STOCK MARKET VALUE PREDICTION BASED ON MACHINE LEARNING APPROACH

CHEW MIN WEI

Bachelor of Computer Science (Software Engineering) with Honors

UNIVERSITI MALAYSIA PAHANG

UNIVERSITI MALAYSIA PAHANG

DECLARATION OF THESIS AND COPYRIGHT

Author's Full Name : CHEW MIN WEI

Date of Birth

Title : STOCK MARKET VALUE PREDICTION BASED ON MACHINE LEARNING APPROACH

Academic Session : SEMESTER I 2022/2023

I declare that this thesis is classified as:

☐	CONFIDENTIAL	(Contains confidential information under the Official Secret Act 1997)*
☐	RESTRICTED	(Contains restricted information as specified by the organization where research was done)*
☑	OPEN ACCESS	I agree that my thesis to be published as online open access (Full Text)

I acknowledge that Universiti Malaysia Pahang reserves the following rights:

1. The Thesis is the Property of Universiti Malaysia Pahang
2. The Library of Universiti Malaysia Pahang has the right to make copies of the thesis for the purpose of research only.
3. The Library has the right to make copies of the thesis for academic exchange.

Certified by:

(Student's Signature)

(Supervisor's Signature)

TS. DR. KOHBALAN A/L MOORTHY
SENIOR LECTURER
FACULTY OF COMPUTING
COLLEGE OF COMPUTING & APPLIED SCIENCES
UNIVERSITI MALAYSIA PAHANG
26600 PEKAN, PAHANG DARUL MAKMUR
TEL : 09-424 4681 FAX : 09-424 4606

New IC/Passport Number Date: 10 February 2023

Name of Supervisor
Date: 16/02/2023

NOTE : * If the thesis is CONFIDENTIAL or RESTRICTED, please attach a thesis declaration letter.

THESIS DECLARATION LETTER (OPTIONAL)

Librarian,
Perpustakaan Universiti Malaysia Pahang,
Universiti Malaysia Pahang,
Lebuhraya Tun Razak,
26300, Gambang, Kuantan.

Dear Sir,

CLASSIFICATION OF THESIS AS RESTRICTED

Please be informed that the following thesis is classified as RESTRICTED for a period of three (3) years from the date of this letter. The reasons for this classification are as listed below.

Author's Name
Thesis Title

Reasons (i)

(ii)

(iii)

Thank you.

Yours faithfully,

____
(Supervisor's Signature)

Date:

Stamp:

Note: This letter should be written by the supervisor, addressed to the Librarian, *Perpustakaan Universiti Malaysia Pahang* with its copy attached to the thesis.

SUPERVISOR'S DECLARATION

I/We* hereby declare that I/We* have checked this thesis/project* and in my/our* opinion, this thesis/project* is adequate in terms of scope and quality for the award of the degree of *Doctor of Philosophy/ Master of Engineering/ Master of Science in …………………………..

____

(Supervisor's Signature)

Full Name : TS. DR. KOHBALAN A/L MOORTHY
SENIOR LECTURER
FACULTY OF COMPUTING
COLLEGE OF COMPUTING & APPLIED SCIENCES
UNIVERSITI MALAYSIA PAHANG
26600 PEKAN, PAHANG DARUL MAKMUR
TEL : 09-424 4681 FAX : 09-424 4666

Position :

Date : 16/02/2023

____

(Co-supervisor's Signature)

Full Name :

Position :

Date :

STUDENT'S DECLARATION

I hereby declare that the work in this thesis is based on my original work except for quotations and citations which have been duly acknowledged. I also declare that it has not been previously or concurrently submitted for any other degree at Universiti Malaysia Pahang or any other institutions.

CHEW

(Student's Signature)

Full Name : CHEW MIN WEI

ID Number : CB19049

Date : 10 February 2022

STOCK MARKET VALUE PREDICTION BASED ON
MACHINE LEARNING APPROACH

CHEW MIN WEI

Thesis submitted in fulfillment of the requirements
for the award of the degree of
Bachelor of Computer Science (Software Engineering) with Honors

Faculty of Computing

UNIVERSITI MALAYSIA PAHANG

FEBRUARY 2023

ACKNOWLEDGEMENTS

First of all, I cannot express enough thanks to my supervisor, Dr. Kohbalan A/L Moorthy and Final Year Project coordinator, Dr. Danakorn Nincarean A/L Eh Phon for all the guidance, patience, encouragement and assistance in completing this research project. Thanks to my parent, Mr Chew Chong Ee and Mrs Nai Siew Heok for their encouragement along the way. Last but not least, I would like to thank all my friends that motivates and assist me throughout the research.

ABSTRAK

Ramalan harga pasaran saham telah menjadi bidang yang menarik kepada ramai penyelidik terutamanya dalam bidang Pembelajaran Mesin (ML) dan analisis siri masa. Dengan model ramalan yang baik, pelabur boleh memaksimumkan pulangan dengan risiko minimum dalam portfolio. Walau bagaimanapun, tidak semua pelaburan menjamin dengan pulangan. Ini disebabkan oleh cara tradisional ramalan harga saham memerlukan setiap pelabur memperoleh kemahiran teknikal dan pengetahuan kewangan. Oleh itu, cara tradisional untuk meramalkan pasaran saham perlahan-lahan digantikan dengan Artificial Intelligent (AI) untuk meningkatkan ketepatan model ramalan dan mengurangkan keperluan bagi pelabur untuk berjaya dalam pasaran saham. Sebagai manusia, batasan mental dan fizikal mengekang keupayaan untuk melaksanakan tugas dengan sempurna dan berkesan sepanjang hayat. Pasaran saham dipengaruhi oleh banyak aspek dan turun naik akan menyebabkan kebanyakan manusia melakukan keputusan pelaburan yang emosi dan buruk. Oleh itu, peramal harga saham automatik berdasarkan pembelajaran mesin memainkan peranan penting dalam membantu pelabur membuat keputusan pelaburan yang baik. Kajian dan analisis model ramalan sedia ada adalah penting untuk membina dan melaksanakan peramal harga saham berasaskan pembelajaran mesin. Peramal yang dibina perlu menghasilkan graf visual untuk tujuan pengesahan dan pengesahan bagi memastikan kecekapan dan ketepatan model supaya mudah diikuti oleh pelabur walaupun pemula. Peramal harga saham berasaskan pembelajaran mesin akan menyediakan garis panduan yang membolehkan pelabur membuat keputusan pelaburan yang rasional dengan risiko kerugian yang agak rendah.

ABSTRACT

Stock market price prediction has been an attractive area to many researchers particularly in the field of Machine Learning (ML) and time series analysis. With a good prediction model, investors can maximize returns with minimal risks in portfolio. However, not every investment guarantees with returns. This is due to the traditional ways of stock price prediction requires every investor to acquire technical skills and financial knowledge. Hence, traditional way of predicting the stock market is slowly replaced by Artificial Intelligent (AI) to improve the accuracy of prediction model and lower the requirement for investors to be success in the stock market. As a human, mental and physical limitations constraint the ability to perform task perfectly and effectively in the entire life. Stock market is influenced by many aspects and the fluctuations will cause most of the human to do emotional and bad investment decision. Hence, an automated stock price predictor based on machine learning plays an important role in helping investors to make a good investment decision. Study and analysis the existing prediction model is crucial to build and implement a machine learning based stock price predictor. The predictor built has to produce a visual graph for validation and verification purpose to make sure the efficiency and accuracy of model so that it is easy for investors even beginners to follow. A machine learning based stock price predictor will provide guidelines that enable investors to make rational investment decision with a relatively low risk of losing money.

TABLE OF CONTENT

DECLARATION		
TITLE PAGE		
ACKNOWLEDGEMENTS		**ii**
ABSTRAK		**iii**
ABSTRACT		**iv**
TABLE OF CONTENT		**v**
LIST OF TABLES		**viii**
LIST OF FIGURES		**ix**
LIST OF SYMBOLS		**x**
LIST OF ABBREVIATIONS		**xi**
CHAPTER 1 INTRODUCTION		**1**
1.1	Introduction	1
1.2	Problem Background	2
1.3	Problem Statements	3
1.4	Goal & Objectives	3
1.5	Research Scope	3
1.6	Significance of Project	4
1.7	Summary	4
CHAPTER 2 LITERATURE REVIEW		**5**
2.1	Overview	5
2.2	Classical Approaches for Stock Market Prediction	5

2.2.1 Fundamental Analysis 5

2.2.2 Technical Analysis 6

2.3 Machine Learning Approaches for Stock Market Prediction 7

2.3.1 Support Vector Machines (SVMs) 7

2.3.2 Random Forest (RF) 9

2.3.3 Artificial Neural Network (ANN) 9

2.4 Existing Work Related to Stock Market Value Prediction 11

2.4.1 Support Vector Machine (SVM) 11

2.4.2 Random Forest (RF) 13

2.4.3 Artificial Neural Network (ANN) 17

2.5 Comparative Analysis 19

2.6 Summary 20

CHAPTER 3 METHODOLOGY 21

3.1 Overview 21

3.2 Research Framework 21

3.3 Long Short Term Memory (LSTM) 22

3.4 Dataset 26

3.5 Performance Measurement 27

3.6 Hardware & Software Requirements 27

3.7 Summary 28

CHAPTER 4 RESULTS AND DISCUSSIONS 29

4.1 Introduction 29

4.2 Proposed Technique 29

4.3 Experimental Setup for LSTM Architecture 30

4.3.1 Acquisition of Stock Data 30

4.3.2 Visualizing Stock Prices History 31

4.3.3 Data Pre-processing and Preparation of training set 32

4.3.4 Preparation of test set 33

4.3.5 Setting Up LSTM Network Architecture 34

4.3.6 Training the LSTM Model 34

4.3.7 Model Evaluation 35

4.4 Result of Experimental Setup and Discussions 35

CHAPTER 5 CONCLUSION **39**

5.1 Introduction 39

5.2 Research Constraints 39

5.3 Future Work 39

REFERENCES **41**

LIST OF TABLES

Table 2.1: Data Analysis of Prediction Results 13

Table 2.2: Performance of 5-days-ahead models 16

Table 2.3: Performance of 10-days-ahead models 17

Table 2.4: Comparative analysis between Support Vector Machines (SVMs), Random Forest (RF) and Artificial Neural Network (ANN) 19

Table 3.1: Hardware Specifications 27

Table 3.2: Software Specifications 27

Table 4.1: Table of Result for LSTM Stock Market Price Prediction Model 37

LIST OF FIGURES

Figure 2.1: Schematic diagram of separated hyperplane. 8
Figure 2.2: The data segmentation effect of RBF kernel function. 8
Figure 2.3: Schematic illustration of Random Forest (RF). 9
Figure 2.4: The structure of three-layer fully connected neural network. 10
Figure 2.5: An illustration of relationship between inputs and output for ANN 11
Figure 2.6: Screenshot of part of the data 12
Figure 2.7: Actual value and predictive value of the prediction 12
Figure 2.8: Visual representation of the model used 15
Figure 2.9: Prediction results for NSE RELIANCE 18
Figure 3.1: Overview Phases in the Research Framework 22
Figure 3.2: LSTM Memory Cell Structure 24
Figure 3.3: Architecture of LSTM model 26
Figure 3.4: Historical Daily Price of Apple (APPL) 26
Figure 4.1: Experiment Setup for 3 different LSTM architecture 30
Figure 4.2: Code for importing libraries and preview data 31
Figure 4.3: Preview of data 31
Figure 4.4: Code for visualizing historical prices movement of AAPL 31
Figure 4.5: Historical Prices of AAPL for past five years 32
Figure 4.6: Code for preparing training set 33
Figure 4.7: Code for preparation of test set 33
Figure 4.8: Code for setting up LSTM network architecture 34
Figure 4.9: Summary of LSTM network architecture 34
Figure 4.10: Code for training LSTM model 35
Figure 4.11 Code for example RMSE result 35
Figure 4.12: Existing Work LSTM Architecture 36
Figure 4.13: Setup 1 LSTM Architecture 36
Figure 4.14: Setup 2 LSTM Architecture 36
Figure 4.15: Stock Market Price Prediction Graph 38

LIST OF SYMBOLS

w	*Normal direction of the plane*
b	*Form of threshold*
X	*Feature Vector*

LIST OF ABBREVIATIONS

LSTM	Long Short Term Memory
OTC	Over The Counter
AI	Artificial Intelligent
FA	Fundamental Analysis
TA	Technical Analysis
SVM	Support Vector Machine
RF	Random Forest
ANN	Artificial Neural Network
MACD	Moving Average Convergence/Divergence (MACD)
MA	Moving Average
RBF	Radio Basis Function
RSI	Relative Strength Index
RNN	Recurrent Neural Network
ML	Machine Learning

CHAPTER 1

INTRODUCTION

1.1 Introduction

Stocks, also known as equities, represent an ownership of a part of a publicly traded company and units of stock are called as “shares”. Corporations issue stock to collect funds to expand their businesses or to undertake new projects. For investors, investing in stocks are one of the ways to grow their money and outpace inflation over time rather than placing it in the bank and lose values over time. Investors can gain profits from shares in two main ways. Some stocks pay dividends to holders typically in quarterly basis or investors can sell stock when the stock price increases from their purchased price to make profits.

The stock market refers to the collection of exchanges and other venues where the buying, selling, and issuance of shares of publicly held companies take place and it is one of the most common types in financial markets. These financial transactions take place either on official exchanges or in over-the-counter (OTC) marketplace that follow a set of defined regulations. All the stocks that are traded at stock exchanges must conform to the government regulations to protect investors from fraudulent practices. Stock market plays an important role in allocating resources to facilitate smooth operation of capitalist economies.

From the beginning of the stock market, predicting the accurate stock price has been the goal for every investor. Every investor or trader is hoping to make profits from their investment in the stock market. People who can make the correct buy and sell decision will end up making profits, but no one can time the market perfectly every time. To make a correct decision, investors have to predict based on the whole bunch of technical analysis data such as company's chart patterns, price trends and different types of indicators. However, investors will find it nearly impossible to analyse and anticipate the market using all available data.

In general, there are two traditional ways known are used for stock price prediction which are fundamental analysis (FA) and technical analysis (TA). Fundamental analysis focuses on company fundamental information to determine the potential future value of respective company. Technical analysis focuses on technical indicators provided and historical data to predict the stock prices in shorter term. Fundamental analysis is more suitable for long term investors while technical analysis is for short term trader. However, people will combine two ways to obtain a more accurate result for their prediction.

Machine learning is a branch of artificial intelligent (AI) which focuses on the use of algorithm and data to imitate the way that humans learn and gradually improve the accuracy of results. By using statistical method, different algorithms are trained to make predictions or classifications. Data-driven decisions are so important to make sure keeping up with competition and will not fall further behind.

Stock market prediction using machine learning algorithm offers a great help to discover the future value of stock. A machine does not require any sleep or rest. It can handle huge amount of data and analyse in a much faster pace compared to human efforts. However, predicting how the stock market will perform to reduce investment risk is a very challenging task to achieve.

1.2 Problem Background

Predicting the market is a challenging task because the future is so unexpected and unclear. No matter how good our analysis is, it is limited to the information that is available at the moment. It's difficult to predict if a stock will rise or fall in value. A mature investment decision will mostly include a take profit and stop-loss strategy, while not always the case. Inexperienced investors believe that their equity positions will continue to rise and that if they are right, they will be able to exit at the top. In fact, it's harsh since a vague plan rarely works out. As a result, whether the investment results in a profit or a loss, all investors must have a plan on how they will enter and exit a trade.

Prediction of stock market mostly rely on technical and fundamental analysis done by human effort. This results in bad investment decision is made even with the help of indicators and data.

1.3 Problem Statements

1. Humans have physical and mental limitations which make us impossible to handle all data required to make predictions. Besides that, humans make mistakes that will result in wrong information given and produce inaccurate predictions.

2. Not everyone expertise in analyzing financial data and different kinds of indicators based on the existing prediction module. It requires a lot of financial knowledge and skills for investors to make a better investment prediction.

3. Stock price is affected by many aspects and human often invest emotionally. Traditional way of prediction only provides potential values for stocks and does not provide an overall picture or trend how the stock will perform for investors to refer. Investors will make emotional decision when the stock perform unusually.

1.4 Goal & Objectives

A machine learning based prediction model is proposed to make automated stock price prediction based on historical data.

1. To study and analyse existing prediction method for stock market prediction.
2. To implement and build stock market predictor based on machine learning algorithm.
3. To validate and verify the stock market predictor based on machine learning algorithm.

1.5 Research Scope

1. The stock market prediction module is only applicable for students who interested in investment for learning purpose only.
2. The stock market prediction module prototype developed does not forecast ahead any stock market prices.

3. The prototype shows the difference between real prices and predicted prices based on algorithm in visual graph.
4. The prototype is developed by using Jupyter notebook with Python language.

1.6 Significance of Project

Inflation rate climbs has been a global problem and it affect every person in the world. Instead of holding cash in hand that will decrease in term of purchasing power, people tend to involve in stock market to make some profits against it. A good stock market prediction module will benefit less experience investors to not include any bad decision due to emotion and have a proper guideline for them to refer.

Besides that, develop a stock market predictor based on machine learning will eliminates human error and increase the efficiency of analysing data. People no longer need to handle whole bunch of data instead a machine can help to solve this problem as they can operate for long hours without rest and increase the accuracy of prediction compared to any other way like technical analysis or fundamental analysis.

The stock market prediction is very important to provide a more convincing guideline for investors to make a good decision on their investment. Young adults like students can learn financial knowledge on how the market will behave with the stock market predictor provided. Generations that rich with financial knowledge will manage their assets better and this leads the society or even country to prosperity.

1.7 Summary

In conclusion, stock market prediction based on machine learning is very helpful for beginner to experts depending how they use the module developed. It can act as a reference when doing investment decision or even for learning purpose for beginners to avoid doing emotional decision. However, the stock market is very unpredictable and affected by many factors which makes the accuracy of predictor relatively low. Hence, it is important to find a best suite machine learning algorithm for stock market prediction to obtain a more accurate result for reference purpose.

CHAPTER 2

LITERATURE REVIEW

2.1 Overview

Chapter 2 focuses on review and discussion on machine learning algorithms that are related to this study. Two traditional approaches in stock prediction will be introduced as well. Three different type machine learning algorithms are applied in stock market value prediction which is Support Vector Machine (SVM), Artificial Neural Network (ANN) and Random Forest (RF) will be discussed and compare their respective advantages and disadvantages.

2.2 Classical Approaches for Stock Market Prediction

In general, there are two different ways to make prediction for stock market which is Fundamental Analysis (FA) and Technical Analysis (TA) by financial professionals in the past. Fundamental analysis depends on a company's fundamental information like market position, expenses and annual growth rates. Technical analysis concentrates on previous stock prices, indicators, historical candlesticks charts and patterns to forecast future values of the stock.

2.2.1 Fundamental Analysis

Fundamental Analysis (FA) is a way of calculating a genuine value of a company by referring to related economic and financial factors. If look in long term, a company will move to a value agreeing place with the forecast. The goal of fundamental analysis is to provide a reference for investors to examine whether the current price is undervalued or overvalued. If a company's stock price is undervalued, then the stock price should rise and vice versa. The analysis is performed by taking metrics such as revenues, earnings, return on equity, profit margins, and other related data into consideration to determine the future potential of the company.

The two commonly used and straightforward metrics to predict long-term value for fundamental analysis are Price to Earnings ratio (P/E) and the Price by Book ratio (P/B). Company with a lower P/E ratio yield higher returns compared to company with a higher P/E ratio. In P/B ratio, the company value specified by the market to the company value specified on paper is compared. The company may be overvalued if the ratio is high and vice versa. Fundamental analysis can be used to determine poor stocks and quality stocks. Although fundamental analysis is a strong technique with several limitations and risks. Experts are required to handle whole bunch of data to react with the price movement which is time and cost consuming. In reality, the behaviour of stock price is affected by many factors.

2.2.2 Technical Analysis

Technical Analysis (TA) is a trading discipline employed to discover trading opportunities by analysing technical indicators for better investment decision. Technical analysis predicts the stock value movement according to the historical data and analysis technical indicators to forecast the stock price. There are few technical indicators that are commonly used:

- Price trends
- Chart patterns
- Volume and momentum indicators
- Moving Average Convergence/Divergence (MACD)
- Support and resistance levels

Technical analysis is built based on Dow's work and experts typically accept three general assumptions for the discipline. The assumptions include the market price discounts everything, prices move in trends, and historic trends usually repeat the same patterns. However, technical analysis is not easy to be used as it involves many indicators that require certain level of understanding to apply it into daily investment decision (A. Vij, K. Saxena and A. Rana, 2021).

2.3 Machine Learning Approaches for Stock Market Prediction

Machine Learning (ML) technique are gaining popularity in the financial field as this approach can improve performance and accuracy of the prediction model. Attempting to create a good model that can properly forecast stock values is a difficult task as there are always unpredictable factors like reputation of company or politics stability of countries will affect the trends of the stock market. Machine learning algorithm can effectively self-learning, recognize and learn patterns from the historical data to help investors with their decisions.

2.3.1 Support Vector Machines (SVMs)

The Support Vector Machine (SVM) is one of the supervised learning approaches and obtain significant result in time series prediction. It can be used to solve classification and regression problems. Support Vector Classification (SVC) is the algorithm that classifies training samples by separating samples in a hyperplane. In two-dimension plane, a line will assign training samples to either side of lines and the divider is known as decision boundary. The following equation depict the decision boundary (S. O. Ojo, P. A. Owolawi, M. Mphahlele and J. A. Adisa, 2019):

$$w.x + b = 0 \tag{2.1}$$

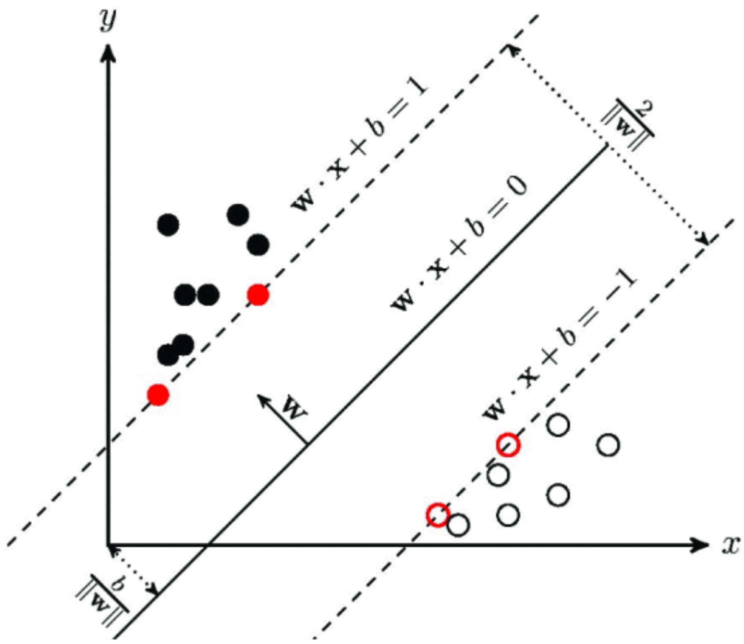

Figure 2.1: Schematic diagram of separated hyperplane.

The basic model of SVM assumes all data points are linearly separable. For linearly non-separable cases, kernel maps the inputs to high-dimensional features place for the conversion to separable classes. The most important part is determining which type of kernel should be used in each model because it has a significant impact on SVM prediction performance (F. L. Marchai, W. Martin and D. Suhartono, 2021). There are many kinds of kernels available such as linear, non-linear and radial basis function (RBF).

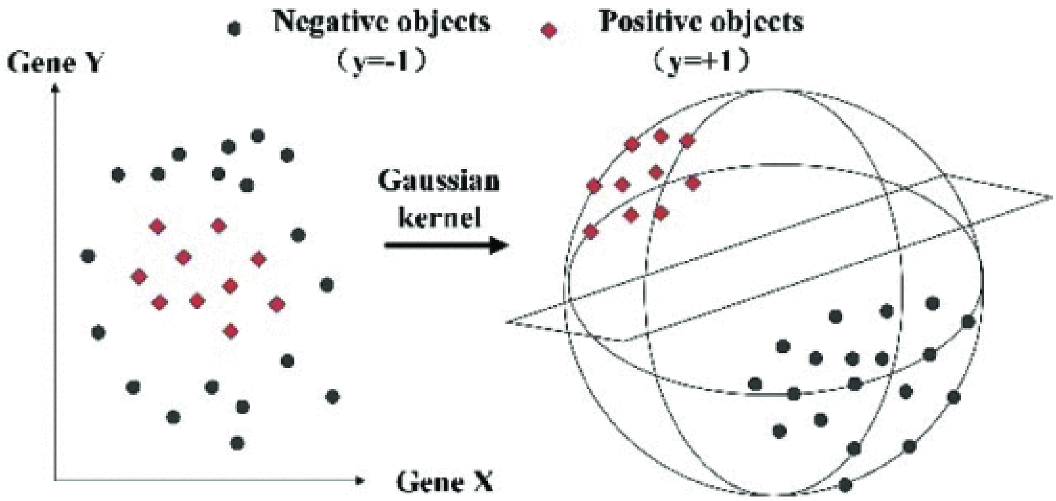

Figure 2.2: The data segmentation effect of RBF kernel function.

The gaussian kernel function is also known as RBF kernel function. The RBF kernel function can reduce the complexity of the calculation process while also increasing the model's efficiency. SVM model has lower risk of overfitting (Surbhi Sharma and Baijnath Kaushik, 2018).

2.3.2 Random Forest (RF)

Random Forest (RF) algorithm is made up of collection of decision tree to improve the accuracy. The model creates a forest by averaging all of the tree forecast results. The main idea behind is that multiple classifiers can increase the correct classification of the test data (M. Nabipour, P. Nayyeri, H. Jabani, S. S. and A. Mosavi, 2020).

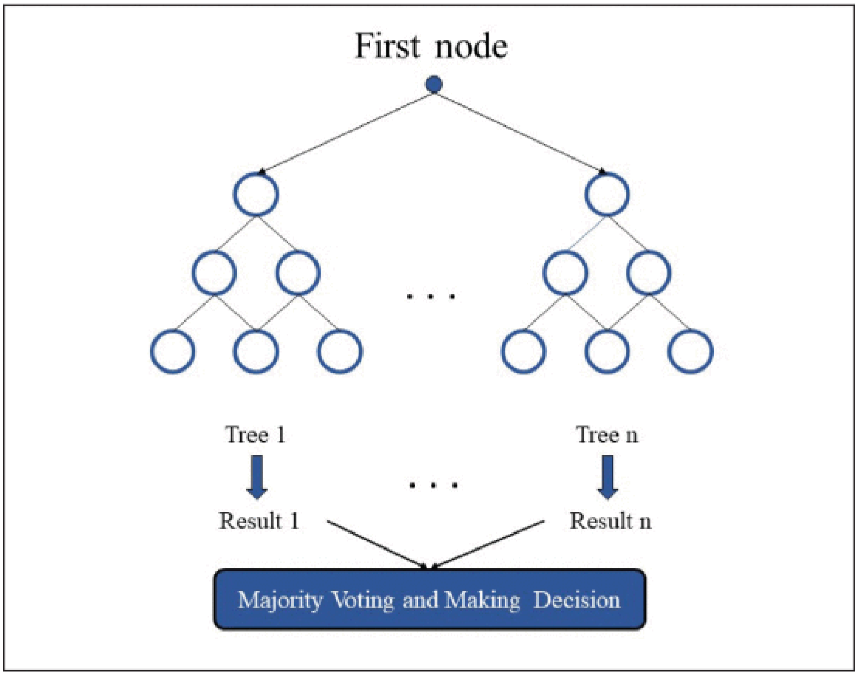

Figure 2.3: Schematic illustration of Random Forest (RF).

2.3.3 Artificial Neural Network (ANN)

Artificial Neural Network (ANN) have been widely applied in the stock price prediction model due to its better performance in completing the constructions of linear models. ANN technique is used with backpropagation in stock price prediction. Forward propagation is done during training phase in neutral network. There are four layers in a fully connected neural network: an output layer, an input layer, and two hidden layers. Each node in a layer is linked to every other node in the following layer.. The input of model uses the features data and the output layer will produce a final prediction result after the calculation of two hidden layers (X. Yuan, J. Yuan, T. Jiang and Q. U. Ain, 2020).

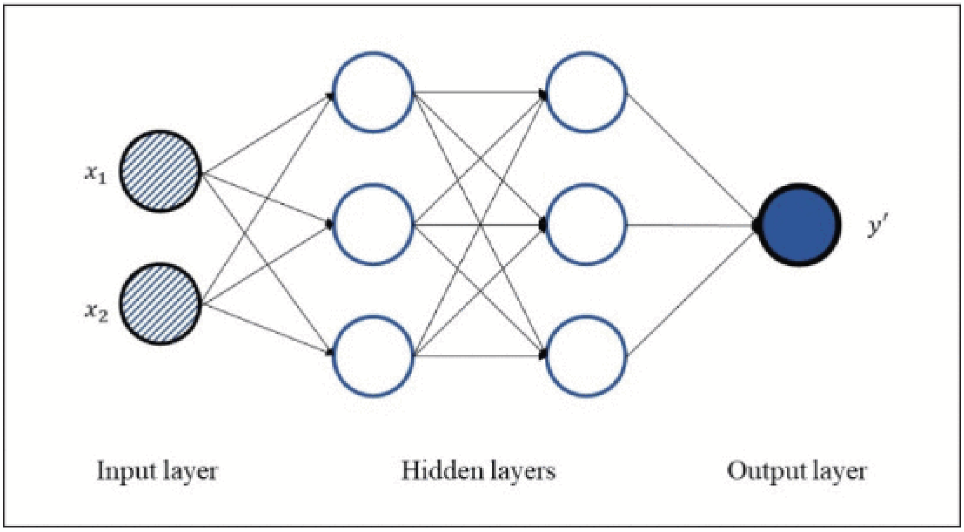

Figure 2.4: The structure of three-layer fully connected neural network.

The weighted sum of the input values result got by a node is added to a bias. A nonlinear function is used to calculate the node's output, which produces new input for the following layer. The final output is computed as a network goes from the input layer to the output layer, this approach is implemented by all nodes in the network.

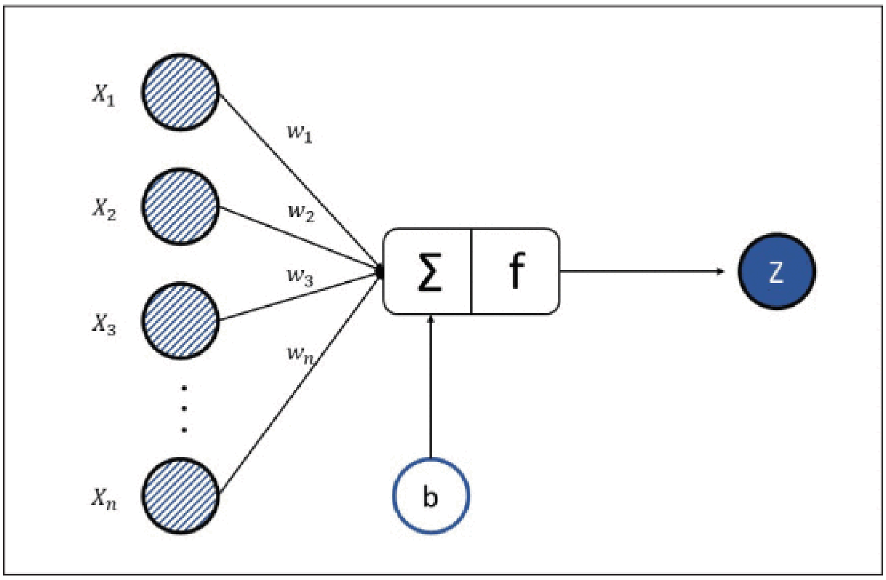

Figure 2.5: An illustration of relationship between inputs and output for ANN

2.4 Existing Work Related to Stock Market Value Prediction

2.4.1 Support Vector Machine (SVM)

Z. Liu, Z. Dang and J. Yu have proposed a model based on the RBF-SVM algorithm to provide more dependable guidance for stock market research and investment decisions, further enhance the stock market value prediction and evaluate the short-term trend of stock price changes.

Support Vector Machine (SVM) is a type of machine learning technique that was developed from Vapnik's fundamental theory of statistical learning. SVM offers a significant advantage in reducing structural hazards, which can significantly lower sample point error. Additionally, SVM can ensure the global optimal results in solving and processing optimization problems.

The RBF kernel function has the advantages of fewer parameters, which can improve the effectiveness of the overall model and minimise the complexity of the computation. In this study, the non-linear model based on SVM is constructed using the RBF kernel function. The expression of the RBF kernel function is:

$$f(x_j, x_i) = exp\left\{-\|x_j - x_i\|^2 / \sigma^2\right\} \tag{2.2}$$

The Yahoo Finance website serves as the main source of experimental data. The data content covers data for several organizations between 2016/01/01 and 2021/10/01.

Date	Open	High	Low	Close	Adj Close	Volume
2016-01-04	25.652500	26.342501	25.500000	26.337500	24.111496	270597600
2016-01-05	26.437500	26.462500	25.602501	25.677500	23.507277	223164000
2016-01-06	25.139999	25.592501	24.967501	25.174999	23.047247	273829600
2016-01-07	24.670000	25.032499	24.107500	24.112499	22.074553	324377600
2016-01-08	24.637501	24.777500	24.190001	24.240000	22.191271	283192000

Figure 2.6: Screenshot of part of the data

Based on the SVM method proposed, a stock market value prediction model is constructed on the collected stock test data. Actual value and predictive value of the prediction are shown in Figure 2.7.

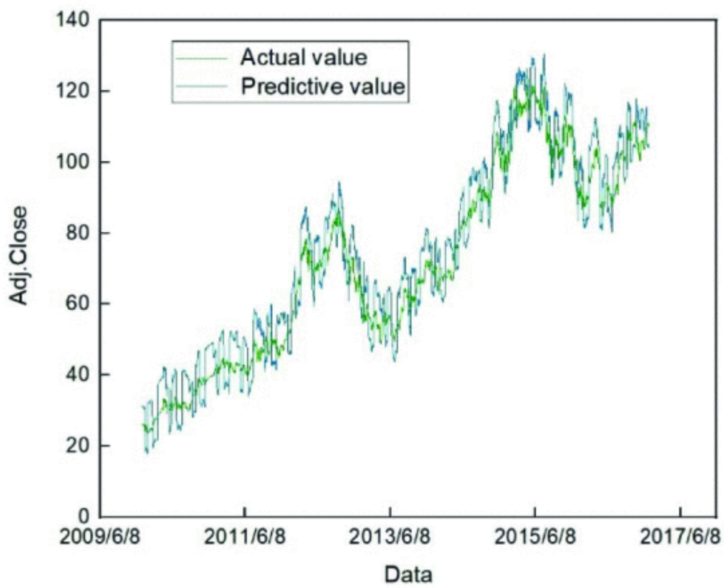

Figure 2.7: Actual value and predictive value of the prediction

Additionally, a comprehensive analysis of the accuracy, standard deviation, and running duration of the model are performed to evaluate the prediction model's accurateness more scientifically. Table 2.1 shows the result.

Table 2.1: Data Analysis of Prediction Results

Algorithm	EVALUATION CRITERIA		
	Mean_Accuracy	*Standard_Deviation*	*Total_Time*
RBF-SVM	65.64%	0.2338	145.9
PCA-SVM	62.17%	0.3645	178.5
GA-SVM	64.67%	0.2648	162.3
DFS-BPSO	61.31%	0.3784	174.6

It is intuitively obvious that the predictive value's fluctuation range in the figure is basically consistent with the real value's fluctuation range. The amount of the errors does not exceed the permissible range, according to the performance evaluation indicators in Table 2.1, even though there are some mistakes at some points on the curve. The SVM-based prediction model proposed in this research has a very accurate prediction impact since the final predicted value is almost identical to the volatility of the real price. The study demonstrates the good accuracy of the SVM-based prediction model and the conformance of the final predictive value with the actual value (Z. Liu, Z. Dang and J. Yu, 2020).

2.4.2 Random Forest (RF)

T. Manojlović and I. Štajduhar have constructed a random forest algorithm based predictive model for stock market trends to predict 5 and 10-days-ahead directions of the CROBEX index and selected stocks. The key concept of ensemble learning methods is sometimes a single classifier is insufficient for correct classification of test data but with multiple classifiers, it will improve model accuracy.

Random forest's basic idea is straightforward. First, creates a set of k bootstrap samples D that are identical samples obtained through sampling with replacement. This implies that each sample, regardless of how many times it has been used, is re-useable. A decision tree is built using n randomly chosen features for every bootstrap sample. Every new instance is classified by k trees to determine its class, and the forecast with the most votes is chosen as the class label.

In this research, classification task is performed by implementing the WEKA (Waikato Environment for Knowledge Analysis) toolkit and used the random forests algorithm. It obtains class label through a combination of simple random trees as simple classifiers for voting. WEKA combines a bagging algorithm and random trees for random forest algorithm. To perform bootstrapping, the bagging technique is employed, and the random tree algorithm is employed to train decision trees with randomly selected features.

Classification accuracy and F-measure were used to evaluate the prediction models in use. Classification accuracy is calculated by dividing the number of correctly classified instances with the overall number of instances. In other words, the following formula is provided to determine the categorization accuracy:

$$\text{Accuracy} = \frac{TP+TN}{TP+TN+FP+FN'} \tag{2.3}$$

where TP represents the quantity of true positives. the number of true positives, TN represents the quantity of true positives. the number of true negatives. FP represents the quantity of true positives.

To calculate F-measure, it is necessary to calculate precision and recall:

$$Precision_{positive} = \frac{TP}{TP+\ pp'} \tag{2.4}$$

$$Recall_{positive} = \frac{TP}{TP+\ FN'} \tag{2.5}$$

$$Precision_{negative} = \frac{TN}{TN + FN'} \tag{2.6}$$

$$Recall_{negative} = \frac{TN}{TN + FP'} \tag{2.7}$$

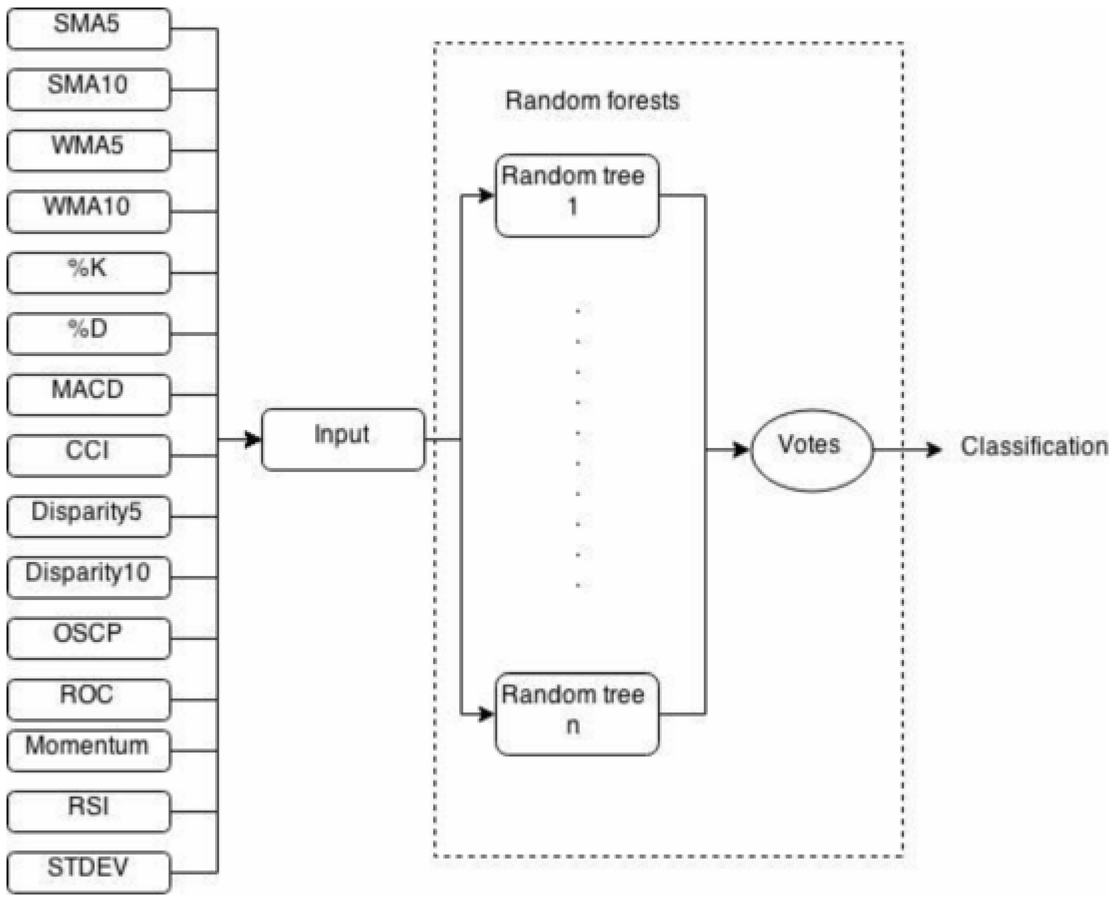

Figure 2.8: Visual representation of the model used

F-measure, also named the F-score, is defined as the harmonic mean of precision and recall:

$$F_{positive} = 2\frac{precision_{positive} * recall_{positive}}{precision_{positive} + recall_{positive}'} \tag{2.8}$$

$$F_{negative} = 2\frac{precision_{negative} * recall_{negative}}{precision_{negative} + recall_{negative}'} \tag{2.9}$$

The weighted average F-measure can be calculated provided with the negative and positive F-measure.

Ten randomly selected subsamples of data with almost equal class distributions are divided from the data. The leftover subsample is then used for validation after using nine out of ten subsamples to train the model. This process is carried out ten times, with each validation using a different partition. The average of the outputs from each model is then used to calculate the result. Table 2.2 displays the results of 5-day direction predictions, whereas Table 2.3 displays the results of 10-day direction predictions.

Table 2.2: Performance of 5-days-ahead models

Symbol	***Accuracy***	$F_{positive}$	$F_{negative}$	***Weighted average F-score***
CROBEX	0.767	0.726	0.797	0.765
HT-R-A	0.772	0.739	0.798	0.771
PBZ-R-A	0.761	0.711	0.797	0.76
KRAS-R-A	0.76	0.721	0.789	0.759

Table 2.3: Performance of 10-days-ahead models

Symbol	*Accuracy*	$F_{positive}$	$F_{negative}$	*Weighted average F-score*
CROBEX	0.816	0.79	0.836	0.815
HT-R-A	0.825	0.804	0.842	0.825
PBZ-R-A	0.81	0.767	0.84	0.808
KRAS-R-A	0.787	0.758	0.81	0.787

According to experimental findings, random forests have a great degree of performance accuracy when predicting stock market trends. The 5-days-ahead models' average accuracy is 0.765 (76.5%) while the other one is 0.808 (80.8%). Weighted average F-measure of 0.763 and 0.808 respectively for both models. The results indicate that random forests can be used to create predictive models that successfully forecast the direction of stock market trends (T. Manojlović and I. Štajduhar, 2015).

2.4.3 Artificial Neural Network (ANN)

MrugaGurjar utilizes the historical stock data to train the ANN. Features like moving average (MA) and stochastic oscillator are extracted and the dataset is then separated into training and testing set.

Backpropagation is employed with the ANN approach for stock price forecasts. Forward propagation occurs in a neutral network during the training phase. The output layer nodes generate the output value following the forward pass. The entire input to the node is first determined during the forward pass, and the activation function is then used to determine the node's output.

The neurons in a feed-forward ANN receive various inputs, and the total input of the neurons is determined using the following formula:

$$TotalInput = n_1 * w_1 + n_2 * w_2 + \dots + n_m * w_m + 1 * w_b \qquad (2.2)$$

Where:

$n_1, n_2, \dots, n_m$ = Input neurons

$w_1, w_2, \dots, w_n$ = Weights corresponding to input neurons

w_b = Weight corresponding to bias

Output of neuron is calculated using activation function:

$$Activationfucntion = 1/\left(1 + e^{(-TotalInput)}\right) \qquad (2.1)$$

Where:

Total Input = The total input to the neuron

A typical technique for training artificial neural networks is the back propagation of errors and used in conjunction with an optimization method such as gradient descent. Link weights are adjusted during the back-propagation phase by comparing the forward pass output to the expected output. The predicted stock prices are giv en for the next day, next 3 days, next 5 days and the predictive result is shown as a graphical form (Gurjar, M., Naik, P., Mujumdar, G., & Vaidya, T, 2018).

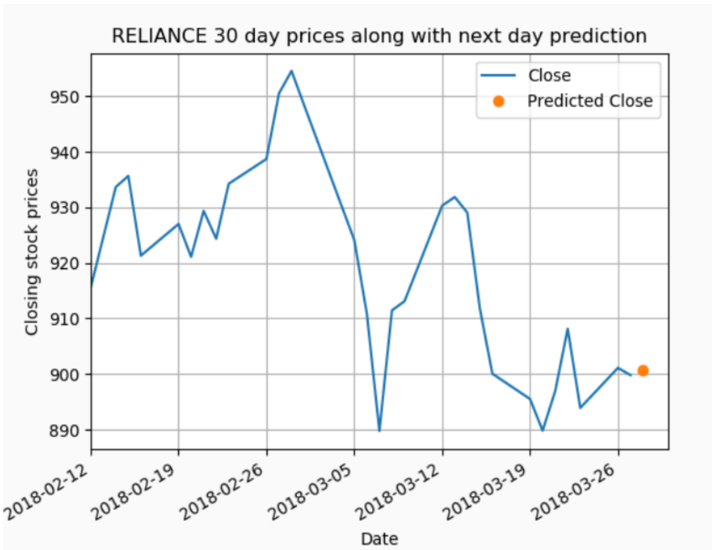

Figure 2.9: Prediction results for NSE RELIANCE

2.5 Comparative Analysis

Table 2.4: Comparative analysis between Support Vector Machines (SVMs), Random Forest (RF) and Artificial Neural Network (ANN)

Machine Learning Algorithm	Advantages	Limitations
Support vector machine (SVM) (Ravikumar and P. Saraf, 2020) (F. L. Marchai, W. Martin and D. Suhartono, 2021) (X. Yuan, J. Yuan, T. Jiang and Q. U. Ain, 2020)	- Lower risk of overfitting. - Able to handle multiple feature spaces. - More robust compared to Linear Regression. - Classifies semi-structured and unstructured data, such as texts and photos, more accurately.	- Dealing with large and complex datasets are very expensive. - Data noises may cause it to perform poorly. - Difficult to understand the resultant model, weight and impact of variables. - Generic SVM cannot classify more than two classes unless extended.
Random forest (RF) (Ravikumar and P. Saraf, 2020) (X. Yuan, J. Yuan, T. Jiang and Q. U. Ain, 2020) (I. Kumar, K. Dogra, C. Utreja and P. Yadav, 2018)	- The mean values from the outputs of its constituent decision trees is calculated by Random Forest to lower the chance of variance and overfitting of training data compared to Decision Tree. - Perform well for large datasets. - It can estimate which variables or attributes are crucial in the classification.	- More complicated and computationally expensive. - Need to define the number of base classifiers. - Higher risk of overfitting.
Artificial neural network (ANN) (A. Vij, K. Saxena and A. Rana, 2021)	- Able to detect nonlinear relationships between dependent and independent variables that are complex.	- 'Black box' characteristic means that user does not involve to the decision-making process.

(S. O. Ojo, P. A. Owolawi, M. Mphahlele and J. A. Adisa, 2019) (F. L. Marchai, W. Martin and D. Suhartono, 2021)	- Less formal statistical training is required by the model. - Availability of multiple training algorithms. - Classification and regression problems can apply this algorithm.	- Classification problem is computationally expensive to train the network. - Pre-processing is required for predictor or independent variables.

2.6 Summary

In conclusion, this chapter discuss about traditional and modern approaches in current stock market prediction model. There are many machine learning algorithm like Support Vector Machine (SVM), Random Forest (RF) and Artificial Neural Network (ANN) applied in prediction model. These algorithms have shown their value in improving the accuracy of prediction model compared to the traditional ways, but they have their own advantages and disadvantages as well. Stock price prediction model with high accuracy is not an easy to task to achieve as there are many factors that will affect the price so it is important to have assistant from machine learning algorithms to obtain a more reliable signal on the stock market.

CHAPTER 3

METHODOLOGY

3.1 Overview

This chapter focuses on the research framework and the methodology of machine learning algorithm proposed which is Long Short Term Memory (LSTM). The research framework is conducted step by step to achieve the objectives set. Detail about LSTM is also explained and step to build a predictor based on this model.

3.2 Research Framework

Phase 1 of this research includes study on the domain, gathering data and processing the data. The study covers what is stock market, how it works and why machine learning is important in stock market value prediction. Stock market value data is collected from Yahoo Finance as the input of proposed technique, which is Long Short Term Memory (LSTM). Next, the data will be processed before implementing it into the LSTM model. Phase 2 covers the design of LSTM architecture and the development of LSTM model. The last phase which is phase 3 will enter the validation and verification of the result after the result is being produced by the LSTM model. Lastly, the result produced by each LSTM architecture is used to compared through the value of Root Mean Square Error (RMSE) to determine either the existing work or proposed technique model works better.

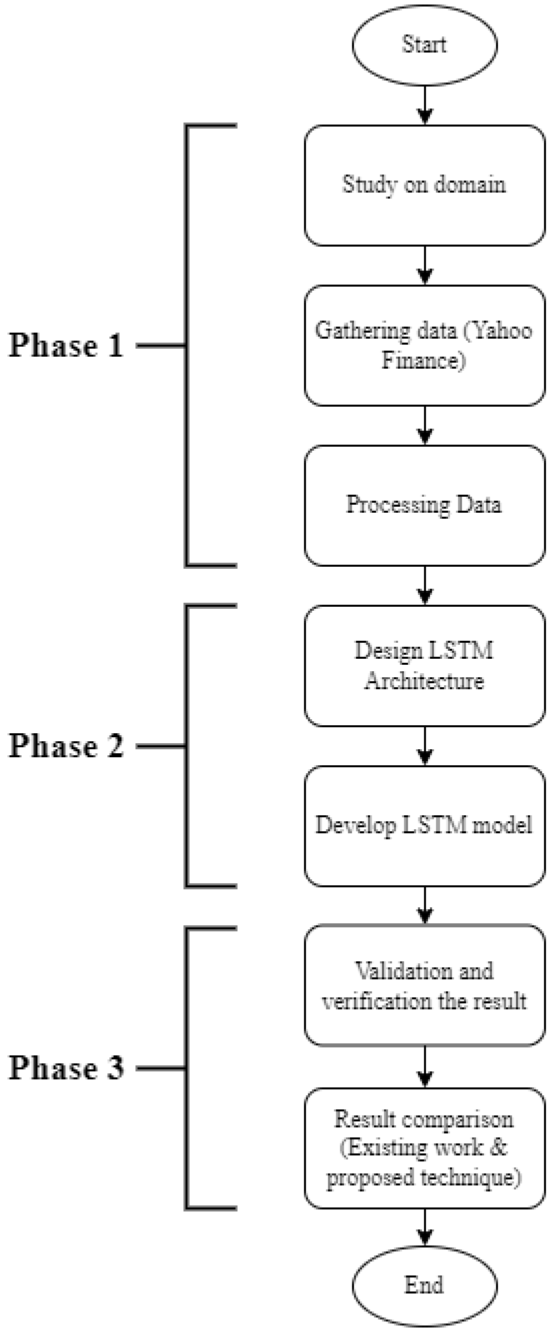

Figure 3.1**:** Overview Phases in the Research Framework

3.3 Long Short Term Memory (LSTM)

Long Short Term Memory (LSTM) has now become quite popular for handling time series prediction tasks (S. Hochreiter and J. Schmidhuber, 1997). LSTM as a modified RNN method, it performs well on different varieties of problems and has been frequently applied even until now. LSTM algorithm is very popular in financial time series forecasting especially in the stock. To show the impact of earlier incidences on the stock market's opening price, data information of paper articles are consolidated by Akita et al. LSTM system is included with formula to carry out accurate forecasts, that is take care with numerical and printed information (R. Akita, A. Yoshihara, T. Matsubara and K. Uehara, 2019). LSTM algorithm and CNN-sliding window methods are applied for stock price prediction (S. Selvin, R. Vinayakumar, E. Gopalakrishnan, V. K. Menon and K. Soman, 2017). Besides that, the improvement of LSTM model in terms of accuracy compared to other regression models through their research is demonstrated (M. A. Istiake Sunny, M. M. S. Maswood and A. G. Alharbi, 2020).

LSTM figures out issues like how to regather data over time and present the gate units and memory cells in the neural network design. Cell states in the memory cell keep track of recently experienced data. The combination of cell state controls the output for each moment memory cell receive information and refresh the cell state. If the memory cell receives any other information, the output is processed to utilize information then refreshed the cell state. LSTM's default behaviour is to retain information for a period of time and they do not acquire it through struggle.

LSTM consists of an input layer, two hidden layers and an output layer. Apiece neuron also known as "memory cell" connects previous information to the current errand in the LSTM model. The network is effectively connected to its memory and input by using the memory cells and it provides a higher predictive capacity.

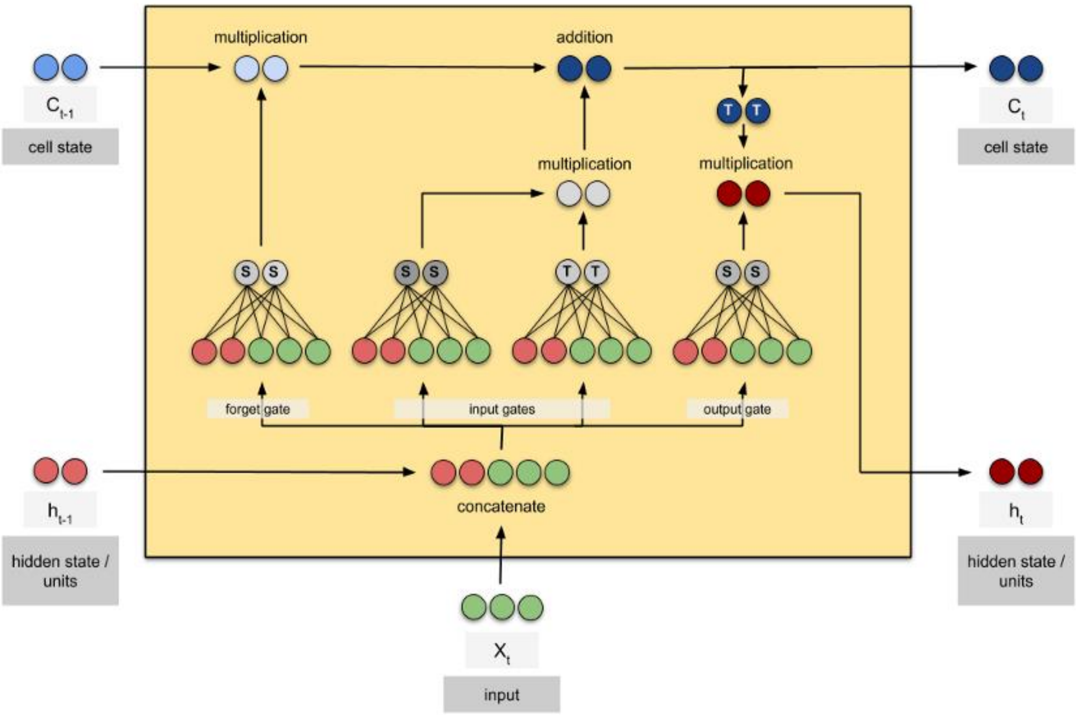

Figure 3.2: LSTM Memory Cell Structure

LSTM unit consists of three main components:

- Forget Gate: It decides which information can be ignored or require attention. The output values should be disregarded if the values are 0. In another hand, when the output values close to 1, the cell state should proceed with the sections.
- Input gate: The input network analysis decides which cell information is retained and which cell states are updated based on the previous output, input and previous cell state.
- Output Gate: It reacts according to the cell state and the input state, this section decides information to be transferred to the subsequent point in the network.

Functions in LSTM cell:

- tanh: The second derivative of the function can be held for a long time before zero is needed to solve the vanishing gradient issue.

- Sigmoid: The sigmoid output values can either be 0 or 1, it is used to discontinue or recall its knowledge. This allows the information from the LSTM's unit to pass through.

Figure 3.3 shows the total 6 steps involved in building the LSTM based stock market price predictor:

Step 1: The dataset for this research is a collection of Google (GOOG) stock price daily history, which will be employed in the stock market value prediction model. In the model, the stock data can directly be imported from Yahoo Finance.

Step 2: Only selected features are input into the neural network after feature selection is conducted. The closing price is chosen in this research.

Step 3: To build a LSTM model, stock prices data are separated into a training set and a test set. Next, normalize the data so that all the values are ranged from 0 to 1. 70% of the closing prices from our acquired stock data as training set and 30% of data as testing set.

Step 4: Now, set up the LSTM network architecture using the open-source machine learning package Tensorflow.

Step 5: At this phase, train LSTM model by fitting it with the training set. Prior to that, an optimizer and a loss function are set for the model.

Step 6: Evaluate the model with the testing set and calculate the RMSE value.

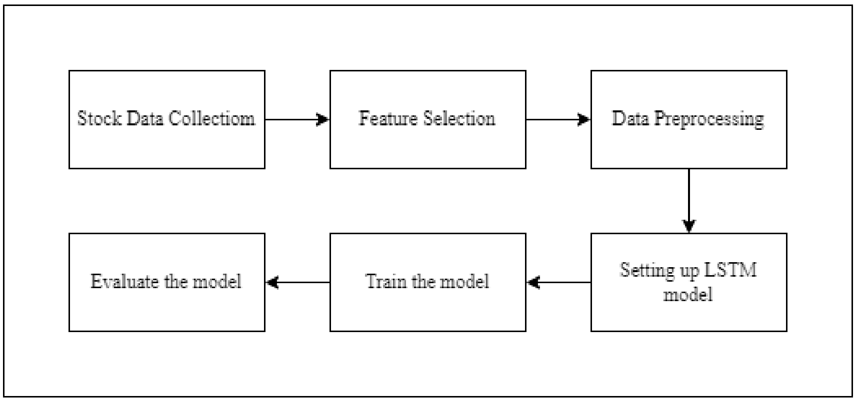

Figure 3.3: Architecture of LSTM model

3.4 Dataset

The dataset used for this research is collection of historical daily stock prices from Apple (APPL). The dataset is obtained from Yahoo Finance (Yahoo is part of the Yahoo family of brands, 2022). The data includes Open Price (stock price when the market opens), High (the highest price achieve on the trading day), Low (the lowest price achieve on the trading day), Close Price (stock price when the market closes), Adjusted Closing Price (Dividends, stock splits, and new stock offerings are some of the factors that affect the stock value, which is calculated using the close price) and Volume (number of shares bought and sold). A collection of observations collected through repeated measurements over time is called time series data. Figure 3.4 shows one of the parts of historical daily price of Apple (APPL) starting from January 1, 2016 and ending on October 1, 2021.

	Open	High	Low	Close	Adj Close	Volume
Date						
2016-01-04	25.652500	26.342501	25.500000	26.337500	24.111496	270597600
2016-01-05	26.437500	26.462500	25.602501	25.677500	23.507277	223164000
2016-01-06	25.139999	25.592501	24.967501	25.174999	23.047247	273829600
2016-01-07	24.670000	25.032499	24.107500	24.112499	22.074553	324377600
2016-01-08	24.637501	24.777500	24.190001	24.240000	22.191271	283192000

Figure 3.4: Historical Daily Price of Apple (APPL)

3.5 Performance Measurement

The root mean squared error, RMSE, is one of the most often used ways of calculating accuracy metrics. It demonstrates how far predictions depart from actual measurements using Euclidean distance. Due to the fact that it employs actual measurements at each projected data point, RMSE is frequently utilised in supervised learning applications. Formula 3.1 will be used to evaluate the result:

$$RMSE = \sqrt{\frac{\sum_{i}^{N}(Predicted - Actual)^2}{N}} \tag{3.1}$$

3.6 Hardware & Software Requirements

Hardware and software requirement are based on the development of stock price predictor.

Table 3.1: Hardware Specifications

Hardware	**Specification**
ROG STRIX LAPTOP G531GT.308	- Intel(R) Core(TM) i5-9300H CPU@2.40GHz processor - Intel(R) UHD Graphics 630 graphic card - Intel(R) Wireless-AC 9560 wireless card

Table 3.2: Software Specifications

Software	**Specification**
Operating System	Microsoft Windows 10 Home Single Language 64-bit
Microsoft Office Word 2013	Documentation of development from chapter 1 to chapter 5
Microsoft Office Power Point 2013	Preparation of presentation slide

Jupyter Notebook	Development of stock price predictor system
Zoom & Loom	Recording of presentation video

3.7 Summary

In this research paper, LSTM algorithm is reviewed and discussed. Research framework is constructed to make sure that research for LSTM is on the right track. We also discussed about how memory cell in LSTM works and the components' respective role inside. Each step is stated to build a LSTM model for stock market value prediction and the result will be evaluated based on Root Mean Square Error (RMSE) value.

CHAPTER 4

RESULTS AND DISCUSSIONS

4.1 Introduction

This chapter briefly discusses about the implementation of the research. The implementation is defined to meet the objectives that were stated to show the result is relevance to the research. Long-Short Term Memory (LSTM) algorithm was proposed in the experiment where the dataset was divided into training and testing phase. The flow of this chapter is started from the implementation the basic step for building the LSTM model, the process training and testing the dataset.

4.2 Proposed Technique

LSTM model is made up of a single hidden layer LSTM followed by a feedforward output layer. This model's extension, the Stacked LSTM, includes numerous hidden LSTM layers, each of them consists of a number of memory cells. The model gets deeper when stacking the LSTM hidden layers, more effectively defining the method as deep learning. More hidden layers can be added to deepen the multilayer perceptron neural network.

A dense layer in a neural network is one whose connections to its preceding layers are strong, meaning each neuron in the layer is connected to each neuron in the layer preceding it. In Artificial Neural Network (ANN) networks, this layer is the one that is most used the most. Basically, Each neuron in a dense layer is used to modify the dimensions of the vectors. Figure 4.1 shows the experiment setup for 3 different LSTM architecture.

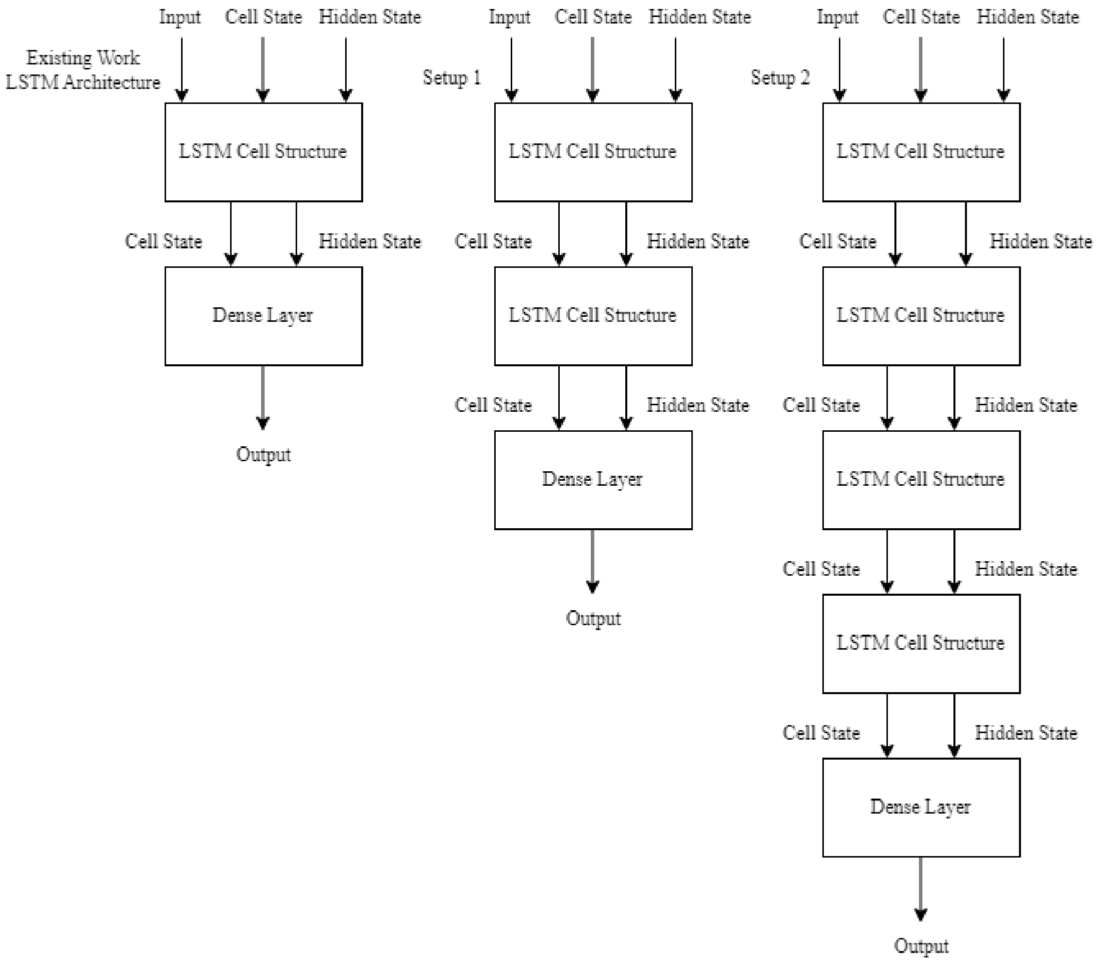

Figure 4.1: Experiment Setup for 3 different LSTM architecture

4.3 Experimental Setup for LSTM Architecture

4.3.1 Acquisition of Stock Data

Firstly, import all the required libraries in order to build the LSTM based stock market price predictor. yFinance is an open-source Python library that allows retrieving stock data from Yahoo Finance without spend money. Utilize the yFinance download method to acquire the stock prices of AAPL over the last 5 years, beginning on January 1, 2016, and ending on October 1, 2021. Figure 4.3 shows the preview of the data.

```python
import math
import yfinance as yf
import numpy as np
import pandas as pd
from sklearn.preprocessing import MinMaxScaler
import matplotlib.pyplot as plt
import tensorflow as tf
from tensorflow import keras
from tensorflow.keras import layers

stock_data = yf.download('AAPL', start='2016-01-01', end='2021-10-01')
stock_data.head()
```

Figure 4.2: Code for importing libraries and preview data

	Open	High	Low	Close	Adj Close	Volume
Date						
2016-01-04	25.652500	26.342501	25.500000	26.337500	24.111496	270597600
2016-01-05	26.437500	26.462500	25.602501	25.677500	23.507277	223164000
2016-01-06	25.139999	25.592501	24.967501	25.174999	23.047247	273829600
2016-01-07	24.670000	25.032499	24.107500	24.112499	22.074553	324377600
2016-01-08	24.637501	24.777500	24.190001	24.240000	22.191271	283192000

Figure 4.3: Preview of data

4.3.2 Visualizing Stock Prices History

Prior to preparing to build a LSTM model, let's take a look at the historical prices' movement of AAPL by plotting a line chart. Set the plot figure size and title for the line graph. Use the Matplotlib plot method to create a line chart for historical close prices of AAPL. Set the x-axis and y-axis labels of the line graph. Figure 4.5 shows the historical prices of AAPL for past five years.

```python
plt.figure(figsize=(15, 8))
plt.title('Stock Prices History')
plt.plot(stock_data['Close'])
plt.xlabel('Date')
plt.ylabel('Prices ($)')
```

Figure 4.4: Code for visualizing historical prices movement of AAPL

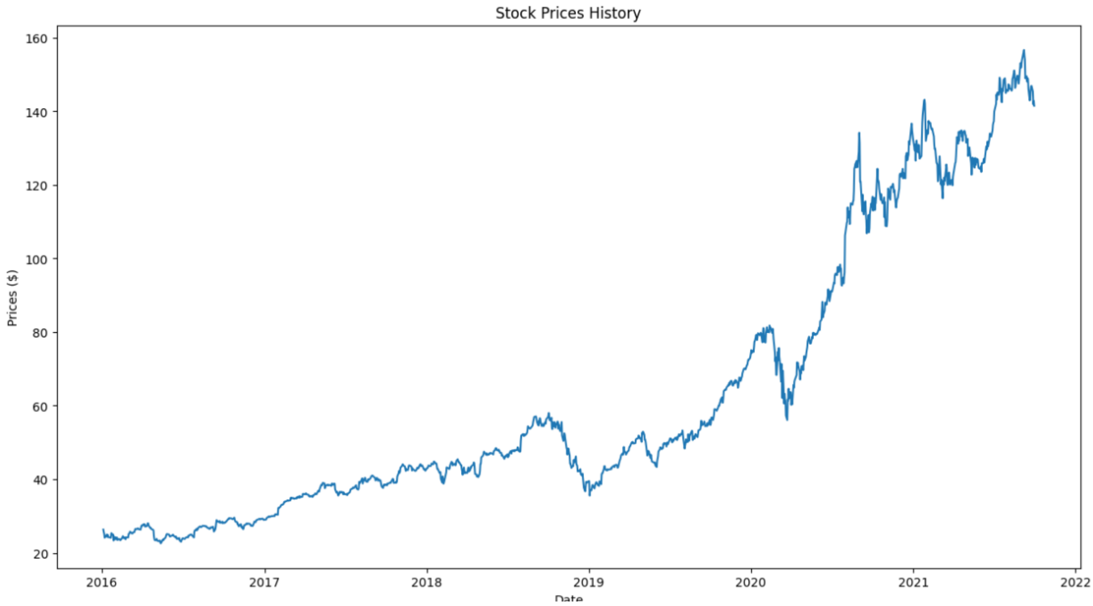

Figure 4.5: Historical Prices of AAPL for past five years

4.3.3 Data Pre-processing and Preparation of training set

Separate out stock prices data into a training and a testing set to build a LSTM model. First, extract the closing prices from the gathered stock data and turn it into a series of numbers. Calculate the data size for 70% of the dataset. The math.ceil method is to ensure the data size is rounded up to an integer. Normalize all stock data ranging from 0 to 1 by using the Scikit-Learn MinMaxScaler. Normalized data is also transformed into a 2D array. Set apart the first 70% of the stock data as the training set. For a sequence of feature data (x train) and a sequence of label data (y train), make an empty list. Create a 60-days window of historical prices (i-60) as our feature data (x_train) and the following 60-days window as label data (y_train). Convert the feature data (x_train) and label data (y_train) into numpy array as it is the data format accepted by the Tensorflow when training a neural network model. Reshape again the x_train and y_train into a three-dimensional array.

```
close_prices = stock_data['Close']
values = close_prices.values
training_data_len = math.ceil(len(values)* 0.7)

scaler = MinMaxScaler(feature_range=(0,1))
scaled_data = scaler.fit_transform(values.reshape(-1,1))
train_data = scaled_data[0: training_data_len, :]

x_train = []
y_train = []

for i in range(60, len(train_data)):
    x_train.append(train_data[i-60:i, 0])
    y_train.append(train_data[i, 0])

x_train, y_train = np.array(x_train), np.array(y_train)
x_train = np.reshape(x_train, (x_train.shape[0], x_train.shape[1], 1))
```

Figure 4.6: Code for preparing training set

4.3.4 Preparation of test set

Extract the closing prices from our normalized dataset (the last 30% of the dataset). Similar to the training set, create feature data (x_test) and label data (y_test) from test set. Convert the feature data (x_test) and label data (y_test) into numpy array. Reshape again the x_test and y_test into a three-dimensional array.

```
test_data = scaled_data[training_data_len-60: , : ]
x_test = []
y_test = values[training_data_len:]

for i in range(60, len(test_data)):
  x_test.append(test_data[i-60:i, 0])

x_test = np.array(x_test)
x_test = np.reshape(x_test, (x_test.shape[0], x_test.shape[1], 1))
```

Figure 4.7: Code for preparation of test set

4.3.5 Setting Up LSTM Network Architecture

Now, we are ready to use an open-source machine learning library, Tensorflow, to set up LSTM model architecture. Define a sequential model, which has a stack of layers that is linear. Add a LSTM layer and give it 100 network units. Set the return_sequence to true so that the output of the layer will be another sequence of the same length. At last, add a densely connected layer that specifies the output of 1 network unit. Figure 4.9 shows the summary of our LSTM network architecture.

```
model = keras.Sequential()
model.add(layers.LSTM(50, return_sequences=False, input_shape=(x_train.shape[1], 1)))
model.add(layers.Dense(1))
model.summary()
```

Figure 4.8: Code for setting up LSTM network architecture

```
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 lstm_1 (LSTM)               (None, 60, 50)            10400

 lstm_2 (LSTM)               (None, 50)                20200

 dense_1 (Dense)             (None, 1)                 51

=================================================================
Total params: 30,651
Trainable params: 30,651
Non-trainable params: 0
_________________________________________________________________
```

Figure 4.9: Summary of LSTM network architecture

4.3.6 Training the LSTM Model

At this stage, prepared to put the training set into LSTM model in order to train it. First define an optimizer and a loss function for model before moving forward. Use the "Adam" optimizer and set the loss function to the mean square error. By fitting it with the training set, train the model. Test it by running the training for 3 epochs with batch size of 1.

```
model.compile(optimizer='adam', loss='mean_squared_error')
model.fit(x_train, y_train, batch_size= 1, epochs=3)
```

```
Epoch 1/3
953/953 [==============================] - 16s 13ms/step - loss: 3.4578e-04
Epoch 2/3
953/953 [==============================] - 12s 13ms/step - loss: 1.0638e-04
Epoch 3/3
953/953 [==============================] - 12s 13ms/step - loss: 8.4158e-05
```

Figure 4.10: Code for training LSTM model

4.3.7 Model Evaluation

Apply the root mean square error (RMSE) measure to evaluate model's effectiveness after testing the trained LSTM model against the test data. Apply the model to predict the stock prices based on the test set. Use the inverse_transform method to denormalize the predicted stock prices. Apply the RMSE formula to calculate the degree of discrepancy between the predicted prices and real prices (y_test) and display the result.

```
predictions = model.predict(x_test)
predictions = scaler.inverse_transform(predictions)
rmse = np.sqrt(np.mean(predictions - y_test)**2)
rmse
```

```
10/10 [==============================] - 1s 14ms/step

0.7326575678525087
```

Figure 4.11 Code for example RMSE result

4.4 Result of Experimental Setup and Discussions

In this experiment, the LSTM model is built using the Keras framework. Google created the high-level Keras API, which is built on the TensorFlow framework. The entire experiment is carried out on single, two layers LSTM and four layers LSTM respectively. Figure 4.11, Figure 4.12 and Figure 4.13 show the parameters LSTM network with 50 neurons per layer.

```
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 lstm (LSTM)                 (None, 50)                10400

 dense (Dense)               (None, 1)                 51

=================================================================
Total params: 10,451
Trainable params: 10,451
Non-trainable params: 0
_________________________________________________________________
```

Figure 4.12: Existing Work LSTM Architecture

```
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 lstm_1 (LSTM)               (None, 60, 50)            10400

 lstm_2 (LSTM)               (None, 50)                20200

 dense_1 (Dense)             (None, 1)                 51

=================================================================
Total params: 30,651
Trainable params: 30,651
Non-trainable params: 0
_________________________________________________________________
```

Figure 4.13: Setup 1 LSTM Architecture

```
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 lstm_3 (LSTM)               (None, 60, 50)            10400

 lstm_4 (LSTM)               (None, 60, 50)            20200

 lstm_5 (LSTM)               (None, 60, 50)            20200

 lstm_6 (LSTM)               (None, 50)                20200

 dense_2 (Dense)             (None, 1)                 51

=================================================================
Total params: 71,051
Trainable params: 71,051
Non-trainable params: 0
_________________________________________________________________
```

Figure 4.14: Setup 2 LSTM Architecture

Table 4.1: Table of Result for LSTM Stock Market Price Prediction Model

LSTM Architecture / Experimental Run	Existing Work (Single Layer) LSTM Architecture	Setup 1 (2 Layer) LSTM Architecture	Setup 2 (4 Layer) LSTM Architecture
1	2.04	2.89	8.54
2	1.11	0.22	7.34
3	5.15	2.06	3.17
4	0.91	1.02	4.75
5	6.22	1.07	0.18
6	3.63	4.86	2.52
7	0.75	3.83	0.53
8	0.58	1.05	11.0
9	1.66	0.69	2.27
10	6.85	2.37	1.42
Average	2.89	2.01	4.23

From Table 4.1, we can see that the average Root Mean Square Error (RMSE) for 2 layers of LSTM has better performance compared to one layer and four-layer LSTM model. The model's performance improves significantly when we add another LSTM layer. More additional layers, however, would not be beneficial since the model might overfit or stagnate (X. Zhang, 2021).

In their use of LSTMs for voice recognition, Graves et al. introduced stacked LSTMs or called deep LSTMs. A single LSTM layer follows a standard feedforward output layer in the original LSTM model. The stacked LSTM is an extension of this model that has numerous hidden LSTM layers with several memory cells on each layer. The model is deeper due to the stacked LSTM hidden layers, the term "deep learning technique" is more properly described. The success of the approach on a variety of challenging prediction problems is credited to the depth of neural networks.

It is known that the additional hidden layers integrate the learnt representation from previous layers to produce new representations at a higher abstraction level. For example, from lines to shapes to objects. Single hidden layer multilayer perceptron of suitable size is enough for most of the function. Another approach that uses fewer neurons and trains more quickly is to deepen the network. In the end, depth optimization is a form of representational optimization.

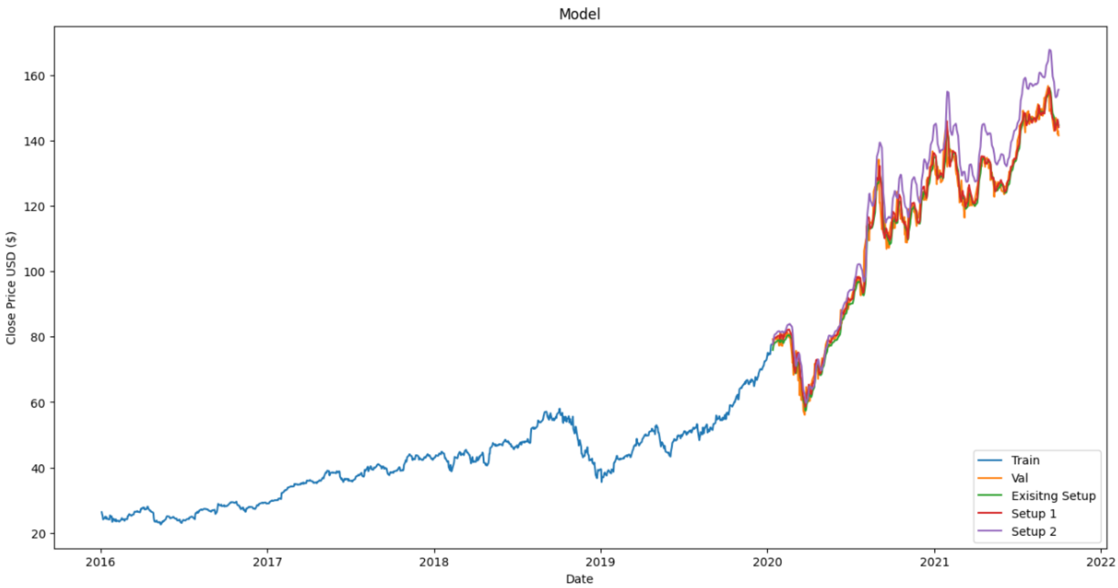

Figure 4.15: Stock Market Price Prediction Graph

From Figure 4.14, we can see that the overall price trend of each LSTM model follows the real value of the price, however 2 layers LSTM model outperform other two LSTM model with lower Root Mean Square Value (RMSE) and shows that it is a better model in stock market price prediction. In general, multilayer LSTM model should perform better but it does not apply to every type of problem. Two layers LSTM model is enough to handle the prediction for stock market price as it is not a very complex problem and has many different parameters. With the result obtained, we can conclude that two-layer LSTM model is optimal for stock market price prediction, and it works well.

CHAPTER 5

CONCLUSION

5.1 Introduction

In this research, the historical prices of Apple Company were studied from year 2016 to 2021. First, standardisation is employed to pre-process the original data. Second, the training samples are learnt by using the LSTM stacked model and single-layer model, respectively. Lastly, the test set is used to verify its performance, and both single-layer and stacked LSTM are evaluated using the expected outcome. The suggested stacked LSTM, in contrast to earlier research, offers a more complicated representation of the time-series data, obtaining data at various scales and enhancing time-series prediction precision.

In conclusion, LSTM is one of the effective approaches for predicting stock prices. The predicted stock prices should not be relied on 100% without further analysis. This is because the prediction is only based on historical price movements, which are often not the only factors affecting future price movements.

5.2 Research Constraints

In this research, we can only back test previous data, there is a major drawback to employing machine learning algorithms to forecast stock values. Unexpected events can cause market movements to diverge from historical trends. Political activities, disasters, wars, etc. which can affect the stock market are not taken into consideration in this research for the stock market price predictor. Further fundamental and market analysis is required to support investment decision-making.

5.3 Future Work

There may be some discrepancy between the experimental conditions and the actual scenario since the experiment does not take into account market dynamics or investors' investing arbitrariness. To summarise, in the future study, we can think about gathering data on the market, investor sentiment, and other elements to help the model trains better and increase forecast accuracy so that we can offer a useful resource for stock investors.

REFERENCES

A. Vij, K. Saxena and A. Rana, "Prediction in Stock Price Using of Python and Machine Learning," 2021 9th International Conference on Reliability, Infocom Technologies and Optimization (Trends and Future Directions) (ICRITO), 2021, pp. 1-4, doi: 10.1109/ICRITO51393.2021.9596513.

F. L. Marchai, W. Martin and D. Suhartono, "Stock Prices Prediction Using Machine Learning," 2021 8th International Conference on Information Technology, Computer and Electrical Engineering (ICITACEE), 2021, pp. 79-84, doi: 10.1109/ICITACEE53184.2021.9617222.

Gurjar, M., Naik, P., Mujumdar, G., & Vaidya, T. (2018). Stock market prediction using ANN. International Research Journal of Engineering and Technology, 5(3), 2758-2761.

I. Kumar, K. Dogra, C. Utreja and P. Yadav, "A Comparative Study of Supervised Machine Learning Algorithms for Stock Market Trend Prediction," 2018 Second International Conference on Inventive Communication and Computational Technologies (ICICCT), 2018, pp. 1003-1007, doi: 10.1109/ICICCT.2018.8473214.

K. J, H. E, M. S. Jacob and D. R, "Stock Price Prediction Based on LSTM Deep Learning Model," 2021 International Conference on System, Computation, Automation and Networking (ICSCAN), 2021, pp. 1-4, doi: 10.1109/ICSCAN53069.2021.9526491.

K. Chen, Y. Zhou and F. Dai, "A lstm-based method for stock returns prediction: A case study of china stock market", 2015 IEEE international Conference on big data (big data), pp. 2823-2824, 2015.

M. A. Istiake Sunny, M. M. S. Maswood and A. G. Alharbi, "Deep Learning-Based Stock Price Prediction Using LSTM and Bi-Directional LSTM Model," 2020 2nd Novel Intelligent and Leading Emerging Sciences Conference (NILES), 2020, pp. 87-92, doi: 10.1109/NILES50944.2020.9257950.

M. Nabipour, P. Nayyeri, H. Jabani, S. S. and A. Mosavi, "Predicting Stock Market Trends Using Machine Learning and Deep Learning Algorithms Via Continuous and Binary Data; a Comparative Analysis," in IEEE Access, vol. 8, pp. 150199-150212, 2020, doi: 10.1109/ACCESS.2020.3015966.

P. Srivastava and P. K. Mishra, "Stock Market Prediction Using RNN LSTM," 2021 2nd Global Conference for Advancement in Technology (GCAT), 2021, pp. 1-5, doi: 10.1109/GCAT52182.2021.9587540.12.

R. Akita, A. Yoshihara, T. Matsubara and K. Uehara, "Deep learning for stock prediction using numerical and textual information," 2016 IEEE/ACIS 15th International Conference on Computer and Information Science (ICIS), 2016, pp. 1-6, doi: 10.1109/ICIS.2016.7550882.

S. Hochreiter and J. Schmidhuber, "Long Short-Term Memory," in Neural Computation, vol. 9, no. 8, pp. 1735-1780, 15 Nov. 1997, doi: 10.1162/neco.1997.9.8.1735.

S. O. Ojo, P. A. Owolawi, M. Mphahlele and J. A. Adisa, "Stock Market Behaviour Prediction using Stacked LSTM Networks," 2019 International Multidisciplinary Information Technology and Engineering Conference (IMITEC), 2019, pp. 1-5, doi: 10.1109/IMITEC45504.2019.9015840.

S. Ravikumar and P. Saraf, "Prediction of Stock Prices using Machine Learning (Regression, Classification) Algorithms," 2020 International Conference for Emerging Technology (INCET), 2020, pp. 1-5, doi: 10.1109/INCET49848.2020.9154061.

Surbhi Sharma and Baijnath Kaushik, 2018. Quantitative Analysis of Stock Market Prediction for Accurate Investment Decisions in Future. Journal of Artificial Intelligence, 11: 48-54.

S. Selvin, R. Vinayakumar, E. Gopalakrishnan, V. K. Menon and K. Soman, "Stock price prediction using lstm rnn and cnn-sliding window model", 2017 international conference on advances in computing communications and informatics (icacci), pp. 1643-1647, 2017.

T. Manojlović and I. Štajduhar, "Predicting stock market trends using random forests: A sample of the Zagreb stock exchange," 2015 38th International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO), 2015, pp. 1189-1193, doi: 10.1109/MIPRO.2015.7160456.

X. Yuan, J. Yuan, T. Jiang and Q. U. Ain, " ," in IEEE Access, vol. 8, pp. 22672-22685, 2020, doi: 10.1109/ACCESS.2020.2969293.

X. Zhang, C. Li, K. -L. Chen, D. Chrysostomou and H. Yang, "Stock Prediction with Stacked-LSTM Neural Networks," 2021 IEEE 21st International Conference on Software Quality, Reliability and Security Companion (QRS-C), 2021, pp. 1119-1125, doi: 10.1109/QRS-C55045.2021.00166.

Z. Liu, Z. Dang and J. Yu, "Stock Price Prediction Model Based on RBF-SVM Algorithm," 2020 International Conference on Computer Engineering and Intelligent Control (ICCEIC), 2020, pp. 124-127, doi: 10.1109/ICCEIC51584.2020.00032.