STOCK PRICE PREDICTION USING DEEP LEARNING METHODS IN HIGH-FREQUENCY TRADING

EMRE ALBAYRAK

SEPTEMBER 2021

ÇANKAYA UNIVERSITY

THE GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES

DEPARTMENT OF COMPUTER ENGINEERING

COMPUTER ENGINEERING MASTER THESIS

STOCK PRICE PREDICTION USING DEEP LEARNING METHODS IN HIGH-FREQUENCY TRADING

EMRE ALBAYRAK

SEPTEMBER 2021

ÖZET

DERİN ÖĞRENME METOTLARINI KULLANARAK YÜKSEK FREKANSLI İŞLEMLERDE BORSA FİYAT TAHMİNİ

ALBAYRAK, Emre
Yüksek Lisans, Bilgisayar Mühendisliği Anabilim Dalı

Tez Yöneticisi: Dr. Öğr. Üyesi Nurdan SARAN
Eylül 2021, 69 sayfa

Borsa analizleri finansal, politik ve sosyal göstergeler göz önünde bulundurularak yapılırken, büyük veri ve derin öğrenme teknolojilerindeki önemli gelişmeler araştırmacı ve yatırımcıların dikkatini bilgisayar destekli analizlere yöneltmiştir. Bu çalışmada temel olarak kullanılan Bütünleşik Otoregresif Hareketli Ortalama (ARIMA) modelinin yanında Uzun Kısa-Dönem Hafızalı (LSTM) ağlar, Kapı Özyinelemeli Geçitler (GRU), Uzun Kısa-Dönem Hafızalı ağlarda Dikkat Mekanizması olmak üzere dört farklı model incelenmiştir. Borsa İstanbul verileriyle gerçekleştirilen çalışmada gün içi verileriyle tahminler gerçekleştirilmiştir. Yapılan test çalışmaları sonucunda Kapı Özyinelemeli Geçitler'in diğer modellere göre daha iyi sonuç verdiği görülmüştür.

Anahtar Kelimeler: BIST, RNN, LSTM, GRU, ARIMA, Dikkat Mekanizması

ABSTRACT

STOCK PRICE PREDICTION USING DEEP LEARNING METHODS IN HIGH-FREQUENCY TRADING

ALBAYRAK, Emre
M.Sc., Department of Computer Engineering

Supervisor: Asst. Prof. Dr. Nurdan SARAN

Sep 2021, 69 pages

The stock market analysis examines and evaluates the stock market by considering the financial, political, and social indicators to make future predictions. Breakthrough results of advancements in big data and deep learning technologies attract the attention of researchers and traders to computer-assisted stock market analysis. There are several studies on stock market analysis using conventional machine learning and deep learning models. In this paper, we used Autoregressive Integrated Moving Average (ARIMA) as a base model and compared it with three different models of recurrent neural networks: Long Short-Term Memory (LSTM) networks, Gated Recurrent Unit (GRU), LSTM with an attention layer model. We compare the results and performance of four different models on Borsa Istanbul data while making intraday predictions. Even though the LSTM results are very close to the GRU model, GRU slightly outperforms the others.

Keywords: BIST, RNN, LSTM, GRU, ARIMA, Attention Mechanism

ACKNOWLEDGMENTS

I would like to express my sincere gratitude to Dr. Nurdan SARAN for her supervision, special guidance, suggestions, and encouragement through the development of this thesis. It is a pleasure to express my special thanks to my family for their valuable support.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

Abbreviations

ANN : Artificial Neural Network

RNN: Recurrent Neural Network

CNN : Convolutional Neural Network

DNN : Deep Neural Network

LSTM : Long Short-Term Memory

GRU: Gated Recurrent Unit

SVM : Support Vector Machines

LOG: Logistic Regression

BIST : Borsa Istanbul

MSE: Mean Squared Error

MAE: Mean Absolute Error

SP500: Standards and Poors 500

HSI: Hang Seng Index

DJIA : Dow Jones Industrial Average

CHAPTER 1

INTRODUCTION

1.1 LITERATURE REVIEW

Stock market analysis is one of today's most trending topics in finance, along with cryptocurrencies, forex, high-performance trading, etc. The analysis process examines the economic indicators to infer the current situation and predict the incoming situation. It has been a challenging task for many decades worldwide, and it has attracted the attention of both professionals and individual traders. The ease and pace of reaching the information increased the number of people affiliated with stock market analysis. It allows conducting successful analysis to individuals at modest expenses. The increasing attention has led to new lines of businesses such as online stock market analysis and trading platforms or online investment consultancy services etc. Besides economics and traders, it also attracts the attention of many researchers from diverse backgrounds such as economics and engineering disciplines. Even though the majority of the studies are related to stock market analysis, there are several recent works conducted in predicting forex and cryptocurrencies [25] [18]. The stock market itself is highly volatile and non-stationary. Besides, the decentralized structure of cryptocurrencies and lack of authority in the field make the market much more eligible for manipulators and frauds than the stock market. In [48] the authors compared past financial data of gulf countries to clarify the relationship between the stock market and cryptocurrencies. In terms of the trader's behavior stock market and cryptocurrencies are substitutes for each other, not complements. In [49], the authors researched the stock market and cryptocurrencies relationships in middle east region countries. The findings show that a strong relationship between these two markets. According to the results, in non-gulf middle east countries, for each 1% increase in cryptocurrencies, there is a 0.13% increase in stock market performance. In gulf countries, each 1% rise reduces the stock market by 0.13%. The insecure aspects of cryptocurrencies strengthen the stock market's hand against the crypto market, and the stock market still remains one of the most popular trading tools.

There are two commonly used methods of stock market analysis: fundamental and technical analysis. Fundamental analysis merges a wide range of diverse data sources to harness data in the stock market analysis process. Data sources may vary from political news, government policies to insider information. In [34] the authors remark the importance of external and long-term macro-economic factors in fundamental analysis. Fundamental analysis relies more on the reasoning of the traders rather than the financial indicators. However, the technical analysis relies on financial indicators such as stockcharts, current market trends, trade volumes, etc. Past trading activities, price trends, and change patterns are considered highly important in technical analysis. In addition to fundamental price variables, technical indicators and feature engineering have an important role in the analysis process. Technical analysis is used to understand short-term stock behavior, whereas fundamental analysis is used for long-term predictions [23]. In numerous recent works, Convolutional Neural Networks (CNN), Long Short-Term Memory (LSTM) networks, Artificial Neural Network (ANN) frameworks and machine learning algorithms are used on Borsa Istanbul (BIST) data [2] [22] [23] [29]. The use of extensive feature engineering is common in all of the studies. The additional features are technical indicators in these works. Only in [22] gold-dollar ratio is used as a plus to technical indicators. Root Mean Squared Error (RMSE), F-Measure, and Macro Averaged F-Measure metrics are used to compare the models' performance. All of the studies underlines the positive impact of extensive feature engineering on prediction results [2] [22] [23] [29]. Regardless of the analysis method that forecasting stock prices accurately is a challenging task due to its noisy, non-stationary and highly volatile nature [45] [8]. Speculators and competitors are the factors that make it even harder to capture the stock price trends' complexity [40] [17].

In [41] the authors researched 122 papers published between 2007 and 2018 about stock market prediction. The results show that 66% of the papers use technical analysis methods in stock prediction while the 23% and 11% use fundamental analysis and combined analysis methods. 98% of the fundamental analysis studies used the data from social network sites to infer the sentiment on financial markets. According to [41] the most common technical indicators used in papers are Simple Moving Average (SMA), Exponential Moving Average (EMA), Moving Average ConvergenceDivergence (MACD), Relative Strength Index (RSI), and Rate of Change (ROC).

There are two common hypotheses on the market's predictability which are the Random Walk Hypothesis (RWH) and Efficient Market Hypothesis (EMH) [15] [16]. RWH suggests that stock price is stochastic; hence, it is impossible to predict it correctly. EMH can be considered a revised version of RWH that discusses the predictability of the stock price in three categories: weak, semi-strong, and strong. Despite the views on the unpredictability of the stock prices, recent works are claiming that the feature selection can have a positive effect on stock price prediction and make it possible to get successful predictions [62]. Even though many studies converge both technical analysis and fundamental analysis with deep learning methods [2] [21], in this study, we will only focus on technical analysis methods.

Deep learning algorithms, also known as neural networks, are a family of algorithms that excel at making predictions about unseen data. These algorithms have been behind breakthrough results in computer vision, natural language processing, and speech recognition. The renaissance in artificial intelligence technology and the rapid domination of the field is enough to make them subjects of interest to any data scientist.The complexity of predicting stock prices accurately and emerging big data and deep learning technologies have led to the use of computers in stock market analysis [7] [45]. Affordable commodity hardware, rapidly evolving information retrieval technologies,and the increasing volumes of data make it possible to process large volumes of data to infer meaningful patterns and relationships [24]. Many machine learning methods have been used in stock price predicting, such as ANN, Support Vector Machines (SVM), mode decomposition models [29] [47]. Various deep learning algorithms have also been applied to forecasting stock price problems such as LSTM, CNN and algorithms using attention mechanisms [23] [45] [17] [7]. Deep learning methods help us in understanding the intrinsic non-linear relationships and latent factors inside the stock data harnessed to get the correct predictions on stock prices [40].

Besides the deep learning methods, conventional statistical methods are still used in stock price prediction applications. AutoRegressive Integrated Moving Average (ARIMA) model is one of the statistical models which is widely used in financial forecast applications due to its state of the art effectiveness and robustness [4]. ARIMA is based on averages of different subsets in the whole data, and it uses averages of past data to make predictions about the future. ARIMA is commonly used as a base model for comparison purposes with both statistical or deep learning models [37] [1]. In addition to the distinctive use of ARIMA in prediction and comparison phases, ARIMA is converged with deep learning as hybrid frameworks in much recent work in order to exploit the power of ARIMA in data processing phases

[55] [31] [46].

National stock markets, cryptocurrencies, forex, and gold prices are examples of deep learning algorithms applied to financial markets. Availability of high volumes of trading data and easy-to-implement algorithms directed the interest of both researchers and traders to this subject. This increasing interest has led to the massive use of deep learning algorithms in stock price prediction and many academic papers. Besides the use of LSTM, GRU, and simple ANN algorithms, Graph theory-based models [44], and ensemble models are now the subject of many works. Even though most of the studies aim to predict the next day price of the issue, there are recent works to make predictions in short-term intervals, such as hourly intraday predictions.

Neurons are simple computational units and fundamental building blocks of neural networks. Neurons are organized into layers, and neural networks are made up of layers. Learning is the process of adjusting weights and biases so that the network produces the correct outputs. In supervised learning, we assume that we have the right answer per data point. A series of transformations have been applied to data in different layers. The result is compared to the correct data point at the end of the process, and an error is calculated. Then, we propagate the error between the model's output and the correct value backward through the model. Then the gradient descent is calculated as back-propagation, and the weights of the neurons are updated. The aim of updating the weights is to minimize the error value. This is the typical learning process of each iteration in supervised deep learning.

The Recurrent Neural Network (RNN) is a class of ANNs used in applications when the previous state of the neurons matter. RNNs are good at learning the sequential characteristics of the input data while training. RNNs are applied time-series data problems such as natural language processing, speech recognition, stock price prediction, etc. In the learning process, RNNs use gradient-descent for error back-propagation to minimize the error loss. In some cases, the gradient becomes so small, and then the update of the weights becomes vanishingly small. This problem is called "vanishing gradient", and it prevents the network from effectively learning the hidden patterns of the data. LSTM network was proposed in 1997 to overcome the "vanishing gradient problems". The internal structure of the LSTM network, which is consisted of memory cells, prevents the vanishing gradient problem, and it provides solid learning of hidden patterns of the input data [19]. GRU is a mechanism that is very similar to LSTM with a few internal changes. It's considered to have a simpler architecture than LSTMs. GRUs are introduced in 2014 [10]. The attention mechanism focuses on crucial points of input data rather than the whole. In every iteration, the network might pay attention to a different part of the data [45] [9].

In [23], the authors proposed a CNN framework to predict the hourly stock price direction of the 100 stocks in BIST. A total number of 25 features are used in the network. They have implemented a feature utilization mechanism to order the features before they are given to the network as inputs. The feature utilization mechanism uses the correlation between the instances and features to rank the features. The experimental results show that the proposed framework outperforms the Logistic Regression (LR) and randomly ordered CNN models. In [29], the authors compared two different models' which are ANN and SVM performance, on predicting the movement direction of BISTstocks. Ten additional features of technical indicators, price information, etc., were selected as the feature set. Experimental results showed that ANN outperforms the SVM model. In [17], the authors compared LSTM, Random Forest (RAF), Deep Neural Network (DNN), and a Logistic Regression Classifier (LOG). They have predicted the daily directional movements of the constituent stocks of the Standard & Poor's 500 index (S&P 500). In [58], the authors applied sequence reconstruction and a utilized CNN method on various stocks from S&P 500 and S&P 500 index itself. The proposed model is compared with ARIMA, ARIMA with Wavelet Transform, LSTM, and HMM (Hidden Markov Model). Experimental results show that the proposed model outperforms other methods even though the LSTM model results can be considered competitive.

In [45], the authors compared three models, LSTM, LSTM with wavelet transform, and GRU on SP500, Hang Seng Index (HSI), and Dow Jones Industrial Average (DJIA) datasets. In [36], the authors compare two models on Hong Kong Stock data, LSTM and LSTM, with an attention mechanism. According to experimental results, the model with an attention mechanism outperforms the LSTM. In [32], the authors applied an attention-based LSTM model on China stock market data. Limit Order Books (LOB) data is used instead of traditional historical price data such as open, high, low, close, etc. The reason behind the data preference of LOB instead of historical price data is that LOBs contain more detailed information. The authors claimed that experimental results validated their proposed solution is effective in predicting stock price trends. In [56], the authors used the LSTM model with an attention layer on three selected China stock market indexes which are the Shanghai index, Shenzhen index, and the China Securities Index (CSI 300) index. Historical price data is used in this work to predict the trend of selected stock markets. According to the experimental results, the proposed LSTM network with an attention layer succeeded in predicting the stock market trends. In [63], an attention-based LSTM, baseline LSTM, and ARIMA models are used to predict Russell 2000, DJIA, and Nasdaq indexes. The experimental results proved that the proposed attention-based LSTM model outperforms other models according to theerror metrics.

In [27], selected stock papers from Taiwan Stock Exchange Corporation (TWSE) areused to compare the LSTM model and Convolutional LSTM model with the attention layer. Besides the historical price columns, several technical indicators such as stochastic oscillators and Moving Average Convergence/Divergence (MACD) are included in the input data. The proposed model consists of one convolutional layer, two LSTM layers, and one attention layer. According to the Mean Squared Error (MSE) metrics of experimental results, the proposed model positively impacts predicting the stock pricetrends. In [35], the attention-based LSTM method is applied in predicting both S&P 500 index and individual stock papers. In addition to the historical price data, the news and events are analyzed through Natural Language Processing (NLP) techniques, and the features are included in the input data. The attention-based LSTM model outperforms the other models in predicting the S&P 500 index, and then the model is applied to the individual stocks. Results show that including news and texts after processing as a feature into the data and using the attention-based LSTM model positively impacts stock price prediction in terms of the error values.

In [50], the authors compared three models of Vanilla RNN, which is the simple RNN architecture, LSTM, and GRU on Nepal Stock Exchange data. In that experiment, GRU performed better for stock price prediction over the LSTM and Vanilla RNN. The authors suggested that the reason behind these results is the compact architecture of GRU over LSTM and the less amount of data needed for training the model [57] [33]. Besides the technical analysis methods, which only focus on graphs, charts, price trends, etc., several studies apply deep neural networks in fundamental analysis methods. There are experimental studies using news and social media to get better results in predicting the stock market. In [30], the authors used an ensemble of deep neural networks consisting of CNNs, RNNs, and LSTM with BIST data. In addition, tweets in Turkish and English are retrieved to be included in the model creating phase. The authors have processed the data through a variety of natural language processing algorithms. In conclusion, they have considered the effect of related tweets in predicting the BIST index, and the experimental results showed that their proposed model gives better results than previous studies. In [5], the authors studied Polynomial Regression and Random Forest Regression, RNN, and LSTM using BIST data. The experimental results showed Random Forest Regression model overperformed all other three models, including RNN and LSTM network. In [12], the authors studied Multiplayer Perceptrons (MLP), SVMs, and LSTM networks with BIST data in predicting both close and open prices ofselected 42 stocks of BIST. The authors made several preprocessing operations to get the data ready for time-series analysis. In their experiment, the Multiplayer Perceptrons model is a class of ANNs. It commonly refers to the basic model of feed-forwarded ANN, which is consisted of at least three layers, including input, hidden, and output layers. According to the experimental results, MLP overperforms the other two models in predicting the opening prices. In [2], the LSTM network is used to predict the next day closing prices of two different stocks in BIST, which are AKBNK and GARAN. The authors created 29 technical indicators and the basic indicators of open, close, high, low, and volume values. The study remarks on the importance of feature engineering phases in predicting stock prices. In [28], LSTM network is used on five different stocks of BIST which are Turkish Airlines (THYAO), Akbank (AKBNK), Arçelik (ARCLK), Aselsan (ASELS), and Garanti (GARAN). After pre-processing phases, the data is fed into the LSTM model to create a model that can predict the next day's closing price. Experimental results show reasonably good performance in prediction. The accuracy of predictions for all of the five stocks is above 95%. In [20], 8 different banking stocks from BIST were used with a different set of technical indicators. The author makes hourly predictions of stock prices. LSTM networks, LSTM with attention layer, SVMs, and Light Gradient Boosting Machine (LighGBM) ensemble model were applied to BIST data. The experimental results show that LSTM with the attention layer outperforms the other models. In [53], the authors applied ARIMA, Linear Regression (LR), and LSTM models to the 30 selected stocks of BIST, and experimental results show that ARIMA outperforms the other models. According to experimental results of [51] comparing three different models, ARIMA outperforms LSTM and GRU on predicting BIST30, BIST50, and BIST100 indices. In [22], the authors used CNN to predict the price of three different stocks of BIST. In addition to the basic features, the gold-dollar ratio is included in the data. According to the experimental results, the gold-dollar ratio feature made a positive impact on prediction performance. In [38], the authors collected daily stock price data with five minutes of intervals of a company in the Indian Stock exchange. They applied both statistical models, machine learning, and deep learning methods. The experimental results show that while the LSTM model outperforms other statistical models and machine learning algorithms, the CNN model performs better than the LSTM model. Using intraday stock data and making predictions accordingly shows the interest of both traders and researchers in intraday stock price prediction. In [11], the authors applied RBM (Restricted Boltzmann Machine) and a three-layered DNN to the Korean stock market data, and the experimental results show that DNN outperforms the other models. This work remarks on the success of deep learning algorithms over other statistical models and machine learning algorithms in learning the hidden patterns of time-series datasets. In [52], the researchers used two years of Chinese stock market data for making short-term predictions. They have applied extensive feature engineering and fed the data into an LSTM model. The experimental results show that the proposed LSTM model outperforms most of the related works mentioned in that study.

In [26], the authors researched Chinese Stock Prices by using the ARIMA model to demonstrate the effects of the World Financial Crisis and the correlation between the Chinese stock prices and the Chinese manufacturing industry. They aimed to infer the patterns between the long-term economic movements and the stock prices. Their work proved that the World Financial Crisis affected China’s stock exchange and manufacturing industry, and the ARIMA model succeeded in such a long-term analysis. In [39], the authors used 56 Indian stock papers listed in Indian National Stock Exchange from different sectors to make stock price predictions. They applied the ARIMA model to the data, and the accuracy is above 85% for all the sectors. The model gives the best accuracy results for Fast Moving Consumer Goods (FMCG) sector over banking, information technologies, automobile, power, infrastructure, and steel. In [37], the authors compared three different models, which are ARIMA, ANN, and LSTM, to make a comparison between these three popular models. As a result, the authors indicate that LSTM overperforms the basic ANN and ARIMA models due to its internal design and ability to handle large data series. The work [4] suggests the ARIMA model for stock price prediction by applying the model on both New York Stock Exchange (NYSE) and Nigeria Stock

Exchange (NSE) data. The study favors the ARIMA model in the short-term prediction of stock prices. In [1], the authors examined the ARIMA model and ANNs by applying the models on NYSE data. The experimental results showed that ANNs perform better than the ARIMA model in stock price prediction. The work [43] suggests a hybrid model consisting of ARIMA and SVM model to converge both model's strengths into a hybrid model to make better predictions in stock prices. There are four different models proposed in this study. Theauthors have used the two models as standalone, a simple combination, and a hybrid model with optimal parameter selection. The models are compared through the MSE, Mean Absolute Error (MAE), Mean Absolute Percentage Error (MAPE), and RMSE values. According to the experimental results, the simple combination of ARIMA andSVM does not enhance the prediction performance as expected. However, combining two models with hyperparameter optimization as a hybrid model overperforms the other models as expected. In [55], the authors used the RNN model and ARIMA together in predicting Taiwan Stock Exchange (TSEWSI). They applied RNN based features which are extracted by using ARIMA. ARIMA model is used in the feature selection phase of pre-processing. They have used weekly data of certain stocks of TSEWSI and claimed that the results of the suggested model, which is a combination of RNN and ARIMA, over-performs the RNN model trained by using raw data. In [61], the authors proposed a hybrid model consisting of Wavelet Denoising, Attention-Based Recurrent Neural Network, and ARIMA model. USD/JPY currency data is used in this work. Wavelet transform is used in pre-processing phases to make the data more stabilized.

The attention-Based RNN model learns the latent factors and hidden patterns in the time-series data, and the ARIMA model is used in the final prediction phases. The authors claimed that the dynamic structure of the hybrid model makes it a perfect fit for forex markets. They have used the RMSE metric to compare the results, and according to the experimental results, the hybrid model overperforms the standalone neural network models. In [3] the authors implemented the ARIMA model on stock data to make short-term predictions. The data used in the work was acquired from ASE (Amman Stock Market) in Jordan. The data is daily stock price data, and it ranges from 1993 to 2017. MSE metric was used to evaluate the models during the grid search phase, and according to the experimental results, the ARIMA model gives promising results.In [13], the authors applied the ARIMA model on Indian

Stock Exchange data. The data ranges from 2007 to 2011. Four companies with the highest capital values are selected for the research. MAPE and similar other metrics are used to compare the models. According to the experimental results, ARIMA gives satisfactory results. In [6], monthly closing prices of the Indian Stock Exchange are collected for a period of 5 years. ARIMA model was applied to the collected data to make short-term predictions. The author draws attention to the importance of similar analysis in inferring the economic situation. In [31], a variety of different models are coupled into hybrid models. The models are ARIMA, SVM, ANN and RF. The hybrid models are ARIMA-SVM, ARIMA-ANN, and ARIMA-RF. According to the experimental results, ARIMA-SVM outperforms other models. In [14], a hybrid model of ARIMA and an artificial neural network was applied to Chinese Stock data. According to the results, the hybrid model outperforms another model. In [46], a hybrid model of ARIMA and ANN and the model was used on Sri Lanka National Stock Exchange to estimate short-term stock price prediction. According to the experimental results. In [54], the authors conducted research on predicting the next day's closing price of five different companies to compare ANN and Random Forest methods. Additional technical indicators are generated in this work besides the historical price data. The experimental results proved that ANN gives more results than Random Forest in predicting the stock prices. ARIMA, Support Vector Regression (SVR), MLP, LSTM with no Phase-Space Reconstruction (PSR), and LSTM with PSR methods are compared in stock price prediction task in [60]. PSR method was used in pre-processing phases. The experimental results showed that LSTM with the PSR method performed better than the other models in predicting the stock prices. The most popular machine learning techniques in stock price prediction applications, such as ARIMA, ANNs, MLP, RNN, LSTM, SVM, etc., are compared in terms of types ofinput, strengths, and weaknesses in [42]. The authors remarked on the positive effect of sentiment analysis on stock price prediction, which utilizes news, social media, political changes, etc.

The need for successful technical analysis in finance becomes more demanding with the advance of technology. Computer-assisted financial predictions attracted the attention of both traders, individuals, and academicians in an increasing trend. Conventional statistical methods, machine learning algorithms, and deep learning algorithms are widely applied to address the challenge in stock markets, cryptocurrencies, forex, etc. All these methods can be applied as distinctive solutions or as a hybrid model. Besides using these methods in technical analysis, the cutting-edge technologies also enable us to exploit the strengths of both fundamental analysis and technical analysis. Inferring the meaning of price charts, technical indicators, and political news in stock price prediction gives very promising results. Considering all of the works cited above, computer-assisted financial predictions, especially those using deep learning methods, will remain a hot topic in both finance and technology.

1.2 OBJECTIVES AND CONTRIBUTIONS

Our main objective is to study state-of-the-art statistical models and deep learning methods in high-frequency trading using BIST data to make intraday predictions. We use the most popular and high-end deep learning methods such as LSTM with attention mechanism with BIST data to come up with a comparative study. We also implemented the ARIMA model for comparison purposes. This study uses LSTM, GRU, LSTM with attention mechanism, and ARIMA to forecast the stock prices. Both LSTM and GRU are parts of RNNs. We study both four methods with BIST stock data and compare the results. Our work is one of the first studies in forecasting BIST stock prices using LSTM with Luong attention. We study intraday hourly prediction with BIST stock data.

1.3 ORGANIZATION OF THE THESIS

This thesis has 4 chapters. Chapter 1 gives a brief introduction of stock analysis, deep learning and thesis outline. It also summarizes the related works. Chapter 2 gives detailed explanation of the models. It describes data retrieval, pre-processing operations and feature engineering processes. Chapter 3 has the model creation phases, results and discussion sections containing detailed graphics and results. Chapter 4 is the conclusion section of our paper.

CHAPTER 2

METHODOLOGY

2.1 DATASET

The data used in this study is hourly data of 98 different stocks in BIST, and the data is acquired through the Matriks Information Delivery Services Inc. The data has a date range from 2001 to 2020 that results in 772533 lines. Every file has 7 fundamentals columns of historical price data. Date, time, open price, close price, high price, low price and volume data are fundamental columns. In order to get successful prediction results we created technical indicators based on these 7 fundamental columns. Before we apply the feature engineering on the data we had to make initial filtering.

The data is filtered through multiple criteria such as the minimum distinct days, minimum time interval, date range, etc. Some parts of the data that belong to certain stocks are sparse in terms of the number of days. For example, we have only three days of data for a stock in a particular month, while others have 20 days. Besides that, some stocks had 3 hours of data for a certain day and 8 hours for the rest. Data of certain stocks' data range is between 2011 and 2016, and some are distinctively different. The changing inflation rates in the date range of 2001 and 2020 and the revaluation and redenomination process of Turkish Lira by the removal of the 6 zeros in that era have made us shorten the date range to a recent period.

In Table 2.1, the raw data example of ASELS stock after the initial filtering is given. It has 7 fundamental columns. It has more than 7 hours data per day and the data is later than 1 January 2015.

2.1.1 Data Retrieval

We had to reconsider the criteria and pre-process the data accordingly as the data preparation. We selected the minimum distinct days parameter as 800,

Table 2. 1: Data Pattern for Stock ASELS.

Data Pattern for Stock ASELS						
Date	***Time***	***Open***	***High***	***Low***	***Close***	***Volume***
19.01.2016	09:00	8.83	8.91	8.83	8.89	59599
19.01.2016	10:00	8.88	8.90	8.85	8.86	73356
19.01.2016	11:00	8.86	8.89	8.84	8.89	40513
19.01.2016	12:00	8.89	8.89	8.87	8.89	13522
19.01.2016	13:00	8.89	8.89	8.87	8.89	4053
19.01.2016	14:00	8.89	9.00	8.88	8.97	171011
19.01.2016	15:00	8.98	9.08	8.97	9.03	384621
19.01.2016	16:00	9.02	9.03	8.93	8.95	113097
19.01.2016	17:00	8.95	9.00	8.92	9.00	105489
20.01.2016	09:00	8.95	8.95	8.91	8.92	70761
20.01.2016	10:00	8.92	8.93	8.84	8.86	131160
20.01.2016	11:00	8.86	8.87	8.84	8.87	107839
20.01.2016	12:00	8.87	8.87	8.82	8.82	83392
20.01.2016	13:00	8.83	8.85	8.82	8.85	160975
20.01.2016	14:00	8.85	8.87	8.84	8.87	123985
20.01.2016	15:00	8.87	8.88	8.84	8.86	127552
20.01.2016	16:00	8.86	8.87	8.83	8.84	266924
20.01.2016	17:00	8.83	8.84	8.79	8.81	221120

minimum interval parameter as 7 and start date parameter as 01/01/2015.

After the filtering phase, every stock file has the data after 01/01/2015 and has at least 800 days of data and at minimum 7 hours of data per day. The final data consists of 29 different stocks and 180909 lines. The stocks in final data are the followings: AEFES, AKBNK, AKENR, AKSA, ALARK, ARCLK, ASELS, AYGAZ, BAGFS, BFREN, BOLUC, CIMSA, CLEBI, DOAS, DOHOL, ECILC, ENKAI, EREGL, FENER, FROTO, GARAN, GOLTS, GOODY, GSDHO, IHLAS, ISCTR, ISGYO, KARTN, KCHOL. In addition, we take the USD/TRY currency as another feature into consideration.

2.1.2 Pre-processing

Every file has seven columns of data: date, time, open price, high price, low price, close price, and volume. The date column refers to the day and the time column specifies thehour. We pre-process the files as told above and shrink the size of the files accordingly. Then we create a list of technical indicators based on the hourly values of open, high, low, close, and volume data.

There are 28 technical indicators we calculate during the pre-processing, and we have also added USD/TRY currency data. The currency data has a date range from 2010to 2019. We don't want to be the currency column sparse to see that it has made any difference. The total data size shrinks again while we merge the currency data as we expected. Some of the selected technical indicators are listed below and their formulasare given in Table 2.2.

- SMA: Simple Moving Average is moving average of prices over a given daterange
- ADL: Accumulation Distribution Line is an indicator of whether the stock priceis accumulated or distributed
- CCI: Commodity Channel Index is the difference between the current and historicalaverage price
- RSI: Relative Strength Index is an indicator that evaluates the strength of pricechanges of a stock
- TP: Typical Price is the average of daily high, low, and closing price of a stock
- EFI: Elder's force index indicates the strength of a move based on price andvolume of a stock
- VPT: Volume Price Trend shows strength of a trend based on percentage changesof price and volume of a stock

Table 2. 2: Selected Technical Indicators Formulas.

Indicator	***Formula***
SMA	$\frac{C_t - C_{t-1} - C_{t-n}}{n}$
ADL	$\frac{(C_t - L_t) - (H_t - C_t)}{H_t - L_t} \times Vt$
CCI	$\frac{M_t - SM_t}{0.015 D_t}$
RSI	$100 - \frac{100}{1 + \left(\sum_{i=0}^{n-1} U\, p_{t-i}/n\right) / \left(\sum_{i=0}^{n-1} D\, w_{t-i}/n\right)}$
TP	$\frac{H_t + L_t + C_t}{3}$
EFI	$(C_t - C_{t-1}) \times V_t$
VPT	$VPT_{t-1} + \left(\frac{(C_t - C_{t-1})}{C_{t-1}} \times Vt\right)$

In SMA formula C_t is price of an asset at a period n and n is the number of total periods. In ADL formula C_t is the closing price, L_t is low price for the period, H_t is high price for the period and V_t is the period volume. In CCI formula M_t is typical price, SM_t is simple moving average and D_t is mean deviation. In RSI formula U_p is upward price change, D_w is downward price change and n is the number of total periods. In TP H_t is high price, L_t is low price and C_t is closing price. In EFI C_t is current close price, C_{t-1} is prior close price and V_t is volume force index. In VPT VPT_{t-1} is previous VPT, C_t is current close price, C_{t-1} is prior close price and V_t is volume.

In Table 2.3 a detailed list of the technical indicators created during the pre-processingphases is given. Some of the indicators have several sub-indicators, that's why the sub-indicators are not included in the table mentioned.

Table 2. 3: Technical Indicators.

Indicator	***Formula***
SMA	Simple Moving Average
RSI	Relative Strength Index
CCI	Commodity Channel Index
DMI	Directional movement indicator
ADX	Average Directional Index
VPT	Volume Price Trend
EFI	Elder's Force Index
WOBV	Weighter OBV
VZO	Volume Zone Oscillator
PZO	Price Zone Oscillator
TP	Typical Price
ADL	Accumulation-Distribution Line
SMMA	Smoothed Moving Average
TR	True Range
SAR	Stop-and-Reverse
VWAP	Volume Weighted Average Price
SSMA	Smoothed Simple Moving Average
DEMA	Double Exponential Moving Average
TEMA	Triple Exponential Moving Average
TRIX	Exponential Moving Average Oscillator
CURRENCY	Usd/Try

date_time	Open	High	Low	Close	Volume	Sma2	Sma3	Sma4	Sma5	Sma6	Sma7	Rsi	Cci	Up_move	Down_move	Dmp
08/02/16 10:00 AM	0.016	0.014	0.015	0.014	0.003	0.014	0.013	0.013	0.013	0.013	0.013	0.319	0.357	0.486	0.245	0.003
08/02/16 11:00 AM	0.015	0.013	0.013	0.012	0.003	0.013	0.013	0.013	0.013	0.013	0.013	0.187	0.286	0.482	0.248	0
08/02/16 12:00 PM	0.013	0.011	0.013	0.013	0.001	0.012	0.012	0.013	0.013	0.012	0.013	0.319	0.293	0.479	0.244	0
08/02/16 02:00 PM	0.013	0.012	0.012	0.01	0.004	0.011	0.011	0.012	0.012	0.012	0.012	0.196	0.264	0.485	0.246	0
08/02/16 03:00 PM	0.011	0.01	0.011	0.01	0.001	0.01	0.011	0.011	0.011	0.011	0.011	0.196	0.262	0.479	0.245	0
08/02/16 04:00 PM	0.011	0.01	0.011	0.01	0.004	0.01	0.01	0.01	0.01	0.011	0.011	0.196	0.288	0.483	0.243	0
08/02/16 05:00 PM	0.011	0.009	0.01	0.009	0.004	0.01	0.009	0.009	0.01	0.01	0.01	0.152	0.272	0.482	0.246	0
09/02/16 10:00 AM	0.01	0.008	0.007	0.008	0.003	0.008	0.009	0.009	0.009	0.009	0.009	0.106	0.227	0.482	0.253	0
09/02/16 11:00 AM	0.009	0.007	0.008	0.008	0.002	0.008	0.008	0.008	0.008	0.008	0.009	0.187	0.27	0.481	0.239	0
09/02/16 12:00 PM	0.009	0.007	0.009	0.008	0	0.008	0.007	0.008	0.008	0.008	0.008	0.176	0.294	0.483	0.241	0
09/02/16 02:00 PM	0.008	0.007	0.004	0.004	0.003	0.006	0.006	0.006	0.007	0.007	0.007	0.075	0.208	0.482	0.259	0
09/02/16 03:00 PM	0.004	0.003	0.003	0.003	0.006	0.003	0.004	0.005	0.005	0.006	0.006	0.066	0.209	0.472	0.246	0
09/02/16 04:00 PM	0.004	0.007	0.004	0.008	0.004	0.005	0.004	0.005	0.005	0.005	0.006	0.447	0.337	0.496	0.241	0.023
09/02/16 05:00 PM	0.009	0.008	0.008	0.007	0.003	0.007	0.005	0.005	0.005	0.005	0.005	0.425	0.402	0.488	0.23	0.006

date_time	Dmm	Adx	Vpt	Efi	Wobv	Vzo	Pzo	Tp	Adl	Smma	Tr	Sar	Vwap	Ssma	Dema	Tema	Trix	Currency
08/02/16 10:00 AM	0	0.827	0.007	0.001	0.00019	0.284	0.299	0.014	0.999	0.012	0.005	0.017	0.013	0.013	0.013	0.013	0.03	0.032
08/02/16 11:00 AM	0.006	0.823	0.007	0.001	0.00017	0.233	0.245	0.012	0.999	0.012	0.009	0.017	0.013	0.013	0.012	0.012	0.03	0.033
08/02/16 12:00 PM	0.001	0.823	0.007	0.001	0.00017	0.288	0.361	0.012	0.999	0.012	0.004	0.016	0.012	0.013	0.012	0.012	0.03	0.033
08/02/16 02:00 PM	0.004	0.835	0.007	0.001	0.00013	0.215	0.299	0.011	0.999	0.011	0.009	0.016	0.012	0.012	0.011	0.011	0.03	0.035
08/02/16 03:00 PM	0.002	0.85	0.007	0.001	0.00013	0.243	0.327	0.01	0.999	0.011	0.005	0.015	0.012	0.012	0.01	0.01	0.03	0.036
08/02/16 04:00 PM	0	0.863	0.007	0.001	0.00013	0.292	0.35	0.01	0.999	0.011	0.004	0.014	0.012	0.012	0.01	0.01	0.03	0.036
08/02/16 05:00 PM	0.003	0.88	0.007	0.001	0.00011	0.225	0.29	0.009	0.999	0.011	0.004	0.014	0.012	0.011	0.009	0.009	0.03	0.035
09/02/16 10:00 AM	0.013	0.905	0.006	0.001	0.00009	0.185	0.237	0.008	0.999	0.011	0.014	0.013	0.012	0.011	0.008	0.008	0.03	0.036
09/02/16 11:00 AM	0	0.927	0.007	0.001	0.00009	0.271	0.354	0.008	0.999	0.01	0.004	0.012	0.012	0.01	0.008	0.007	0.03	0.035
09/02/16 12:00 PM	0	0.947	0.007	0.001	0.00009	0.258	0.293	0.008	0.999	0.01	0.001	0.011	0.012	0.01	0.007	0.007	0.03	0.035
09/02/16 02:00 PM	0.02	0.97	0.006	0.001	0.00002	0.206	0.24	0.004	0.999	0.01	0.02	0.01	0.011	0.009	0.005	0.005	0.03	0.037
09/02/16 03:00 PM	0.003	0.992	0.006	0.001	0.00000	0.149	0.195	0.003	0.999	0.009	0.007	0.009	0.011	0.008	0.004	0.003	0.03	0.033
09/02/16 04:00 PM	0	0.935	0.006	0.001	0.00009	0.297	0.318	0.006	0.999	0.009	0.019	0.009	0.011	0.008	0.005	0.004	0.03	0.034
09/02/16 05:00 PM	0	0.868	0.006	0.001	0.00008	0.253	0.261	0.007	0.999	0.009	0.006	0.003	0.01	0.007	0.005	0.005	0.03	0.033

Figure 2. 1: *Stock ASELS data ready to feed into the model.*

In this thesis, only 3 selected stocks are used for comparison and illustration; AKBNK,AKENR and ASELS as they are the first companies in alphabetical order.

Figure 2.1 shows a glimpse of our data for ASELS stock which is ready to be forwarded into our model. The data as we retrieve from the Matriks Inc. has been gone through initial filtering, pre-processing and technical indicators creation phases. The representation of the data in this figure is the version that used in the model creation.

2.1.3 Data Exploratory Analysis

In data exploratory analysis section, several data exploratory and visualization tools in Python language are used in order to examine the relations between features. The tools we applied on our data are pandas_profiling and seaborn libraries. Inferring the general structure of data and to get a glimpse of data internals are our main objectives. The heat map report generated by pandas_profiling enable us to check the null values, anymissing values, most frequent values, minimum and maximum values of features andthe correlation between the features.

The correlation between technical indicators is given in Figure 2.2. The figure shows the correlation heat-map of our data features. It has also numeric values to represent the relationship between the features. The lighter the color gets or the number closer to 1 represents the strong relationship between the features. The darker colour and numbernot close to 1 represents the opposite.

In the heat map plot the negative relationship between the Accumulation Distribution Line parameter and most of the other features draws the attention at first sight. Most of the features have positive inter-correlation at a low level. When we check correlationsbetween the fundamental features which are open, close, high, low and volume and the other technical indicators, volume column has positive correlation with most of the features. In addition currency column that we include into the data has positive correlations between most of the other features.

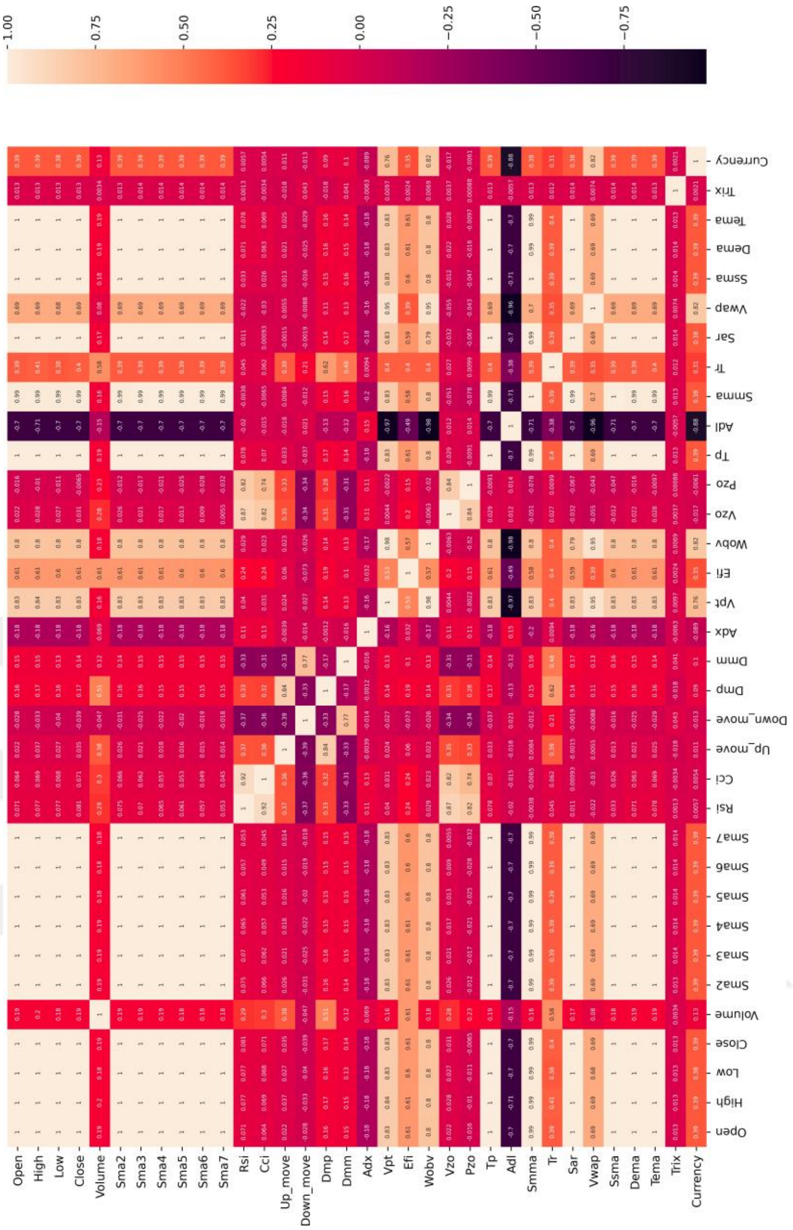


Figure 2. 2: Technical Indicators Correlation Plot.

2.2 METHODS

The methods in this work are selected based on our literature review, the research we conducted on the most recent technologies, and the domain-related requirements such as the type of data which is time-series in our work or gathering enough amount of qualified data etc. Conducting a technical analysis on BIST data and making successful predictions within intraday frequency are our aims. Both statistical and deep learning models are applied in this study. ARIMA is used as a base model from the statistical models perspective and LSTM, GRU, LSTM with attention mechanism models are used from the artificial neural networks perspective. ARIMA model is based on calculating the averages of subsets in the whole data and making predictions on the future trends. The neural network models are the RNN models which are widely used time-series data related applications. LSTM can be considered as an improvement of Vanilla RNN and GRU is a variant of LSTM. LSTM with an attention mechanism is an encoder-decoder model that utilizes attention mechanism.

2.2.1 AutoRegressive Integrated Moving Average (ARIMA)

AutoRegressive Integrated Moving Average model is a statistical used in time-series data analysis using past data to predict the future trends. The ARIMA model is based on ARMA (AutoRegressive Moving Average) model. The ARMA model consists of two parts AR (AutoRegressive) and MA (Moving Average) parts. AutoRegressive refers to a model considering the current value is based on the past values. Moving average is a calculation consisted of averages of different subsets of the whole data set. The main difference between ARIMA and ARMA is that ARIMA model converts the non-stationary data into stationary before it starts operating on it. The Akaike Information Criteria (AIC) is a used measure metric in a statistical model to compare models performances [59].

The AutoRegressive model formula as follows:

$$y_t = \sum_{i=1}^{p} a\,(i).\,y(t-i) + \epsilon(t)$$

where $a_1, a_2, \ldots, a_p$ coefficients of the recursive filter, p is the order of the model, $\epsilon(t)$ are output uncorrelated error or simply white noise.

The Moving Average model formula as follows:

$$MA = \frac{A_1 + A_2 + \ldots + A_n}{n}$$

where A refers to average in period n and n is the number of time periods.

The ARIMA model classified as ARIMA(p,d,q) model and p refers to autoregressivepart, d denotes the integrated part and q refers to the moving average part.

2.2.2 Artificial Neural Networks

Artificial neural networks are a special class of machine learning algorithms that mimic the human brain and widely used in computer vision problems, speech recognition, image processing etc.

RNNs, CNNs, Residual Neural Networks, Generative Adversarial Networks

(GAN) are some of the most common types of ANNs. RNNs are able to make sense of information that occurs across time. CNNs are widely applied to computer vision problems. They are specialized to take advantage of the particular local structure of the windowed images. Residual neural networks are an innovation that have made it much easier to train deep networks. GANs specialized in modelling complex probability distributions. For example if we feed excessive kitchen images to these models, then they would be able to generate novel images similar to the ones in the training but not exactly the same. Figure 2.3 shows the input and hidden layers of an ANN.

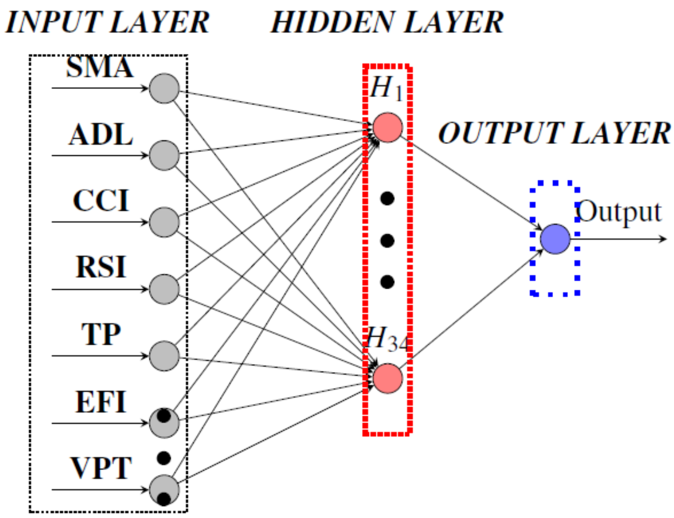


Figure 2. 3: The overall structure of ANN.

RNNs are used to solve problems that deal with sequence information. In a standard feed-forward neural network, a neuron aggregates the input it takes, and then it passes the input through an activation function (ReLU, Sigmoid, Tanh, etc.) to produce an output. In a recurrent neural network, a neuron sends the output back to itself. If a recurrent neuron is unrolled, the recurrent neuron's input at T-1 gives output at T-1 and then gets passed into the neuron in its state at time t and then has an output at time t and so on. Recurrent neuron takes both inputs from a previous time step as well as inputs from the current timestep. In RNNs, these neurons that take inputs from previous timesteps are also known as "memory cells". LSTM and GRU are both sub-classes of RNNs and in the following sections these two types of networks will be explained in detail.

2.2.2.1 Long Short-Term Memory (LSTM)

One of the critical problems with analyzing larger time-series data sets with recurrent neural networks is the vanishing gradient. Backpropagation goes backward from the output to the input layer, propagating the error gradient. For deeper networks, issues can arise from backpropagation, sometimes called vanishing or exploding gradients. As we traverse back to the lower level layers or front layers closer to the input layers, gradients often get smaller, eventually causing neurons' weights to never change at these lower levels. A different activation function or batch normalization techniques might be proposed as solutions for vanishing gradient problems at a scale. However, due to the length of the sequence input, such as a long time-series data, these proposed solutions could slow down the training.

Another issue RNNs face is that the network will begin to "forget" the first inputs aftera while, as information is lost at each step going through the RNN. At that point, a long-term memory solution is needed for recurrent neural networks. LSTM cell was created to help to address the vanishing gradient issue in 1997 [19]. LSTM cell was introduced with three different gates, input gate, forget gate, and output gate. Input gateto decide what to store in cell state, forget gate is where to determine what information is going to forget or throw away from the cell state, output gate is to decide what to update in the cell state [19][9].

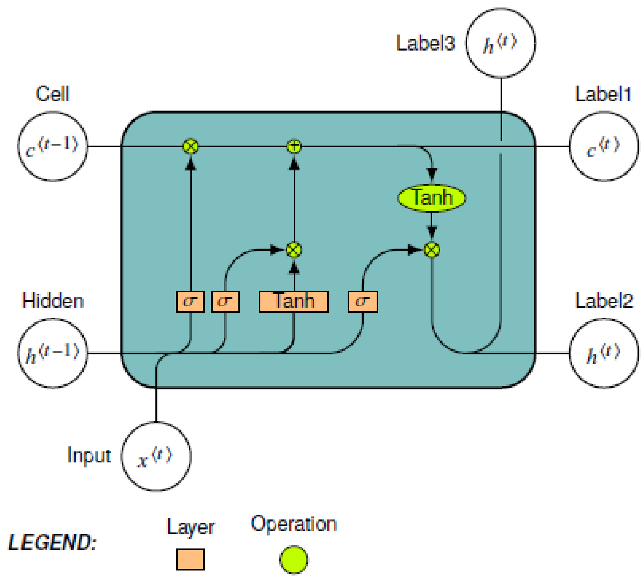


Figure 2. 4: The overall structure of LSTM.

In Figure 2.4 the inputs are current input, cell state and the hidden state. x_t is the current input, c_{t-1} is cell state which is the memory from the last LSTM unit, h_{t-1} is hidden state which is output of the last LSTM unit. The outputs are Label1, Label2, Label3 inthe illustration. h_t is the current output which is called Label3 in the figure. c_t is the new updated memory. It can be considered as the next cell state and it is called Label1in the figure. X is the scaling information operation and $+$ is the adding information operation. σ and $Tanh$ refer to sigmoid and hyperbolic tangent functions.

In the detailed equations i_t is the input gate, f_t is the forget gate and o_t is the output gate.

W is the connection between the previous hidden state and current hidden state. U is the weight matrix between the inputs and the hidden layer. $\widetilde{C_t}$ is the candidate hidden state that is calculation of current input and the previous hidden state. C_t is the internal memory of the cell. C_t is computed based on the previous memory, multiplied by the forget gate, new hidden state and the input gate.

Detailed LSTM equations as follows:

$$
\begin{aligned}
i_t &= \sigma\left(x_t U^i + h_{t-1} W^i\right) \\
f_t &= \sigma(x_t U^f + h_{t-1} W^f) \\
o_t &= \sigma(x_t U^o + h_{t-1} W^o) \\
\widetilde{C_t} &= tanh(x_t U^g + h_{t-1} W^g) \\
C_t &= \sigma\left(f_t \times C_{t-1} + i_t \times \widetilde{C_t}\right) \\
h_t &= tanh(C_t) \times o_t
\end{aligned}
$$

2.2.2.2 Gated Recurrent Unit (GRU)

A variant of the LSTM cell is called the GRU introduced in 2014 [10]. GRU simplifiesthe internal structure of LSTM by combining the forget and input gates into a single gate called "update gate". It also merges the cell state and hidden state. The update gate is used to decide what information is to be passed, reset gate is used to determinewhat information is going to forget. These gates stand for the long term and short term memory, respectively.

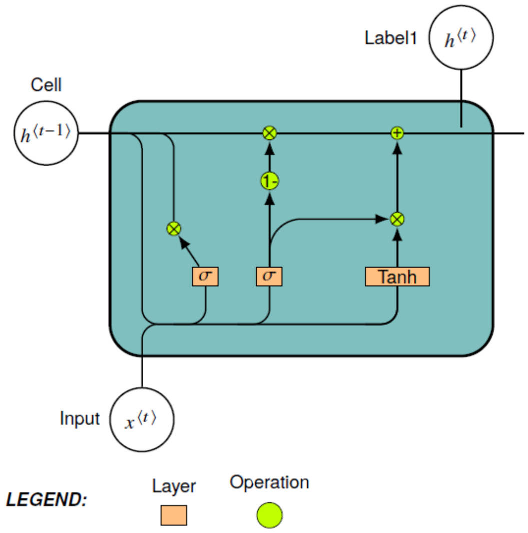


Figure 2. 5: The overall structure of GRU.

GRU merges the current cell state and hidden state of LSTM into one state as the output of the previous stage. In Figure 2.5 the inputs are current input and cell state. x_t is the current input, h_{t-1} is output of the previous stage. h_t is the current output which is called Label1 in the figure. GRU preserves the functions and most of the operations ofLSTM as a variant of LSTM.

In the detailed equations z_t denotes the update gate and r_t refers to the reset gate. $\widetilde{h}_t$ iscandidate hidden state and h_t is the hidden state. W is the weight matrix.

Detailed GRU equations as follows:

$$
\begin{aligned}
z_t &= \sigma(W_z . [h_{t-1}, x_t]) \\
r_t &= \sigma(W_r . [h_{t-1}, x_t]) \\
\widetilde{h}_t &= tanh(W . [r_t \times h_{t-1}, x_t]) \\
h_t &= (1 - z_t) \times h_{t-1} + z_t \times \widetilde{h}_t
\end{aligned}
$$

2.2.2.3 LSTM with Attention Mechanism

The attention mechanism in deep learning focuses on certain features of data during the training process. One of the most frequently use cases of attention mechanism is seq2seq models, also known as encoder-decoder model[10].

Sequence to sequence deep learning models are considered as recurrent neural networks widely used in natural language processing, recognition, and applications that use time-series data. Encoders and Decoders can be either LSTM or GRU models. In thismechanism, the encoder passes the input data of the internal states to the decoder, andthe decoder generates the output based on the internal states of input data acquired from the encoder.

In the seq2seq model with attention mechanism, the decoder considers internal states of the entire sequence while producing the output instead of the last state of the encoder, which is the standard seq2seq model behavior. The attention mechanism allows the decoder to access the entire data while focusing on specific features to produce the output.

In [36] the authors distinguish the attention based neural networks into two broad categories which are global and local attention models. The difference between these two models relies on the way of handling the encoder's output. In global attention all the hidden states of encoder are taken into consideration which means that all of the features are given importance. However, in local attention selected subsets of hidden states are used in producing the output.

Bahdanau and Luong are two common models of attention that are widely used to address time-series data application problems [45] [9] [10]. The main difference between these two mechanisms is the way of calculating the score similarities. Luong attentionuses simple matrix multiplications, and this makes it faster and more space-efficient. Both models can be regarded as global attention models which use all the hidden state information while producing the output called "context vector" in attention mechanism[36]. In addition to that Luong attention can also be applied as local attention mechanism.

The following equations show the difference between Bahdanau and Luong's way of calculating the score similarities. h_s denotes the hidden states of the input data, which is passed to the decoder from the encoder. h_t refers to the previous output of the decoder. c_t is the context vector, and W_t is the weight matrix. We can infer these scores as the level of the relationship between the encoder's all hidden states and previous decoder's output.

$$score(h_t, \widetilde{h_s}) = \begin{cases} h_t^T W \widetilde{h_s} & \text{Luong} \\ v_a^T tanh(W_1 h_t + W_2 \widetilde{h_s}) & \text{Bahdanau} \end{cases}$$

These scores are used to calculate the attention weights as the following equations:

$$a_{ts} = \frac{exp\left(score(h_t, \widetilde{h_s})\right)}{\sum_{s'=1}^{S} e\,xp\left(score(h_t, \widetilde{h'_s})\right)}$$

Context vector is calculated using the attention weights as the following:

$$c_t = \sum_s a_{ts} \widetilde{h_s}$$

The phases except calculating the score similarities are the same for both Bahdanau and Luong attention mechanisms.

In this thesis, we have used the Luong global attention mechanism, which is introduced in 2015 [36]. The attention mechanism mainly differs from the other models while assigning weights to the neurons in the network. It makes the network consider the features of data by adjusting the weights[36].

2.3 EVALUATION

In RNNs, MSE and MAE metrics are widely used metrics as regression metrics. Mean Squared Error is the average squared error between the prediction results and true values. Mean Absolute Error is average distance between the prediction results and true values. Loss function is the calculating an error value while comparing the model's intermediary results and true values. Then the weights of the neuron are updated accordingly for thenext epoch and so on.

2.3.1 Model Performance Metrics

We consider MSE and MAE metrics while comparing all three models. It is importantto have loss value decreasing smoothly as the number of epochs increases. You may notice the loss functions' formulas used in our models in Table 2.4

Table 2. 4: Loss Functions Used in Our Models.

Loss Functions	
Function	***Formula***
Mean squared error	$\frac{1}{n}\sum_{t=1}^{n} e_t^2$
Mean absolute error	$\frac{1}{n}\sum_{t=1}^{n} \lvert e_t \rvert$

CHAPTER 3

EXPERIMENTS AND RESULTS

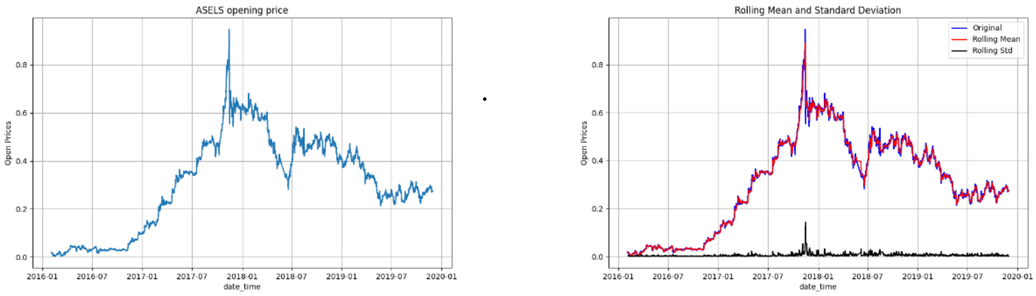
3.1 CREATING THE MODEL

The final data has 34 columns. Open, High, Low, Close, and Volume columns, 28 technical indicators, and currency columns. We read the data into a dataframe then weselect the open price column as the prediction column. We split the data into train andtest parts. Then we turn the dataframe into NumPy arrays and time-series format to make them eligible for supervised learning. Our aim is to predict the open price by using past hours of data called the time step. For example, the runs with 30 as the time step, the input is the 30 hours of data, and the output is 31st open price data. We pass the time step and the index of the prediction column to the "build time series" function and get an output of two NumPy arrays, which are the inputs and output. We also need to define another parameter called batch size before we feed the data to the model. Batch size is the number of inputs the network process before it updates the weights. There is a trade-off between the speed of the network training and its ability to generalize the data. Then trim the data to make its size divisible by the batch size.

3.1.1 ARIMA Model

We have implemented ARIMA model on AKBNK, AKENR and ASELS stocks. Firstwe made a stationary check using Augmented Dickey Fuller (ADF) test to decide on whether the data is stationary or not stationary. Then we applied seasonality decompose on our data. We split the data into two parts of train and test and we used auto_arima function to conduct a grid search on ARIMA model hyperparameters to decide on the best parameters. At the end we have used MSE and MAE to evaluate our models.

Figure 3.1 a plots the open price of ASELS over time. We have used open price as prediction column on ARIMA model. Figure 3.1b depicts the mean and standard deviation for ASELS indicating that our data is not stationary because the mean and



(a) ASELS Data Open Price Distribution. **(b)** Mean and Standard Deviation for ASELS.

Figure 3. 1: ASELS Open Price

standard deviation are not flat lines. The Figure 3.1b should be considered together with our results of ADF test in Table 3.1. According to the test results, we cannot apply Null hypothesis due to p-value is bigger than 0.05 and test statistics value exceeds the critical values. As a result our data is non-linear and not-stationary.

Table 3. 1: ADF Results for ASELS.

ADF Results	
Function	***Value***
Test Statistics	-1.665738
p-value	0.448846
Number of lags used	35
Number of observations used	6484
critical value (1%)	-3.431359
critical value (5%)	-2.861986
critical value (10%)	-2.567007

We decomposed the data to separate the trend and the seasonality and Figure 3.2 shows the plots regarding the seasonality decompose. Figure 3.3 shows the train and test datasplit. We applied auto_arima grid search to decide the best parameters for our model. Figure3.4 depicts the results of grid search phase. The top left shows the residual error, the top right shows the density plot suggesting a normal distribution with a mean of zero. In bottom left if there is a significant deviation it means a skewed distribution. The bottom right shows the residual errors which are not auto-correlated. The Figure 3.4 should be regarded together with Table 3.2. According to the grid search results the best performance-giving model is ARIMA(2,1,0)(0,0,0)[0].

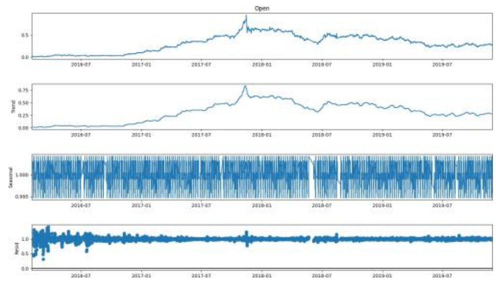


Figure 3. 2: ASELS Seasonality and Trend Decompose.

Figure 3. 3: ASELS Data Train and Test Split.

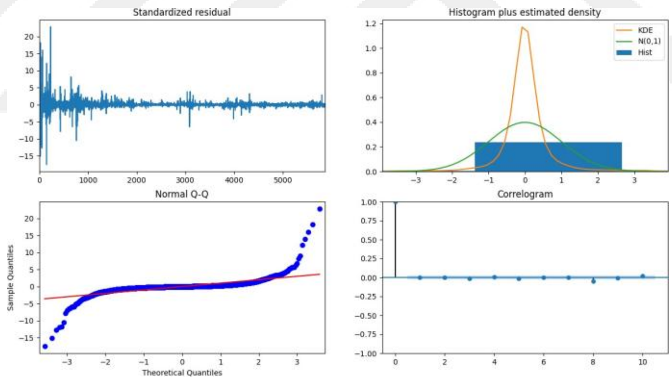


Figure 3. 4: ASELS Grid Search Results.

Figure 3.5 shows the stock price prediction for ASELS using ARIMA model. We have implemented the same phases of ARIMA on other stocks AKBNK and AKENR.AKBNK and AKENR plots are in Figure 3.6 and Figure 3.7 respectively.

Table 3. 2: ARIMA Grid Search Results for ASELS.

Grid Search Results		
Model	***AIC***	***Time***
(0,1,0)(0,0,0)[0] intercept	21590.937	0.49
(1,1,0)(0,0,0)[0] intercept	21588.951	0.44
(0,1,1)(0,0,0)[0] intercept	21588.940	0.36
(0,1,0)(0,0,0)[0]	21592.212	0.20
(1,1,1)(0,0,0)[0] intercept	21603.205	2.62
(2,1,1)(0,0,0)[0] intercept	21635.972	0.63
(2,1,0)(0,0,0)[0] intercept	21637.304	1.15
(3,1,0)(0,0,0)[0] intercept	21636.100	1.26
(3,1,1)(0,0,0)[0] intercept	21634.100	0.52
(2,1,0)(0,0,0)[0]	**21638.425**	**0.84**
(1,1,0)(0,0,0)[0]	21590.228	0.25
(3,1,0)(0,0,0)[0]	21637.200	0.48
(2,1,1)(0,0,0)[0]	21637.072	0.45
(1,1,1)(0,0,0)[0]	21588.686	0.32
(3,1,1)(0,0,0)[0]	21635.200	0.55

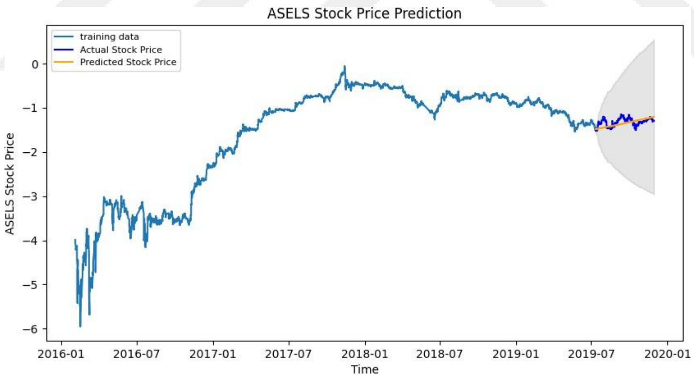


Figure 3. 5: ASELS Stock Price Prediction Using ARIMA.

We have compared AKBNK, AKENR and ASELS stock in terms of the MSE and MAE values in Table 3.3. ARIMA model gives best performance on ASELS stock accordingto our experimental tests.

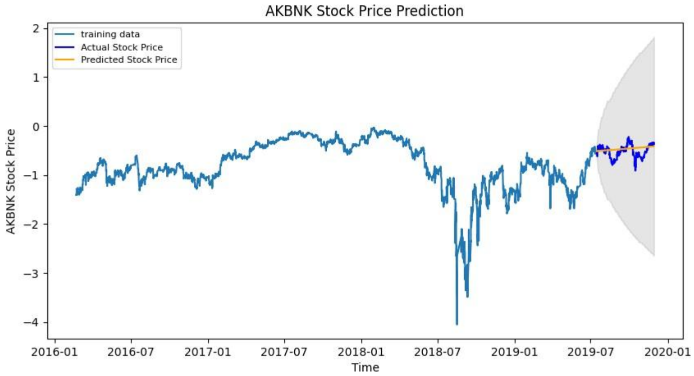


Figure 3. 6: AKBNK Stock Price Prediction Using ARIMA.

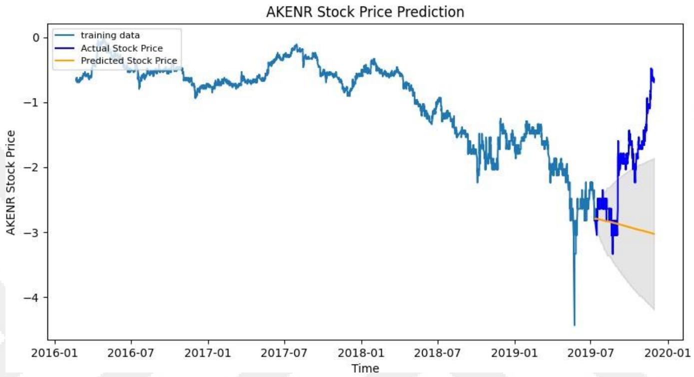


Figure 3. 7: AKENR Stock Price Prediction Using ARIMA.

Table 3. 3: Comparison Results for ARIMA.

Comparison Results		
Stock	***MSE***	***MAE***
ASELS	***0.010564***	***0.083637***
AKBNK	0.019475	0.106687
AKENR	1.405669	0.969286

3.1.2 RNN Results for Three Stocks

LSTM network needs to have an input as [batch size, time step, features]. We feed the data to create the model. Then we use the model to make predictions. We plot the comparison results. We choose different hyperparameters such as the number of epochs, loss function, learning rate, and optimizer for our runs. We have tested different combinations of hyperparameters.

The data preparation and pre-processing phases are the same for all three models. LSTM and GRU models have 7 layers: input layer, 2 hidden layers, and a dense output layer. There are dropout layers between these layers. We use 0.2 as a dropout ratio, and the number of features is 34 in our models. We use the hyperbolic tangent function as a loss function.

In LSTM with the attention layer model, we use the same hyper-parameter values and2 LSTM cells, one as an encoder and the other one as a decoder. We use a truncate custom function similar to the ’build time series function in the other two models to turn the data into a time-series format. An attention layer is added to this model after the LSTM layers. We use the Luong attention mechanism.

In all three models, we have used ’30’ as a time step, which means we operate 30 hoursof past data to predict the next hour. We choose 0.0001 as the learning rate in both LSTM and GRU setups. When we increase the learning rate, in the training process the loss value becomes Not a Number (NaN) at the beginning of the training. Then, we have to lower the learning rate in order to prevent the network from memorizing insteadof learning hidden patterns.

We use the adam optimizer in both our implementations. We made experiments for both adam and rmsprop optimizers. We choose adam over rmsprop hence it slightly outperforms the rmsprop. Our batch size is 20 for both GRU and LSTM implementations. We have made several runs and different combinations of hyperparameters. Then we choose 20 as batch size. Number of epochs is a similar parameter to number of epochs in terms of the trade-off between the speed of network training and ability to generalize the data. Number of epochs that we use choose on our experiments 50, 100 and 150. Our test environment is Ubuntu 18.04 and our graphics card is GeForce GTX 1050 Ti Mobile. We have trained and tested our models on Tensorflow Keras environment. Our Keras version is 2.4.0. All our codes are in python language.

In this study, we have compared three different stocks from Borsa Istanbul with three different models LSTM, GRU, and LSTM with an Attention mechanism. We have run our models multiple times with different hyperparameters combination. We used both MSE and MAE values to compare our models. Our stocks are AKBNK, AKENR and ASELS from three different fields of business.

Table 3. 4: Comparison Results for Stock ASELS.

Comparison Results						
Model	***Epoch***	***Batch***	***TimeStep***	***Opt***	***MSE***	***MAE***
LSTM	50	20	30	adam	0.0014	0.0257
GRU	50	20	30	adam	0.0013	0.0267
ATTENTION	50	20	30	adam	0.0471	0.1724
LSTM	100	20	30	adam	0.0011	0.0227
GRU	100	20	30	adam	0.0010	0.0223
ATTENTION	100	20	30	adam	0.0460	0.1703
LSTM	150	20	30	adam	0.0009	0.0213
GRU	150	20	30	adam	**0.0008**	**0.0205**
ATTENTION	150	20	30	adam	0.0456	0.1670

Table 3.4 shows the comparison results of ASELS stock data with 3 different models.For all three number of epochs of 50, 100 and 150 GRU outperforms the other models. LSTM slightly falls behind the GRU. For all three models increasing the number of epochs results in better error loss values.

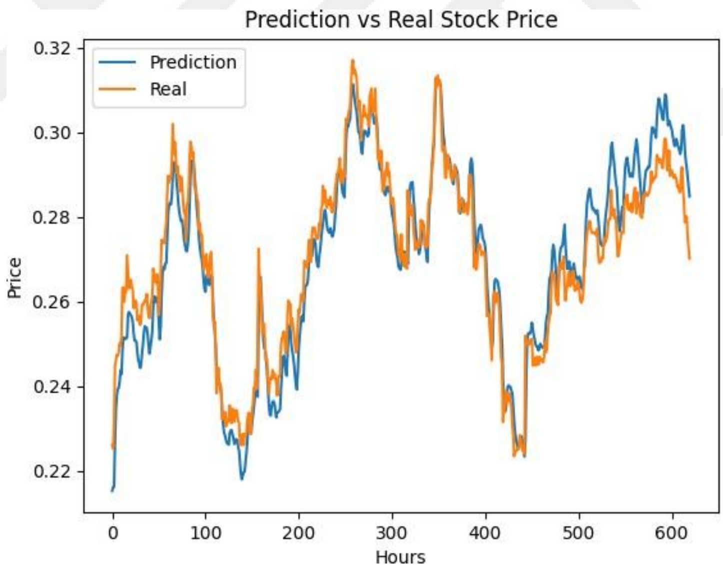


Figure 3. 8: Stock ASELS GRU result figure for 50 epochs.

Figure 3.8 shows prediction and real data graphics of GRU with 50 epochs on ASELSstock data. It generalize the data almost perfect and it has an MSE value of 0.0013.

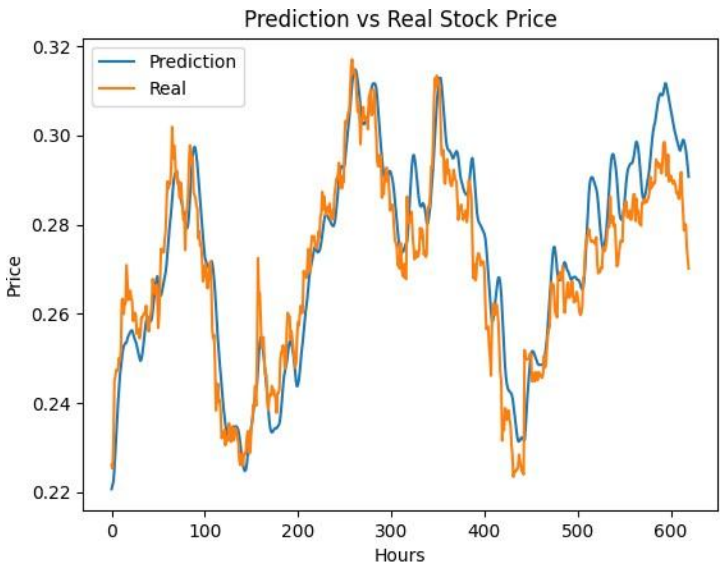


Figure 3. 9: Stock ASELS LSTM result figure for 50 epochs.

Figure 3.9 shows prediction and real data graphics of LSTM with 50 epochs on ASELS stock data. It generalize the data with very few deficits and it has an MSE value of 0.0014. LSTM network performance slightly falls behind the GRU model.

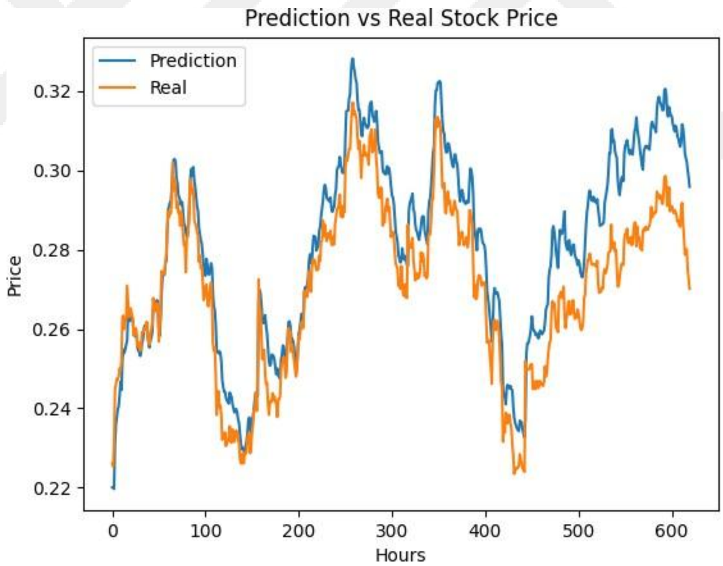


Figure 3. 10: Stock ASELS GRU result figure for 150 epochs.

Figure 3.11 shows residuals graphics of GRU model with 150 epochs on ASELS stockdata.

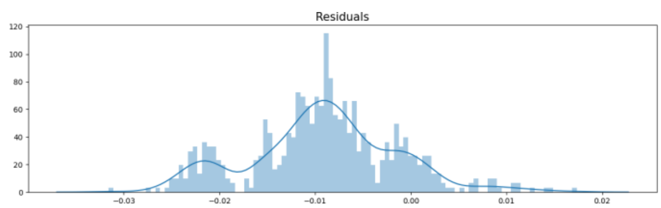


Figure 3. 11: Residuals of figure ASELS GRU result for 150 epochs.

Figure 3.10 shows prediction and real data graphics of GRU with 150 epochs on ASELS stock data. It generalize the data almost perfect and it has an MSE value of 0.0008. Increasing the number of epochs results better in prediction performance both in the graphics and the error loss value. It gives the best performance over LSTM and LSTM with attention layer models on ASELS stock data.

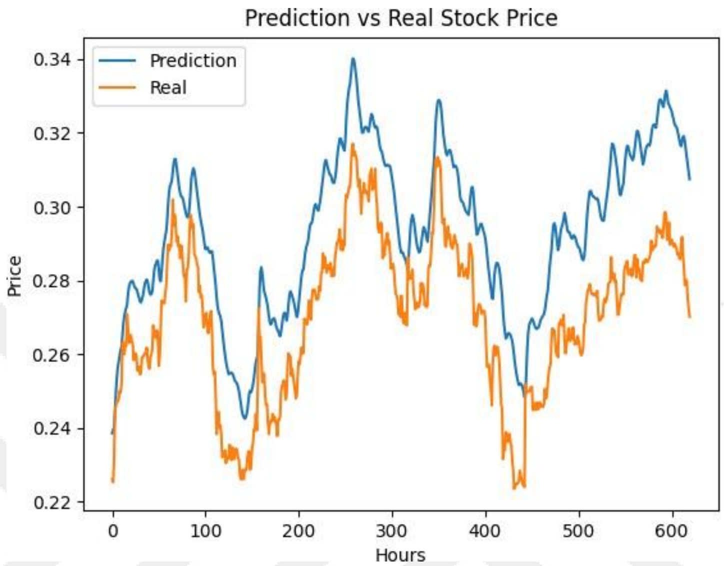


Figure 3. 12: Stock ASELS LSTM result figure for 150 epochs.

Figure 3.12 shows prediction and real data graphics of LSTM with 150 epochs on ASELS stock data. It generalize the data with very few deficits and it has an MSE value of 0.0009. Increasing the number of epochs results better in prediction performance both in the graphics and the error loss value, but again it falls behind the GRU's performance.

For all three number of epochs of 50, 100 and 150 GRU outperforms the other models.

Table 3.5 shows the comparison results of AKBNK stock data with 3 different models.For all three number of epochs of 50, 100 and 150 GRU outperforms the other models.

Table 3. 5: Comparison Results for Stock AKBNK.

Comparison Results						
Model	***Epoch***	***Batch***	***TimeStep***	***Opt***	***MSE***	***MAE***
LSTM	50	20	30	adam	0.0040	0.0467
GRU	50	20	30	adam	0.0029	0.0385
ATTENTION	50	20	30	adam	0.0489	0.1823
LSTM	100	20	30	adam	0.0030	0.0383

Table 3.5. (Cont.)

GRU	100	20	30	adam	0.0019	0.0301
ATTENTION	100	20	30	adam	0.0467	0.1802
LSTM	150	20	30	adam	0.0021	0.0317
GRU	150	20	30	adam	**0.0014**	**0.0257**
ATTENTION	150	20	30	adam	0.0484	0.1817

GRU overperforms the other two models. LSTM slightly falls behind the GRU. For allthree models increasing the number of epochs results in better error loss values.

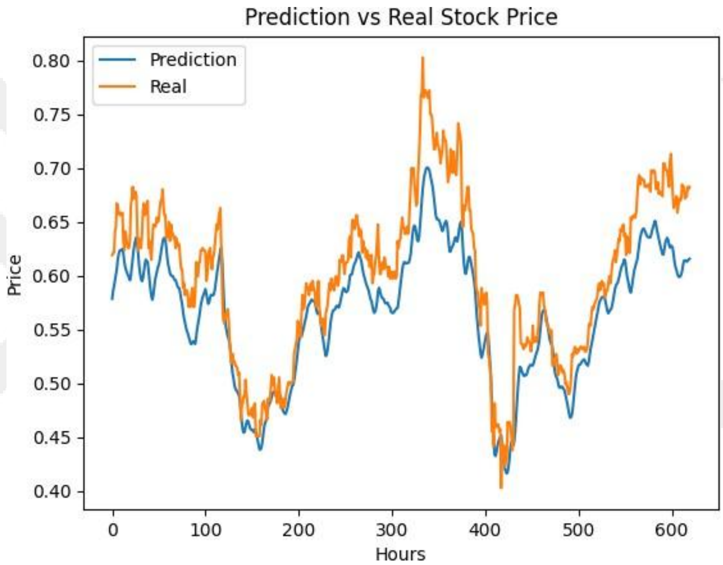


Figure 3. 13: Stock AKBNK LSTM result figure for 50 epochs.

Figure 3.13 shows prediction and real data graphics of LSTM with 50 epochs on AKBNK stock data. It generalize the data with very few deficits and it has an MSE value Figure 3.13 shows prediction and real data graphics of LSTM with 50 epochs on AKBNK stock data. It generalize the data with very few deficits and it has an MSE value of 0.0040.

Figure 3.14 shows prediction and real data graphics of LSTM with 150 epochs on AKBNK stock data. It generalize the data with very few deficits and it has

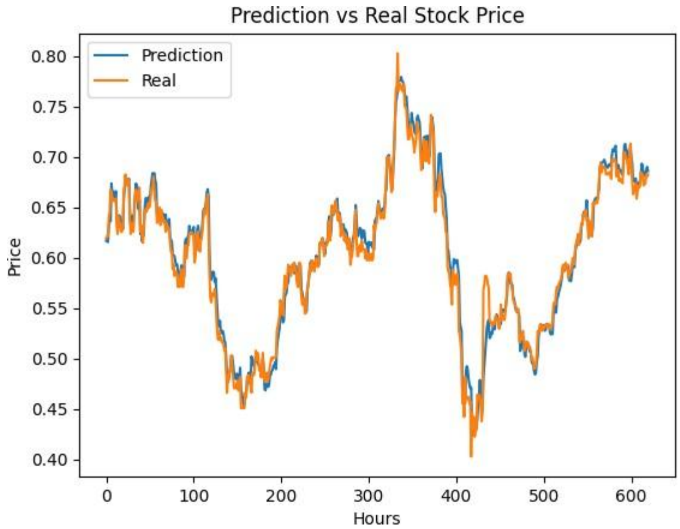


Figure 3. 14: Stock AKBNK LSTM result figure for 150 epochs.

an MSE value of 0.0021. It has better performance when it is compared to the LSTM with 50 epochs' results.

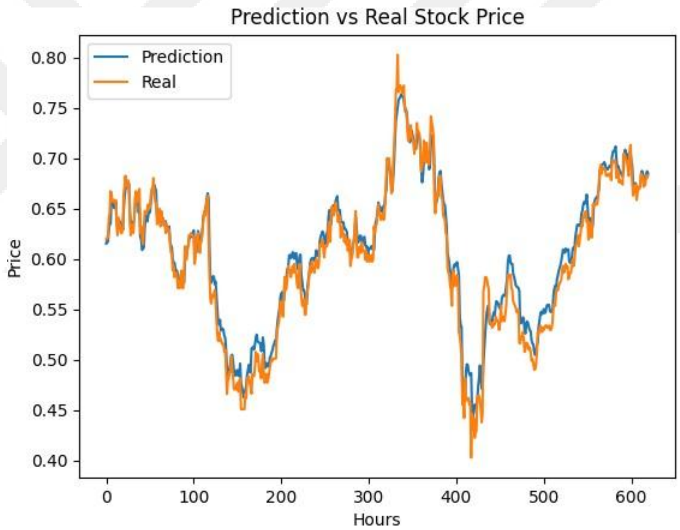


Figure 3. 15: Stock AKBNK GRU result figure for 50 epochs.

Figure 3.15 shows prediction and real data graphics of GRU with 50 epochs on AKBNK stock data. It generalize the data almost perfect and it has an MSE value of 0.0029. It has better performance in comparison to the LSTM model with the same parameters.

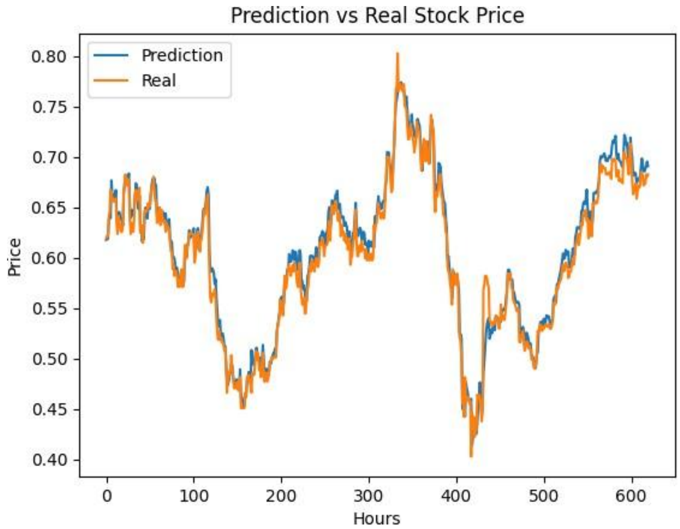


Figure 3. 16: Stock AKBNK GRU result figure for 150 epochs.

Figure 3.16 shows prediction and real data graphics of GRU with 150 epochs on AKBNK stock data. It generalize the data almost perfect and it has an MSE value of 0.0014. Increasing the number of epochs results better in prediction performance both in the graphics and the error loss value. It gives the best performance over LSTM and LSTM with attention layer models on AKBNK stock data.

Table 3.6 shows the comparison results of AKENR stock data with 3 different models. GRU overperforms the other two models. LSTM slightly falls behind the GRU. For allthree models increasing the number of epochs results in better error loss values.

Table 3.7 shows the comparison results for ASELS, with timestep value 20. LSTM slightly falls behind the GRU. Table 3.8 shows the comparison results for AKBNK with timestep value 20. LSTM slightly falls behind the GRU. Table 3.9 shows the comparison results AKENR with timestep value 20. Figure 3.17 shows prediction and real data graphics of LSTM with 50 epochs on AKENRstock data. It generalize the data with very few deficits and it has an MSE value of 0.0040. Figure 3.18 shows prediction and real data graphics of LSTM with 150 epochs on AKENR stock data. Increasing the number of epochs results in better performance and it has an MSE value of 0.0019.

Figure 3.19 shows prediction and real data graphics of GRU with 50 epochs on AKENR stock data. It generalize the data almost perfect and it has an MSE value of 0.0037.

Figure 3.20 shows prediction and real data graphics of GRU with 150 epochs on AKENR stock data. It generalize the data almost perfect and it has an MSE value of

0.0017. Increasing the number of epochs results better in prediction performance both

Table 3. 6: Comparison Results for Stock AKENR.

Comparison Results						
Model	***Epoch***	***Batch***	***TimeStep***	***Opt***	***MSE***	***MAE***
LSTM	50	20	30	adam	0.0040	0.0430
GRU	50	20	30	adam	0.0037	0.0421
ATTENTION	50	20	30	adam	0.0597	0.2057
LSTM	100	20	30	adam	0.0029	0.0356
GRU	100	20	30	adam	0.0024	0.0339
ATTENTION	100	20	30	adam	0.0550	0.1991
LSTM	150	20	30	adam	0.0019	0.0290
GRU	150	20	30	adam	**0.0017**	**0.0284**
ATTENTION	150	20	30	adam	0.0530	0.1959

Table 3. 7: Comparison Results for Stock ASELS with TimeStep 20.

Comparison Results						
Model	***Epoch***	***Batch***	***TimeStep***	***Opt***	***MSE***	***MAE***
LSTM	50	20	20	adam	0.0015	0.0268
GRU	50	20	20	adam	0.0013	0.0261
ATTENTION	50	20	20	adam	0.0444	0.1724
LSTM	100	20	20	adam	0.0011	0.0229
GRU	100	20	20	adam	0.0010	0.0220
ATTENTION	100	20	20	adam	0.0430	0.1703
LSTM	150	20	20	adam	0.0009	0.0207
GRU	150	20	20	adam	**0.0008**	**0.0206**
ATTENTION	150	20	20	adam	0.0405	0.1670

in the graphics and the error loss value. It gives the best slightly the best performance over LSTM and LSTM with attention layer models on AKENR stock data.

Table 3. 8: Comparison Results for Stock AKBNK with TimeStep 20.

Comparison Results						
Model	***Epoch***	***Batch***	***TimeStep***	***Opt***	***MSE***	***MAE***
LSTM	50	20	20	adam	0.0041	0.0471
GRU	50	20	20	adam	0.0030	0.0393
ATTENTION	50	20	20	adam	0.0505	0.1840
LSTM	100	20	30	adam	0.0028	0.0374
GRU	100	20	20	adam	0.0019	0.0299

Table 3.8. (Cont.)

ATTENTION	100	20	20	adam	0.0467	0.1792
LSTM	150	20	20	adam	0.0021	0.0316
GRU	150	20	20	adam	**0.0014**	**0.0257**
ATTENTION	150	20	20	adam	0.0464	0.1785

Table 3. 9: Comparison Results for Stock AKENR with TimeStep 20.

Comparison Results						
Model	***Epoch***	***Batch***	***TimeStep***	***Opt***	***MSE***	***MAE***
LSTM	50	20	20	adam	0.0042	0.0430
GRU	50	20	20	adam	0.0043	0.0427
ATTENTION	50	20	20	adam	0.0597	0.2055
LSTM	100	20	20	adam	0.0024	0.0336
GRU	100	20	20	adam	0.0023	0.0341
ATTENTION	100	20	20	adam	0.0550	0.1992
LSTM	150	20	20	adam	0.0019	0.0293
GRU	150	20	20	adam	**0.0020**	**0.0301**
ATTENTION	150	20	20	adam	0.0539	0.1987

3.1.3 LSTM with Attention Mechanism Results for Three Stocks

In this paper, we only include figures for the results of attention model with ASELS stock for 150 epochs, but in our experimental results other stocks give similar results. Figure 3.21 and Figure 3.22 shows the MSE comparison results of attention model for 50 and 100 epochs respectively. The three featured graphics of MSE comparison for attention model show that the error is saturated around the 150 epochs. The behaviouris also very similar in other models. So, we only give results for 50, 100 and 150 epochs in this paper.

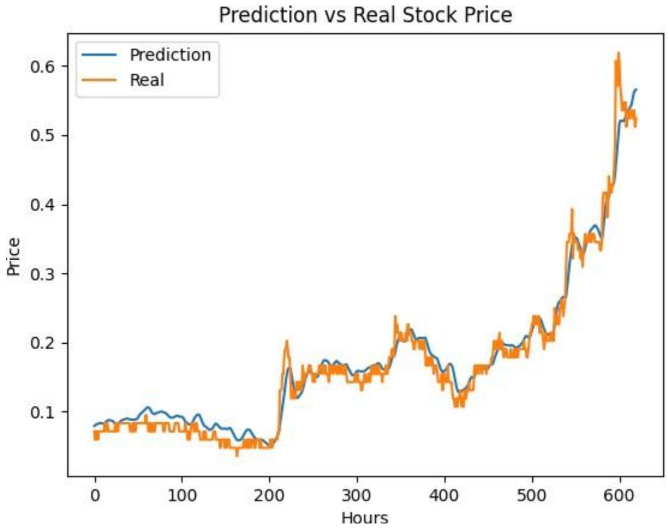


Figure 3. 17: Stock AKENR LSTM result figure for 50 epochs.

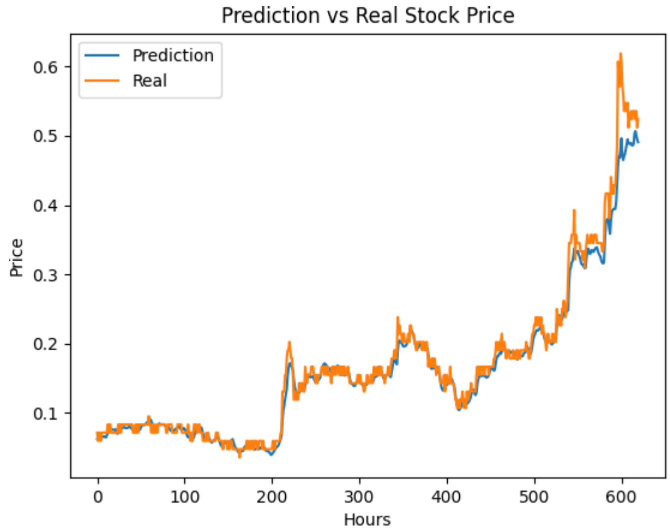


Figure 3. 18: Stock AKENR LSTM result figure for 150 epochs.

As we compare the attention results with other models we can observe that the error value is higher than other models. The reason relies on the result can be related to the Luong global attention mechanism. In [36] and [5] the authors remarks that the global attention mechanism could have difficulties for processing long sequences of data due to the its way of calculating the context vector. In global attention mechanism all the hidden states of encoder are considered in producing context vector and it requires much training to obtain good results. Increasing the epoch size or applying local attention mechanism might affect the results in a positive manner.

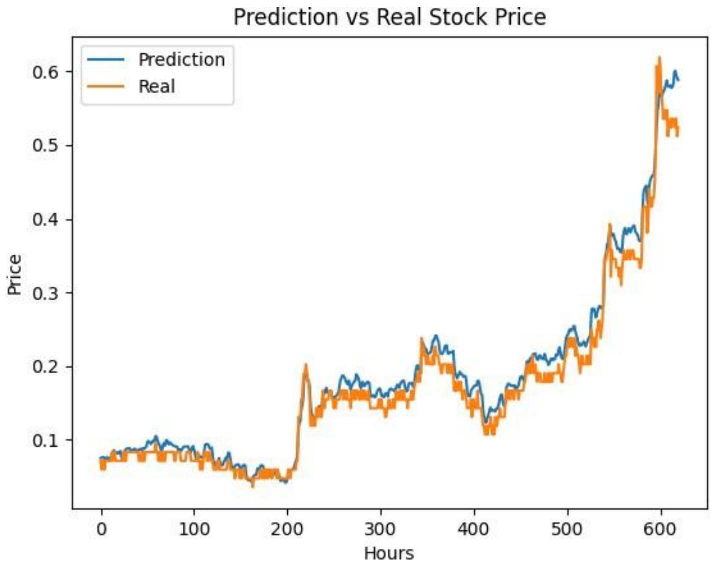


Figure 3. 19: Stock AKENR GRU result figure for 50 epochs.

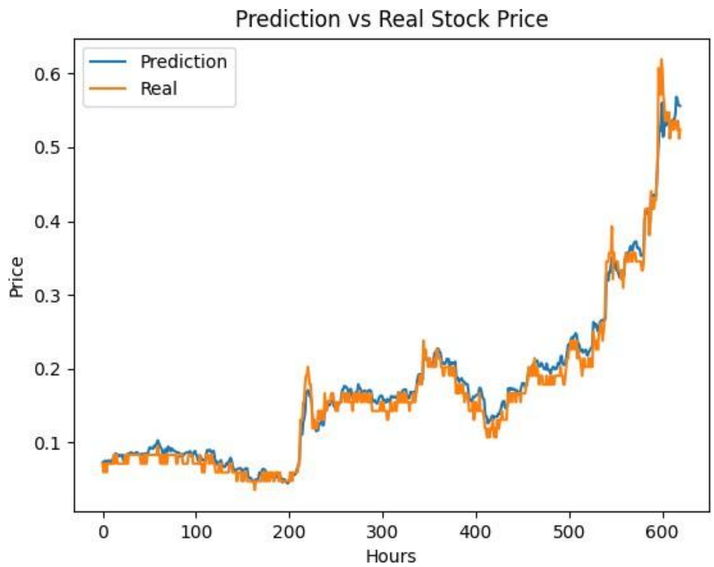


Figure 3. 20: Stock AKENR GRU result figure for 150 epochs.

Figure 3.21 shows the MSE values for the attention model through the number of epochs for the 50 epochs run on ASELS data. The validation MSE value has waves along thetrain MSE value.

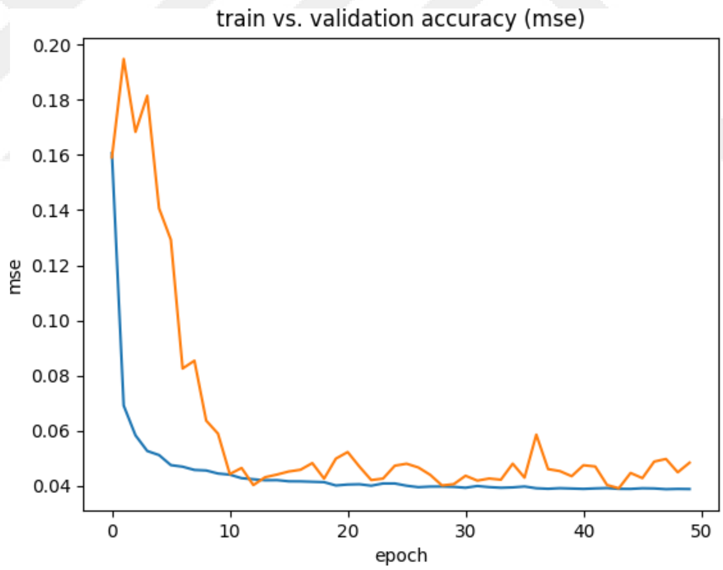


Figure 3. 21: LSTM with Attention Mechanism Train vs. Validation MSE for 50 epochs.

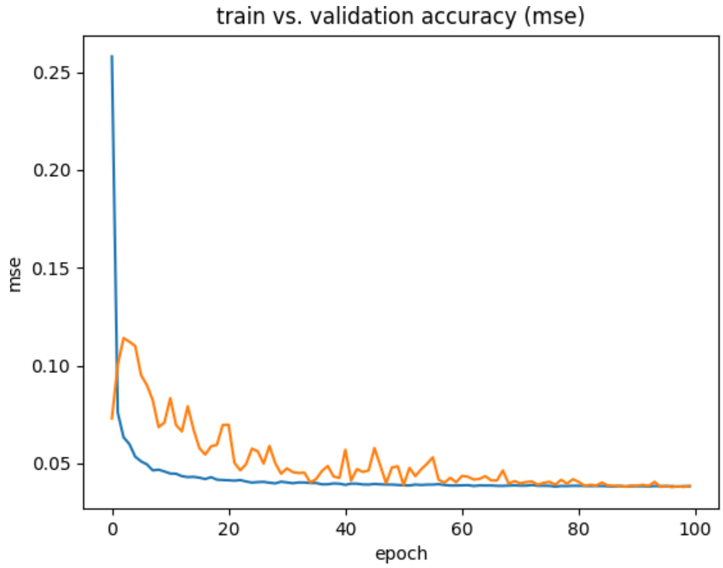


Figure 3. 22: LSTM with Attention Mechanism Train vs. Validation MSE for 100 epochs.

Figure 3.22 shows the MSE values for the attention model through the number of epochs for the 100 epochs run on ASELS data. The validation MSE value has fewer waves along the train MSE value than the run with 50 epochs.

Figure 3.23 shows the architecture of LSTM with Attention Mechanism with 150 epochs on ASELS data. It depicts the data dimensions and how layers are organized.

Figure 3.24 shows the MSE values for the attention model through the number of epochs for the 150 epochs run on ASELS data. As the number of epochs increases the loss value smoothly decreases and it has not any drastic change near the end of epochs.

Figure 3.25 shows the value distribution for the attention model for the 150 epochs runon ASELS data. It depicts the the distribution of real values and prediction values forthe test data.

Table 3.10 shows that the comparison results of ASELS stock with all the four models implemented in this work. According to the experimental results considering the MSEand MAE metrics, GRU model outperforms the other models.

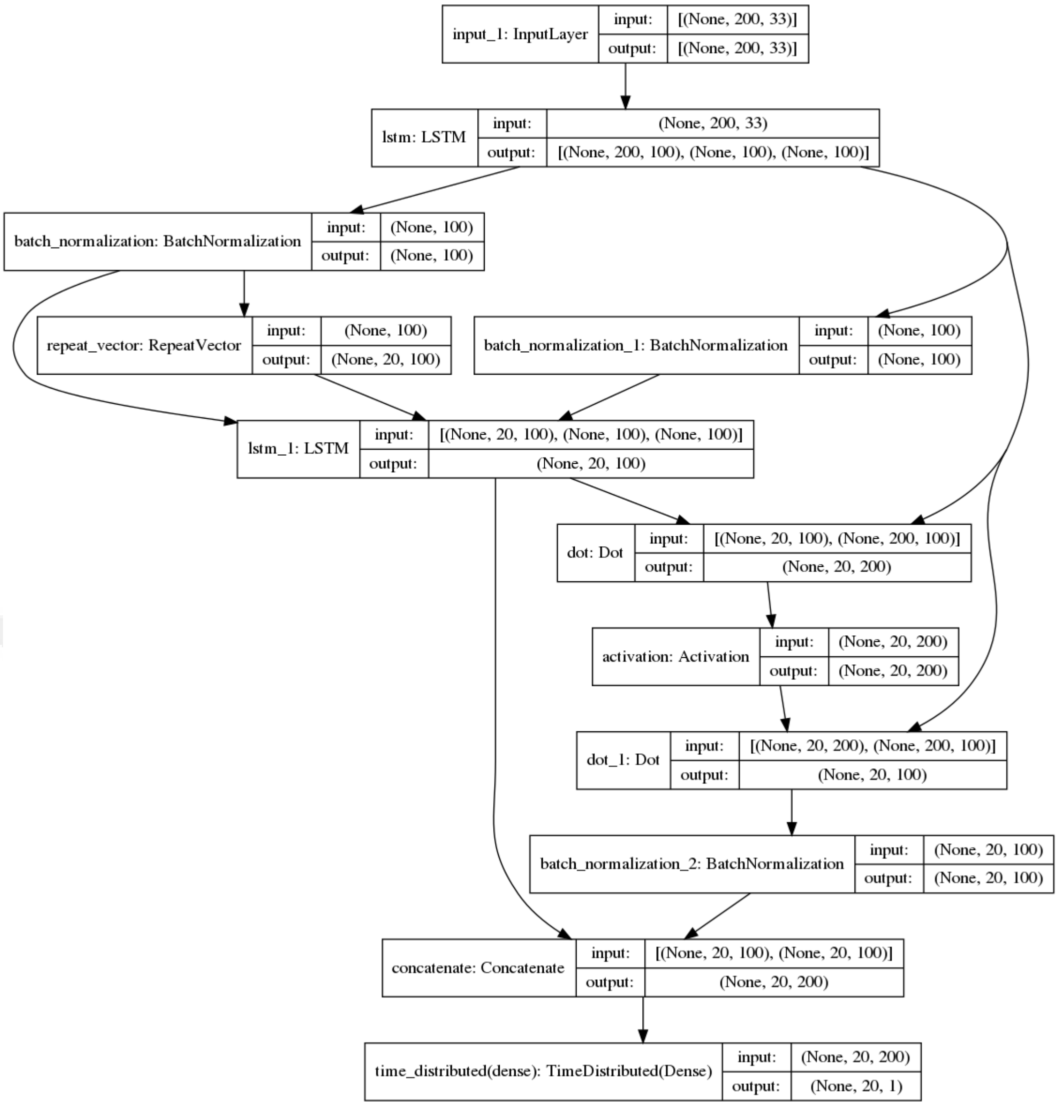


Figure 3. 23: LSTM with Luong Attention Mechanism Network Architecture.

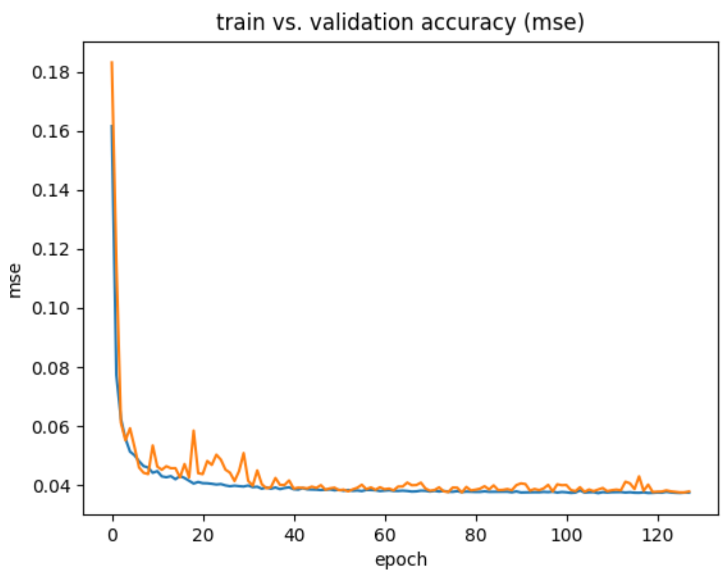


Figure 3. 24: LSTM with Attention Mechanism Train vs. Validation MSE for 150epochs.

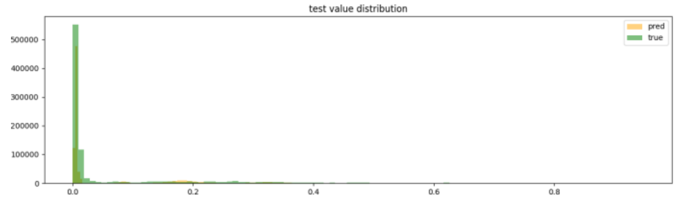


Figure 3. 25: LSTM with Attention Mechanism Test Value Distribution.

Table 3. 10: Comparison Results of ASELS for All Four Models.

Comparison Results		
Stock	***MSE***	***MAE***
ARIMA	0.0105	0.0836
LSTM	0.0009	0.0213
GRU	***0.0008***	***0.0205***
ATTENTION	0.0405	0.1670

3.1.4 Effects of Hyperparameters

When we increase the number of epochs, error values are decreasing respectively. In LSTM model, RELU gives better results over sigmoid function and hyperbolic tangent function gives the best results. In this thesis, we have compared three models by considering the error values, and the GRU model seems to give the best results overall.

When we take the results with different hyperparameters into consideration, we can conclude that the selection of activation function affected the results primarily. Table3.11 shows that LSTM model results with different activation functions. Hyperbolic tangent function outperforms the other activation functions. We use hyperbolic tangentin our other models based on these results.

Table 3. 11: Activation Function Comparison Results for LSTM.

Comparison Results			
Function	***Stock***	***MSE***	***MAE***
SIGMOID	AKBNK	0.0097	0.0701
RELU	AKBNK	0.0049	0.0512
TANH	**AKBNK**	**0.0042**	**0.0474**
SIGMOID	AKENR	0.0080	0.0625
RELU	AKENR	0.0042	0.0446
TANH	**AKENR**	**0.0036**	**0.0420**
SIGMOID	ASELS	0.0042	0.0444
RELU	ASELS	0.0016	0.0274
TANH	**ASELS**	**0.0015**	**0.0266**

Table 3.11 shows the comparison results of activation functions on LSTM model with 3 different data sources. For all different data stocks, hyperbolic tangent function gives the best results in terms of MSE and MAE.

In overall we have compared three different models of LSTM, LSTM with attention layer, GRU and ARIMA on BIST hourly data. We have predicted the hourly prices of three different stocks for comparison purposes. We have applied different combinations of number of epochs, activation functions, learning rates, batch sizes etc. GRU model over-performed the other models. Using GRU model with the best combination of hyper-parameters we have run our model on remaining 26 stocks.

3.1.5 Results for all Stocks

We have compared four different models; LSTM, GRU, LSTM with an attention mechanism, and ARIMA. In the experiment with 50 epochs, GRU seems to give betterresults than the other models. LSTM with an attention layer cannot outperform the other models due to the long length of the data sequence.

In this paper we include results of only 3 different stocks, but in our experimental results other stocks of BIST gives similar results to the ones in our paper.

After the pre-processing phases of our data there are 29 different stocks. We have tested and compared our models on 3 different stocks which are ASELS, AKBNK, AKENR. GRU model outperforms the other models according to our experimental runs. Then, we run our best model on the remaining 26 stocks with best performing parameters. Table 3.12 shows the comparison results for the remaining stocks.

Table 3. 12: GRU Results For All Stocks.

Comparison Results		
Stock	***MSE***	***MAE***
ALARK	0.0006713	0.0180291
BAGFS	0.0000139	0.0024756
BOLUC	***0.0000023***	***0.0007603***
CLEBI	0.0000367	0.0039341
DOAS	0.0000049	0.0014774
ECILC	***0.0000020***	***0.0007982***
EREGL	0.0000058	0.0015925
FROTO	0.0000591	0.0052464
GOLTS	0.0000341	0.0036391

Table 3.12 (Cont.)

GSDHO	***0.0000017***	***0.0004908***
ISCTR	0.0000046	0.0014149
KCHOL	0.0000117	0.0025253
ARCLK	0.0000241	0.0038115
AYGAZ	0.0000097	0.0022570
BFREN	0.0007204	0.0173188
CIMSA	0.0000087	0.0020132
DOHOL	0.0000032	0.0006939
ENKAI	0.0000042	0.0009044
FENER	0.0000152	0.0025375
GARAN	0.0000035	0.0013620
GOODY	0.0000037	0.0011822
IHLAS	***0.0000017***	***0.0004308***
ISGYO	0.0000033	0.0006017
KARTN	0.0020999	0.0297030
AKSA	0.0000102	0.0023948
AEFES	0.0000204	0.0033044

CHAPTER 4

CONCLUSION

In this study, we remark the increasing trend towards the stock market prediction, cryptocurrencies, forex, and high-frequency trading. Both traders and researchers are attracted to these topics, and deep learning algorithms are applied in many examples. We also have emphasized the importance of feature engineering and extensive use in deep learning applications. We used 28 technical indicators based on primary historical price data and USD/TRY ratio to enhance our data. We gathered the data from MATRIKS Inc., and we used the five fundamental features of data for our analysis: Open, High, Low, Close, Volume. Then, we created a large number of technical indicators using the initial data. After the pre-processing operations, we had 29 different stocks data. We showed our experiments on three different stocks from BIST: AKBNK, AKENR, ASELS. We fed our data to four models: LSTM, GRU, LSTM with attention layer, and ARIMA. In model creation, we made experimental runs with many different combinations of parameters in order to decide on our choice of the number of epochs, batch size, learning rate, activation function, loss function, time step, etc. We used 50, 100, and 150 for the number of epochs, 20 as the batch size, 20 and 30 as the time step, 0.0001 as the learning rate. We used hyperbolic tangent function as activation function and RMSE and MSE functions as the loss functions. We intended to make a technical analysis of BIST data and predict the next hour's open price for selected stock papers.

This paper is one of the first papers that applies LSTM with attention mechanism and hourly prediction of stock prices method to the BIST data. In our work, we have compared four models for three different stocks, and experimental results show that GRU outperforms the other models in forecasting the stock prices of BIST. ARIMA is used as a base model to compare the effects of statistical models with neural networks. Both LSTM and GRU performed better than the ARIMA model. LSTM with attention layer performance falls behind the other models. The reason might be the global attention model that we implemented. Due to its internal structure to produce the output, it may have difficulties in processing long sequences.

Applying a local attention model or more training epochs might improve the LSTM with attention layer model performance. Overall, GRU's performance is the best over both statistical and other neural network models. We then run the GRU model with the best performance-giving parameters on our remaining 26 stocks. We can conclude that the reason behind GRU's success over other models is the compact structure of GRU and its internals.

4.1 FUTURE WORK

This thesis only studied technical analysis with various methods such as LSTM, GRU, LSTM with attention mechanism, and ARIMA models on BIST data. Extensive usageof feature engineering is applied to our data, and even more technical indicators can be included in the data in order to enhance the model. Oil prices or consumer goods price changes can be added as an indicator. In addition to that, technical analysis and fundamental analysis can be converted into a single mechanism to utilize the capabilities of stock price prediction of predictive models. Crawling political and economic news from the internet and including them as an indicator into data features after inferring the meaning might boost the predictive performance. This improvement requires a decent understanding and skills of Natural Language Processing (NLP) applications.

Hybrid models can be implemented in future work. In this work, we used LSTM, GRU, LSTM with attention mechanism and ARIMA models distinctively. Hybrid models can be applied to time-series forecasting problems in various ways. For instance, the ARIMA model can be used in the feature engineering phases to feed our ARIMA-based features into an RNN model. Hybrid models can also be used to combine multiple predictions in the prediction phases to improve predictions' accuracy and make the model robust and resilient against overfitting problems.

Retrieving live BIST data and pipeline it through the pre-processing phases make it possible to have a livestock price prediction system. After pre-processing steps, the recently gathered data can be fed into the model to get the predictions. If there are any pre-defined rules to decide on buy/sell based on the predictions, the system may evolve into a live trading system. Another critical requirement is that there must be an authorization and authentication process with a bank or finance institution to accomplish buy and sell orders on BIST. The date range of data that we acquired in this work is between 2011 and 2020 years. If a live system would exist, the model could be improved with the live data. In this way, the model would be prone to learn the new and changing data patterns and the most recent price trends. The difficulties of stock price prediction due to its non-stationary nature can be decreased by feeding the model with the mostrecent data.

REFERENCES

[1] ADEBIYI, A. A., ADEWUMI, A. O., and AYO, C. K. (2014). Comparison of ARIMA and artificial neural networks models for stock price prediction. Journal of Applied Mathematics, 2014:1–7.

[2] AKŞEHİR, Z., KILIÇ, E., AKLEYLEK, S., DÖNGÜL, M., and COŞKUN, B. (2020). Stocks prices prediction with long short-term memory. In Proceedings of the 5th International Conference on Internet of Things, Big Data and Security. SCITEPRESS - Science and Technology Publications.

[3] ALMASARWEH, M. and WADI, S. A. (2018). ARIMA model in predicting banking stock market data. 12(11):309.

[4] ARIYO, A. A., ADEWUMI, A. O., and AYO, C. K. (2014). Stock price prediction using the ARIMA model. In 2014 UKSim-AMSS 16th International Conference on Computer Modelling and Simulation. IEEE.

[5] ARSLANKAYA, S. and TOPRAK, Ş. (2021). Using machine learning and deep learning algorithms for stock price prediction. Uluslararası Muhendislik Arastirma ve Gelistirme Dergisi, 13(1):178–192.

[6] BANERJEE, D. (2014). Forecasting of indian stock market using time-series ARIMA model. IEEE.

[7] BORONOVKOVA, S. and TSIAMAS, I. (2019). An ensemble of lstm neural networks for high-frequency stock market classification. Journal of Forecasting, 38(6):600–619.

[8] CHEN, Q., ZHANG, W., and LOU, Y. (2020). Forecasting stock prices using a

hybrid deep learning model integrating attention mechanism, multi-layer perceptron, and bidirectional long-short term memory neural network. IEEE Access, 8:117365– 117376.

[9] CHEN, S. and GE, L. (2019). Exploring the attention mechanism in lstm-based hong kong stock price movement prediction. Quantitative Finance, 19(9):1507–1515.

[10] CHO, K., MERRIENBOER, B. V., GULCEHRE, C., BAHDANAU, D., BOUGARES, F., SCHWENK, H., and BENGIO, Y. (2014). Learning phrase representations using rnn encoder–decoder for statistical machine translation. Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP).

[11] CHONG, E., HAN, C., and PARK, F. C. (2017). Deep learning networks for stock market analysis and prediction: Methodology, data representations, and case studies. Expert Systems with Applications, 83:187–205.

[12] DEMİRERL, U., ÇAM, H., and ÜNLÜ, R. (2020). Predicting stock prices using machine learning methods and deep learning algorithms: The sample of the istanbul stock exchange. GAZI UNIVERSITY JOURNAL OF SCIENCE.

[13] DEVI, B. U., SUNDAR, D., and ALLI, P. (2013). An effective time series analysis for stock trend prediction using ARIMA model for nifty midcap-50. 3(1):65–78.

[14] DU, Y. (2018). Application and analysis of forecasting stock price index based on combination of ARIMA model and BP neural network. IEEE.

[15] FAMA, E. F. (1965). The behavior of stock-market prices. The Journal of Business, 38(1):34.

[16] FAMA, E. F. (1970). Efficient capital markets: A review of theory and empirical work. The Journal of Finance, 25(2):383.

[17] FISCHER, T. and Krauss, C. (2018). Deep learning with long short-term memory networks for financial market predictions. European Journal of Operational Research, 270(2):654–669.

[18] GAO, W. and SU, C. (2020). Analysis on block chain financial transaction under artificial neural network of deep learning. Journal of Computational and Applied Mathematics, 380:112991.

[19] GREFF, K., SRIVASTAVA, R. K., KOUTNIK, J., STEUNEBRINK, B. R., and SCHMIDHUBER, J. (2017). Lstm: A search space odyssey. IEEE

Transactions on Neural Networks and Learning Systems, 28(10):2222–2232.

[20] GUNDUZ, H. (2021). An efficient stock market prediction model using hybrid feature reduction method based on variational autoencoders and recursive feature elimination. Financial Innovation, 7(1).

[21] GUNDUZ, H. and CATALTEPE, Z. (2015). Borsa istanbul (bist) daily prediction using financial news and balanced feature selection. Expert Systems with Applications, 42(22):9001–9011.

[22] GUNDUZ, H., CATALTEPE, Z., and Yaslan, Y. (2017a). Stock market direction prediction using deep neural networks. In 2017 25th Signal Processing and Communications Applications Conference (SIU), pages 1–4.

[23] GUNDUZ, H., YASLAN, Y., and CATALTEPE, Z. (2017b). Intraday prediction of borsa istanbul using convolutional neural networks and feature correlations. Knowledge- Based Systems, 137:138–148.

[24] HASAN, A., KALIPSIZ, O., and AKYOKUŞ, S. (2017). Predicting financial market in big data: Deep learning. In 2017 International Conference on Computer Science and Engineering (UBMK), pages 510–515.

[25] HU, Z., ZHAO, Y., and KHUSHI, M. (2021). A survey of forex and stock price prediction using deep learning. Applied System Innovation, 4(1):9.

[26] JARRETT, J. E. and KYPER, E. (2011). ARIMA modeling with intervention to forecast and analyze chinese stock prices. International Journal of Engineering Business Management, 3:17.

[27] JHANG, W.-S., GAO, S.-E., WANG, C.-M., and HSIEH, M.-C. (2019). Share price trend prediction using attention with LSTM structure. IEEE.

[28] KALYONCU, S., JAMIL, A., KARATAŞ, E., RASHEED, J., and DJEDDI, C. (2020). Stock market value prediction using deep learning. Data Science and Applications, 3(2):10–14.

[29] KARA, Y., Acar BOYACIOGLU, M., and Ömer Kaan BAYKAN (2011). Predicting direction of stock price index movement using artificial neural networks and support vector machines: The sample of the istanbul stock exchange. Expert Systems with Applications, 38(5):5311–5319.

[30] KILIMCI, Z. H. (2019). Borsa tahmini için derin topluluk modellleri (DTM) ile finansal duygu analizi. Gazi Üniversitesi Mühendislik-Mimarlık Fakültesi Dergisi.

[31] KUMAR, M. and THENMOZHI, M. (2014). Forecasting stock index returns

using ARIMA-SVM, ARIMA-ANN, and ARIMA-random forest hybrid models. 5(3):284.

[32] LI, Y., LI, L., ZHAO, X., MA, T., ZOU, Y., and CHEN, M. (2020). An attention-based LSTM model for stock price trend prediction using limit order books. 1575:012124.

[33] LIEN MINH, D., SADEGHI-NIARAKI, A., HUY, H. D., MIN, K., and MOON, H. (2018). Deep learning approach for short-term stock trends prediction based on two-stream gated recurrent unit network. IEEE Access, 6:55392–55404.

[34] LIN, B.-S., CHU, W.-T., and WANG, C.-M. (2018). Application of stock analysis using deep learning. In 2018 7th International Congress on Advanced Applied Informatics (IIAI-AAI), pages 612–617.

[35] LIU, H. (2018). Leveraging financial news for stock trend prediction with attention- based recurrent neural network.

[36] LUONG, T., PHAM, H., and MANNING, C. D. (2015). Effective approaches to attention- based neural machine translation. Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing.

[37] MA, Q. (2020). Comparison of ARIMA, ANN and LSTM for stock price prediction. E3S Web of Conferences, 218:01026.

[38] MEHTAB, S. and SEN, J. (2020). A time series analysis-based stock price prediction using machine learning and deep learning models. International Journal of Business Forecasting and Marketing Intelligence, 6(4):272.

[39] MONDAL, P., SHIT, L., and GOSWAMI, S. (2014). Study of effectiveness of time series modeling (arima) in forecasting stock prices. International Journal of Computer Science, Engineering and Applications, 4(2):13–29.

[40] NGUYEN, T.-T. and YOON, S. (2019). A novel approach to short-term stock price movement prediction using transfer learning. Applied Sciences, 9(22):4745.

[41] NTI, I. K., ADEKOYA, A. F., and WEYORI, B. A. (2019). A systematic review of fundamental and technical analysis of stock market predictions. Artificial Intelligence Review, 53(4):3007–3057.

[42] OBTHONG, M., TANTISANTIWONG, N., JEAMWATTHANACHAI, W., and WILLS, G. (2020). A survey on machine learning for stock price prediction: Algorithms and techniques. SCITEPRESS - Science and

Technology Publications.

[43] PAI, P.-F. and LIN, C.-S. (2005). A hybrid ARIMA and support vector machines model in stock price forecasting. Omega, 33(6):497–505.

[44] PATIL, P., WU, C.-S. M., POTIKA, K., and ORANG, M. (2020). Stock market prediction using ensemble of graph theory, machine learning and deep learning models. In Proceedings of the 3rd International Conference on Software Engineering and Information Management. ACM.

[45] QIU, J., WANG, B., and ZHOU, C. (2020). Forecasting stock prices with long-short term memory neural network based on attention mechanism. PLOS ONE, 15(1):1–15.

[46] RATHNAYAKA, R. M. K. T., SENEVIRATNA, D., JIANGUO, W., and ARUMAWADU, H. I. (2015). A hybrid statistical approach for stock market forecasting based on artificial neural network and ARIMA time series models. IEEE.

[47] REZAEI, H., FAALJOU, H., and MANSOURFAR, G. (2021). Stock price prediction using deep learning and frequency decomposition. Expert Systems with Applications, 169:114332.

[48] SAMI, M. and ABDALLAH, W. (2020a). Applied finance letters cryptocurrency and stock markets : Complements or substitutes? evidence from gulf countries. Applied Finance Letters, 9.

[49] SAMI, M. and ABDALLAH, W. (2020b). How does the cryptocurrency market affect the stock market performance in the mena region? Journal of Economic and Administrative Sciences, ahead-of-print.

[50] SAUD, A. S. and SHAKYA, S. (2020). Analysis of look back period for stock price prediction with RNN variants: A case study on banking sector of NEPSE. Procedia Computer Science, 167:788–798.

[51] SELÇUK, A., YİĞİT, Ö. E., and ERSOY, Ö. (2020). Prediction of bist price indices: a comparative study between traditional and deep learning methods. Sigma Journal of Engineering and Natural Sciences, 38(4):1693–1704.

[52] SHEN, J. and SHAFIG, M. O. (2020). Short-term stock market price trend prediction using a comprehensive deep learning system. Journal of Big Data, 7(1).

[53] TEKİN, S. and ÇANAKOĞLU, E. (2019). Analysis of price models in istanbul stock exchange. In 2019 27th Signal Processing and Communications

Applications Conference (SIU), pages 1–4.

[54] VIJH, M., CHANDOLA, D., TIKKIWAL, V. A., and KUMAR, A. (2020). Stock closing price prediction using machine learning techniques. 167:599–606.

[55] WANG, J.-H. and LEU, J.-Y. (1996). Stock market trend prediction using arima-based neural networks. In Proceedings of International Conference on Neural Networks (ICNN'96), volume 4, pages 2160–2165 vol.4.

[56] WEI, D. (2019). Prediction of stock price based on LSTM neural network. IEEE.

[57] WEISS, G., GOLDBERG, Y., and YAHAV, E. (2018). On the practical computational power of finite precision RNNs for language recognition. In Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers). Association for Computational Linguistics.

[58] WEN, M., LI, P., ZHANG, L., and CHEN, Y. (2019). Stock market trend prediction using high-order information of time series. 7:28299–28308.

[59] YANG, Y. (2005). Can the strengths of aic and bic be shared? a conflict between model indentification and regression estimation. Biometrika, 92(4):937–950.

[60] YU, P. and YAN, X. (2019). Stock price prediction based on deep neural networks. 32(6):1609–1628.

[61] ZENG, Z. and KHUSHI, M. (2020). Wavelet denoising and attention-based RNN ARIMA model to predict forex price. In 2020 International Joint Conference on Neural Networks (IJCNN). IEEE.

[62] ZHANG, X., HU, Y., XIE, K., WANG, S., NGAI, E., and LIU, M. (2014). A causal feature selection algorithm for stock prediction modeling. Neurocomputing, 142:48–59.

[63] ZHANG, X., LIANG, X., ZHIYULI, A., ZHANG, S., XU, R., and WU, B. (2019). AT-LSTM: An attention-based LSTM model for financial time series prediction. 569:0520