

LSTM FORECASTING ON THE MACEDONIAN STOCK EXCHANGE (MSE)¹

Jovanoski Davor, MSc

University of Applied Sciences Burgenland (UAS), Austria

djovanoski@gmail.com

Abstract

Stock market forecasting defines a guiding role for investors decision making, therefore it is an important topic of research in academia and finances. With the advancement of artificial intelligence and deep learning, its inclusion into the financial world became imperative. Machine learning algorithms focus on training models over large amounts of available stocks data and making an accurate stock forecast is challenging as the market is volatile. However, it is necessary for reducing risk in market decision-making.

This research attempts to explore the potential of a forecasting model using Long Short-Term Memory (LSTM) in order to predict stock prices for several companies traded on the MSE. This multivariate time series forecasting, focuses on the maximum, minimum, starting and last stock price. We train the model using the monthly reports provided by the MSE for the past 15 years and in order to understand the long term market, we forecast the next several months. The conducted experiments focus on observing the error in order to assess the consistency between the predicted and the actual stock prices, showing the viability of this model for forecasting on the MSE.

Keywords: MSE, LSTM, Forecasting, Machine Learning.

¹ original research paper

INTRODUCTION

Forecasting stock exchange prices has always been a challenge. Having a referent information flow for the future, promises a reduction in risk for any future decision making regarding stock prices. Forecasting stock exchange prices by considering its dynamic factors is an important part of a business investment plan.

With the focus of this paper being the MSE, an exchange with limited research, (Okičić, 2015) investigates Central and Eastern Europe transitional and underdeveloped stock markets and explains the behavior of financial time series whilst focusing on returns and conditional volatility. Related to the investigation from (Kovačić, 2007), where MSE stock return time series displayed stylized facts as volatility, clustering, low starting and slow decaying autocorrelation function of squared returns.

Applications of techniques for stock prices forecasting is a often researched subject; (Shah et al., 2019) they categorized them in statistical, pattern recognition, machine learning sentiment analysis and hybrid. With this research we focus on the machine learning aspect, which in (Soni et al., 2022) furthermore is categorized as traditional machine learning, deep learning and neural networks, time series analysis, graph based approach.

Focusing on the premise of the MSE with its volume of trades and its volatility, we investigate how a deep learning model acts when presented with the limited data from the monthly reports for several companies. A variation of the LSTM model previously trained over daily stock prices (Jain et al., 2018), is now presented with less data for training expecting viable forecasting of stock prices.

RELATED WORK

Monthly forecasting is not an unknown research topic (Agung, 2019), which research is well established in sales, climate data and living expenses forecasts. Having the focus on the stock market with its volatility usually follows investigations over the future trend of the stock (Gong & Sun, 2009).

Machine learning techniques for stock price predictions categorize as reinforced, supervised and unsupervised learnings are summarized in (Shah et al., 2019) and deep learning methods in (Soni et al., 2022), one of which are the LSTM networks. The authors of (Lv et al., 2022) introduced their hybrid model based on LSTM and show its improvements over other LSTM models.

LSTM

LSTM introduced in (Hochreiter & Schmidhuber, 1997) used in the field of deep learning represents a variety of recurrent neural networks (RNNs) which can learn long-term dependencies, usually focused on sequence prediction problems. Analogous to RNN which has both long-term and short-term memory, LSTM has feedback connections; thus, it can process the entire sequence of data, apart from single data points. LSTM driver is a memory cell with its known state. Information can be added to or removed from the cell state in LSTM and is regulated by gates. These gates optionally let the information flow in and out of the cell, known as input gate, forget gate and output gate. The model in this research is a variation of the LSTM architecture mentioned in (Jain et al., 2018).

METHODOLOGY

Designing a model for predicting fluctuating stock prices is an interesting and challenging problem. With the increase of data availability, we can extract information from multiple perspectives regarding the stock price, thus offering different approaches to predict it. Although similar machine learning models focus on short-term investment decisions (Soni et al., 2022), we present an alternative approach to understanding the long-term market. With this model we forecast up to several months in the future with the expectation of progressive loss of accuracy and increased error. Focusing only over the monthly values implies less input data, therefore we require an increase on the time-period considered before the forecasting. With those preconditions we focus our forecasting over companies which are offered on the MSE for more than 15 years. We consider their stock prices as time series analysis over the monthly economic variable changes. The technical implementation of the model is based on Keras v2.9.0 (Chollet, 2015) and Tensorflow v2.9.1 (TensorFlow Developers, 2022) using python as programming language.

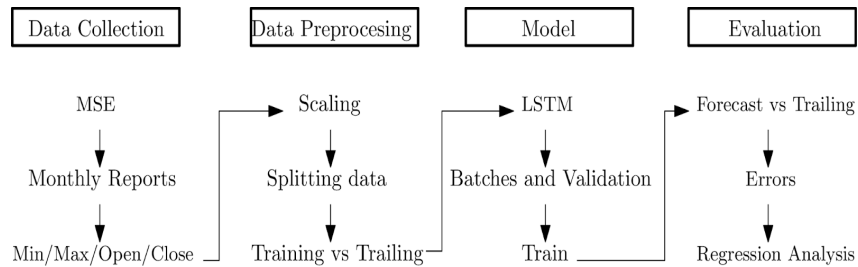


Figure 1. Method flow-chart

As depicted in (Figure 1) initially we fetch the monthly reports from MSE considering the maximum, minimum, opening and closing price as the input data for the financial multivariate time-series. With an initial observation over the descriptive statistics including Augmented Dickey–Fuller (ADF) test we check the stationarity of the time-series.

After the extraction of data from the monthly reports, we linearly transform the “Max”, “Min”, “Open” and “Close” values using a standard scaler. To evaluate our model, we separate the data between training and trailing data, with the intent for the trailing data to represent the comparison we forecast using our model. Therefore, when we want to forecast n -months we evaluate and measure our model over those trailing n -months, thus allowing for calculations and comparisons against the estimations. Following the idea of mini-batches as frames in a movie (Khodabakhsh et al., 2020), we reshape the training data considering every three months of data to predict the forth.

The LSTM is based on the (Jain et al., 2018) architecture visualized with (Figure 2) using the same activation function - rectified linear unit (ReLU), same loss function - mean squared error (MSE) and different optimizer - stochastic gradient descent (SGE). To prevent overfitting, we use the regularization method Dropout to ignore some weight updates and activation while training the network. Ending in two dense layers which allow for deep connection to the preceding layer focusing on the multivariate output of the network.

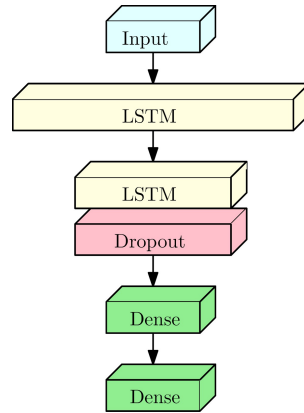


Figure2. LSTM Architecture (Jain et al., 2018)

By fractionating the input data to enable for validation split, we allow for the model to train over one partition and validate against the other. With each

propagation through the network we consider mini-batches. Training of the model runs over many epochs considering minimizing training loss and validation loss.

The forecasted output focuses only on what would be the next value in the time-series which in our case is the next month. Therefore, we consider multiple steps in the future using an iterative-based method, with the output of each previous step as one of the inputs for the next step. The main disadvantage of this method is that the error accumulates with each successive step later to a very large number, as noted in (Yunpeng et al., 2017).

We evaluate the proposed model using the forecasted values compared with the trailing real values for all variables together, by calculating the errors as in (Lv et al., 2022). With p_t as predicted, y_t as real and $\bar{y}$ as average of the real values for the forecasted n -months. We choose following statistics to assess the consistency and deviation between predicted and observed stock values. Equation (1) Mean Absolute Error (MAE) measures the average magnitude of errors in the forecasted set. Equation (2) Root / Mean Square Error (RMSE/MSE) is a quadratic scoring rule that also measures the average magnitude of the error. Equation (3) Mean Absolute Percentage Error (MAPE) measures the percentage error of the forecast in relation to the real values. Equation (4) R-squared (R^2) is a statistical measure in a regression model that determines the proportion of variance in the dependent variable that can be explained by the independent variable, with higher value usually meaning higher prediction accuracy.

$$MAE = \frac{1}{n} \sum_{t=1}^n |p_t - y_t| \quad (1)$$

$$RMSE = \sqrt{\frac{1}{n} \sum_{t=1}^n (p_t - y_t)^2} \quad (2)$$

$$MAPE = \frac{1}{n} \sum_{t=1}^n \left| \frac{p_t - y_t}{y_t} \right| \times 100 \quad (3)$$

$$R^2 = 1 - \frac{\sum_{t=1}^n (p_t - y_t)^2}{\sum_{t=1}^n (p_t - \bar{y})^2} \quad (4)$$

By conducting a linear regression with regression line $y=ax+b$, we assess the correlation between the real and predicted values for all variable's min, max, open and close stock prices. The standard error (SE), p -value and t -value are used to quantify the relationship of the regression analysis. With n -months considered, we observe the slope a with expectation of low SE for a , p under the standard cutoff of 0.05 and high t for a .

EXPERIMENTAL RESULTS AND DISCUSSIONS

In this section, we present the forecasting ability of the model for the MSE datasets, with specific model parameter settings including the metrics, forecasted results, errors and analysis.

Datasets

The dataset is collected from MSE for several companies: KMB, ALK, GRNT and MPT. These companies are present in the MSE monthly reports in the range from 01-2006 to 06-2022. We consider the statistical analysis of the opening price of each company with the count, average, maximum, minimum, standard deviation and ADF test results found in (Table 1). We can see a large gap between minimum and maximum and a large standard deviation, indicating that the opening values present good volatility within the research range. In the ADF test for KMB and ALK p -values are greater than significant value 0.05. Thus, the series is non-stationary and volatile. GRNT and MPT exhibit the opposite characteristic.

Table 1. Descriptive statistics of the opening stock price

Company	count	Mean	min	max	std	ADF*
KMB	198	4374.262	1700	12800	2555.358	0.996
ALK	198	7150.914	3000	18849	3906.562	0.951
GRNT	198	795.590	263	2717	452.281	0.040
MPT	198	43301.348	12000	173000	29958.697	0.047

* $p > 0.05$

Model Parameters and Metrics

To build the LSTM network we use Keras (Chollet, 2015) and its sequential model structure and train it with the model parameters listed in (Table 2).

In (Figure 4) the plot represents the number of epochs, the training loss, validation loss and accuracy. With each epoch, the training models are modified over time with the intent to minimize the loss with each successive batch. The training loss is a metric used to assess how a learning model fits

the training data, and the validation loss is a metric used to assess the performance of the model on the validation set. Here we observe that the validation loss and training loss decrease, and their difference tends to minimize with each successive epoch. However, KMB and ALK margins are lower and have some overfitting, compared to MPT and GRNT larger margins and some underfitting. For KMB and ALK the training loss is larger than the validation loss, and opposite for GRNT and MPT.

Table 2. Parameter details for the model

Parameter	Explanation	Values
LSTM layers	Number nodes of first and second layer	128,32
Dropout rate	Ignore 30% updates of nodes	0.300
Dense	Reduce the dimensions	16,4
Forecast shape	Monthly input data structure for training vs forecast	3 vs 1
n-months	How many months in the future we want to forecast	1,3,6
Batch size	Pass through the network at one time	4
Optimizer	Adjuster algorithm for weight updates	SGD
Loss function	Goal function to minimize the loss	MSE
Activation	How the weighted input transforms into output	ReLU
Validation split	Split the series training vs prediction	0.300
Epochs	Number of trainings	100

Here, for the accuracy we observe that the model vaguely increases its and remains within the 20%-45% range. Thus, making this model less accurate than the ones examined in (Soni et al., 2022) with more than 50% accuracy. Suspecting of possible improvement if we introduce more input training data. As we are focusing on monthly reports that would mean to increasing the time range. However, MSE's earliest monthly data available is from 2002 onwards compared to our examination since 2006 onwards.

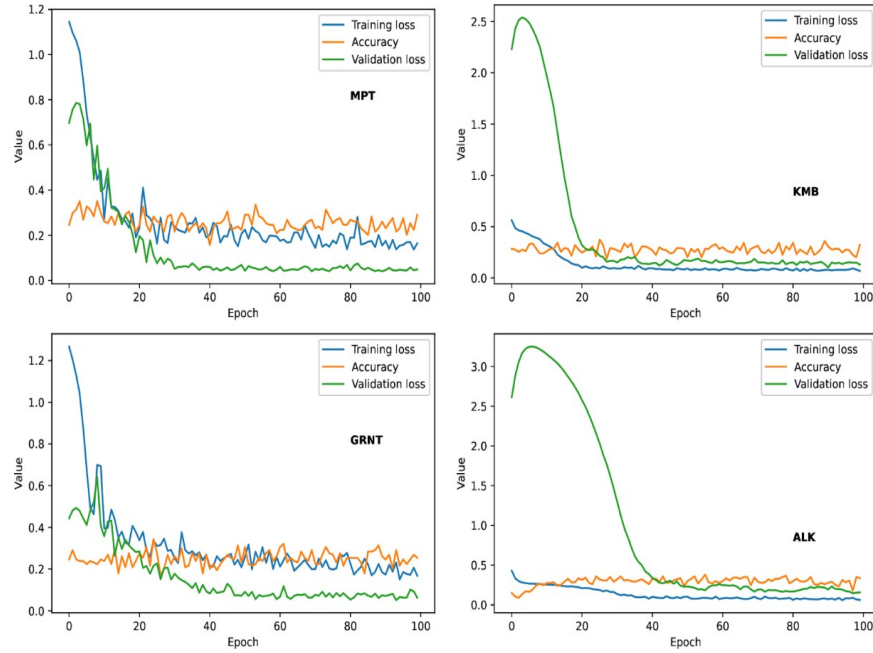


Figure 4. Training loss, validation loss and accuracy for the 3-months prediction model

Forecast Results and Errors

As demonstrated in (Figure 5 and Figure 6) focusing on the forecasted interval 3 or 6 months, we verify the consistency between the real and the predicted values. We observe a similar scenario as before where KMB and ALK show closer relation between the forecasted and the real values compared to MPT and GRNT considering their volatility, hence we notice their predictions span further away from the real values. Similarly, with the observation over the trend of the stock, for KMB and ALK the model yields expected trend compared to MPT and GRNT.

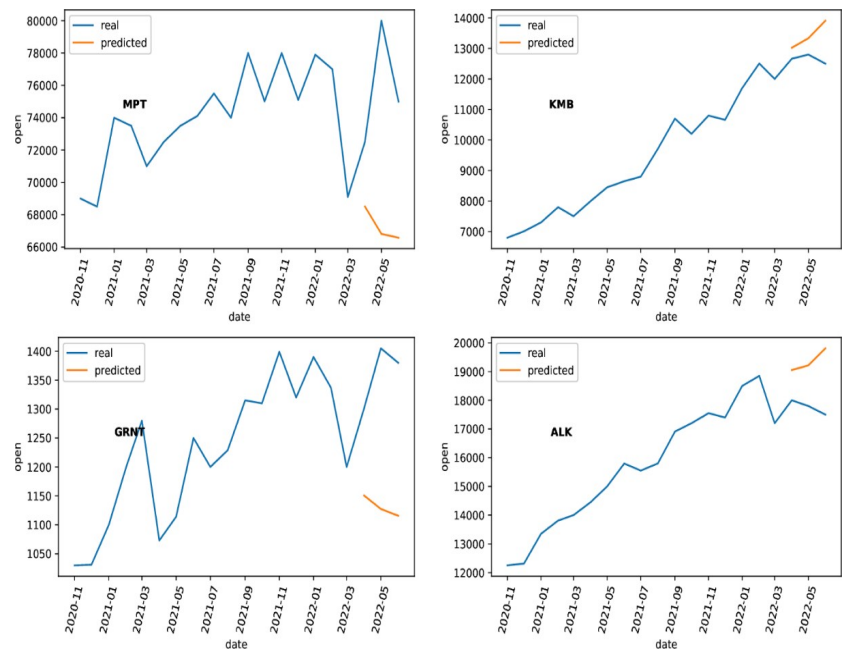


Figure 5. Opening price 3-months prediction

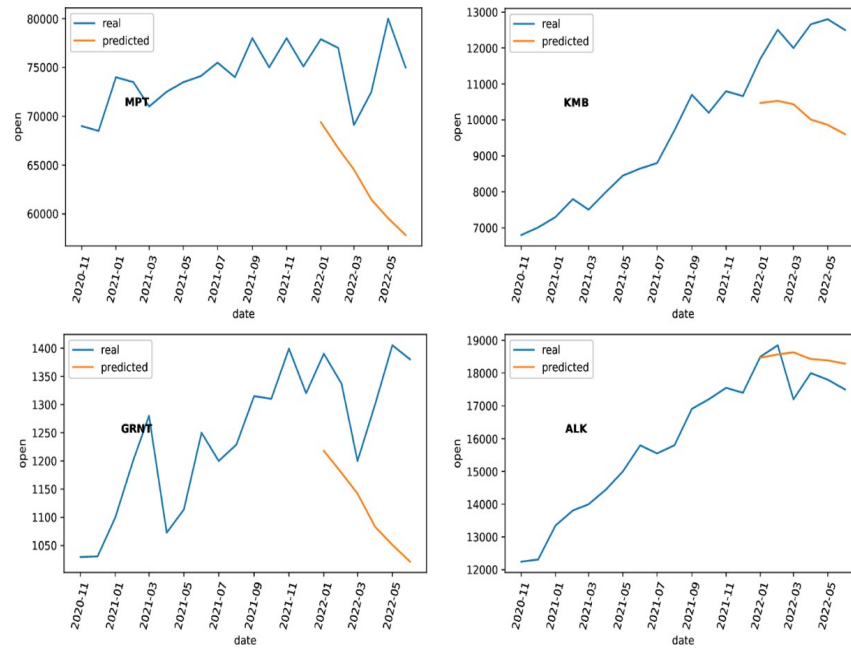


Figure 6. Opening price 6-months prediction

We calculate the errors from the forecasted against the trailing values for all variables combined. From (Table 3 and Table 4) we observe that with the increase of forecasted n -months we get an increase of errors for each company, except for GRNT and ALK previously with affinities towards separate groupings. Common interpretation of R^2 is how well the regression model explains observed data. The quality of the statistical measure depends on many factors, such as the nature of the variables employed in the model, the units of measure of the variables, and the applied data transformation. A low R^2 value is generally an unwanted sign for forecasting models. Compared to the daily prediction models in (Lv et al., 2022) which tend towards 1 for R^2 , our model resulted with low-mid values, thus its usage in terms of practical application as a forecasting model are questionable.

Table 3. 3-months forecasting arguments

Company	MAE	MSE	RMSE	MAPE %	R^2
KMB	955.055	1478547.204	1215.955	7.726	0.360
ALK	1903.621	4962589.673	2227.687	10.749	0.362

GRNT	250.008	65678.229	256.277	18.238	0.516
MPT	7823.027	75957272.567	8715.346	10.340	0.359

Table 4. 6-months forecasting arguments

Company	MAE	MSE	RMSE	MAPE %	R ²
KMB	2206.231	5117299.695	2262.144	17.946	0.463
ALK	1582.128	3136657.076	1771.061	8.938	0.521
GRNT	224.053	63151.615	251.299	16.579	0.460
MPT	11495.724	160620802.325	12673.626	15.339	0.231

Regression Analysis

With the observation over the slope for all companies in (Table 5 and Table 6) regression analysis with p lower than the standard cutoff of 0.05, relatively high standard error and positive t . Similarly, observing the plots in (Figure 7 and Figure 8) the points are scattered unevenly with low tendency towards the diagonal. In addition, observing the *min* and *max* values on the real and predicted axis, and their relative difference we expect larger error for the forecast. Therefore, the regression analysis does not yield a practically applicable forecasting model.

Table 5. 3-months forecast regression results

Company	Parameter	Estimation	SE	t	p
KMB	a	0.314	0.133	2.374	0.039
	b	8176.079	1766.218	4.629	0.001
ALK	a	0.171	0.072	2.386	0.038
	b	14294.943	1372.329	10.417	0.000
GRNT	a	0.455	0.140	3.266	0.008
	b	861.711	157.201	5.482	0.000
MPT	a	0.450	0.190	2.371	0.039
	b	45029.979	12884.822	3.495	0.006

Table 6. 6-months forecast regression results

Company	Parameter	Estimation	SE	t	p
KMB	a	0.595	0.137	4.359	0.000
	b	6279.168	1381.053	4.547	0.000
ALK	a	0.252	0.051	4.901	0.000

GRNT	<i>b</i>	13373.590	907.525	14.736	0.000
	<i>a</i>	0.135	0.154	0.884	0.386
	<i>b</i>	1185.212	171.502	6.911	0.000
MPT	<i>a</i>	0.309	0.120	2.571	0.017
	<i>b</i>	55200.678	7656.972	7.209	0.000

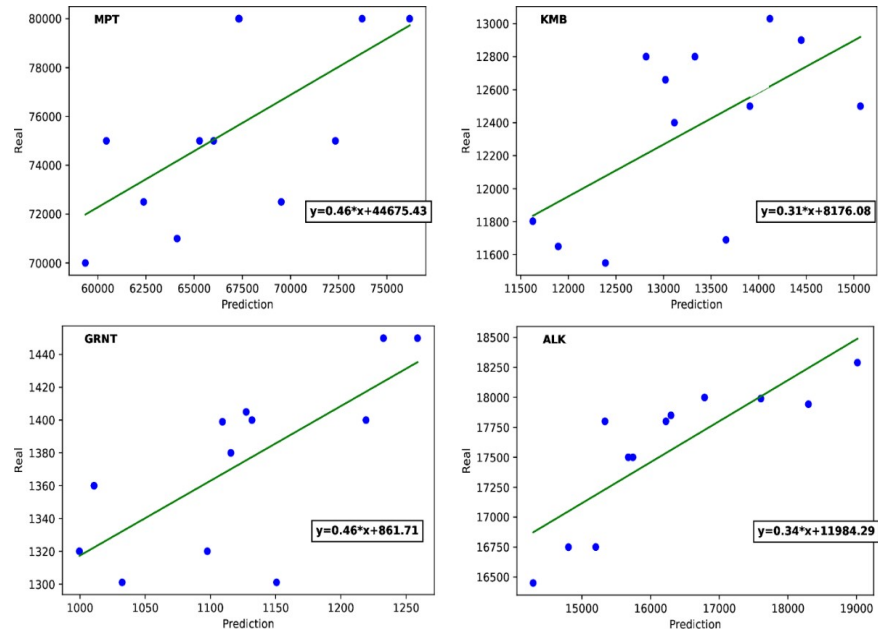


Figure 7. Linear regression analysis 3-months

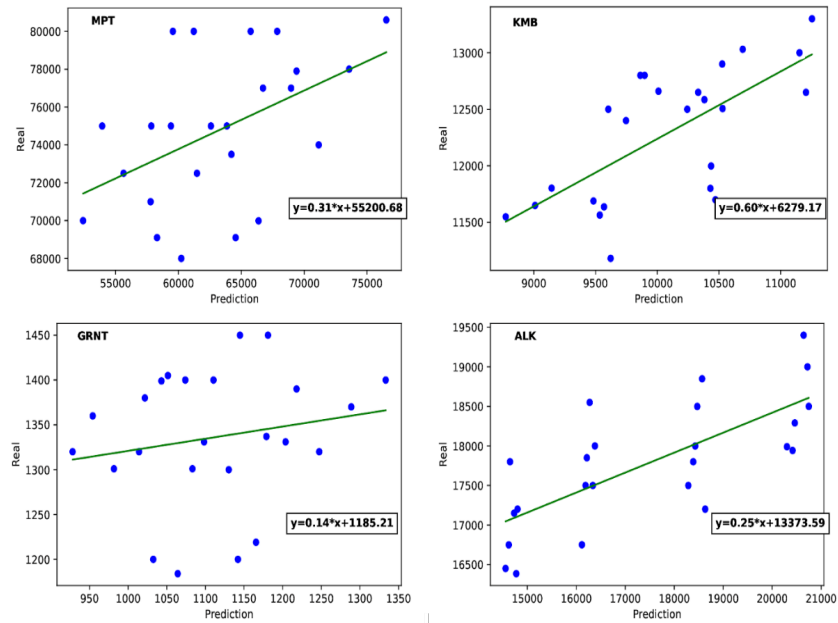


Figure 8. Linear regression analysis 6-months

CONCLUSION

In some financial markets, especially where volatility is often occurring, a prediction model, even a deep learning model, has limited forecasting ability. Considering the empirical evidence and the limitations of input data as we only focus over the monthly reports indirectly inhibiting the model. Training with vast amounts of data is preferable for any deep learning model. Thus, our model showed signs of some overfitting and some underfitting depending on the companies and training chosen. From the results it is evident that this model can provide useful forecasts for specific companies with specific volatility. However, is not recommended for making investments over the resulting data. The linear regression analysis verified that focusing over a shorter period of 3-months offers better forecasts compared to the 6-months. Therefore, such approach does not yield a practical application forecasting model for every company at MSE.

Future research can consider the daily and weekly reports from the MSE, with the adjustment of the number of forecasted steps. Such

forecasting approach will input more training data investigating the effect over the accuracy and error.

REFERENCES

1. Chollet, F., & others. (2015). Keras. <https://Keras.io/>; GitHub repository. <https://github.com/fchollet/keras>
2. Gong, J., & Sun, S. (2009, June). A new approach of stock price prediction based on logistic regression model. 2009 International Conference on New Trends in Information and Service Science. <http://dx.doi.org/10.1109/niss.2009.267>
3. Hochreiter, S., & Schmidhuber, J. (1997). Long short-term memory. *Neural Computation*, 9(8), 1735–1780. <https://doi.org/10.1162/neco.1997.9.8.1735>
4. Jain, S., Gupta, R., & Moghe, A. A. (2018, December). Stock Price Prediction on Daily Stock Data using Deep Neural Networks. 2018 International Conference on Advanced Computation and Telecommunication (ICACAT). <http://dx.doi.org/10.1109/icacat.2018.8933791>
5. Khodabakhsh, A., Ari, I., Bakır, M., & Alagoz, S. M. (2020). Forecasting multivariate time-series data using LSTM and mini-batches. In *Data Science: From Research to Application* (pp. 121–129). Springer International Publishing. http://dx.doi.org/10.1007/978-3-030-37309-2_10
6. Lv, P., Wu, Q., Xu, J., & Shu, Y. (2022). Stock index prediction based on time series decomposition and hybrid model. *Entropy*, 24(2), 146. <https://doi.org/10.3390/e24020146>
7. Okičić, J. (2015). An empirical analysis of stock returns and volatility: The case of stock markets from Central And Eastern Europe. *South East European Journal of Economics and Business*, <https://doi.org/10.2478/jeb-2014-0005>
8. Shah, D., Isah, H., & Zulkernine, F. (2019). Stock market analysis: A review and taxonomy of prediction techniques. *International Journal of Financial Studies*, 7(2), 26. <https://doi.org/10.3390/ijfs7020026>
9. Soni, P., Tewari, Y., & Krishnan, D. (2022). Machine learning approaches in stock price prediction: A systematic review. *Journal of Physics: Conference Series*, 2161(1), 012065. <https://doi.org/10.1088/1742-6596/2161/1/012065>
10. TensorFlow Developers. (2022, May 23). TensorFlow. Zenodo. <https://zenodo.org/record/6574269>

11. Yunpeng, L., Di, H., Junpeng, B., & Yong, Q. (2017, November). Multi-step ahead time series forecasting for different data patterns based on LSTM recurrent neural network. 2017 14th Web Information Systems and Applications Conference (WISA). <http://dx.doi.org/10.1109/wisa.2017.25>
12. Zlatko Kovačić. (2007). Forecasting volatility: Evidence from the Macedonian stock exchange. MPRA Paper 5319. University Library of Munich, Germany.