FLF-LSTM: A novel prediction system using Forex Loss Function

Salman Ahmed [a], Saeed-Ul Hassan [a,*], Naif Radi Aljohani [b], Raheel Nawaz [c]

[a] Information Technology University, 346-B, Ferozepur Road, Lahore, Pakistan
[b] Faculty of Computing and Information Technology, King Abdulaziz University, Jeddah, Kingdom of Saudi Arabia
[c] Department of Operations, Technology, Events and Hospitality Management, Manchester Metropolitan University, Manchester, United Kingdom

Keywords:
Forex prediction system
Long Short-Term Memory (LSTM)
Forex forecasting
Forex Loss Function
Deep learning

A B S T R A C T

Foreign Exchange or Forex is the sale purchase market point of foreign currency pairs. Due to the high volatility in the forex market, it is difficult to predict the future price of any currency pair. This study shows that a significant enhancement in the prediction of forex price can be achieved by incorporating domain knowledge in the process of training machine learning models. The proposed system integrates the Forex Loss Function (FLF) into a Long Short-Term Memory model called FLF-LSTM — that minimizes the difference between the actual and predictive average of Forex candles. Using the data of 10,078 four-hour candles of EURUSD pair, it is found that compared to the classic LSTM model, the proposed FLF-LSTM system shows a decrease in overall mean absolute error rate by 10.96%. It is also reported that the error in forecasting the high and low prices is reduced by 10% and 9%, respectively. The proposed model, in comparison to the Recurrent Neural Network-based prediction system, shows an overall reduction of 73.57% in mean absolute error, by exhibiting up to 68.71% and 72.31% error reduction in high and low prices, respectively. In comparison to Auto-Regressive Integrated Moving Average, our proposed model shows a 13% reduced error. Specifically, in the open, high, and low prices, the error is reduced by 28.5%, 14.2%, 9.3%, respectively. Finally, we compare our model with another well-known time series forecasting model, i.e., FB Prophet — where FLF-LSTM demonstrates 31.8%, 47.7%, 23.6%, 47.7% error reduction in open, high, low, and close prices, respectively. The data and the code used in this study can be accessed at the following URL: https://github.com/slab-itu/forex_flf_lstm.

1. Introduction

Foreign Exchange (forex) is a decentralized, global, and highly resilient market where currency pairs are traded [1]. Trade is made based on bid and ask price. This market determines the foreign exchange rates based on supply and demand rules. In terms of trade, the forex is slightly different from the stock exchange [2]. In the stock exchange, shares of a particular company are purchased and sold when the prices increase (long trade). Whereas in forex trading, a short trade becomes possible where profit is made even if the prices go down [3]. When the prices depreciate, a quote currency is bought against the base currency, which leads to profit, and when the prices elevate, the base currency is bought against the quote currency. On average, the forex market has a 3 to 5 trillion-dollar business exchange, which fascinates the researchers to predict the trend in order to generate more profit [4,5].

Unlike stocks, the forex market is one of the utmost intricate markets because of its characteristics of high volatility, nonlinearity, and irregularity. Forex market is not controlled by a single institute or an organization, which makes it highly volatile and difficult to predict due to which forex prediction becomes an incredibly challenging problem [6]. There exist various methods to analyze the forex market, major ones being the fundamental and technical analysis. The fundamental analysis relies on forex news to predict the trend of the market, such as inflation rate, interest rate, economic growth, etc. [7]. Jin et al. [8] modeled real-time news from different sources, i.e., Google Search Volume and Twitter, to predict the trends by correlating the news to fluctuation in the financial market. Yang et al. [9] improved attention-based networks for sentiment analysis by designing the co-attention mechanism, which modeled both the target level and the context level attention representation. The authors also proposed location weighted functions that consider the location information to enhance the performance. Thompson et al. [10] worked on the extraction of event descriptions automatically from a news source by developing a meta-knowledge annotation scheme for news events. Meta-knowledge could be modality, subjectivity, source, polarity, or specificity of the event. A similar fundamental analysis based on the news source has been done by many recent publications [11–13].

Forex technical analysis is based on historical data, and it forecasts the trends of the market based on the prior data [14]. Yazdi & Lashkari [15] demonstrated the profit acquired by Moving Average Convergence and Divergence (MACD) by using an indicator known as MACD. Google trends could be used to find the trends regarding specific currency to evaluate supply and demand. Yu et al. [16] showed that the growth rate of pageviews on google trends has statistically significant effects on Bitcoin return volatility. Elliott Wave Pattern, in general, has high accuracy in terms of trend prediction. Volna et al. [17] conducted a study to detect Elliot Wave Pattern with 99% accuracy, insinuating the likeliness of getting a clearer trend. Nevertheless, these kinds of patterns may not be visible all the time. Yong et al. [18] investigated the forex time series data and concluded that there exist repeating patterns in forex time series data, which recommend the researches to use technical analysis for accurate prediction of the market. Although both fundamental and technical analysis is useful for the prediction of the market, people tend to rely more on technical analysis [19].

In this research study, LSTM was used to forecast the price of the next candle based on the H4 timeframe candles input, which could be helpful for the scalper. Note that the scalper is the type of trader who trades for a small number of pips and makes multiple trades in a single day. Multiple experiments were conducted on LSTM. The first experiment, named Single-LSTM, predicted the closing price of the next candle by providing the Open, High, Low, Close (OHLC) price of the current candle as an input. The second experiment, named Multi-LSTM, replicated the LSTM model from the first experiment four times to predict the open, high, low, and the close price of the next candle individually. The third experiment, named OHLC-LSTM, was conducted by providing the OHLC prices of the current value and predicting OHLC prices of the next candle. The results of all these experiments concluded that the MSE (the loss function) used in all these experiments does not have the domain knowledge of the market, which led us to the designing of the Forex Loss Function (FLF) - called as FLF-LSTM. To prove the performance of our model, we used the FLF loss function with Recurrent Neural Network (RNN) blocks and compared it with the proposed model. The results show that FLF-LSTM has reduced forecasting error by 73.57%. Similarly, we compared the proposed model with existing well-known time series forecasting models such as Auto-Regressive Integrated Moving Average (ARIMA) and FB Prophet. The results are confirming the improvement in the forecasting error when predicted by our newly proposed FLF-LSTM model.

The rest of the paper has been organized as follows: Section 2 contains a review of the recent literature in order to analyze the current status of the research in this field. The details of the data extraction and processing are provided in Section 3. Note that the dataset has been collected by writing our own script in Meta Quote Language 4 (MQL4) for Meta Trader 4 (MT4) in the form of an OHLC format. Further, Section 4 elaborates on the experimental details and results. Here, four different LSTM models i.e., Single-LSTM, Multi-LSTM, OHLC-LSTM, and FLF-LSTM, have been designed to forecast the price of the next candle in relation to FLF-RNN, ARIMA, and FB Prophet. Finally, Section 5 presents the conclusion and future work.

2. Literature review

Forex markets can be analyzed by different methods. Engel [20] investigated multiple econometric models that can be used for the prediction of the market; however, these types of models are not useful for making predictions of less than a year trend. Recently, Ali [21] illustrated that the Relative Strength Index (RSI) indicator could be used to find the relative strength of the market by finding the reversal and continuity of the market in a certain direction. Hansun and Karistanda [22] analyzed different types of moving averages, i.e., Simple Moving Average (SMA), Exponential Moving Average (EMA), and Weighted Moving Average (WMA). They demonstrated the prediction through EMA with an error of 10–5 using MSE as a loss function. Hansun [22], extended the work of Hansun et al. [22] and showed that Brown's Weighted Exponential Moving Average (B-WEMA) outperformed simple EMA with no big point difference. Wilinski [23] used a Markov model to forecast the price of the currency pair. Granger [24] investigated ARIMA, and Yazdi et al. [15] conducted Moving Average Convergence and Divergence (MACD) analysis to determine the reversal of the market and tested their model through robot for nine years and generated an overall profit of 24,424 pips on EURUSD.

Instead of using technical analysis, many of the researchers have conducted experiments using machine learning techniques. Thu et al. [25] used SVM to predict the trend of the market, and their results showed that SVM with Expert Advisor, a bot used for automatic trading, can generate more profit than simple Expert Advisor. They maximized the distance in support vectors to predict the class of "up" and "down" trends. Lee et al. [26] designed the network for stock market prediction for overall trend and relative regional direction. Results proved that the SVM showed the best performance in direction prediction. Tang et al. [27] used PCA data and concluded a better performance using KNN by reducing redundant and noisy information, which can help in the prediction of stocks and forex market. Apart from statistical and machine learning approaches, a trend of deploying deep learning is also observed in this domain. Galeshchuk [28] investigated that ANN with a single hidden layer sometimes outperforms the time series model in forecasting the price, proving to be useful for short-term prediction of the market. Jin et al. [8] proposed to track multiple social media platforms to predict the trend of the market. Weng et al. [29] used a search trend, unique web page visitors, and news sentiment data to predict the stock market by incorporating technical indicator data to improve their efficiency. Hu et al. [1] filtered the news, which is either fake or has no impact on the trend for the precise prediction and proposed Hybrid Attention Network using the LSTM model. Abdi et al. [30] showed the problem in the currently designed sentimental analysis model and tried to overcome those situations such as words with similar semantic context but opposite sentiment polarity, word coverage limit of an individual lexicon and word sense variations to obtain the exact sentiment from a sentence.

Galeshchuk and Mukherjee [13] compared deep networks with SVM and ANNs and observed ANNs perform well on such data. Selvin et al. [31] showed the comparison of multiple deep learning models and concluded the deep model system to be capable of identifying some interrelation with the data. Chandrinos et al. [32] developed an expert system to verify technical signals coming from breakout trading strategy using ANN and decision tree. Huang et al. [11] modified the LSTM model by integrating Bayesian optimization and highlighted the problem of window size selection and static parameter initialization. On the other hand, Lin and Chen [33] modified the LSTM to add Genetic Algorithm behavior in the model. The single layer of LSTM with 150 neurons had presented promising results in Lin's experiment. From the literature, it can be observed that LSTM is recent in the field of forecasting forex time series data, with the modification of LSTM working better than the vanilla LSTM. The study intends to modify LSTM by introducing a loss function that encompasses some domain knowledge of forex. The proposed method in this paper demonstrates promising results.

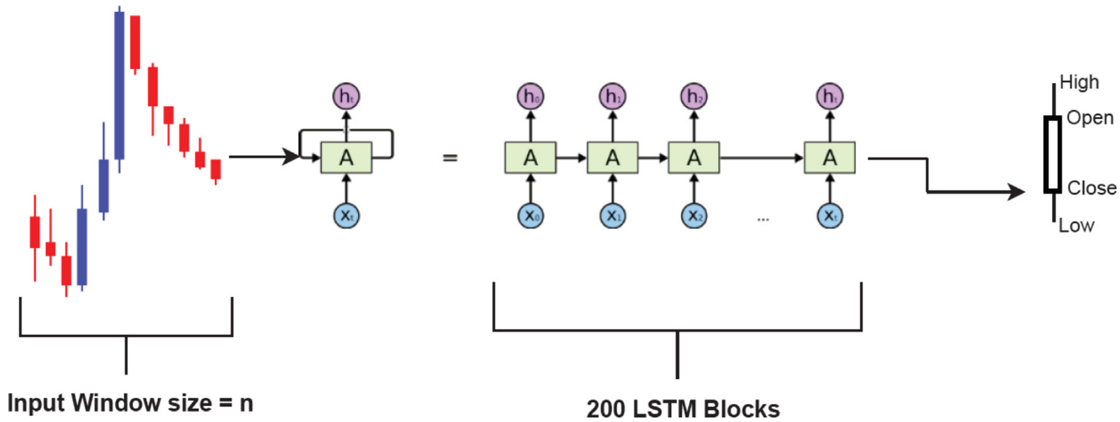

Fig. 1. Single-LSTM model; candles are the input; the next closing price of the timeframe is predicted.

3. Data and methods

This section explains the actionable smart dataset processing for each experiment, which is used for input and expected output of the deep learning model. Methodology and the hyperparameters of each experiment are also explained.

3.1. Smart data processing

The H4 candles EURUSD data was collected from June 2015 to Sep 2018 using an MT4 tool. This data is provided by the XM broker (https://www.xm.com/). The dataset was collected through a custom script written in MetaQuote Language 4, each row in a dataset represents open, high, low, and close price in a four-hour movement. The data contains the open, high, low, and close price of the currency pair, which is referred to as a candle, i.e., type of chart, to represent price. It comprises sequential candles price in OHLC format for training and testing; these candles are processed as per the need of the individual experiment to generate output. The total dataset of 10,078 candles was divided into 60–40 ratio, 6047 candles for the training, and 4031 candles for the testing for each experiment. For the first experiment, i.e., Single-LSTM model, the closing price of the next candle was predicted by using the current candle as an input. The output label of the model was calculated by Eq. (1)

$$Y_i = \varsigma_{i+1,3}, \tag{1}$$

where Y is the output, i.e., the closing price of the next candle and ς is the candle from the dataset. For the second experiment, i.e., the Multi-LSTM model, OHLC prices were prognosticated with the same input ς for every model but a different output, which was calculated using Eqs. (2), (3), (4), and (5). For this experiment, the same LSTM, as used previously in the Single-LSTM model, was replicated.

$$Y_{openi} = \varsigma_{i+1,0} \tag{2}$$

$$Y_{closei} = \varsigma_{i+1,3} \tag{3}$$

$$Y_{highi} = \varsigma_{i+1,1} \tag{4}$$

$$Y_{lowi} = \varsigma_{i+1,2} \tag{5}$$

$$Y_i = \varsigma_{i+1} \tag{6}$$

For the third experiment, i.e., OHLC-LSTM model, Eq. (6) was used where the candle was passed as an input and the expected output was the next candle. The same processed data is also used for the proposed method using the FLF-LSTM model. Details of all these experiments are provided in the next section.

3.2. Approaches

Multiple experiments were conducted, which led to the creation of the Forex Loss Function. Given below is the detail of each and every experiment conducted throughout the research period.

3.2.1. Single-LSTM model

An architecture using LSTM containing 200 cells using NADAM optimizer and a mean squared error (MSE) loss function was designed. The configuration was finalized after intensive experimentation. The window size of the candle one produced the minimum loss. This experiment was named as Single-LSTM Model (SLM). A candle is the input of this model containing four price values, i.e., OHLC. The closing value of the next candle is predicted from this model (see Fig. 1). The purpose of this experiment was to find the magnitude movement of the next four-hour market. It was observed that sometimes the market might retrace back near to its starting position. Hence, creating a Doji or hammer-like candle pattern in these cases of closing price may not help. These cases do not provide useful information to the scalper to estimate the direction of the market for the next four hours. Another experiment was done to predict full candles using multiple same models.

3.2.2. Multi-LSTM model

A new architecture was designed by repeating the same model as tested in the SLM. The model was replicated to predict an open, high, low, and close price of the next candle (see Fig. 2) from an individual model. The previous experiment was also included in it, and hence it was named as the Multi-LSTM Model (MLM). Configurations are the same for each model, as described in the earlier experiment. It was noted that this system does not learn the characteristics of the forex candle, for example, sometimes it predicts the low price higher than the n price, which is not possible. It was because all models are independent and they have no relation, but candle has relational characteristics, for example, the open price can neither be higher than high price, nor lower than the low price, but multi-LSTM was producing these out of range values that is it violates the structure of the candle. Thus, we claimed that Multi-LSTM is unable to learn the chart. It was concluded to shift the forecasting to one model, which led to the next experiment.

3.2.3. OHLC-LSTM model

It had already been witnessed that multiple models cannot work together to predict a single next candle. Hence a model was designed, which takes the current candle as an input and predicts the next whole candle in the form of OHLC (see Fig. 3). This experiment was named as the OHLC-LSTM (OHM). This architecture

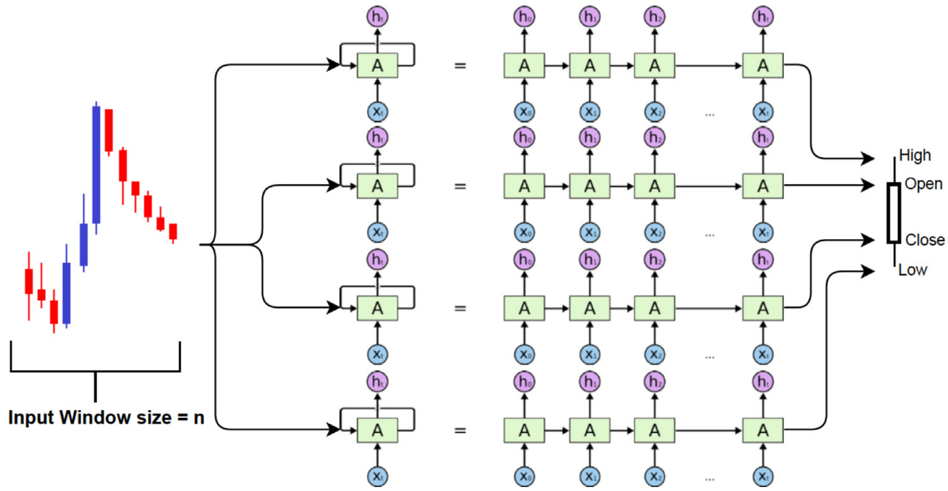

Fig. 2. Multi-LSTM model; OHLC price of the current candle is the input to the individual models, and open, high, low, close prices are predicted separately.

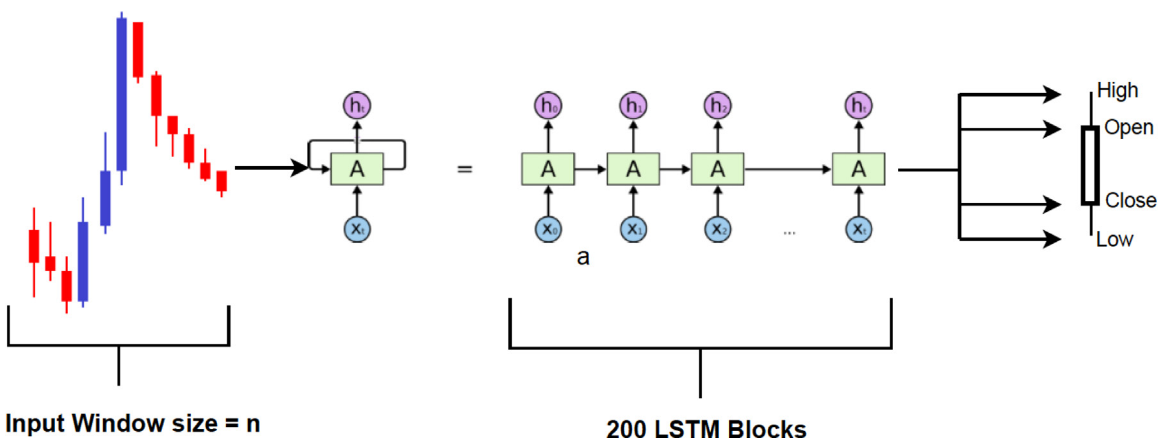

Fig. 3. OHLC-LSTM model; candles are the input; the next candle price of the timeframe is predicted. This same model is also applied in FLF-LSTM.

solved the former problem. Now, even though the candle prices are in order but the best error function so far, i.e., mean squared error does not have domain knowledge of the forex market. If the loss function could be designed specifically for the defined problem, greater precision can be obtained. To prove the above claims, a forex loss function (FLF) was designed, which led to the next experiment.

3.2.4. FLF-LSTM model

In all the experiments, it was observed that the mean squared error (see Eq. (7)) is producing minimum loss during the training and testing. The reason being the data, which is up to five decimal points and squaring small difference, produces smaller values, leading to smooth learning and minimum loss. This feature has been introduced in FLF and modified mean square error by subtracting the difference of the average of the high and low prices of actual and predicted values from high and low values. Similarly, we subtracted the difference of the average of the open and close price of the actual and predicted value in open and close values before taking the square and mean. In order to understand it mathematically, please refer to Eq. (8).

The output has four values; open, high, low, and close price of the candle, respectively. The open and the close values have been altered by including the difference of the average of actual and predicted length of the body of the candle, i.e. γ_i. Note that the length of the body is calculated by differentiating the open and close value of the candle.

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(Y_i - \widehat{Y}_i)^2 \tag{7}$$

where

$$\lambda = 0.9, \qquad \sigma = 0.1$$

$$\alpha_i = \lambda * \left(Y_i - \hat{Y}_i\right)$$

$$\beta_i = \sigma * \left(\frac{Y_{i,high} + Y_{i,low}}{2} - \frac{\hat{Y}_{i,high} + \hat{Y}_{i,low}}{2}\right)$$

$$\gamma_i = \sigma * \left(\frac{Y_{i,open} + Y_{i,close}}{2} - \frac{\hat{Y}_{i,open} + \hat{Y}_{i,close}}{2}\right)$$

$$FLF = \frac{1}{n}\sum_{i=1}^{n}\left[\alpha_{i,open} - \gamma_i\alpha_{i,high} - \beta_i\alpha_{i,low} - \beta_i\alpha_{i,close} - \gamma_i\right]^2 \tag{8}$$

Additionally, the high and low values were altered by subtracting the difference of the average of actual and predicted length of the candle, i.e., β_i, where $Y_{i,open}$, $Y_{i,high}$, $Y_{i,low}$ *and* $Y_{i,close}$ are the actual values of open, high, low, and close price respectively of the candle at specific index "i". On the other hand, $\hat{Y}_{i,open}$, $\hat{Y}_{i,high}$, $\hat{Y}_{i,low}$ *and* $\hat{Y}_{i,close}$ are the predicted values of the model. LSTM architecture for this experiment is similar to the OHM (as shown in Fig. 3).

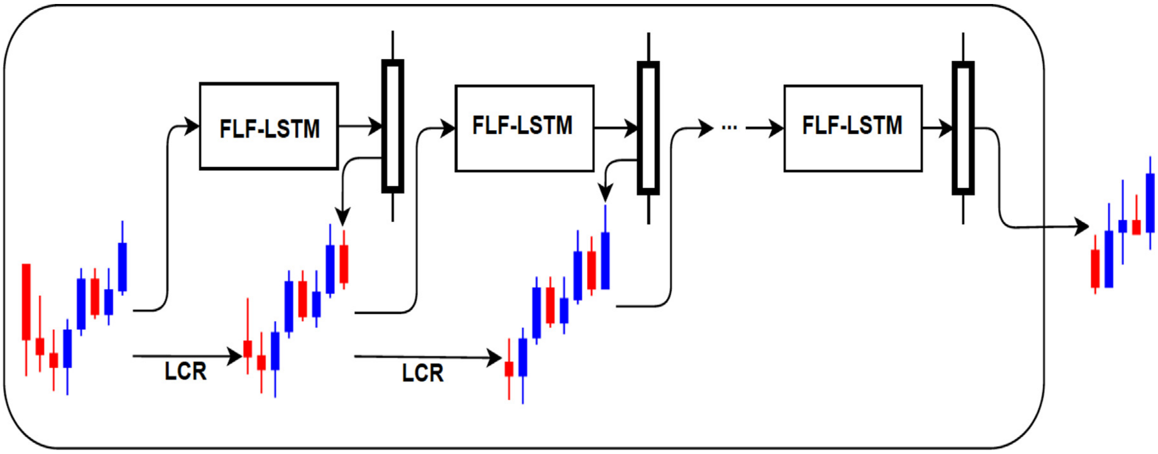

Fig. 4. System Designed by FLF-LSTM where input is the candles which are passed to the model, iteratively "last candle removed" (LCR), and predicted candle appended at the front of input for next candle prediction.

This experiment was named as FLF-LSTM (FLF). Also, note that the length of the candle is calculated by differentiating the high and low value of the candle.

3.2.5. System design and implementation

A system was developed using the proposed FLF-LSTM model, which can be used to predict the trend of the forex. Fig. 4 shows a system diagram where the *n* number of candles is passed to the model, which produces a single candle in the form of OHLC that could be either bull or bearish. This forecasted candle is appended in input, and the Last Candle is Removed (LCR) from it. This process is repeated to produce next m candles to make a trend. The problem with this system is that error in forecasted candles is accumulated after every iteration because the predicted candle, which may have little error is appended in the input, hence producing more error. To solve this problem, random bias *epsilon* range from [0.00001 to −0.00001] was added in all the OHLC values, which did not perform well. Hence, this problem will be sorted out in future research.

4. Experiments and results

All the experiments were implemented on Jupyter Notebook, using python Keras Neural Network library. The experiments were run on a machine with the Core-i7 intel process, 16 GB DDR3 Random Access Memory (RAM) from the Samsung brand, having Microsoft Windows-10 Operating System for LSTM and Neural network models, we used Keras version 2.3.1 under the TensorFlow version 2.0.0.

The custom function was implemented using Keras backend. In all the experiments (except for the proposed experiment) mean square error and NADAM were used as an loss function and optimizer, respectively, using parameters such as learning rate = 0.00001, beta_1 = 0.09, beta_2 = 0.0999, epsilon = none, schedule decay rate = 0.0004, epoch size = 150 and batch size = 72. Dataset was randomly shuffled while training. All these parameters were selected based on extensive experiments. For the proposed method, the optimizer has the same parameters, while the loss function is the FLF (see Eq. (8)). We executed multiple experiments to find a suitable activation function for each model. Table 1 shows that relu is performing better as compared to tanh and sigmoid. When comparing to other models, FLF-LSTM is outperforming Multi-LSTM and OHLC-LSTM by 67.06% and 10.96% respectively by reducing mean absolute error.

These results are computed by a comparison of the actual and predicted open, high, low, and close price, which are taken from the H4 timeframe candle of EURUSD. These prices are selected randomly from 4031 candles.

Table 1
Mean absolute error of multiple models with different activation functions.

Model	Activation Function	Open values	High values	Low values	Close values
FLF-LSTM	Tanh	0.005010	0.004020	0.005188	0.004359
	Sigmoid	0.127865	0.127864	0.117745	0.145468
	Relu	**0.001549**	**0.001885**	**0.002990**	**0.002332**
FLF-RNN	Tanh	0.008731	0.009542	0.017097	0.008723
	Sigmoid	0.123778	0.131410	0.112228	0.133470
	Relu	0.008117	0.006011	0.010818	0.008255
Multi-LSTM	Tanh	0.005642	0.007064	0.004682	0.006939
	Sigmoid	0.130249	0.131158	0.121249	0.134488
	Relu	0.007510	0.007889	0.007190	0.006782
OHLC-LSTM	Tanh	0.003787	0.004712	0.005929	0.005113
	Sigmoid	0.130249	0.131158	0.121249	0.134488
	Relu	0.001917	0.002028	0.003218	0.002607

Table 2
Mean Absolute Error (MAE) reduction by the FLF-LSTM as compared to the FLF-RNN in open, high, low, and close value.

Price	FR_{MAE}	FL_{MAE}	$\frac{FR_{MAE}-FL_{MAE}}{FR_{MAE}} * 100$
Open	8.11×10^{-3}	1.50×10^{-3}	81.50%
High	6.01×10^{-3}	1.88×10^{-3}	68.71%
Low	1.08×10^{-2}	2.99×10^{-3}	72.31%
Close	8.25×10^{-3}	2.33×10^{-3}	71.75%

MAE of FLF-RNN = $FR_{MAE;}$ MAE of FLF-LSTM = FL_{MAE}.

In Table 2, we compared the mean absolute error of FLF-RNN and FLF-LSTM, and the result shows that the LSTM-based model performs better as compared to the RNN-based prediction model. Furthermore, we have compared the training time taken by each experiment on a specified system. As shown in Fig. 5, FLF-RNN takes minimum time to train as compared to FLF-LSTM. While Multi-LSTM has multiple models for training, hence, taking the maximum time.

Furthermore, we compare open, high, low, and close actual prices with respect to predicted Multi-LSTM, OHLC-LSTM, and FLF-LSTM prices (see Fig. 6). These graphs show that the model has learned the trend of the market and is predicting the prices based on the actual trend.

The proposed model was compared with the existing well-known forecasting algorithms (see Table 3). First, the FLF-LSTM results were compared with the FB Prophet [34]. The test set length period was forecasted by providing the trained set to the

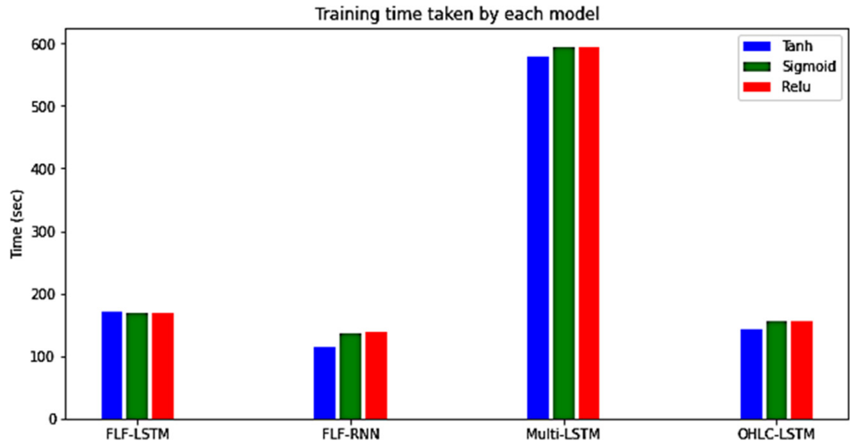

Fig. 5. Training time is taken by each model.

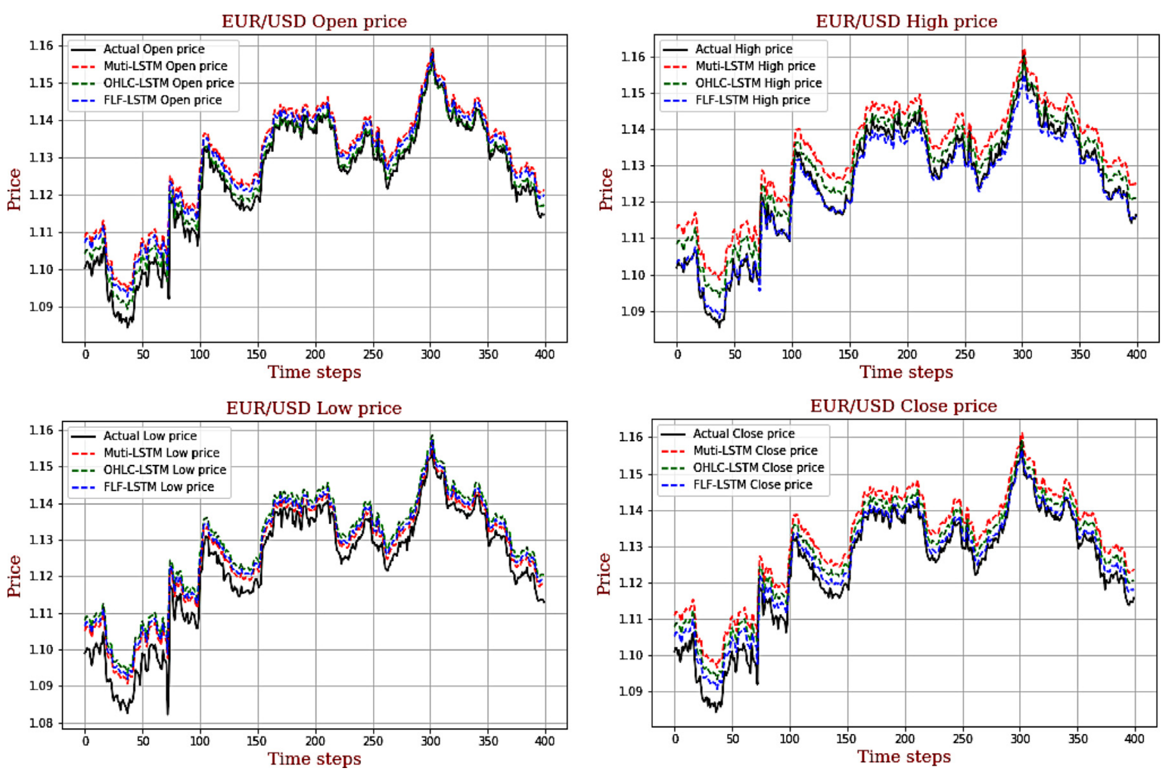

Fig. 6. Open, high, low and close actual price as compared to Multi-LSTM, OHLC-LSTM and FLF-LSTM predicted price.

Table 3
Prediction error by FB Prophet, ARIMA and FLF-LSTM, and error reduced by FLF-LSTM as compared to ARIMA and FB Prophet.

Price	AR_{MAE}	$\frac{AR_{MAE}-FL_{MAE}}{AR_{MAE}} * 100$	FP_{MAE}	$\frac{FP_{MAE}-FL_{MAE}}{FP_{MAE}} * 100$
Open	2.1e−3	28.5%	2.2e−3	31.8%
High	2.1e−3	14.2%	3.2e−3	47.7%
Low	3.2e−3	9.3%	3.8e−3	23.6%
Close	2.3e−3	0.0%	4.4e−3	47.7%

MAE of ARIMA = AR_{MAE}, MAE of FLF-LSTM = FL_{MAE}, MAE of FB Prophet = FP_{MAE}.

FB Prophet, but results were reaching the negative values, which is not possible; hence one-period price value in a single time was forecasted.

Four different models were used to forecast open, high, low, and close prices individually. As shown in Table 3, FLF-LSTM based model has a reduced average 37.7% prediction error as compared to FB Prophet. Similarly, our model was compared with the ARIMA (Granger [24]). Four different models were designed to forecast the open, high, low, and close prices individually using the p = 5, d = 1, and q = 0. These values were selected after the experiments. Overall, Table 3 shows that the FLF-LSTM results are better than the ARIMA by 13%. The literature shows that RNN-based models do not learn long patterns in the data [35]. This trend is also observed in our experiments, as our proposed LSTM-based model outperforms the RNN-based model. While ARIMA performs next to FLF-LSTM due to its ability to handle seasonality trends in the data by design, FB Prophet is not working well on this type of data that further may be improved with the inclusion of human-interpretable parameters and domain knowledge in the future studies.

5. Concluding remarks

The study has proposed a novel system by integrating the Forex Loss Function with LSTM, that outperforms classic LSTM-based prediction systems by decreasing the mean absolute error on EURUSD dataset as follows: 10.54% in the closing price, 7.05% in high price and 7.08% in the low price along with a reduction in error by 19.19% in the opening price of the next H4 candle. The results show an overall 10.96% improvement in prediction error than classic LSTM. Comparing with the FLF-RNN, FLF-LSTM is performing better, and reduces 73.57% overall forecasting error. As compared to the ARIMA and the FB Prophet, benchmark results of the proposed FLF-LSTM show 13% and 37.7% reduced error, respectively. This study shows that by incorporating domain knowledge in the learning process enhances the predictability. Overall, the proposed system implementation opens a new gateway for the scientific community that seeks to predict forex market trends. In future studies, we propose to build knowledge-based decision support recommender systems [36–38] for forex trade. The data and code used in this study can be accessed at the following URL: https://github.com/slab-itu/forex_flf_lstm.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

[1] Ziniu Hu, Weiqing Liu, Jiang Bian, Xuanzhe Liu, Tie-Yan Liu, 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, ACM, 2018, pp. 261–269.

[2] João Carapuço, Rui Neves, Nuno Horta, Reinforcement learning applied to forex trading, Appl. Soft Comput. 73 (2018) 783–794.

[3] Rajashree Dash, Performance analysis of a higher-order neural network with an improved shuffled frog leaping algorithm for currency exchange rate prediction, Appl. Soft Comput. 67 (2018) 215–231.

[4] Bernardo Jubert de Almeida, Rui Ferreira Neves, Nuno Horta, Combining support vector machine with genetic algorithms to optimize investments in forex markets with high leverage, Appl. Soft Comput. 64 (2018) 596–613.

[5] Carlos Martín, David Quintana, Pedro Isasi, Grammatical evolution-based ensembles for algorithmic trading, Appl. Soft Comput. 84 (2019) 105713.

[6] Jonatan Henríquez, Werner Kristjanpoller, A combined independent component analysis–neural network model for forecasting exchange rate variation, Appl. Soft Comput. 83 (2019) 105654.

[7] S. Nootyaskool, W. Choengtong, Hidden Markov models predict foreign exchange rate, in: 2014 14th International Symposium on Communications and Information Technologies (ISCIT), IEEE, 2014, pp. 99–101.

[8] Fang Jin, Wei Wang, Prithwish Chakraborty, Nathan Self, Feng Chen, Naren Ramakrishnan, Tracking multiple social media for stock market event prediction, in: Industrial Conference on Data Mining, Springer, Cham, 2017, pp. 16–30.

[9] Chao Yang, Hefeng Zhang, Bin Jiang, Keqin Li, Aspect-based sentiment analysis with alternating coattention networks, Inf. Process. Manage. 56 (3) (2019) 463–478.

[10] Paul Thompson, Raheel Nawaz, Ioannis Korkontzelos, William Black, John McNaught, Sophia Ananiadou, News search using discourse analytics, in: 2013 Digital Heritage International Congress (DigitalHeritage), Vol. 1, IEEE, 2013, pp. 597–604.

[11] Raheel Nawaz, Paul Thompson, John McNaught, Sophia Ananiadou, Meta-knowledge annotation of bio-events, in: LREC, Vol. 17, 2010, pp. 2498–2507.

[12] Raheel Nawaz, Paul Thompson, Sophia Ananiadou, Negated bio-events: analysis and identification, BMC Bioinformatics 14 (1) (2013) 14.

[13] Svitlana Galeshchuk, Sumitra Mukherjee, Deep networks for predicting direction of change in foreign exchange rates, Intell. Syst. Account. Finance Manage. 24 (4) (2017) 100–110.

[14] Omer Berat Sezer, Ahmet Murat Ozbayoglu, Algorithmic financial trading with deep convolutional neural networks: Time series to image conversion approach, Appl. Soft Comput. 70 (2018) 525–538.

[15] Seyed Hadi Mir Yazdi, Ziba Habibi Lashkari, Technical analysis of forex by MACD indicator, Int. J. Hum. Manage. Sci. (IJHMS) 1 (2) (2013) 159–165.

[16] Ju Hyun Yu, Juyoung Kang, Sangun Park, Information availability and return volatility in the bitcoin market: Analyzing differences of user opinion and interest, Inf. Process. Manage. 56 (3) (2019) 721–732.

[17] Eva Volna, Martin Kotyrba, Robert Jarušek, Prediction by means of Elliott waves recognition, in: Nostradamus: Modern Methods of Prediction, Modeling and Analysis of Nonlinear Systems, Springer, Berlin, Heidelberg, 2013, pp. 241–250.

[18] Yoke Leng Yong, Yunli Lee, David Ngo, An investigation into the recurring patterns of forex time series data, in: 2015 IEEE International Symposium on Robotics and Intelligent Sensors (IRIS), IEEE, 2015, pp. 313–317.

[19] Yu-Hon Lui, David Mole, The use of fundamental and technical analyses by foreign exchange dealers: Hong Kong evidence, J. Int. Money Finance 17 (3) (1998) 535–545.

[20] Charles Engel, Exchange rates and interest parity, in: Handbook of International Economics, Vol. 4, Elsevier, 2014, pp. 453–522.

[21] Jauhar Ali, Analyzing the effect of currency strength changes on their future price movement, in: Proceedings of the 2018 10th International Conference on Machine Learning and Computing, ACM, 2018, pp. 111–115.

[22] Seng Hansun, Marcel Bonar Kristanda, Performance analysis of conventional moving average methods in forex forecasting, in: 2017 International Conference on Smart Cities, Automation & Intelligent Computing Systems (ICON-SONICS), IEEE, 2017, pp. 11–17.

[23] Antoni Wilinski, Time series modeling and forecasting based on a Markov chain with changing transition matrices, Expert Syst. Appl. 133 (2019) 163–172.

[24] Clive W.J. Granger, Some properties of time series data and their use in econometric model specification, J. Econometrics 16 (1) (1981) 121–130.

[25] Thuy Nguyen Thi Thu, Vuong Dang Xuan, Using support vector machine in forex predicting, in: 2018 IEEE International Conference on Innovative Research and Development (Icird), IEEE, 2018, pp. 1–5.

[26] Tae Kyun Lee, Joon Hyung Cho, Deuk Sin Kwon, So Young Sohn, Global stock market investment strategies based on financial network indicators using machine learning techniques, Expert Syst. Appl. 117 (2019) 228–242.

[27] Li Tang, Heping Pan, Yiyong Yao, K-nearest neighbor regression with principal component analysis for financial time series prediction, in: Proceedings of the 2018 International Conference on Computing and Artificial Intelligence, ACM, 2018, pp. 127–131.

[28] Svitlana Galeshchuk, Neural networks performance in exchange rate prediction, Neurocomputing 172 (2016) 446–452.

[29] Bin Weng, Lin Lu, Xing Wang, Fadel M. Megahed, Waldyn Martinez, Predicting short-term stock prices using ensemble methods and online data sources, Expert Syst. Appl. 112 (2018) 258–273.

[30] Asad Abdi, Siti Mariyam Shamsuddin, Shafaatunnur Hasan, Jalil Piran, Deep learning-based sentiment classification of evaluative text based on multi-feature fusion, Inf. Process. Manage. 56 (4) (2019) 1245–1259.

[31] Sreelekshmy Selvin, R. Vinayakumar, E.A. Gopalakrishnan, Vijay Krishna Menon, K.P. Soman, Stock price prediction using LSTM, RNN and CNN-sliding window model, in: 2017 International Conference on Advances in Computing, Communications and Informatics (ICACCI), IEEE, 2017, pp. 1643–1647.

[32] Spyros K. Chandrinos, Georgios Sakkas, Nikos D. Lagaros, AIRMS: A risk management tool using machine learning, Expert Syst. Appl. 105 (2018) 34–48.

[33] Moule Lin, Changxi Chen, Short-term prediction of stock market price based on GA optimization LSTM neurons, in: Proceedings of the 2018 2nd International Conference on Deep Learning Technologies, ACM, 2018, pp. 66–70.

[34] Sean J. Taylor, Benjamin Letham, Forecasting at scale, Amer. Statist. 72 (1) (2018) 37–45.

[35] Alex Sherstinsky, Fundamentals of recurrent neural network (rnn) and long short-term memory (lstm) network, Physica D 404 (2020) 132306, http://dx.doi.org/10.1016/j.physd.2019.132306.

[36] Fahad Anwaar, Naima Iltaf, Hammad Afzal, Raheel Nawaz, HRS-CE: A hybrid framework to integrate content embeddings in recommender systems for cold start items, J. Comput. Sci. 29 (2018) 9–18.

[37] Sundus Ayyaz, Usman Qamar, Raheel Nawaz, HCF-CRS: A hybrid content based fuzzy conformal recommender system for providing recommendations with confidence, PLoS One 13 (10) (2018) e0204849.

[38] Hajra Qadir, Osman Khalid, Muhammad U.S. Khan, Atta Ur Rehman Khan, Raheel Nawaz, An optimal ride sharing recommendation framework for carpooling services, IEEE Access 6 (2018) 62296–62313.