performance analysis of various activation functions in generalized MLP architectures of neural networks. *International Journal of Artificial Intelligence and Expert Systems*, *1*(4), 111-122.

[2] Bengio, Y., Simard, P., &Frasconi, P. (1994). Learning long-term dependencies with gradient descent is difficult. *IEEE transactions on neural networks*, *5*(2), 157-166.

[3] Greff, K., Srivastava, R. K., Koutník, J., Steunebrink, B. R., &Schmidhuber, J. (2016). LSTM: A search space odyssey. *IEEE transactions on neural networks and learning systems*, *28*(10), 2222-2232.

[4] Kajitani, Y., Hipel, K. W., & McLeod, A. I. (2005). Forecasting nonlinear time series with feed-forward neural networks: a case study of Canadian lynx data. *Journal of Forecasting*, *24*(2), 105-117.

[5] Ghiassi, M., Saidane, H., & Zimbra, D. K. (2005). A dynamic artificial neural network model for forecasting time series events. *International Journal of Forecasting*, *21*(2), 341-362.

[6] Ghiassi, M., Saidane, H., & Zimbra, D. K. (2005). A dynamic artificial neural network model for forecasting time series events. *International Journal of Forecasting*, *21*(2), 341-362.

[7] Liao, Z., & Wang, J. (2010). Forecasting model of global stock index by stochastic time effective neural network. *Expert Systems with Applications*, *37*(1), 834-841.

[8] Tsay, R. S. (2014). Financial Time Series. *Wiley StatsRef: Statistics Reference Online*, 1-23.

[9] Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., &Salakhutdinov, R. (2014). Dropout: a simple way to prevent neural networks from overfitting. *The journal of machine learning research*, *15*(1), 1929-1958.

[10] McKinney, W. (2010, June). Data structures for statistical computing in python. In *Proceedings of the 9th Python in Science Conference* (Vol. 445, pp. 51-56).

[11] Pascanu, R., Mikolov, T., &Bengio, Y. (2013, February). On the difficulty of training recurrent neural networks. In *International conference on machine learning* (pp. 1310-1318).

[12] Elman, J. L. (1990). Finding structure in time. *Cognitive science*, *14*(2), 179-211.

[13]Datasets Used:
https://www.kaggle.com/kapilsharmma/stock-price-prediction/code
https://www.kaggle.com/prateekgupta/nyse_daily_prices_Q/code

Parth Report2

by Hari Singh

Submission date: 27-May-2020 06:08PM (UTC+0530)
Submission ID: 1332769489
File name: Parth_Plag-Check.docx (1.23M)
Word count: 7402
Character count: 36952

Parth Report2

ORIGINALITY REPORT

6%	2%	3%	5%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	Submitted to National College of Ireland Student Paper	1%
2	Le, Ho, Lee, Jung. "Application of Long Short-Term Memory (LSTM) Neural Network for Flood Forecasting", Water, 2019 Publication	1%
3	Y. Bengio, P. Simard, P. Frasconi. "Learning long-term dependencies with gradient descent is difficult", IEEE Transactions on Neural Networks, 1994 Publication	<1%
4	Submitted to Kookmin University Student Paper	<1%
5	Rachna Sable, Shivani Goel, Pradeep Chatterjee. "Empirical Study on Stock Market Prediction Using Machine Learning", 2019 International Conference on Advances in Computing, Communication and Control (ICAC3), 2019 Publication	<1%

6	link.springer.com Internet Source	<1%
7	python-course.eu Internet Source	<1%
8	www.bvicam.ac.in Internet Source	<1%
9	sefiks.com Internet Source	<1%
10	Submitted to University of Glasgow Student Paper	<1%
11	www.frontiersin.org Internet Source	<1%
12	shodhganga.inflibnet.ac.in Internet Source	<1%
13	tslbloggerpage.blogspot.com Internet Source	<1%
14	Submitted to University of Nottingham Student Paper	<1%
15	commandstech.com Internet Source	<1%
16	Submitted to Rochester Institute of Technology Student Paper	<1%
17	Submitted to Central Queensland University Student Paper	

		<1%
18	people.idsia.ch Internet Source	<1%
19	Submitted to CSU, Fullerton Student Paper	<1%
20	Submitted to Middle East Technical University Student Paper	<1%
21	Amrit Pal, Kunal Jain, Pinki Agrawal, Sanjay Agrawal. "A Performance Analysis of MapReduce Task with Large Number of Files Dataset in Big Data Using Hadoop", 2014 Fourth International Conference on Communication Systems and Network Technologies, 2014 Publication	<1%
22	Submitted to RDI Distance Learning Student Paper	<1%
23	intelligentonlinetools.com Internet Source	<1%
24	ajaykumarjogawath.wordpress.com Internet Source	<1%
25	Sepp Hochreiter, Jürgen Schmidhuber. "Long Short-Term Memory", Neural Computation, 1997	<1%

	Publication	
26	Submitted to University of Hull Student Paper	<1%
27	Submitted to University of Ulster Student Paper	<1%
28	Submitted to Bolton Institute of Higher Education Student Paper	<1%
29	Submitted to Study Group Australia Student Paper	<1%

Exclude quotes	Off	Exclude matches	Off
Exclude bibliography	Off		

JAYPEE UNIVERSITY OF INFORMATION TECHNOLOGY, WAKNAGHAT

PLAGIARISM VERIFICATION REPORT

Date:15-07-2020

Type of Document (Tick): PhD Thesis | M.Tech Dissertation/ Report | B.Tech Project Report ✓ | Paper

Name: Parth Verma **Department:** CSE **Enrolment No** 161241

Contact No. 7807211243 **E-mail.** Parthcse007@gmail.com

Name of the Supervisor: Dr. Hari Singh

Title of the Thesis/Dissertation/Project Report/Paper (In Capital letters): Stock Market Prediction And Analysis using Deep Learning And Hadoop

UNDERTAKING

I undertake that I am aware of the plagiarism related norms/ regulations, if I found guilty of any plagiarism and copyright violations in the above thesis/report even after award of degree, the University reserves the rights to withdraw/revoke my degree/report. Kindly allow me to avail Plagiarism verification report for the document mentioned above.

Complete Thesis/Report Pages Detail:

- Total No. of Pages =59
- Total No. of Preliminary pages =9
- Total No. of pages accommodate bibliography/references =2

Parth

(Signature of Student)

FOR DEPARTMENT USE

We have checked the thesis/report as per norms and found **Similarity Index** at 6 (%). Therefore, we are forwarding the complete thesis/report for final plagiarism check. The plagiarism verification report may be handed over to the candidate.

(Signature of Guide/Supervisor) **Signature of HOD**

FOR LRC USE

The above document was scanned for plagiarism check. The outcome of the same is reported below:

Copy Received on	**Excluded**	**Similarity Index (%)**	**Generated Plagiarism Report Details** (Title, Abstract & Chapters)	
	• All Preliminary Pages • Bibliography/Images/Quotes • 14 Words String		Word Counts	
Report Generated on			Character Counts	
		Submission ID	Total Pages Scanned	
			File Size	

Checked by
Name & Signature **Librarian**

……

JAYPEE UNIVERSITY OF INFORMATION TECHNOLOGY, WAKNAGHAT

PLAGIARISM VERIFICATION REPORT

Date:15-07-2020

Type of Document (Tick): PhD Thesis | M.Tech Dissertation/ Report | B.Tech Project Report ✓ | Paper

Name: Parth Verma **Department:** CSE **Enrolment No** 161241

Contact No. 7807211243 **E-mail.** Parthcse007@gmail.com

Name of the Supervisor: Dr. Hari Singh

Title of the Thesis/Dissertation/Project Report/Paper (In Capital letters): Stock Market Prediction And Analysis using Deep Learning And Hadoop

UNDERTAKING

I undertake that I am aware of the plagiarism related norms/ regulations, if I found guilty of any plagiarism and copyright violations in the above thesis/report even after award of degree, the University reserves the rights to withdraw/revoke my degree/report. Kindly allow me to avail Plagiarism verification report for the document mentioned above.

Complete Thesis/Report Pages Detail:

- Total No. of Pages =59
- Total No. of Preliminary pages =9
- Total No. of pages accommodate bibliography/references =2

(Signature of Student)

FOR DEPARTMENT USE

We have checked the thesis/report as per norms and found **Similarity Index** at .....6..... (%). Therefore, we are forwarding the complete thesis/report for final plagiarism check. The plagiarism verification report may be handed over to the candidate.

(Signature of Guide/Supervisor) **Signature of HOD**

FOR LRC USE

The above document was scanned for plagiarism check. The outcome of the same is reported below:

<table>
<tr><td>Copy Received on</td><td rowspan="4">• All Preliminary Pages
• Bibliography/Images/Quotes
• 14 Words String</td><td rowspan="2"></td><td>Word Counts</td><td></td></tr>
<tr><td>Report Generated on</td><td>Character Counts</td><td></td></tr>
<tr><td rowspan="2"></td><td>Submission ID</td><td>Total Pages Scanned</td><td></td></tr>
<tr><td></td><td>File Size</td><td></td></tr>
</table>

Checked by
Name & Signature **Librarian**

……