Application of Machine Learning Algorithms to Predict the Closing Price of the Johannesburg Stock Exchange All-Share Index

Makgwedi Precious Makganoto
1799341

Supervisors:
Dr. Hima Vadapalli
Dr. Thabang Mokoteli

A research report submitted in partial fulfillment of the requirements for the degree of Master of Science in the field of e-Science

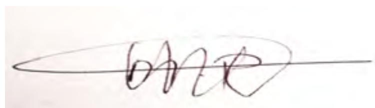
in the

School of Computer Science and Applied Mathematics
University of the Witwatersrand, Johannesburg

21 August 2020

Declaration

I, Makgwedi Precious Makganoto, declare that this research report is my own, unaided work. It is being submitted for the degree of Master of Science in the field of e-Science at the University of the Witwatersrand, Johannesburg. It has not been submitted for any degree or examination at any other university.



Makgwedi Precious Makganoto

21 August 2020

Abstract

Stock markets are regarded as one of the most important indicators of the economy's strength and development. Predicting stock prices is of critical importance for investors who wish to minimise the risks of investments. Stock price prediction is a difficult task since stock prices are influenced by factors such as the financial status of the company, socio-economic conditions of the country, political atmospheres, and natural hazards. The Efficient Market Hypothesis (EMH) states that stock markets behave like a random walk and due to this reason, it is complex to forecast the stock market. Researchers use time series forecasting, technical, and fundamental analyses to predict the stock values while proving or disproving the EMH. In the past, researchers used traditional methods such as Autoregressive Integrated Moving Average (ARIMA) to predict stock prices. Currently, deep learning architectures are widely used to solve time-dependent problems and can provide a huge push to the problem of stock price prediction. The main objective of this study is to develop a framework that forecasts the daily closing price of All-Share index data based on deep learning techniques. To achieve this objective, Long Short Term Memory (LSTM) and Gated Recurrent Unit (GRU) are employed. A Vector Autoregressive (VAR) model is used to benchmark the deep learning techniques. The analysis is based on the Financial Times Stock Exchange (FTSE)/ Johannesburg Stock Exchange (JSE) All-Share (J203) data collected from Iress Expert. The results show that all the methods are able to predict the closing price of the index. GRU predicted the future closing price with an average Mean Absolute Percentage Error (MAPE) of 9.349% maximum while LSTM was able to predict with the maximum average error of 9.459%. A VAR model performed with the maximum average error of 2.152%.

Acknowledgements

I glorify God for his infinite love and mercy throughout my life. I thank Dino Gavrielides and Dr. Thabang Mokoteli from Wits Business School for providing me with the FTSE/JSE All-Share index data. I acknowledge Prof. Celik, Casey Sparkes, and Helen Robertson for the support and continuous encouragement. I thank Olaperi Okuboyejo for helping with grammar and proofreading of the report. I also thank Evans Thulare for assisting with proofreading of the report. I wish to extend my sincere thanks and appreciation to my supervisors Dr. Hima Vadapalli and Dr. Thabang Mokoteli for their support and guidance in completing this research report. Finally, I acknowledge the DST-CSIR National e-Science Postgraduate Teaching and Training Platform (NEPTTP) for financial support.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

The stock market, also known as the share or equity market is an aggregation of markets and exchanges where regular activities such as selling, purchasing, or supplying of stocks (shares) take place. A stock exchange is a place where brokers and traders of stocks trade in shares, bonds, and other securities. Companies make a profit through stock markets and businesses that trade publicly grow by raising extra funds as a result of trading. The stock markets are regarded as the most significant indicators of the economy's strength and development [25]. The movement of share prices is recorded in stock market indices. There are several approaches people can use to predict stock prices. These approaches comprise of time series forecasting, technical, and fundamental analyses. Prediction of stocks can either be long, medium, or short-term. Short-term prediction deals with forecasting of stocks for some weeks, days, minutes, etc. The medium-term forecasting includes the prediction of stock for a period of 1 to 2 years and the long-term covers prediction over a 2 year period [35]. Investment analysis is the process of evaluating and researching an industry or security to predict future performance and determine its suitability to investors.

Fundamental analysis is a class of investment analysis that deals with estimating the share prices of a company using economic factors such as sales, earnings, and profits. This technique is relevant for long-term forecasting. Technical analysis is the process of forecasting future values of stocks based on historical data. The most frequently used method for future stock price prediction is the Moving Average (MA). MA methods are suitable for short-term prediction. Time series analysis includes both linear and non-linear models. Linear models include statistical methods such as Autoregressive (AR), Autoregressive Integrated Moving Average (ARIMA) while non-linear models involve Autoregressive Conditional Heteroskedasticity (ARCH), Generalised Autoregressive Conditional Heteroskedasticity (GARCH), and deep learning techniques, etc. [35].

The Efficient Market Hypothesis (EMH) states that stock markets behave like a random walk and for this reason, it is complex to forecast the stock prices [15]. The essential complexity of the financial system makes stock price prediction a difficult task [24]. According to [7], EMH states that it is not worth it to use historical data to forecast security and stock prices. However, they argued that previous studies proved that the financial market does not behave like a random walk and EMH is just a part of the chaotic market hypothesis. The paper by [7] also explained that EMH consists of 3 hypotheses which include strong form, semi-strong form, and weak form stock hypotheses. In a weak form hypothesis, the stock prices mirror all information that is obtainable by exploring the market trading data. The strong form hypothesis consists of all information that is relevant to the firm and available to all the members of the company. Moreover, it is impossible to perform forecasting in this hypothesis.

1.1 Background

1.1.1 Johannesburg Stock Exchange

Capital formation is one of the most crucial challenges faced by developing countries, hence, stock exchanges play a major role in these countries [11]. The stock market is a significant measure of the global economy and it deals with trading of shares or goods and services listed in the stock exchange. There are several stock exchanges around the world which include the New York Stock Exchange (NYSE), National Association of Securities Dealers Automated Quotations (NASDAQ), London Stock Exchange (LSE), National Stock Exchange (NSE), Namibian Stock Exchange (NSX), etc. In Africa, the most commonly known stock exchange is the Johannesburg Stock Exchange (JSE) based in South Africa at the Gauteng

Province. The JSE is the only full-service securities exchange in South Africa where consumers and retailers buy or sell shares, stocks, interest rates products, equity, commodity, and currency derivatives in the markets [26].

There are different portfolios of shares included in the JSE such as All-Share, Top 40, and All-Gold indices. All-Share is a major JSE index consisting of more than 50 stocks, the top 40 market capitalisation companies, and 22 shares from all sectors and industries. The Top 40 index is composed of all listed companies that are under the top 40 in a stock exchange and it measures the South African stock market performance. All-Gold Index includes all listed companies that mine gold [24]. In this study, we are estimating the closing price of the All-Share index using machine learning algorithms specifically Recurrent Neural Network (RNN) special types called Long Short Term Memory (LSTM) and Gated Recurrent Unit (GRU). This study compares the results of these approaches to the Vector Autoregressive (VAR) model. Machine learning is the most popular subject in many domains around the world [38]. Researchers worked on stock price prediction using various technical, fundamental, and statistical indicators. Many experimenters and investors believe that developing neural networks can solve market complexities. The most commonly used techniques for time series forecasting are Artificial Neural Networks (ANN), Autoregressive Integrated Moving Average (ARIMA) models, and regression methods such as Support Vector Regression (SVR).

1.1.2 Time series

A time series is a collection of values that are measured over time. It is denoted by X_t where X is an observation and $t = 1, 2, 3, \dots n$ is a time period. There are 4 components of time series which include trend, seasonal, cyclical, and irregular. A time series model can either be multiplicative or additive. Trend and seasonal components are combined in a multiplicative system and added to the error variable. The magnitude of the seasonal effect is constant over time in additive models. A time series can either be stationary or non-stationary. A process is said to be stationary if its mean, variance, and covariance do not change over time. A process with trends and seasonality where the mean, variance, and covariance are changing over time is known to be a non-stationary process. In this study, a VAR model is used as a reference model for predicting the stock prices of the All-Share index. The goal is to compare the performance of the machine learning methods to the statistical models.

1.1.3 Vector Autoregressive

Vector Autoregressive (VAR) models were first introduced in the field of economics by Christopher Sims in 1980. A VAR determines the dynamic behavior of economic and financial time series [10]. Researchers use VAR models to predict the Gross Domestic Product (GDP), money supply, employment, etc. In finance, VAR models forecast spot prices, future values of securities, and foreign exchange rates across markets. It is also applied in accounting and marketing to make predictions of sales, earnings, and to test the effect of different factors on consumer behavior and predict future change. VAR models are found to perform better than other traditional methods in economics [20]. Techniques such as regression, exponential smoothing, and GARCH are also used for forecasting.

A VAR model generalises the Autoregressive (AR) model and it predicts multiple time series. In this model, all the variables are represented as a linear function of past lags of the variable itself and other variables included in the system. The AR model is a regression model that predicts future values based on past values.

AR of order p is represented by:

$$x_t = \phi_1 x_{t-1} + \phi_2 x_{t-2} + \phi_3 x_{t-3} + \ldots + \phi_p x_{t-p} + z_t \tag{1.1}$$

where $\phi_1, \phi_2, \ldots, \phi_p$ are the AR coefficients at lags $1, 2, \ldots, p$, and $x_{t-1}, x_{t-2}, \ldots, x_{t-p}$ are the past series, and z_t is the residual. The residual term relates to the present time period t.

A VAR model of order p, $VAR(P)$ is denoted by:

$$\mathbf{Z}_t = \mathbf{a} + \mathbf{A}_1 Z_{t-1} + \mathbf{A}_2 Z_{t-2} + \ldots + \mathbf{A}_p Z_{t-p} + \epsilon_t \tag{1.2}$$

where the variables

$\mathbf{Z}_t$: $(n \times 1)$ vector of the features.
$\mathbf{a}$: $(n \times 1)$ constant vector of intercepts.

$\mathbf{A}$: $(n \times n)$ coefficients matrices.

ϵ : $(n \times 1)$ vector of observations that are independently and identically distributed with a zero mean and a constant variance. It is also referred to as a white noise.

1.1.4 Recurrent Neural Network

RNN is a group of ANN that is commonly known as the most popular deep learning method in the field of Natural Language Processing (NLP), digit recognition, and financial forecasting. It is mostly used to analyse and model time series data or any other time-dependent process and it is used to predict both current and future states. It has the ability to learn from the previous information and this characteristic does not exist in ANN. It performs the same job for every component of a sequence and the outcome is dependent on the previous computations. It is sometimes regarded as multiple copies of similar networks, each being responsible for passing a message to a successor because of its loops that permit information to persist [28].

The internal memory of RNN allows it to remember its inputs, which is suitable for solving machine learning problems that involve sequential data. The information travels from the input to the output layer through the hidden layers in a Feedforward Neural Network (FFNN). The information passes through each node exactly once. FFNN lacks the memory for storing inputs received in the past, therefore, it is unable to recall what has happened previously. FFNN can only recall what happened during the training process. In RNN, information rotates through the loop and it considers the current input including what it learned in the past from the inputs received. RNN is unable to deal with long-term dependencies and gradient vanishing problem. The gradient is a slope of a function, it measures how the output of a function changes. When the value of the slope is zero the model stops learning and when the value is high the model learns very fast [30].

Figure 1.1 shows a variation among the RNN and FFNN in terms of the flow of information [12].

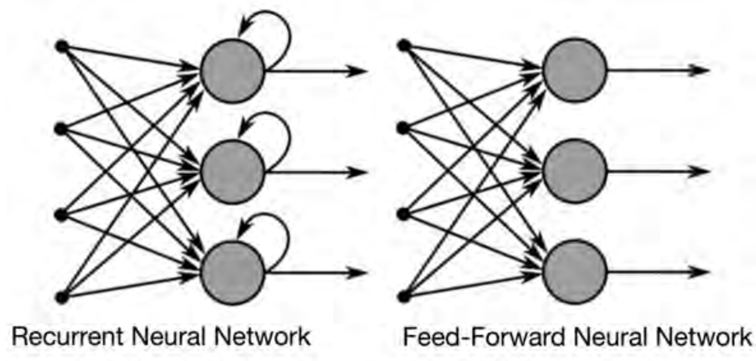


FIGURE 1.1: RNN vs. FFNN

The LSTM is a class of RNN designed to overcome long-term dependencies and vanishing gradient problems. The problem is caused by the smaller value of the gradient. When the gradient is small, the weight matrices of the initial layer fails to update effectively during the training process [36]. LSTM extends the memory of RNN and therefore, can recall long-term inputs. It consists of 3 gates which include output, forget, and input gates. The input gate is responsible for adding information to the cell state, forget gate removes information from the cell state while the output gate produces the outputs based on the information selected from the cell state. Figure 1.2 shows the architecture of the LSTM. The gates in LSTM have a sigmoid shape that allows it to use backpropagation. The values in a sigmoid function range between 0 and 1 [30].

The equations below are used to compute the LSTM with the forget gate. Suppose there is $c_o = 0$ and $h_0 = 0$ where a subscript t represent a timestep.

$$f_t = \sigma_g(W_f x_t + Z_f h_t - 1 + a_f) \tag{1.3}$$

$$i_t = \sigma_g(W_i x_t + Z_i h_t - 1 + a_i) \tag{1.4}$$

$$o_t = \sigma_g(W_o x_t + Z_o h_t - 1 + a_o) \tag{1.5}$$

$$c_t = f_t \circ c_{t-1} + i_t \circ \sigma_c(W_c Z_t + Z_c h_t - 1 + a_c) \tag{1.6}$$

$$h_t = o_t \circ +\sigma_h(c_t) \tag{1.7}$$

where the variables

$z_t \in \mathbb{R}^d$: inputs for the current timestamp.
$f_t \in \mathbb{R}^h$: the forget gate.
$i_t \in \mathbb{R}^h$: input gate.
$o_t \in \mathbb{R}^h$: output gate.
$h_t \in \mathbb{R}^h$: output produced by the LSTM block at the previous timestamp.
$c_t \in \mathbb{R}^h$: cell state.
$\mathbf{W} \in \mathbb{R}^{h \times d}$: weight of the respective layers and d and h are the number of features and hidden units.

Figure 1.2 shows a representation of the LSTM with its 3 gates [12].

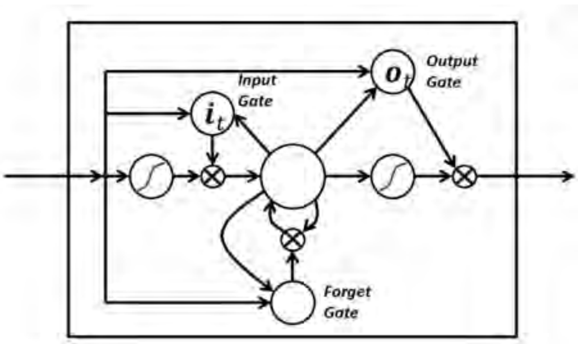


FIGURE 1.2: Structure of LSTM

The GRU is a simplified variation of the LSTM architecture. It was also established to deal with vanishing gradient problems and long-term dependencies existing in standard RNN [30]. Within a GRU, the forget and input gates are combined to form 1 gate called the forget gate. It has an additional gate called the reset gate. The reset and update gates decide what information must be passed to the output. GRU can store information from the previous time without removing information that is not useful to the prediction. It was proven that GRU performs well on a smaller dataset and it takes fewer parameters compared to the LSTM model. The equations below are used to compute a GRU hidden unit.

$$z_t = \sigma_g(W_t z_t + Z_z h_{t-1} + a_z) \tag{1.8}$$

$$r_t = \sigma_g(W_r z_t + Z_r h_{t-1} + a_r) \tag{1.9}$$

$$h_t = (1 - z_t) \circ h_{t-1} + z_t \circ \sigma_h(W_h Z_t + Z_h(r_t \circ h_{t-1}) + a_h) \tag{1.10}$$

where the variables

z_t : is the input features.
h_t : output values.
z_t : vector of the update gate.
r_t : vector of the reset gate.
w, z and a : constant vectors.
σ_g : sigmoid activation function.
σ_h : hyperbolic tangent activation function.

Figure 1.3 represents the GRU structure [12].

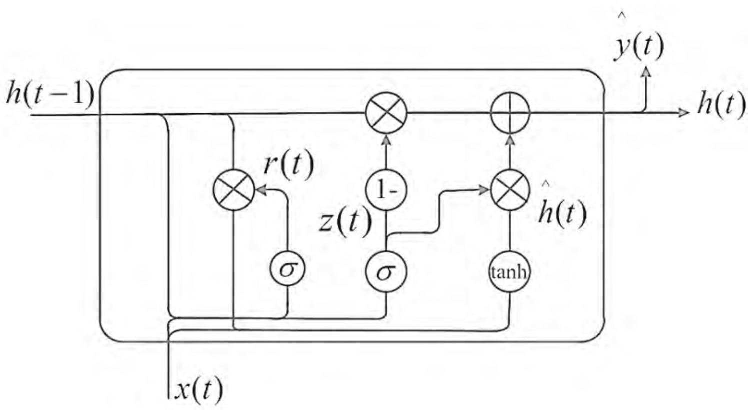


FIGURE 1.3: Structure of GRU

RNN is used in many domains such as NLP, time series prediction, image captioning, etc. It is preferred because of its ability to process a series of inputs. Even though it remembers past information its memory is unable to store information for a long period and due to this reason LSTM and GRU were introduced. Hence, LSTM and GRU are more suitable for solving time-dependent problems.

1.2 Problem Statement

In the past, technical analysts and stockbrokers performed forecasting using historical prices, the volume of stock, patterns of stock prices, and basic trends. Recently stock price prediction is a complex task since, stock values are affected by factors such as the company's financial status, political atmospheres, natural hazards, and the socio-economical condition of the country. The stock returns such as profit and dividends from the share market are consistently uncertain and ambiguous, therefore, traditional methods cannot give accurate prediction results [7]. There is a little theoretical and empirical analysis of the South African stock market. Several researchers applied statistical techniques such as ARIMA to forecast the share prices of the JSE. Author [24] observed that JSE had high degrees of error and bias in its prediction models due to the use of fundamental analysis on stock computations. As a result, he proposed the use of ANN to come up with a solution. The author found that ANN can predict the stock prices, however, it was outperformed by a random walk model. LSTM and GRU techniques have not been previously explored on the JSE data. Therefore, the current study applies these approaches for the first time to the JSE data to fill the gap. The VAR model will be used as a benchmark for deep learning methods.

1.3 Research Aims and Objectives

1.3.1 Research Aims

The purpose of this research is to apply LSTM and GRU to investigate the future closing price of the JSE All-Share index using 10 years of data.

1.3.2 Objectives

The objectives of this study are as follows:

- To identify if the volume, opening, highest, and lowest prices can be used to predict the closing price of the JSE data.
- To build a system that adopts GRU and LSTM to estimate the closing price of the All-Share index.
- To perform a comparative analysis between deep learning models and a statistical model specifically the VAR model.

- To assess the performance of LSTM and GRU in the stock price prediction of the South African market.

1.4 Research Questions

The following questions are answered by this research:

- How do LSTM and GRU perform compared to a VAR model?
- Can LSTM outperform GRU given the same feature setup?
- How does a set of hyper-parameters specifically, the number of epochs impact the performance of GRU and LSTM?

1.5 Motivation

The stock exchange plays a vital role in the economy as a whole and few studies have been conducted on the stock price prediction in South Africa. The existing studies are based on statistical methods while others explored the performance of ANN. LSTM and GRU are widely used to solve time-dependent problems and they are proven to give better results when compared to other methods. The findings of this study might help investors to make profitable decisions regarding investments and contribute to the literature.

1.6 Limitations

The key limitation faced in this study was time-constraint. The models took a long time to converge thus we were unable to perform hyper-parameter tuning. We have tried parameter tuning by implementing a grid search for Talos. The algorithm ran for over 20 hours and the computer crashed during the training process because of poor computing power. We attempted to use the Wits University cluster, although, we were unable to train the models due to network problems. Moreover, we failed to install the required packages. As a result, we were unable to investigate the performance of our models in-depth due to these limitations.

1.7 Conclusion

In this chapter, introduction, a background to the study, problem statement, aim and objectives of the study, research questions, motivation and limitations of the study were presented. The remaining chapters are arranged as follows: Chapter 2 highlights reviews of previous studies on the stock price prediction. Chapter 3 presents the methodology used to answer the objectives of the research report. Accuracy measures to be used are also summarised in Chapter 3. Results and discussion of findings are discussed in Chapter 4. Chapter 5 provide conclusions made by the study including recommendations.

Chapter 2

Literature Review

The background and introduction to the research report including aim and objectives are discussed in Chapter 1. This chapter presents the results of a review of related literature based on machine learning and statistical techniques used for stock price prediction. The techniques include Artificial Neural Network (ANN), Support Vector Machine (SVM), Convolutional Neural Network (CNN), Recurrent Neural Network (RNN), Random Forest (RF), Autoregressive (AR) and Autoregressive Integrated Moving Average (ARIMA).

2.1 Statistical techniques

The study by [16] used historical data for the National Association of Securities Dealers Automated Quotations (NASDAQ) index to make short-term predictions of stock prices. They applied various mathematical techniques to build models. They implemented 2 models, the first model was the Least-squares and the Fourier series expansion while the second model was ARIMA. A total of 10 stocks obtained from the Internet Information Providers Industry of the technology sector were used to build the systems. The models predicted the closing price of stocks for a period of 30 working days. The performance of the model was investigated by taking the discrepancy between the true values and the values predicted by a model. The 2 models successfully predicted the stock prices with a confidence interval of 95%.

The movement of the Johannesburg Stock Exchange (JSE) All-Share data were investigated by [23]. The authors used the South African Public mood data collected from Twitter Application Programming Interfaces to perform the experiments. They collected millions of tweets and examined them over 55 day trading periods. The dataset was collected for 39 trading days and the variable of interest was the mood state. A total of 4 states of mood in South Africa did not show any correlation with the movement of the All-Share index prices. It was revealed that depression has a direct negative relationship with the current day of All-Share index values. It was found that the fatigue mood and the closing price for the next day have a positive significant relationship. These indicated that the fatigue mood can determine the future values of the JSE shares. The researchers concluded that their results support the theory of behavioral finance [37], which says public mood may affect the stock market.

A research conducted by [33] used the AR technique to forecast stock prices. The history of data for the New York Stock Exchange (NYSE) was used to build an AR model that estimates the values of stocks. The correlation technique was used to investigate the relationship between stocks. The coefficients of the regression model were used to forecast future stock values. The input data were selected and grouped into train and test samples. The training data was used to fit an AR model. They employed the Moore and Penrose methods to estimate the coefficients of the regression model. The estimated coefficient values were used to predict the stock prices. The testing set was used to investigate the difference between the actual and estimated stock prices over time. It was found that the actual stock values were very close to the predicted stock prices. Therefore, they indicated that forecasting the returns on investment can help stockbrokers and financial institutions to perform future stock price prediction.

The closing values of the National Stock Exchange (NSE) Nifty 50 index were predicted by [4] using the Box-Jenkins methodology. They have used the Nifty 50 historical data to develop a model that can estimate the closing price of the stock market. The data contained prices recorded from January 2015 to December 2015 and it was made up of 245 observations. The Box-Jenkins (ARIMA) model was used to estimate future stock prices of the index. The ARIMA(0,1,1) was chosen as the model to make predictions because it was found to have the lowest value of the Bayesian Information Criterion (BIC). Their model forecasted with the smallest Mean Absolute Percentage Error (MAPE) and the data fitted the model with a correlation coefficient (R^2) of 94%. They concluded that the closing price of the stock market has a declining fluctuation trend for future trading days.

2.2 Machine Learning techniques

2.2.1 Artificial Neural Network

The study by [29] used ANN to build a prediction system that can guide investors of stock to make profitable financial choices. They designed the radial base function neural network and a Multi-layer Perceptron (MLP) architectures to forecast the closing price of the NASDAQ 100, All-share index, Nikkei 225, and Dow Jones Industrial Average. The historical data of the stock exchanges used covered the period from 5 January 2004 to 31 May 2005. Their model predicted the stock values with an accuracy of 74% for Nikkel 225 and Dow Jones Industrial Average while a minimum accuracy of 64% was obtained for the remaining datasets.

The performance of ANN with Hybridized Market Indicators in stock price forecasting was investigated by [1]. The authors used both technical and fundamental analyses of share market indicators to estimate future stock prices. A secondary data of various companies collected from the Published Stock Data was utilised and 18 features were considered as the inputs to the model. They trained a Feedforward Neural Network (FFNN) with backpropagation. The network was trained for 10000 epochs using the training data. The results produced by the network were evaluated by computing the difference between the estimated values and the true values. The outcomes indicated that a hybridized technique can predict the daily stock prices at the highest accuracy, it was found to perform better than technical analysis based approaches.

Techniques such as Artificial Neuro-Fuzzy Inference System (ANFIS) and ANN models were implemented by [6] to forecast the closing price of shares of the Dhaka Stock Exchange (DSE). A historical data of 5 biggest companies was used to build ANFIS and ANN models. The data covered a 3 years period from January 2013 to April 2015. The fields included in the data are the daily opening, closing, highest, and lowest prices, and the volume of shares traded daily. The closing price was the target variable while all other features were used as the input variables. They used the Root Mean Squared Error (RMSE) and R^2 to evaluate the performance of the predictions made by the model. They indicated that the RMSE closer to 0 implies that the model performs with the highest accuracy and when the value is close to

1 there is a strong correlation among the features. They concluded that the most significant method to forecast the stock prices of the DSE is the ANFIS.

The paper of [24] mentioned that it is impossible to forecast future prices of assets based on historical data according to the weak form of a market hypothesis. The financial system's underlying dynamics and the behavior of the market make forecasting difficult. The author proposed Artificial Intelligence (AI) techniques which included neuro-fuzzy systems, ANN, and SVM to predict future stock prices based on the JSE data for the All-share index. A random walk and Autoregressive Moving Average (ARMA) models were also implemented to compare the results. Daily data collected from the year 2002 to 2005 was used to build the models. The data excludes holidays and weekends and it was partitioned into 70% training and 30% testing. It was found that AI, linear (ARMA), and random walk models can forecast the stock prices of the All-share index. Neuro-fuzzy systems, ANN, and SVM were found to outperform the ARMA model while the random walk model was found to perform better than all other models. It was also found that the performance of the AI techniques depends on the accuracy measures. The MAPE, RMSE, Symmetric Mean Absolute Percentage Error (sMAPE), Mean Squared Error (MSE), and confusion matrix were used to evaluate the predictive accuracy of the model.

2.2.2 Support Vector Machine

The experiment performed by [13] compared the performance of logistic regression, SVM, and ANN. Their main objective was to develop a framework that can forecast stock prices for a given trading day. The authors used secondary data of the S&P 500 that was made up of tweets. The data used covered the period from January 2008 to April 2010. The technical indicators which included Relative Strength Index-Movement, On Balance Volume-Movement, Price Momentum Oscillator-Movement, Stochastic-Oscillator, and Weighted Moving Average-Movement were used as inputs to the models. Amongst the 3 models, the best performing model was selected as the final model to rely on when forecasting stock prices of an index. The authors found that the best performing model is the SVM, it outperformed all other models. Moreover, it was observed that the ANN model was outperformed by both the logistic regression and SVM models.

The performance of regression techniques was assessed by [27] on the stock price prediction. The goal was to build linear and polynomial regression models for forecasting the closing price of the S&P 500 index. They used historical data obtained from the Center for Research in Security Prices (CRSP) database. The data used to fit the Support Vector Regression (SVR) models have 2920 samples starting from the year 2005 to 2013. They trained the SVR models using both the Radial Basis Function (RBF) and polynomial kernels. The window sizes 5, 48, 92, 136, and 180 were also used to forecast the closing price 45 days into the future. The SVR model with the RBF kernel performed better than a model with a polynomial kernel. The accuracy of the model depended on the training window size.

A set of features such as price momentum and price volatility were used to predict the closing price of the NASDAQ 100 [21]. The authors used daily stock prices to compute both the price momentum and volatility for each stock and the overall sector. The goal of the analysis was to forecast the closing price in the next coming days and decide whether the price has risen or decreased relative to how it was on a trading day. The SVM model with RBF Kernel estimated the closing price with the smallest accuracy for a short-term prediction. The model also predicted stock values with the highest accuracy between 55% and 60% for the long-term prediction. According to the authors, the short-term forecasts supported the Efficient Market Hypothesis (EMH).

The study by [17] applied SVR to predict the up-to-the-minute and daily closing prices of the Brazilian small and large capitalisation companies. The study employed the SVR technique to create a regression model that was responsible for forecasting the prices of assets. A random walk model was also implemented to compare the results. The dataset used was obtained from the Brazilian, American, and Chinese stocks. The day-to-day stock prices covered a 15 years period. The data was partitioned into 70% training and 30% testing for both the daily and minute prices. They have predicted the closing price for the next 10 minutes of each trading session using the minute data. The MAPE and RMSE were accuracy measures used in their study. The SVR model with a linear kernel displayed more predictive power over the random walk model, which indicated that the RMSE and Mean Absolute Error (MAE) values for the linear SVR kernel model are lower.

2.2.3 Random Forest

The authors [18] worked on predicting the movement of stock for the Tehran Stock Exchange index. A total of 3 classification algorithms namely: Naive Bayesian classifier, RF, and decision tree were compared. Daily historical data acquired from Tehran Stock Exchange Technology Management Co. was used to implement the models. The data was from 17 April 2007 to 18 March 2012. The data was made up of 1184 samples. About 80% of the entire dataset was utilised as the training set, while 20% was reserved for testing purposes. Technical indicators such as Stochastic K%, simple 10 day Moving Average, Relative Strength Index, Weighted 10 day Moving Average, etc. were used as the input. The results obtained indicated that the decision tree model outperformed random forest and Naive Bayesian classifier models with an accuracy of 80.08%. The random forest model was able to perform with an accuracy of 78.81%.

A research conducted by [22] focused on forecasting the stock prices using RF technique. The authors used the CROBEX index data and information from different sectors of companies listed on the Zagreb Stock Exchange. The authors predicted the stock prices for 5 and 10 days into the future by implementing the RF model. Technical indicators such as Stochastic %D, Moving Average Convergence Divergence, and 5 days and 10 days disparity, etc. were forwarded to the model as inputs. Historical information used included the period between January 2008 and December 2013 and features used were the lowest, opening, highest, and closing prices. These features were used to compute the technical indicators. The classification accuracy and F-measure were used as accuracy measures. The RF model was evaluated by stratified 10-fold cross-validation. The model predicted stock prices 5 days ahead with an accuracy of 76.5% and the weighted average F-measure of 0.763. The model performed with an accuracy of 80.8% and F-measure of 0.080 for 10 day ahead predictions. They concluded that RF can be employed for forecasting stock prices.

It has been highlighted by [5] that there is a strong relationship between market risk and forecasting errors. The authors proposed RF and XGBoosted tree approaches to predict stock prices. The data for software, electronic, automobile, and sports companies were used to fit the models. The data was made up of the date, closing price, volume, and other variables used to derive stock indicators such as Stochastic

Oscillator, Relative Strength Index, etc. The models were trained for 3, 5, 10, 15, 60, and 90 trading days. They found that XGBoost and RF can be used to predict whether the stock prices will increase or decrease based on previous information.

2.2.4 Recurrent Neural Network

Deep learning architectures which include RNN, Long Short Term Memory (LSTM), and CNN were previously applied by [35] to evaluate their performance on the short-term stock price prediction of the companies listed in the NSE. The dataset was made up of the minute wise stock price recorded from July 2014 to June 2015 for 1721 companies listed in the NSE. The data consist of features such as timestamp, the volume of stock traded per minute, transaction id, stock prices, and day stamp. A 100 minutes window size with an overlap of 90 minutes was used to predict the values 10 minutes ahead. The data of the Infosys stock price for a period from July to October 2014 was used as the train data while October 2014 data served as the testing data. The networks were trained for 1000 epochs and RMSE was used as the accuracy measure. A model with the smallest value of RMSE was selected for final predictions. The CNN system was found to perform better compared to LSTM and standard RNN. The authors concluded that CNN yielded better predictions as it does not rely on the present window for prediction.

The purpose of the research presented in [32] was to explore the performance of LSTM and standard RNN techniques to forecast the indices of a share market. The NIFTY 50 index data was used for training and validating the models. They focused on a 5 year period data starting from January 2011 to December 2016. The variables: date, volume, opening, closing, highest, and lowest prices were included in the dataset. The authors implemented an LSTM model that was made up of a sequential input layer and 2 LSTM layers. The activation functions ReLU and linear were used. The contrast between the target values produced by the output layer and the true values of the target was used to compute the error. They trained their model using 250 and 500 epochs including a window size of 22 trading days. The RMSprop was used as the optimiser. The model produced better predictions when trained for 500 epochs. The LSTM performed with an RMSE of 0.013. Keras was used for the implementation of their model.

A study carried out by [3] focused on estimating the stock prices of the S&P 500. The aim was to compare 2 types of LSTM architectures: bidirectional and stacked LSTM. They forecasted the stock prices for both short and long-term periods. Secondary data available on the Yahoo finance website was used. The selected data was recorded for the period 01 January 2010 to 30 November 2017. During short-term prediction, they forecasted the closing values for the next day using a window size of 10 that was equivalent to 10 trading days. Moreover, for the long-term prediction, they estimated the stock values for the next 30 days. The dataset was partitioned into 80% for training and 20% for testing. To achieve their goal, a neural network structure was designed with a different number of neurons 4, 8, 16, and 32. The data was trained using a different number of epochs in the range 1 to 10. The authors used MAE, RMSE, and correlation coefficient (R^2) as accuracy measures. They deduced that both the techniques were able to accurately forecast the stock prices for the long and short-term. Bidirectional LSTM performed with the highest accuracy and produced a better convergence for short-term than a long-term prediction.

2.3 Conclusion

In this research, we investigate the performance of both the Gated Recurrent Unit (GRU) and LSTM on the price prediction of the JSE All-Share index data. The features used are opening, highest, lowest, and volume of shares traded for 10 years period (June 2008 to July 2018). These variables are chosen because they present what was happening in a stock market for given trading days. A thorough analysis of past research led us through the appropriate techniques suitable for estimating the future values of the stock market. We observed from [1], [31], [6], and others that stock prices can be predicted with the highest accuracy by using ANNs. Other researchers employed SVM, RF, RNN, LSTM, and GRU while others used traditional methods such as ARIMA, AR, regression, random walk, etc. This study evaluates the performance of both LSTM and GRU against the VAR model on the stock price prediction and compares the results to the findings of [3] and [35].

Chapter 3

Research Methodology

This chapter highlights the approaches used to answer the research questions provided in Section 1.3.2. Section 3.1 describes the data used for forecasting. The data pre-processing steps performed before implementing the models are also described. Lastly, the methods and forecast accuracy metrics are discussed. Figure 3.1 shows the proposed system. The data is pre-processed and divided into training and testing sets. The training data is used to fit the models and testing data is used to evaluate the performance of the models.

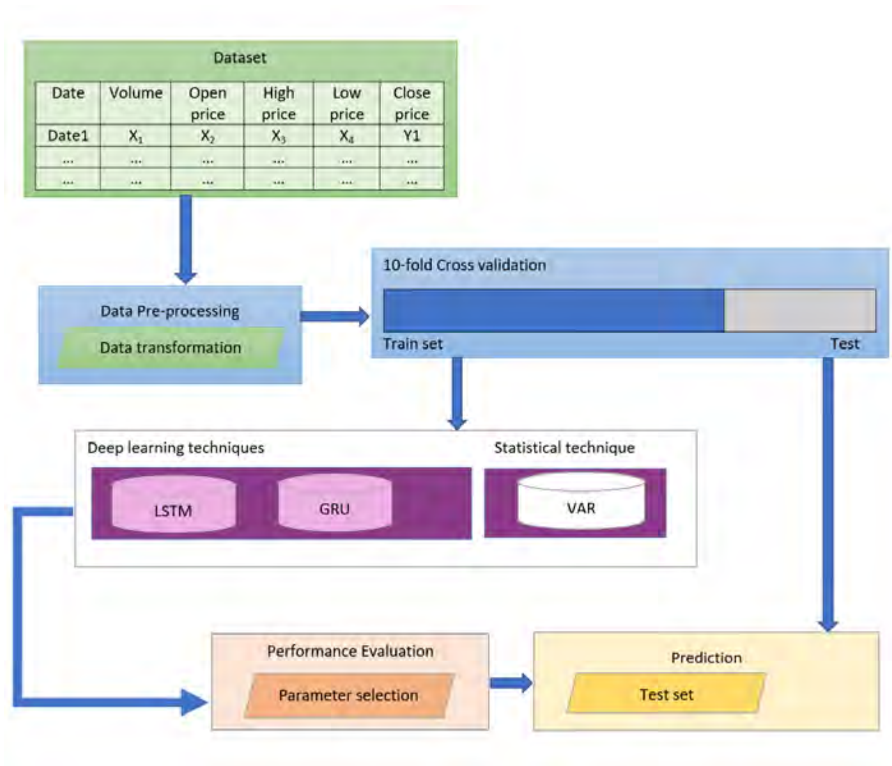


FIGURE 3.1: Proposed system

3.1 Research design

This study aimed to predict the closing price of the Johannesburg Stock Exchange (JSE) All-Share Index using both traditional and deep learning techniques. Through an in-depth analysis of literature in Chapter 2, it was found that numerous studies have successfully predicted the closing price of stock markets using machine learning approaches which include Artificial Neural Network (ANN), Recurrent Neural Network (RNN), Support Vector Regression (SVR), Multi-layer Perceptron (MLP), Long Short Term Memory (LSTM), and Gated Recurrent Unit (GRU). It was proven that statistical techniques such as the Autoregressive Integrated Moving Average (ARIMA) method can also be employed to predict the stock values. Most researches argued that machine learning algorithms have the ability to accurately predict stock prices compared to statistical techniques [13] and [24]. This study applies both GRU and LSTM methods to meet the stated objectives. The Vector Autoregressive (VAR) model is used to benchmark these 2 deep learning techniques. The experiments are conducted using Python in a Jupyter environment.

3.2 Data

This study used the JSE All-Share index data obtained from the Iress Expert. The data covered a 10 year period from 20 June 2008 to 21 July 2018. It was made up of 2500 observations with a total of 12 features. The features included in the data were stock prices such as the lowest, closing, highest, opening, Total Return Index (TRI) closing, and volume. Other variables such as interest yield, total distribution yield, capital payment yield, Dividend Yield (DY), Price to Earnings (P/E), and date were also included. Figure 3.2 shows the first few records of the data.

Date	High	Low	Open	Close	TRI Close	Volume	Interest Yield	Capital Payment Yield	Total Distribution yield	EY	P/E	DY
2008/06/20	31295.46	30577.17	31295.46	30580.63	3277.96	220592198	0.00	0.00	2.53	6.48	15.43	2.53
2008/06/23	30893.41	30370.91	30589.36	30473.44	3267.18	166902116	0.00	0.00	2.54	6.50	15.39	2.54
2008/06/24	30784.39	30121.44	30473.44	30222.88	3240.32	203936859	0.00	0.00	2.56	6.55	15.27	2.56
2008/06/25	30222.88	29882.91	30222.88	29964.55	3212.62	250539073	0.00	0.00	2.58	6.62	15.10	2.58
2008/06/26	30249.56	29750.97	29964.55	29988.12	3215.15	224698362	0.00	0.00	2.58	6.62	15.11	2.58
2008/06/27	30375.47	29613.68	29988.12	30375.47	3256.68	234933104	0.00	0.00	2.54	6.53	15.31	2.54
2008/06/30	30757.09	30375.47	30375.47	30413.43	3262.45	193864136	0.00	0.00	2.55	6.53	15.33	2.55
2008/07/01	30507.10	29859.85	30413.43	30003.76	3218.50	217738796	0.00	0.00	2.59	6.62	15.12	2.59
2008/07/02	30136.42	29279.15	30003.76	29303.69	3143.40	194872216	0.00	0.00	2.65	6.77	14.76	2.65
2008/07/03	29303.69	28133.58	29303.69	28392.19	3045.63	252486420	0.00	0.00	2.74	6.99	14.30	2.74
2008/07/04	28616.06	28045.13	28392.19	28172.28	3022.04	169080800	0.00	0.00	2.76	7.05	14.19	2.76

FIGURE 3.2: A sample of the JSE All-Share index data

3.2.1 Variables

This study aimed to forecast the closing price of the JSE All-Share index. The variables used in this study were selected based on the literature review. Previous researchers used stock prices such as the highest, lowest, and opening values of shares traded daily as well as the volume of shares sold on that day as inputs. Other studies used technical indicators which included On Balance Volume-Movement, Relative Strength Index, Stochastic Oscillator, Weighted Moving Average, Price Momentum Oscillator, etc. We predicted the closing price since it reflects all the activities of the index in a day. It is known to be the last price at which a stock is exchanged in a financial market on a specific day. The inputs were the lowest price, opening price, volume, and the highest price of shares exchanged on a given day. These prices provide information about the day's move in cents and as percentages. The opening price is the first value at which any listed stock is sold on a specific day. The highest and lowest prices are the maximum and minimum prices of stock in a given trading day. The volume is the number of stocks traded daily.

3.3 Traditional methods

This research focused on a multivariate time series, hence, a VAR model was implemented. The model was constructed using the Statsmodels library provided in the Python package.

3.3.1 Data pre-processing

Data pre-processing is the process of cleaning and transforming the raw data into an understandable format. Data pre-processing steps in time series include making the data stationary by removing trend and seasonal components. This can be achieved by taking the log-transformation of the series or differencing. The variables to be included in the model are visualised to identify any existing trends. They are shown in Figure 3.3 and is observed that the stock prices have a similar increasing trend. The volume seems to fluctuate at a constant mean and variance. The results shown in the graph are proven further by performing a statistical test in Section 3.3.2.

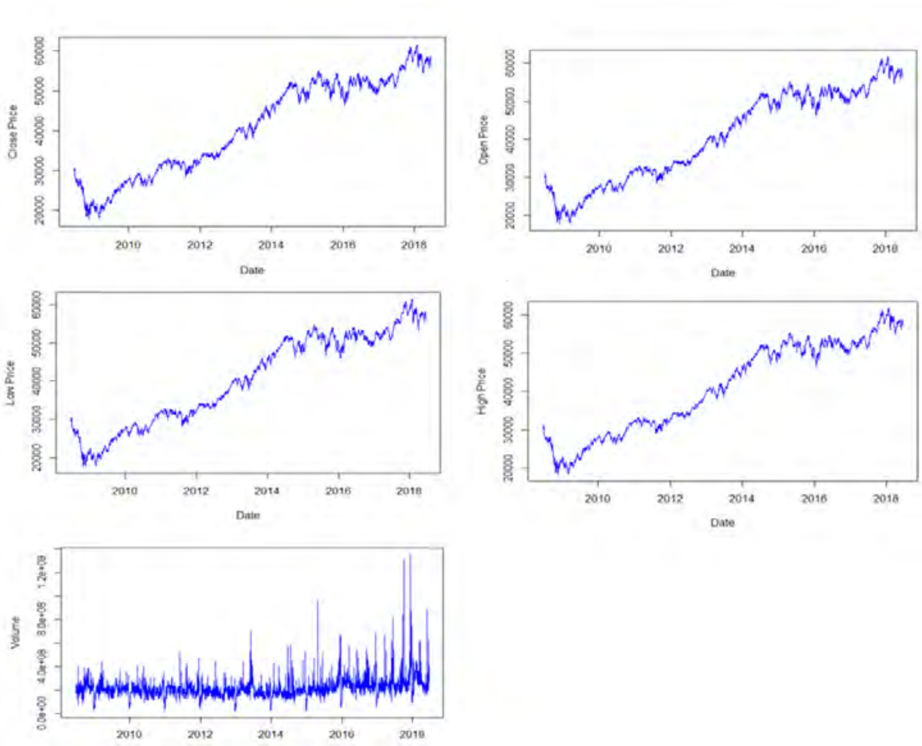


FIGURE 3.3: Time series plots of the JSE All-Share index data

3.3.2 Testing for stationarity

A series needs to be stationary before implementing a VAR model. A series is said to be stationary if its variance, mean, and covariance does not change over time. The unit-root tests such as Augmented Dickey-Fuller (ADF), Philip-Perron, and Kwiatkowski–Phillips–Schmidt–Shin (KPSS) are used to test for stationarity. The ADF is the most commonly used test, hence, it was applied in this study. In the ADF test, the null hypothesis states that the data is not stationary. A time series is stationary if the p-value is less than the significance level of 5%. A p-value greater than 5% indicates that the process is non-stationary and therefore the series requires transformation.

Table 3.1 shows the results from the ADF test. It is observed that the p-value of all the variables are above the significance level of 5% except for the volume. These indicate that the closing, opening, lowest, and highest prices are not stationary, their variance, mean, and covariance are changing over time, therefore, the data requires differencing.

TABLE 3.1: Results from ADF test

Variables	P-value
Volume	0.0000
Close Price	0.9115
Open Price	0.9128
Low Price	0.9140

3.3.3 Differencing

Differencing is the process of taking the difference between consecutive values of the series. Figure 3.3 and Table 3.1 show that the data is non-stationary hence differencing is required. To achieve stationarity, the data is log-transformed and differenced. Transforming the data using logarithms stabilises the variance while differencing stabilises the mean of the series by eliminating trends and seasonality in a process. The first difference is given by the change in consecutive observations in the original time series and it is computed using Equation 3.1. The results of the ADF test for a differenced time series are displayed in Table 3.2. The p-value for all the series is less than 5%, hence, the processes are stationary.

$$\Delta x = x_t - x_{t-1} \tag{3.1}$$

where x denotes the observations at time t.

TABLE 3.2: Results from ADF test for differenced time series

Variables	P-value
Close Price	0.000
Open Price	0.000
Low Price	0.000
High Price	0.000

3.3.4 Granger Causality

It is necessary to determine if there is a bidirectional relationship between the variables. This is achieved by performing the Granger causality test. Granger causality tests the null hypothesis that the past values of the series do not cause other series. The null hypothesis is rejected if the p-value is below 5% level of significance, and the maximum number of lags used to test for causality was 12 [8]. The hypothesis to be tested are listed below.

- H_{01}: The closing price does not cause the volume.
- H_{02}: The closing price does not cause the opening price.
- H_{03}: The closing price does not cause the highest price.
- H_{04}: The closing price does not cause the lowest price.

Table 3.3 shows the results for the Granger causality test. The p-value for all the hypotheses is less than the significance level of 5% except for the first hypothesis H_{01}. We fail to reject H_{01} since the p-value is beyond the significance level indicating that there is no causal relationship between the volume and the closing price. The opening, highest, and lowest prices are helpful for predicting the closing price of the stock market.

TABLE 3.3: Results from ADF test for differenced time series

Variables	**P-value**
Close and Volume	0.8967
Close and Open	0.0111
Close and High	0.0007
Close and Low	0.0208

3.3.5 Data Split

The main goal of this study was to perform a short-term prediction of the closing price of the JSE All-Share index, using data over a period of 10 years. We predicted the closing price for 5, 10, and 15 days ahead. The selected days are motivated by the research carried out by [22]. The train-test split depends on the number of days to be predicted and is listed below:

- The last 5 observations from data, from 15 June 2018 to 21 June 2018 were used as the testing data for 5 days predictions. The remaining samples were used as training data.

- The last 10 observations from data, from 08 June 2018 to 21 June 2018 were used as the testing data for 10 days predictions. The remaining samples were used as training data.

- The last 15 observations from data, from 01 June 2018 to 21 June 2018 were used as the testing data for 15 days predictions. The remaining samples were used as training data.

The values estimated by the model were compared to the testing data. The accuracy of the predictions was assessed using the Mean Absolute Percentage Error (MAPE), and the coefficient of determination (R^2).

3.3.6 Finding the optimal order of (P)

The order P of a VAR model represents the number of past information to be used as predictors in a time series, hence, selecting the optimal lag length is important. The use of a smaller lag length can result in autocorrelated residuals while many lags cause over-fitting of the model [8]. The most commonly known criteria for selecting the lag length are Akaike Information Criterion (AIC), Bayesian Information Criterion (BIC), Schwarz Criterion (SC), Hannan–Quinn information criterion (HQ), and Final Prediction Error (FPE). In literature, the order P was chosen based on the smaller value of the AIC. According to [10], selecting the number of lags based on the criterion does not always result in a model that produces best forecasts. In this research, different lag lengths were tested to overcome the problem.

3.3.7 Vector Autoregressive

For this project, the volume, closing, opening, highest, and lowest prices of stock were included in a VAR model. This indicates that when predicting the stock price, the price value in a system is represented by multiple linear regression equations, each variable explaining the stock prices and volume. The closing price is explained by the values of its lags and lags of other prices and volume included in the system. For example, suppose we want to determine the stock values for today at time t, these values will be explained by the stock prices at time $t-1$ which is yesterday and by stock prices at time $t-2$ which is the day before yesterday, etc.

3.3.8 Model diagnostic

Model diagnostics is the process of determining if the model assumptions hold. The model assumptions include having a normally distributed residuals with a mean of 0 and a constant variance. Residuals are the difference between the actual values and the values predicted by the model. We used the Durbin Watson (DW) statistic to test for serial correlation in the residual. The null hypothesis states that there is no serial correlation in the residuals. The test statistics for the DW is computed by Equation 3.2. The value of the test statistic lies between 0 and 4. The value of 2 implies no autocorrelation, values less than 2 indicate a positive autocorrelation while values in a range of 2 to 4 show a negative correlation. The test statistic values between 1.5 and 2.5 are regarded as normal [14]. The histogram was also plotted to verify the distribution of the residuals.

$$DW = \frac{\Sigma_{t=2}^{T}(e_t - e_{t-1})^2}{\Sigma_{t=1}^{T} e_t^2} \tag{3.2}$$

where e_t are residuals.

3.3.9 Forecasting

Forecasting is the final step performed after determining the goodness of fit. In this study, the main objective was to predict the future closing price of the South African stock market. Predictions are visualised using the plots. Lastly, the performance of the VAR model was determined using the accuracy metrics MAPE and R^2. The value of R^2 lies between 0 and 1, it explains how well the data fits the model. The highest value of R^2 indicates a better fit for the model.

3.4 Deep learning models (LSTM and GRU)

To achieve our goal, both GRU and LSTM models were constructed using Keras API with a TensorFlow backend. The implementation of our experiment was conducted using the Python programming language.

3.4.1 Data Pre-processing

The volume of shares traded daily and the stock prices vary in range. The performance of machine learning models is improved by transforming the data with different scales. Data normalisation can be useful and is required when implementing machine learning models for time series data. In this research, the data was transformed using the Scikit-learn package called the min-max normalisation. Figure 3.4 shows a trend of the transformed closing price. The closing price of the All-Share index seems to increase over time.

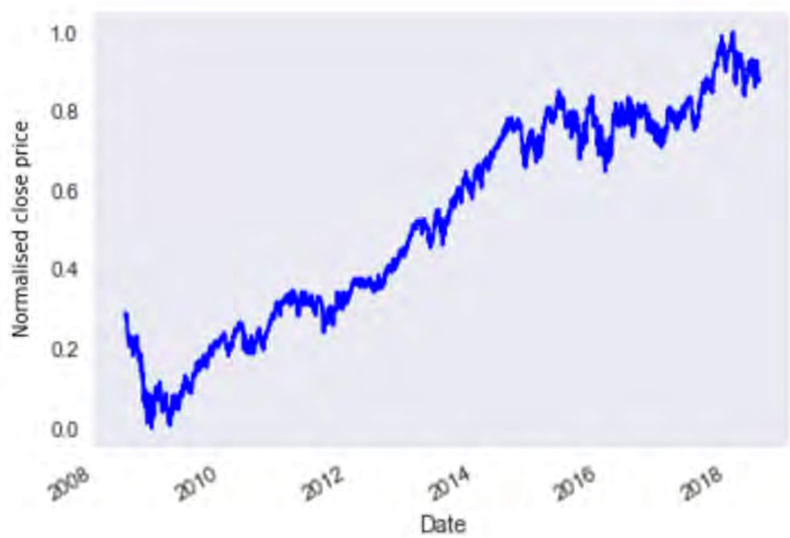


FIGURE 3.4: Normalised index values of the JSE All-Share

The All-Share data consisted of 5 variables and 2500 observations. The input signals used for short-term predictions of the South African market were the volume, highest, opening, and lowest prices of the stock market. The data was shifted for 5 days and shift in data replaced the last 5 observations starting from 15 June 2018 to 21 June 2018 with null values. The null values for the shifted period are removed from the dataset. The final data used contained 2495 samples because of the shift in periods. The data was converted to a 2-dimensional array of the size (2495, 5) including the outcome variable. A trading day was represented by 1×4 where 4 denotes the number of features to be included in the predictions. LSTM and GRU cannot be trained on a sequence with large observations. As a result, a random batch of 128 sequences with a sequence length of 60 observations was generated. A 2-dimensional array was reshaped into a 3-dimensional array of (128, 60, 4) and (128, 60, 1) where 4 represents the number of input signals and 1 the target signal.

3.4.2 Data Split

There is a need to validate the model after the training process. This can be achieved by cross-validation (CV). A CV is a process of testing the effectiveness of machine learning models. It can be inferred that the model is under-fitting, over-fitting, or well-generalised based on its performance on the unseen data. The most commonly used method for CV is the train-test split and k-folds split technique. In the train-test split technique, the data is divided into testing and training sets. The testing set is used for validation purpose and there is a higher chance of bias if the data is not enough or limited. When there is enough data and both testing and training sets follow the same distribution, the train-test split method is acceptable. There is no bias with k-fold since it ensures that each observation has a chance of appearing in both the training and testing sets.

Studies conducted previously guided us in allocating training and testing samples when performing experiments. Most researchers used the train-test split based on the rules of (90% vs. 10%), (70% vs. 30%), (80% vs. 20%), etc. [17] and [18]. The authors [22], [2], and [9] evaluated their models using a 10-fold CV. For this study, 10-fold CV was used to assess the performance of the deep learning models and to minimise the risk of over-fitting. A total of 10 experiments were performed, 9 folds were used as the training data in the first experiment while 1 fold was used as the testing data. In the second experiment, a different set of 9 folds was used to train the network and a different fold was used as the test set. The same process was repeated for all the experiments. Figure 3.5 shows a representation of the 10-fold CV. A CV score was imported from the Scikit-learn library to keep track of errors generated during the training process.

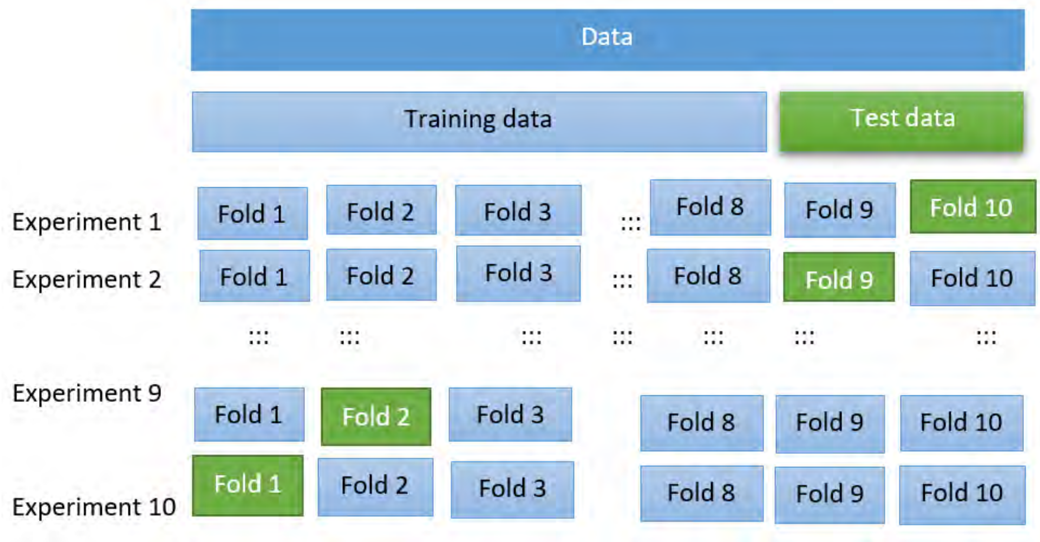


FIGURE 3.5: A 10-fold cross-validation

3.4.3 Model training

A set of hyper-parameters used to train both the GRU and LSTM models was selected based on the literature review. In the literature, researchers trained models using a different number of epochs, activation functions, optimisers, and initialisers. Most studies trained their models based on the values lying between 10 to 1000 epochs. The most commonly used activation functions are Relu, linear, and sigmoid. The optimisers such as Adam and RMSprop are widely used with the learning rate of 0.1, 0.001, or 0.0001 for machine learning models [3], [32], [35]. In the literature, deep learning models performed with the RMSE of 0.014, 0.01236, and 0.00859 and MAPE between 2% to 8% [32] and [34]. A summary of the hyper-parameters used in this study is presented in Table 3.4.

TABLE 3.4: Hyper-parameters

Hyper-parameters	Values
LSTM and GRU cells	512
Activation Function	Linear
Dropout	0.2
Initialiser	RandomUniform(minval=-0.05, maxval=0.05)
Optimiser	Adam
Learning rate	0.001
Loss	Root Mean Squared Error
Metric	Mean Absolute Error
Epochs	150, 50

3.5 Evaluation Metrics

The RMSE and MAPE are the most commonly used metrics for evaluating the accuracy of the models. RMSE measures the variation between the observed values and the values forecasted by the model and it is frequently used for regression analysis. MAPE is a measure of the variation among 2 continuous variables. In this study, we employ MAPE to evaluate the performance of our models. The values of RMSE and MAPE always lies between 0 and 1.

The equations for the RMSE and MAPE are given below:

$$RMSE = \sqrt{\frac{1}{N}\Sigma_{i=1}^{n}(\bar{y} - \hat{y})^2} \tag{3.3}$$

where $\bar{y}$ is the mean of all the output data and $\hat{y_t}$ is the predicted output values.

$$\text{MAPE} = \frac{1}{n}\sum_{t=1}^{n}|y_t - \hat{y}_t| \times 100 \tag{3.4}$$

where y_t is the true value of the output and $\hat{y_t}$ is the predicted output values.

3.5.1 Training errors

The networks were trained using hyper-parameters listed in Table 3.4. Table 3.5 displays the errors obtained on the training set. GRU model performed a short-term forecasting with an average error 9.75% maximum while LSTM produced forecasts

TABLE 3.5: MAPE values for LSTM and GRU per epochs on the training data

	150 epochs		50 epochs	
Folds	**GRU**	**LSTM**	**GRU**	**LSTM**
1	9.841%	9.611%	9.357%	8.231%
2	10.354%	9.307%	9.862%	9.965%
3	9.525%	8.855%	8.768%	7.317%
4	9.915%	10.587%	9.115%	8.041%
5	9.406%	9.937%	10.496%	8.181%
6	9.668%	9.556%	8.384%	9.591%
7	8.471%	9.305%	8.388%	8.743%
8	8.950%	9.666%	10.213%	8.560%
9	11.26%	10.168%	10.466%	9.279%
10	10.126%	9.370%	9.411%	8.912%
Average	9.75%	9.636%	9.446%	8.682%

at the maximum error 9.636% on the training data. The performance of the models rely on parameters used. The error values changes as we change the number of epochs. The error values for the training data was compared to the error values based on the testing data in Chapter 4. A comparison of the errors helps to determine if the models over-fit or not.

3.6 Conclusion

In this study, we focused on the prediction of the closing price of the JSE All-share index using the volume, highest, lowest, and opening prices as the input. The models were implemented based on a 10 years period data. We constructed 2 deep learning models, LSTM and GRU, and compared their results in terms of MAPE defined in Section 3.5 above. The VAR model was also used as a reference model. The results of the study might be helpful because a good system that predicts the stock values might lead to financial gain. The models could direct stock market investors to make profitable decisions regarding investments and the results might also contribute to literature since there are fewer studies conducted on the South African market.

Chapter 4

Results and Discussion

This chapter presents the analysis and discussion of findings based on the baseline model Vector Autoregressive (VAR) and deep learning models Long Short Term Memory (LSTM) and Gated Recurrent Unit (GRU). The results are presented in both tabular and graphs. Section 4.1 discusses the outcomes from the baseline model while Section 4.2 provides a discussion of findings for the deep learning models.

4.1 Traditional models

The main goal of the study was to predict the closing price of the Johannesburg Stock Exchange (JSE) All-Share index for the short-term. The Akaike Information Criterion (AIC) was used to select the number of lags for a VAR model. The AIC suggested that we build a VAR model of order 28 since the minimum value was observed at lag 28. The results are presented in Appendix A. As discussed in Section 3.3.6 of the methodology, [10] mentioned that the criterion used to select length lag does not always result in a model that produces the best results. The same problem was encountered in this study. A VAR(28) was unable to forecast the future closing price of the index. We repeatedly fitted a VAR model using various lag lengths 12, 30, 40, 52, 60, 72, 83, 84, and 86. A VAR(84) produced better forecasts and was chosen for the short-term prediction of the All-Share index values. A summary of VAR(84) results is illustrated in Appendix B. The rows show each response variable represented as the lags of the variable itself and other variables included in the system. The validity of the model is discussed in the next sections.

4.1.1 Model Diagnostics

This section provides a discussion of the goodness of fit for the VAR(84) model. The behaviour of the residuals was examined to determine the adequacy of the model. We have used the Durbin Watson (DW) statistic to test for serial correlation in the residuals. The results for the DW statistic are shown in Table 4.1. The values of the statistic are closer to 2 for all the variables. It was highlighted in Section 3.3.8 that the values of the statistic less than 1 and more than 3 are a cause for concern, however, values between 1.5 and 2.5 indicate that the residuals are uncorrelated and fairly normal. The results were proven further by visualising the distribution of the residuals and the autocorrelation plots.

TABLE 4.1: Durbin Watson Test

Variables	**VAR(84)**
Volume	2.004
Low	2.0
High	1.994
Open	1.995
Close	1.999

Figure 4.1 shows the Kernel Density Estimate (KDE) plot, autocorrelation plot (ACF), and the residuals plot from VAR(84) fitted to the JSE All-Share index data. The KDE and histogram have a bell shape which indicates normality, however, it seems slightly skewed. The residual plot shows the movements at a constant variance and a mean of 0. The autocorrelation plot does not show any significant spikes and there is no pattern in the number of lags, hence, the results support the DW statistic of no serial correlation in the residuals. The assumptions of the model adequacy are met, hence, a VAR(84) is a better fit for predicting the future closing price of the All-Share data.

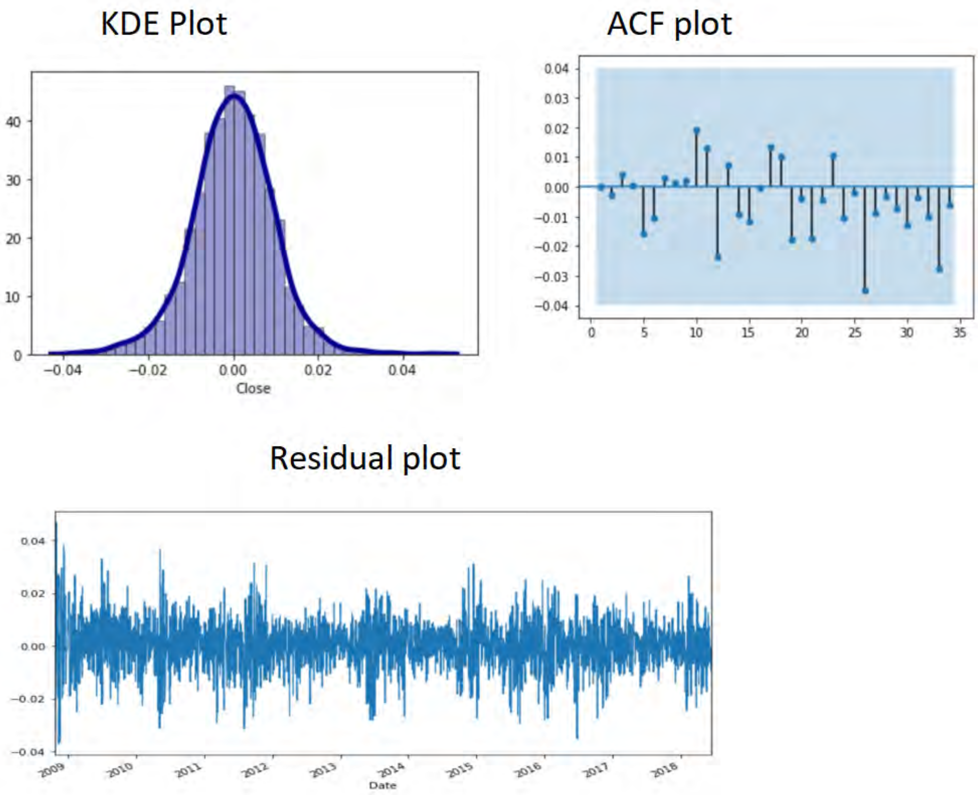


FIGURE 4.1: The diagnostic plot for VAR(84)

4.1.2 Forecasting

TABLE 4.2: The actual versus 5 days predictions of index

Date	Actual index values	Predicted index values	MAPE
2018-06-15	57660.50	58239.73	1.005%
2018-06-18	57236.84	58086.29	1.484%
2018-06-19	56253.30	57710.32	2.590%
2018-06-20	56651.65	57727.40	1.899%
2018-06-21	56234.43	57494.25	2.240%

The performance of a VAR model depends on the lag length and selected variables. A fitted VAR model of order 84 was used to forecast the future closing price of the JSE All-Share index for the short-term. We predicted the stock prices for 5, 10, and 15 days forward. The number of days to be predicted are selected based on the study conducted by [22]. The actual values and 5 days predictions are presented in Table 4.2. To validate our model, the accuracy metric Mean Absolute Percentage Error (MAPE) was used. For each trading day, the model predicted the stock price with the error between 1.005% and 2.590%.

Figure 4.2 shows the movement of true values of stocks and forecasts over time. When the actual values of stock prices increase, predictions also increase. A VAR system predicted the stock prices with an average error of 1.844% presented in Table 4.3. The coefficient of determination (R^2) of 0.964 shows that 96% of the data fit a VAR model. The study by [19] concluded that VAR models perform best for short-term forecasting. The findings of the current study support the literature.

TABLE 4.3: Accuracy for 5 days predictions

Forecast Accuracy	VAR(84)
Average MAPE	1.844%
Correlation Coefficient	0.964

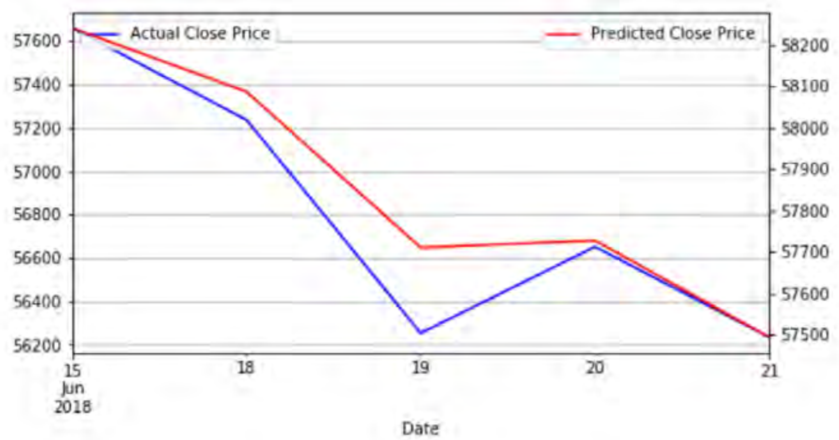


FIGURE 4.2: A 5 days predictions of VAR(84)

The performance of the VAR(84) model is investigated further by forecasting 10 and 15 days ahead. The predictions are plotted to identify the movement of the predicted daily stock prices. The results for 10 days forecasts are reported in Table 4.4. The error rate for each day's predictions increases as the number of days to be predicted increases. The daily errors for 10 days forecast lie between 0.371% to 4.343%.

TABLE 4.4: The Actual versus 10 days predictions of the index

Date	Actual index values	Predicted index values	MAPE
2018-06-08	58223.72	58439.70	0.371%
2018-06-11	58146.09	58395.06	0.428%
2018-06-12	58207.82	58477.52	0.463%
2018-06-13	58437.22	58927.71	0.839%
2018-06-14	58495.67	59675.20	2.016%
2018-06-15	57660.50	59315.23	2.870%
2018-06-18	57236.84	59095.58	3.247%
2018-06-19	56253.30	58696.51	4.343%
2018-06-20	56651.65	58526.75	3.310%
2018-06-21	56234.43	58276.94	3.632%

Table 4.5 shows the ability of VAR(84) in predicting stock prices 10 days forward. The model successfully predicted the stock values with an average error of 2.152%. The value of R^2 indicates that only 35% of the data fit a VAR model. A VAR(84) performs better for 5 days forecasts compared to 10 days predictions. The results might be due to lag length used for predicting stock prices 10 days into the future since VAR systems depend on the number of lags selected. Figure 4.3 displays the values predicted by the model against the true values. There is an overlap between the predictions and actual values in the last 2 trading days. The same trend was observed for 5 days forecasts.

TABLE 4.5: Accuracy for 10 days forecast

Forecast Accuracy	VAR(84)
Average MAPE	2.152%
Correlation Coefficient	0.345

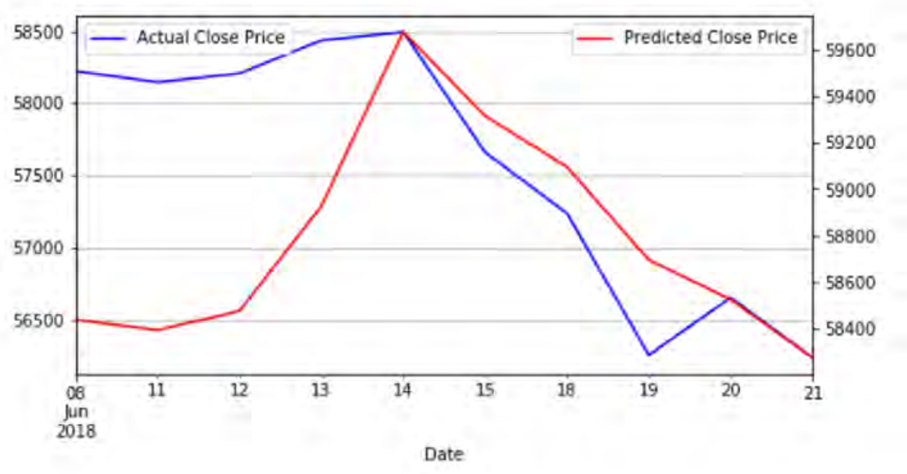


FIGURE 4.3: A 10 days predictions of VAR(84)

Lastly, we predicted the closing price of the All-Share index for 15 days into the future and the results are summarised in Table 4.6. The error values for each day forecast are lower, which indicates that a VAR(84) can forecast the stock values for 15 days. The errors for the daily predictions lie between 0.631% and 2.818%. The difference between 15 days forecasts and true values is lower compared to 10 and 5 days forecasts.

It is highlighted in Table 4.7 that a VAR(84) predicted the stock prices with an error of 1.427% and the R^2 of 0.127 indicates that only 13% of the data fitted the VAR model. When the number of days to be predicted increases, the R^2 value decreases showing that only a small percentage of the entire data fit the model. The results implied that a VAR of order 84 is a better fit to forecast stock prices for 5 trading days according to the R^2 value. The MAPE suggested that a VAR(84) can successfully predict the closing price for 5, 10, and 15 days into the future. The model performed better for 15 days predictions in terms of the MAPE. Furthermore, the model produced better forecasts for 5 days compared to 10 days. Figure 4.4 displays the movement of stock prices against 15 days forecasts.

TABLE 4.6: The Actual versus 15 days predictions of index

Date	Actual index values	Predicted index values	MAPE
2018-06-01	57282.14	56860.12	0.737%
2018-06-04	57870.89	56829.86	1.799%
2018-06-05	57779.11	56778.53	1.732%
2018-06-06	58081.86	57440.78	1.104%
2018-06-07	58391.64	57425.49	1.655%
2018-06-08	58223.72	57473.63	1.288%
2018-06-11	58146.09	57575.10	0.982%
2018-06-12	58207.82	57669.60	0.925%
2018-06-13	58437.22	58055.56	0.653%
2018-06-14	58495.67	58865.02	0.631%
2018-06-15	57660.50	58492.03	1.442%
2018-06-18	57236.84	58271.59	1.808%
2018-06-19	56253.30	57838.60	2.818%
2018-06-20	56651.65	57693.14	1.838%
2018-06-21	56234.43	57356.43	1.995%

TABLE 4.7: Accuracy for 15 days forecast

Forecast Accuracy	VAR(84)
Average MAPE	1.427%
Correlation of Coefficient	0.1272

FIGURE 4.4: 15 days predictions of VAR(84)

4.2 Deep Learning Models

The aim of this research was to investigate the performance of deep learning techniques in predicting stock values. We predicted the future closing price for the short-term. The findings of the deep learning architectures used to predict the stock prices are discussed in this section. The models were trained given the same feature setup and hyper-parameters discussed in Chapter 3. The aim was to determine the model that can produce better forecasts. The efficiency of the deep learning models in predicting the stock prices was analysed using the MAPE.

In the literature, most studies trained their models based on the values lying between 10 to 1000 epochs. For this work, the models were trained using 50 and 150 epochs. Hyper-parameters such as the activation functions, optimisers, learning rate, etc, forwarded to the networks were kept constant and are presented in Section 3.4.3. A comparison of the error rate for the deep learning models is shown in Table 4.8. It is revealed that deep learning models rely on the selected hyper-parameters to perform well.

The number of epochs influenced the performance of both GRU and LSTM models. GRU outperformed LSTM with an average MAPE of 9.286% for 150 epochs. The LSTM models forecasted the stock price with an average error of 9.459%. For a smaller number of epochs 50, GRU did not perform better compared to LSTM. It predicted the values with an error of 9.349% while LSTM produced forecasts at an error of 8.931%. It is difficult to conclude that the LSTM model outperforms the GRU or vice versa as we can see that for 150 epochs GRU performs better than LSTM while LSTM performs best compared to GRU for 50 epochs. The average errors obtained on the unseen data in Table 4.8 are closer to the average errors based on the training data presented in Table 3.4. The results indicate that our models are not over-fitting. Both the deep learning models can forecast the All-Share index values into the future. In literature, it is concluded that machine learning models can accurately predict stock price and the same results are observed in this research.

TABLE 4.8: Mean absolute percentage errors for LSTM and GRU per epochs on the test data

	150 epochs		50 epochs	
Folds	**GRU**	**LSTM**	**GRU**	**LSTM**
1	7.143%	7.553%	8.851%	6.842%
2	9.352%	9.831%	10.203%	9.948%
3	10.392%	9.831%	11.366%	6.93%
4	10.121%	7.894%	9.293%	8.747%
5	11.438%	12.128%	11.677%	10.944%
6	6.39%	5.886%	5.846%	9.372%
7	9.81%	11.636%	9.949%	11.093%
8	7.302%	7.18%	7.249%	7.156%
9	9.372%	10.26%	9.015%	8.205%
10	11.854%	12.436%	10.043%	10.069%
Average	9.286%	9.459%	9.349%	8.931%

Figure 4.6 shows the predicted and actual closing stock values of the JSE All-share index. It shows a comparison of the movement of the stock prices per time steps. The top graphs display the actual values against the forecasts for 50 epochs while the bottoms graphs show the estimated values for 150 epochs. The values predicted by the models are closer to the actual values of stocks, indicating that deep learning methods accurately predict the closing price of the index. The studies carried out by [3], [32], and [35] concluded that the deep learning models can be used to predict the closing price for both the short-term and long-term. The findings of this research support the results from the literature.

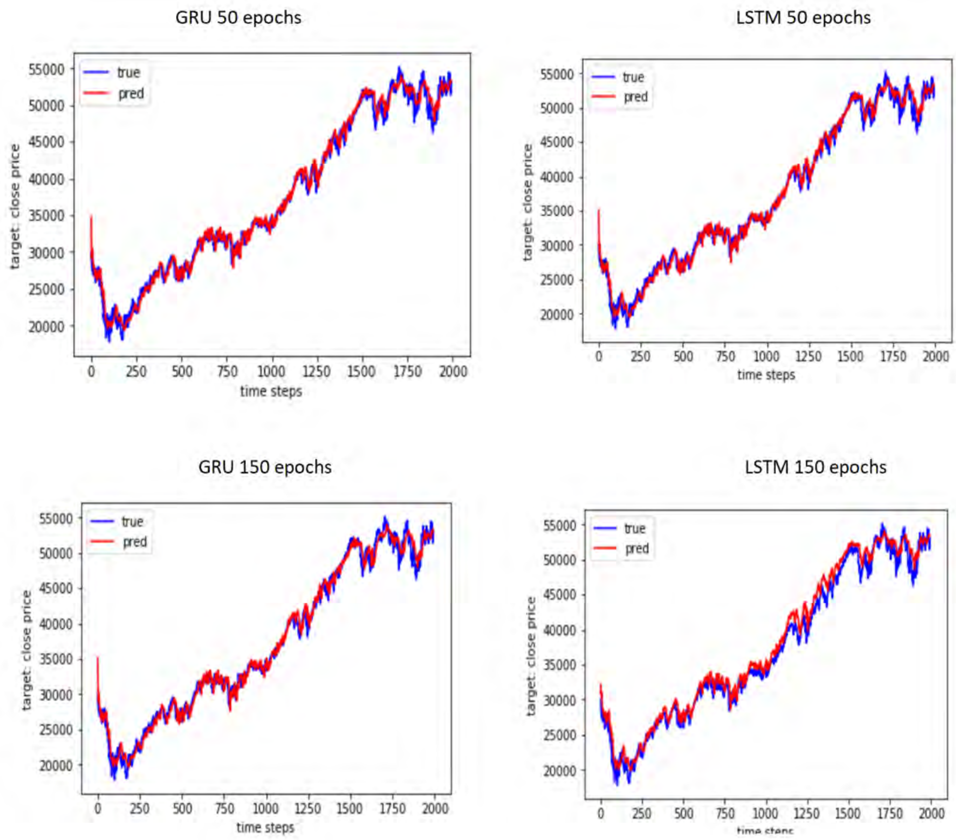


FIGURE 4.5: GRU versus LSTM predictions per epochs

4.3 Conclusion

In this chapter, a discussion of findings for the study was provided. It has been proven that a traditional model VAR and deep learning models LSTM and GRU can predict the closing price of the All-Share index into the future. The results obtained support the findings of the previous studies. Furthermore, it was shown that the performance of all the models depends on the selected parameters. The next chapter presents the conclusion and recommendations of the study.

Chapter 5

Conclusions and Future Work

5.1 Conclusions

The aim of this research was to predict the closing price of the Johannesburg Stock Exchange (JSE) All-Share index. Forecasting of stock prices is considered as a difficult task. Deep learning architectures Long Short Term Memory (LSTM) and Gated Recurrent Unit (GRU) were proposed to solve the complexity of the financial system of the South African market. We predicted the closing price of the JSE data using the highest, volume, opening, and lowest prices of stock. These features were forwarded into a 2 layer GRU and LSTM models. These models were given the same feature setup and hyper-parameters. The closing price was predicted using historical prices recorded for a period of 10 years. We were able to successfully forecast the closing prices of the All-Share index using deep learning techniques. A Vector Autoregressive (VAR) model was used to benchmark these 2 deep learning techniques and the results indicated that the VAR model performed better than both machine learning models in terms of Mean Absolute Percentage Error (MAPE).

The validity of the VAR model for a short-term prediction of the South African market was investigated by predicting stock prices for 5, 10, and 15 days into the future. The model performed better for 15 days forecast compared to 5 and 10 days in terms of the MAPE. The coefficient of correlation R^2 value indicated that the highest percentage of 96% fit the VAR(84) model well for 5 days forecast. Deep learning models were able to predict the stock prices with an average MAPE of 9.349% for GRU and 9.459% for LSTM. Both the deep learning models were outperformed by the VAR model. The VAR model predicted the closing price of the All-Share data with an average MAPE 2.152% maximum.

5.2 Future Work

Recommendations for future work are discussed in this section. An important task that was not carried out in this research is hyper-parameter tuning, hence, it can be applied in the future. The prediction of stock prices can be performed using other stock indicators such as On Balance Volume-Movement, Stochastic Oscillator, Weighted Moving Average, Relative Strength Index, Price Momentum Oscillator, etc. since they are not used in the current study. Furthermore, the performance of other techniques such as Vector Error Correction (VECM), Generalised Autoregressive Conditional Heteroskedasticity (GARCH), Convolutional Neural Network (CNN), and deep belief network can be explored on the JSE data.

Appendix A

Vector Autoregressive lag selection

TABLE A.1: VAR order selection

Lag	AIC	BIC	FPE	HQIC
0	-40.99	-40.97	1.586e-18	-40.98
1	-47.56	-47.49	2.204e-21	-47.54
2	-48.12	-47.99	1.267e-21	-48.07
3	-48.44	-48.25	9.183e-22	-48.37
4	-48.62	-48.37	7.649e-22	-48.53
5	-48.74	-48.44	6.786e-22	-48.63
6	-48.81	-48.44*	6.342e-22	-48.68
7	-48.86	-48.44	6.023e-22	-48.71
8	-48.90	-48.42	5.771e-22	-48.73
9	-48.94	-48.40	5.572e-22	-48.74
10	-48.96	-48.36	5.436e-22	-48.75*
11	-48.97	-48.31	5.381e-22	-48.73
12	-48.98	-48.26	5.336e-22	-48.72
13	-49.00	-48.22	5.245e-22	-48.72
14	-49.01	-48.17	5.212e-22	-48.70
15	-49.02	-48.12	5.158e-22	-48.69
16	-49.03	-48.08	5.085e-22	-48.68
17	-49.05	-48.03	4.998e-22	-48.68
18	-49.06	-47.99	4.941e-22	-48.67
19	-49.07	-47.94	4.904e-22	-48.66
20	-49.07	-47.88	4.891e-22	-48.64

21	-49.07	-47.82	4.894e-22	-48.62
22	-49.08	-47.77	4.826e-22	-48.61
23	-49.09	-47.72	4.789e-22	-48.59
24	-49.09	-47.67	4.771e-22	-48.58
25	-49.09	-47.61	4.773e-22	-48.55
26	-49.09	-47.55	4.778e-22	-48.53
27	-49.09	-47.49	4.797e-22	-48.51
28	-49.10*	-47.44	4.748e-22*	-48.50
29	-49.09	-47.37	4.773e-22	-48.47
30	-49.09	-47.31	4.801e-22	-48.44

Appendix B

VAR model summary

TABLE B.1: Summary of a VAR model

	coefficient	std. error	t-stat	prob
const	0.000626	0.000232	2.697	0.007
L1.Volume	-0.001527	0.000896	-1.704	0.088
L1.Low	0.051951	0.067222	0.773	0.440
L1.High	0.043098	0.072308	0.596	0.551
L1.Open	1.117302	0.562670	1.986	0.047
L1.Close	-0.025215	0.059541	-0.423	0.672
L2.Volume	-0.002527	0.001045	-2.417	0.016
L2.Low	0.017525	0.092476	0.190	0.850
L2.High	0.046175	0.100270	0.461	0.645
L2.Open	0.371021	0.791368	0.469	0.639
L2.Close	-1.249394	0.574594	-2.174	0.030
L3.Volume	-0.002638	0.001142	-2.309	0.021
L3.Low	-0.058762	0.110545	-0.532	0.595
L3.High	0.055307	0.121489	0.455	0.649
L3.Open	1.720831	0.971023	1.772	0.076
L3.Close	-0.439227	0.807541	-0.544	0.587
L4.Volume	-0.003281	0.001202	-2.728	0.006
L4.Low	-0.143547	0.125057	-1.148	0.251
L4.High	0.152114	0.140224	1.085	0.278
L4.Open	2.209489	1.102713	2.004	0.045
L4.Close	-1.759103	0.995548	-1.767	0.077

L5.Volume	-0.001530	0.001239	-1.235	0.217
L5.Low	-0.016379	0.138205	-0.119	0.906
L5.High	0.073010	0.155495	0.470	0.639
L5.Open	3.697701	1.224881	3.019	0.003
L5.Close	-2.295101	1.132271	-2.027	0.043
L6.Volume	-0.000694	0.001261	-0.550	0.582
L6.Low	-0.003767	0.148337	-0.025	0.980
L6.High	-0.139822	0.169500	-0.825	0.409
L6.Open	3.305220	1.330862	2.484	0.013
L6.Close	-3.716935	1.260196	-2.949	0.003
L7.Volume	-0.000553	0.001293	-0.427	0.669
L7.Low	-0.048939	0.156450	-0.313	0.754
L7.High	-0.245345	0.180301	-1.361	0.174
L7.Open	3.514761	1.428516	2.460	0.014
L7.Close	-3.067130	1.368422	-2.241	0.025
L8.Volume	-0.000495	0.001330	-0.372	0.710
L8.Low	0.002632	0.163411	0.016	0.987
L8.High	-0.409062	0.189498	-2.159	0.031
L8.Open	5.006221	1.518022	3.298	0.001
L8.Close	-3.232566	1.468327	-2.202	0.028
L9.Volume	-0.000059	0.001376	-0.043	0.966
L9.Low	0.046746	0.169760	0.275	0.783
L9.High	-0.339161	0.198550	-1.708	0.088
L9.Open	3.858848	1.597578	2.415	0.016
L9.Close	-4.707434	1.559785	-3.018	0.003
L10.Volume	-0.001390	0.001412	-0.985	0.325
L10.Low	0.057408	0.175595	0.327	0.744
L10.High	-0.274641	0.206254	-1.332	0.183
L10.Open	4.588699	1.674200	2.741	0.006
L10.Close	-3.631143	1.640407	-2.214	0.027
L11.Volume	-0.001607	0.001435	-1.120	0.263
L11.Low	0.015548	0.181240	0.086	0.932

L11.High	-0.261591	0.213673	-1.224	0.221
L11.Open	4.916734	1.746869	2.815	0.005
L11.Close	-4.383096	1.715460	-2.555	0.011
L12.Volume	-0.002609	0.001460	-1.787	0.074
L12.Low	0.115303	0.186746	0.617	0.537
L12.High	-0.304095	0.219756	-1.384	0.166
L12.Open	5.243710	1.819808	2.881	0.004
L12.Close	-4.699163	1.786477	-2.630	0.009
L13.Volume	-0.001464	0.001483	-0.988	0.323
L13.Low	0.217079	0.192100	1.130	0.258
L13.High	-0.285096	0.225572	-1.264	0.206
L13.Open	4.904585	1.885782	2.601	0.009
L13.Close	-5.173691	1.858145	-2.784	0.005
L14.Volume	-0.001574	0.001515	-1.039	0.299
L14.Low	0.150653	0.196938	0.765	0.444
L14.High	-0.274501	0.231321	-1.187	0.235
L14.Open	4.847447	1.940817	2.498	0.013
L14.Close	-4.800547	1.922283	-2.497	0.013
L15.Volume	-0.002179	0.001540	-1.415	0.157
L15.Low	-0.035299	0.201925	-0.175	0.861
L15.High	-0.315378	0.236680	-1.333	0.183
L15.Open	3.687442	1.989623	1.853	0.064
L15.Close	-4.597237	1.976497	-2.326	0.020
L16.Volume	-0.000264	0.001559	-0.170	0.865
L16.Low	0.014175	0.206296	0.069	0.945
L16.High	-0.387251	0.242697	-1.596	0.111
L16.Open	3.049935	2.038869	1.496	0.135
L16.Close	-3.362373	2.024406	-1.661	0.097
L17.Volume	-0.000817	0.001580	-0.517	0.605
L17.Low	-0.121972	0.211149	-0.578	0.563
L17.High	-0.409742	0.247660	-1.654	0.098
L17.Open	4.122745	2.082644	1.980	0.048

L17.Close	-2.611833	2.072315	-1.260	0.208
L18.Volume	-0.000930	0.001606	-0.579	0.563
L18.Low	-0.117421	0.214939	-0.546	0.585
L18.High	-0.313230	0.252637	-1.240	0.215
L18.Open	3.640380	2.128841	1.710	0.087
L18.Close	-3.696504	2.115821	-1.747	0.081
L19.Volume	-0.001844	0.001636	-1.127	0.260
L19.Low	-0.060909	0.217731	-0.280	0.780
L19.High	-0.240547	0.257206	-0.935	0.350
L19.Open	3.719032	2.174084	1.711	0.087
L19.Close	-3.292532	2.160149	-1.524	0.127
L20.Volume	-0.000461	0.001659	-0.278	0.781
L20.Low	0.010213	0.219870	0.046	0.963
L20.High	-0.265703	0.261217	-1.017	0.309
L20.Open	4.651052	2.218023	2.097	0.036
L20.Close	-3.462383	2.202672	-1.572	0.116
L21.Volume	-0.001186	0.001677	-0.707	0.480
L21.Low	0.029520	0.222354	0.133	0.894
L21.High	-0.171202	0.264163	-0.648	0.517
L21.Open	4.172243	2.259003	1.847	0.065
L21.Close	-4.493391	2.244207	-2.002	0.045
L22.Volume	-0.001729	0.001693	-1.021	0.307
L22.Low	0.010919	0.224681	0.049	0.961
L22.High	-0.336233	0.266503	-1.262	0.207
L22.Open	3.074082	2.296139	1.339	0.181
L22.Close	-3.965480	2.284134	-1.736	0.083
L23.Volume	-0.001680	0.001707	-0.984	0.325
L23.Low	-0.084501	0.227013	-0.372	0.710
L23.High	-0.341829	0.268738	-1.272	0.203
L23.Open	2.876739	2.325684	1.237	0.216
L23.Close	-2.768395	2.320958	-1.193	0.233
L24.Volume	-0.000508	0.001723	-0.295	0.768

L24.Low	-0.016757	0.228996	-0.073	0.942
L24.High	-0.365170	0.271332	-1.346	0.178
L24.Open	1.674374	2.352780	0.712	0.477
L24.Close	-2.535943	2.350788	-1.079	0.281
L25.Volume	-0.000962	0.001744	-0.551	0.581
L25.Low	-0.092980	0.230980	-0.403	0.687
L25.High	-0.376140	0.273812	-1.374	0.170
L25.Open	2.336681	2.378787	0.982	0.326
L25.Close	-1.241187	2.376598	-0.522	0.601
L26.Volume	-0.000377	0.001760	-0.214	0.831
L26.Low	0.099415	0.233273	0.426	0.670
L26.High	-0.429351	0.276385	-1.553	0.120
L26.Open	2.861319	2.399990	1.192	0.233
L26.Close	-1.925249	2.400949	-0.802	0.423
L27.Volume	-0.000774	0.001772	-0.437	0.662
L27.Low	0.198839	0.235261	0.845	0.398
L27.High	-0.400000	0.279413	-1.432	0.152
L27.Open	2.761231	2.420532	1.141	0.254
L27.Close	-2.608918	2.420566	-1.078	0.281
L28.Volume	-0.000028	0.001783	-0.016	0.987
L28.Low	0.278962	0.236879	1.178	0.239
L28.High	-0.391566	0.281832	-1.389	0.165
L28.Open	3.391473	2.439115	1.390	0.164
L28.Close	-2.598938	2.438784	-1.066	0.287
L29.Volume	-0.000521	0.001792	-0.291	0.771
L29.Low	0.223551	0.237704	0.940	0.347
L29.High	-0.115288	0.283849	-0.406	0.685
L29.Open	4.904680	2.453978	1.999	0.046
L29.Close	-3.417008	2.455203	-1.392	0.164
L30.Volume	-0.001263	0.001794	-0.704	0.481
L30.Low	0.155452	0.238579	0.652	0.515
L30.High	-0.125453	0.285371	-0.440	0.660

L30.Open	3.978329	2.469485	1.611	0.107
L30.Close	-4.971185	2.468492	-2.014	0.044
L31.Volume	-0.001214	0.001795	-0.676	0.499
L31.Low	0.163950	0.239443	0.685	0.494
L31.High	-0.132945	0.286484	-0.464	0.643
L31.Open	4.115658	2.482103	1.658	0.097
L31.Close	-4.022454	2.483513	-1.620	0.105
L32.Volume	-0.001313	0.001800	-0.729	0.466
L32.Low	0.064547	0.240515	0.268	0.788
L32.High	-0.112775	0.288118	-0.391	0.695
L32.Open	3.728314	2.494826	1.494	0.135
L32.Close	-4.107700	2.496131	-1.646	0.100
L33.Volume	-0.000339	0.001806	-0.187	0.851
L33.Low	0.061350	0.241344	0.254	0.799
L33.High	-0.129889	0.289592	-0.449	0.654
L33.Open	4.784163	2.506355	1.909	0.056
L33.Close	-3.664843	2.509331	-1.460	0.144
L34.Volume	-0.000673	0.001813	-0.371	0.711
L34.Low	0.086508	0.242290	0.357	0.721
L34.High	-0.100993	0.290886	-0.347	0.728
L34.Open	5.415299	2.518649	2.150	0.032
L34.Close	-4.770971	2.521924	-1.892	0.059
L35.Volume	-0.001689	0.001815	-0.931	0.352
L35.Low	0.117788	0.243299	0.484	0.628
L35.High	-0.065719	0.292027	-0.225	0.822
L35.Open	6.216382	2.531062	2.456	0.014
L35.Close	-5.409519	2.535638	-2.133	0.033
L36.Volume	0.000210	0.001813	0.116	0.908
L36.Low	0.209212	0.244530	0.856	0.392
L36.High	-0.161541	0.292327	-0.553	0.581
L36.Open	6.050494	2.541994	2.380	0.017
L36.Close	-6.294221	2.548893	-2.469	0.014

L37.Volume	-0.000340	0.001815	-0.187	0.851
L37.Low	0.303931	0.245883	1.236	0.216
L37.High	-0.048421	0.292442	-0.166	0.868
L37.Open	5.341206	2.548877	2.096	0.036
L37.Close	-6.187446	2.561117	-2.416	0.016
L38.Volume	-0.000105	0.001821	-0.058	0.954
L38.Low	0.363993	0.247047	1.473	0.141
L38.High	0.085636	0.292710	0.293	0.770
L38.Open	4.098021	2.550122	1.607	0.108
L38.Close	-5.706358	2.569475	-2.221	0.026
L39.Volume	-0.001490	0.001826	-0.816	0.415
L39.Low	0.346212	0.248257	1.395	0.163
L39.High	0.072848	0.293268	0.248	0.804
L39.Open	4.979397	2.551836	1.951	0.051
L39.Close	-4.538147	2.571930	-1.764	0.078
L40.Volume	-0.000677	0.001829	-0.370	0.711
L40.Low	0.266113	0.249530	1.066	0.286
L40.High	0.082257	0.293631	0.280	0.779
L40.Open	5.164182	2.551858	2.024	0.043
L40.Close	-5.321798	2.575083	-2.067	0.039
L41.Volume	-0.000943	0.001823	-0.517	0.605
L41.Low	0.172622	0.250620	0.689	0.491
L41.High	0.133615	0.293906	0.455	0.649
L41.Open	5.250688	2.551217	2.058	0.040
L41.Close	-5.520742	2.576356	-2.143	0.032
L42.Volume	0.001736	0.001816	0.956	0.339
L42.Low	0.231598	0.251064	0.922	0.356
L42.High	-0.024980	0.294087	-0.085	0.932
L42.Open	6.508356	2.552049	2.550	0.011
L42.Close	-5.483681	2.577643	-2.127	0.033
L43.Volume	0.000547	0.001813	0.302	0.763
L43.Low	0.145889	0.251248	0.581	0.561

L43.High	0.114413	0.294109	0.389	0.697
L43.Open	7.431213	2.552280	2.912	0.004
L43.Close	-6.772894	2.579728	-2.625	0.009
L44.Volume	-0.000604	0.001808	-0.334	0.738
L44.Low	0.175258	0.251414	0.697	0.486
L44.High	0.320664	0.293978	1.091	0.275
L44.Open	6.276631	2.553842	2.458	0.014
L44.Close	-7.784751	2.580168	-3.017	0.003
L45.Volume	-0.000540	0.001803	-0.299	0.765
L45.Low	0.063613	0.250894	0.254	0.800
L45.High	0.330380	0.293497	1.126	0.260
L45.Open	5.509381	2.549985	2.161	0.031
L45.Close	-6.719900	2.581989	-2.603	0.009
L46.Volume	-0.000119	0.001801	-0.066	0.947
L46.Low	0.031417	0.250091	0.126	0.900
L46.High	0.132572	0.292710	0.453	0.651
L46.Open	6.522342	2.543477	2.564	0.010
L46.Close	-5.765043	2.578477	-2.236	0.025
L47.Volume	-0.000821	0.001797	-0.457	0.648
L47.Low	-0.051456	0.249065	-0.207	0.836
L47.High	0.187611	0.291615	0.643	0.520
L47.Open	5.030804	2.539250	1.981	0.048
L47.Close	-6.677867	2.571816	-2.597	0.009
L48.Volume	-0.001425	0.001793	-0.795	0.427
L48.Low	-0.149037	0.248540	-0.600	0.549
L48.High	0.395852	0.290413	1.363	0.173
L48.Open	4.998798	2.531304	1.975	0.048
L48.Close	-5.181627	2.566227	-2.019	0.043
L49.Volume	-0.002551	0.001786	-1.428	0.153
L49.Low	-0.173067	0.248185	-0.697	0.486
L49.High	0.305409	0.289374	1.055	0.291
L49.Open	3.718093	2.520224	1.475	0.140

L49.Close	-5.193485	2.557036	-2.031	0.042
L50.Volume	-0.000383	0.001785	-0.214	0.830
L50.Low	-0.138747	0.247830	-0.560	0.576
L50.High	0.336088	0.288534	1.165	0.244
L50.Open	3.664997	2.507626	1.462	0.144
L50.Close	-3.945793	2.543882	-1.551	0.121
L51.Volume	0.000638	0.001779	0.359	0.720
L51.Low	0.004025	0.247424	0.016	0.987
L51.High	0.300861	0.287735	1.046	0.296
L51.Open	3.392640	2.495442	1.360	0.174
L51.Close	-3.895133	2.528670	-1.540	0.123
L52.Volume	-0.000028	0.001768	-0.016	0.987
L52.Low	0.152852	0.246853	0.619	0.536
L52.High	0.377676	0.286819	1.317	0.188
L52.Open	3.488035	2.479981	1.406	0.160
L52.Close	-3.853081	2.514568	-1.532	0.125
L53.Volume	-0.000218	0.001761	-0.124	0.902
L53.Low	0.264695	0.245723	1.077	0.281
L53.High	0.368200	0.285730	1.289	0.198
L53.Open	2.819555	2.462378	1.145	0.252
L53.Close	-4.046837	2.498262	-1.620	0.105
L54.Volume	-0.000863	0.001755	-0.491	0.623
L54.Low	0.328513	0.244393	1.344	0.179
L54.High	0.345461	0.283943	1.217	0.224
L54.Open	2.539591	2.444432	1.039	0.299
L54.Close	-3.460932	2.480661	-1.395	0.163
L55.Volume	-0.000572	0.001754	-0.326	0.744
L55.Low	0.246212	0.243433	1.011	0.312
L55.High	0.355406	0.282055	1.260	0.208
L55.Open	1.795480	2.422823	0.741	0.459
L55.Close	-3.167182	2.463558	-1.286	0.199
L56.Volume	-0.000988	0.001745	-0.566	0.571

L56.Low	0.282516	0.242618	1.164	0.244
L56.High	0.294106	0.280377	1.049	0.294
L56.Open	2.791682	2.401317	1.163	0.245
L56.Close	-2.407443	2.443503	-0.985	0.325
L57.Volume	-0.001154	0.001732	-0.666	0.505
L57.Low	0.317949	0.241686	1.316	0.188
L57.High	0.324017	0.278530	1.163	0.245
L57.Open	2.326398	2.377223	0.979	0.328
L57.Close	-3.366653	2.423421	-1.389	0.165
L58.Volume	-0.001654	0.001719	-0.963	0.336
L58.Low	0.365645	0.240222	1.522	0.128
L58.High	0.295650	0.276410	1.070	0.285
L58.Open	1.418071	2.352047	0.603	0.547
L58.Close	-2.953590	2.401259	-1.230	0.219
L59.Volume	0.000083	0.001699	0.049	0.961
L59.Low	0.390390	0.238511	1.637	0.102
L59.High	0.283792	0.273862	1.036	0.300
L59.Open	1.698609	2.324320	0.731	0.465
L59.Close	-2.108743	2.377315	-0.887	0.375
L60.Volume	0.000783	0.001684	0.465	0.642
L60.Low	0.376250	0.236800	1.589	0.112
L60.High	0.284684	0.271222	1.050	0.294
L60.Open	1.538930	2.291332	0.672	0.502
L60.Close	-2.380783	2.351701	-1.012	0.311
L61.Volume	-0.000162	0.001662	-0.098	0.922
L61.Low	0.357076	0.235012	1.519	0.129
L61.High	0.335440	0.268453	1.250	0.211
L61.Open	0.636773	2.261560	0.282	0.778
L61.Close	-2.216244	2.320648	-0.955	0.340
L62.Volume	0.000472	0.001644	0.287	0.774
L62.Low	0.337479	0.233262	1.447	0.148
L62.High	0.328245	0.265219	1.238	0.216

L62.Open	1.440925	2.231986	0.646	0.519
L62.Close	-1.292545	2.291640	-0.564	0.573
L63.Volume	0.002018	0.001629	1.239	0.215
L63.Low	0.302249	0.230906	1.309	0.191
L63.High	0.447773	0.261904	1.710	0.087
L63.Open	1.191218	2.197886	0.542	0.588
L63.Close	-2.194898	2.262977	-0.970	0.332
L64.Volume	0.000526	0.001612	0.327	0.744
L64.Low	0.340751	0.228604	1.491	0.136
L64.High	0.452037	0.258789	1.747	0.081
L64.Open	1.484492	2.160809	0.687	0.492
L64.Close	-1.914504	2.229513	-0.859	0.391
L65.Volume	-0.000346	0.001596	-0.217	0.828
L65.Low	0.331287	0.225860	1.467	0.142
L65.High	0.361838	0.255518	1.416	0.157
L65.Open	1.292768	2.124248	0.609	0.543
L65.Close	-2.223013	2.194311	-1.013	0.311
L66.Volume	0.000387	0.001576	0.245	0.806
L66.Low	0.325539	0.223110	1.459	0.145
L66.High	0.249884	0.251213	0.995	0.320
L66.Open	1.815052	2.084775	0.871	0.384
L66.Close	-1.883030	2.160384	-0.872	0.383
L67.Volume	-0.000331	0.001547	-0.214	0.831
L67.Low	0.392208	0.219911	1.783	0.075
L67.High	0.335458	0.246225	1.362	0.173
L67.Open	2.304092	2.041497	1.129	0.259
L67.Close	-2.445949	2.123048	-1.152	0.249
L68.Volume	0.000561	0.001522	0.369	0.712
L68.Low	0.383918	0.215918	1.778	0.075
L68.High	0.282142	0.241061	1.170	0.242
L68.Open	0.788736	1.997334	0.395	0.693
L68.Close	-2.980292	2.080812	-1.432	0.152

L69.Volume	-0.000597	0.001499	-0.398	0.691
L69.Low	0.351169	0.211227	1.663	0.096
L69.High	0.293225	0.235439	1.245	0.213
L69.Open	-0.566487	1.945414	-0.291	0.771
L69.Close	-1.468424	2.038343	-0.720	0.471
L70.Volume	-0.001206	0.001480	-0.815	0.415
L70.Low	0.327814	0.206817	1.585	0.113
L70.High	0.257708	0.228588	1.127	0.260
L70.Open	-0.801510	1.889097	-0.424	0.671
L70.Close	-0.069095	1.987494	-0.035	0.972
L71.Volume	-0.002250	0.001456	-1.545	0.122
L71.Low	0.256269	0.202144	1.268	0.205
L71.High	0.285176	0.221822	1.286	0.199
L71.Open	-0.372814	1.831866	-0.204	0.839
L71.Close	0.227138	1.932002	0.118	0.906
L72.Volume	-0.002417	0.001426	-1.695	0.090
L72.Low	0.252780	0.196936	1.284	0.199
L72.High	0.166823	0.215269	0.775	0.438
L72.Open	-0.610503	1.768793	-0.345	0.730
L72.Close	-0.091064	1.875183	-0.049	0.961
L73.Volume	-0.001580	0.001406	-1.124	0.261
L73.Low	0.237404	0.191375	1.241	0.215
L73.High	0.060473	0.208603	0.290	0.772
L73.Open	-1.304652	1.704257	-0.766	0.444
L73.Close	0.266151	1.812732	0.147	0.883
L74.Volume	0.000593	0.001385	0.428	0.669
L74.Low	0.245137	0.185693	1.320	0.187
L74.High	0.051058	0.201141	0.254	0.800
L74.Open	-0.855628	1.633251	-0.524	0.600
L74.Close	1.032829	1.748087	0.591	0.555
L75.Volume	0.000396	0.001364	0.290	0.772
L75.Low	0.355855	0.180036	1.977	0.048

L75.High	0.040324	0.192748	0.209	0.834
L75.Open	-1.398931	1.555592	-0.899	0.368
L75.Close	0.518715	1.675763	0.310	0.757
L76.Volume	0.001928	0.001330	1.450	0.147
L76.Low	0.381039	0.173467	2.197	0.028
L76.High	0.009183	0.184772	0.050	0.960
L76.Open	-1.883412	1.472079	-1.279	0.201
L76.Close	1.032867	1.596237	0.647	0.518
L77.Volume	0.000667	0.001289	0.517	0.605
L77.Low	0.468566	0.165951	2.824	0.005
L77.High	0.016229	0.176822	0.092	0.927
L77.Open	-1.579714	1.384417	-1.141	0.254
L77.Close	1.460518	1.510129	0.967	0.333
L78.Volume	0.000364	0.001256	0.290	0.772
L78.Low	0.418832	0.157829	2.654	0.008
L78.High	0.087305	0.167701	0.521	0.603
L78.Open	-2.047171	1.287784	-1.590	0.112
L78.Close	1.088691	1.417918	0.768	0.443
L79.Volume	-0.000433	0.001229	-0.352	0.725
L79.Low	0.341217	0.149758	2.278	0.023
L79.High	0.056473	0.156947	0.360	0.719
L79.Open	-1.243564	1.183928	-1.050	0.294
L79.Close	1.605781	1.318411	1.218	0.223
L80.Volume	-0.001798	0.001212	-1.483	0.138
L80.Low	0.213531	0.138808	1.538	0.124
L80.High	0.136170	0.144142	0.945	0.345
L80.Open	-0.150417	1.062721	-0.142	0.887
L80.Close	0.921743	1.211225	0.761	0.447
L81.Volume	-0.002414	0.001185	-2.037	0.042
L81.Low	0.179911	0.125408	1.435	0.151
L81.High	0.212609	0.129963	1.636	0.102
L81.Open	-0.307051	0.921291	-0.333	0.739

L81.Close	-0.220589	1.086687	-0.203	0.839
L82.Volume	-0.002607	0.001133	-2.302	0.021
L82.Low	0.133620	0.109794	1.217	0.224
L82.High	0.230120	0.113660	2.025	0.043
L82.Open	-1.575081	0.752734	-2.092	0.036
L82.Close	-0.074789	0.940825	-0.079	0.937
L83.Volume	0.000748	0.001047	0.715	0.475
L83.Low	0.102656	0.090930	1.129	0.259
L83.High	0.036772	0.095402	0.385	0.700
L83.Open	-0.295984	0.538541	-0.550	0.583
L83.Close	1.294623	0.764791	1.693	0.090
L84.Volume	-0.000049	0.000903	-0.055	0.956
L84.Low	0.011889	0.064378	0.185	0.853
L84.High	0.004066	0.068548	0.059	0.953
L84.Open	0.009962	0.050434	0.198	0.843
L84.Close	0.212484	0.542339	0.392	0.695

Bibliography

[1] AA Adebiyi et al. “Stock price prediction using neural network with hybridized market indicators”. In: *Journal of Emerging Trends in Computing and Information Sciences* 3.1 (2012), pp. 1–9.

[2] Falah YH Ahmed, Yasir Hassan Ali, and Siti Mariyam Shamsuddin. “Using K-Fold Cross Validation Proposed Models for Spikeprop Learning Enhancements”. In: *International Journal of Engineering & Technology* 7.4.11 (2018), pp. 145–151.

[3] Khaled A Althelaya, El-Sayed M El-Alfy, and Salahadin Mohammed. “Evaluation of bidirectional lstm for short-and long-term stock market prediction”. In: *2018 9th International Conference on Information and Communication Systems (ICICS)*. IEEE. 2018, pp. 151–156.

[4] A Mohamed Ashik and K Senthamarai Kannan. “Forecasting National Stock Price using ARIMA model”. In: *Global and Stochastic Analysis* 4.1 (2017), pp. 77–81.

[5] Suryoday Basak et al. “Predicting the direction of stock market prices using tree-based classifiers”. In: *The North American Journal of Economics and Finance* 47 (2019), pp. 552–567.

[6] Mustain Billah, Sajjad Waheed, and Abu Hanifa. “Predicting Closing Stock Price using Artificial Neural Network and Adaptive Neuro Fuzzy Inference System (ANFIS): The Case of the Dhaka Stock Exchange”. In: *International Journal of Computer Applications* 129.11 (2015), pp. 1–5.

[7] S Bogle and W Potter. “A machine learning predictive model for the jamaica frontier market”. In: *Proceedings of the 2015 Int’l Conference of Data Mining and Knowledge Engineering*. 2015.

[8] Eliezer Bose, Marilyn Hravnak, and Susan M Sereika. "Vector autoregressive (VAR) models and granger causality in time series analysis in nursing research: dynamic changes among vital signs prior to cardiorespiratory instability events as an example". In: *Nursing research* 66.1 (2017), p. 12.

[9] Jason Brownlee. *Deep learning with Python: develop deep learning models on Theano and TensorFlow using Keras*. Machine Learning Mastery, 2016.

[10] Russell Chaplin. "The predictability of real office rents". In: *Journal of Property Research* 16.1 (1999), pp. 21–49.

[11] Songsheng Chen, Bushra Komal, et al. "Impact of stock market development on economic growth: Evidence from lower middle income countries". In: *Management and Administrative Sciences Review* 5.2 (2016), pp. 86–97.

[12] Wim De Mulder, Steven Bethard, and Marie-Francine Moens. "A survey on the application of recurrent neural networks to statistical language modeling". In: *Computer Speech & Language* 30.1 (2015), pp. 61–98.

[13] Tina Ding, Vanessa Fang, and Daniel Zuo. "Stock market prediction based on time series data and market sentiment". In: *URL http://murphy. wot. eecs. northwestern. edu/˜ pzu918/EECS349/final_dZuo_tDing_vFang. pdf* (2013).

[14] Andy Field. *Discovering statistics using IBM SPSS statistics*. sage, 2013.

[15] Alexandra Gabriela ğiĠan. "The efficient market hypothesis: Review of specialized literature and empirical research". In: *Procedia Economics and Finance* 32 (2015), pp. 442–449.

[16] Andro Gogichaishvili. "Stock Price Forecasting Models". In: (2014).

[17] Bruno Miranda Henrique, Vinicius Amorim Sobreiro, and Herbert Kimura. "Stock price prediction using support vector regression on daily and up to the minute prices". In: *The Journal of Finance and Data Science* 4.3 (2018), pp. 183–201.

[18] Sadegh Bafandeh Imandoust and Mohammad Bolandraftar. "Forecasting the direction of stock market index movement using three data mining techniques: the case of Tehran Stock Exchange". In: *International Journal of Engineering Research and Applications* 4.6 (2014), pp. 106–117.

[19] Yixian Liu, Matthew C Roberts, and Ramteen Sioshansi. "A vector autoregression weather model for electricity supply and demand modeling". In: *Journal of Modern Power Systems and Clean Energy* 6.4 (2018), pp. 763–776.

[20] Helmut Lütkepohl. "Vector autoregressive models". In: *Handbook of Research Methods and Applications in Empirical Macroeconomics*. Edward Elgar Publishing, 2013.

[21] Saahil Madge. "Predicting stock price direction using support vector machines". In: *Independent work report spring* (2015).

[22] Teo Manojlović and Ivan Štajduhar. "Predicting stock market trends using random forests: A sample of the Zagreb stock exchange". In: *2015 38th International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO)*. IEEE. 2015, pp. 1189–1193.

[23] Stiaan Maree and Kevin Johnston. "Critical insights into the design of big data analytics research: How Twitter "moods" predict stock exchange index movement". In: (2015).

[24] Lufuno Ronald Marwala. "Forecasting the stock market index using artificial intelligence techniques". PhD thesis. 2010.

[25] Najeb MH Masoud. "The impact of stock market performance upon economic growth". In: *International Journal of Economics and Financial Issues* 3.4 (2013), p. 788.

[26] Denise Mhlanga. *Johannesburg Stock Exchange history*. https://www.property24.com/articles/johannesburg-stock-exchange-history/16977. Accessed March 10, 2019. 2013.

[27] Lucas Nunno. "Stock market price prediction using linear and polynomial regression models". In: *Computer Science Department, University of New Mexico: Albuquerque, NM, USA* (2014).

[28] Christopher Olah. *Understanding LSTM Networks*. https://colah.github.io/posts/2015-08-Understanding-LSTMs/. Accessed February 26, 2019. 2015.

[29] Pretesh B Patel and Tshilidzi Marwala. "Forecasting closing price indices using neural networks". In: *Systems, Man and Cybernetics, 2006. SMC'06. IEEE International Conference on*. Vol. 3. IEEE. 2006, pp. 2351–2356.

[30] Michael Phi. *Illustrated Guide to LSTM's and GRU's: A step by step explanation*. https://towardsdatascience.com/illustrated-guide-to-lstms-and-gru-s-a-step-by-step-explanation-44e9eb85bf21. Accessed February 26, 2019. 2018.

[31] Mingyue Qiu and Yu Song. "Predicting the direction of stock market index movement using an optimized artificial neural network model". In: *PloS one* 11.5 (2016), e0155133.

[32] Murtaza Roondiwala, Harshal Patel, and Shraddha Varma. "Predicting stock prices using LSTM". In: *International Journal of Science and Research (IJSR)* 6.4 (2017), pp. 1754–1756.

[33] PK Sahoo and Krishna Charlapally. "Stock price prediction using regression analysis". In: *International Journal of Scientific & Engineering Research* 6.3 (2015).

[34] Arjun Singh Saud and Subarna Shakya. "Analysis of look back period for stock price prediction with RNN variants: A case study on banking sector of NEPSE". In: *Procedia Computer Science* 167 (2020), pp. 788–798.

[35] Sreelekshmy Selvin et al. "Stock price prediction using LSTM, RNN and CNN-sliding window model". In: *2017 International Conference on Advances in Computing, Communications and Informatics (ICACCI)*. IEEE. 2017, pp. 1643–1647.

[36] Chi-Feng Wang. *The Vanishing Gradient Problem*. https://towardsdatascience.com/the-vanishing-gradient-problem-69bf08b15484. Accessed June 15, 2020. 2019.

[37] Yi-Hsien Wang, Chin-Tsai Lin, and Jung Dan Lin. "Does weather impact the stock market? Empirical evidence in Taiwan". In: *Quality & Quantity* 46.2 (2012), pp. 695–703.

[38] Yusen Wang, Wenlong Liao, and Yuqing Chang. "Gated Recurrent Unit Network-Based Short-Term Photovoltaic Forecasting". In: *Energies* 11.8 (2018), p. 2163.