Taimoor Arshad
01-134181-099
Usama Zaheer
01-134181-063

Stock Exchange Predictor

Bachelor of Science in Computer Science

Supervisor:
Miss Sahar Arshad

Department of Computer Science
Bahria University,Islamabad

January, 2022

Acknowledgement

I would like to express my deepest appreciation to all those who provided me the possibility to complete this report. A special gratitude I give to our final year project supervisor Miss Sahar Arshad, whose contribution in stimulating suggestions and encouragement, helped me to coordinate my project especially in writing this report.
Furthermore, a special thanks goes to my teammate Usama Zaheer who help me to assemble the parts and gave suggestion about tasks.
I have to appreciate the guidance given by other supervisor as well as the panels especially in our project presentation that has improved our presentation skills thanks to their comment and advice.

Taimoor Arshad & Usama Zaheer
Islamabad, Pakistan

January, 2022

Abstract

The stock market is known for its non-liner, unpredictable and dynamic nature. It has always been a hot and profitable place to learn. In the area of financial forecasting and forecasting, in-depth course applications have been shown to improve accuracy and yield better results. Machine learning-based stock prediction allows to forecast a company's stock value in the future. The whole point of stock market forecasting is to generate revenue. In this project we have used Long-Short Term Memory architecture for analysis and development of a stock exchange predictor. The suggested approach is thorough since it incorporates stock market data pre-processing and specialized reading algorithm to forecast stock market prices. Our goal is to use an effective prediction model and produce accurate results with a very low percentage of error.

Contents

List of Figures

Acronyms and Abbreviations

ANN Artificial Neural Network.

ARIMA Autoregressive Integrated Moving Average.

DT's Decision Trees.

GARCH Generalized Autoregressive Conditional Heteroskedasticity.

LSTM Long Short-Term Memory.

MSE Mean Squared Error.

NER Named Entity Recognition.

NYSE New York Stock Exchange.

RNN Recurrent Neural Network.

SEP Stock Exchange Predictor.

SVM Support Vector Machine.

Chapter 1

Introduction

1.1 Project background and overview

The stock market is a marketplace for investors to purchase and sell investments, such as stocks, which are ownership shares in a public corporation. When people want to buy stocks, they will usually buy them online through the stock market. Securities exchanges play a significant part to play in the accomplishment of trade and the general soundness of an economy. Stock exchanges can be found in all global financial institutions that control markets and create indicators where traders can be exposed for the purpose of bringing financial growth to their investment value. In the case of stock trading, the stock market is a place where stock prices are defined and where supply and demand can play in both primary and secondary markets. Healthy market levels can suggest significant investment in a sector or economy, and market prices are typically considered as a measure of business and investor optimism.
A stock market is simply a virtual marketplace where asset buyers and sellers are gathered to exchange standardized share instruments. Stock exchanges administer stock markets by accepting applications from public firms to list their shares for public sale. To become recorded, organizations should meet broad qualifying limits, intended to protect merchants, and guarantee a degree of standardization in shares across the market. Stock markets allow companies to access private equity funds, which can be used to finance business growth and development, or to purchase new assets. Businesses would be limited in the operations they could fund if they didn't have access to private investment, and hence would consequently not be able to really profit by the value in their organization. Likewise, stock exchanges make it achievable for entrepreneurs to eventually cash out their situations for benefit, by selling their portions in the open market. For people and dealers, stock exchanges additionally play a significant part to play as these permit brokers to produce a profit from their capital to such an extent that it is advantageous for funding to be put resources into the financial sectors. This dual functionality makes it important for both organizations and retailers as a way of teaching about capital course.
Our aim is to provide a solution for problem mentioned above. We have developed web application that predicts the trend of the stock exchange. The major goal of this web application is to fulfill the user's wishes and provide a better predicting environment for the user to work in effective way, so they can save money and invest more effectively. The most usual method to buy and sell stocks in the stock market is through stock trading, which involves buyers and sellers meeting to establish the trade's value. You can buy shares from existing investors who want to sell them, and vice versa, through a stockbroker. A stock market's goal is to make it easier for buyers and sellers to trade assets, hence lowering the

risks of investing. So, a stock market can be considered as a super-sophisticated market providing a linkage between buyers and sellers.

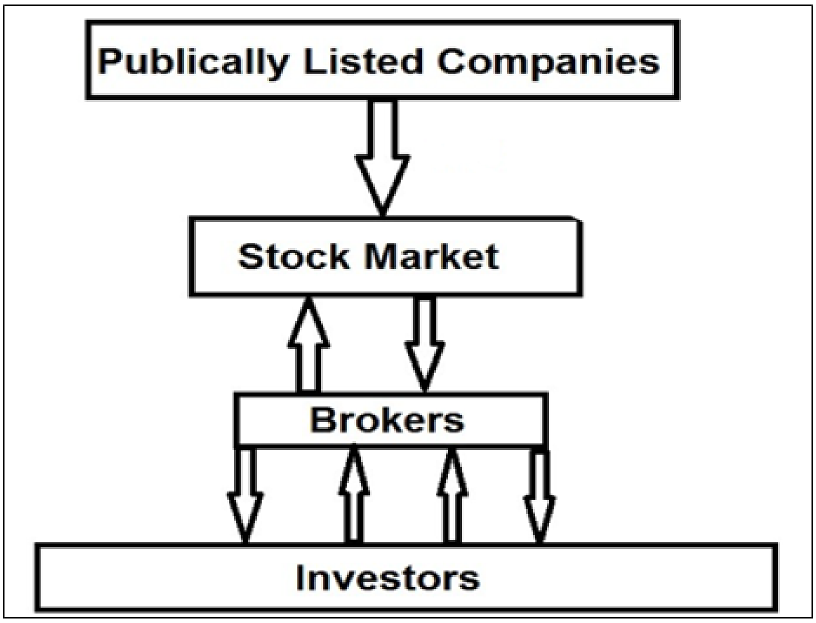

Figure 1.1: Stock Market Overview

1.2 Problem Description

Since a long time, people who wanted to invest in stock market have to go to brokers for interment in stock exchange. The brokers would notice the trends of stock markets and they would advise their clients that in which company's shares they should invest and when to buy or sale them. By this way, the investors had to totally rely on the broker and his advice. But nowadays when everything is online and transparent people can see the market and share prices and they can now buy shares directly. They can make their account on stock exchange market and link that to their bank accounts. This way they do not have to go to the broker and that concept is now changed. However, people are unable to predict the market trends and most of the times face huge loss.

1.3 Project Objectives

The stock market is extremely tough to predict for anyone. Many people desire to invest in the stock market, but they don't know where to start. Countless people invest on their own and end up losing a lot of money. If an effective algorithm can be developed to anticipate a short-term price for each stock, the level of investment and business prospects in the stock market may increase. The primary goal of this project is to provide an automated intelligent system to predict the stock market trends.

1.4 Project Scope

The "Stock Exchange predictor" is a web-based application that will assist the user in predicting the stock market's trends. It will assist the user in predicting the stock/share trends of KSE-100. The KSE 100 Index is meant to track the performance of the top 100 market capitalization companies in this industry. The essential goal of the KSE100 list is to have a benchmark by which the stock value execution can measure up over the period of time. In particular, the KSE 100 is intended to give investors a feeling of how the Pakistan value market is performing. Thus, KSE100 is identical to other indicators that measure different aspects of Pakistan's economy, such as the gross national product, consumer price index, and so on.

Free-Float computation can be utilized to build stock records for good market portrayal over those built based on overall market capitalization of organizations. Free-Float means it is the proportion of a company's total shares that are readily available for trading on the Stock Exchange. It typically excludes shares held by controlling directors, sponsors, and promoters, as well as government and other locked-in shares that are not available for trade in the regular course of business. It gives weight for constituent organizations according to their genuine liquidity on the lookout and is unaffected by closely owned large-cap companies.

During market hours, costs of the Index scrips at which exchanges are executed, are consequently utilized by the trading computers to compute the KSE100 Index and provides real time information to all trading workstations connected to the PSX trading computers.

Free-Float Calculation Methodology			
Total Outstanding Shares			**xxx**
Less	Government Holdings	xxx	
	Shares held by Directors / Sponsors / Senior Management Officers and their Associates	xxx	
	Shares in Physical Form	xxx	
	Shares held by Associate Companies / Group Companies (Cross Holdings)	xxx	
	Shares issued under Employees Stock Option Scheme that cannot be sold in the Open market in normal course	xxx	
	Treasury Shares	xxx	
	Any other category that are barred from selling at the review date	xxx	
Free Float			**xxx**

Figure 1.2: Free-float Calculation

User can also predict the individual company shares. User will be able to predict the variations in the stock market. User can register himself and it will be verified by the Admin. Admin will be able to display all the Stock information on the web-portal. Admin can modify the user's information. Admin would be able to control all the information of the stock.

The application also has the capability to generate graphs, histogram, crystal clear reports of all the stock which can be exported to pdf or csv etc.

1.5 Information about the Stock Market

1.5.1 Stocks

The equity of a stock of the company is represented by the equity of its stockholders. In relation to the total number of shares, a single share of stock indicates fractional ownership of the company. Shares are made up of a corporation's stock. The amount of them is mentioned at the time when the company is formed. The shareholders can allow the company to sell more shares of the company. In some of the cases, each stock has a declared value, and it reflects its value on the on the balance record of the company.

1.5.2 Shares of a Stock

A share of stock is a unit of ownership in the company. The number of shares decides how large the piece of possession in a company you have. If a company wants to divide the profit among its owners, it will be divided according to the number of shares a person owns. The owner of the shares has a documented stock certificate that ensures that ownership of the stock. If there is some voting in a company it will be dependent upon how many shares a voter owns.

1.5.3 Stock Market

The stock market is a marketplace for buying, selling, and issuing stocks. It is the gathering of stock buyers and sellers who represent ownership claims on companies. Most of the time, include securities listed on a public stock exchange as but can also be traded privately. Stock markets provide capital to companies which they can use to support and grow their firms. If a company issues a hundred shares of stock for $1000 each, the company will have a hundred thousand dollars to spend in its business. The stock market allows investors to participate in the profits of publicly listed businesses.

1.5.4 Stock Market Prediction

The process of attempting to anticipate the future value of a company's shares is known as stock market prediction.

1.5.5 Shareholders

Any individual, firm, or organization that possesses a stock in a company is known as shareholder. If the company does well and succeeds, shareholders usually get stated profits. In most companies' shareholders get right to vote on matters such as elections to the board of directors, the right to access the assets of a firm during its liquidation etc.

1.6 ML Classifiers employed for forecasting data

1.6.1 Artificial Neural Networks

Artificial neural networks (ANN's) are an area of artificial intelligence that is based on brain and nervous system research. In a network, nodes are grouped into layers, that begins with an input layer and ends with an output layer. Signals are transmitted (propagated) thru the connected nodes as they examine from previous reports and try to lessen prediction mistakes. As the system attempts to enhance its performance, the weights for signals between linked nodes are adjusted.

Investigating models is how neural networks are taught. Every model comprises known data and outcomes, producing a weighted probability link between the two that is kept in the net's information architecture. When training a neural network from a given sample, the difference between the network's processed output prediction and a target output is usually determined. It is an error. By using error and learning technique, the network then adjusts its weighted association. And with each iteration the neural network's output will get increasingly like the intended output. When enough changes have been made to predetermined criteria, the training will come to an end. In addition to the whole dataset, ANN can also provide an output result based on a sample of data.

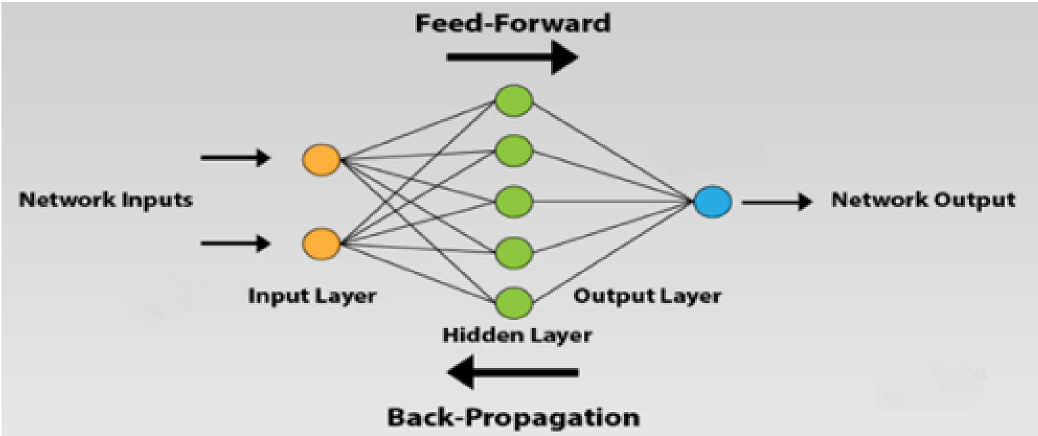

Figure 1.3: ANN Diagram

1.6.2 Decision Trees

Decision Trees (DTs) are a supervised learning method that is non-parametric. They are applied in the classification and regression of data. This is used to forecast a response to a data set. Decision Tree is one of the easiest classification algorithms to understand and interpret. The decision tree algorithm can be used for solving regression and classification problems too. The goal is to create a training model that can use to predict the class or value of the target variable by learning simple decision rules deduced from prior training data. Decision trees sort the instances along the tree from the root to a leaf node and the leaf node assigns classification to the example. Both root and record attributes are compared with each other. By following the results of comparisons, we jump to the next node by following the branch associated with that value.
Based on the target variable, there are two main varieties of decision trees. One is categorical variable decision trees, and the other is continuous variable decision trees.
In a categorical variable decision tree categorical targets variables are grouped into categories whereas in continuous variable decision tree is a continuous target variable decision tree.
The advantages of decision trees are it is easily understandable and easy to decipher. They can deal with problems with many outputs. The disadvantage of decision trees is it can produce trees that are too complicated. They are also very unstable like a very small change in data might produce very different results.

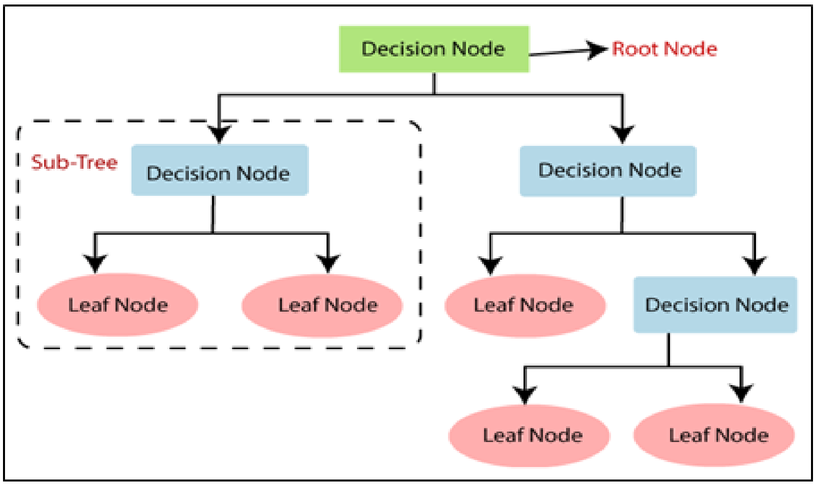

Figure 1.4: Decision Tree

1.6.3 Support Vector Machine

Support Vector Machines (SVM) will be used for predicting stock price. It's a supervised machine learning method that can handle both classification and regression problems. It is also referred to as a hybrid model. It is usually used to solve classification problems. SVM has become quite popular amongst other techniques because of its capacity to cope with complex nonlinear patterns.

It's a supervised learning technique that maximizes geometric margins while reducing clas-

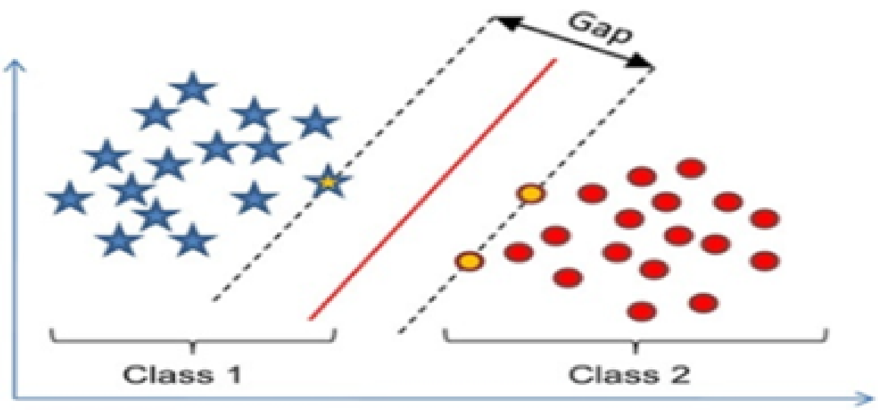

Figure 1.5: Support Vector Machine

sification error. That is why is also called maximum margin classifier. In a high-dimensional featured space, a maximum separating hyper plane is generated. On both sides of the hyper-plane, there are two parallel hyper planes that divide the data. SVM classification is immune to outliers due to a feature that allows it to ignore outliers and identify the hyper-plane with the greatest margin.

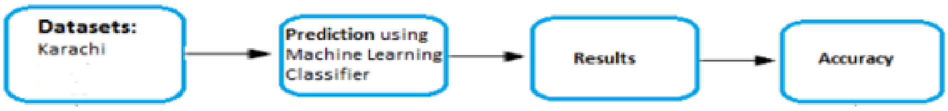

Figure 1.6: ML Algorithms Flow

1.6.4 LSTM

The full form of LSTM is Long Short-term Memory. The LSTM architecture is a type of artificial neural network that is used in deep learning. Unlike traditional forward feed neural networks, the LSTM incorporates feedback connections as well. It can handle individual data points as well as full data sequences. Long-range dependencies are difficult to capture

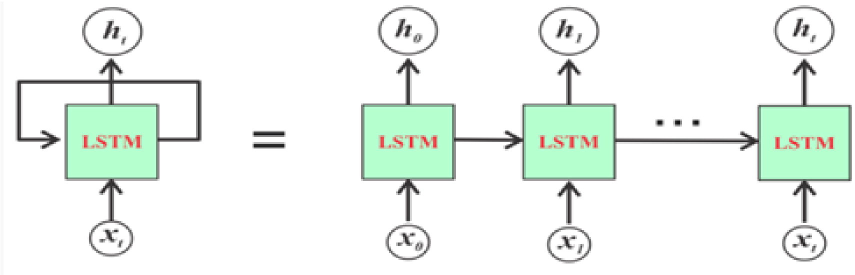

Figure 1.7: LSTM representation

with traditional recurrent neural networks. It indicates that if the data set is too large and there are several RNNs, a vanishing gradient problem may occur. This problem arises when the gradient decreases exponentially as it transmits down the layer while training a very deep neural network. The issue is that on some occasions, the gradient will be so minor that the weight will be unable to change its value and in the worst-case scenario, the neural network's ability to learn will be entirely disabled. This issue is very familiar in exceptionally deep neural networks. As a result, LSTM was developed to address the vanishing gradient problem in RNNs.

Long short-term memory is an alteration to RNNs hidden layer as it can remember RNN weights and inputs for a long time. Additional gates have been used to control which hidden cell information is exported as output and to the next hidden state. It is able to capture long range dependencies. There are total 3 gates in LSTM. Forget gate, Input gate and Output gate. The Forget gate deletes information from the cell state that is no longer valuable. Extra information is added to the cell state by Input gate. Output gate keeps track of previous inputs, and it decides hidden state's next value The LSTM mechanism has allowed the network to learn when to forget, ignore, or store information in memory cells.

For sequential models, LSTM is a popular deep learning algorithm. Two very famous example of LSTM are Apple's Siri and Google voice search. LSTM is the success story for those algorithms. Starting off with apple in 2003. Apple was the first large IT firm to integrate a smart assistance that is Siri into an operating system and the Siri was a by-product of some other company. Therefore, Siri was a company's adoption of a standalone application that was been purchased along with the creators who made it. It was in 2010. The initial reviews about Siri were that Siri was intense but after some time the users became more impatient with the failings and all too often it wrongly interpreted commands. And then

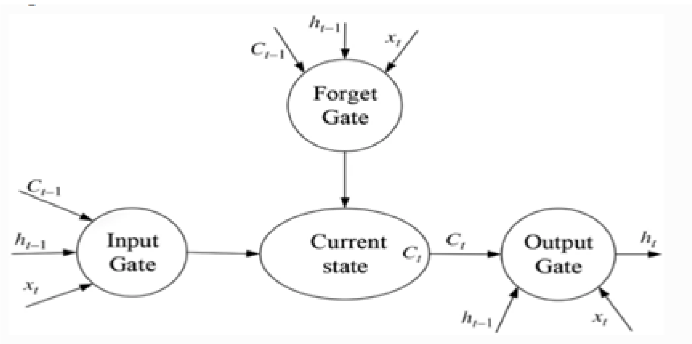

Figure 1.8: LSTM Unit Structure

no matter what you do there was no fix for it. It was at this time that Apple switched to a neural-based voice recognition algorithm for Siri. Some of the previous technique remain operational like applying hidden Markov models but most of the time you know DNN or deep neural network using LSTM was used. Even though users did not find any changes on the outside but from within it was a supercharged deep learning model.
Google also implemented LSTM. Google implemented google voice search somewhere around 2009. Google voice transcription has initially used something called as gaussian matrix model. This was nothing but an acoustic model and it was considered to be a state-of-the-art speech recognition for almost 30 plus years. It was in 2012 when there was a huge growth in deep neural network and when google implemented deep neural network using multiple layer networks there was a huge performance gap, but things really improved when recurrent neural network especially with LSTM RNN first launched on an android speech recognition in May 2012. Compared to deep neural network LSTM Rnn's have additional recurrent connections and memory cell that allows them to remember the previous data.

1.7 Comparative Analysis

Many analysts in past have conducted studies comparing different models and their results. This study aims to find the most effective predictive model for time series, with less errors and higher accuracy in the predictions. The results are measured by the **root mean square error (RMSE)** method. Hence referring to that **Dr Vaibhav Kumar**, who holds a PhD degree in the area of Deep Learning for Stock Market Prediction, published an article in which LSTM was compared with Arima model. On the dataset, predictions were generated, and LSTM outperformed Arima.

Table 1.1: RMSE Results

RMSE with LSTM	29.923716650748005
RMSE with Arima	72.640563493789940

Another analysis was performed by Dr John McCrae and Sai Krishna Lakshmi Narayanan. They conducted a comparative study of SVM and LSTM Algorithms for Stock Market Prediction at the National University of Ireland Galway. LSTM outperformed SVM in this case as well.

Table 1.2: RMSE Results w.r.t LSTM and SVM

RMSE with LSTM	399.39900
RMSE with SVM	682.63000

Dr John McCrae and Sai Krishna Lakshminarayanan conducted a comparative study of SVM and LSTM Algorithms for Stock Market Prediction at the National University of Ireland Galway [6].
LSTM outperformed SVM in this case as well.

Chapter 2

Literature Review

2.1 History of Stock Markets

In today's global economy, stock exchanges are amongst the most prominent feature. For economic growth, countries throughout the world depend on stock markets. The first real stock exchanges were not established prior to the 1500s. The first stock market of the world is usually associated in the past to Belgium. Despite this, must be often assumed that Antwerp possessed the first stock exchange system of the world. Antwerp was the business capital of Belgium and was also home to the powerful Van der Beurze family. Consequently, Beurzen was the first stock exchange.
Stocks were one thing that all of these marketplaces lacked. Even while the architecture was like stock exchanges at that moment, nobody was in fact trading company's shares. Government and individual debt were instead managed by the markets.
East India Company is the first publicly recognized traded company of the word. In 1602, shares on the Amsterdam stock exchange was issued by the Dutch East India company. Stocks as well as bonds were received by Shareholders. Respectively all shareholder was authorized to a certain proportion of the profit of East India Company.
The Stock Exchange of London was established in 1801. This is because companies were not authorized to provide shares prior to the 1825, there was a very restricted trading. The NYSE began trading shares on its inaugural day and quickly grew to become the world's most powerful exchange. The London Stock Exchange operated Europe whereas the New York Stock Exchange operated the United States and the whole world.
On 18th September 1947, Pakistan Stock Exchange (PSX) was established. Only five companies were listed at first, and the KSE 50 index was the first index introduced. Trading was done using an open outcry technique until 2002, when automated trading was introduced.
In October 1970, another stock exchange was built in Lahore. In October 1989, another stock exchange named "Islamabad Stock Exchange" was built in Islamabad. Its primary aim was to provide a trading infrastructure, data system, skillful resources, availability, and a fair marketplace to compete with the best as well as provide the requirements of less developed areas of Pakistan such as Northern areas.
All three Pakistani exchanges have their own trading interfaces, indices, management, and so on, and so had no common linkages. There was a conflict of interest among the investors as a result of this. As a result, in 2016, all three exchanges were combined, and the Pakistan Stock Exchange was born (PSX). Up till June 2021 there are 552 companies listed in PSX with a market valuation of almost 8,000 billion Pakistani rupees.

2.2 Studies to Analyze Stock Markets

Machine learning approaches for predictive systems have been introduced into financial markets as a result of computational advances. This contains studies that are primarily relied on long short-term memory (LSTM) and support vector machines (SVMs) to forecast stock market movements. SVMs are a one-dimensional model that can be used to address both classification and regression issues. This technique employs supervised learning. The Long Short-Term Memory (LSTM) is an architecture of recurrent neural network (RNN) that addresses linear problems. It is a domain of deep learning.

The LSTM, according to **M.Roondiwala** [9], is the most prevalent RNN design. LSTM establishes a processing device mainly known as memory cell that exchanges traditional artificial neurons in the secret network layer. Therefore, networks can efficiently connect memory and input and make it appropriate to actively take data array through time with a high predictive limit. This prediction can also be executed on the shares of NIFTY50, as indicated in a particular article.

Manoj S Hegde [7] researched if LSTM which is type of RNN, able to solve very complicated linear problems, and whether RNN (Recurrent Neural Networks) can be used to forecast stock values.
Tao Xing and Yuan Sun [12] proposed a model that takes into account a company's history equity share price and uses an RNN approach called LSTM. This suggested method takes into consideration a share's past data and applies prediction to a certain attribute. Shares have an opening price, a day high, a day low, a prior day price, a close price, and a trading date. The proposed model forecasts a share price over a period of time by analyzing time series data.

In Zbikowski (2015) [4] study, the author utilized SVM and an independent variable selection procedure within technical stock market indicators. The goal of this study was to see if a customized Support Vector Machine classifier could be utilized efficiently to forecast short-term stock market patterns. The writers analyzed data of each day to come up with a price estimate for two days. They looked at the stock market's performance and hence evaluate classifiers.
Ballings (2015) [2] study was for evaluating more than one classifier for stock price direction prediction. Different classifiers are utilized to forecast the direction of a stock of company for coming years. The authors of this study matched the outcomes of these classifiers to join methods, such as Random Forests, Kernel factory and Ada Boost. These ensemble procedures used several classifiers, most importantly utilized the similar algorithms, consequently in different classifications.
Another unique study was done by **Schumacher and Chen (2009)** [10]. SVM was used in conjunction with textual analysis in that study to look at the consequence of news stories about the prices of stock. They developed a statistical Procedures that uses machine learning for the study of financial news by combining different word-based representations such as Bag of Words and Named entities etc.

Chapter 3

Requirement Specifications

3.1 Existing System

In the past, some traditional statistical tools have been used to predict the trends of stocks such as **ARIMA** (Autoregressive Integrated Moving Average), **GARCH** (Generalized Autoregressive Conditional Heteroskedasticity). **ARIMA** makes their prediction linearly whereas **GARCH** tend to cluster. These previous traditional statistical methods consequently made a huge impact on non -linear attributes of financial time series. Now-a-days these statistical methods have been replaced with Machine learning techniques such as **LSTM** (Long Short-Term Memory) and **SVM** (Support vector Machine). These techniques have proven to be more suitable for stock market prediction. In stock market prediction, time series are the econometric methods that are used to predict the prospective trends of stock based on previous history.
Pros of time series:

- Identify patterns.
- Helps to clean data.
- Predicts the future values.

Cons of time series:

- Training time is comparatively expensive.
- It is sensitive to various initializations of random weights.

3.2 Proposed System

Proposed can easily fill the flaws and resolve the problems. Proposed system is **LSTM** (Long Short-Term Memory), supervised learning algorithms used for classification, regression and outliers detection. In these supervised learning algorithms, data is labeled. It can solve linear and non-linear problems. Pros of **LSTM** are as follows:

- Needs a lot of data.
- There are no prerequisites like stationary, no level shifts.
- Uses neural networks to model non-linear functions.

3.3 Requirement Specifications

As mentioned above that existing systems have some flaws. That's why we use proposed system which is Long Short-Term Memory. As we know that, in time series forecasting, **ARIMA** (autoregressive integrated moving average) model has been most conventional linear model. It does not support nonlinear patterns. On the contrary, LSTM purposes methodology which is non- linear model. Support vector machines (SVMs) also used to solve nonlinear prediction problems. Therefore, LSTM model is much better than ARIMA model in terms of predicting stock trends and produce much better accuracy and results.
Functional requirement:

- The system should be able to make an approximate price of shares.
- The system should collect accurate data from the stock market website in consistent manner.
- Using Time series analysis, user can also look previous data information of the stocks.

Non-Functional Requirement:

- The accuracy of the system should be better.
- The system should have user friendly interface.
- In short amount of time, the system should perform effectively

3.4 Use Case Diagram

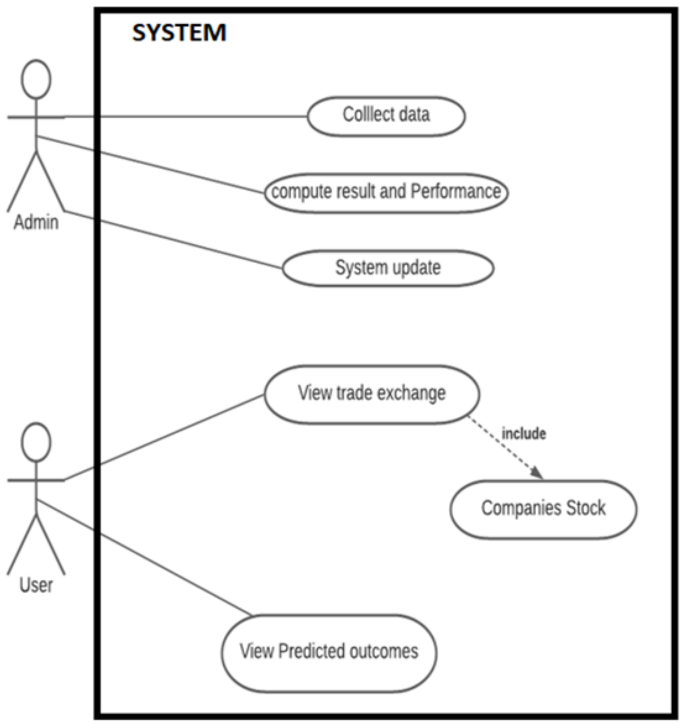

Figure 3.1: Use Case Diagram

Chapter 4

Design

4.1 System Architecture

The system operates on .CSV file, which have a record of all the attributes of stock exchange data. With the above acknowledgement, data preprocessing is performed and refined to predict an information that is closed for desired future date. The CSV files are taken from Kaggle website. Once the complete understanding is available, it catered for the LSTM algorithm to perform stock market prediction and gave a complete visual of data by using python. After that, this prediction of financing is partitioned into time scales such as months, days, hours etc. and, consumer will give appropriate feedback of the prediction.

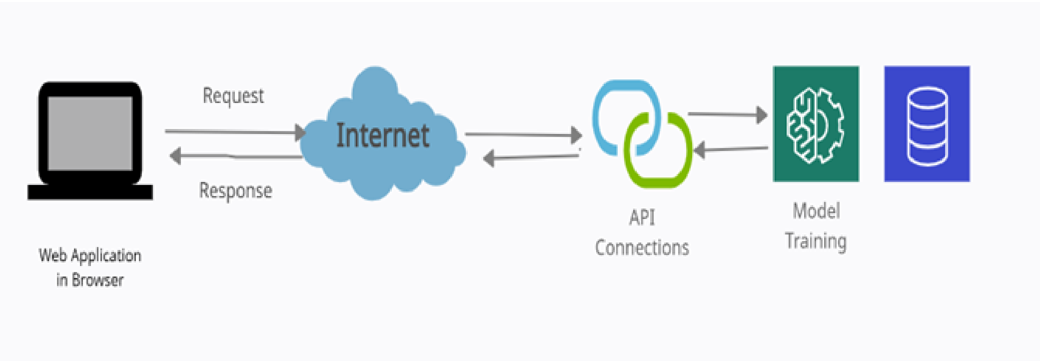

Figure 4.1: Client Server Architecture diagram of proposed system

4.1.1 System Context Diagram

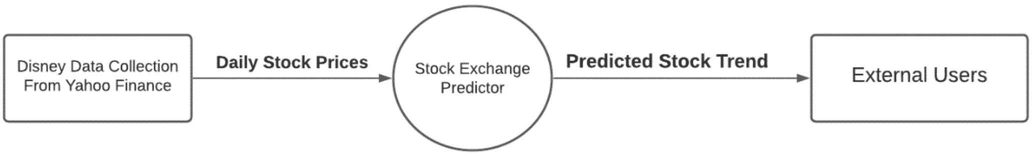

Figure 4.2: System Context Diagram

4.1.2 Flow Chart for the proposed flow of events

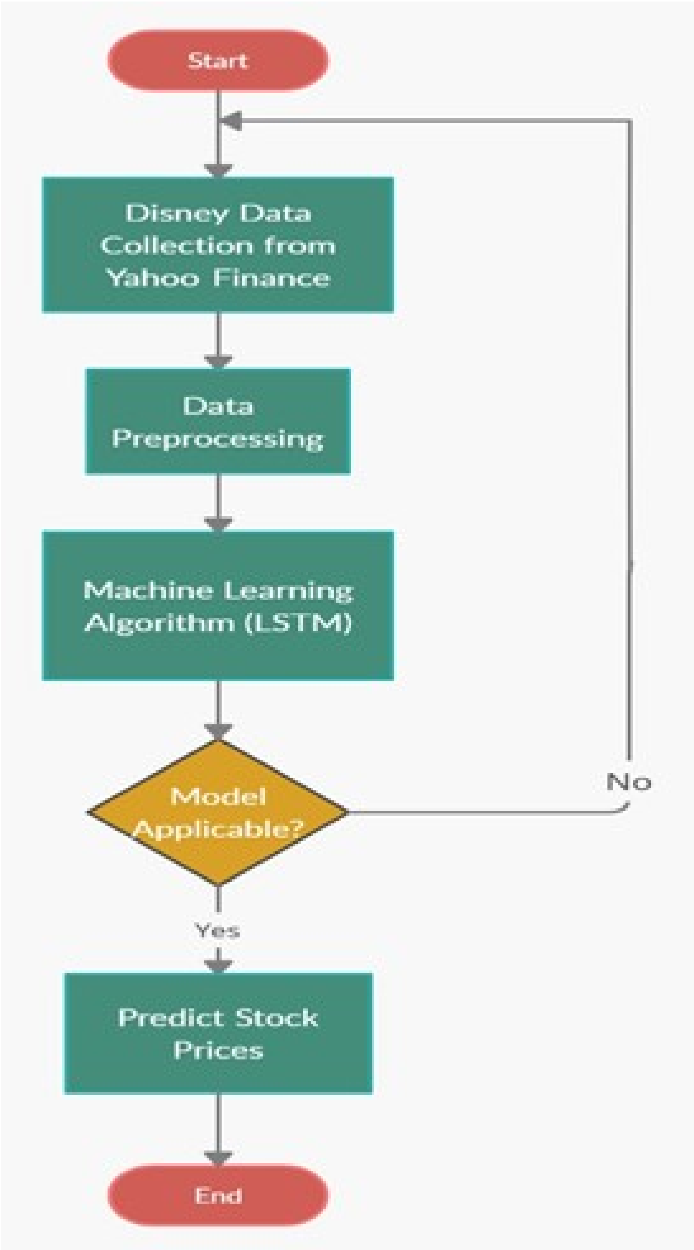

Figure 4.3: System Flow Chart

4.2 Front End development tool

We have used flutter as our front-end developing tool.

4.2.1 Front End User interfaces

A Sign in page is the first thing a person sees when they open an application. If a user does not already have an account, he should create one. For that he must go for Sign up.

To verify credentials, the system utilizes Firebase Authenticator. Hence it is not possible for a user to enter an incorrect email address; otherwise, an error will occur. It will also verify if the entered password is of normal standards or not. Weather sign in or sign up the application allows a user to login.

After Login application proceeds and it take you to the next page. On the next page a

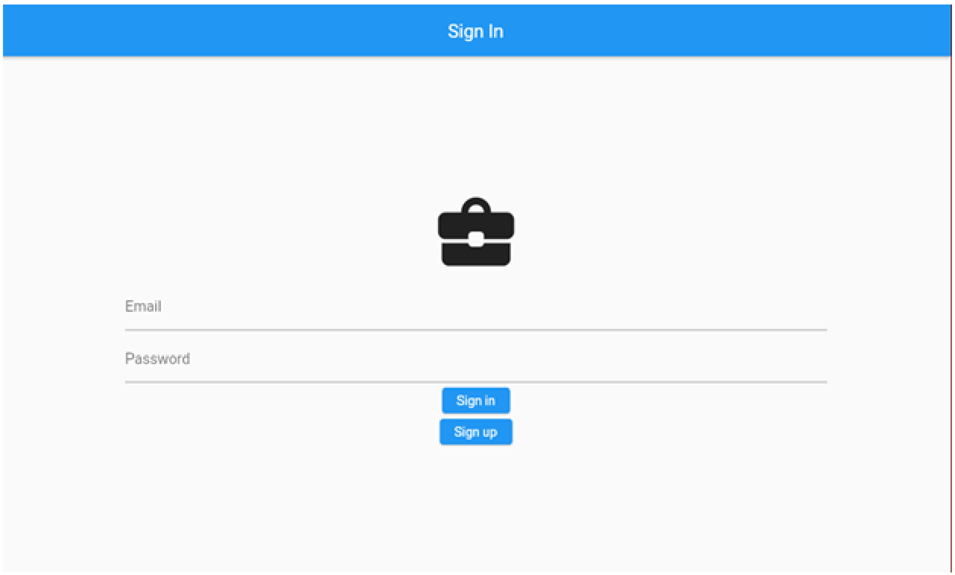

Figure 4.4: Sign In page

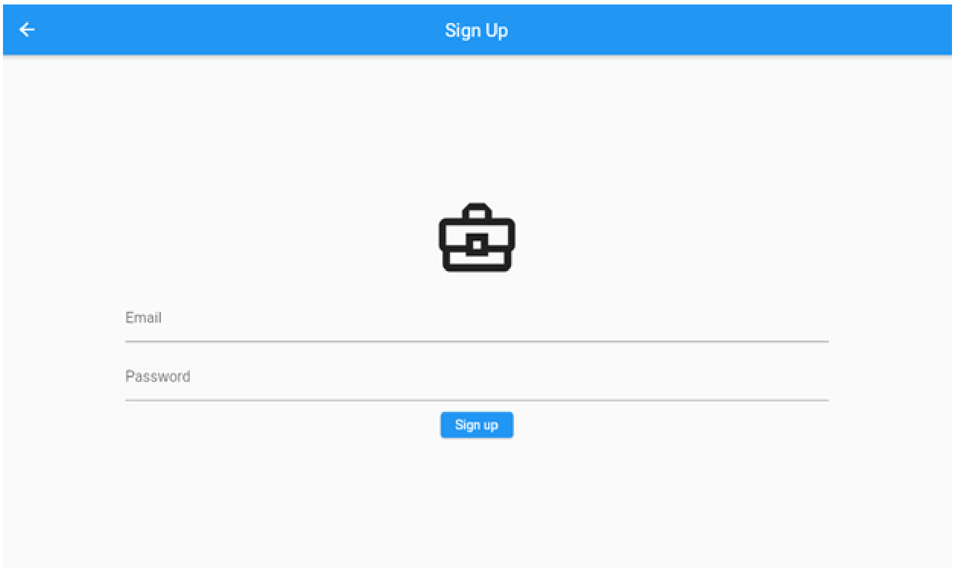

Figure 4.5: Sign up page

user enters company's name. Against that company name an API request is generated and it works on backend. It uses data set in the LSTM model and thus it gives output in image form. API gives response to the application in the image form of Base64 string format. The application then converts the string and displays the results.

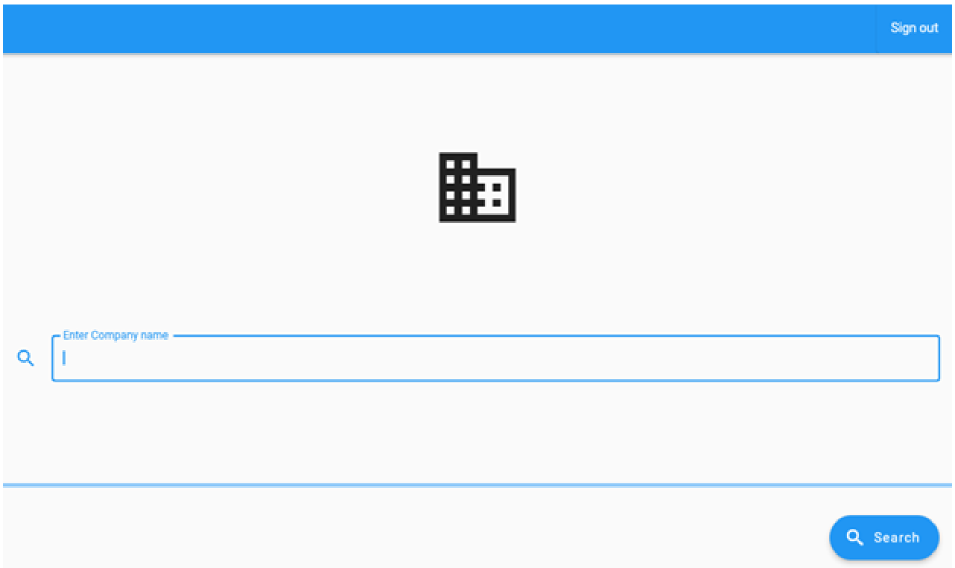

Figure 4.6: Search page

Figure 4.7: Resultant Page

Chapter 5

System Implementation

In this chapter Model Implementation starts from experimental setup to its evaluation on test set is discussed. Following is the detail.

5.1 Environment and Tools

The Python programming language is utilized to complete this project. Google Colab Environment was used to prepare and test the specified Model. It is like the Jupyter notebook inside the cloud environment, and it controls all setup configurations. Colab is a free-to-use cloud-based notebook environment. It offers capabilities that allow you to modify documents in the same manner as Google Docs allows you to. Colab provides support for many popular and high-level machine learning libraries. and can be loaded into your notebook rapidly.

5.1.1 Optimizer

Adam Optimizer is used with default initial learning rate and weight decay. Adam (lr=0.001, decay=0.9).

5.1.2 Loss Function

Mean Squared Error (MSE) is used as loss function in all Models. Below formula that is used to calculate Loss Function MSE which explains the mean square difference between true value and Predicted value divided by total number of values.

$$L(y, \hat{y}) = \frac{1}{N} \sum_{i=0}^{N} (y - \widehat{y_i})^2$$

where,

y= true value
$\widehat{y} = Predictedvalue$
N= total number of data points

5.1.3 Data Set Overview

A data set was required for the implementation of this project. A data set from Disney Land, USA was used for this purpose. Yahoo Finance provided a CSV file for download. Yahoo

Finance is a unit of the Yahoo network of media properties. It offers stock quotations, press announcements, financial reports, and original material, as well as financial news, data, and opinion.
We wanted to transform an array of values into a dataset matrix after receiving the csv file. This was since arrays are incapable of making predictions, hence it was necessary in the matrix.

5.1.4 Dataset Overview

A data set was required for the implementation of this project. A data set from Disney Land, USA was used for this purpose. Yahoo Finance provided a CSV file for download. Yahoo Finance is a unit of the Yahoo network of media properties. It offers stock quotations, press announcements, financial reports, and original material, as well as financial news, data, and opinion.
We wanted to transform an array of values into a dataset matrix after receiving the csv file. This was since arrays are incapable of making predictions, hence it was necessary in the matrix.

To ensure that the findings are repeatable, a random seed is applied. In other words,

```
input_file="/content/DIS.csv"

# convert an array of values into a dataset matrix
def create_dataset(dataset, look_back=1):
  dataX, dataY = [], []
  for i in range(len(dataset)-look_back-1):
    a = dataset[i:(i+look_back), 0]
    dataX.append(a)
    dataY.append(dataset[i + look_back, 0])
  return np.array(dataX), np.array(dataY)
```

Figure 5.1: create dataset

utilizing this argument ensures that anyone running your code again would obtain the same results. Following that, the dataset is loaded.

In the network, closed values (column [5]) are used because there are only 5 columns

```
# fix random seed for reproducibility
np.random.seed(5)

# load the dataset
df = read_csv(input_file, header=None, index_col=None, error_bad_lines=False)
```

Figure 5.2: read dataset

in dataset. The range of values is normalized (0,1). Normalization is a data preparation method used often in machine learning. This is a technique for translating numerical column numbers into databases in the same scale without distorting the range of values obliterating information. For this algorithm to correctly model the data, it must be normalized.
Then the data is divided into training and testing sets, with 50% each. The implementation

is done with Keras-Tensorflow.

The LSTM network has 1 output layer (1 dense layer) and 25 hidden neurons. LSTM

```
# take close price column[5]
all_y = df[5].values
dataset=all_y.reshape(-1, 1)

# normalize the dataset
scaler = MinMaxScaler(feature_range=(0, 1))
dataset = scaler.fit_transform(dataset)

# split into train and test sets, 50% test data, 50% training data
train_size = int(len(dataset) * 0.5)
test_size = len(dataset) - train_size
train, test = dataset[0:train_size,:], dataset[train_size:len(dataset),:]

# reshape into X=t and Y=t+1, timestep 240
look_back = 240
trainX, trainY = create_dataset(train, look_back)
testX, testY = create_dataset(test, look_back)

# reshape input to be [samples, time steps, features]
trainX = np.reshape(trainX, (trainX.shape[0], 1, trainX.shape[1]))
testX = np.reshape(testX, (testX.shape[0], 1, testX.shape[1]))
```

Figure 5.3: Train model

network includes hidden: 25 neurons, 1 input layer, 1 output layer, dropout is 0.1, optimizer is Adam, batch size is 240, timestep are 240 and 1000 epochs.

- Input layer is made of artificial input neurons and brings the primary data into the system for artificial neurons to process in subsequent layers.‘
- Output layer is the last layer of neurons in an artificial neural network generates the program’s outputs.
- Hidden Layers’ purpose is to extract features from the input data and utilize them to link a given input with the proper output.
- Optimizers are procedures used for decreasing an error function (loss function) or improving production efficiency. They are math functions that are based on the learnable factors of a model, like Biases and Weights. Optimizers helps to minimize losses by deciding how to change the weights of neural network.
- Dropout is a method for stop overfitting in a model. In each training phase of the training phase, Dropout works by setting the output edges of the hidden units into zero.
- Time Steps are time ticks. It refers to the length of each of your samples in terms of time.

- The number of samples in a batch is known as batch size. They must be processed before the model can be updated.

- Epoch is a unit of time required to train a neural network for a single cycle utilising all of the training data.

We specify 0.1 for the Dropout layers, which means that 10% of the layers will be reduced. After that, we add the Dense layer. After that, we use the popular Adam optimizer to assemble our model and set the loss to mean squared error.
The following is training weights using 25 epochs. And after that the model is saved to

```
# create and fit the LSTM network, optimizer=adam, 25 neurons, dropout 0.1
model = Sequential()
model.add(LSTM(25, input_shape=(1, look_back)))
model.add(Dropout(0.1))
model.add(Dense(1))
```

Figure 5.4: create LSTM network

file.
Following that, the model will be loaded and utilized to create predictions. Predicted and

```
model.summary()
plot_model(model, to_file='model_plot.png', show_shapes=True, show_layer_names=True)

model.compile(loss='mse', optimizer='adam')
model.fit(trainX, trainY, epochs=25, batch_size=32, verbose=1)

model.save('LSTM_model')
```

```
=================================================================
Total params: 26,626
Trainable params: 26,626
Non-trainable params: 0
_________________________________________________________________
Epoch 1/25
72/72 [==============================] - 2s 2ms/step - loss: 9.2925e-04
Epoch 2/25
72/72 [==============================] - 0s 3ms/step - loss: 2.7971e-04
Epoch 3/25
72/72 [==============================] - 0s 3ms/step - loss: 2.1359e-04
Epoch 4/25
72/72 [==============================] - 0s 2ms/step - loss: 1.8492e-04
Epoch 5/25
72/72 [==============================] - 0s 3ms/step - loss: 1.6964e-04
Epoch 6/25
72/72 [==============================] - 0s 2ms/step - loss: 1.4381e-04
Epoch 7/25
```

Figure 5.5: Summary of Model

actual values will be included in the output results.
The plot file will have both real and predicted data.

```
modelLoad = keras.models.load_model('LSTM_model')

# make predictions
trainPredict = modelLoad.predict(trainX)
testPredict = modelLoad.predict(testX)
```

```
# invert predictions
trainPredict = scaler.inverse_transform(trainPredict)
trainY = scaler.inverse_transform([trainY])
testPredict = scaler.inverse_transform(testPredict)
testY = scaler.inverse_transform([testY])

# calculate root mean squared error
trainScore = math.sqrt(mean_squared_error(trainY[0], trainPredict[:,0]))
print('Train Score: %.2f RMSE' % (trainScore))
testScore = math.sqrt(mean_squared_error(testY[0], testPredict[:,0]))
print('Test Score: %.2f RMSE' % (testScore))

# shift train predictions for plotting
trainPredictPlot = np.empty_like(dataset)
trainPredictPlot[:, :] = np.nan
trainPredictPlot[look_back:len(trainPredict)+look_back, :] = trainPredict
```

```
# shift test predictions for plotting
testPredictPlot = np.empty_like(dataset)
testPredictPlot[:, :] = np.nan
testPredictPlot[len(trainPredict)+(look_back*2)+1:len(dataset)-1, :] = testPredict

# plot baseline and predictions
plt.plot(scaler.inverse_transform(dataset))
plt.plot(trainPredictPlot)
print('testPrices:')
testPrices=scaler.inverse_transform(dataset[test_size+look_back:])

print('testPredictions:')
print(testPredict)

# export prediction and actual prices
df = pd.DataFrame(data={"prediction": np.around(list(testPredict.reshape(-1)), decimals=2), "test_price": np.around(list(testPrices.reshape(-1)), decimals=2)})
df.to_csv("lstm_result.csv", sep=';', index=None)

# plot the actual price, prediction in test data=red line, actual price=blue line
plt.plot(testPredictPlot)
plt.show()
```

Figure 5.6: Prediction

Chapter 6

System Performance Evaluation

6.1 LSTM Model

The LSTM model which is implemented in this project is as following.

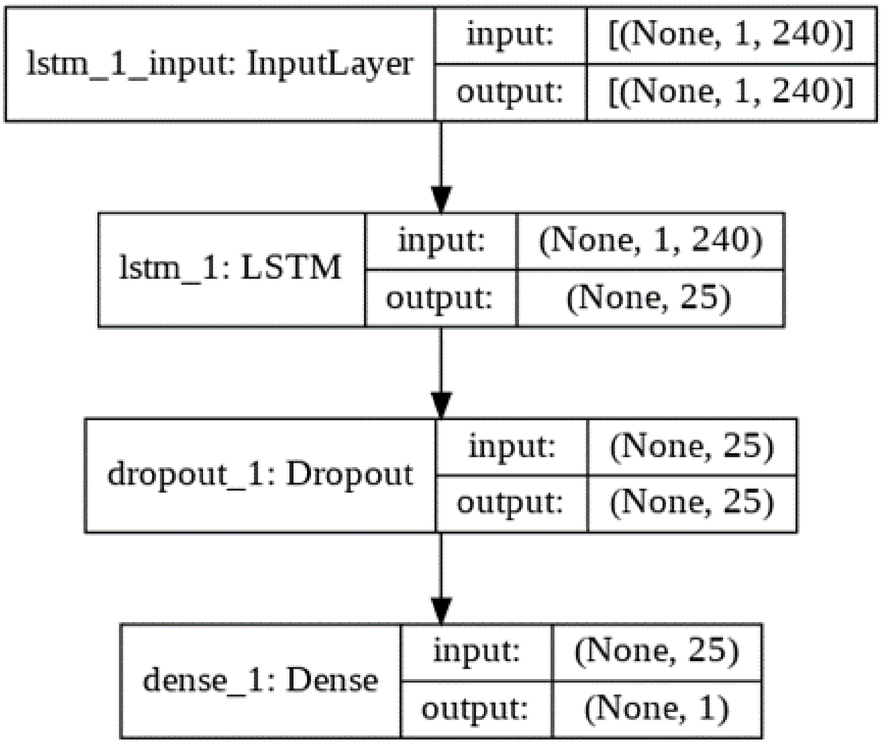

Figure 6.1: LSTM Model

6.2 Model Scores

Following are the test results secured during training and testing of the model

TRAIN SCORE	TEST SCORE
0.93 RMSE	12.41 RMSE

Test Prices:
Test Predictions:
[[28.589241]
[28.697668]
[28.931154]
...
[78.23565]
[77.91422]
[78.16686]]
Accuracy: Almost 80% Accuracy was obtained during training and testing process. In

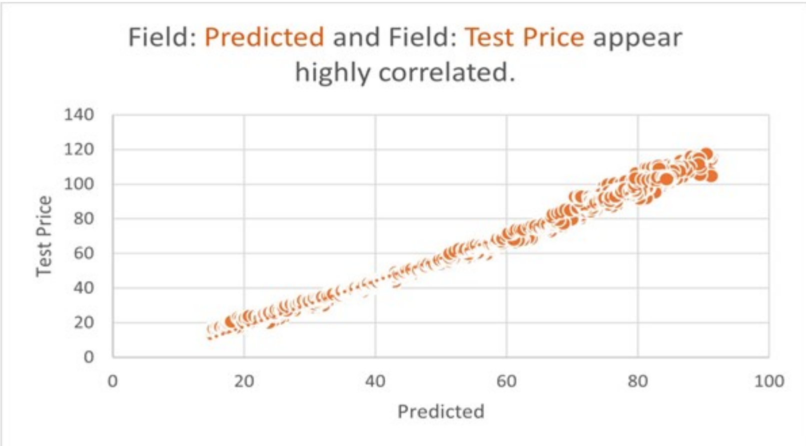

Figure 6.2: Comparison Between Predicted and Test Price

this graph red line represents prediction in test data whereas blue line represents actual price.

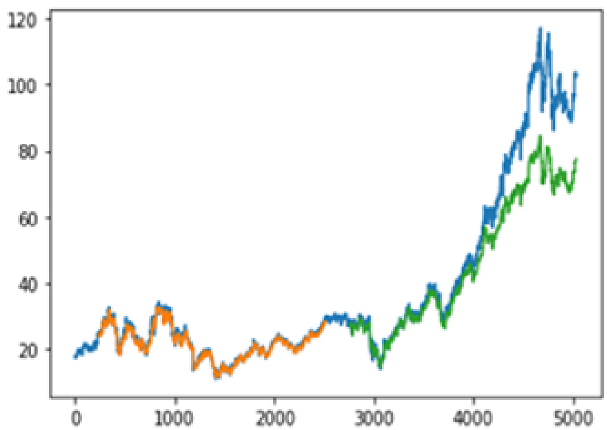

Figure 6.3: Graph Plot

6.3 Evaluation Measure

For model evaluation, RMSE has been calculated for all models with same parameters so that we can select the best model under same environment

1. **Epochs** are 1000
2. **Batch size** is 32
3. **No. of Neurons** are 25
4. **Verbose** is 1

6.4 Results in File

The predicted results are stored in a csv file. The name of the file is “lstm$_{r}esult.csv$”

1 to 10 of 2277 entries Filter

prediction;test_price
28.59;28.78
28.7;28.81
28.93;28.14
28.97;28.2
28.74;28.74
28.59;29.0
28.7;28.63
28.84;28.29
28.98;28.28
28.96;28.16

Show 10 per page 1 2 10 100 200 220 228

Figure 6.4: Resultant CSV file

Chapter 7

Conclusion

Predicting a stock market is very difficult due to its highly volatile behaviour. In our project, we created an artificial intelligence-based stock exchange forecasting architecture using LSTM for time series forecasting. To do so, we used a dataset from Yahoo Finance to evaluate our model and to extract features and hidden patterns to improve the LSTM's design. Regardless the limitations of our exploration, for example a small dataset and the sheer unpredictability of the stock market, we tested and validated that our model's forecasting estimates are fairly accurate and near to the actual values. Despite the minimal data we had, we were able to train the LSTM to rather decent results. We employed Flutter using API to construct our front end such that the user could forecast value. The prediction will become even more accurate if the model is trained with a bigger number of data sets.

References

[1] Bahram Armoon, GBD 2019 Viewpoint Collaborators, GBD 2019 Viewpoint Collaborators, et al. Five insights from the global burden of disease study 2019. *The Lancet*, 396, 2020.

[2] Michel Ballings, Dirk Van den Poel, Nathalie Hespeels, and Ruben Gryp. Evaluating multiple classifiers for stock price direction prediction. *Expert systems with Applications*, 42(20):7046–7056, 2015.

[3] Deeksha Chandola, Harsh Gupta, Vinay Anand Tikkiwal, and Manoj Kumar Bohra. Multi-step ahead forecasting of global solar radiation for arid zones using deep learning. *Procedia Computer Science*, 167:626–635, 2020.

[4] Bruno Miranda Henrique, Vinicius Amorim Sobreiro, and Herbert Kimura. Stock price prediction using support vector regression on daily and up to the minute prices. *The Journal of finance and data science*, 4(3):183–201, 2018.

[5] Wasiat Khan, Mustansar Ali Ghazanfar, Muhammad Awais Azam, Amin Karami, Khaled H Alyoubi, and Ahmed S Alfakeeh. Stock market prediction using machine learning classifiers and social media, news. *Journal of Ambient Intelligence and Humanized Computing*, pages 1–24, 2020.

[6] Sai Krishna Lakshminarayanan and John P McCrae. A comparative study of svm and lstm deep learning algorithms for stock market prediction. In *AICS*, pages 446–457, 2019.

[7] Dazhou Li, Chuan Lin, Wei Gao, Guangbao Yu, Jian Gao, and Wenbo Xia. Short-term iot data forecast of urban public bicycle based on the dbscan-tcn model for social governance. *Wireless Communications and Mobile Computing*, 2021, 2021.

[8] Ishita Parmar, Navanshu Agarwal, Sheirsh Saxena, Ridam Arora, Shikhin Gupta, Himanshu Dhiman, and Lokesh Chouhan. Stock market prediction using machine learning. In *2018 first international conference on secure cyber computing and communication (ICSCCC)*, pages 574–576. IEEE, 2018.

[9] Murtaza Roondiwala, Harshal Patel, and Shraddha Varma. Predicting stock prices using lstm. *International Journal of Science and Research (IJSR)*, 6(4):1754–1756, 2017.

[10] RP Schumaker and Hsinchun Chen. Textual analysis of event prediction using breaking financial news. *ACM Transactions on Information Systems*, 27:1–19, 2009.

[11] Jingyi Shen and M Omair Shafiq. Short-term stock market price trend prediction using a comprehensive deep learning system. *Journal of big Data*, 7(1):1–33, 2020.

[12] Tao Xing, Yuan Sun, Qian Wang, and Guo Yu. The analysis and prediction of stock price. In *2013 IEEE International Conference on Granular Computing (GrC)*, pages 368–373. IEEE, 2013.