Journal of Data Analysis and Information Processing, 2023, 11, 103-120
https://www.scirp.org/journal/jdaip
ISSN Online: 2327-7203
ISSN Print: 2327-7211

# A Hybrid Neural Network Model Based on Transfer Learning for Forecasting Forex Market

**Salum Hassan Faru[1], Anthony Waititu[2], Lawrence Nderu[3]**

[1]Department of Mathematics, The Pan African University, Institute for Basic Sciences, Technology and Innovation (PAUSTI), Nairobi, Kenya
[2]Department of Statistics and Actuarial Science, Jomo Kenyatta University of Agriculture and Technology, Juja, Kenya
[3]Department of Computing and Information Technology, Jomo Kenyatta University of Agriculture and Technology, Juja, Kenya
Email: hassan.salum@students.jkuat.ac.ke

**How to cite this paper:** Faru, S.H., Waititu, A. and Nderu, L. (2023) A Hybrid Neural Network Model Based on Transfer Learning for Forecasting Forex Market. *Journal of Data Analysis and Information Processing*, **11**, 103-120.
https://doi.org/10.4236/jdaip.2023.112007

**Received:** January 20, 2023
**Accepted:** March 27, 2023
**Published:** March 30, 2023



## Abstract

The forecasting research literature has developed greatly in recent years as a result of advances in information technology. Financial time-series tasks have made substantial use of machine learning and deep neural networks, but building a prediction model from scratch takes time and computational resources. Transfer learning is growing popular in tackling these constraints of training time and computational resources in several disciplines. This study proposes a hybrid base model for the financial time series prediction employing the recurrent neural network (RNN) and long-short term memory (LSTM) called RNN-LSTM. We used random search to fine-tune the hyperparameters and compared our proposed model to the RNN and LSTM base models and evaluate using the RMSE, MAE, and MAPE metrics. When forecasting Forex currency pairs GBP/USD, USD/ZAR, and AUD/NZD our proposed base model for transfer learning outperforms RNN and LSTM base model with root mean squared errors of 0.007656, 0.165250, and 0.001730 respectively.

## Keywords

Deep Learning, Transfer Learning, Time Series Analysis, RNN, LSTM

## 1. Introduction

Foreign Exchange (dubbed Forex or FX) is a global currency trading market considered the most liquid global financial market. According to the Bank for International Settlements, trading in Forex markets averaged $5.3 trillion daily in April 2013 [1]. Due to its anonymous nature and increased instability of price rates, the Forex market is deemed very complicated and volatile, resembled with the black box [2].

The Forex market is an example of a financial time series market where currency exchange rates are traded in pairs. These pairs are categorized into three groups, namely major, minor, and exotic currency pairs [3]. Predicting price time series in financial markets, which have a non-stationary nature, takes much work [4] [5]. They are dynamic, chaotic, noisy, and non-linear series that the market is prejudiced by the general economy, characteristics of the industries, politics, and even the psychology of investors [6] [7]. Researchers for forecasting the Forex market have proposed many different methods. In the late 1990s, popular ways were statistical methods [3]. The neural network was a revolutionary discovery for time series forecasting problems, and many other methodologies have arisen from this occasion [8] [9].

Due to the limitations of classic machine learning methods, researchers and data scientists have recently embraced the concept of transfer learning. Traditional machine learning models, such as RNN, LSTM, CNN, GRU, SVM, and others, necessitate training from scratch, which is computationally expensive and demands vast data to attain good performance [10]. They also use an isolated training strategy, in which each model is trained separately for a specific task without relying on prior information. Researchers are now using transfer learning to overcome these constraints; however, transfer learning in time series prediction problems has yet to be widespread [11].

Pre-trained models for time-series predictions still need to be improved, despite transfer learning's growing popularity in tackling these constraints of training time and computational resources in several disciplines like Natural Language Processing (NLP), image, video, and audio problems. Google's Tensor Flow Hub and NVIDIA contain several pre-trained models for various problem domains such as text, photo, video, and audio.

In recent years, the most popular Contract for Differences (CFDs) brokers have displayed standardized risk warnings, including the proportion of losses on a CFD provider's accounts held by retail investors, and the data shows that losing funds made up 54% to 83% of the total, with 76% being the average[1]. From this observation, despite having different prediction models, the percentage of profitable traders needs to be bigger!

This study proposes a hybrid model for the Forex market prediction employing the recurrent neural network (RNN) and long-short-term memory (LSTM) called RNN-LSTM. Based on transfer learning, the model is used to predict Forex market currency pairs as a pre-trained model for future work related to the time series problem. The study creates a dashboard to help users make the best trade selections and maximize their winning rates to reduce the number of unprofitable traders.

The rest of the paper is structured as follows; Section 2 designates the litera-

[1]https://www.financemagnates.com/forex/brokers/which-brokers-house-the-most-winning-clients-post-esma/.

ture review, and Section 3 materials and methods. Section 4 presents the experimental study and results analysis. Section 5 discusses of results, and Section 6 concludes the work.

## 2. Literature Review

As a result of the advancement of information technology, the forecasting research literature has considerably increased. The results reveal that neural network models outperform statistical models like ARIMA (Autoregressive integrated moving average), indicating its applicability for forecasting foreign exchange rates. For the Turkish TL/US dollar exchange rate series, [12] used both the ARIMA time series model and neural networks. The outcomes demonstrate the superiority of ANNs over statistical models, with the ANN approach outperforming the ARIMA time series model in terms of accuracy [9].

The Recurrent Neural Network (RNN) has been employed in most studies because it has the advantage of remembering some information about the sequence through hidden states. In contrast to a standard neural network, the input and output are entirely independent. However, [13] has observed the drawbacks of RNN and stated that gradient-based learning approaches take far too long, since the error vanishes or explodes as it propagates back. In a RNN, the issue of vanishing and exploding gradients has been addressed in various ways, and the LSTM is one of the most well-known.

A C-RNN forecasting technique based on deep RNN and CNN for Forex time series data was offered by [14]. Data-driven analysis was employed by the researcher to fully harness the Spatiotemporal properties of Forex time series data. The C-RNN foreign exchange time-series data forecast approach has increased applicability and accuracy compared to LSTM and CNN, according to an experimental comparison of the predicted strategy on the exchange rate data of nine major foreign exchange currencies.

The direction of the Forex market using LSTM was forecasted by [15]. The macroeconomic information and the technical indicator dataset were both examined by the researcher. The study created a hybrid model that integrated the two LSTMs that corresponded to the study's two datasets, and it was proven to be quite successful on actual data. The key motive for the researcher to use a hybrid was to avoid the shortcomings of LSTMs. When ME_LSTM and TI_LSTM models were executed separately, many transactions with wrong signals were generated, reducing the accuracy.

For predicting stock prices, [16] suggested a novel deep learning-based model. They constructed a hybrid model by merging the well-known LSTM and GRU neural network models. The model uses the LSTM output as the input to the GRU unit. They employed S&P 500 historical time series data and traditional evaluation criteria like MSE, MAPE, and so on for model validation.

Transfer learning is a method for encoding features from a model that has already been learned, preventing us from having to create a new model from scratch. A pre-trained model is often developed on a large datasets, and the weights gained from the trained model can be used with your custom neural network for any other similar application. These freshly constructed models can be used directly for task predictions or in training processes for related applications. This method reduces both the training time and the generalization error [17].

To forecast short-term stock price movement, [11] created a deep transfer with related stock information (DTRSI) model that blends transfer learning and a deep neural network. The researcher developed an LSTM base model and trained it using substantial stock data from the US and Korean markets. When the model parameters were tuned, it predicted the stock price of Hyundai Motor Co. with an average accuracy of 64.54 per cent and the stock price of Amazon Co. Inc with an average accuracy of 62.65 per cent.

Also, [17] suggested a methodology for the stock market prediction that combined transfer learning of data from industrial chains with deep learning algorithms such as multilayer perceptron (MLP), RNN, LSTM, and GRU. These algorithms were used to forecast the trend of 379 stock market indices in China. They discovered that RNNs are occasionally the best prediction option when dealing with detailed time-series data. For this reason, the MLP was selected for transfer learning stock market index prediction, and the maturity yield surpasses the buy-and-hold strategy.

Due to the drawbacks of RNN and limitations of LSTM, and to leverage the strength of RNN and LSTM, this study suggests the hybrid RNN-LSTM model used as the primary model for transfer learning time-series tasks. Despite the application of transfer learning, [11] studies show that the prediction performance of stock markets could be more impressive. Consequently, additional studies can be conducted to enhance the performance. To the best of our knowledge, no paper has used the hybrid model RNN-LSTM to pre-train the source domain for time series prediction problems, especially Forex market currency prediction using transfer learning, thus filling the gap.

## 3. Materials and Methods

A data science process is a series of instructions outlining how a person or a team should carry out a data science project. According to [18], the three most common processes used for data science projects are Cross Industry Standard Process for Data Mining (CRISP-DM), Knowledge Discovery in Databases (KDD), and Sample-Explore-Modify-Model-Assess (SEMMA).

The most often used framework for carrying out data science initiatives is CRISP-DM, which provides a structured and systematic approach to data mining projects, which leads to increased efficiency and success. KDD highlights the high-level applications of particular Data Mining techniques and refers to the general process of discovering knowledge in massive databases[2]. The CRISP-DM data science process was used in this study due to the nature of our work and the objectives.

[2]https://www.javatpoint.com/kdd-process-in-data-mining.

## 3.1. Data Description

Depending on the broker, the Forex market has more than 100 currency pairs. For instance, Exness broker[3] has seven major pairs, 26 minor pairs, and 67 exotic pairings. For training our hybrid base model, we employed six major currency pairs, seven minor currency pairs, and seven exotic currency pairs for 20 currency pairs. We used five currency pairs as our target currency pairs: one central currency pair, two minor currency pairs, and two exotic currency pairs, as given in Table 1.

Table 1. Currency pairs used.

| | Source data | |
|---|---|---|
| **Major** | **Minor** | **Exotic** |
| AUD/USD | AUD/CHF | AUD/JPY |
| EUR/USD | CAD/JPY | CHF/JPY |
| NZD/USD | EUR/CHF | EUR/NOK |
| USD/JPY | GBP/AUD | EUR/AUD |
| USD/CAD | GBP/NZD | CAD/HKD |
| USD/CHF | NZD/JPY | AUD/SGD |
| | AUD/CAD | ZAR/JPY |
| | **Target data** | |
| GBP/USD | CAD/CHF | USD/ZAR |
| | EUR/CAD | AUD/NZD |

## 3.2. Technical Indicators

Price movements follow trends, according to technical analysts. Those price changes often follow recognized patterns that can be partially attributed to market psychology based on the widely-held belief that market participants act the same when faced with analogous situations. We have also used the five most popular technical indicators to forecast the closing price for the future hour, including the stochastic oscillator, Bollinger band, relative strength index (RSI), and exponential moving averages (EMA).

## 3.3. Modeling

### 3.3.1. Recurrent Neural Networks (RNN) Model

According to [19], the RNN belongs to the class of neural networks that process sequential data. They accept a series of vectors $(x_1, x_2, \cdots, x_n)$ as input and generate a second series $(h_1, h_2, \cdots, h_n)$ that contains details of the input sequence at each step. In particular, RNNs use a recurrent hidden state, whose activation at each iteration depends on the activation at the previous iteration, to address variable-length sequences.

Figure 1 shows the architecture of RNN model. The implementation of updating recurrent hidden state $h_t^R$ is as follows:

[3]https://www.exness.com/forex/.

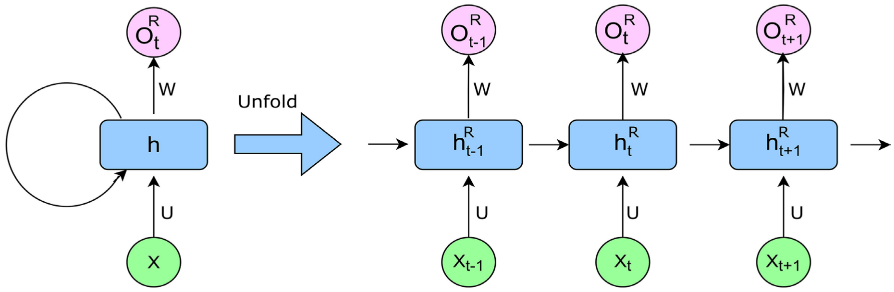



**Figure 1.** RNN model architecture.

$$h_t^R = g\left(W \times x_t + U \times h_{t-1}^R + b\right) \tag{1}$$

$$O_t^R = g\left(W^o \times h_t^R + b^o\right) \tag{2}$$

In this case, $g$ is a bounded and smooth function, such as a hyperbolic tangent or logistic function and $O_t^R$ is the output/predicted value from recurrent neural network. The network's input vector $x_t$, along with its prior hidden state $h_{t-1}^R$ and the bias $b$, determine its recurrent hidden state $h_t^R$ per time $t$ step. The weight matrices, $W$ and $U$, are filters that choose how much significance to assign based on the input at hand and the hidden state from the past. They generate an error, which is returned via back propagation and used to change the weights of their parameters until the error cannot be reduced further [20].

### 3.3.2. Long Short-Term Memory (LSTM) Model

The LSTM, which [21] proposed, is a specific type of recurrent network that addresses the problems of vanishing and exploding gradients by including memory units that enable the system to comprehend when to forget the earlier hidden states and when to revamp the hidden conditions with the latest information. In the literature, models with hidden units and different linkages within the memory unit have been put forth with remarkable practical success [20].

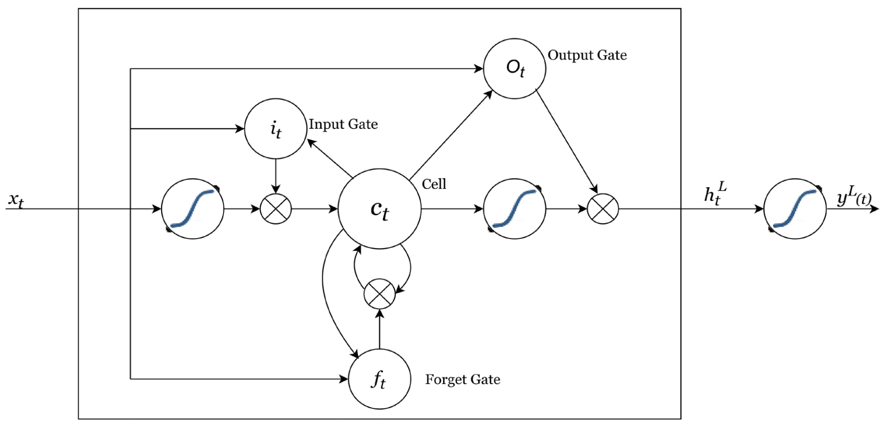



**Figure 2.** LSTM network.

The LSTM transition equations are as following [20]:

$$i_t = \sigma\left(W^i \times x_t + U^i \times h_{t-1}^L + b^i\right) \tag{3}$$

$$f_t = \sigma\left(W^f \times x_t + U^f \times h_{t-1}^L + b^f\right) \tag{4}$$

$$o_t = \sigma\left(W^o \times x_t + U^o \times h_{t-1}^L + b^o\right) \tag{5}$$

$$u_t = \tanh\left(W^u \times x_t + U^u \times h_{t-1}^L + b^u\right) \tag{6}$$

$$c_t = i_t \odot u_t + f_t \odot c_{t-1} \tag{7}$$

$$h_t^L = o_t \odot \tanh\left(c_t\right) \tag{8}$$

$$y^L\left(t\right) = \sigma\left(W^o \times h_t^L\right) \tag{9}$$

From Equation (3) to (9), $i_t$, $f_t$, and $o_t$ is the input, forget, and an output gate respectively, where $c_t$ denote a memory cell, an activation function is given by $u_t$, LSTM hidden state by $h_t^L$, and the predicted value is given by $y^L\left(t\right)$. Figure 2 illustrates the defaulting connection among these LSTM units.

In contrast to a standard recurrent unit, which simply replaces its scope per time step, the LSTM can decide whether to maintain the present memory with the aid of the additional gates. Intuitively, the output gate determines the amount of internal memory state accessible, while the input gate chooses what fresh information will be updated. How much of the previous memory cell is forgotten is then determined by the forget gate.

### 3.3.3. Proposed Hybrid RNN-LSTM Base Model

We provide thorough information about our proposed hybrid model in this section. The hybrid is made up of RNN and LSTM, as seen in Figure 3. As [3] stated, both are sophisticated neural network models that can outperform regression-based predictions in terms of accuracy.

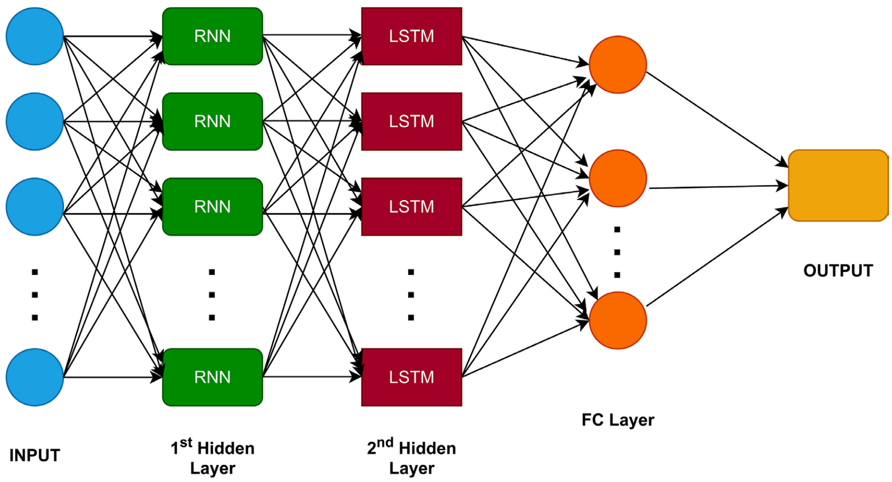



Figure 3. A hybrid RNN-LSTM base model architecture.

Figure 3 represents the architecture of hybrid base model where FC is fully connected layer. The primary logic behind employing these networks was that currency price prediction is a regression-based problem. To begin, we fed the data into RNN network, and obtain the output $O_t^R$ as shown in Equation (2).

To get the final prediction $\hat{y}_t$, we pass the output of the RNN layer to the LSTM layer.

Mathematically, the input in LSTM layer for our hybrid will change from $x_t$ to $O_t^R$ as shown below,

$$i_t^H = \sigma\left(W^i \times O_t^R + U^i \times h_{t-1}^L + b^i\right) \tag{10}$$

$$f_t^H = \sigma\left(W^f \times O_t^R + U^f \times h_{t-1}^L + b^f\right) \tag{11}$$

$$o_t^H = \sigma\left(W^o \times O_t^R + U^o \times h_{t-1}^L + b^o\right) \tag{12}$$

$$u_t^H = \tanh\left(W^u \times O_t^R + U^u \times h_{t-1}^L + b^u\right) \tag{13}$$

$$c_t^H = i_t^H \odot u_t^H + f_t^H \odot c_{t-1}^H \tag{14}$$

$$h_t^H = o_t^H \odot \tanh\left(c_t^H\right) \tag{15}$$

And lastly, the predicted price $\hat{y}_t$ is give by;

$$\hat{y}_t = \sigma\left(W^o \times h_t^H\right) \tag{16}$$

where $W^o$ is the weight matrix of the output gate and $h_t^H$ is the hidden state of Hybrid unit given by Equation (15).

### 3.3.4. Transfer Learning

Transfer learning can manage scenarios where domains and distributions differ, unlike typical machine learning and data mining techniques, which presume that training and testing data are from the same feature space and distribution. These features allow the model to make advantage of relevant source data and apply the underlying knowledge to the target job, resulting in increased performance [22].

Due to this benefit, in our study, we heavily rely on datasets of currency pairs in the Forex market for training the base model and learning knowledge from them before applying the knowledge to the target data (*i.e.*, where data from only the target currency pair is used). **Figure 4** depicts the transfer learning framework employing a hybrid base model for the source and destination domains.

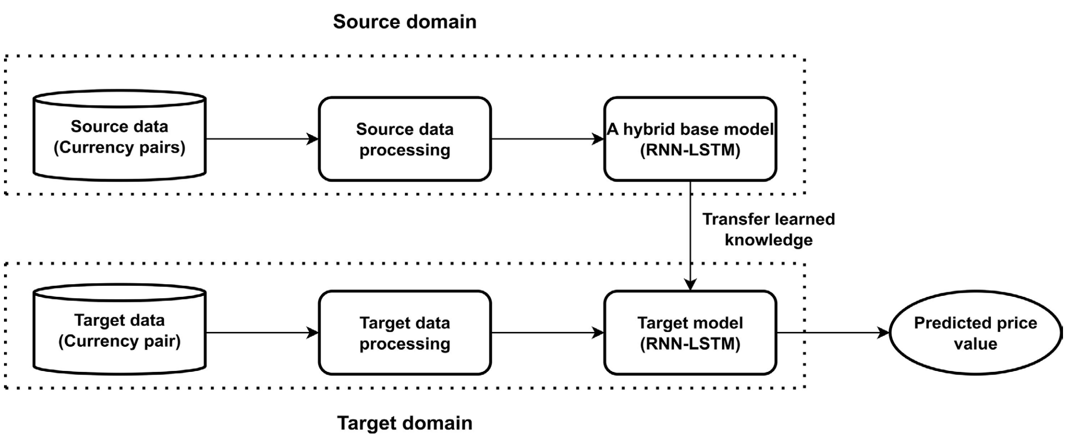



**Figure 4.** A framework of transfer learning.

# 4. Experiment and Results

The research environment's software and hardware configuration are displayed in Table 2.

Table 2. Configuration of an experimental environment.

| Items | Parameter |
|---|---|
| Central processing unit (CPU) | Core(TM) i7-7600U CPU@2.80GHz |
| Graphics processing unit (GPU) | Intel(R) HD Graphics 620 |
| Random-access memory (RAM) | 16 GB |
| Programming language | Python3.8.8 |
| Deep learning framework | TensorFlow v2.9.1 |

For our experiment, the historical data of Forex market where downloaded from the Tickstory website[4], with the time period of 3 months and hourly time frame.

## 4.1. Data Preprocessing

None of the datasets we used contained missing values. To avoid bias, we transformed the data into a scale range of [0, 1], acknowledging that variables recorded at different scales do not equally contribute to model fitting and learned function. We divided the dataset into the ratio of 6:2:2, with 60% for training, 20% for validation, and 20% for testing. We created the training dataset with shape (1052, 120, 20) and the testing dataset with shape (293, 120, 20) using the sliding window method approach. The shape reflects the number of samples, time steps, and currency pairs used in each set. The 120 time steps indicate the number of prior time steps used for forecasting the subsequent one-hour closing price, and the 20 currency pairs indicate the total number of currency pairs used in the model.

## 4.2. Performance Metrics

Every prediction model must be evaluated in order to determine its accuracy [23]. In this study, the three basic assessment metrics for regression issues for model evaluation: root mean squared error (RMSE), mean absolute error (MAE), and mean absolute percentage error (MAPE) are used to evaluate and compare the RNN, LSTM, and hybrid RNN-LSTM base model performance.

RMSE show an estimation of the residual between the real $y_t$ value and anticipated value, $\hat{y}_t$, MAE estimates the average magnitude of the errors in forecasts without looking at their direction, and MAPE indicates an average absolute percentage error; it is preferable if the MAPE is as low as possible [16]. Equations representing these metrics are given below:

[4]https://tickstory.com/download-tickstory/.

$$\mathrm{RMSE} = \sqrt{\frac{1}{N} * \sum_{t=1}^{N} \left( y_t - \hat{y}_t \right)^2} \tag{17}$$

$$\mathrm{MAE} = \frac{1}{N} * \sum_{t=1}^{N} \left| y_t - \hat{y}_t \right| \tag{18}$$

$$\mathrm{MAPE} = \frac{1}{N} * \sum_{t=1}^{N} \left| \frac{y_t - \hat{y}_t}{y_t} \right| \tag{19}$$

### 4.3. Results Analysis

We provide our findings for both source and target domain. We used the closing prices of 20 and 5 currency pairs to train and evaluate our hybrid base model for the source domain.

#### 4.3.1. Case 1

We trained and validated the base model for our hybrid RNN-LSTM model using 20 currency pairings as the source domain, as shown in **Table 1**. After training and validating the base model, we transferred the learned weights and biases to the target domain model to anticipate currency pairings.

However, **Table 3** shows unsatisfactory outcomes for our forecasts of the target domain currency pair, like GBP/USD. As a result, we opted for scenario 2.

**Table 3.** Model forecast evaluation case 1.

| Model | RMSE | MAE | MAPE |
|---|---|---|---|
| RNN | 16.138934 | 16.138692 | 0.935115 |
| LSTM | 15.880864 | 15.880837 | 0.934288 |
| RNN-LSTM | 15.757492 | 15.754454 | 0.955955 |

#### 4.3.2. Case 2

In this instance, we reduced the number of currency pairs in the source domain utilized for training and validating the basic model from 20 to 5, as shown in **Table 4**.

**Table 4.** Currency pairs used in case 2.

| Source data | | |
|---|---|---|
| **Major** | **Minor** | **Exotic** |
| AUD/USD | EUR/CHF | ZAR/JPY |
| USD/CAD | GBP/NZD | |
| **Target data** | | |
| GBP/USD | USD/JPY | USD/ZAR |
| EUR/USD | | AUD/NZD |

We trained and validated our base model with the default parameters and the RMSE and Loss (MSE) for 5 currency pairs used are shown in **Figure 5**.

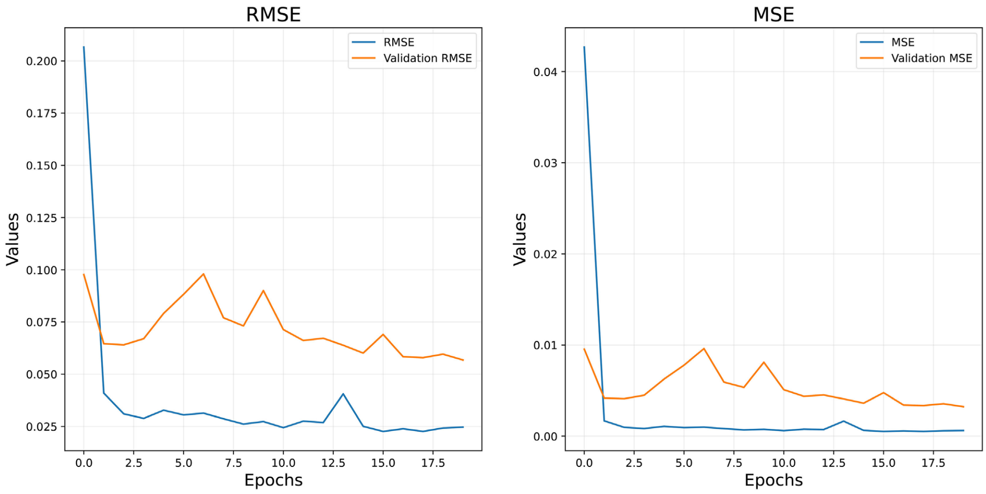



**Figure 5.** RMSE and loss before parameter tuning.

To reduce the training and validation RMSE and Loss, we fine-tuned the hyperparameters of our hybrid RNN-LSTM base model as there are displayed in **Table 5**. To navigate the vast search space, we used a random search to find the best hyperparameter optimization. The optimal set of hyperparameters with the lowest error consisted of 2 hidden layers for RNN and LSTM, 40 neurons, a dropout probability of 0.2, a batch size of 64, and 70 epochs. The improved RMSE and Loss are shown in **Figure 6**.

**Table 5.** Hyperparameters.

| Hyperparameter | Interval |
|---|---|
| Hidden layers for RNN and LSTM | 2 to 3 |
| Number of neurons | 14 to 70 |
| Dropout rate | 0.1 to 0.9 |
| Batch size | 16, 32, and 64 |
| Epochs | 10 to 100 |

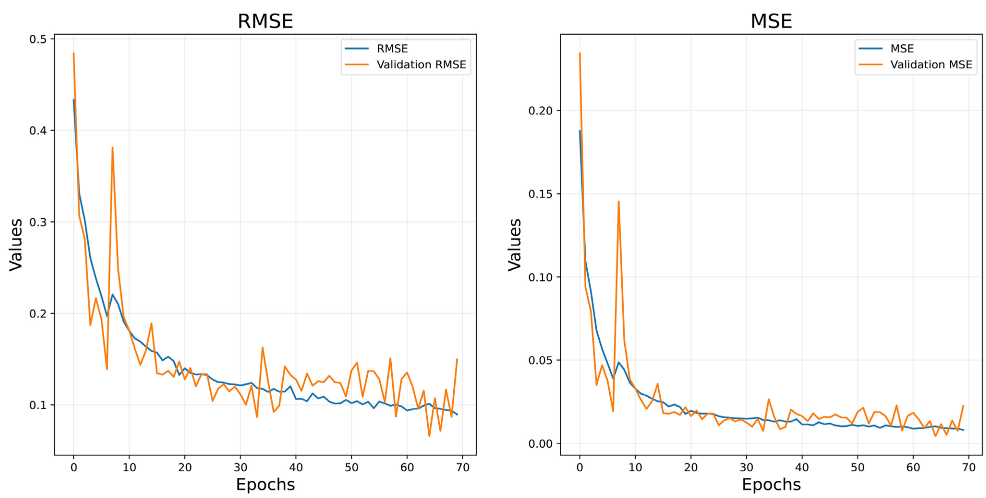



**Figure 6.** RMSE and loss after parameter tuning.

We utilized the weights and biases from our base hybrid model for the target domain and trained our dataset on top of it. To forecast the one-hour closing price of frequently traded Forex currency pairings like GBP/USD, EUR/USD, USD/JPY, USD/ZAR, and AUD/NZD, we used both historical data of closing prices for the past 3 months and technical indicators.

To train the target model for prediction, we froze the top third of the base model's layers, attached our dense layer, trained it with 64 batches, and ran 70 epochs. We made predictions for the currency pairs using both the RNN base model and the LSTM base model, with the same structure as the proposed RNN-LSTM hybrid model, for comparison purposes. The results of the predicted closing price are displayed in Table 6.

**Table 6.** Model forecast evaluation.

| Model | Pairs | RMSE | MAE | MAPE |
|---|---|---|---|---|
| RNN | GBP/USD | 0.022116 | 0.020246 | 0.017770 |
| | EUR/USD | 0.008276 | 0.007461 | 0.007584 |
| | USD/JPY | 2.091997 | 1.764091 | 0.012051 |
| | USD/ZAR | 0.275423 | 0.263631 | 0.014701 |
| | AUD/NZD | 0.005693 | 0.004293 | 0.003826 |
| LSTM | GBP/USD | 0.017378 | 0.015948 | 0.014063 |
| | EUR/USD | **0.003796** | **0.003054** | **0.003117** |
| | USD/JPY | **1.072977** | **0.879307** | **0.005954** |
| | USD/ZAR | 0.322514 | 0.312474 | 0.017475 |
| | AUD/NZD | 0.003056 | 0.002676 | 0.002392 |
| RNN-LSTM | GBP/USD | **0.007656** | **0.006473** | **0.005756** |
| | EUR/USD | 0.016708 | 0.015403 | 0.015498 |
| | USD/JPY | 3.443044 | 3.260152 | 0.022532 |
| | USD/ZAR | **0.16525** | **0.13648** | **0.007774** |
| | AUD/NZD | **0.001730** | **0.001355** | **0.001210** |

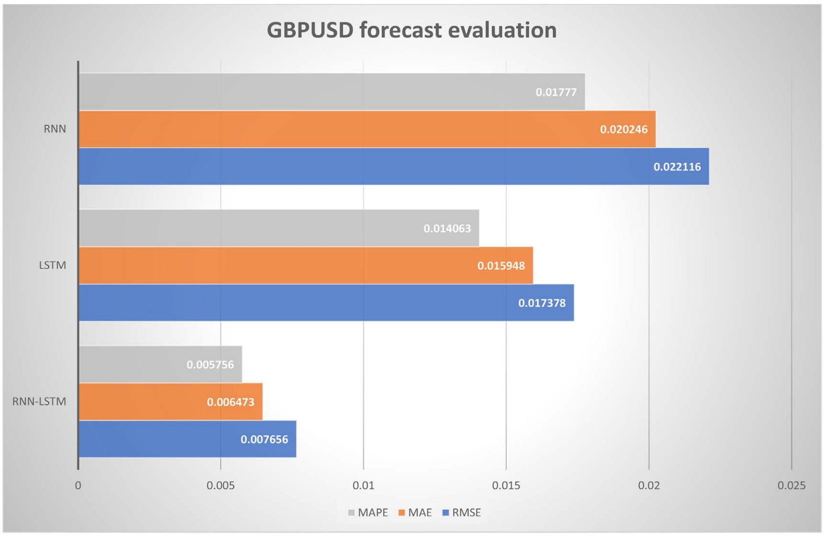
Our hybrid RNN-LSTM model achieved low RMSEs for GBP/USD, USD/ZAR, and AUD/NZD, with values of 0.007656, 0.16525, and 0.001730 respectively, according to Table 6. The LSTM base model had small RMSEs for EUR/USD and USD/JPY, with values of 0.003796 and 1.072977. The RNN-LSTM model produced the smallest MAE and MAPE values for GBP/USD, USD/ZAR, and AUD/NZD, while the LSTM model produced the smallest values for EUR/USD and USD/JPY.

## 5. Discussion

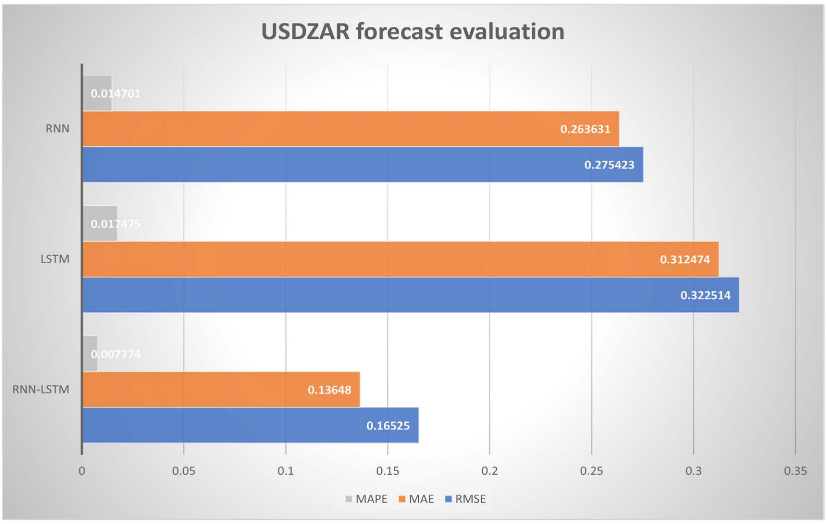
In this study, we initially proposed to train and validate the hybrid RNN-LSTM base model using 20 different currency pairs in the source domain (Table 1). However, when forecasting the target domain currency pair, the results were not satisfactory (Table 3). One of the factors that might have contributed to this was the use of a large number of datasets with different scales before applying transfer learning.

To address this issue, we retrained the base model using only 5 closely related currency pairs that have similar scales. The results showed improvement, as seen in Table 6, suggesting that the proposed hybrid RNN-LSTM base model works better with small datasets.

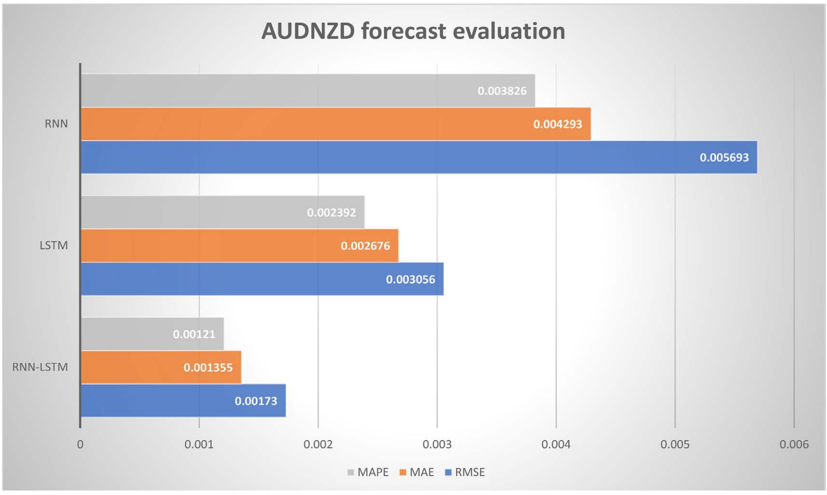
When forecasting GBP/USD, USD/ZAR, and AUD/NZD as shown in Figure 7, our proposed hybrid RNN-LSTM base model for transfer learning outperforms RNN and LSTM base model with both RMSE, MAE and MAPE. The LSTM basic model provided superior forecasts for the EUR/USD and USD/JPY as depicted in Figure 8.




(a)

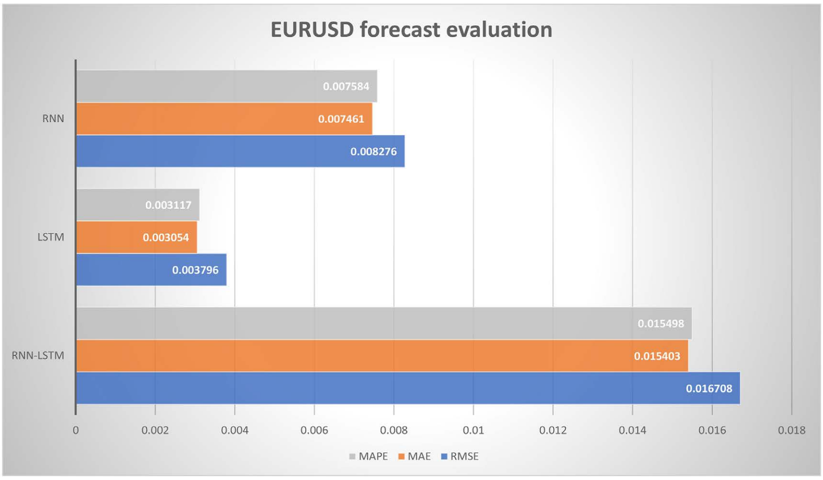



(b)

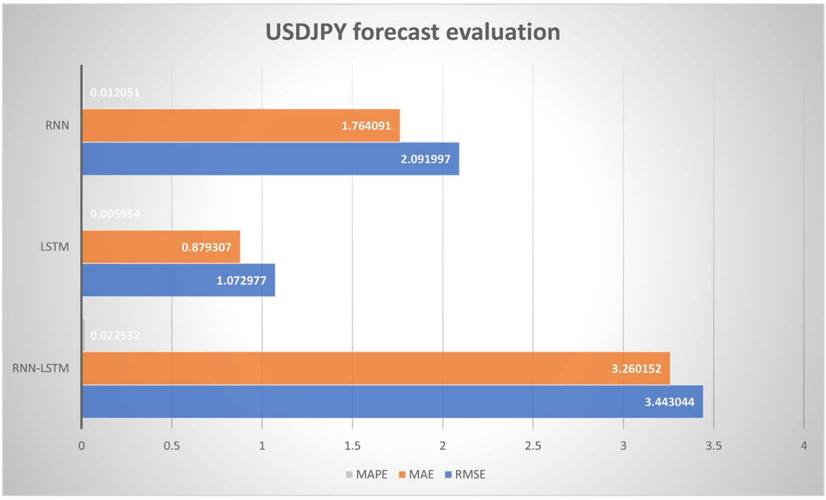



(c)

**Figure 7.** Evaluation metrics of GBP/USD, USD/ZAR, and AUD/NZD for all models.




(a)




(b)

**Figure 8.** Evaluation metrics of EUR/USD and USD/JPY for all models.

This indicates that our hybrid RNN-LSTM model is more computationally efficient and requires less time to train compared to other stock and Forex prediction models. Additionally, the use of transfer learning and the fine-tuning of hyperparameters also helped to improve the performance of the model. These results demonstrate the potential of our proposed hybrid RNN-LSTM model for real-time stock and Forex prediction.

Overall, our suggested hybrid RNN-LSTM base model outperforms RNN and LSTM base models when forecasting the currency pairings in the target domain, as evidenced by the prediction of three of the forecasted currency pairs more accurately by our proposed RNN-LSTM base model, and the AUDNZD was forecasted more precisely than other currency pairs examined. Figure 9 displays the out-of-sample forecast for the AUDNZD.

## 6. Conclusions

The results of our study indicate that the hybrid RNN-LSTM base model provides better accuracy compared to the RNN and LSTM base models when forecasting the target domain currency pairings. Our proposed model requires a lower number of epochs to converge and has a shorter average computation time of 4.29 minutes, making it a more efficient option for Forex traders. Also, using a smaller set of closely related currency pairs in the source domain improved the model's performance. This study highlights the effectiveness of using transfer learning and hybrid RNN-LSTM models for financial forecasting, particularly in the Forex market.

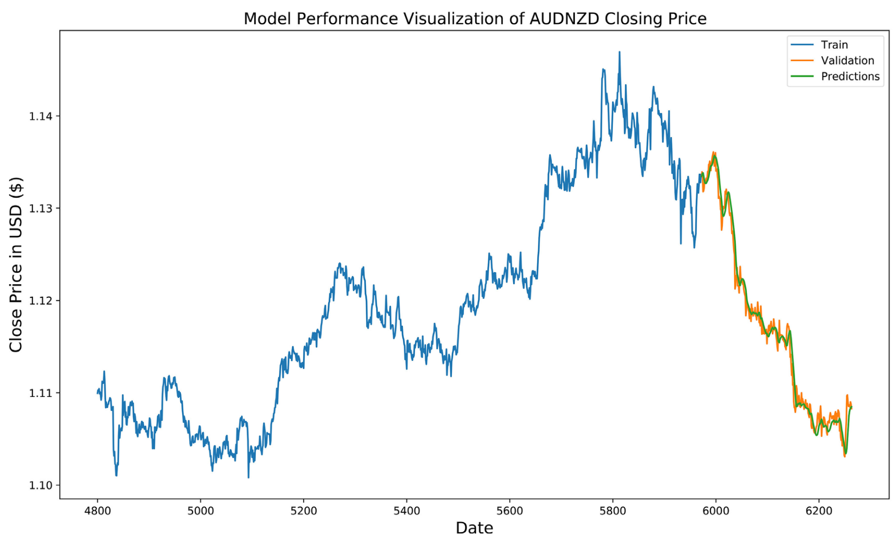



Figure 9. Out-sample forecast of AUDNZD.

Additionally, fine-tuning the hyperparameters and exploring different optimization algorithms and loss functions can also be considered in future studies to further improve the performance of the proposed model. Additionally, incorporating other factors such as economic news and sentiment analysis, can

also be considered as they play a crucial role in the Forex market and can help to increase the accuracy of the predictions. Overall, our proposed hybrid RNN-LSTM base model has shown promising results in predicting the Forex market, but there is still room for improvement and further exploration in the field. The web deployment for this study for real time dataset can be accessed through the following link
https://mrfaru-fx-forecasting-fx-forecasting-app-tprth8.streamlit.app/.

## Acknowledgment

My supervisors, Prof. Anthony Waititu and Dr. Lawrence Nderu, deserve my gratitude for guiding me through this project and allowing me to give it my all. Without them, I knew that this trip would be difficult for me. I appreciate your continued help, family and friends. Most importantly, I want to thank my lovely mother Jamila Habale and my devoted and loving wife Nuru Mohamed, who never ceases to inspire me.

## Conflicts of Interest

The authors declare no conflicts of interest regarding the publication of this paper.

## References

[1] Cirillo, S., Lloyd, S. and Nordin, P. (2014) Evolving Intraday Foreign Exchange Trading Strategies Utilizing Multiple Instruments Price Series. https://arxiv.org/abs/1411.2153v1

[2] Anastasakis, L. and Mort, N. (2009) Exchange Rate Forecasting Using a Combined Parametric and Nonparametric Self-Organising Modelling Approach. *Expert Systems with Applications*, **36**, 12001-12011. https://doi.org/10.1016/j.eswa.2009.03.057

[3] Islam, M.S., Hossain, E., Rahman, A., Hossain, M.S. and Andersson, K. (2020) A Review on Recent Advancements in FOREX Currency Prediction. *Algorithms*, **13**, 186. https://doi.org/10.3390/a13080186

[4] Tay, F.E. and Cao, L. (2001) Application of Support Vector Machines in Financial Time Series Forecasting. *Omega*, **29**, 309-317. https://doi.org/10.1016/S0305-0483(01)00026-3

[5] Zhang, N., Lin, A. and Shang, P. (2017) Multidimensional k-Nearest Neighbor Model Based on EEMD for Financial Time Series Forecasting. *Physica A*: *Statistical Mechanics and Its Applications*, **477**, 161-173. https://doi.org/10.1016/j.physa.2017.02.072

[6] Chen, H., Xiao, K., Sun, J. and Wu, S. (2017) A Double-Layer Neural Network Framework for High-Frequency Forecasting. *ACM Transactions on Management Information Systems*, **7**, 4. https://doi.org/10.1145/3021380

[7] Henrique, B.M., Sobreiro, V.A. and Kimura, H. (2019) Literature Review: Machine Learning Techniques Applied to Financial Market Prediction. *Expert Systems with Applications*, **124**, 226-251. https://doi.org/10.1016/j.eswa.2019.01.012

[8] Balabanov, T., Zankinski, I. and Dobrinkova, N. (2012) Time Series Prediction by Artificial Neural Networks and Differential Evolution in Distributed Environment. *Large-Scale Scientific Computing* 8*th International Conference*, *LSSC* 2011, Sozo-

pol, 6-10 June 2011, 198-205. https://doi.org/10.1007/978-3-642-29843-1_22 https://link.springer.com/chapter/10.1007/978-3-642-29843-1_22

[9] Rehman, M., Khan, G.M. and Mahmud, S.A. (2014) Foreign Currency Exchange Rates Prediction Using CGP and Recurrent Neural Network. *IERI Procedia*, **10**, 239-244. https://doi.org/10.1016/j.ieri.2014.09.083

[10] Gu, Q., Dai, Q., Yu, H. and Ye, R. (2021) Integrating Multi-Source Transfer Learning, Active Learning and Metric Learning Paradigms for Time Series Prediction. *Applied Soft Computing*, **109**, Article ID: 107583. https://doi.org/10.1016/j.asoc.2021.107583

[11] Nguyen, T.T. and Yoon, S. (2019) A Novel Approach to Short-Term Stock Price Movement Prediction Using Transfer Learning. *Applied Sciences* (*Switzerland*), **9**, 22. https://doi.org/10.3390/app9224745

[12] Cem Kadilar, D., Şdmşek, M., Gör Çağdaş Hakan Aladağ, A., Üniversitesi, H., Fakültesi, F. and Bölümü, D. (2009) Forecasting the Exchange Rate Series with Ann: The Case of Turkey. *Istanbul University Econometrics and Statistics e-Journal*, **9**, 17-29. https://dergipark.org.tr/en/pub/iuekois/issue/8991/112073

[13] Hochreiter, S. (1998) The Vanishing Gradient Problem during Learning Recurrent Neural Nets and Problem Solutions. *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, **6**, 107-116. https://doi.org/10.1142/S0218488598000094

[14] Ni, L., Li, Y., Wang, X., Zhang, J., Yu, J. and Qi, C. (2019) Forecasting of Forex Time Series Data Based on Deep Learning. *Procedia Computer Science*, **147**, 647-652. https://doi.org/10.1016/j.procs.2019.01.189

[15] Yıldırım, D.C., Toroslu, I.H. and Fiore, U. (2021) Forecasting Directional Movement of Forex Data Using LSTM with Technical and Macroeconomic Indicators. *Financial Innovation*, **7**, 1-36. https://doi.org/10.1186/s40854-020-00220-2

[16] Hossain, M.A., Karim, R., Thulasiram, R., Bruce, N.D. and Wang, Y. (2019) Hybrid Deep Learning Model for Stock Price Prediction. *Proceedings of the* 2018 *IEEE Symposium Series on Computational Intelligence*, *SSCI* 2018, Bengaluru, 18-21 November 2018, 1837-1844. https://doi.org/10.1109/SSCI.2018.8628641

[17] Wu, D., Wang, X. and Wu, S. (2022) Jointly Modeling Transfer Learning of Industrial Chain Information and Deep Learning for Stock Prediction. *Expert Systems with Applications*, **191**, Article ID: 116257. https://doi.org/10.1016/j.eswa.2021.116257

[18] Azevedo, A. and Santos, M.F. (2008) KDD, SEMMA and CRISP-DM: A Parallel Overview. *IADIS European Conference Data Mining*, Amsterdam, 24-26 July 2008, 182-185.

[19] Rathore, V.S., Worring, M., Mishra, D.K., Joshi, A. and Maheshwari, S. (2018) Emerging Trends in Expert Applications and Security: Proceedings of ICETEAS 2018. Springer, Berlin, 485-495. https://doi.org/10.1007/978-981-13-2285-3

[20] Huynh, H.D., Dang, L.M. and Duong, D. (2017) A New Model for Stock Price Movements Prediction Using Deep Neural Network. *Proceedings of the* 8*th International Symposium on Information and Communication Technology*, Nha Trang, 7-8 December 2017, 57-62. https://doi.org/10.1145/3155133.3155202

[21] Hochreiter, S. and Schmidhuber, J. (1997) Long Short-Term Memory. *Neural Computation*, **9**, 1735-1780. https://doi.org/10.1162/neco.1997.9.8.1735

[22] Farahani, A., Pourshojae, B., Rasheed, K. and Arabnia, H.R. (2020) A Concise Review of Transfer Learning. *Proceedings* 2020 *International Conference on Computational Science and Computational Intelligence*, *CSCI* 2020, Las Vegas, 16-18 De-

cember 2020, 344-351. https://doi.org/10.1109/CSCI51800.2020.00065

[23] Ubert de Almeida, B., Ferreira Neves, R. and Horta, N. (2018) Combining Support Vector Machine with Genetic Algorithms to Optimize Investments in Forex Markets with High Leverage. *Applied Soft Computing Journal*, **64**, 596-613. https://doi.org/10.1016/j.asoc.2017.12.047