

Forecasting of the Stock Price Using Recurrent Neural Network – Long Short-term Memory

Michal DOBROVOLNY¹, Ivan SOUKAL^{1*}, Ali SALAMAT², Anna CIERNIAK-EMERYCH³ and Ondrej KREJCAR¹

¹ University of Hradec Kralove, Hradec Kralove, Czech Republic; michal.dobrovolny@uhk.cz; ivan.soukal@uhk.cz; ondrej.krejcar@uhk.cz

² Universiti Teknologi Malaysia, Kuala Lumpur, Malaysia; aselamat@utm.my

³ University of Economics in Wroclaw, Wroclaw, Poland; aemerych@wp.pl

* Corresponding author: ivan.soukal@uhk.cz

Abstract: We employ a recurrent neural network with Long short-term memory for the task of stock price forecasting. We chose three stocks from the same sub-industry: Visa, Mastercard, and PayPal. This paper aims to test the LSTM network's prediction on stock prices and propose the best settings for selected stock price forecasting. The secondary goal is to assess how the settings differed in the case of two highly correlated stocks (Visa-Mastercard year correlation coefficient average: 0.97) and the case of only weak correlated stock (Visa-PayPal correlation coefficient average: 0.39). We tested 117 different settings of LSTM neural networks. The settings differed by the number of epochs/splits (from ten to fifty-eight by the step of four) and the range (minute, hour, and day). Our dataset was the stock price from 1.6.2020 to 15.1.2021. The best performing network has been trained on a 10-day period for Visa and 10-minute for Mastercard and PYPL. However, the differences were negligible, so we did not find the number of epochs as a key setting, unlike in the case of FOREX.

Keywords: neural networks; stock price; forecasting; long short-term memory

JEL Classification: C53; C58

1. Introduction

Machine learning is a very promising and topical field. It counts among awards winning technology in many fields. Researchers are successfully finding more and more fields where Artificial intelligence (AI) can apply. Also, there are already proven fields — for instance, natural language processing, time series or image data (Abdel-Nasser & Mahmoud, n.d.; Peña-Barragán et al., 2011). In image data processing can be found examples about fixing an image (Wolterink et al., 2017; Yang et al., 2018), compression (Sun et al., 2020), super-resolution (Dobrovolny, Mls et al., 2020), image classification (Mambou et al., 2018) or forecasting (Abbasimehr et al., 2020; Chalvatzis & Hristu-Varsakelis, 2020; Dobrovolny, Soukal et al., 2020; Li et al., 2020; Zhang et al., 2021). Regardless of the field, researchers constantly try to improve their models with many variations and settings.

The aim of this paper is to determine what time period and the number of epochs is the best for the prediction of selected stock prices. We focused on three stocks – VISA,

MasterCard and PayPal with NYSE tickers V, MA, and PYPL. Such stocks were deliberately chosen to test how Recurrent neural network (RNN) will handle the titles from the same sub-industry (Financials – Credit/Payment Services) where moreover, two of them are expected to be strongly correlated and the remaining one is close to the role of disrupting competitor/innovator for more than two business cycles. Therefore, we summarize the research questions related to the main goal. Firstly, will the prediction result in the same time splits and overall quality for V and MA stocks? Secondly, will the prediction result in the same time splits and overall quality for V and PYPL from the same sub-industry?

This paper is divided into several parts. In the first chapter, we described areas of using neural networks for forecasting time series. In the second chapter, we described the basic RNN concept and LSTM network architecture. In the third section we presented our experiment and results. In the last chapter, we discuss our results and future work.

2. Methodology

2.1. Topic Overview

Regarding the Web of Science database, the topic of forecasting stock prices is increasing. We used a query ALL=(FORECAST AND STOCK AND PRICE AND NEURAL). Since 2014 there is more than double number of published articles; see Figure 1.

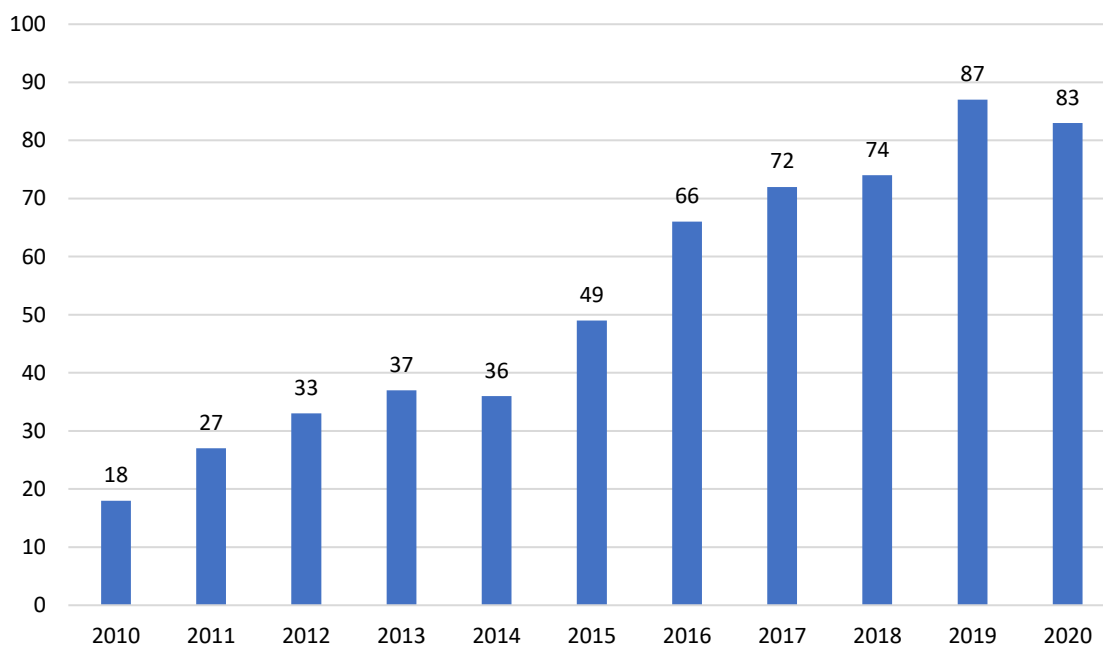


Figure 1. Yearly count of articles published on the Web of Science.

Keyword analysis in Figure 2 shows the connection between neural networks and keywords such as forecasting, stock-market, prediction in a total of the top 50 keywords in 4 clusters. The minimum occurrence of keywords was five times. The amount of all used keywords is 2182 from 757 articles. This analysis suggested the suitability of using RNN for stock price prediction. We also found out there are not many research papers using long short-term memory (LSTM).

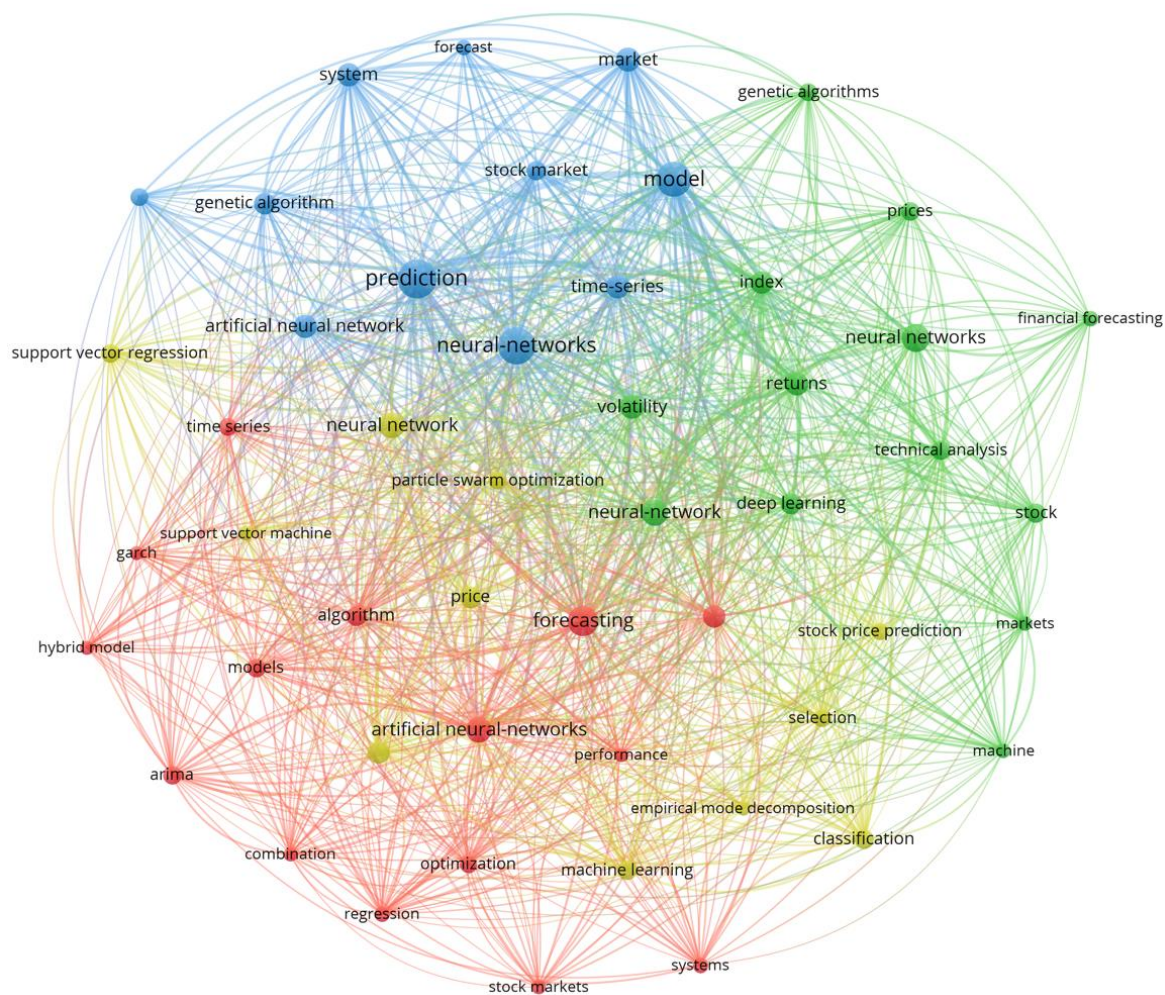


Figure 2. Keyword head map of a search for " FORECAST AND STOCK AND PRICE AND NEURAL" phrase.

2.2. Recurrent Neural Networks

If the RNN series is too long, it would be difficult for them to transfer data from the measures to later ones. So, if we try to process a paragraph of time series to make predictions, from the beginning, RNN's can leave out important details.

RNN's suffer from the vanishing gradient problem during backpropagation. Gradients are values used to update the weight of a neural network. The dilemma of the gradient disappearing is when the gradient shrinks as it propagates back over time. If the gradient value is extremely small, it does not lead to too much learning.

So, layers that obtain a small gradient update avoid learning in recurrent neural networks. Those are the earlier layers, typically. So RNN's can forget what they saw in longer sequences because these layers do not remember, since they have a short-term memory (Hochreiter & Schmidhuber, 1997).

As a recurrent neural network, an LSTM has an identical control flow. This processes information that passes on data as it propagates forward. The variations are the operations inside the cells of the LSTM. These activities are used to allow the LSTM to retain or forget data. It can get a little confusing now to look at these activities, so we'll go through this step by step.

LSTM's core concept is the cell state, and there are different gates. The cell state acts as a transport highway that transmits relative data all the way down the sequence chain. You can think of it as the network's "memory". In theory, the cell state will hold relevant information during the sequence's processing. So even data from the earlier time steps will lead to later time steps, decreasing the short-term memory impact. Data is added or removed to the cell state through gates as the cell state goes on its journey. The gates are various neural networks that determine which cell state data is enabled. Finally, as Hochreiter and Schmidhuber (1997) mentioned, during a training, the gates will learn what data is necessary to retain or forget. An alternative to LSTM is Gated Recurrent Units (GRU).

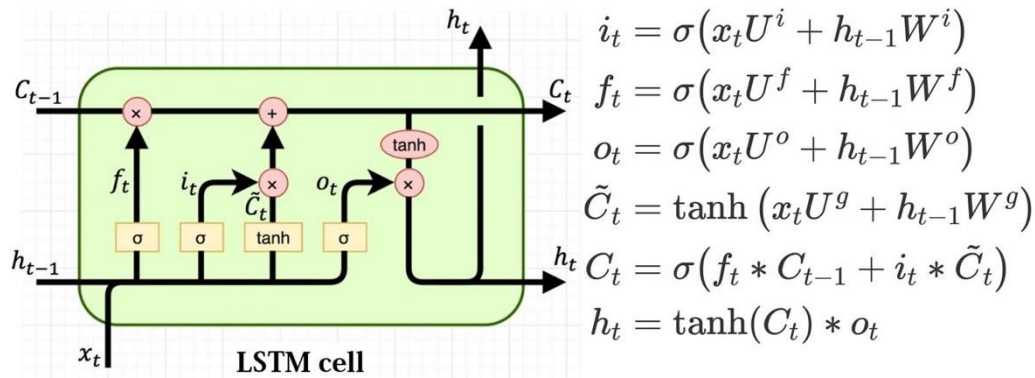


Figure 3. An illustration of LSTM cell with memory pipe (Varsamopoulos et al., 2020).

2.3. Dataset, HW and Data Treatment

As a programming language, we decided to use Python and its packages. Available packages were the main reason to choose the Python as a programming language. For data, we used package yfinance. This package provides a powerful API to download stock data on any point of history when they were available. It also provides a possibility to choose a time interval the value of a particular stock is aggregated into. For instance, the minute, hour or day period. This great opportunity we did use in our work to determine the best time period to make predictions which was the main purpose of this paper.

Data we used were stock values from 1.6.2020 until 15.01.2021. We tested intervals of 1 minute, 1 hour and 1 day. This dataset contained various values – Date as an index, Open, High, Low, Close, Adj Close, and Volume. In every phase of training and evaluation, we used only the High value.

LSTM, RNN, and NN generally are very performance demanding. Our computation computer has two dedicated cards with a total of 7,934 CUDA cores. These cards are one of the top-performing gaming cards of current NVIDIA cards. Because of framework support, we decided to use NVIDIA cards only. One of our cards is 1080TI with 11,176 MB graphic memory and 1.607 MHz max clock rate. Another one is 2080TI with 11,019 MB of graphics memory and 1,545 max clock rates. The used processor is i7-8700 with a 3.20 GHz clock.

Prediction validation was calculated by MSE and mean absolute error (MAE). Both methods are commonly used for neural network loss calculation. In our case, the used loss function was MSE

$$MSE = \frac{\sum_{t=1}^N e_t^2}{N} \quad (1)$$

where e_t is the difference between the predicted and the actual value, N is a number of observations.

3. Results

Stock prediction is a very exciting field of study. Firstly, we focused on the correlation between the stocks. All are from the same industry providing substitutional services at a similar range. The daily data from the last five year timespan showed an extremely tight relation of price quotes between V and MA, see Figure 4. This relation became almost perfect in 2017 and 2019, reaching a correlation of 0.985. The correlation coefficient average was 0.97. The difference between V-MA open and close prices correlation was less than 1 %. A very tight relation was also present in the daily volume correlation analysis showing a correlation of 0.71. The price of PYPL is correlated to V, hence to MA as well, nevertheless the relation of a moderate to low strength for open prices for which the correlation coefficient average reached 0.39. A much stronger relation showed close prices with a correlation coefficient ranging from moderate to high positive relation strength values. The difference between V-PYPL open and close prices correlation was ranging from 18% to 66%.

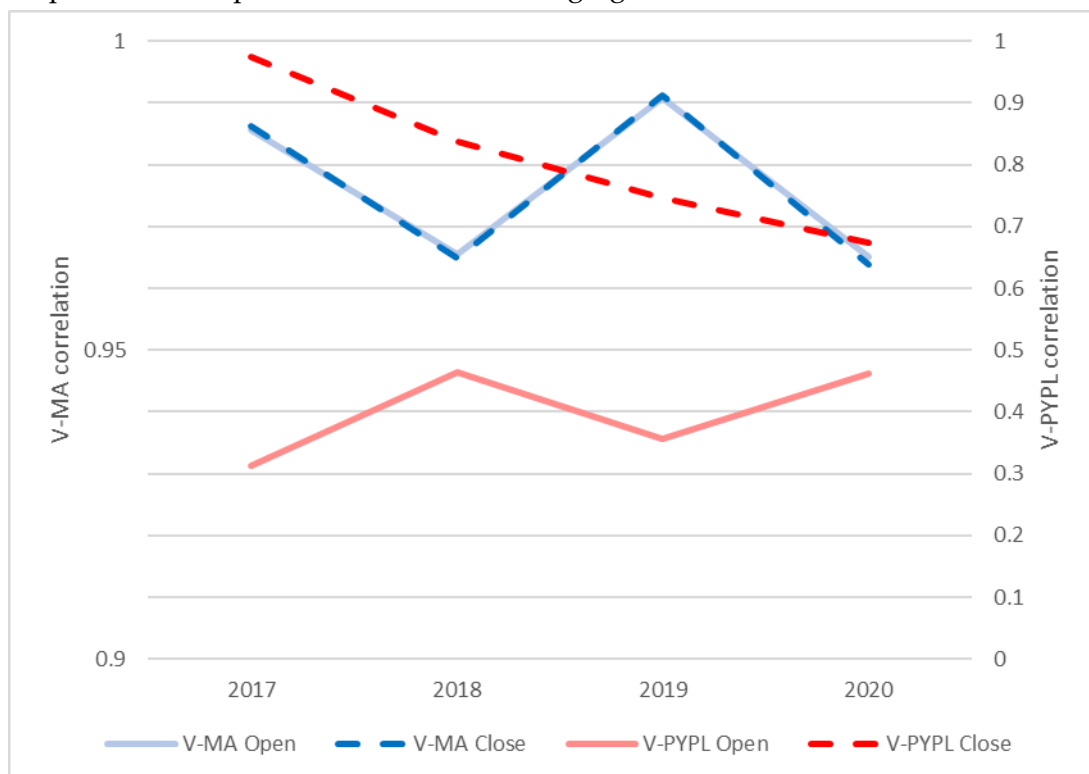


Figure 4. Correlation coefficients per year for V-MA (left axis) and V-PYPL (right axis) stock price quotes.

The weakest relation was found in the daily traded volume analysis. V-PYPL correlation coefficient of 0.37 indicated only a weak positive relation.

Our simple model with only one LSTM layer shows excellent performance on this dataset. The architecture we tested was constructed on top of LSTM layer. Our model

contains three LSTM units and one dense layer. The used optimizer was RMSprop, and the measured loss was MSE. We tested different time splits/epochs ranging from 10 to 58 by the step of four periods to test the impact of different time periods. In total, we have done 117 tests. All our experiments had batch size 32. We tested all stock for time periods 1 minute, 1 hour and 1-day range.

Figure 5 shows MSE values results for all three stocks based on minute range. We performed minute, hour, and day range-based prediction; however, for the demonstration, we chose only the minute range variant because it contained minimal MSE value for two out of three stocks.

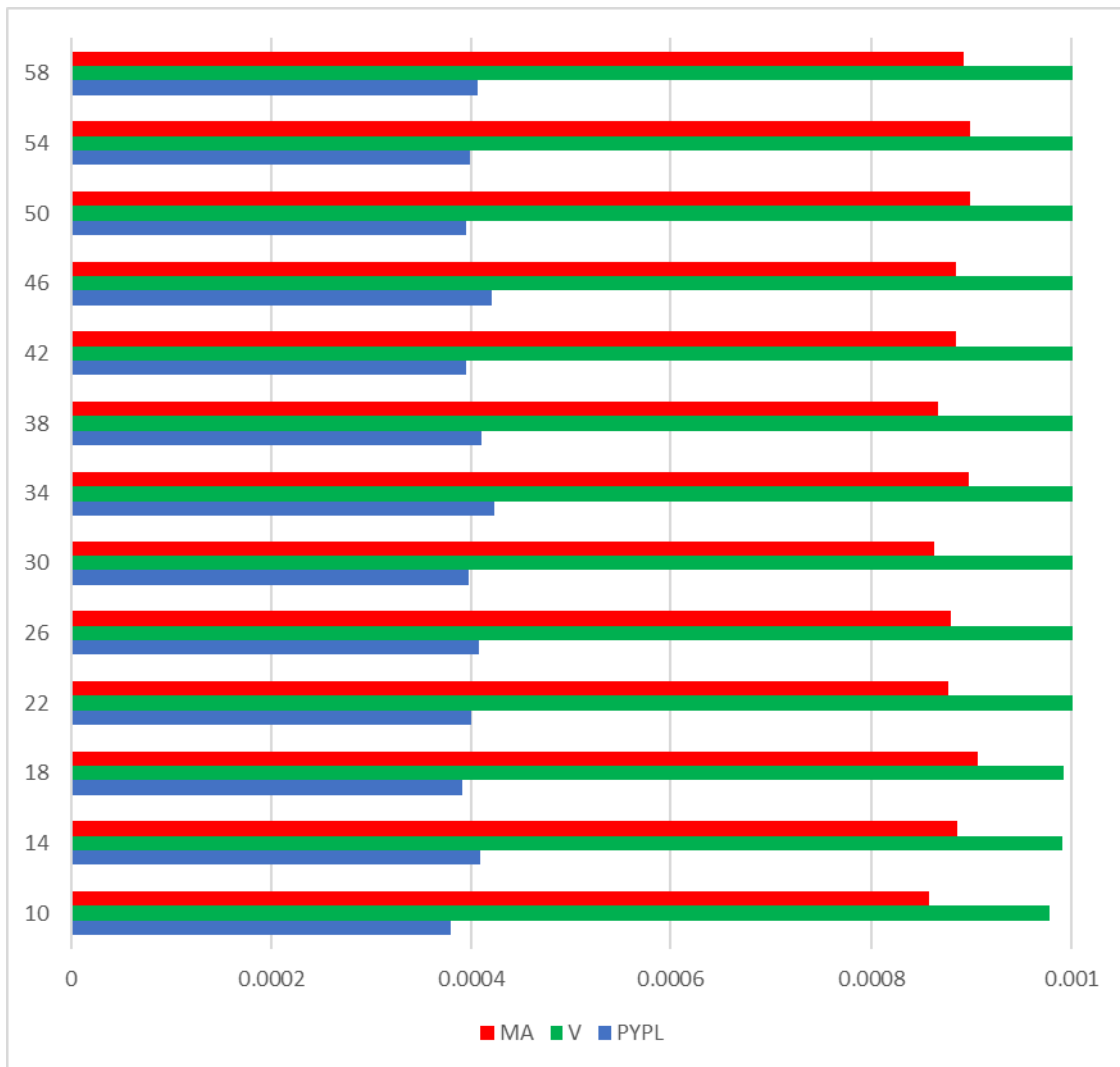


Figure 5. MSE values for MA, V, and PYPL stocks based on minute range and different epochs.

Firstly, the best quality showed prediction for the PYPL stock across the time split periods or epochs. Secondly, the lowest MSE values were calculated at 10 splits regardless of a stock. Thirdly, the difference between V and MA MSEs was less than 0.002. This is three-times lower than the average MSE difference of 0.006 between V and PYPL. Fourthly, the difference between MSE minimum and maximum was as small as 6.65E-5 for V, 8.99E-5 for MA, and 5.64E-5 for PYPL.

As mentioned before, we also tested the impact of different time ranges on the predicted value. We tested minute, hour, and day ranges. The optimal time range was minute one for PYPL and MA, and day one for V. The optimal number of epochs/splits was, regardless of the range, ten. Figure 5. presents MSE minimal results for different ranges.

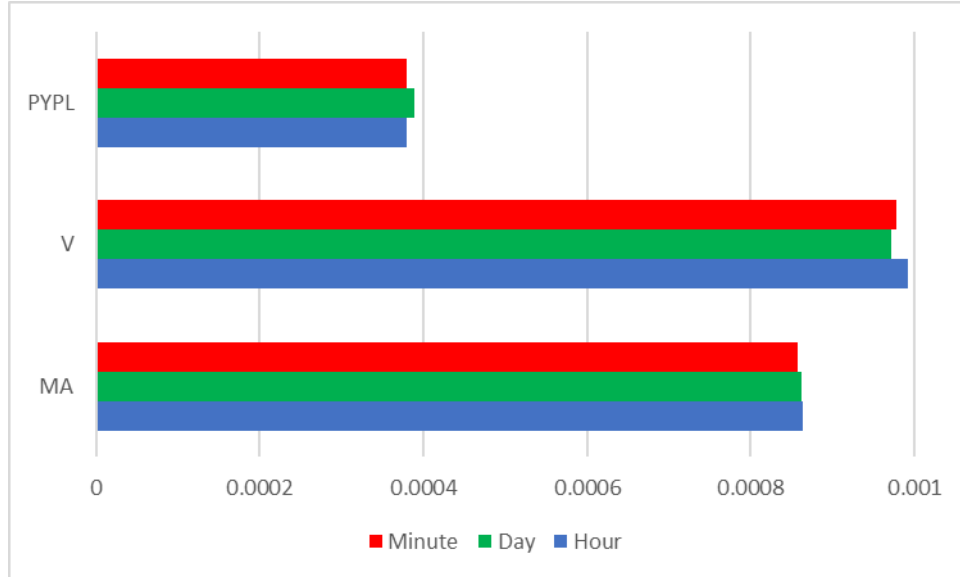


Figure 5. Comparison of MSE on different time period per stock.

Figure 5 shows that the difference in MSE minimal values were almost the same regardless of the range. To finish the presentation of the results, we provide a demonstration of the price chart with a prediction so there can be assessed how successfully our prediction matched the real price move for ten epoch and 1-minute range settings, see Figure 6.

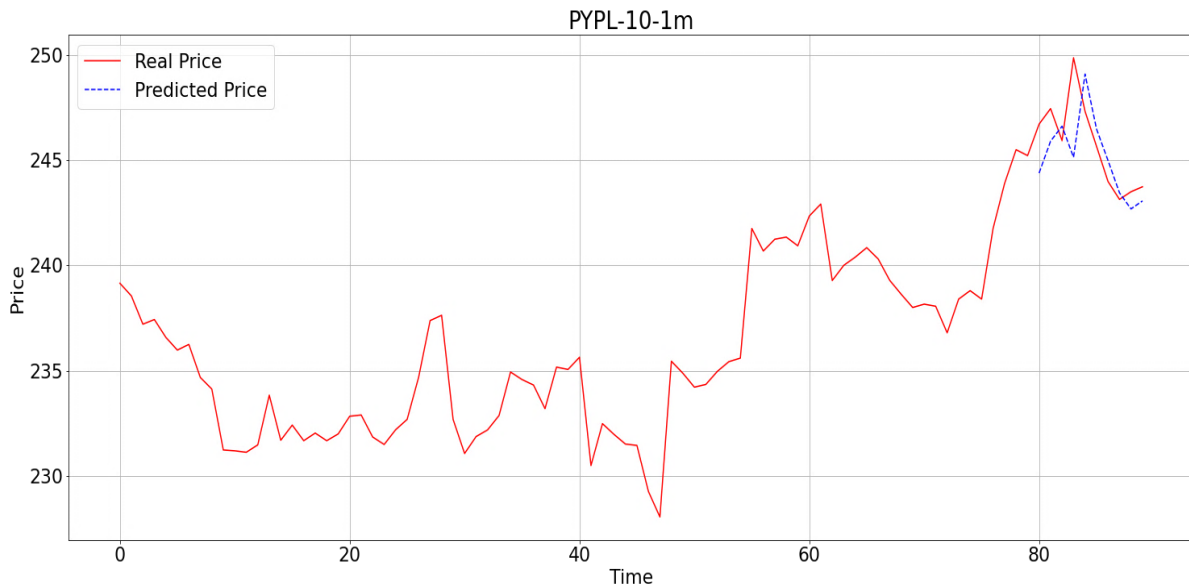


Figure 6. Predicted price for next 10 minutes of PYPL stock with input size of 10 minutes.

4. Discussion

Firstly, we interpret the correlation analysis. Correlation coefficients prove an extremely tight relation in V-MA stock price quotes and trading volumes since 2017. That is more than

two years before our own testing timespan starts. To reach the main goal of finding optimal time splits, we indeed managed to pick two separate, yet almost identical, companies from the stock price point of view. In the case of V-PYPL or due to a tight V-MA relation any V-PYPL can be considered to be almost identical to MA-PYPL, we found a positive however, much weaker relationship. Unlike V-MA the price quotes showed not only moderate relationship in price quotes but the traded volume analysis resulted in only weak positive relationship. Mainly open price quotes correlation manifested weaker bond which even decreased over time. We claim that in the case of PYPL we selected a company from the same sub-industry providing a similar range of services; however, its price changes had only weak relation to V or MA and so we managed to select the stock that differs from V or MA in terms of price shifts.

Secondly, our analysis of different ranges, as well as different epochs/splits, resulted into very little prediction quality variation. The difference between MSE minimum and maximum was extremely low. This applied to both in the same range and even when comparing ranges altogether. So yes, the best results were achieved with ten splits/epochs in a month range for PYPL and MA, and ten splits in day range for V. Nevertheless, it was almost as if it does not matter, from a practical point of view, which time split/how many epochs or range we used in settings. From this particular result, we can see that there is a great difference between predicting a stock value and FOREX value, where choosing the best settings had a significant impact (Dobrovolny, Soukal, et al., 2020).

Thirdly, we compare our finding of a low number of epochs but, in general, very little impact on the result quality/efficiency measured by MSE, to other authors. Other authors chose LSTM NN for stock price prediction with varying results. Chalvatzis and Hristu-Varsakelis (2020) employed structurally simple LSTM NN for price prediction generation based on price observations over a similar number of trading days and employed, among other metrics, the same prediction quality assessment by MSE. Their epoch settings were 11, 22, and 44. Although the focus was different, the interesting point is that they also found the shortest epoch as the optimal one, in our case 10, is the case of (Chalvatzis & Hristu-Varsakelis, 2020) it was 11.

Much higher number of splits/epochs were utilized in (Zhang et al., 2021). They tested two settings of 100 and 200 epochs. They assessed quality/efficiency by rootMSE. The best prediction results were found by a feature set (trading, liquidity, traditional attention, and new attention proxies) as model input, with no regard to a number of epochs. The state when the difference between lower and higher was negligible is in accord with our findings.

Study (Li et al., 2020) set the number of training epochs up to 500. Nevertheless, such a high number was never reached. Loss curves in the training set and the validation set reached the minimum at about the 50th epoch and sharply increased in the subsequent training iterations. Six different epoch settings were employed by (Abbasimehr et al., 2020) from 50 to 500. The hyperparameters for the best-resulted LSTM model included 500 epochs by rootMSE.

As mentioned before, the number of epochs choices for LSTM NN stock price prediction models showed varying results. We conclude the discussion section by the claim that our results were in accord with several.

In connection with our current work, we would like to continue in the field of applications such as a stock index prediction, cryptoassets, or commodities, especially gold for which (Král & Olszanska, 2020) claimed that regardless of a period, gold investment was profitable.. At the same time, over the current results, we would like to program a robot for automatic trading, which would be used to verify our results over real-time data in operation.

5. Conclusions

We studied a highly topical issue of machine learning application for stock price prediction. We employed LSTM NN, for which we tested different settings of epochs to find the optimal one assessed by the MSE criterion. We chose for our model three stocks listed in the NYSE: V, MA, and PYPL. All three were selected on purpose to investigate our secondary objective. That was a comparison of prediction for two highly, almost perfectly, correlated stock versus much less, but still making business in the same sub-industry, stock. We confirmed that the relation between V-MA prices is an extremely tight, almost perfect positive correlation. Their close relationship in the three-year timespan was confirmed by also traded volume correlation above 0.7. The relation between V-PYPL was much less strong, mainly regarding open prices showing only a weak positive correlation coefficient. Similarly, the correlation in traded volumes was weak.

The price prediction by LSTM NN results was evaluated by MSE values. We tested different time splits/epochs ranging from 10 to 58 by the step of four periods to test the impact of different time periods. The best results were achieved with ten splits/epochs in a month range for PYPL and MA, and ten splits in day range for V. Nevertheless, it was almost as if it does not matter, from a practical point of view, which time split/how many epochs or range we used in settings. From this particular result, we can see a great difference between predicting stock value and FOREX value, where choosing the best settings had a significant impact. This result was in accord with another study (Zhang et al., 2021) and low number of epochs was in accord with (Chalvatzis & Hristu-Varsakelis, 2020). The rest of the applications utilized significantly more epochs to reach optimal MSE or rootMSE values.

References

- Abbasimehr, H., Shabani, M., & Yousefi, M. (2020). An optimized model using LSTM network for demand forecasting. *Computers & Industrial Engineering*, 143, 106435. <https://doi.org/10.1016/j.cie.2020.106435>
- Abdel-Nasser, M., & Mahmoud, K. (n.d.). Accurate photovoltaic power forecasting models using deep LSTM-RNN. In *Neural Computing and Applications* (pp. 2727–2740).
- Chalvatzis, C., & Hristu-Varsakelis, D. (2020). High-performance stock index trading via neural networks and trees. *Applied Soft Computing*, 96, 106567. <https://doi.org/10.1016/j.asoc.2020.106567>
- Dobrovolny, M., Mls, K., Krejcar, O., Mambou, S., & Selamat, A. (2020). Medical Image Data Upscaling with Generative Adversarial Networks. In I. Rojas, O. Valenzuela, F. Rojas, L. J. Herrera, & F. Ortuño (Eds.), *Bioinformatics and Biomedical Engineering* (Vol. 12108, pp. 739–749). Springer International Publishing. https://doi.org/10.1007/978-3-030-45385-5_66

- Dobrovolny, M., Soukal, I., Lim, K. C., Selamat, A., & Krejcar, O. (2020). Forecasting of FOREX Price Trend using Recurrent Neural Network - Long short-term memory. In P. Maresova, P. Jedlicka, K. Firlej, & I. Soukal (Eds.), *Hradec Economic Days 2020* (pp. 95–103). <https://doi.org/10.36689/uhk/hed/2020-01-011>
- 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>
- Král, M., & Olszanska, A. (2020). Investing in Gold: Good or Bad Choice? 20-Year History. In P. Maresova, P. Jedlicka, K. Firlej, & I. Soukal (Eds.), *Hradec Economic Days 2020* (pp. 382–389). <https://doi.org/10.36689/uhk/hed/2020-01-044>
- Li, X., Wu, P., & Wang, W. (2020). Incorporating stock prices and news sentiments for stock market prediction: A case of Hong Kong. *Information Processing & Management*, 57(5), 102212. <https://doi.org/10.1016/j.ipm.2020.102212>
- Mambou, S., Maresova, P., Krejcar, O., Selamat, A., & Kuca, K. (2018). Breast Cancer Detection Using Infrared Thermal Imaging and a Deep Learning Model. *Sensors*, 18(9), 2799. <https://doi.org/10.3390/s18092799>
- Peña-Barragán, J. M., Ngugi, M. K., Plant, R. E., & Six, J. (2011). Object-based crop identification using multiple vegetation indices, textural features and crop phenology. *Remote Sensing of Environment*, 115(6), 1301–1316. <https://doi.org/10.1016/j.rse.2011.01.009>
- Sun, Y., Chen, J., Liu, Q., & Liu, G. (2020). Learning image compressed sensing with sub-pixel convolutional generative adversarial network. *Pattern Recognition*, 98, 107051. <https://doi.org/10.1016/j.patcog.2019.107051>
- Varsamopoulos, S., Bertels, K., & Almudever, C. G. (2020). Comparing neural network based decoders for the surface code. *IEEE Transactions on Computers*, 69(2), 300–311. <https://doi.org/10.1109/TC.2019.2948612>
- Wolterink, J. M., Leiner, T., Viergever, M. A., & Isgum, I. (2017). Generative Adversarial Networks for Noise Reduction in Low-Dose CT. *IEEE Transactions on Medical Imaging*, 36(12), 2536–2545. <https://doi.org/10.1109/TMI.2017.2708987>
- Yang, Q., Hou, X., & Zhang, L. (2018). Measurement of natural and cyclical excess capacity in China's coal industry. *Energy Policy*, 118, 270–278. <https://doi.org/10.1016/j.enpol.2018.03.052>
- Zhang, Y., Chu, G., & Shen, D. (2021). The role of investor attention in predicting stock prices: The long short-term memory networks perspective. *Finance Research Letters*, 38, 101484. <https://doi.org/10.1016/j.frl.2020.101484>