Stock Market Forecasting Using Machine Learning Models

Atakan Site
Computer Engineering Department
Dokuz Eylül University
İzmir, Turkey
atakan.site@ceng.deu.edu.tr

Derya Birant
Computer Engineering Department
Dokuz Eylül University
İzmir, Turkey
derya@cs.deu.edu.tr

Zerrin Işık
Computer Engineering Department
Dokuz Eylül University
İzmir, Turkey
zerrin@cs.deu.edu.tr

***Abstract*—Stock market forecasting is a challenging problem. In order to cope with this problem, various techniques and methods have been proposed. In this study, the stock close values are tried to be forecasted as monthly and weekly. For this purpose, values of two traded stocks (Google, Amazon) are predicted using with models being processed. In addition, stock data were taken from two different indexes (Dow Jones Industrial Average (DJIA) and S&P 500) for a realistic assumption. Stock market data includes long term dependencies. For this reason, classical Recurrent Neural Networks (RNN)-based models could not effectively work for in such data. Therefore, Long Short-Term Memory (LSTM) Networks and Gated Recurrent Unit (GRU) based models were developed and their efficiencies were observed in this study. These models were also compared with traditional machine learning approaches and the obtained gains were calculated.**

With LSTM, the close prices of datasets were predicted more consistently than the other models. In particular, the LSTM model is more successful than the GRU model with similar error metrics in dealing with fluctuations in datasets. Besides, linear machine learning models are well behind the deep learning-based models in weekly prediction of datasets.

Keywords—forecasting; Recurrent Neural Networks; Long Short-Term Memory; Gated Recurrent Unit; index; stock market.

I. INTRODUCTION

Stock market forecasting is an important and argumentative issue in the field of finance. There are many different hypotheses in this area. Some researchers emphasize that stock market values best reflect all available information about market. With this assumption, the stock markets are considered to be unpredictable. Others tried to predict the market through identification of economic indicators (leading indicators, coincident indicators, lagging indicators), technical analysis, and machine learning. There are several machine learning techniques for stock market forecasting. These techniques mostly use numerical or structured economic indicators with technical analysis results from stock data. These numerical data usually use past stock price parameters to predict future stock prices (e.g., daily, weekly, yearly). Forecasts for the future play a key role for investors to make an accurate decision about their investments and to forecast stock performance. These effects of investors and other similar factors make the problem of stock market forecasting more difficult and complicated. Machine learning based approaches have been tried to find solutions to these problems. As a result of the logarithmic increase in the processing power of computers, some previously proposed approaches have been implemented more effectively. These improvements have given speed and meaning to deep learning-based studies. Adding depth to any learning model also ensures great flexibility. Various hybrid learning models have been proposed for the solution of multi-parameter problems with this philosophy. There are many different techniques and studies in the literature. Fabbri and Morro have benefited from RNN networks that act as a memory unit in stock prediction [1]. Ensemble learning based solutions, that combine different weak machine learning algorithms, are quite common in stock market forecasting. Patel *et al*. forecasted stock market movement direction of Indian stock markets [2]. They used and compared classification accuracy of Artificial Neural Networks (ANN), Supported Vector Machines (SVM), Random Forest (RF), and Naive Bayes (NB) algorithms. The best results were obtained with the RF algorithm which is an ensemble learning based model. Some researchers have developed hybrid machine learning models. Hybrid models can work with data obtained from different sources constitute the most important studies in this topic. Hassan *et al*. proposed a fusion model of Hidden Markov Model (HMM), ANN, and Genetic Algorithm (GA) [3]. Proposed model includes two main parts. First part of model performs optimization of initial parameters for HMM by iterative execution of the ANN and GA models. Second part of model calculates weighted average of price differences for the stock price sequence. The value obtained with weighted average is one day stock forecast. In addition to these studies, the stock market forecasting includes many different techniques and different approaches. The method to be applied according to the desired problem and the steps of these methods change.

The most common problems to be solved in this area can be generalized as the percentage change of the stocks

978-1-7281-2868-9/19/$31.00 ©2019 IEEE

within a certain period of time, as the values of close prices and their volatility. A sequence-based problem, such as stock market forecasting, is a good example of time series. Component analysis of time series is an important issue for investors. In particular, seasonality and trend analysis are important issues for stock market forecasting.

The contribution of this study is the development deep learning models that can achieve good performance by using DJIA and S&P 500 index series without any external data. The proposed models are very efficient and optimized with the suitable hyperparameters for the datasets. Datasets were selected to correspond to different problems. With the classic RNN-based model, LSTM and GRU models, which developed specifically for sequence-based problems, are modeled to have the same network depth. Therefore, performance fluctuations between the proposed RNN, LSTM, and GRU can be observed more easily.

II. BACKGROUND

As mentioned in previous section, many researches have been done on stock market prediction. Especially hybrid and multi-task approaches are the most suitable solution developed for these kinds of problems. However, hybrid approaches generally consume a lot of resources and usually do not use only stock market indicators. Wang *et al*. combined stock data and its technical indicators, so that to increase the prediction performance by making sentiment analysis from exchange news [4]. It is very difficult to process the data obtained from different resources, so it became a necessity to work with models working on financial data even if optimistic results are obtained. Therefore, the use of deep learning-based models is inevitable when considering the increasing amount of time series data. In order to properly process this large amount of data, some studies have been proposed that include different feature selection methods. Gündüz *et al*. has evaluated the prediction performance of their models by applying feature selection methods such as gain ratio, relief to the extend dataset with technical indicators [5]. The preprocessing of the data, the extraction of the different features and parameters increase the generalization performance of the learning models. Some of the important works in this field are the systems that act as human analysts to predict stock movement. Weng *et al.* have designed an expert system with and graphical user interface that interacts with the state of art machine learning models, different resources and feature selection methods to achieve best prediction results [6]. As can be seen, the stock market forecasting is one of today's most popular topics. Stock market forecasting is a dynamic, noisy and difficult problem. As seen in the literature, many approaches and models have been developed to solve this problem. Unfortunately, there are numerous failing stock prediction works. The models we proposed in this study use only stock indexes. We will try to succeed on the stock market prediction without additional data from different sources and detailed technical analysis methods. For this purpose, the datasets have been suitable for the operation of the models. The sliding window technique was applied to improve the forecast performance and to determine which time interval will be given to the model. In statistics this type of approach called a forecast horizon. There is no optimal forecast horizon, the size of forecast horizon depends on the statistical criteria of the datasets.

III. METHODOLOGY

A. Data Sources

To make a comparison with the models, we used two different datasets extracted from Yahoo Finance [7]. In order to evaluate the reliability of our results, we used datasets with different time intervals and different number of observations. Besides we used the S&P 500 for AMZN dataset also, the GOOGL for DJIA as a target market. Datasets do not include irregular seasonal effects or causal variables like holidays. Our GOOGL dataset includes open, high, low, close and adjusted close values between (2006-01-03) to (2017-12-29).

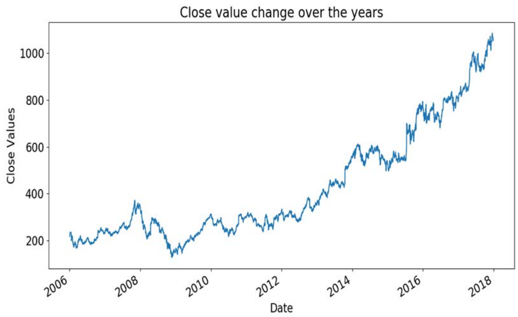

Fig. 1. Close price of GOOGL stock based on DJIA target market between (2006-01-03) and (2017-12-29)

As can be seen in Figure 1, GOOGL data show a positive trend. In order to evaluate 3019 observations between specified range, we divided the GOOGL dataset by 80%-20% (train/test ratio).

The AMZN dataset, contains a smaller time range and includes less observations than GOOGL one. Especially short-term forecasting cannot be performed effectively by some of deep learning-based models. With this dataset, we try to test how the proposed models will have a prediction in the case of such data. In addition, both data sets contain different close price frequencies. We have also included some of the traditional machine learning methods used in the forecasting of the linear and univariate time series in our study. These models will be taken baseline models. The performance difference between the RNN, LSTM, GRU and presented baseline models was observed on datasets with similar trend and components but with wider time interval and number of observations.

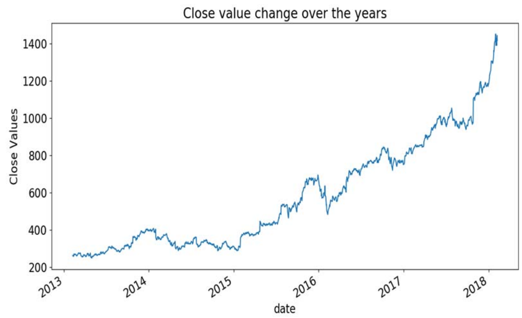

Fig. 2. Close price of AMZN stock based on S&P 500 target market between (2013-02-08) and (2018-02-07)

In Figure 2, AMZN data show a positive trend. In order to evaluate 1259 observations between specified range, we divided the AMZN dataset by 80%-20% (train/test ratio).

Generally, when long-term dependencies on time series increase, traditional machine learning methods and autoregressive approaches such as ARIMA, SARIMA, ARMA etc. are not sufficient [8], [9]. In this study, the performance of the proposed models on such a problem is evaluated by using machine learning models.

B. Data Preprocessing

Because of exchange data is not traded at closing days (weekends, holidays) of exchange, we had to implement the sliding windowing method, which made it possible the prediction for our in the taken range (weekly and monthly). We examined the autocorrelation in the data to understand characteristics of data which shows the degree of similarity between the values of the same variables over successive time intervals. It is also a measure of the linearity of datasets that have similar trend and other characteristics with each other except time range and size. Equation 1 shows the autocorrelation calculation in which T indicates time interval. Autocorrelation function defined as follows,

$$r_k = \frac{\sum_{t=k+1}^{T}(y_t - \acute{y})(y_{t-k} - \acute{y})}{\sum_{t=1}^{T}(y_t - y)^2} \quad (1)$$

where $y(t)$ is represents sorted dataset by ascending order with time t, $y(t-k)$ represents same dataset by lagged by k units and variable $\acute{y}$ represents mean of original dataset.

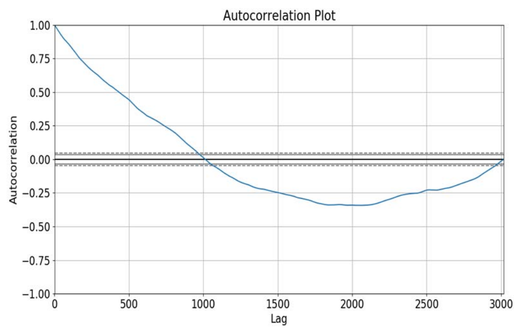

Fig. 3. Autocorrelation plot for GOOGL dataset.

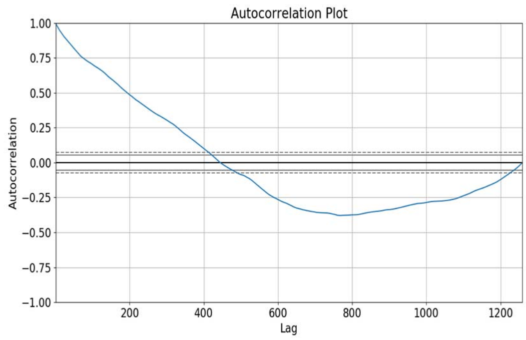

Fig. 4. Autocorrelation plot for AMZN dataset.

In Figure 3 and Figure 4 show the relationship between the previous and next terms of our time series. There are linearities between variables present in all datasets. A z-score normalization (Equation 2) was applied to all variables in dataset to achieve consistent scale for better forecasting results. A z-score normalization defined as follows,

$$z = \frac{(x-\mu)}{\sigma} \quad (2)$$

where μ is the mean of the dataset and σ is standard deviation.

After the normalization process, the models to be applied on the datasets receive different types of input. As we have emphasized before, we used linear and non-linear machine learning models in our study as a competitor of the deep learning models. For this reason, reshaped our time series in the format accepted by the models. Linear and non-linear machine learning models take input data as matrix, since the LSTM, GRU and RNN models accept three-dimensional data, our data are given to these models in the tensor form. Three-dimensions input has the components of; samples from dataset, forecast horizon and used features from dataset.

C. Model Overview

In general, our problem is a regression-based problem. We are trying to forecast the close values of exchange at weekly or monthly intervals. In terms of stock market prediction model, which is trained to take all stock indexes to make a prediction, we used Linear Regression (LR), Ridge Regression (RR), Support Vector Regression (SVR), RNN, LSTM, and GRU.

RNNs have performed much better in data sets with sequential dependencies than feed forward neural networks (FNN). This is achieved by providing a feedback which feed the network activations from a previous step as inputs to the network in order to influence predictions at the current time step [1]. The RNNs allow dynamical usage of the defined time series as it will update the weights across each time window when it runs together with sequential data. However, FNNs only receive input patterns and generate associations between output patterns. There are two common RNNs used in the problems involving time series data. These are Long Short-Term Memory (LSTM) [10] and Gated Recurrent Unit (GRU) [11]. Both models are based on RNNs, but they improve some of the structural constraints such as vanishing gradient problem caused by multiplication of weight matrix and the derivation of used activation function. The vanishing gradient problem occurs when there is an interrupted gradient flow in the regular RNNs if the time window size is too large. In RNNs, the backpropagation of the error occurs in a special way. During the backpropagation, the time instant is also given as an input to the network as shown in Figure 5. This approach is called backpropagation through time (BPTT).

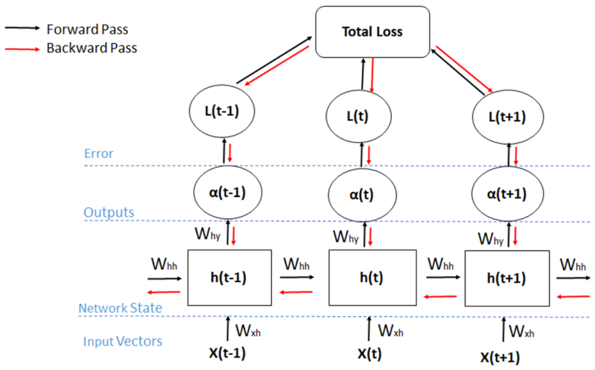

Fig. 5. Backpropagation through time

LSTMs use the identity function as an activation function in the recurrent layers in order not to be affected by the vanishing gradient problem that may occur as a result of BPTT. This feature ensures the uninterrupted gradient flow in the network.

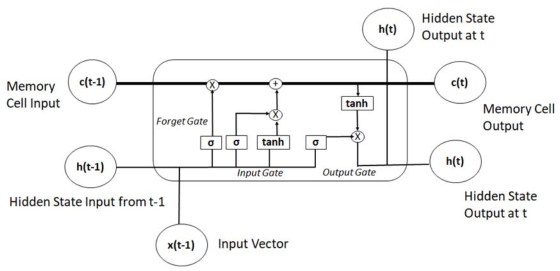

Fig. 6. Architecture of LSTM cell

LSTMs have several gates that allow the network to learn as shown Figure 6.

Forget gate allows ability to remove or add information from memory cell input to the memory cell output.

Input gate allows add and update new values to the current cell state of network.

Output gate controls output flow from the cell state.

Besides, GRUs are a specialized type of the LSTM network. The main difference between the two network structures is that the GRUs do not contain any memory and they include only two gates. GRUs combine forget and input gate to a single update gate. This main difference allows GRUs to consume less resources and work much more efficient than LSTMs. According to LSTM, deeper GRU networks can be created with similar resource consumption and running time complexity.

D. *Architecture of Proposed Models*

The proposed all deep learning models include two stacked recurrent layers. The main purpose of this type of structure is to improve the generalization capability of the networks. The hyperparameters to be taken by the designed models are tuned for all datasets with Tree of Parzen Estimators and Grid Search. Mean Squared Error (MSE), Mean Average Error (MAE), and Mean Absolute Percent Error (MAPE) metrics were used to evaluate the forecasting accuracy of models. Table I summarizes hyperparameters of each model.

TABLE I. HYPERPARAMETERS OF DEEP LEARNING MODELS

Model	Hyperparameters
LSTM	Hidden Layer (1): 32 Neurons, activation function: linear Dropout Layer (0.2 ratio) Hidden Layer (2): 16 Neurons, activation function: linear Dropout Layer (0.2 ratio) Dense Layer: 1 Neuron, activation function: linear Adam optimizer for Stochastic Gradient Descent (SGD) with 0.002 learning rate
GRU	Hidden Layer (1): 32 Neurons, activation function: tanh Dropout Layer (0.2 ratio) Hidden Layer (2): 16 Neurons, activation function: tanh Dropout Layer (0.2 ratio) Dense Layer: 1 Neuron, activation function: linear Stochastic Gradient Descent (SGD) with momentum = 0.9
RNN	Hidden Layer (1): 32 Neurons, activation function: tanh Dropout Layer (0.2 ratio) Hidden Layer (2): 16 Neurons, activation function: tanh Dropout Layer (0.2 ratio) Dense Layer: 1 Neuron, activation function: linear RMSProp optimizer for Stochastic Gradient Descent (SGD) with 0.002 learning rate

Deep learning models were evaluated with their own parameters. The number of neurons in the hidden layers of each model remain the same. Even if each model has similar network depth, the learning rate for an RNN-based model was kept as low as possible because of vanishing gradients problem. In this way, as the number of data and forecast horizon grew, the performance drop of RNN was prevented further than other models. The performance of three different approaches with similar structures were compared. Each model is more successful with a different gradient descent optimizer. In addition to the models, the deeper network structures were tested in the setting of these hyperparameters, but the best results for each model were achieved with the listed structures in Table I. It is more appropriate to use different hyperparameters and much larger datasets with various indicators for deeper network structures.

Some linear and non-linear machine learning algorithms were used and their hyperparameters tuned with Grid Search algorithm. These models applied on the datasets were used to observe how linear and non-linear models behave on data of different size and time intervals. Also, all of our datasets contain positive trend and linearity. So, as the data size of linear machine learning algorithms (LR, RR) increases, there is a chance to observe their performance changes.

TABLE II. HYPERPARAMETERS OF TRADITIONAL LEARNING MODELS

Model	Hyperparameters
SVR	'C': 1.5, 'epsilon': 0.1,' gamma': 1e-07, 'kernel': polynomial, 'degree': 3
Linear Regression	With default parameters. (used for baseline model)
Ridge Regression	'alpha': 1.0, 'copy_X: True, 'fit_intercept': True, 'max_iter': None, 'solver':auto, 'tol': 0.001

The selection of hyperparameters was performed according to MSE metric with 10-fold cross validation produced by testing all combinations in the hyperparameter space with the Grid Search algorithm. The optimum setting of hyperparameters is given in Table II.

IV. EXPERIMENTAL RESULTS AND DISCUSSION

Models have been evaluated on two datasets. Table III and Table IV show the prediction performances of all models in terms of MSE, MAE, and MAPE for weekly and monthly data, respectively. Deep learning models are trained with 300 epochs on GOOGL and AMZN datasets. Since these datasets contain less observations and narrower time intervals for high number of epochs, overfitting problem has occurred in high number of epochs. Linear models have shown the best prediction performance with low number of observations and narrower time intervals. The increase in the number of observations and the wider of time intervals led better predictions with the deep learning-based models. Successful prediction was performed with proper observations and appropriate time intervals. Since the stock market data has very dynamic behavior, the time period covered should not be very wide. Since the demands of the market and investors are dynamically changing. For this reason, data covering a very wide time interval is not sufficient for a good forecasting.

Additionally, the stock prediction problem was evaluated for weekly and monthly data. In deep learning models, the difference between the two approaches do not change if the number of observations is not high. SVR, which is a non-linear model, led the poorest predictions in general. This result could be due to the linear relationship in datasets. We believe that an SVR model is more suitable for one step ahead forecasting for stationary datasets. LSTM and GRU were the most successful ones out of all deep learning-based models. One of the important factors in the formation of this result is less effect of vanishing gradients problem in these models. The GRU models work much faster than LSTM models and have achieved very similar results throughout the runs. As shown in Figure 7 and Figure 8, GRU model does not show sufficient performance even if it has close error values according to the LSTM model, especially where the fluctuations are high in dataset. Besides, as can be seen in Figure 9 and Figure 10, wider time intervals negatively affected the RNN model with interrupted gradient flow. LSTM model contains a more sophisticated memory than GRU models, so it has the ability to predict more temporal related data. GRUs will perform better than LSTMs when data is limited or the risk of overfitting is high (i.e., high noise levels).

TABLE III. WEEKLY PREDICTION RESULTS OF ALL MODELS

Dataset	GOOGL	AMZN
LSTM	MSE:0.027 MAE:0.013 MAPE:0.018	MSE:0.025 MAE:0.013 MAPE:0.019
GRU	MSE:0.026 MAE:0.014 MAPE:0.17	MSE:0.026 MAE:0.0145 MAPE:0.017
RNN	MSE:0.023 MAE:0.012 MAPE:0.015	MSE:0.028 MAE:0.016 MAPE:0.020
SVR	MSE:0.16 MAE:0.28 MAPE:0.81	MSE:0.036 MAE:0.17 MAPE:0.24
LR	MSE:0.0033 MAE:0.0087 MAPE:0.0153	MSE:0.0016 MAE:0.0085 MAPE:0.013
RR	MSE:0.037 MAE:0.0094 MAPE:0.017	MSE:0.0037 MAE:0.014 MAPE:0.021

TABLE IV. MONTHLY PREDICITON RESULTS OF ALL MODELS

Dataset	GOOGL	AMZN
LSTM	MSE:0.0247 MAE:0.125 MAPE:0.169	MSE:0.023 MAE:0.011 MAPE:0.018
GRU	MSE:0.025 MAE:0.127 MAPE:0.17	MSE:0.024 MAE:0.016 MAPE:0.019
RNN	MSE:0.028 MAE:0.17 MAPE:0.184	MSE:0.024 MAE:0.014 MAPE:0.0194
SVR	MSE:0.12 MAE:0.24 MAPE:0.72	MSE:0.033 MAE:0.16 MAPE:0.238
LR	MSE:0.00132 MAE:0.0078 MAPE:0.0128	MSE:0.0015 MAE:0.0082 MAPE:0.012
RR	MSE:0.0022 MAE:0.016 MAPE:0.0145	MSE:0.0044 MAE:0.014 MAPE:0.022

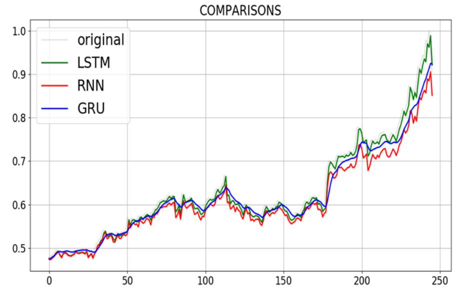

Fig. 7. Monthly Results (AMZN dataset)

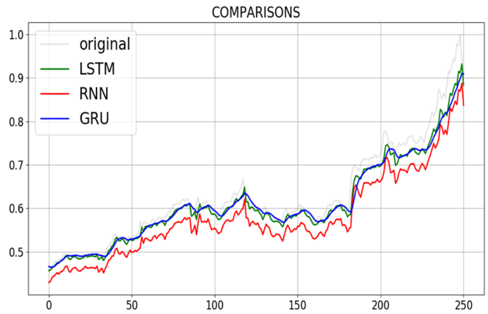

Fig. 8. Weekly Results (AMZN dataset)

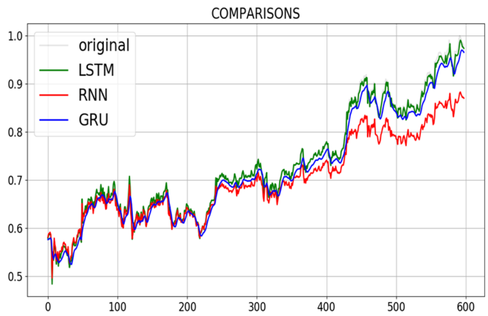

Fig. 9. Monthly Results (GOOGL dataset)

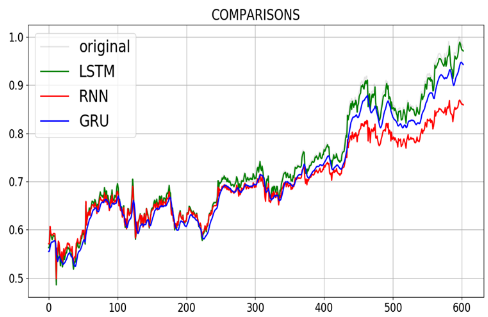

Fig. 10. Weekly Results (GOOGL dataset)

V. CONCLUSIONS AND FUTURE WORK

This study presents an evaluation of prediction performance of different machine learning models for two different stock exchange datasets. The results revealed that a satisfactory prediction performance can be achieved with deeper RNN models. The comparison between RNN variants showed; very variable results. LSTM networks provided better predictions compared to other models with similar network depth. Weekly forecasting problem is quite inadequate for linear machine learning models due to effects on holidays. Different fluctuations are observed in the close prices after holidays and similar factors. Because of these effects, monthly forecasting has resulted in much better performance with linear machine learning models. However, we believe that the performance of linear machine learning models will be further reduced in different datasets that contain too much noise. LSTM and GRU models were less affected by the change in forecast horizon and the fluctuations in the dataset.

In future work, LSTM and GRU models can be tested with non-fixed structure and datasets with higher observations, different trend and some seasonality effects. In addition, various technical indicators may be used to predict the closing values of stocks. Besides, convolutional neural networks can be applied to data in a variety of time series, can handle the time series in a different way and perform forecasting.

REFERENCES

[1] Mirco Fabbri, Gianluca Moro, "Dow jones trading with deep learning: the unreasonable effectiveness of recurrent neural networks," *7th International Conference on Data Science, Technology and Applications*, Jul 26, 2018, Porto/Portugal, 2018.

[2] Jigar Patel, Sahil Shah, Priyank Thakkar, Ketan Kotecha, "Predicting stock and stock movement using trend deterministic data preparation and machine learning techniques," *Expert Systems with Applications, 259-268, 2015.*

[3] Md. Rafiul Hassan, Baiknuth Nath, Michael Kirley, "A fusion model of HMM, ANN and GA for stock market forecasting," *Expert Systems with Applications*, 171-180, 2007.

[4] Chris Wang, Yilun Xu, Qingyang Wang "Novel approaches to sentiment analysis for stock prediction," *Stanford University*, USA, 2018.

[5] Hakan Gündüz, Zehra Çataltepe, Yusuf Yaslan, "Stock daily return prediction using expanded features and feature selection," *Turkish Journal of Electrical Engineering & Computer Sciences, 4829-4840, 2017.*

[6] Bing Weng, Mohamed A. Ahmed, Fadel M. Megahed, "Stock market one-day ahead movement prediction using disparate data sources," *Expert Systems with Applications*, 2017.

[7] Yahoo Finance. URL:https://www.finance.yahoo.com/

[8] Sima Siami Namini, Neda Tavakoli, Akbar S. Namin, "A comparision of ARIMA and LSTM in forecasting time series," *17th Conference on Machine Learning and Applications (ICMLA),* 2018.

[9] Oussama Fathi, "Time series forecasting using a hybrid ARIMA and LSTM model,"Velvet Consulting, France, 2019.

[10] Sepp Hochreiter, Jürger Schmidhuber, "Long short-term memory," *Neural Computation*, 1735-1780, 1997.

[11] Kyunghyun Cho et al, "Learning phrase representations using RNN encoder-decoder for statistical machine translation," *Conference on Empirical Methods in Natural Language Processing*, October 25-29, 2014, Doha/Qatar, 2014.