

# FOREX Prices Prediction Using Deep Neural Network and FNF

Asmaa M. Moustafa<sup>1\*</sup>, Mohamed Waleed Fakhr<sup>2</sup>, Fahima A. Maghraby<sup>3</sup>

Arab Academy for Science Technology & Maritime Transport-College of Computing and Information Technology,  
Heliopolis, Cairo Governorate, Egypt

**Abstract**—One of the largest financial markets on the planet is the foreign exchange (FOREX) market. Banks, retail traders, businesses, and individuals trade more than \$5.1 trillion in FOREX daily. It is very challenging to predict prices in advance due to the market's complex, volatile, and highly fluctuating nature. In this study, the new FOREX Normalization Function (FNF) is proposed and used with different models to predict the prices of the AUD/USD, EUR/USD, USD/JPY, CHF/INR, USD/CHF, AUD/JPY, USD/CAD, and GBP/USD. Two models are proposed in this study. The first model contains FNF as a normalization and feature extractor, followed by a Convolutional Neural Network (CNN). The second model utilizes FNF and a Support Vector Regressor (SVR). The forecasts are set for a one-day timeframe, with predictions made for 1, 3, 7, and 15 days ahead. The efficient ability of the proposed method to solve the FOREX prediction problem is proven by performing experiments on nine real-world datasets from different currencies. Additionally, the models are evaluated using Mean Absolute Error (MAE) and Mean Squared Error (MSE). Applying the presented models to 9 different datasets improved the results by an average between 0.5% and 58% of MAE.

**Keywords**—FOREX prediction; CNN; normalization function; SVR

## I. INTRODUCTION

FOREX, also known as "foreign exchange" involves changing one currency into another. Every day, traders trade trillions of dollars [1]. Due to significant currency rate fluctuations, this market is unpredictable, complicated, and subject to frequent changes [2]. The market is always open, although trading takes place in the four main time zones: European, Asian, Australian, and North American [3]. The opening and closing hours of each of these zones differ. The market is protected from scammers since it takes significant money to impact exchange rates. Over the past few decades, scholars have become increasingly interested in forecasting foreign exchange. Unlike the stock market, the foreign exchange market doesn't require large amounts of cash. Leverage is one of the most critical tools related to the market. Leverage is the process of increasing the future return on investment by using borrowed funds [4]. Traders, individuals, professionals making expensive purchases, entrepreneurs, and investors employ leverage. This approach is beneficial for those with little finances and is also a vital element of the FOREX market that attracts private and small investors.

Both technical analysis and fundamental analysis can be used to forecast FOREX prices. While technical analysis only uses historical time series data to make FOREX market

predictions, the fundamental analysis considers various variables, including the company's and the nation's economic and industrial conditions [5]. Algorithmic trading refers to trading in which automated programmed algorithms implement orders instead of human traders. Algorithmic trading is utilized by hedge funds, pension funds, and other financial institutions [6], [7]. A lot of work has been put in by both academics and trading companies to find possible factors that could lead to much higher profits [8]. Numerous studies have attempted to forecast the movement of the FOREX market. The most crucial decision in FOREX is predicting the direction of currency price movement. Accurately forecasting currency prices can yield several advantages for traders and vice versa. In recent years, the academic community has made a lot of effort to develop machine learning models for FOREX market prediction.

On the other hand, numerous verifiable study types have been undertaken to understand and anticipate currency patterns in the FOREX market using machine learning algorithms. Generally, many methods are categorized into three categories: machine learning models, deep learning models, and hybrid forms.

Machine learning algorithms include Random Forest, Support Vector Machine, XGBoost, etc. Deep learning-based methods demonstrate how advanced neural models can significantly enhance prediction results. Like statistical methods, these methods require knowledge of effective signals to be utilized as input. The researcher utilized deep learning methods such as RNN, LSTM, CNN, GRU, and Transformers [2]. Long short-term memory (LSTM), a recurrent neural network (RNN), excels at modeling temporal patterns and is commonly employed in various tasks involving time series problems. CNNs are used to analyze price patterns by utilizing images of financial data as input [9]. This study attempts to answer the following questions: What normalization method improves FOREX prediction accuracy? What is the percentage of improvement on different dataset results? Which model, when used with the FOREX normalization function, gives the best results? Therefore, the objectives of this study are to utilize the FNF method with various machine and deep learning models and to compare the proposed model with two baseline models. Then, show the percentage of error reduction made by FNF.

This paper's primary contributions are summed up as follows:

- FOREX Normalization Function FNF is proposed to predict FOREX prices. FNF generates 84 different normalized features.
- FNF and Convolutional Neural Networks FNF-CNN are used in the first model 1 dimension convolution layer applied on FNF features to predict the close price of the next 1,3,7,15 days.
- FNF and Support Vector Machine FNF-SVR are used in the second model. Four kernels are applied on FNF features to predict the close price of the next 1,3,7,15 days. The forecasts are set for a one-day timeframe. Applying the previous models to 9 different datasets improved the results by an average between 0.5% and 58% of MAE.

The rest of this research is organized in the following manner: Section II reviews the related work, Section III provides an overview of the key scientific concepts, and Section IV describes the proposed models and their architectures, used datasets, evaluation metrics, and training configuration. Section V contains the results of the proposed models compared to baselines, discussion and ablation study. Finally, the paper concludes in Section VI.

## II. RELATED WORK

Various methods have been used in previous years to forecast the FOREX market. Numerous approaches have been attempted, mostly based on Artificial Intelligence principles. Some methods contain just one processing technique, while others combine two or more techniques. Researchers have used a variety of linear and nonlinear models for FOREX forecasting. Naive models such as Exponential smoothing, Autoregressive Moving Average model (ARMA), and Autoregressive Conditional Heteroskedasticity models (ARCH) and their variants (GARCH, EGARCH, etc.) are some of the most often used strategies for modeling volatility in time series [10]. Using machine learning methods like artificial neural networks (ANN) has been the subject of extensive study in recent years. The outstanding quality of ANNs used for time series forecasting problems is their innate capacity for nonlinear modeling without any assumption regarding the statistical distribution being invalid based on the observations. The most popular is the multi-layer perceptron (MLP), which has one hidden layer. Support vector machines (SVM), initially designed to address classification issues, are currently used for time series forecasting. Least-square SVM (LS-SVM) and Dynamic Least-square SVM are two common SVM models for forecasting time series [11]. Some researchers have applied deep learning models to predict FOREX prices. Hee Kueh and Leonard have proposed a comprehensive intelligent system for automated FOREX trading. The algorithm utilizes an ensemble methodology to make decisions; each strategy preprocesses technical data in order to produce a distinct buy or sell signal. The ensemble model takes all the signals from the different strategies and uses majority voting logic to decide what to do next [12]. The problem with this system is its low accuracy, and it was tested only on the EUR/USD dataset. Pornwattanavichai and Maneeroj [13] proposed a cascading model for the FOREX market. They made forecasts using

Fundamental Data and Technical indications based on BERT. They used the EUR/USD dataset from February 3, 2003, through February 28, 2020, which included 4,455 days. However, this system has limitations; it has not been tested on many datasets. Junior and Appiahene [14] developed a conceptual framework centered on a FOREX forecasting module that uses the Hurst test to determine whether a time series is predictable. They then applied a two-layer stacked LSTM architecture and correlation analysis to multiple currency datasets, including EUR/AUD, AUD/JPY, and AUD/USD. However, the problem with this framework is that it is not generalized to many currency pairs or window times. Dash S. and Sahu [15] employed a Deep Predictive Coding Network Optimized with a Reptile Search Algorithm for short-term forecasting over three days to forecast exchange rates of the CHF/INR, USD/EUR, and AUD/JPY currency pairs. However, this system produces a high mean absolute error value and does not support long-horizon forecasting.

Salman and Saeed U proposed the FLF-LSTM model to predict EUR/USD prices. They enhanced prediction using a custom loss function named FLF with a single LSTM and different activation functions [16]. Areej and Mohamed [17] used RBF, MLP, and SVM algorithms as classifiers to predict the direction of the price and compared them based on percentage classification performance. Ikhagvadorj and Tsenduren [18] proposed a framework consisting of seven neural networks with different activation functions. The outputs of these neural networks are concatenated and then fed into the softmax layer to produce probabilities or importance weights for each neural network. This model uses extended Min-Max normalization for financial time series data. Haixu and Jiehui [19] used the Autoformer for long-term FOREX price prediction at different time steps (96, 192, 336, 720). Autoformer is a variation of the transformer that uses Auto-Correlation instead of self-attention. Auto-correlation focuses on the connections of sub-series among underlying periods, while self-attention focuses on the connection between time points.

In conclusion, previous research indicates low results, and the impact of preprocessing has not been researched in detail. Also, the datasets used were not diverse and did not contain a large number of values for the time horizon. This paper addresses the impact of data preprocessing and scaling to enhance price prediction by using FNF. Nine datasets are used in this research, and the horizon values are 1, 3, 7, and 15. FNF is used in this paper with different models. The proposed models outperform the baselines.

## III. BACKGROUND

In this section, the necessary context for introducing our method is presented. First, CNN, SVM, LSTM, and XGBoost are examined.

### A. CNN

The Convolutional Neural Network (CNN) has demonstrated remarkable advances in several domains associated with pattern recognition and image processing throughout the previous decade. One of the primary advantages

of CNNs is their ability to effectively decrease the parameter count within Artificial Neural Networks (ANNs) [20].

1) *Convolutional Neural Network Element*: To develop a comprehensive understanding of (CNNs), it is necessary to examine their fundamental components. The input layer receives input and transfers it to the convolution layer[21]. The parameters of a convolutional neural network are arranged into an array of three-dimensional structural units called kernels or filters. Let us assume that the filter's dimensions in the qth layer are  $F_q \times F_q \times d_q$  and  $L_q$  height of layer q,  $B_q$  width of layer q. The following equation defines the convolutional process from the q-th layer to the (q + 1)th layer [22].

$$h_{ijp}^{(q+1)} = \sum_{r=1}^{F_q} \sum_{s=1}^{F_q} \sum_{k=1}^{d_q} w_{rsk}^{(p,q)} h_{i+r-1, j+s-1, k}^{(q)} \quad (1)$$

$$\forall i \in \{1 \dots, L_q - F_q + 1\}$$

$$\forall j \in \{1 \dots, B_q - F_q + 1\}, \forall p \in \{1 \dots, d_{q+1}\}$$

The 3-dimensional tensor  $W(p,q) = w_{ijk}^{(p,q)}$  represents the parameters of the pth filter in the qth layer. The indices i, j, and k represent the positions along the filter's height, width, and depth. The qth layer's feature maps are represented by the three-dimensional tensor  $H(q) = h_{i,j,k}^{(q)}$  [22]

The pooling layer will then perform downsampling along the provided input's spatial dimension. Researchers use either max pooling or average pooling [23]. The fully-connected layer will perform the same functions observed in conventional artificial neural networks. It is also recommended that the Rectified Linear Unit (ReLU) activation function be employed between these layers to enhance performance [23]. The output layer generates the final prediction value [21]. A common CNN is shown in Fig. 1 [24].

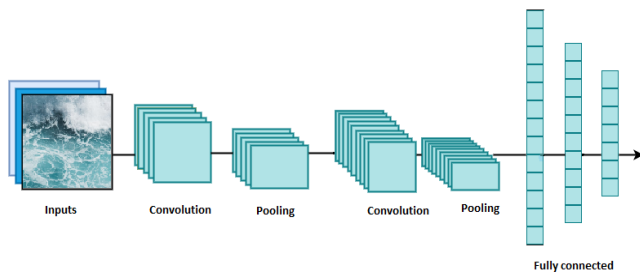


Fig. 1. Convolutional neural network architecture [24].

## B. SVM

Support Vector Machines (SVM) can be used for prediction when the outcome is binary, multinomial, or continuous. Classification is a common term for regression with Bernoulli outcomes in statistical learning. Multinomial regression is also known as multiclass classification. The procedure is called regression when the results are continuous [25].

1) *Support Vector Regression (SVR)*: In SVR, the  $\epsilon$ -insensitive loss function is minimized. If the loss is smaller than  $\epsilon$ , then the loss equals zero. The following equation (2) is utilized, known as the simple linear loss function:

$$L_\epsilon = \begin{cases} 0 & , \text{if } |y_i - f(x_i)| < \epsilon \\ |y_i - f(x_i)| - \epsilon & , \text{otherwise} \end{cases} \quad (2)$$

In support vector machines, kernels can achieve nonlinear regressions, such as radial basis function (RBF) and polynomial kernel [26]. The SVR can also be used with a linear kernel. The linear kernel equation is presented in (3), but nonlinear kernels are more flexible.

$$K(x_i, x_{i'}) = \sum_{j=1}^p x_{ij} x_{i'j} \quad (3)$$

$i \in \{1, \dots, n\}$  and  $i' \in \{1, \dots, n\}$ . We will call this inner product  $K \cdot P$  is the number of  $x$  variables, and  $j=1, \dots, p$ . The variables  $p$  and  $x$  are mapped into a higher-dimensional space by nonlinear kernels [25], [26].

RBF, a radial basis function, is the most common option for a nonlinear kernel. Equation 4 presents it[25].

$$K(x_i, x_{i'}) = \exp\{-\gamma \sum_{j=1}^p (x_{ij} - x_{i'j})^2\} \quad (4)$$

where  $\gamma > 0$  is an additional parameter for adaptability. When a test observation is quite far from a training observation, the exponent becomes strongly negative, and  $K(x_i, x_{i'})$  reaches zero. Sometimes, more variables, such as polynomials, must be added as a function of the original variables. Polynomial variables expand the number of regression variables. The following equation shows the polynomial equation [25] [26].

$$K(x_i, x_{i'}) = (\beta_0 + \gamma \sum_{j=1}^p x_{ij} x_{i'j})^d \quad (5)$$

Where  $\gamma > 0$  and  $\beta_0$  are additional parameters for adaptability,  $\beta_0$  "biases" the similarity metric for all samples. Applying this kernel implies adding polynomial powers of the  $x$  variables [25].

## C. LSTM

The Long Short-Term Memory (LSTM) is an architectural design of recurrent neural networks (RNNs) that aims to provide a more precise characterization of temporal sequences and their long-range dependencies in comparison to traditional RNNs [27]. In this section, LSTM architectures will be explored.

1) *LSTM architectures*: The Long Short-Term Memory (LSTM) model has specialized memory blocks within its recurrent hidden layer. The memory blocks in the network consist of memory cells that possess self-connections, enabling them to retain the temporal state of the network. These memory cells are accompanied by specialized multiplicative units known as gates, which regulate the information flow within the network. In the original architecture, each memory block comprised an input and output gate. The input gate regulates the influx of input activations into the memory cell, while the output gate regulates the transmission of cell activations from the current cell to the remaining components of the network [28] [29]. Subsequently, the forget gate was incorporated into the

memory block. Furthermore, the current Long Short-Term Memory architecture incorporates peephole connections that link the internal cells to the gates inside the same cell. This design allows the LSTM to acquire accurate output timing information [30].

An LSTM network calculates a mapping from an input sequence  $x = (x_1, \dots, x_t)$  to an output sequence  $y = (y_1, \dots, y_t)$  by calculating the network unit activations using the following equations iteratively from  $t = 1$  to  $T$ :

$$i_t = \sigma(W_{ix} x_t + W_{im} m_{t-1} + W_{ic} ct-1 + b_i) \quad (6)$$

$$f_t = \sigma(W_{fx} x_t + W_{fm} m_{t-1} + W_{fc} ct-1 + b_f) \quad (7)$$

$$c_t = f_t \odot ct-1 + i_t \odot g(W_{cx} x_t + W_{cm} m_{t-1} + b_c) \quad (8)$$

$$o_t = \sigma(W_{ox} x_t + W_{om} m_{t-1} + W_{oc} ct + b_o) \quad (9)$$

$$m_t = o_t \odot h(c_t) \quad (10)$$

$$y_t = \varphi(W_{ym} m_t + b_y) \quad (11)$$

The weight matrices  $W$  terms denote weight matrices,  $W_{ic}$ ,  $W_{fc}$ ,  $W_{oc}$  are represent diagonal weight matrices corresponding to peephole connections.  $\sigma$  represents the logistic sigmoid function, while  $b$  represents bias vectors ( $b_i$  represents the input gate bias vector). The input gate, forget gate, output gate, and cell activation vectors  $i$ ,  $f$ ,  $o$ , and  $c$  are identical in magnitude to the cell output activation vector  $m$ .  $\odot$  represents the element-wise product of the vectors  $g$  and  $h$ , where  $g$  and  $h$  represent the cell input and cell output activation functions, respectively [31].

#### D. XGBoost

XGBoost, an abbreviation for extreme gradient boosting, is well recognized as a common, robust, and efficient implementation of gradient boosting [32]. XGBoost is an ensemble model that efficiently implements decision trees to create a composite model with superior prediction performance compared to individual techniques employed alone. The output of XGBoost is calculated using the following equation:

$$\hat{Y}_i^T = \sum_{k=1}^T f_k(x_i) = \hat{y}_i^{T-1} + f_T(x_i) \quad (12)$$

where  $\hat{y}_i^{T-1}$  is the generated tree,  $f_T(x_i)$  is the newly created tree model, and  $T$  is the total number of tree models [33].

### IV. RESEARCH METHODOLOGY

In this section, two models are proposed: FNF-CNN and FNF-SVR. The first model uses FNF to normalize data and extract new features; those features become input for CNN. The second model also uses FNF and then a Support Vector Machine. Finally, four different kernels are used. The architecture of the models is explained in this section.

#### A. Proposed Approach

1) *Model I: FNF-CNN*: Multiple models and preprocessing methods are used in this paper to enhance results and present the impact of FNF. For example, the Moving Average and normalization of the close price are calculated using the following equation:

$$FNFc = MAw(close) - close \quad (13)$$

Here, FNFc represents the normalized close price, and MAw is the Moving Average of window size  $w$ . The first proposed model calculates the FNF equation for open, high, low, and close. The time windows range from 1 to 21 days. Then, all moving average features are normalized (FNF) and fed to a 1-dimensional convolutional neural network (CNN). MSE is the loss function, and ReLU is the activation function, as shown in Fig. 2.

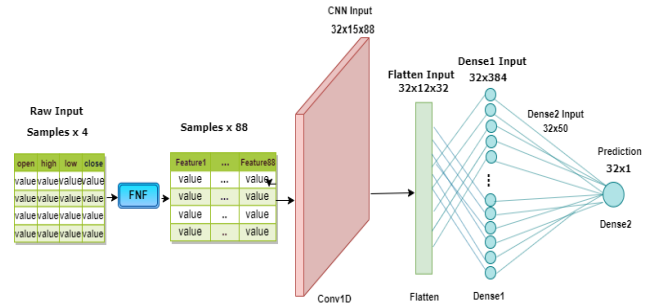


Fig. 2. The architecture of FNF-CNN to predict one day ahead.

A deep learning framework is used with extended Min-Max normalization [18] as the first baseline model. The raw data consists of open, high, low, and close prices from the previous time step, and the target is to predict close prices for the next day. The second baseline model is the Deep Predictive Coding Network Optimized with Reptile Search Algorithm (RSA-DPCN) [15]. RSA-DPCN is used to predict the future price of currency pairs for short-term time frames, such as three, seven, and 15 days ahead of the closing price of EURUSD, AUDJPY, and CHFINR [15]. Fig. 3 shows the proposed model to predict three days ahead of the closing price.

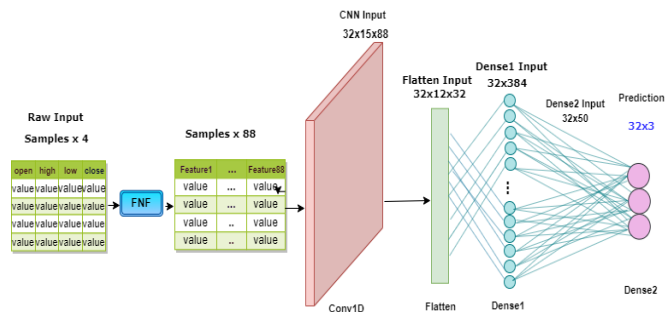


Fig. 3. This model is similar to the previous model in Fig. 2, but it is used to predict the price for the next three days.

2) *Model II: FNF-SVR*: The Moving Average from Windows 1 to 21 is calculated for all features in the second model. Then, all moving average features are scaled and fed to the Support Vector Regressor (SVR), as shown in Fig. 4. Different kernels are used to obtain the best results.

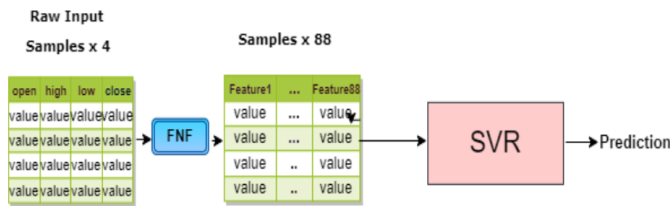


Fig. 4. Architecture of FNF-SVR.

XGBoost and LSTM with FNF are used to compare the results of different machine and deep learning models. For all models, MSE is used as a loss function. TensorFlow 2, Keras, Pandas, and XGBoost frameworks are used in the experiments.

B. Dataset

All models in this paper have been applied to the following datasets: The datasets in group 1 contain daily prices of (GBP/USD, EUR/USD, USD/CHF, USD/JPY, AUD/USD, USD/CAD) from 2000 to 2019. The datasets are partitioned into three parts: training (80%), validation (20%), and testing (the last 365 days)[18]. The dataset group 2 contains daily prices of AUD/JPY, CHF/INR, and EUR/USD from 2015 to 2020 [15].

C. Evaluation Metrics

Mean Absolute Error (MAE) is used to assess the model's performance and is expressed by Eq. (14) [34].

$$MAE = \frac{\sum_{t=1}^T |y_t - \hat{y}_t|}{T} \tag{14}$$

Here,  $y_t$  denotes the actual value of the price at period  $t$ ,  $\hat{y}_t$  Denotes the forecasted price value at period  $t$ , and  $T$  denotes the sample size. In other words, the MAE is the mean of the absolute difference between the predicted and actual prices throughout the test set. The actual price and the predicted price differ significantly when the MAE is high [34].

Mean Squared Error (MSE) is another method used to evaluate model performance. MSE is the average squared difference between the actual currency price and the values predicted by the model. Eq. (15) presents MSE [16].

$$MSE = \frac{\sum_{t=1}^T (y_t - \hat{y}_t)^2}{T} \tag{15}$$

D. Training Configuration

In the first group of datasets, when the FNF-CNN model is used, the time windows range from 1 to 21 days. This range is

used because increasing it above 21 leads to increased preprocessing without improving results and reducing the number below 22 reduces the accuracy of the results. The inputs to the first layer are (15x88), with nine previous time steps and 88 features. The number of previous steps was chosen as 9 because increasing it leads to increasing prediction error. The number of filters is 32, the kernel size is 4, and the next layer is flattened these hyperparameters chosen after many trails. These parameters were chosen after experimenting with the number of filters: 64 and 128. The experiments showed that using 32 filters leads to better results. Filter sizes of 8 and 16 were also tested, but they did not affect the improvement of the results. The last two layers are dense, and ReLU is used as the activation function. The FNF-LSTM model is a combination of FNF to extract features and LSTM to predict closing prices, and Adam is used as an optimizer. The learning rate and epochs of FNF-CNN and FNF-LSTM are 0.001 and 250, respectively. When we use a large number of epochs, overfitting occurs. Therefore, the appropriate value was equal to 250. The number of layers used is 5 and 3 in FNF-CNN and FNF-LSTM, respectively. The FNF- XGBoost model, which includes FNF and XGBoost Regressor, is used with 80 estimators and max depth=70, and the results look promising. All of these hyperparameters were selected based on many trials.

V. RESULTS AND DISCUSSION

A. Results

This section shows the results compared to the results of baseline one and baseline two. FNF-SVR-RBF refers to the FNF-Support Vector Regressor model with a radial basis function. The FNF-SVR-p2 and p3 refer to the FNF-Support Vector Regression model with a polynomial degree 2 and 3 kernels, respectively. Table I shows the MAE results of baseline 1 compared to the proposed models. Table II is the same as Table I but for MSE.

Using FNF generates 84 features, and this variety of features affects results by reducing error. CNN supports efficient feature learning. This resulted in the proposed models performing better than the baseline 1. Additionally, it is worth noting that the SVR model with FNF yielded the best results among all the models in the tables. Fig. 5 shows the price compared to the FNF-SVR, FNF-CNN, and FNF-LSTM prediction for EUR/USD and USD/JPY from dataset group 1. These graphs show that the model has learned the market trend and predicts the prices based on the actual trend.

TABLE I. MAE Results of Baseline1 compared with Proposed Models

Models	EUR/USD	USD/JPY	USD/CHF	GBP/USD	USD/CAD	AUD/USD
Baseline1	0.0043	0.4479	0.0037	0.0062	0.0046	0.0034
FNF-SVR-RBF	0.003141	<b>0.302782</b>	<b>0.002922</b>	0.005400	0.003571	0.002756
FNF-SVR-p3	0.012265	1.144953	0.191367	0.091945	0.043742	0.005579
FNF-SVR-p2	0.010982	0.654655	0.191753	0.049823	0.020536	0.00724
FNF-SVR- Linear	<b>0.003112</b>	0.307247	0.003063	<b>0.005268</b>	<b>0.003515</b>	<b>0.002612</b>
FNF-CNN	0.004465	0.359933	0.013730	0.006086	0.003576	0.004962
FNF-LSTM	0.008481	0.86004	0.070747	0.014158	0.005835	0.004061
FNF-XGBoost	0.005732	0.480439	0.004759	0.105812	0.005034	0.004356



TABLE II. MSE RESULTS OF BASELINE1 COMPARED WITH PROPOSED MODELS

Models	EUR/USD	USD/JPY	USD/CHF	GBP/USD	USD/CAD	AUD/USD
Baseline1	0.000032	0.359160	0.000035	0.000069	0.000036	0.000020
FNF-SVR-RBF	<b>0.000017</b>	<b>0.161011</b>	<b>0.000015</b>	<b>0.000050</b>	0.000022	<b>0.000013</b>
FNF-SVR-p3	0.000171	1.513236	0.036760	0.008682	0.001948	0.000047
FNF-SVR-p2	0.000137	0.561193	0.036908	0.002587	0.000446	0.000067
FNF-SVR- Linear	<b>0.000017</b>	0.166712	0.000016	0.000048	<b>0.000021</b>	0.000012
FNF-CNN	0.000028	0.203918	0.000269	0.000061	0.000022	0.000033
FNF-LSTM	0.000083	0.868147	0.005479	0.000249	0.000044	0.000023
FNF-XGBoost	0.000050	0.382606	0.000036	0.012168	0.000041	0.000031

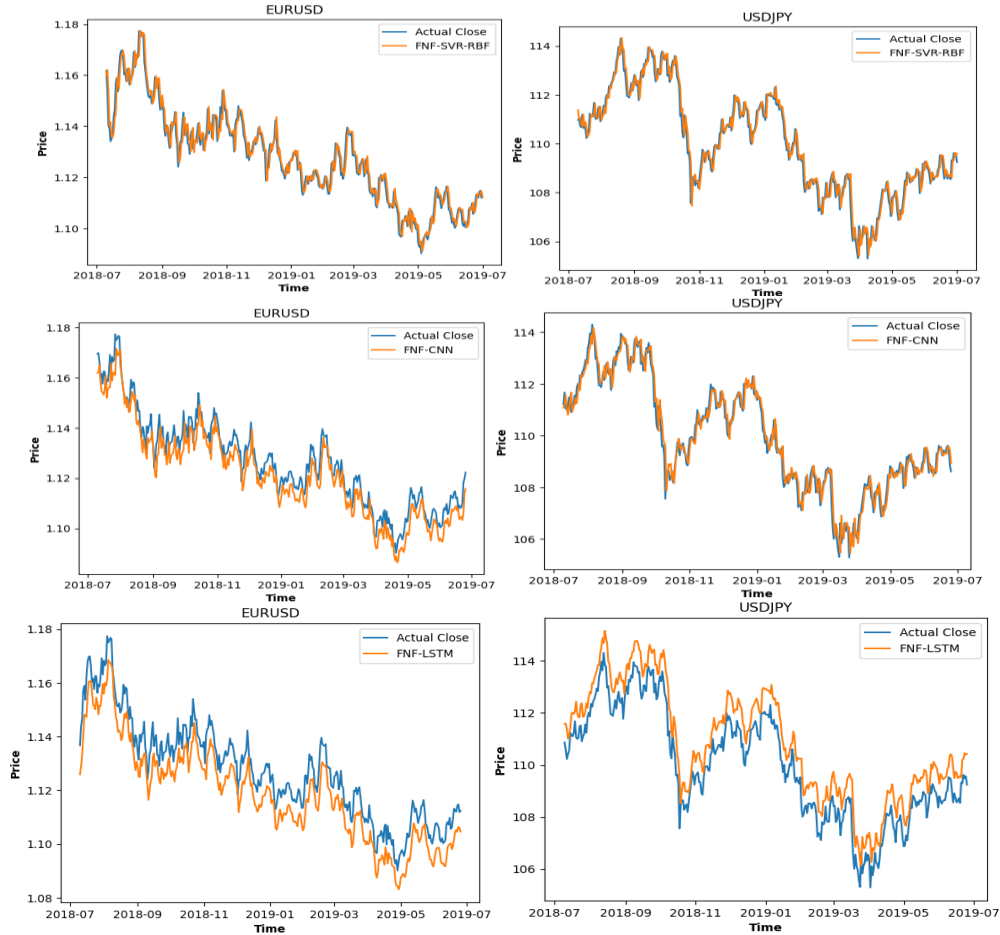


Fig. 5. On the left, the Actual closing price of EUR/USD compared to FNF-SVR-RBF, FNF-CNN, and FNF-LSTM predictions, and the right shows USD/JPY dataset results.

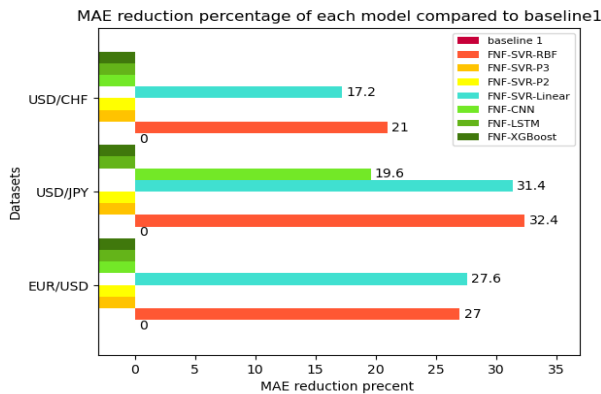


Fig. 6. The MAE reduction percentage of each model compared to baseline 1 applied on USD/CHF, USD/JPY, and EUR/USD datasets

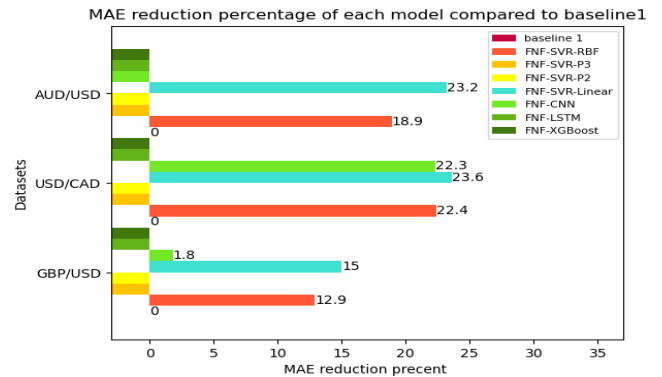


Fig. 7. The percentage reduction in MAE for each model, compared to baseline 1, applied to the AUD/USD, USD/CAD, and GBP/USD datasets.

In the previous results, using FNF with SVR and CNN outperformed baseline 1 in most datasets. Fig. 6 and 7 show the percentage reduction in MAE for each model compared to baseline 1. The preprocessing and feature extraction stages are the main reasons for error reduction.

Table III shows the results for the AUD/JPY dataset of baseline 2 compared to the proposed models. Table IV and Table V are the same as Table III but for the CHF/INR and

EUR/USD datasets. The 84 features extracted using FNF also enhance the result of multistep prediction, like 3, 7, and 15 days. From the results, the FNF-SVR and FNF-CNN models outperform baseline two on the AUD/JPY and CHF/INR datasets. Based on CHF/INR and AUD/JPY datasets, we find that the results of the FNF-SVR model are close to the results of the FNF-CNN model. FNF-LSTM also outperforms baseline 2 in most datasets because LSTM learns temporal dependency.

TABLE III. AUD/JPY RESULTS FOR 3,7,15 DAYS AHEAD PREDICTION

Horizon Metrics	3 days		7 days		15 days	
	MAE	MSE	MAE	MSE	MAE	MSE
Baseline 2	1.148	1.317904	1.258	1.582564	1.298	1.684804
FNF-SVR-RBF	0.489560	0.425518	0.644829	0.728458	0.902696	1.256283
FNF-SVR-p3	0.498748	0.421315	<b>0.630987</b>	<b>0.680119</b>	<b>0.852732</b>	<b>1.157992</b>
FNF-SVR-p2	<b>0.482397</b>	0.412050	0.806809	1.105423	0.872990	1.196919
FNF-SVR-Linear	0.539413	0.488757	0.797897	1.045897	1.286681	2.590383
FNF-CNN	0.495194	<b>0.40988</b>	0.847844	1.208709	1.267425	2.508938
FNF-LSTM	0.497183	0.426937	0.709538	0.863648	1.038998	1.656788
FNF-XGBoost	0.633611	0.693637	0.806809	1.105423	1.292051	2.972607

TABLE IV. CHF/INR RESULTS FOR 3,7,15 DAYS AHEAD PREDICTION

Horizon Metrics	3 days		7 days		15 days	
	MAE	MSE	MAE	MSE	MAE	MSE
Baseline 2	0.5148	0.265019	1.1148	1.242779	1.5148	2.294619
FNF-SVR-RBF	0.491143	0.367525	0.687234	0.748931	1.087740	1.849327
FNF-SVR-p3	1.281250	2.016307	1.136352	1.708272	1.032554	1.522211
FNF-SVR-p2	0.555829	0.451755	0.590223	0.548790	<b>0.782684</b>	<b>1.036713</b>
FNF-SVR-Linear	0.380511	0.241400	0.740709	0.843808	1.626221	4.194043
FNF-CNN	<b>0.368026</b>	<b>0.227829</b>	<b>0.540637</b>	<b>0.463773</b>	1.022167	1.716291
FNF-LSTM	1.122078	1.596323	0.728758	0.810617	1.140256	1.820308
FNF-XGBoost	2.640845	8.427829	3.065497	10.727666	3.330058	12.443699

TABLE V. EUR/USD RESULTS FOR 3,7,15 DAYS AHEAD PREDICTION

Horizon Metrics	3 days		7 days		15 days	
	MAE	MSE	MAE	MSE	MAE	MSE
Baseline 2	<b>0.001</b>	<b>0.000001</b>	<b>0.0017</b>	<b>0.0000289</b>	<b>0.0067</b>	<b>0.0000449</b>
FNF-SVR-RBF	0.005288	0.000046	0.007573	0.000100	0.011719	0.000245
FNF-SVR-p3	0.005756	0.000053	0.008000	0.000109	0.012118	0.000257
FNF-SVR-p2	0.473575	0.351067	0.007794	0.000104	0.011822	0.000249
FNF-SVR-Linear	0.005490	0.000049	0.007681	0.000102	0.011551	0.000240
FNF-CNN	0.006116	0.00006	0.009524	0.000142	0.01604	0.000409
FNF-LSTM	0.009236	0.000117	0.02616	0.000764	0.01023	0.000168
FNF-XGBoost	0.006295	0.000068	0.009578	0.000158	0.012472	0.000297

The following section shows the percentage of error reduction by many models compared to baseline2. There is a significant reduction in errors in the AUD/JPY and CHF/INR datasets. The improvement is observed for the 3, 7, and 15

horizons, as shown in Fig. 8. However, using the proposed models on the EUR/USD dataset does not enhance the results because the strength of trend and seasonality in the EUR/USD time series is low.

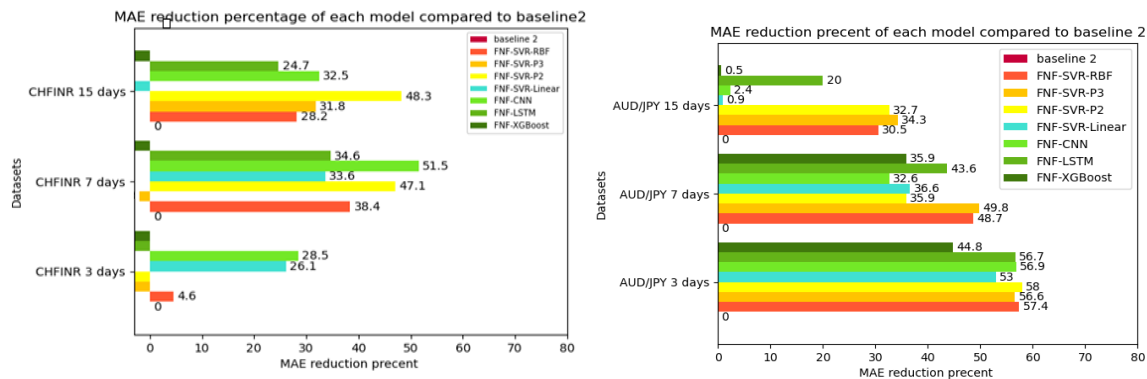


Fig. 8. The MAE reduction percentage of each model, compared to baseline 2, was applied to the AUD/JPY and CHF/INR datasets.

The results appear to vary between different datasets because each dataset has different statistical properties, and the strength of trend and seasonality differs from one time series to another.

**B. Discussion and Ablation Study**

This section presents an ablation study for the proposed models, FNF-SVR and FNF-CNN, and compares their results with and without FNF. Additionally, this section includes an ablation study applied to the FNF-LSTM model.

1) *FNF-CNN*: This study examines the impact of using FNF with CNN and the effect of each layer in CNN. The following tables demonstrate the impact of each layer in CNN. In the first baseline, the objective is to predict the next closing price of the next day. The USD/JPY Dataset from group 1, which was used in the first baseline, is shown in Table VI. Table VI illustrates the impact of removing specific layers and FNF on the USD/JPY dataset from group 1. Numbers in the header of the table refer to different model element

combinations. The Conv1D refers to the one-dimensional convolution layer, and MaxPooling1D refers to the one-dimensional max pooling layer. The results of the proposed model outperform all other compared components of the models. Tables VII and VIII present the results of USD/CHF and AUD/USD datasets. In most datasets of the first baseline, the FNF-CNN model outperforms other candidate CNN components. If we look at the results of the second column (FNF-CNN) and fifth column (model element combinations 3), which shows the results of the model with FNF and without it, the impact of scaling and feature extraction enhances results in most datasets. Using FNF with the CNN model improved the results by 12.3% and 26.0% in terms of MAE for the datasets USD/JPY and AUD/USD, respectively. These tables also confirm that the layers used in FNF-CNN are the ones that generally produce the best results on different datasets.

TABLE VI. FNF-CNN ABLATION STUDY APPLIED ON USD/JPY FROM GROUP 1 DATASET

Component	FNF-CNN	1	2	3	4	5
Conv1D	✓	✓	✓	✓	✓	✓
MaxPooling1D		✓	✓		✓	
Flatten	✓	✓	✓	✓	✓	✓
dense1	✓	✓	✓	✓		
dense2	✓	✓	✓	✓	✓	✓
FNF	✓	✓				
MAE	<b>0.359933</b>	0.374199	0.611925	0.410345	0.747184	0.459767
MSE	<b>0.203918</b>	0.225625	0.512484	0.267927	0.856830	0.307425

TABLE VII. FNF-CNN ABLATION STUDY APPLIED ON USD/CHF FROM GROUP 1 DATASET

Component	FNF-CNN	1	2	3	4	5
Conv1D	✓	✓	✓	✓	✓	✓
MaxPooling1D		✓	✓		✓	
Flatten	✓	✓	✓	✓	✓	✓
dense1	✓	✓	✓	✓		
dense2	✓	✓	✓	✓	✓	✓
FNF	✓	✓				
MAE	<b>0.013730</b>	0.019878	0.023732	0.012414	0.024553	0.036576
MSE	<b>0.000269</b>	0.000669	0.000616	0.000206	0.000643	0.001400

TABLE VIII. FNF-CNN ABLATION STUDY APPLIED ON AUD/USD FROM GROUP 1 DATASET

Component	FNF-CNN	1	2	3	4	5
Conv1D	✓	✓	✓	✓	✓	✓
MaxPooling1D		✓	✓		✓	
Flatten	✓	✓	✓	✓	✓	✓
dense1	✓	✓	✓	✓		
dense2	✓	✓	✓	✓	✓	✓
FNF	✓	✓				
MAE	0.004962	<b>0.003877</b>	0.009933	0.006707	0.009564	0.008913
MSE	0.000033	<b>0.000023</b>	0.000116	0.000057	0.000106	0.000092

The second baseline target predicts the closing price of 3, 7, and 15 days ahead. Tables IX, X, and XI show the results of the CNN ablation study on the CHF/INR dataset from group 2. The target is to predict the closing price for the next 3, 7, and 15 days. The results show that using FNF improves performance compared to not using it because FNF generates

many features that enhance prediction accuracy. Additionally, these tables validate that the FNF-CNN layers are those that typically yield the best results across a variety of horizons. Using FNF with the CNN model improved the results by 51.1%, 16.0%, and 17.0% MAE for the datasets CHF/INR when horizons equal 3, 7, and 15, respectively.



TABLE IX. FNF-CNN ABLATION STUDY APPLIED ON CHF/INR FROM GROUP 2 DATASETS TO PREDICT 3 DAYS

Component	FNF-CNN	1	2	3	4	5
Conv1D	✓	✓	✓	✓	✓	✓
MaxPooling1D		✓	✓		✓	
Flatten	✓	✓	✓	✓	✓	✓
dense1	✓	✓	✓	✓		
dense2	✓	✓	✓	✓	✓	✓
FNF	✓	✓				
MAE	0.368026	<b>0.364406</b>	0.748122	0.753149	0.884571	27.391575
MSE	0.227829	<b>0.213372</b>	0.792341	0.791035	1.11417	2195.586932

TABLE X. FNF-CNN ABLATION STUDY APPLIED ON CHF/INR FROM GROUP 2 DATASETS TO PREDICT 7 DAYS

Components	FNF-CNN	1	2	3	4	5
Conv1D	✓	✓	✓	✓	✓	✓
MaxPooling1D		✓	✓	✓	✓	
Flatten	✓	✓	✓	✓	✓	✓
dense1	✓	✓		✓		
dense2	✓	✓	✓	✓	✓	✓
scaling	✓	✓	✓			
MAE	<b>0.540637</b>	0.622345	0.707641	0.643276	0.624963	12.457177
MSE	<b>0.463773</b>	0.649072	0.786002	0.677533	0.659157	941.73922

TABLE XI. FNF-CNN ABLATION STUDY APPLIED ON CHF/INR FROM GROUP 2 DATASETS TO PREDICT 15 DAYS

Component	FNF-CNN	1	2	3	4	5
Conv1D	✓	✓	✓	✓	✓	✓
MaxPooling1D		✓	✓	✓	✓	
Flatten	✓	✓	✓	✓	✓	✓
dense1	✓	✓		✓		
dense2	✓	✓	✓	✓	✓	✓
scaling	✓	✓	✓			
MAE	<b>0.735953</b>	1.013976	0.777572	0.887162	0.868603	27.690644
MSE	<b>0.917616</b>	1.69364	0.956507	1.195118	1.157751	2204.428353

2) *Impact of Kernels and FNF on the SVR model:* This section presents the impact of FNF with different SVR kernels. The USD/CHF and AUD/USD datasets from group 1 are used to predict the next day's closing price. The FNF results outperform those without it, as shown in Fig. 9. In the second baseline, the target is to predict the closing price in the next 3, 7, and 15 days; the MultiOutputRegressor from the Keras library was used to do this.

To discuss the previous results in detail, we will explain the percentage of improvement in the results for each time series separately. Applying FNF-SVR on the USD/JPY dataset from group 1 enhances the results by 0.2%, 0.7%, 0.5%, and 2.6% of MAE when using RBF, Poly degree 3, Poly degree 2, and linear kernels, respectively. Using FNF and SVR on the AUD/USD dataset from group 1 enhances the results by 4.2%, 76.1%, and 32.2% of MAE when using RBF, Poly degree 3, and Poly degree 2 kernels, respectively. When we apply the

SVR-FNF model on the CHF/INR dataset from group 2 to predict the next three days, the results enhance by 6.7%, 99.7%, and 29.5% of MAE when using RBF, Poly degree 3, and Poly degree 2 kernels, respectively. When we applied the previous experiment on the same dataset but with a horizon equal to 7 days, the results enhanced by 86.7%, 99.7%, and 52.0% of MAE. However, when the horizon is equal to 15 days, the results enhanced by 84.3%, 99.8%, and 34.6%, respectively.

3) *FNF-LSTM:* In this section, the FNF-LSTM ablation study is presented. The following Datasets, USD/JPY, USD/CHF, and AUD/USD from group 1, are used and displayed in Fig. 10. This figure shows MAE is reduced when FNF is used. Fig. 11 shows the results of the same experiment on the CHF/INR and AUD/JPY datasets from group 2. The results demonstrate that the use of FNF enhances prediction in most datasets.

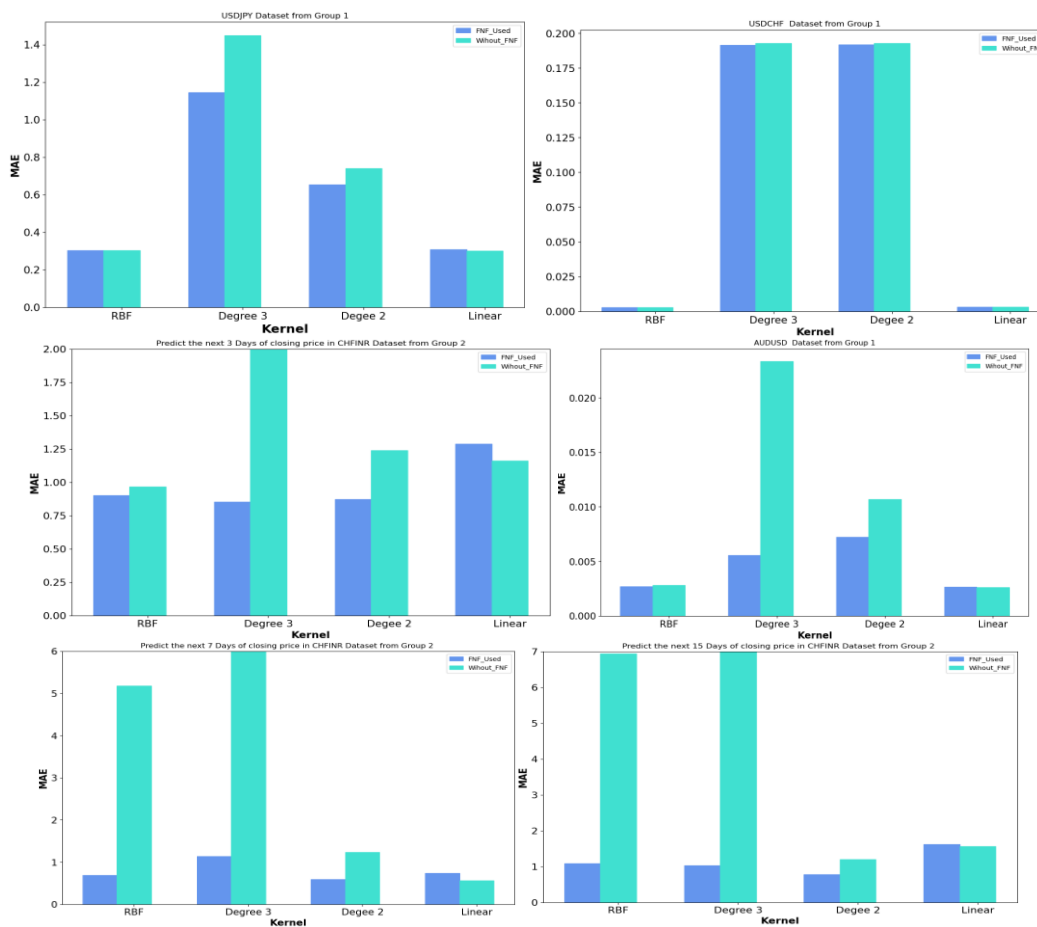


Fig. 9. (a) Impact of FNN and SVR kernels applied on USD/JPY, USD/CHF, and AUD/USD datasets from group 1 at the top and middle left of this figure. (b) The last three charts show the effect of FNN and Kernels on the CHF/INR dataset from Group 2 to predict the next 3, 7, and 15 days.

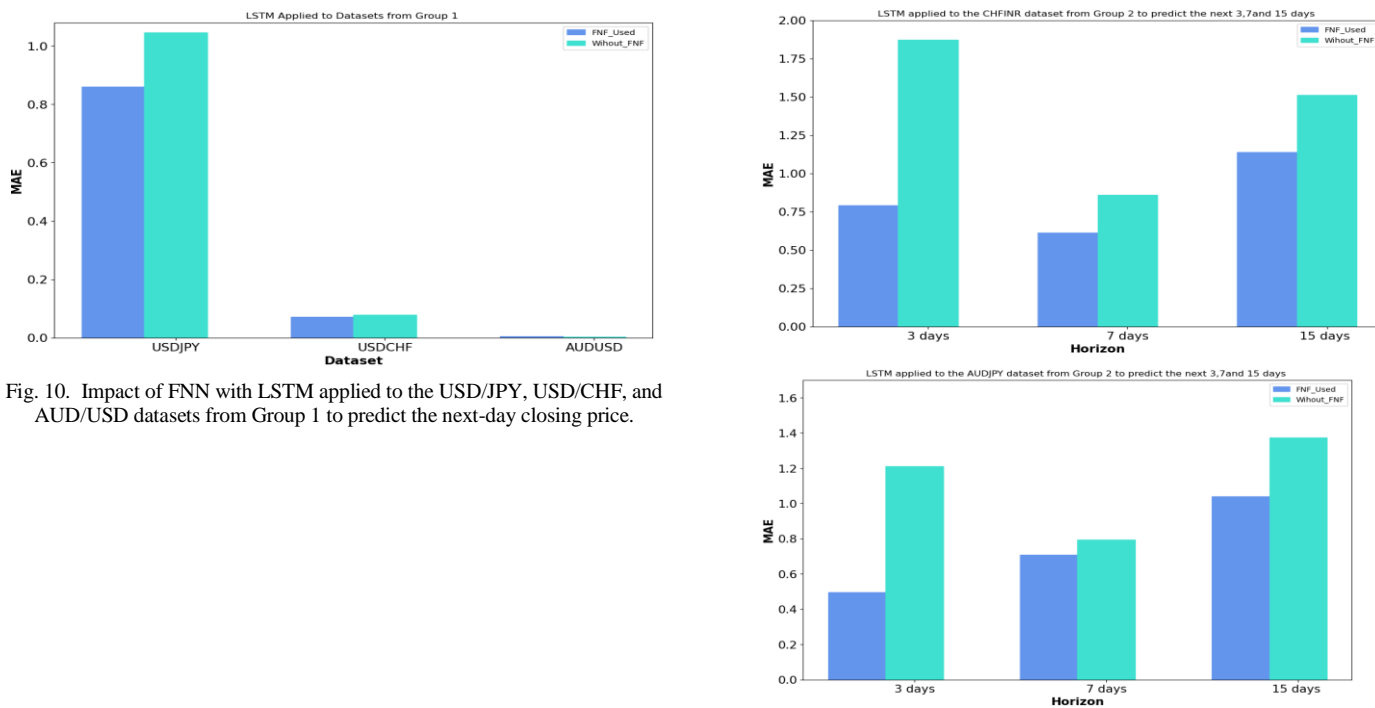


Fig. 10. Impact of FNN with LSTM applied to the USD/JPY, USD/CHF, and AUD/USD datasets from Group 1 to predict the next-day closing price.

Fig. 11. FNN-LSTM Ablation Study applied to CHF/INR and AUD/JPY Datasets from Group 2 to predict the next 3, 7, and 15 days.

Because every dataset has unique statistical characteristics, and each time series has a variable strength of trend and seasonality, the results appear to differ between them. LSTM can learn temporal dependencies, but when we use FNF-LSTM on the USD/JPY and USD/CHF datasets from group 1, the results are enhanced by 17.8% and 9.5%, respectively, while the results of AUD/USD are not enhanced. Applying FNF-LSTM on the CHF/INR dataset from group 2 to predict the next 3, 7, and 15 days enhances results by 57.8%, 28.7%, and 24.6% of MAE, respectively. Applying FNF-LSTM on the AU/DJPY dataset from group 2 to predict the next 3, 7, and 15 days enhances results by 58.9%, 10.6%, and 24.4% of MAE, respectively.

## VI. CONCLUSION

This paper proposed a FOREX normalization function used as a preprocessing method. This function is used with a machine learning model (SVR) and deep learning model(CNN) to enhance FOREX price prediction. Moving Averages and Scaling on raw data are essential steps to minimize error. Nine FOREX datasets are used in different horizons (1, 3, 7, and 15 days). Mean Absolute Error and Mean Squared Error are used to evaluate all models. The best performance results come from FNF-SVR and FNF-CNN. In this research, we compare different models with FNF and without it. The comparison between the proposed models and the two baseline models shows that our proposed models outperform the baseline models. The development of these proposed models is still in its early stages. Since the models present an exciting and potentially successful research topic, many enhancements must be investigated. The importance of this study is that it reduced the prediction error, which researchers can use this study to build decision support systems used in automated trading. Traders can also use its results to help make decisions to buy and sell currencies.

The limitation of this study is that we did not test the presented models on long-term prediction and did not test them on more datasets. In future work, the same proposed models will be used with different activation functions to find enhanced activation functions for FOREX and train the model using more datasets with varying time frames.

## REFERENCES

- [1] M. Ozturk, I. H. Toroslu, and G. Fidan, "Heuristic based trading system on Forex data using technical indicator rules," *Appl Soft Comput*, vol. 43, pp. 170–186, 2016.
- [2] M. Ayitey Junior, P. Appiahene, O. Appiah, and C. N. Bombie, "Forex market forecasting using machine learning: Systematic Literature Review and meta-analysis," *J Big Data*, vol. 10, no. 1, Dec. 2023, doi: 10.1186/s40537-022-00676-2.
- [3] S. Masry, A. Dupuis, R. B. Olsen, and E. Tsang, "Time zone normalization of FX seasonality," *Quant Finance*, vol. 13, no. 7, pp. 1115–1123, 2013.
- [4] B. J. de Almeida, R. F. Neves, and N. Horta, "Combining Support Vector Machine with Genetic Algorithms to optimize investments in Forex markets with high leverage," *Appl Soft Comput*, vol. 64, pp. 596–613, 2018.
- [5] H. Jamali, Y. Chihab, I. García-Magariño, and O. Bencharef, "Hybrid Forex prediction model using multiple regression, simulated annealing, reinforcement learning and technical analysis," *IAES International Journal of Artificial Intelligence*, vol. 12, no. 2, pp. 892–911, Jun. 2023, doi: 10.11591/ijai.v12.i2.pp892-911.
- [6] H. H. Zahrah and J. Tirtawangsa, "Algorithmic Forex Trading Using Q-learning," in *IFIP International Conference on Artificial Intelligence Applications and Innovations*, Springer, 2023, pp. 24–35.
- [7] A. Shavandi and M. Khedmati, "A multi-agent deep reinforcement learning framework for algorithmic trading in financial markets," *Expert Syst Appl*, vol. 208, p. 118124, 2022.
- [8] B. Huang, Y. Huan, L. Da Xu, L. Zheng, and Z. Zou, "Automated trading systems statistical and machine learning methods and hardware implementation: a survey," *Enterprise Information Systems*, vol. 13, no. 1. Taylor and Francis Ltd., pp. 132–144, Jan. 02, 2019. doi: 10.1080/17517575.2018.1493145.
- [9] A. H. Moghaddam and S. Momtazi, "Image processing meets time series analysis: Predicting Forex profitable technical pattern positions," *Appl Soft Comput*, vol. 108, p. 107460, 2021.
- [10] S. S. Chung and S. Zhang, "Volatility estimation using support vector machine: Applications to major foreign exchange rates," *Electronic Journal of Applied Statistical Analysis*, vol. 10, no. 2, pp. 499–511, 2017.
- [11] X.-L. Gong, X.-H. Liu, X. Xiong, and X.-T. Zhuang, "Forecasting stock volatility process using improved least square support vector machine approach," *Soft comput*, vol. 23, pp. 11867–11881, 2019.
- [12] L. K. Y. Loh, H. K. Kueh, N. J. Parikh, H. Chan, N. J. H. Ho, and M. C. H. Chua, "An Ensembling Architecture Incorporating Machine Learning Models and Genetic Algorithm Optimization for Forex Trading," *FinTech*, vol. 1, no. 2, pp. 100–124, Mar. 2022, doi: 10.3390/fintech1020008.
- [13] A. Pornwattanavichai, S. Maneeroj, and S. Boonsiri, "BERTFOREX: Cascading Model for Forex Market Forecasting Using Fundamental and Technical Indicator Data Based on BERT," *IEEE Access*, vol. 10, pp. 23425–23437, 2022, doi: 10.1109/ACCESS.2022.3152152.
- [14] M. Ayitey Junior, P. Appiahene, and O. Appiah, "Forex market forecasting with two-layer stacked Long Short-Term Memory neural network (LSTM) and correlation analysis," *Journal of Electrical Systems and Information Technology*, vol. 9, no. 1, Dec. 2022, doi: 10.1186/s43067-022-00054-1.
- [15] S. Dash et al., "A Novel Algorithmic Forex Trade and Trend Analysis Framework Based on Deep Predictive Coding Network Optimized with Reptile Search Algorithm," *Axioms*, vol. 11, no. 8, Aug. 2022, doi: 10.3390/axioms11080396.
- [16] S. Ahmed, S. U. Hassan, N. R. Aljohani, and R. Nawaz, "FLF-LSTM: A novel prediction system using Forex Loss Function," *Applied Soft Computing Journal*, vol. 97, Dec. 2020, doi: 10.1016/j.asoc.2020.106780.
- [17] A. A. Baasher and M. W. Fakhr, "Forex trend classification using machine learning techniques," in *Proceedings of the 11th WSEAS international conference on Applied computer science, World Scientific and Engineering Academy and Society (WSEAS) Stevens Point ...*, 2011, pp. 41–47.
- [18] L. Munkhdalai, T. Munkhdalai, K. H. Park, H. G. Lee, M. Li, and K. H. Ryu, "Mixture of Activation Functions with Extended Min-Max Normalization for Forex Market Prediction," *IEEE Access*, vol. 7, pp. 183680–183691, 2019, doi: 10.1109/ACCESS.2019.2959789.
- [19] H. Wu, J. Xu, J. Wang, and M. Long, "Autoformer: Decomposition Transformers with Auto-Correlation for Long-Term Series Forecasting," Jun. 2021, [Online]. Available: <http://arxiv.org/abs/2106.13008>
- [20] O. Bayat, S. Aljawarneh, H. F. Carlak, International Association of Researchers, Institute of Electrical and Electronics Engineers, and Akdeniz Üniversitesi, *Proceedings of 2017 International Conference on Engineering & Technology (ICET'2017)*: Akdeniz University, Antalya, Turkey, 21-23 August, 2017.
- [21] R. Tekchandani and N. Kumar, *Applied Deep Learning: Design and implement your own Neural Networks to solve real-world problems (English Edition)*. BPB Publications, 2023.
- [22] AGGARWAL, Charu C., et al. *Neural networks and deep learning*. "Springer", 2018, 10.978: 3.
- [23] K. O'Shea and R. Nash, "An Introduction to Convolutional Neural Networks," Nov. 2015, [Online]. Available: <http://arxiv.org/abs/1511.08458>

- [24] A. Géron, Hands-on machine learning with Scikit-Learn, Keras, and TensorFlow. “O’Reilly Media, Inc.,” 2022.
- [25] N. Guenther and M. Schonlau, “Support vector machines,” 2016.
- [26] wwwit-ebooksinfo, Introduction to Machine Learning with Python. 2016. [Online]. Available: <http://safaribooksonline.com>
- [27] H. Sak, A. Senior, and F. Beaufays, “Long Short-Term Memory Based Recurrent Neural Network Architectures for Large Vocabulary Speech Recognition,” Feb. 2014, [Online]. Available: <http://arxiv.org/abs/1402.1128>
- [28] F. A. Gers, J. Schmidhuber, and F. Cummins, “Learning to forget: Continual prediction with LSTM,” Neural Comput, vol. 12, no. 10, pp. 2451–2471, 2000.
- [29] D. M. Shprekher, G. I. Babokin, and E. B. Kolesnikov, “Application of neural networks for prediction of insulation condition in networks with isolated neutral,” in 2019 International Russian Automation Conference (RusAutoCon), IEEE, 2019, pp. 1–6.
- [30] F. A. Gers, N. N. Schraudolph, and J. Schmidhuber, “Learning Precise Timing with LSTM Recurrent Networks,” 2002. [Online]. Available: [www.idsia.ch](http://www.idsia.ch)
- [31] H. H. Sak, A. Senior, and B. Google, “Long Short-Term Memory Recurrent Neural Network Architectures for Large Scale Acoustic Modeling.”
- [32] SERRANO, Luis. Grokking Machine Learning. "Simon and Schuster", 2021.
- [33] S. Ben Jabeur, S. Mefteh-Wali, and J. L. Viviani, “Forecasting gold price with the XGBoost algorithm and SHAP interaction values,” Ann Oper Res, 2021, doi: 10.1007/s10479-021-04187-w.
- [34] T. Chai and R. R. Draxler, “Root mean square error (RMSE) or mean absolute error (MAE)?,” Geosci. Model Dev. Discuss, vol. 7, pp. 1525–1534, 2014, doi: 10.5194/gmdd-7-1525-2014.