technical transactions
No. 2022/011
computer sciences
sciendo

How many neurons are needed to make a short-term prediction of the Bitcoin exchange rate?

Michał Frontczak
michal.frontczak@up.krakow.pl | https://orcid.org/0000-0002-5264-4508
Tomasz Hachaj
tomasz.hachaj@up.krakow.pl | https://orcid.org/0000-0003-1390-9021
Pedagogical University of Krakow, Institute of Computer Science

Scientific Editor: Paweł Pławiak, Cracow University of Technology
Technical Editor: Aleksandra Urzędowska, Cracow University of Technology Press
Language Verification: Timothy Churcher, Merlin Language Services
Typesetting: Małgorzata Murat-Drożyńska, Cracow University of Technology Press

Received: March 7, 2022
Accepted: August 9, 2022

Copyright: © 2022 Frontczak, Hachaj. This is an open access article distributed under the terms of the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.

Data Availability Statement: All relevant data are within the paper and its Supporting Information files.

Competing interests: The authors have declared that no competing interests exist.

Citation: Frontczak, M., Hachaj, T. (2022). How many neurons are needed to make a short-term prediction of the Bitcoin exchange rate? . *Technical Transactions, e2022011*. https://doi.org/10.37705/TechTrans/e2022011

Abstract

The goal of our work was to select a neural network architecture that would give the best prediction of the Bitcoin exchange rate using historical data. Our work fits into the very important topic of predicting the value of the cryptocurrency exchange rate, and makes use of recent data which, as a result of the high Bitcoin exchange rate dynamics of the last year, differs significantly from those of previous years. We propose and test a number of neural network-based architectures and conduct a discussion of the results. Unlike previous state-of-the-art works, we conducted a comprehensive comparison of three different neural network-based models: MLP (multilayer perceptron), LSTM (long short-term memory) and CNN (convolutional neural network). We tested them for a wide range of parameters. The results we present are, to the best of our knowledge, the most up to date when it comes to the application of artificial intelligence methods for the prediction of cryptocurrency exchange rates. The best-performing architectures were used for a website that gives real-time predictions of the Bitcoin exchange rate. The website is available at http://stpbtc-ii.up.krakow.pl/. Source codes of our research are available to download in order to make our experiment reproducible.

Keywords: neural network, multiline perceptron, convolutional neural network, long short-term memory, Bitcoin

1. Introduction

The financial industry crisis in 2008 exposed weaknesses and the vulnerability of the markets (Inzirillo, Benjamin, 2021) giving birth to a new, not yet recognized cryptocurrency market. In the beginning, Bitcoin was launched as a payment medium without a broker. Bitcoin is a cryptographic currency (cryptocurrency) that uses a decentralized peer-to-peer system for transaction verification and privacy (Nakamoto, 2009).

Nowadays, virtual currencies are an accepted online payment method for many online services and goods, and they are part of investment products such as contracts for difference, futures contracts, or simple spot contracts. Over the past twelve years, we have witnessed the inception of 10,397 new cryptocurrencies (Number of cryptocurrencies worldwide from 2013 to February 2022, 2022); however, due to their liquidity and a lack of trust, not all of them are available on cryptocurrencies exchanges, i.e. kraken.com exchange allows 118 different cryptocurrencies to trade (Top Cryptocurrency Spot Exchanges, 2022). Bitcoin is the most popular digital currency, with a market value of approximately $758 billion in 2022 (All Cryptocurrencies, 2022). In 2021 alone, the Bitcoin price reached over twice its all-time high (ATH), and gave investors, an opportunity of high revenues in a short time that had never been seen before. These events led more people interested in investing in cryptocurrency.

There are many factors to predict the behaviour of the crypto market, such as news, events, influences, multiple conflicting trading strategies, and unpredictable events resulting in a highly complex, non-linear, and nonstationary system (Nazário, Lima e Silva, Sobreiro, Kimura, 2017; Matic, Packham, Härdle, 2021). Applying the standard methods might not always be sufficient to keep pace with constantly changing, unpredictable markets.

Cryptocurrency investors or traders are using popular techniques from other markets, such as the stock market or, similar to cryptocurrency, the Forex (FX) market. The standard method used by market participants is technical analysis. This utilises a set of tools to predict future returns based on past market data, such as price and volume. It uses indicators or patterns visible on charts to signal to buy or sell a given asset. Another method, using diagrams, is candlestick patterns, which are understood as a pre-defined set of candlesticks that can signal investors to buy or sell. However, this method might have higher chances of success when used with stocks or the forex market; it has no value for cryptocurrency traders (Ho, Chan, Pan, Li, 2021).

The latest techniques combine technical analysis with modern algorithm-based methods such as machine learning and deep learning. The computer's growing computational capabilities have become a new standard for the financial industry to leverage investment strategies with machine-learning (ML) models. For example, in previous research (Sen, Dutta, Mehtab, 2021), the authors used the LSTM model to solve the complex optimization problem of a stock portfolio, in other research (Zhao, Rinaldo, Brookins, 2019), the authors of the paper investigated support vector machines (SVM) to predict the price at 1-hour intervals of the pairs BTCUSD, ETHUSD, and LTCUSD using (OHLCV) data from Coinbase exchange. Convolutional neural networks (CNN) have been explored for predicting forex trend (Tsai, Chen, Wang, 2020; Liu, Li, Li, Li, Xie, 2021), and comparisons of deep reinforcement learning models which use SVM, MLP, LSTM, TCN, transformer to build a strategy for Bitcoin investment have also been conducted.

Another popular technique which combines machine-learning methods is sentiment analysis. This has been covered in articles in which the authors use news information for stock price prediction (Hu, Liu, Bian, Liu, Tie-Yan, 2018; Huang, Capretz, Ho, 2021) or cryptocurrency price prediction based on twitter signals (Li, Chamrajnagar, Fong, Rizik, Fu, 2019). Popular predictive models like Facebook Prophet (Taylor, Letham, 2017) do not perform well on Bitcoin data because this cryptocurrency does not have explicit yearly, weekly, and daily

https://doi.org/10.37705/TechTrans/e2022011

seasonality or plus holiday effects. An extensive survey on cryptocurrencies and price prediction can be found in the literature (Mezquita, Gil-González, Prieto, Corchado, 2022).

Our study is an extension of the topics covered in previous papers (Dutta, Kumar, Basu, 2020; Phaladisailoed, Numnonda, 2018; Ji, Kim, Im, 2019; Mezquita, Gil-González, Prieto, Corchado, 2022; Patel, et. al, 2020; Rane, Dhage, 2019; Ferdiansyah, et. al, 2019; Struga, Olti, 2018; Rizwan, Narejo, Javed, 2019; Mangla, Bhat, Avabratha, Narayana Bhat, 2019). We carried out a broad comparative study of the methods described there as well as extended it by analysing a wide range of parameters of neural networks on the latest Bitcoin data.

The title of our paper is humorous in nature, but it deals with very important and actual issues in the field of economics and finance. In contrast to other research in which the authors use only recurrent neural networks (Dutta, Kumar, Basu, 2020; Phaladisailoed, Numnonda, 2018) and which uses MLP (multilayer perceptron) and LSTM (long short-term memory) (Patel, et. al, 2020; Rane, Dhage, 2019) or only LSTM (Ferdiansyah, et. al, 2019; Struga, Olti, 2018; Rizwan, Narejo, Javed, 2019; Mangla, Bhat, Avabratha, Narayana Bhat, 2019), we proposed the use of four different neural network architectures and a comparison of their performance. Another difference from previously published papers is the fact that the calculations we made were performed on hourly rather than daily rates. This allowed us to test the predictive ability of the model on smaller amounts of time. We also made predictions on several consecutive samples rather than one as in other state-of-the-art works. Additionally, unlike previous work, we consider the prediction results applied to the most recent performance of the Bitcoin exchange rate starting from 2021, when it reached an unprecedented level of volatility. Due to the occurrence of high volatility in the price of this cryptocurrency, which has significantly affected its time characteristics, we believe that the topic of the prediction of the value of Bitcoin cryptocurrency remains an open and challenging task.

Our work fits into the very important topic of predicting the value of a cryptocurrency exchange rate, and makes use of recent data which, as a result of the high Bitcoin exchange rate dynamics of the previous year, differs significantly from those of earlier years. We propose and test a number of neural-network-based architectures and conduct a discussion of the results. Unlike previous state-of-the-art works, we made a comprehensive comparison of three different neural-network-based models: MLP, LSTM and CNN (convolutional neural network). We have tested them for a wide range of parameters. The results we present are, to the best of our knowledge, the most up-to-date when it comes to the application of artificial intelligence methods for the prediction of cryptocurrency exchange rates.

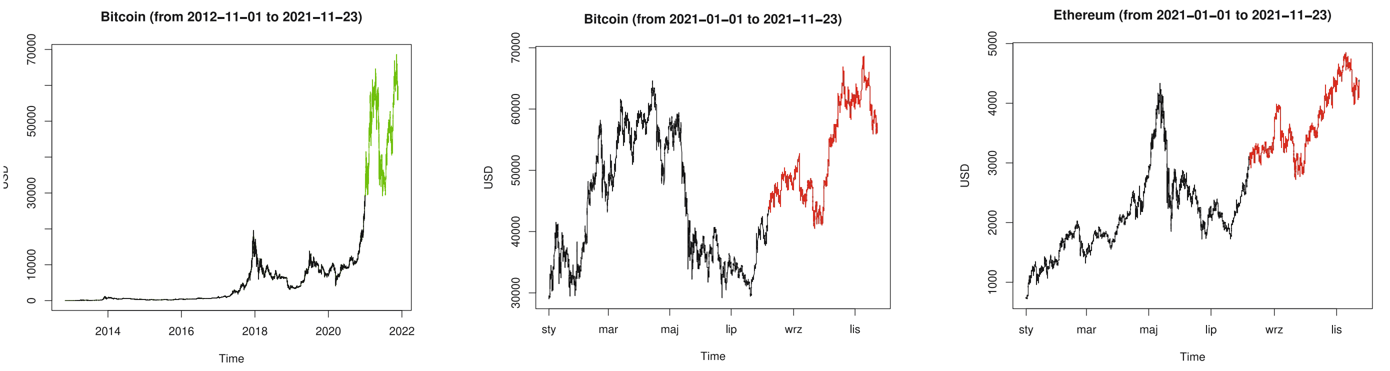
The goal of our work was to select a neural-network architecture that would give the best prediction of the Bitcoin exchange rate using historical data. This best performing architecture was used for a website that gives real-time predictions of the Bitcoin exchange rate. The website is available at http://stpbtc-ii.up.krakow.pl/. Source codes of our research are available to download in order to make our experiment reproducible.

2. Materials and methods

In this section, we will describe the dataset we are working on and the prediction methods based on different neural network architectures that we will use to predict the Bitcoin rate.

2.1. Dataset

The data used for training and testing is publicly available on Bitstamp exchange. We downloaded data relating to BTCUSD and ETHUSD currency pairs in open,



a) b) c)

Fig. 1. Chart (a) shows the values of the close Bitcoin rate throughout the whole available time period when the dataset has been downloaded (November 23, 2021); (b) Bitcoin rate from 1st January 2021 to 23rd November 2021; (c) Ethereum rate from the same time range (own elaboration)

high, low, close, and volume data formats (OHLCV) at 1-hour intervals from Bitstamp public API. The OHLC is commonly used to illustrate price movement over time. One row represents the price changes within the given period. The open and close values are prices at the beginning and end of the period, and the high and low values are maximum and minimum prices during the timeframe. To set a baseline for this research, we focused only on the close price.

The cryptocurrency exchange rate is characterized by very high volatility over time. In recent years, the value of the exchange rates of many cryptocurrencies has greatly increased. Historical exchange-rate values from previous years do not reflect the dynamics of bitcoin volatility that we can observe now. For this reason, we decided to train and test our prediction methods using data from 1st January 2021 to 23rd November 2021. From this range, we used 67% of the initial data to train the prediction algorithms and the remaining 33% to test them.

Figure 1 presents the close values of the dataset we downloaded. In plot (a), green indicates the portion of data on which we trained and tested the network. Plots (b) and (c) show the training set in black and the test set in red.

2.2. Neural-network architectures

In this section, we describe the neutron-network architectures that we will use for Bitcoin exchange-rate prediction. For each network, we tested several ranges of the *look back* parameter, which determined how many previous values of the Bitcoin exchange rate we took into account. The signal is acquired at a frequency of 1 hour. The *look back* values covered 24h (the previous day, 24 samples), 72h (the previous three days, 72 samples), 120h, and 168h. Each architecture was tested to predict several consecutive bitcoin values. This was done so that the network had a different number of output neurons, each corresponding to 1 hour. We tested the prediction (*look forward*) values: 1, 2, 4, 6, 8 and 10 hours.

A. MLP approach

MLP is one of the simplest neural-network models. It can be used for classification, regression or prediction. The network we use consists of three layers fully connected with the ReLu activation function. The first fully connected layer has 2**look back** neurons, the second fully connected layer has *look back* neurons, the third (output) layer with linear activation function has *look forward* neurons (see Figure 2 (a)). The number of parameters of the proposed network architecture ranges from 2401 for *look back* = 24 and *look forward* = 1 to 115090 when *look back* = 168 and *look forward* = 10.

B. CNN approach

Convolutional neural networks (CNN) allow modelling of the relationship between data using convolution operations. Our network is built with two consecutive

https://doi.org/10.37705/TechTrans/e2022011

convolutional layers with 32 one-dimensional filters with a window size of 3. There is then a max pooling layer with a size of 2. The next layer is flattening and then a fully connected layer with 96 neurons with the ReLu activation function. The last layer (output) is with the linear activation function and has *look forward* neurons (see Fig. 2 (b)). The number of parameters of the proposed network architecture ranges from 20225 for *look back* = 24 and *look forward* = 1 to 899354 when *look back* = 168 and *look forward* = 10.

C. LSTM approach

Recursive LSTM networks have been successfully used to model time-varying time series. We used a network that has two LSTM layers with dimensionality of output space equalling 128 and ReLu activation function. The network then has a dense layer with ReLu activation function and 2 * *look back* neurons count. The last layer has *look forward* neurons with a linear activation function (see Fig. 2 (c)). The number of parameters of the proposed network architecture varies from 216210 for *look back* = 24 and *look forward* = 1 to 330362 when *look back* = 168 and *look forward* = 10.

D. CNN with additional Ethereum data

We also used a convolutional neural network with additional input data. In addition to the previous Bitcoin rate, we also add the Ethereum rate data (CNN B+E). The network structure is almost the same as for CNN in Section 2.2 B. The only difference is that it fetches a vector with Ethereum data in addition to the Bitcoin data (see Fig. 2 (d)). The number of parameters of the proposed network architecture ranges from 37313 for *look back* = 24 and *look forward* = 1 to 259370 when *look back* = 168 and *look forward* = 10.

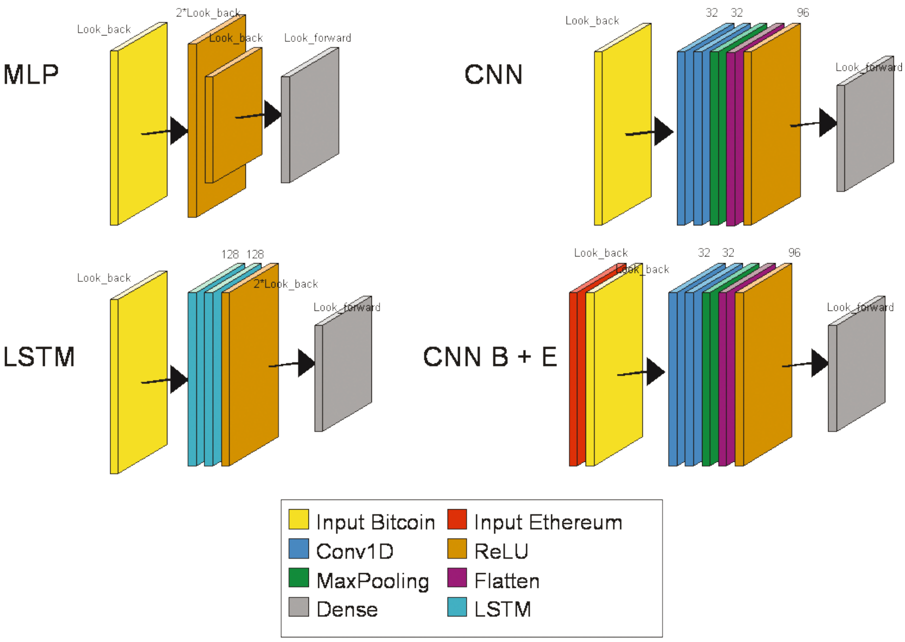


Fig. 2. Diagrams of the neural network architectures we used in our study (own elaboration)

3. Results

We implemented the neural networks described in Section 2 using Python 3.6. We used the Keras 2.4.3 library and TensorFlow 2.3 for computation. Each algorithm was trained for 400 epochs with the Adam (Kingma, D., Ba, J., Adam, 2014) algorithm with a batch size equal to 200. The loss function was set to the mean squared error. Our implementation can be downloaded from https://github.com/mfrontczak/short-term-predictions-of-the-bitcoin. We tested the performance of each neural-network architecture as a function of the number of prior signal samples from which the network predicts subsequent Bitcoin exchange-rate values (*look back*). The second test parameter was the number of subsequent time samples to be predicted by the neural network (*look forward*). We tested the performance of the network for *look back* = 24, 72, 120 and 168 and *look forward* = 1, 2, 4, 6, 8 and 10. The training data was smoothed with the Savitzky-Golay filter, the results (output predictions) were evaluated on unfiltered data. For each configuration, we performed ten training runs and then averaged the RMSE results obtained on the test set. The results are presented in Table 1.

Table 1. RMSE scores are averaged from the results of 10 independent training runs of a given network +/- standard deviation. Each network architecture that obtained the best results for a given look forward value is marked in bold.

(*look back*, *look forward*)	MLP	CNN	LSTM	CNN B+Ef
(24, 1)	551.25±45.10	547.30±28.55	510.76±12.04	536.47±31.24
(72, 1)	617.62±146.81	520.16±29.45	516.39±24.10	596.34±118.65
(120, 1)	887.32±354.21	579.91±85.16	522.07±18.44	606.84±86.83
(168, 1)	821.92±234.86	634.02±174.06	565.76±44.84	718.87±209.77
(24, 2)	589.94±67.45	538.01±22.56	513.41±10.07	549.65±22.41
(72, 2)	594.49±97.19	578.96±82.54	523.84±23.75	557.04±78.93
(120, 2)	683.39±176.57	688.85±223.52	524.01±9.84	616.24±119.18
(168, 2)	688.80±103.27	647.00±159.60	540.41±15.67	663.01±139.10
(24, 4)	620.05±55.91	555.75±19.00	517.92±6.74	576.58±29.24
(72, 4)	657.26±108.72	567.42±46.13	524.68±20.09	595.55±40.70
(120, 4)	725.06±124.81	616.75±150.97	553.35±20.86	632.86±90.13
(168, 4)	679.27±38.60	621.33±114.19	572.54±46.12	726.61±254.36
(24, 6)	633.65±75.99	615.47±92.44	515.07±4.28	614.27±30.31
(72, 6)	696.74±242.78	574.58±48.34	519.70±13.99	669.36±184.21
(120, 6)	744.99±170.71	601.78±117.68	569.41±34.67	613.73±73.28
(168, 6)	713.65±132.83	641.95±102.76	612.04±51.46	770.77±150.30
(24, 8)	698.67±48.21	651.69±41.71	528.53±9.99	634.13±57.16
(72, 8)	682.97±50.65	644.83±90.40	569.90±51.38	626.52±43.46
(120, 8)	769.92±205.93	729.01±459.07	600.56±19.76	750.93±200.11
(168, 8)	863.69±167.72	633.92±79.52	629.77±29.10	698.84±87.49
(24, 10)	777.94±115.10	698.47±51.37	541.62±7.71	727.10±105.27
(72, 10)	798.69±104.89	692.10±85.41	608.07±80.84	754.44±89.17
(120, 10)	832.15±204.88	600.61±52.64	648.76±23.05	745.64±141.94
(168, 10)	902.95±205.09	649.95±92.24	679.68±66.05	1098.33±1005.41

Figure 3 shows the RMSE scores of the best performing algorithms for each *look forward* value. The network that obtained the best results for a given *look forward* value is indicated in Table 1 in bold font. More detailed evaluation based on metrics presented in the literature (Hyndman, Koehler, 2006) is presented in the Appendix in the detailed results of the evaluation of the methods. Fig. 4 presents example visualisations of prediction results for CNN, MLP, and LSTM models.

Among the networks that predict one sample forward, the best results were obtained for the MLP architecture with a *look back* value of 24 (551.25±45.10), for CNN with a *look back* value of 72 (520.16±29.45), for LSTM with a *look back* value of 24 (510.76±12.04) and for CNN B+E with a *look back* value of 24

https://doi.org/10.37705/TechTrans/e2022011

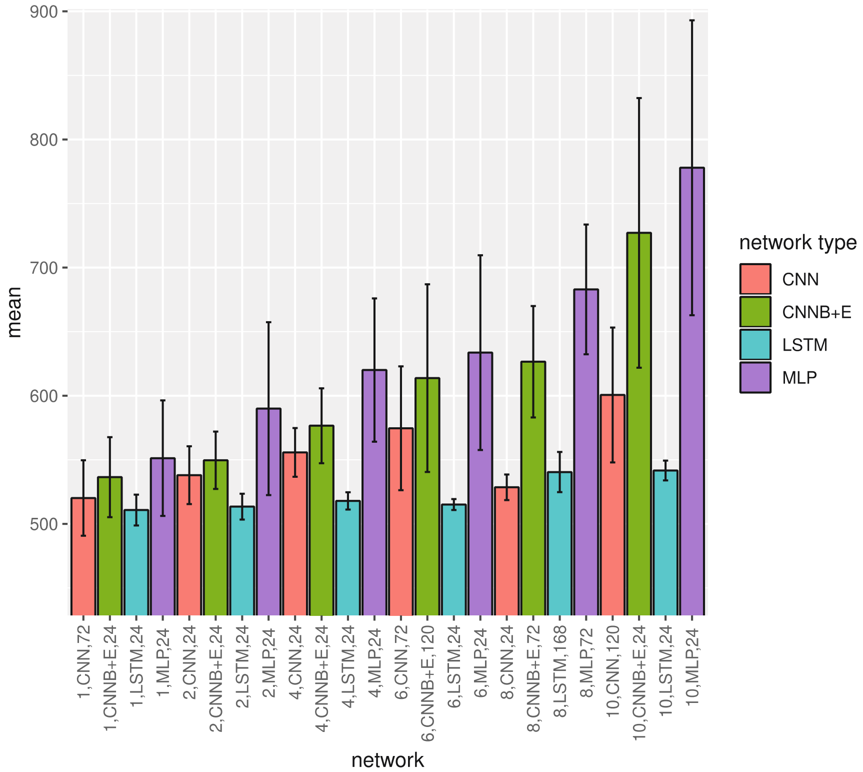


Fig. 3. RMSE bar plot of the best performing algorithms for each look forward value. The different colours indicate the architecture of the network. The signatures under the horizontal axis give the look forward value (first number), followed by the algorithm name and the look back value (last number). The RMSE value is averaged from the results of ten independent training runs of a given network. The standard deviation of the results is represented by black lines (own elaboration)

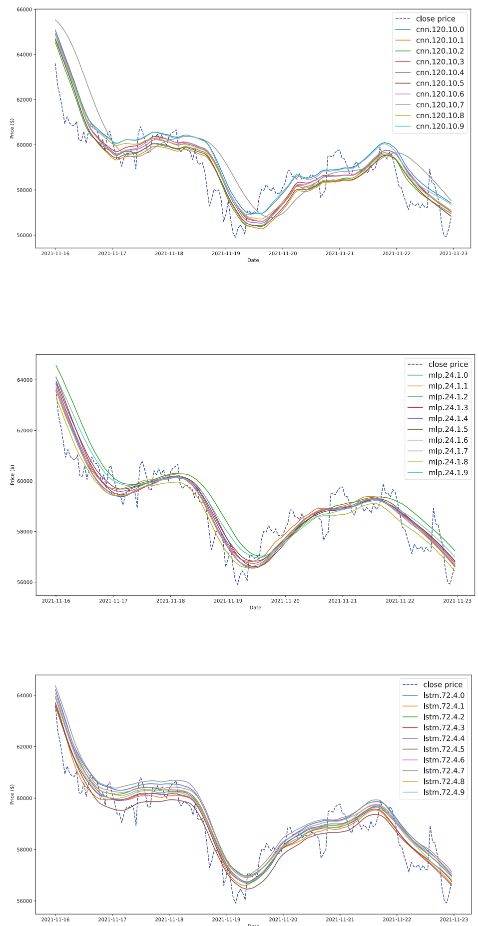


Fig. 4. Example visualisation of prediction results for CNN, MLP and LSTM models. Numbers in legends refer to look back, look forward and model ID. Ten of the same models were trained for evaluation purposes. The dashed line represents the actual close price (own elaboration)

(536.47±31.24). RMSE results are very similar and according to the Wilcoxon test results, do not have a statistically significant difference among *look forward groups*. As the number of samples to be predicted increases, the RMSE error also changes between the networks. However, the relative increase in error is not very large (see Table 1), for example, for the best MLP network results, the relative error difference between the number of samples to be predicted the difference between *look forward* 1 and *look forward* 2 is 7.02%, *look forward* 2 and *look forward* 4 is 5.10%, *look forward* 4 and *look forward* 6 is 2.19%, *look forward* 6 and *look forward* 8 is 7.78%, *look forward* 8 and *look forward* 10 is 13.91%. For CNN, the relative error difference between the number of predicted samples between *look forward* 1 and *look forward* 2 is 3.43%, *look forward* 2 and *look forward* 4 is 3.29%, *look forward* 4 and *look forward* 6 is 3.39%, *look forward* 6 and *look forward* 8 is 8.01%, *look forward* 8 and *look forward* 10 is 13.63%. For CNN B+E, the relative error difference between the number of predicted samples between *look forward* 1 and *look forward* 2 is 2.46%, *look forward* 2 and *look forward* 4 is 4.90%, *look forward* 4 and *look forward* 6 is 6.54%, *look forward* 6 and *look forward* 8 is 2.08%, *look forward* 8 and *look forward* 10 is 16.05%. The network has the smallest RMSE errors in the case of LSTM and the decrease in accuracy with increasing samples is smaller. The relative error difference between the number of predicted samples between *look forward* 1 and *look forward* 2 is 0.52%, *look forward* 2 and *look forward* 4 is 0.88%, *look forward* 4 and *look forward* 6 is 0.55%, *look forward* 6 and *look forward* 8 is 4.92%, *look forward* 8 and *look forward* 10 is 0.22%.

To show the current price and prediction information, we implemented a web application in Python 3.9.7. We used Flask micro-web framework 2.0.2. The website uses a single view with a chart and a drop-down list of the available models. We display forty-eight close price points from bitstamp.net public API. The API endpoint we created takes a model ID, reads all model parameters from the database, and runs prediction on data points at 1 hour (3600 seconds) intervals. Information from the models relating to path, look back and look forward are stored in the database. In addition, we scheduled the script to run hourly to save the first prediction of each available model. The models provided on the website are not further retrained with new data. The website is available at http://stpbtc-ii.up.krakow.pl.

4. Discussion

From the results presented in the previous section, we can see that there is not much difference between the RMSE error value of the CNN, MLP, and CNN B+E algorithms. The error values of these algorithms are very close for each of the considered number of samples during prediction. The best results were obtained for the LSTM network. This is most likely due to the fact that this model allows modelling of long-term, time-varying processes including trends while ignoring local variability. None of the models (besides LSTM and CNN, which were the best when *look forward* where 8 and 10) obtained dominant results, on the basis of which, one could say that it is better than the others. Taking into account forecasts of ten consecutive samples (ten hours ahead) the lowest error value of 541.62±7.71 was achieved for LSTM using twenty-four previous samples for its forecast. Thus, we recommend this model as the most effective for the largest time horizon considered. An interesting fact is the result showing that the use of additional data on the exchange rate of another cryptocurrency, Ethereum, did not improve the prediction accuracy. Results presented in the Appendix (detailed results of the evaluation methods in Table 2, 3, 4 and 5) support the results from Table 1. NRMSE for each model never exceeds 0.04 and MDAPE never exceeds 0.013. MPE values indicate that neither model tends to overestimate or underestimate the predicted values. The results assure us that the prediction results obtained by various training runs of the same algorithm do

https://doi.org/10.37705/TechTrans/e2022011

not differ much from the actual signal (close price) or between each other. This is also visually noticeable in Fig. 4.

Our website is, to the best of our knowledge, the first attempt to present a publicly open and free of charge real-time price prediction of cryptocurrency BITCOIN rate. For displaying the results of our work, we selected the best prediction models, which are running predictions without further retraining. The data presented on the website is not sufficient for making any investment decisions, and the accuracy of the prediction might get worse with each prediction without retraining on new data.

5. Conclusions

The prediction models based on neural networks proposed in this paper allow the making a short-term predictions of changes in the Bitcoin cryptocurrency exchange rate. For all models, the prediction error is at the level of 1% of the total value of this product. On the basis of this, we can conclude that this is an acceptable level of accuracy. The LSTM model, which is most effective for the purpose of predicting exchange rate changes over a period of ten hours, consists of two LSTM with dimensionality of output space equals 128 and ReLu activation function. The model consists of trainable weights and certainly cannot be called a complex solution in terms of its design. Thus, an elaborate mathematical model is not needed to predict short-term changes in the exchange rate of this cryptocurrency based only on past exchange rate values. Of course, longer term prediction is a much more complex issue, and based on the results we have obtained, we know that using only archival rates of a single product is not enough to make effective predictions over a broader time period. In our next study, we will seek to expand our methodology to include the use of a number of diverse market indicators that could be used to enable such long-term prediction. Additionally, we plan to use other, non-market pieces of information (e.g. trends from social networks) which have already tentatively shown their effectiveness (Mittal, Dhiman, Singh, Prakash, 2019; Huang et al. 2021; Wołk, 2020).

The website made for this paper presents real-time results of running the prediction models, which is limited to show only one model at a time. The user can see the prediction of the model to compare it with the real-time BTCUSD rate. The presented results are not sufficient to make any decision and might give false signals. The website is uncomplicated, using basic tools and python libraries to present the results leaves a lot of space for further improvements.

References

Dutta, A., Kumar, S., Basu, M. (2020). A Gated Recurrent Unit Approach to Bitcoin Price Prediction. *J. Risk Financial Manag.*, *13*, 23. https://doi.org/10.3390/jrfm13020023

Ferdiansyah, F., Othman, S.H., Zahilah Raja R., Md, Radzi, Stiawan, D., Sazaki, Y., Ependi, U. (2019). A LSTM-Method for Bitcoin Price Prediction: A Case Study Yahoo Finance Stock Market, International Conference on Electrical Engineering and Computer Science (ICECOS), 206–210, https://doi.org/10.1109/ICECOS47637.2019.8984499

Ho, K.-H., Chan, T.-T., Pan, H., Li, C. (2021). Do Candlestick Patterns Work in Cryptocurrency Trading?, IEEE International Conference on Big Data, 4566–4569, https://doi.org/10.1109/BigData52589.2021.9671826

Huang, Y., Capretz, L.F., Ho, D. (2021). Machine Learning for Stock Prediction Based on Fundamental Analysis. *IEEE Symposium Series on Computational Intelligence (SSCI)*, 01–10, https://doi.org/10.1109/SSCI50451.2021.9660134

Huang X. et al. (2021). LSTM Based Sentiment Analysis for Cryptocurrency Prediction. In: *Database Systems for Advanced Applications*, ed. C.S.

Jensen. Lecture Notes in Computer Science, vol. 12683. Springer, Cham. https://doi.org/10.1007/978-3-030-73200-4_

Hu, Z., Liu, W., Bian, J., Liu, X., Tie-Yan, L. (2018). Listening to Chaotic Whispers: A Deep Learning Framework for News-oriented Stock Trend Prediction. In: Proceedings of the Eleventh ACM International Conference on Web Search and Data Mining (WSDM‘18). New York: Association for Computing Machinery, 261–269. https://doi.org/10.1145/3159652.3159690

Hyndman, R.J., Koehler, A.B. (2006). Another look at measures of forecast accuracy, International Journal of Forecasting, Vol. 22, Issue 4, 679–688, https://doi.org/10.1016/j.ijforecast.2006.03.001

Inzirillo, H., Benjamin, M. (2021). Dimensionality reduction for prediction: Application to Bitcoin and Ethereum, arXiv: 2112.15036

Ji, S., Kim, J., Im, H.A. (2019). Comparative Study of Bitcoin Price Prediction Using Deep Learning. *Mathematics*, *7*, 898. https://doi.org/10.3390/math7100898

Kingma, D., Ba, J. (2014). *A Method for Stochastic Optimization.* International Conference on Learning Representations.

Li, T.R., Chamrajnagar, A.S., Fong, X.R., Rizik, N.R. Fu, F. (2019). Sentiment-Based Prediction of Alternative Cryptocurrency Price Fluctuations Using Gradient Boosting Tree Model. *Front. Phys*., 7: 98. https://doi.org/10.3389/fphy.2019.00098

Liu, F., Li, Y., Li, B., Li, J., Xie, H. (2021). Bitcoin transaction strategy construction based on deep reinforcement learning. *Applied Soft Computing*, 113, 107952. 10.1016/j.asoc.2021.107952.

Mangla, N., Bhat, A., Avabratha, G., Narayana Bhat (2019). Bitcoin Price Prediction Using Machine Learning, *International Journal of Information and Computing Science*.

Matic, J.L., Packham, N., Härdle, W.K. (2021). Hedging Cryptocurrency Options. Retrieved from: https://ssrn.com/abstract=3968594 or http://dx.doi.org/10.2139/ssrn.3968594 [online: 1.03.2022].

Mezquita Y., Gil-González A.B., Prieto J., Corchado J.M. (2022). Cryptocurrencies and Price Prediction: A Survey. In*: Blockchain and Applications*, eds. J. Prieto, A. Partida, P. Leitão, A. Pinto. Lecture Notes in Networks and Systems, Vol. 320. Springer, Cham. https://doi.org/10.1007/978-3-030-86162-9

Mittal, A., Dhiman, V., Singh, A., Prakash C. (2019). Short-Term Bitcoin Price Fluctuation Prediction Using Social Media and Web Search Data, 12th International Conference on Contemporary Computing (IC3), 1–6, https://doi.org/10.1109/IC3.2019.8844899

Nakamoto, S. (2009). Bitcoin: A Peer-to-Peer Electronic Cash System, Cryptography Mailing. Retrieved from: https://metzdowd.com [online: 1.03.2022].

Nazário, R.T.F., Lima e Silva, J., Sobreiro, V.A., Kimura, H. (2017). A literature review of technical analysis on stock markets. *The Quarterly Review of Economics and Finance*, Vol. 66: 115–126. https://doi.org/10.1016/j.qref.2017.01.014.

Number of cryptocurrencies worldwide from 2013 to February 2022 (2022). Retrieved from: https://www.statista.com/statistics/863917/number-crypto-coins-tokens [online: 1.03.2022].

Patel, J., Kalariya, V., Parmar, P., Tanwar, S., Kumar, N., Alazab, M. (2020). Stochastic Neural Networks For Cryptocurrency Price Prediction. IEEE Access. https://doi.org/10.1109/ACCESS.2020.2990659

Phaladisailoed, T., Numnonda, T. (2018). Machine Learning Models Comparison for Bitcoin Price Prediction, 10th International Conference on Information Technology and Electrical Engineering (ICITEE), 506–511, https://doi.org/10.1109/ICITEED.2018.8534911

Rane, P.V., Dhage, S.N. (2019). Systematic Erudition of Bitcoin Price Prediction using Machine Learning Techniques, 5th International Conference on

Advanced Computing & Communication Systems (ICACCS), 594–598, https://doi.org/10.1109/ICACCS.2019.8728424
Rizwan, M., Narejo, S. Javed, M. (2019). Bitcoin price prediction using Deep Learning Algorithm, 13th International Conference on Mathematics, Actuarial Science, Computer Science and Statistics (MACS), 1–7, https://doi.org/10.1109/MACS48846.2019.9024772
Sen, J., Dutta, A., Mehtab, S., (2021). Stock Portfolio Optimization Using a Deep Learning LSTM Model, arXiv.org:2111.04709.
Struga, K., Olti Q. (2018). Bitcoin Price Prediction with Neural Networks, RTA-CSIT.
Taylor, S.J., Letham, B. (2017). Forecasting at scale. *Peer J. Preprints* 5:e3190v2, https://doi.org/10.7287/peerj.preprints.3190v2
Top Cryptocurrency Spot Exchanges. Retrieved from: https://coinmarketcap.com/rankings/exchanges [online: 1.03.2022]. All Cryptocurrencies. Retrieved from: https://coinmarketcap.com/all/views/all [online: 1.03.2022].
Tsai, Y., Chen, J., Wang, J. (2020). Predict Forex Trend via Convolutional Neural Networks. *Journal of Intelligent Systems*, 29, 941– 958.
Wołk, K. (2020). Advanced social media sentiment analysis for short-term cryptocurrency price prediction. *Expert Systems*. 37:e12493. https://doi.org/10.1111/exsy.12493
Zhao, D., Rinaldo, A., Brookins, C. (2019). Cryptocurrency Price Prediction and Trading Strategies Using Support Vector Machines, arXiv:1911.11819.

Appendix – detailed results of the evaluation of the methods

We use the following formulas to calculate errors:

$$error = actual - predicted \tag{1}$$

where:
actual is *n*-element vector of actual (real) values;
predicted is *n*-element vector of predicted (forecasted) values.

Mean squared error:

$$MSE = \frac{\sum_{i=1}^{n} error_i^2}{n} \tag{2}$$

Root mean squared error:

$$RMSE = \sqrt{MSE} \tag{3}$$

Normalized root mean squared error:

$$NRMSE = \frac{RMSE}{max(actual) - min(actual)} \tag{4}$$

Percentage error:

$$perror = \frac{error}{actual + e} \tag{5}$$

where:
$actual + \in \neq 0$ and division is an element-wise division.

Mean percentage error:

$$MPE = \frac{\sum_{i=1}^{n} perror_i}{n} \tag{6}$$

Median absolute percentage error:

$$MDAPE = med(|perror|)$$

where: med is median and |⬚| *is an absolute value*

Table 2. Detailed evaluation of errors of MLP network. Scores are averaged from the results of 10 independent training runs of a given network +/- standard deviation

(*look back*, *look forward*)	MSE	RMSE	NRMSE	MPE	MDAPE
(24, 1)	305712±52103	551.25±45.10	0.0196±0.0016	0.0008±0.0016	0.0066±0.0004
(72, 1)	400845±209146	617.62±146.81	0.0220±0.0052	-0.0011±0.0073	0.0081±0.0028
(120, 1)	900250±699358	887.32±354.21	0.0315±0.0126	-0.0013±0.0153	0.0134±0.0072
(168, 1)	725194±420238	821.92±234.86	0.0292±0.0083	-0.0003±0.0124	0.0117±0.0050
(24, 2)	352124±86559	589.94±67.45	0.0210±0.0024	0.0006±0.0028	0.0070±0.0005
(72, 2)	361916±126179	594.49±97.19	0.0211±0.0035	0.0005±0.0061	0.0076±0.0019
(120, 2)	495074±298545	683.39±176.57	0.0243±0.0063	-0.0014±0.0076	0.0088±0.0029
(168, 2)	484040±147116	688.80±103.27	0.0245±0.0037	-0.0027±0.0056	0.0085±0.0017
(24, 4)	387276±71769	620.05±55.91	0.0220±0.0020	0.0009±0.0021	0.0072±0.0005
(72, 4)	442627±152065	657.26±108.72	0.0234±0.0039	-0.0009±0.0071	0.0084±0.0021
(120, 4)	539730±182708	725.06±124.81	0.0258±0.0044	-0.0039±0.0073	0.0093±0.0022
(168, 4)	462745±53622	679.27±38.60	0.0241±0.0014	0.0018±0.0053	0.0082±0.0008
(24, 6)	406704±98270	633.65±75.99	0.0225±0.0027	0.0011±0.0019	0.0073±0.0006
(72, 6)	538491±438109	696.74±242.78	0.0248±0.0086	0.0008±0.0061	0.0087±0.0037
(120, 6)	581242±284185	744.99±170.71	0.0265±0.0061	0.0025±0.0080	0.0095±0.0033
(168, 6)	525177±228721	713.65±132.83	0.0254±0.0047	0.0030±0.0058	0.0087±0.0028
(24, 8)	490230±65514	698.67±48.21	0.0248±0.0017	0.0013±0.0022	0.0079±0.0005
(72, 8)	468760±69291	682.97±50.65	0.0243±0.0018	0.0019±0.0057	0.0082±0.0009
(120, 8)	630951±386492	769.92±205.93	0.0274±0.0073	-0.0018±0.0089	0.0098±0.0041
(168, 8)	771274±320927	863.69±167.72	0.0307±0.0060	0.0002±0.0105	0.0111±0.0034
(24, 10)	617108±196744	777.94±115.10	0.0277±0.0041	0.0012±0.0027	0.0087±0.0011
(72, 10)	647814±173298	798.69±104.89	0.0284±0.0037	0.0021±0.0087	0.0100±0.0023
(120, 10)	730260±396837	832.15±204.88	0.0296±0.0073	0.0018±0.0096	0.0105±0.0042
(168, 10)	853180±441098	902.95±205.09	0.0321±0.0073	-0.0051±0.0093	0.0112±0.0039

Table 3. Detailed evaluation of errors of CNN network. Scores are averaged from the results of 10 independent training runsof a given network +/- standard deviation

(*look back*, *look forward*)	MSE	RMSE	NRMSE	MPE	MDAPE
(24, 1)	300273±31432	547.30±28.55	0.0195±0.0010	-0.0029±0.0020	0.0067±0.0004
(72, 1)	271352±31695	520.16±29.45	0.0185±0.0010	-0.0007±0.0028	0.0064±0.0004
(120, 1)	342825±113458	579.91±85.16	0.0206±0.0030	0.0007±0.0048	0.0072±0.0015
(168, 1)	429250±270929	634.02±174.06	0.0225±0.0062	-0.0007±0.0069	0.0081±0.0034
(24, 2)	289915±24475	538.01±22.56	0.0191±0.0008	-0.0001±0.0025	0.0066±0.0003
(72, 2)	341329±102376	578.96±82.54	0.0206±0.0029	0.0024±0.0051	0.0074±0.0015
(120, 2)	519474±372801	688.85±223.52	0.0245±0.0079	-0.0010±0.0090	0.0094±0.0043
(168, 2)	441532±233932	647.00±159.60	0.0230±0.0057	-0.0020±0.0056	0.0081±0.0024
(24, 4)	309178±20967	555.75±19.00	0.0198±0.0007	0.0007±0.0022	0.0067±0.0002
(72, 4)	323882±53229	567.42±46.13	0.0202±0.0016	0.0005±0.0039	0.0069±0.0007
(120, 4)	400900±235791	616.75±150.97	0.0219±0.0054	0.0015±0.0035	0.0077±0.0023

https://doi.org/10.37705/TechTrans/e2022011

(*look back*, *look forward*)	MSE	RMSE	NRMSE	MPE	MDAPE
(168, 4)	397792±157915	621.33±114.19	0.0221±0.0041	0.0015±0.0052	0.0076±0.0018
(24, 6)	386496±124926	615.47±92.44	0.0219±0.0033	0.0024±0.0042	0.0076±0.0016
(72, 6)	332249±57252	574.58±48.34	0.0204±0.0017	-0.0004±0.0040	0.0070±0.0008
(120, 6)	374605±166694	601.78±117.68	0.0214±0.0042	0.0011±0.0049	0.0075±0.0020
(168, 6)	421598±142761	641.95±102.76	0.0228±0.0037	0.0008±0.0041	0.0076±0.0014
(24, 8)	426260±56092	651.69±41.71	0.0232±0.0015	0.0019±0.0035	0.0077±0.0007
(72, 8)	423166±122213	644.83±90.40	0.0229±0.0032	0.0034±0.0055	0.0081±0.0017
(120, 8)	721123±1190808	729.01±459.07	0.0259±0.0163	-0.0007±0.0025	0.0093±0.0076
(168, 8)	407548±103783	633.92±79.52	0.0225±0.0028	0.0003±0.0062	0.0079±0.0015
(24, 10)	490230±71577	698.47±51.37	0.0248±0.0018	-0.0022±0.0040	0.0082±0.0008
(72, 10)	485571±119462	692.10±85.41	0.0246±0.0030	-0.0013±0.0064	0.0083±0.0014
(120, 10)	363221±67084	600.61±52.64	0.0213±0.0019	0.0004±0.0033	0.0071±0.0005
(168, 10)	430090±123267	649.95±92.24	0.0231±0.0033	-0.0029±0.0046	0.0078±0.0013

Table 4. Detailed evaluation of errors of LSTM network. Scores are averaged from the results of 10 independent training runs of a given network +/- standard deviation

(*look back*, *look forward*)	MSE	RMSE	NRMSE	MPE	MDAPE
(24, 1)	261004±12655	510.76±12.04	510.7574±0.0004	-0.0004±0.0011	0.0064±0.0001
(72, 1)	267184±25692	516.39±24.10	516.3925±0.0009	-0.0012±0.0017	0.0063±0.0003
(120, 1)	272864±19698	522.07±18.44	522.0710±0.0007	-0.0003±0.0015	0.0062±0.0002
(168, 1)	321890±54214	565.76±44.84	565.7568±0.0016	-0.0016±0.0031	0.0066±0.0006
(24, 2)	263681±10485	513.41±10.07	513.4103±0.0004	-0.0002±0.0014	0.0064±0.0001
(72, 2)	274913±25231	523.84±23.75	523.8373±0.0008	-0.0001±0.0023	0.0063±0.0002
(120, 2)	274677±10450	524.01±9.84	524.0136±0.0003	-0.0011±0.0011	0.0062±0.0002
(168, 2)	292264±16952	540.41±15.67	540.4098±0.0006	-0.0008±0.0018	0.0064±0.0002
(24, 4)	268281±6989	517.92±6.74	517.9186±0.0002	0.0002±0.0016	0.0064±0.0001
(72, 4)	275650±21397	524.68±20.09	524.6781±0.0007	-0.0009±0.0021	0.0063±0.0003
(120, 4)	306586±23387	553.35±20.86	553.3482±0.0007	-0.0006±0.0021	0.0065±0.0002
(168, 4)	329719±56126	572.54±46.12	572.5425±0.0016	-0.0006±0.0027	0.0068±0.0006
(24, 6)	265317±4420	515.07±4.28	515.0729±0.0002	0.0000±0.0009	0.0064±0.0001
(72, 6)	270267±14841	519.70±13.99	519.7026±0.0005	-0.0008±0.0011	0.0062±0.0002
(120, 6)	325310±41643	569.41±34.67	569.4100±0.0012	-0.0017±0.0022	0.0066±0.0005
(168, 6)	376974±65820	612.04±51.46	612.0384±0.0018	-0.0021±0.0029	0.0071±0.0006
(24, 8)	279435±10581	528.53±9.99	528.5308±0.0004	-0.0001±0.0014	0.0065±0.0001
(72, 8)	327166±62982	569.90±51.38	569.9030±0.0018	-0.0001±0.0033	0.0067±0.0006
(120, 8)	361018±24055	600.56±19.76	600.5557±0.0007	-0.0015±0.0023	0.0069±0.0002
(168, 8)	397371±36840	629.77±29.10	629.7686±0.0010	0.0006±0.0016	0.0072±0.0002
(24, 10)	293401±8370	541.62±7.71	541.6158±0.0003	-0.0001±0.0012	0.0066±0.0001
(72, 10)	375629±110267	608.07±80.84	608.0683±0.0029	0.0006±0.0037	0.0070±0.0011
(120, 10)	421363±30681	648.76±23.05	648.7567±0.0008	-0.0017±0.0023	0.0072±0.0002
(168, 10)	465898±94727	679.68±66.05	679.6850±0.0023	-0.0012±0.0036	0.0077±0.0007

Table 5. Detailed evaluation of errors of CNN B+E network. Scores are averaged from the results of 10 independent training runs of a given network +/- standard deviation

(*look back*, *look forward*)	MSE	RMSE	NRMSE	MPE	MDAPE
(24, 1)	288682±34595	536.47±31.24	0.0191±0.0011	0.0009±0.0026	0.0067±0.0005
(72, 1)	368290±172002	596.34±118.65	0.0212±0.0042	0.0038±0.0049	0.0078±0.0025
(120, 1)	375045±114572	606.84±86.83	0.0216±0.0031	0.0014±0.0060	0.0077±0.0016
(168, 1)	556373±332143	718.87±209.77	0.0255±0.0075	0.0043±0.0093	0.0101±0.0042
(24, 2)	302570±24644	549.65±22.41	0.0195±0.0008	0.0019±0.0021	0.0068±0.0003
(72, 2)	315905±101464	557.04±78.93	0.0198±0.0028	0.0008±0.0043	0.0069±0.0014
(120, 2)	392531±168607	616.24±119.18	0.0219±0.0042	0.0008±0.0051	0.0076±0.0018
(168, 2)	456993±199012	663.01±139.10	0.0236±0.0049	0.0037±0.0039	0.0086±0.0022
(24, 4)	333209±33778	576.58±29.24	0.0205±0.0010	0.0020±0.0025	0.0070±0.0004
(72, 4)	356167±49705	595.55±40.70	0.0212±0.0014	0.0024±0.0028	0.0072±0.0007
(120, 4)	407826±116079	632.86±90.13	0.0225±0.0032	-0.0006±0.0042	0.0076±0.0013
(168, 4)	586192±493713	726.61±254.36	0.0258±0.0090	0.0057±0.0058	0.0092±0.0043
(24, 6)	378153±38064	614.27±30.31	0.0218±0.0011	0.0020±0.0025	0.0074±0.0005
(72, 6)	478582±321704	669.36±184.21	0.0238±0.0065	-0.0031±0.0066	0.0084±0.0033
(120, 6)	381493±91941	613.73±73.28	0.0218±0.0026	0.0001±0.0033	0.0073±0.0008
(168, 6)	614422±264700	770.77±150.30	0.0274±0.0053	0.0036±0.0068	0.0096±0.0029
(24, 8)	405061±75586	634.13±57.16	0.0225±0.0020	0.0021±0.0026	0.0075±0.0008
(72, 8)	394225±55603	626.52±43.46	0.0223±0.0015	-0.0025±0.0028	0.0073±0.0006
(120, 8)	599929±381244	750.93±200.11	0.0267±0.0071	0.0039±0.0063	0.0093±0.0039
(168, 8)	495264±122094	698.84±87.49	0.0248±0.0031	-0.0002±0.0035	0.0079±0.0009
(24, 10)	538652±162324	727.10±105.27	0.0258±0.0037	0.0016±0.0060	0.0087±0.0018
(72, 10)	576341±139547	754.44±89.17	0.0268±0.0032	0.0003±0.0058	0.0089±0.0015
(120, 10)	574109±225709	745.64±141.94	0.0265±0.0050	0.0024±0.0040	0.0088±0.0022
(168, 10)	2116081±4683880	1098.33±1005.41	0.0390±0.0357	-0.0002±0.0156	0.0132±0.0120

https://doi.org/10.37705/TechTrans/e2022011

Ile neuronów jest potrzebnych, aby dokonać krótkoterminowej predykcji kursu Bitcoina?

Streszczenie

Celem naszej pracy było stworzenie architektury sieci neuronowej, która przy wykorzystaniu danych historycznych pozwalałaby na dokładną predykcję kursu Bitcoin. Nasza praca wpisuje się w bardzo ważny temat przewidywania wartości kursu kryptowaluty. Niemniej istotny jest fakt, że w naszej pracy wykorzystujemy najnowsze dane, które z powodu dużej dynamiki kursu Bitcoin w ostatnim roku znacznie różnią się od danych z lat wcześniejszych. Proponujemy i testujemy kilka architektur opartych na sieciach neuronowych oraz przeprowadzamy dyskusję wyników. W odróżnieniu od poprzednich prac, przeprowadzamy wszechstronne porównanie trzech różnych modeli opartych na sieciach neuronowych: MLP (multilayer perceptron), LSTM (long short-term memory) i CNN (convolutional neural network). Przetestowaliśmy je dla szerokiego zakresu parametrów. Przedstawione przez nas wyniki są, według naszej wiedzy, najbardziej aktualnymi, jeśli chodzi o zastosowanie metod sztucznej inteligencji do przewidywania kursów kryptowalut. Najlepiej działająca architektura została wykorzystana na stronie internetowej, która w czasie rzeczywistym prognozuje kurs Bitcoina. Strona ta jest dostępna pod adresem http://stpbtc-ii.up.krakow.pl/. Kody źródłowe naszych badań są dostępne do pobrania w celu umożliwienia odtworzenia naszego eksperymentu

Słowa kluczowe: Sieć neuronowa, Wielowarstwowy Perceptron, Konwolucyjna Sieć Neuronowa, Pamięć długo-krótkotrwała, Bitcoin