OPEN ACCESS

ISSN 2460-9056
http://socj.telkomuniversity.ac.id/ijoict/

Intl. Journal on ICT
Vol. 6, No. 2, Dec 2020. pp. 94-105
doi: 10.21108/IJOICT.2020.62.538

Foreign Exchange Rate Prediction Using Long-Short Term Memory:

A Case Study in COVID-19 Pandemic

Hasna Haifa Zahrah [1], Siti Sa'adah [1], Rita Rismala [1]

[1]*School of Computing, Telkom University*
Jl. Telekomunikasi, Bandung, Indonesia

[*] hasnahaifaz@student.telkomuniversity.ac.id

Abstract

The foreign exchange market is a global financial market that is influenced by economic, political, and psychological factors that are interconnected in complex ways. This complexity makes the foreign exchange market a difficult time-series prediction. At the end of 2019, the world was faced with the COVID-19 pandemic that has not only affected public health but also the foreign exchange market, which makes the trading behaviour affected. Long Short-Term Memory network (LSTM) is a type of recurrent neural network (RNN) that can solve long-term dependencies and is suitable to be a financial time-series model. This study implemented the LSTM model to predict the foreign exchange rate at a timeframe of 1 hour and daily in 2020 to get the best hyperparameter based on the RMSE evaluation results. Furthermore, with the obtained hyperparameters, the prediction result of 2020 was then compared with the 2018 and 2019 prediction results. The best RMSE result was obtained in 1-hour timeframe and when 2020's RMSE result was compared to 2018's and 2019's RMSE result, the prediction of 2019 gave the best RMSE result. The LSTM model is able to achieve good results in the 2020 prediction, proven by the RMSE result which is 0.00135.

Keywords: COVID-19, Foreign exchange, Hyperparameter, Long-Short Term Memory, Prediction

I. Introduction

FOREIGN exchange (Forex) market is a global financial market that is by far more complicated as compared to the stock or bonds market[1]. The foreign exchange rate is very volatile because it is influenced by the time and psychology of market participants which is based on economic and political factors. These influences make foreign exchange rate a difficult prediction of time-series data. The smaller the foreign exchange market timeframe is, the more volatile it will be. Even though the foreign exchange market has chaotic, noisy, and non-linear data, there are data patterns or chart patterns that appear repeatedly. Hence, can be used as a basis for market-rate predictions. There are many currency pairs in the foreign exchange market, one of them is EUR/USD. The pair is one of the most traded currency pairs in the foreign exchange market.

At the end of 2019, the world was faced with one of the worst experiences in history. The COVID-19 pandemic has not only affected public health but also causing a direct global destructive economic impact that is present in every area of the globe including financial markets[2]. Naturally, financial markets reacted to the panic associated with the pandemic – in this regard, the foreign exchange market was no different[3]. Thus,

Received on 19 Dec 2020. Revised on 4 Jan 2021. Accepted and Published on 5 Jan 2021.

foreign exchange trading behaviours were affected. Aslam et al. observed that the efficiency levels of EUR decreased and its persistence pattern shifted from anti-persistent to persistent behavior during the COVID-19 period[4]. The observation used exchange rates of EUR currency against US dollar, similar to what is performed in this study.

Neural network methods are a prime candidate for foreign exchange rate prediction because of their ability to handle non-linear, noisy, and complex data[5]. Recurrent neural networks (RNNs) are a powerful model for processing sequential data such as time-series data. Long Short-Term Memory network (LSTM) is a type of Recurrent Neural Network (RNN) that capable of learning long-term dependencies. The model introduces memory cells that are able to effectively associate memories and input remote in time, hence suit to grasp the structure of data dynamically over time with high prediction capacity[6].

In the work of Zhelev and Avresky, they implemented LSTM to predict foreign exchange rate and got 0.0165 as their biggest prediction error although, they didn't tune the hyperparameters[7]. Shiao et al. compared the LSTM model with the SVR on the foreign exchange market prediction and got better results with the LSTM model than the SVR although, if observed closely, the trend of predicted values was just mimicking the last input[8]. However, it has been observed that LSTM can be a financial time-series prediction model even if it is necessary to examine again what hyperparameters are for LSTM to obtain optimal results. Apart from it, the COVID-19 pandemic also has induced chaos and turbulence in the financial market. This study focuses on applying the LSTM model to a daily and 1-hour timeframe market to predict the EUR/USD exchange rate and analyze the impact of the COVID-19 pandemic on the model's performance by comparing it with 2 previous years. Therefore, it can help traders or investors to make a prediction model of the foreign exchange rate during the COVID-19 pandemic hence they can make the right decisions in this difficult time.

II. LITERATURE REVIEW

Several papers in the literature proposed LSTM methodologies for prediction in the financial market including the foreign exchange market. A study [7] that discusses the implementation of LSTM for predicting EUR/USD rate in foreign exchange market with M15 granularity, concluded that the prediction results always inside min and max price of the actual price for the period. However, this study didn't tune the hyperparameters of LSTM despite the high dependency of neural networks on hyperparameter. The algorithm results may fluctuate dramatically under the different configurations of hyper-parameters[9]. A study [8] that compared LSTM and SVR method in predicting foreign exchange rates with different timesteps, concluded that the LSTM model can predict the trend of future data better than the SVR model. However, this study only used one type of timeframe data thus, the model's capabilities for other types of timeframe data couldn't be seen. A study [6] that demonstrated the LSTM model in sequence learning for stock market prediction in China, revealed that normalization was very useful for improving accuracy.

III. RESEARCH METHOD

In this research, LSTM is used as the method for predicting foreign exchange rate with EUR/USD as the currency pair. The optimization method is also implemented in this research using Adam optimization. The processes of system design for this research as shown in Fig. 2 are the data preparation process where the input is EUR/USD data, the training process where the input is train data from the data preparation process, the testing process where the process is using a trained model from the training process and the input is test data from the data preparation process, and the evaluating process to evaluate the prediction result from the testing process.

A. Long-Short Term Memory (LSTM)

Long Short-Term Memory (LSTM) was first proposed in 1997 by Sepp Hochreiter and Jürgen Schmidhuber [10]. It was proposed to improve the performance of RNN that has difficulties in terms of long-term dependencies because there is a Vanishing Gradient in which, the weights of previous output will decrease over time-steps[11]. LSTM uses cell state and gate units to solve the vanishing gradient problem as shown in Fig. 1. Gate units consist of an input gate, forget gate, and output gate, that will carefully remove or add information to the cell state.

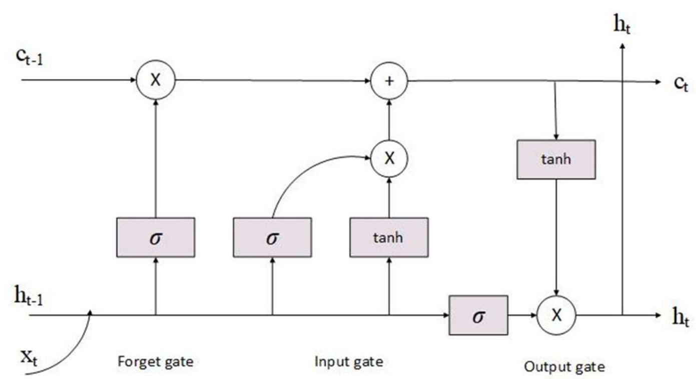

Fig. 1. LSTM Unit Structures

Equation 1 is a formula of forget gate that is used to keep important information and forget information that is not useful in the cell state. Equation 2 is a formula of input gate where it put in new information from the current input value. Equation 4 is a formula of output gate where it decides what output is. The formula of hidden state and cell state is shown in equation 5 and equation 6 where $*$ shows the multiplication of elements. Meanwhile, equation 3 is a formula of hidden state "candidate" which is calculated based on the current input and the previous hidden state. Equation 7 and equation 8 are functions of sigmoid and hyperbolic activation as also shown in Fig. 1. In equation 1,2,3,4,5,6 t is time. x is the input to the LSTM unit in equation 1,2,3,4,7,8. W and b are weights and bias vectors in equation 1,2,3,4 [12].

$$f_t = \sigma(W_{xf}x_t + W_{hf}h_{t-1} + b_f) \tag{1}$$

$$i_t = \sigma(W_{xi}x_t + W_{hi}h_{t-1} + b_i) \tag{2}$$

$$g_t = \tanh(W_{xg}x_t + W_{hg}h_{t-1} + b_g) \tag{3}$$

$$o_t = \sigma(W_{xo}x_t + W_{ho}h_{t-1} + b_o) \tag{4}$$

$$c_t = g_t * i_t + c_{t-1} + f_t \tag{5}$$

$$h_t = o_t * \tanh(c_t) \tag{6}$$

$$\sigma(x) = \frac{1}{1 + e^{-x}} \tag{7}$$

$$\tanh(x) = 2\sigma(2x) - 1 \tag{8}$$

B. Adaptive Moment Estimation (Adam)

Adam was first proposed in 2015 by Diederik P. Kingma and Jimmy Lei Ba[13]. The method computes individual adaptive learning rates for different parameters from estimates of first and second moments of the gradients; the name *Adam* is derived from adaptive moment estimation[13]. In addition to storing an exponentially decaying average of past squared gradients v_t like Adadelta and RMSprop which the function is shown in equation 9, Adam also keeps an exponentially decaying average of past gradients m_t, similar to momentum which the function is shown in equation 10.

$$m_t = \beta_1 m_{t-1} + (1 - \beta_1) g_t \quad (9)$$

$$v_t = \beta_2 m_{t-1} + (1 - \beta_2) g_t^2 \quad (10)$$

In equation 9 and equation 10 t is iteration; g is gradient; β_1 and β_2 are first and second-moment decay rate; m_t and v_t are estimates of the first moment (the mean) and the second-moment (the uncentered variance) of the gradients respectively, hence the name of the method. Adam update rule for parameter update shown in equation 11.

$$\theta_{t-1} = \theta_t - \frac{\eta}{\sqrt{\hat{v}_t} + \epsilon} \hat{m}_t \quad (11)$$

In equation 11 η is learning rate and ϵ is epsilon. Default values and typical choice for parameters are 0.9 for β_1, 0.999 for β_2, $10^{-8}$for ϵ ,and 0.001 for η [14]. Adam combines the advantages of RMSprop and AdaGrad hence, suitable for dealing with problems with large amounts of data and non-stationary objectives.

C. System Design

Fig. 2 shows the proposed system design for foreign exchange rate prediction using LSTM.

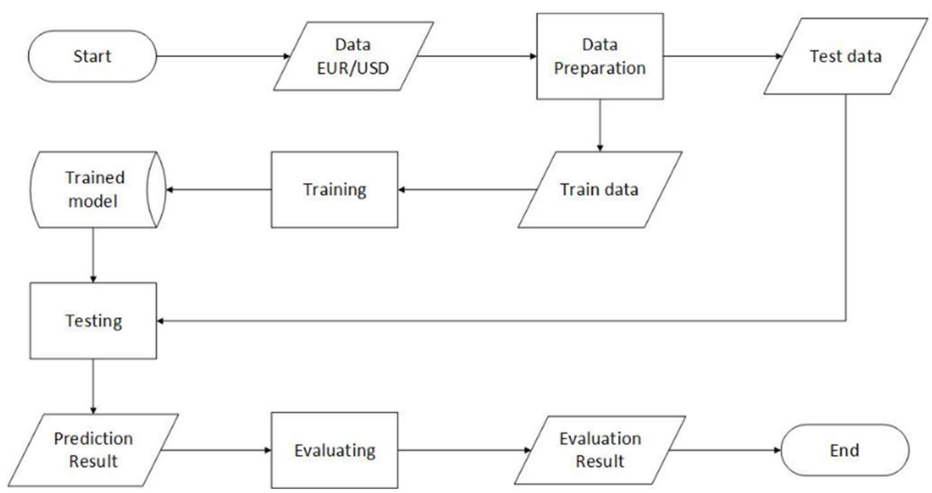

Fig. 2. System Design of Forex Rate Prediction Using LSTM

1) Data Preparation

As shown in Fig.2 EUR/USD data are used as input data to this system. The EUR/USD historical data from 4 May 2010 to 19 October 2020 were taken from MetaQuotes MT4 with 2718 records of daily timeframe data and 64897 records of 1-hour timeframe data. The obtained historical data are in the form of date, time, open, high, low, and close prices as shown in Table I. In this research, the prediction result is based on the close price of EUR/USD.

TABLE I
EXAMPLE OF FOREIGN EXCHANGE DATA

date	time	open	high	low	close	volume
2010.05.04	00:00	1.31884	1.31981	1.31853	1.31980	750
2010.05.04	01:00	1.31982	1.32004	1.31885	1.31942	626
2010.05.04	02:00	1.31947	1.32120	1.31915	1.32083	954
2010.05.04	03:00	1.32084	1.32131	1.31970	1.32073	896

According to [6] normalization improve the prediction accuracy, therefore; close prices are converted into values in the range 0 to 1. This normalization aims to avoid data becoming less influential than it should be. Min-Max Scaling is used as a normalization technique described in equation (12).

$$\hat{x} = \frac{x - \min(x)}{\max(x) - \min(x)} \tag{12}$$

where $\hat{x}$ is normalized data, x is the data that you want to be normalized, $min(x)$ and $max(x)$ are minimum and maximum values of all data.

After normalization is done, data are split into a training and testing set. 2020 data are used as a testing set because we want to analyze the COVID-19 impact and the rest of the data is used as a training set.

2) LSTM Model Training

The training process in Fig. 2 is described in Fig.3 where the train data used as input data are split into 67% train data and 33% validation data. The hyperparameter is then selected to become the hyperparameter set for the training process using LSTM. The hyperparameter candidates are shown in Table II.

TABLE II
HYPERPARAMETER CANDIDATES FOR LSTM MODEL

Hyperparameter	Value
Number of Hidden Layers	[1, 2, 3]
Number of Neurons	[1, 5, 10]
Dropout	[No dropout, 0.1 rate]

The training using LSTM process in Fig.3 is executed by fitting the input value of train data with the label which is the value of one timestep afterward. Optimization also implemented in this process using Adam optimization with default initial values of Adam's parameters. This optimization computes individual adaptive learning rates for different parameters from estimates of first and second-moments of the gradients. Mean squared error is used as a loss function in this training process which is useful for model validation and determining whether the model is overfitting or not.

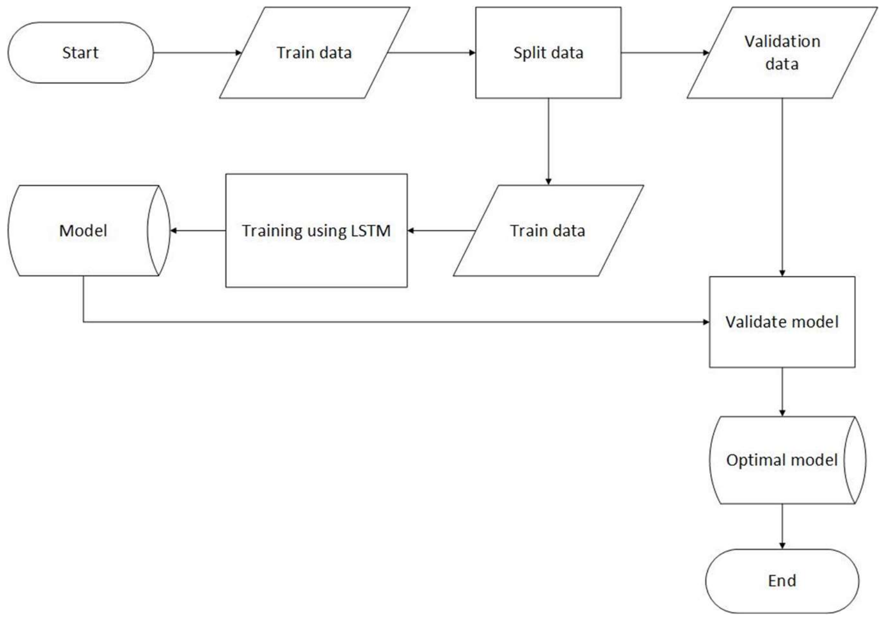

Fig. 3. LSTM Model Training

The output of the training process is a model as shown in Fig.3 and then it is validated using validation data. The model validation process is implemented using the callbacks function, namely the early stop, where if the loss of train and validation data remains the same up to 30 times, the epoch will stop. The maximum number of epochs is 1000.

3) *LSTM Model Testing and Evaluation*

The testing and evaluating process in Fig.2 are described in Fig.4 where test data are used as input data for testing using LSTM process. The process used a trained model that was obtained from the previous training process as shown in Fig.2. The prediction result of the testing process is then evaluated by Root Mean Square Error (RMSE) function which is to calculate the performance model by measuring the root of the average squared difference between the predicted data and the actual data. Therefore, if the RMSE value is getting closer to 0, the accuracy will be higher. The following RMSE equation is described in equation 13.

$$\mathrm{RMSE} = \sqrt{\frac{1}{n}\sum_{i-1}^{n}\left(Y_i - \hat{Y}_i\right)^2} \tag{13}$$

where n, Y, $\hat{Y}$ are the amount of data, the predicted data, and the actual data, respectively.

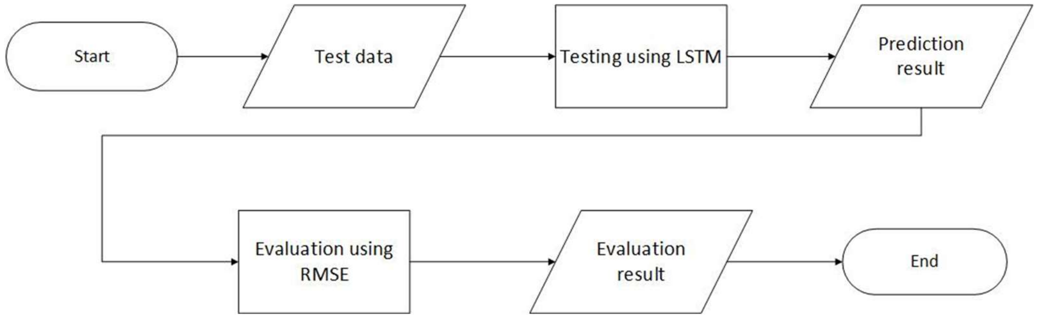

Fig. 4. LSTM Model Testing and Evaluation

IV. RESULTS AND DISCUSSION

A. *Best Hyperparameters*

Testing was done to get the best hyperparameters for the LSTM model in foreign exchange rate prediction based on the RMSE results from test data which are 2020 close prices from 02 January to 19 October 2020.

1) Daily Timeframe

The best RMSE was found to be $0{,}624 \times 10^{-2}$ using 2 hidden layers and 10 neurons with 192 epoch and dropout layer (0,1 rate). With 2510 records of train data used, the resulted RMSE value varies from each hyperparameter as shown in Table III. The use of 1 neuron was found to have a bigger RMSE value than the number of other neurons for each number of hidden layers whether using dropouts or not. Noted that the number of resulted epochs was smaller for 1 neuron than the number of other neurons. The prediction results from a daily timeframe using the best hyperparameters model are shown in Fig.5.

TABLE III
COMPARISON OF RMSE RESULT FOR EACH HYPERPARAMETERS IN DAILY TIMEFRAME

Hidden Layer	Neuron	No Dropout		Dropout (0,1 rate)	
		Epoch	RMSE	Epoch	RMSE
1	1	34	$7{,}764 \times 10^{-2}$	35	$7{,}917 \times 10^{-2}$
1	5	102	$0{,}689 \times 10^{-2}$	83	$1{,}002 \times 10^{-2}$
1	10	83	$0{,}699 \times 10^{-2}$	91	$1{,}002 \times 10^{-2}$
2	1	33	$9{,}722 \times 10^{-2}$	33	$9{,}967 \times 10^{-2}$
2	5	190	$0{,}693 \times 10^{-2}$	215	$0{,}668 \times 10^{-2}$
2	10	225	$0{,}693 \times 10^{-2}$	192	$0{,}624 \times 10^{-2}$
3	1	34	$9{,}538 \times 10^{-2}$	34	$9{,}739 \times 10^{-2}$
3	5	121	$0{,}858 \times 10^{-2}$	64	$1{,}456 \times 10^{-2}$
3	10	136	$0{,}826 \times 10^{-2}$	215	$0{,}79 \times 10^{-2}$

We can see in Fig. 5 that the trend of the prediction results was fit with the actual trend even though, at the end of the prediction results the difference between the predicted values and the actual values were getting bigger. The prediction results were closer to the actual data on a chart trend that tends to be constant and not fluctuate or sideways. It might occur because more data had a constant trend hence, the model learns more of such pattern.

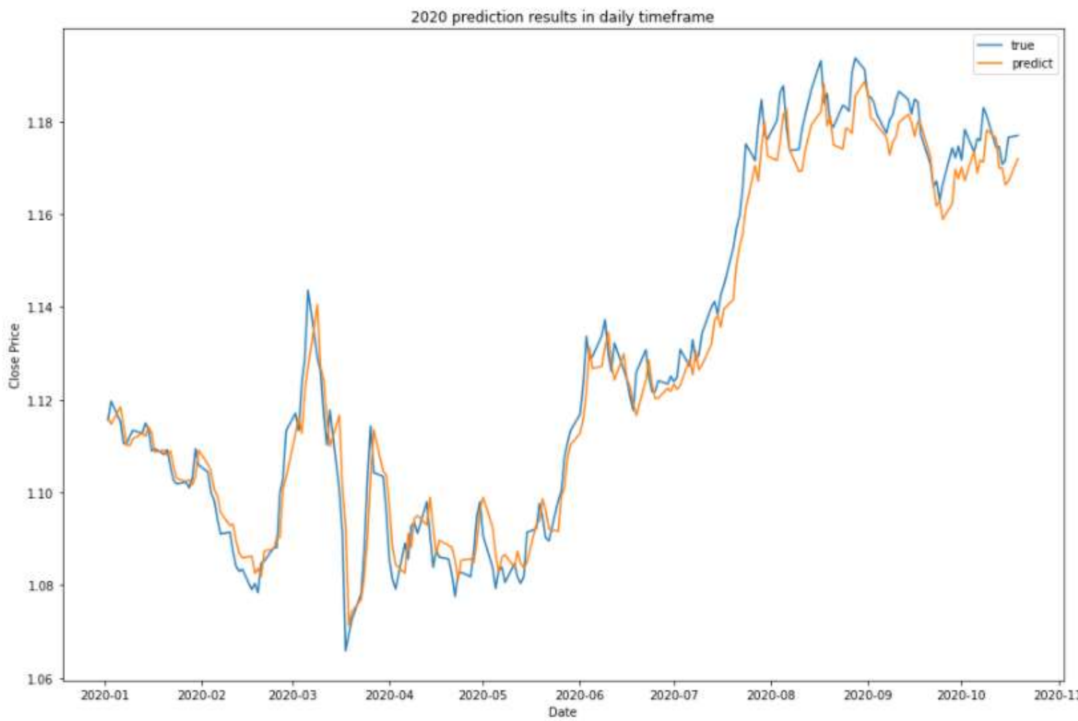

Fig. 5. 2020 Prediction Results in Daily Timeframe Using Best Hyperparameters

2) *1-hour Timeframe*

The best RMSE was found to be $0{,}135 \times 10^{-2}$. Surprisingly, it only took 1 hidden layer and 5 neurons with 272 epochs and without a dropout layer despite the large amounts of train data used which were 59914 records and the characteristic of 1-hour timeframe data which should be more volatile than daily timeframe data. As shown in Table IV, the range of RMSE results in each hyperparameter without a dropout layer is not too large compared with each hyperparameter with a dropout layer. Apparently, using the dropout layer obtained a bigger error in this case. It might because of the large amount of data that was used and the architecture of the model that was not complex thus when the dropout layer was added, the model became simpler and couldn't fit properly with the data. When compared to the daily timeframe, 1-hour timeframe gave a better RMSE result despite being more volatile than the daily timeframe. This might occur because the amount of training data in 1-hour timeframe was larger than in the daily timeframe. The prediction results from 1-hour timeframe using the best hyperparameters model are shown in Fig.6.

TABLE IV
COMPARISON OF RMSE RESULT FOR EACH HYPERPARAMETERS IN 1-HOUR TIMEFRAME

Hidden Layer	Neuron	No Dropout		Dropout (0,1 rate)	
		Epoch	RMSE	Epoch	RMSE
1	1	40	$0{,}193 \times 10^{-2}$	78	$2{,}478 \times 10^{-2}$
1	5	272	$0{,}135 \times 10^{-2}$	74	$1{,}565 \times 10^{-2}$
1	10	39	$0{,}386 \times 10^{-2}$	69	$1{,}105 \times 10^{-2}$
2	1	43	$0{,}215 \times 10^{-2}$	41	$2{,}88 \times 10^{-2}$
2	5	73	$0{,}169 \times 10^{-2}$	74	$1{,}728 \times 10^{-2}$
2	10	73	$0{,}155 \times 10^{-2}$	69	$1{,}105 \times 10^{-2}$
3	1	802	$0{,}188 \times 10^{-2}$	37	$3{,}195 \times 10^{-2}$
3	5	49	$0{,}401 \times 10^{-2}$	74	$1{,}49 \times 10^{-2}$
3	10	51	$0{,}275 \times 10^{-2}$	64	$1{,}052 \times 10^{-2}$

We can see in Fig.6 that the prediction results were more fit and closer to actual data compared to Fig.5. Different from the daily timeframe, the predicted values were closer to actual data at the end of the prediction results. This might occur because a lot of data used in this model thus, the model learns more of patterns including sideways trends.

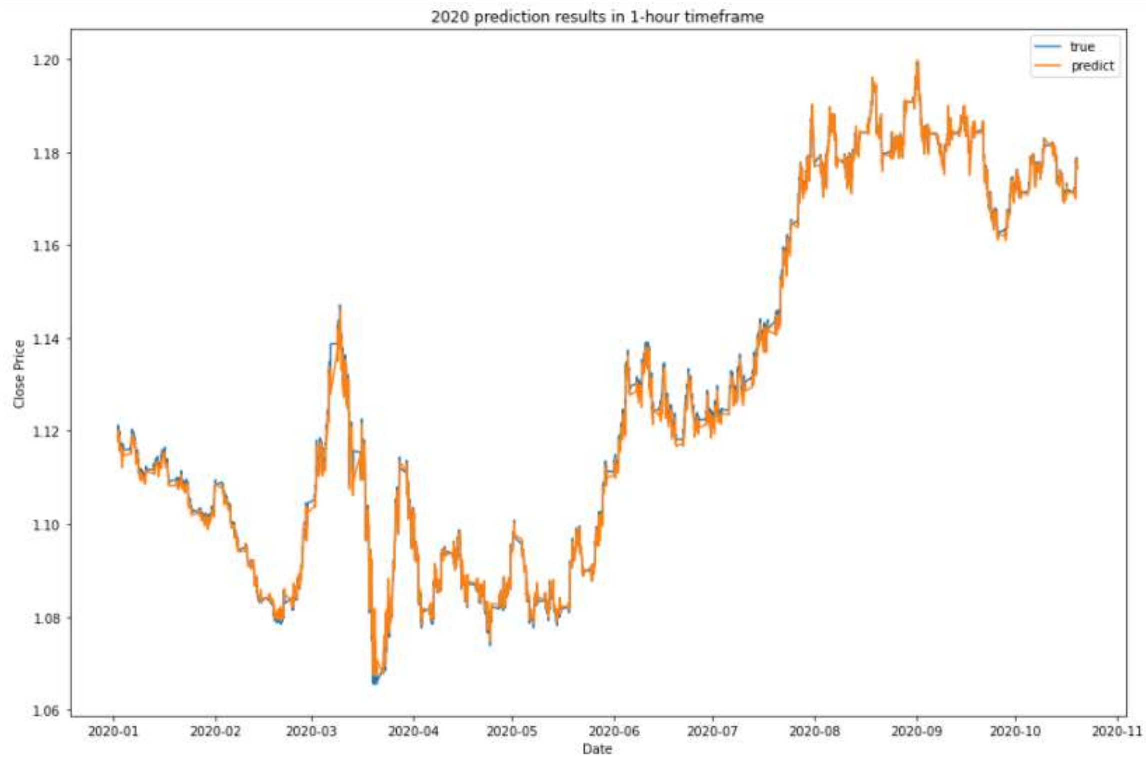

Fig. 6. 2020 Prediction Results in 1-hour Timeframe Using Best Hyperparameters

B. *Comparison of 2020 Prediction with Previous Years*

This testing was done using the same hyperparameters obtained from the best hyperparameters in chapter IV section A to get a comparison between the predictions of 2020 with the predictions of 2018 and 2019 based on the RMSE result. Furthermore, from this comparison, it can be concluded whether the COVID-19 that occurred in 2020 affect the model's performance. 4 May 2010 to 29 December 2017 data used as the training set for these models and 2018 data used as the testing set 1, 2019 data used as the testing set 2, and 2020 data used as the testing set 3. The results in Table V show that 2020 has the biggest RMSE result compared to previous years although, the difference is not too large. However, it means that the performances of these models are better for previous years when there was no COVID-19 pandemic. The RMSE result of 2019 in each timeframe has the smallest value. This might because the chart trend of 2019 was constant throughout the year therefore, the model could capture the constant patterns, different from 2020 which was the opposite.

TABLE V
COMPARISON OF 2020 RMSE RESULT WITH PREVIOUS YEARS

Timeframe	**Hidden Layer**	**Neuron**	**Epoch**	**Dropout**	**RMSE**		
					2018	**2019**	**2020**
Daily	2	10	192	0,1	$0{,}548 \times 10^{-2}$	$0{,}458 \times 10^{-2}$	$0{,}661 \times 10^{-2}$
1-hour	1	5	272	No dropout	$0{,}131 \times 10^{-2}$	$0{,}111 \times 10^{-2}$	$0{,}18 \times 10^{-2}$

The predictions of 2018, 2019, and 2020 in 1-hour timeframe gave a better RMSE result as shown in Table V compared to the predictions in daily timeframe, similar to what we got in section A. This might because the number of data records in 1-hour timeframe was bigger than in daily timeframe which was 47515 records for training set in 1-hour timeframe and 1992 records for training set in daily timeframe. The predictions result in Fig.7 were less fit to actual data compared to Fig.8.

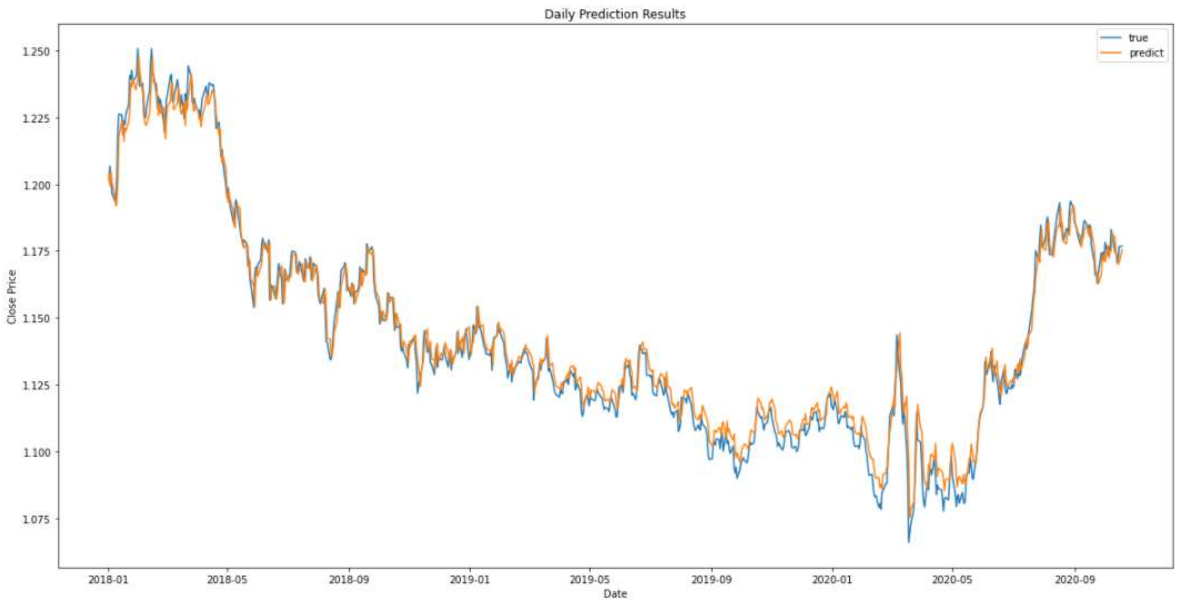

Fig. 7. 2018-2020 Daily Prediction Results

As shown in Fig.7 and Fig.8, at the end of 2019 and early 2020 the difference between the predicted values and the actual values is quite visible. Following that, there were significant fluctuations in February to March 2020 when EUR/USD price rose from 1.07838 to 1.14358 which meant the US Dollar fell against the Euro. It was because the investors bet the Federal Reserve would cut interest rates to offset the impact of a spreading coronavirus[15]. After that, the price fell immediately from 1.14358 to 1.06589 which meant the Euro fell against the US Dollar. It was because the European Central Bank announced more stimulus to fight the coronavirus impact but did not lower interest rates[16]. The chart trend continued to become sideways indicating the previous turmoil that caused a bigger difference between predicted values and actual values. Apparently, the model couldn't really capture the pattern because of the different price ranges in the previous fluctuations. However, at the end of the prediction, the model managed to learn the patterns and gave results that were closer to actual data.

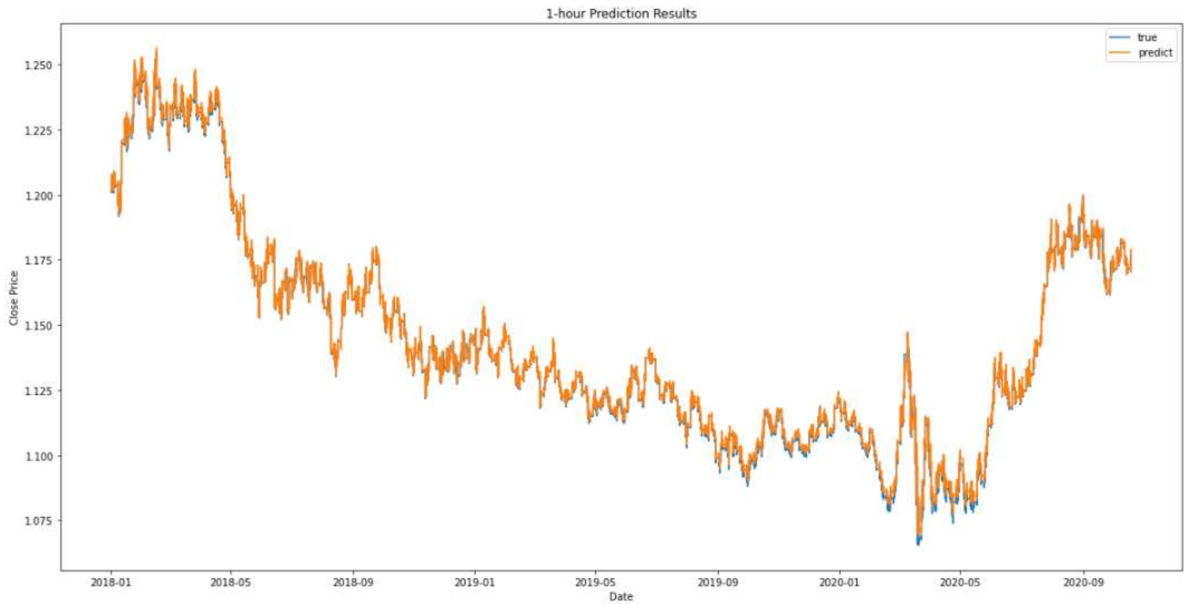

Fig. 8. 2018-2020 1-hour Prediction Results

V. CONCLUSION

The LSTM model was implemented to predict the 2020 EUR/USD rate in a 1-hour and daily timeframe. The best hyperparameters in the daily timeframe were found to be 2 hidden layers and 10 neurons with a dropout layer (0,1 rate) and the RMSE result was $0{,}624 \times 10^{-2}$. Meanwhile, the best hyperparameters in 1-hour timeframe were found to be 1 hidden layer and 5 neurons without a dropout layer and the RMSE result was $0{,}135 \times 10^{-2}$. 1-hour timeframe gave a better prediction result than the daily timeframe. It might occur because the amount of training data in 1-hour timeframe was larger than in the daily timeframe. Thus, the model learned more of the patterns and could give a better result.

At the end of 2019, the world was faced with a COVID-19 pandemic that has induced chaos and turbulence in the financial market. Apparently, it has quite an impact on the LSTM model's performance, proven by the RMSE result for 2020 which is the biggest when compared to 2018's and 2019's RMSE results. The model was more able to handle constant trends such as in 2019 – as evidenced in its RMSE result which is the smallest and was quite difficult to deal with fluctuations such as in early 2020 because of COVID-19 impact. However, the LSTM model managed to learn the patterns and still able to give good results in the 2020 prediction, proven by the chart trend of the prediction results that still fit with the actual data.

We plan to test the model in live trading for future work because if we observed closely in previous results especially in daily timeframe prediction, the trend of predicted values was just mimicking the last input. That means whatever we inputted in; the output will be the same as the last input.

VI. DATA AND COMPUTER PROGRAM AVAILABILITY

Data and programs used in this paper can be accessed at the following site https://github.com/hanazahrah/Forex-Prediction-using-LSTM.

REFERENCES

[1] S. Ranjit, S. Shrestha, S. Subedi, and S. Shakya, "Foreign Rate Exchange Prediction Using Neural Network and Sentiment Analysis," *Proc. - IEEE 2018 Int. Conf. Adv. Comput. Commun. Control Networking, ICACCCN 2018*, no. Icacccn, pp. 1173–1177, 2018, doi: 10.1109/ICACCCN.2018.8748819.

[2] J. W. Goodell, "COVID-19 and finance: Agendas for future research," *Financ. Res. Lett.*, vol. 35, no. March, 2020, doi: 10.1016/j.frl.2020.101512.

[3] B. Njindan Iyke, "The Disease Outbreak Channel of Exchange Rate Return Predictability: Evidence from COVID-19," *Emerg. Mark. Financ. Trade*, vol. 56, no. 10, pp. 2277–2297, 2020, doi: 10.1080/1540496X.2020.1784718.

[4] F. Aslam, S. Aziz, D. K. Nguyen, K. S. Mughal, and M. Khan, "On the efficiency of foreign exchange markets in times of the COVID-19 pandemic," *Technol. Forecast. Soc. Change*, vol. 161, no. June, p. 120261, 2020, doi: 10.1016/j.techfore.2020.120261.

[5] A. A. Philip, "Artificial Neural Network Model for Forecasting Foreign Exchange Rate," vol. 1, no. 3, pp. 110–118, 2011.

[6] K. Chen, Y. Zhou, and F. Dai, "A LSTM-based method for stock returns prediction: A case study of China stock market," *Proc. - 2015 IEEE Int. Conf. Big Data, IEEE Big Data 2015*, pp. 2823–2824, 2015, doi: 10.1109/BigData.2015.7364089.

[7] S. Zhelev and D. R. Avresky, "Using LSTM Neural Network for Time Series Predictions in Financial Markets," *2019 IEEE 18th Int. Symp. Netw. Comput. Appl. NCA 2019*, pp. 1–5, 2019, doi: 10.1109/NCA.2019.8935009.

[8] Y. C. Shiao, G. Chakraborty, S. F. Chen, L. Hua Li, and R. C. Chen, "Modeling and Prediction of Time-Series-A Case Study with Forex Data," *2019 IEEE 10th Int. Conf. Aware. Sci. Technol. iCAST 2019 - Proc.*, pp. 1–5, 2019, doi: 10.1109/ICAwST.2019.8923188.

[9] X. Zhang, X. Chen, L. Yao, C. Ge, and M. Dong, "Deep neural network hyperparameter optimization with orthogonal array tuning," in *Communications in Computer and Information Science*, Dec. 2019, vol. 1142 CCIS, pp. 287–295, doi: 10.1007/978-3-030-36808-1_31.

[10] S. Hochreiter and J. Schmidhuber, "Long Short-Term Memory," *Neural Comput.*, vol. 9, no. 8, pp. 1735–1780, Nov. 1997, doi: 10.1162/neco.1997.9.8.1735.

[11] Y. Bengio, P. Simard, and P. Frasconi, "Learning Long-Term Dependencies with Gradient Descent is Difficult," *IEEE Trans. Neural Networks*, vol. 5, no. 2, pp. 157–166, 1994, doi: 10.1109/72.279181.

[12] R. Dey and F. M. Salemt, "Gate-variants of Gated Recurrent Unit (GRU) neural networks," in *Midwest Symposium on Circuits and Systems*, Sep. 2017, vol. 2017-August, pp. 1597–1600, doi: 10.1109/MWSCAS.2017.8053243.

[13] D. P. Kingma and J. L. Ba, "Adam: A method for stochastic optimization," Dec. 2015, Accessed: Nov. 28, 2020. [Online]. Available: https://arxiv.org/abs/1412.6980v9.

[14] S. Ruder, "Overview Optimization Gradients," *arXiv Prepr. arXiv1609.04747*, pp. 1–14, 2017, [Online]. Available: https://arxiv.org/pdf/1609.04747.pdf.

[15] "Dollar retreats as coronavirus fallout raises expectations of rate cut." https://www.cnbc.com/2020/02/27/forex-markets-us-dollar-coronavirus-in-focus.html (accessed Dec. 16, 2020).

[16] "Euro falls after ECB holds fire, dollar jumps as spreads widen." https://www.cnbc.com/2020/03/12/forex-markets-ecb-coronavirus-in-focus.html (accessed Dec. 16, 2020).