Geliş Tarihi:
21.10.2023
Kabul Tarihi:
19.03.2024
Yayımlanma Tarihi:
28.06.2024

Kaynakça Gösterimi: Gür, Y. E. (2024). Forecasting the euro Exchange rate usıng deep learnıng algorıthms and machıne learnıng algorıthms. *İstanbul Ticaret Üniversitesi Sosyal Bilimler Dergisi, 23(49)*, 1435-1456. doi: 10.46928/iticusbe.1379268

FORECASTING THE EURO EXCHANGE RATE USING DEEP LEARNING ALGORITHMS AND MACHINE LEARNING ALGORITHMS

Araştırma

Yunus Emre Gür
Sorumlu Yazar (Correspondence)
Fırat Üniversitesi
yegur@firat.edu.tr

Yunus Emre GÜR, Fırat Üniversitesi Yönetim Bilişim Sistemleri bölümünde Doktor, Araştırma Görevlisidir. Yönetim Bilişim Sistemleri alanında ders vermekte ve Makine Öğrenimi ve Derin Öğrenme konularında araştırmalar yayınlamaktadır.

FORECASTING THE EURO EXCHANGE RATE USING DEEP LEARNING ALGORITHMS AND MACHINE LEARNING ALGORITHMS

Yunus Emre Gür
yegur@firat.edu.tr

Abstract

Given that time series forecasts are of great importance in the financial world, the main objective of this study is to forecast Euro prices and examine the contribution of these forecasts to financial decision-making processes. Since the Euro is an important component of international trade and investment, accurate price forecasts are of strategic importance for many financial institutions and investors. In this study, we compare the performance of deep learning algorithms and classical machine learning methods for forecasting Euro prices: support vector machines (SVM), Extreme Gradient Boosting (XGBoost), long short-term memory (LSTM), and gated recurrent units (GRU). These methods represent different algorithms that are widely used in financial forecasting and give successful results. The dataset used in the study was divided into two parts: 80% training and 20% testing, and it is also indicated how each algorithm behaved during the training process and which parameters were chosen. The results are presented by comparing the performance of these algorithms, and it is found that the GRU algorithm provides better accuracy than the others. Therefore, the GRU algorithm was chosen to forecast Euro prices for the next 12 months, and the forecasting process was carried out. The results of this study are expected to provide an important perspective to financial decision-makers by comprehensively comparing the performance of deep learning and traditional approaches in Euro price forecasting. It also includes potential research avenues for future work and suggestions for the development of new methods in this area.

Keywords: Time Series Forecasting, Deep Learning, Machine Learning, Euro Exchange Rate Forecast, Gated Recurrent Unit

JEL Classification: C22, C45, C88, G17

DERİN ÖĞRENME ALGORİTMALARI VE MAKİNE ÖĞRENMESİ ALGORİTMALARI KULLANARAK EURO DÖVİZ KURUNUN TAHMİN EDİLMESİ

Özet

Zaman serisi tahminlerinin finans dünyasında büyük önem taşıdığı düşünüldüğünde, bu çalışmanın temel amacı Euro fiyatlarının tahmin edilmesi ve bu tahminlerin finansal karar alma süreçlerine katkısının incelenmesidir. Euro, uluslararası ticaret ve yatırımın önemli bir bileşeni olduğu için doğru fiyat tahminleri, birçok finansal kurum ve yatırımcı için stratejik öneme sahiptir. Bu çalışmada, Euro fiyatlarını tahmin etmek için derin öğrenme algoritmaları ve klasik makine öğrenmesi yöntemlerinin performansları karşılaştırılmıştır: Destek Vektör Makineleri (SVM), Aşırı Gradyan Artırma (XGBoost), Uzun Kısa Vadeli Hafıza (LSTM) ve Geçitli Tekrarlayan Birim (GRU). Bu yöntemler, finansal tahminlerde yaygın olarak kullanılan ve başarılı sonuçlar veren farklı algoritmaları temsil etmektedir. Çalışmada kullanılan veri seti, %80 eğitim ve %20 test olmak üzere ikiye bölünmüş ve her bir algoritmanın eğitim süreci boyunca nasıl davrandığını ve hangi parametrelerin seçildiği de belirtilmiştir. Sonuçlar, bu algoritmaların performansları karşılaştırılarak sunulmuş ve GRU algoritmasının diğerlerine göre daha iyi bir doğruluk sağladığı görülmüştür. Bu nedenle, gelecek 12 aylık Euro fiyatlarını tahmin etmek için GRU algoritması seçilmiş ve tahmin işlemi gerçekleştirilmiştir. Bu çalışmanın sonuçlarının, Euro fiyat tahmininde derin öğrenme ve geleneksel yaklaşımların performanslarını kapsamlı bir şekilde karşılaştırarak finansal karar alıcılara önemli bir bakış açısı sunacağı düşünülmektedir. Ayrıca, gelecek çalışmalar için potansiyel araştırma yollarını ve bu alanda yeni yöntemlerin geliştirilmesine yönelik önerileri de içermektedir.

Anahtar Kelimeler: Zaman Serisi Tahmini, Derin Öğrenme, Makine Öğrenimi, Euro Döviz Kuru Tahmini, Geçitli Tekrarlayan Birim

JEL Sınıflandırması: C22, C45, C88, G17

INTRODUCTION

Forecasting the future of socio-economic development is an inevitable requirement for both private and public organizations at the decision-making stage. These organizations should have the ability to foresee future events and produce appropriate solutions to future challenges with effective plans (Bircan and Karagöz, 2003: 49).

Most of the forecasting studies in the field of social sciences are oriented towards predicting future events based on past and current data. Time series forecasting analysis refers to predicting the possible future values of the system by using past and current data about the system. Such forecasting models are widely applied, especially in financial markets, for variables such as stock indices, exchange rates, economic growth rates, inflation data, and interest rates. Making the most accurate forecasts can help individuals, businesses and governments shape economic policies and help financial actors make better-planned decisions (Bağcı, 2020: 2).

However, exchange rates are at the center of economic crises in increasingly integrated economies around the world. Therefore, an economic crisis occurring in one country may also affect other countries with which that country has relations. Developing countries, unlike developed countries, face some difficulties in the development of financial markets and have to deal with problems such as government intervention in exchange rates. Therefore, it is extremely important for developing countries to make successful exchange rate forecasts or to minimize forecast errors (Özkan, 2011: 186).

In this context, this study focuses on the Euro because of its impact on economic variables as well as its intensive use in the investment world. As the world economy has become increasingly globalized, exchange rates are at the center of many economic crises. This not only affects the economic policies of countries but also affects the personal financial lives of individuals in Turkey, especially through its impact on inflation. In addition to being used intensively for investment purposes in countries struggling with high inflation, the euro is one of the most monitored assets in Turkey.

The expected returns of future economic transactions may be adversely affected by exchange rate fluctuations. For this reason, individuals, businesses, foreign trade firms, economic actors who want to use resources in international markets, savers and speculators who aim to profit from exchange rate volatility, hedging strategies, short-term financing decisions, short-term investment preferences, and long-term financing plans want to predict the future movements of exchange rates (Ketboğa, 2019: 35).

In this context, making accurate forecasts about the future course of Euro prices is critical to economic and financial planning. Therefore, this study examines the ability to predict future prices by comparing different methods for forecasting Euro prices. The comparison between deep learning techniques and traditional forecasting methods is the main focus of this study. A variety of algorithms, such as "S*upport Vector Machines (SVM), XGBoost, Long Short-Term Memory (LSTM), and Gated Recurrent Units (GRU)*" are used in the study to assess the predicting ability of Euro prices. The results showed that GRU outperformed the other methods. Therefore, the GRU algorithm was selected and successfully applied to forecast future 12-month Euro prices. In order to better estimate financial outcomes, this research compares deep learning and conventional machine learning techniques. Euro price forecasting remains a critical issue for financial planning and risk management. It also aims to make a significant contribution to future studies and literature.

The study is organized into the following phases: Reviewing financial time series forecasting research utilizing both conventional techniques and deep learning algorithms, the second portion concentrates on financial time series forecasting. The study's fundamental data collection and methods are covered in depth in the third part. The study's primary results are presented in the fourth part, and broad conclusions, assessments, and suggestions are included in the fifth and final section.

LITERATURE REVIEW

For the purpose of financial time series forecasting, this section discusses research that combines deep learning algorithms with conventional machine learning techniques. Towards this end;

Özdemir et al. (2011) evaluated the effectiveness of machine learning techniques that go beyond traditional statistical methods in forecasting stock returns in financial markets. Logistic regression and support vector machines were utilized among other machine learning techniques to forecast the movement of the ISE-100 index. The results show that both methods have approximately 75% and 86% correct classification rates for modeling and prediction data, respectively. These results showed that the support vector machine method can provide an effective alternative for forecasting stock returns. This study provides an important advance in supporting risk analysis and investment decisions in financial markets.

In order to forecast the future value of the Nepalese rupee in relation to the US dollar, the euro, and the pound sterling, Ranjit et al. (2018) looked at a variety of machine learning approaches. These techniques included artificial neural networks (ANN) and recurrent neural networks (RNN). RNN is a type of neural network with a self-feedback feature. The purpose of this study is to create prediction models using different feed-forward ANN, back-propagation, and RNN architectures and compare their accuracy rates. Furthermore, many ANN architectures were examined, including the GRU, LSTM, SRNN, and Feed Forward Neural Network. For every currency, input parameters including the opening, low, high, and closing prices were utilized. The study's findings demonstrated that LSTM networks outperformed SRNN and GRU networks in terms of performance.

Alpay (2020) examined the usability of deep learning methods in forecasting financial time series. Important time intervals and large delays in time series were analyzed and predicted in the research using an algorithm called LSTM. The evaluations were made using the USD/TRY exchange rate data set between January 1, 2000, and December 31, 2017. The study's findings demonstrated that the LSTM technique may be used to financial data forecasting with success and provide predictions that are more accurate.

Abar (2020) aimed to predict the future performance of gold prices and analyzed the effects of different factors to achieve this objective. The research included factors that are thought to impact gold prices, such as the price of silver, the price of crude oil WTI futures, the US dollar index, the S&P 500 index, the US federal funds compound interest rate, and the US CPI, to develop models. The January 2015 to June 2020 timeframe was covered by the data utilized. This research used linear regression, MARS, and XGBoost models to forecast gold prices. The effectiveness of these models is assessed by comparing the outcomes according to predetermined standards. Among the variables used, the US CPI variable was found to have a significant impact on gold prices. These results suggest that XGBoost and MARS methods can be effective choices for forecasting gold prices and similar series. This study makes an important contribution to support financial risk analysis and investment decisions.

Using data from firms traded on the BIST 30 Index, Ustalı et al. (2021) attempted to forecast the future performance of the stock prices. The firms in the BIST 30 Index from 2010 to 2019 had their quarterly financial statements collated for this purpose, and financial ratios were computed using the assembled information. Monthly closing prices of company stocks for the same period were also collected, and these data were arranged as quarterly averages in order to harmonize them with financial ratios. Following the conclusion of the data preparation stage, many techniques, including ANNs, the random forest (RF) algorithm, and the XGBoost algorithm, were used to forecast each company's future stock values. Then, the forecasting results obtained by each method are compared. The results of the study show that the XGBoost algorithm performs the most effectively and makes more successful forecasts than other methods. In addition, it is concluded that both XGBoost and RF algorithms have higher forecasting accuracy than the ANN. By reviewing the body of research in the area, this study assesses the efficacy of machine learning approaches in future price projections of the stocks of firms listed in the BIST 30 Index and produces significant findings.

In order to study exchange rate projections, Köse (2021) contrasted the results of gray forecasting methods and the LSTM model. Comparisons were made both based on numerical data and visual evaluations. The necessary codes were written using the Python programming language. In comparison to the Grey model and other statistical forecasting techniques, the findings demonstrate the superior efficacy and success of the LSTM model. According to RMSE and other performance measures, LSTM has been observed to make better predictions. However, this study also points out that the LSTM model can be improved and emphasizes that more efficient results can be obtained by improving the layers within it. Consequently, this research shows that for exchange rate forecasting, the LSTM model performs better than the GM (1,1) Grey model.

In order to predict gold prices, Yurtsever (2021) overcame the challenges brought on by the market's intricate and non-linear structure. Between 2001 and 2021, the research examines the impact of the following variables: interest rate indicators, stock market index, exchange rate index, gold prices, and crude oil prices. Artificial neural network techniques including LSTM, Bi-LSTM, and GRU were used to build the models for this purpose. The models were then assessed using metrics for mean absolute error (MAE), mean absolute percentage error (MAPE), and root mean square error (RMSE). With 3.48 MAPE, 61,728 RMSE, and 48.85 MAE values, the LSTM model therefore fared the best.

The examination of numerical data using data mining techniques that was collected from the Central Bank of the Republic of Turkey was the main subject of the research conducted by Ata and Erbudak (2022). This dataset contains information on exchange rates, and the analysis period lasted from January 2020 to May 2021. During this period, exchange rate models were created using various machine-learning algorithms. Decision support trees, support vector machines, gaussian regression, and linear regression are among the approaches examined. 1352 exchange rate data points in all were used in four distinct models. Each model has different accuracy rates. The first model has an accuracy of 99.84%, the second model is 99.18%, the third model is 93.72%, and the fourth model is 86.83%. These results emphasize that the model with the highest accuracy rate can be used to develop strategies and plans. In order to provide exchange rate projections, this research examines the impact of data mining and machine learning techniques and compares the results to the accuracy rates of techniques previously used in the literature.

Özcan (2023) focused on analyzing monthly BIST 100 index data and comparing various machine learning methods for direction prediction. Nine distinct machine learning techniques were used to analyze data spanning from January 2002 to September 2022 in order to forecast whether the index values will rise or fall from month to month. The methods employed in this study encompass Logistic Regression Analysis (LR), Linear Discriminant Analysis (LDA), Naive Bayes Algorithm (NB), Random Forests Algorithm (RF), K-Nearest Neighbour Algorithm (KNN), SVM-RBF, Classification and Regression Trees Algorithm (CART), Artificial Neural Networks (NNET), Support Vector Machines with Gaussian Kernel Function (SVM-RBF), and Support Vector Machines with Polynomial Kernel Function (SVM-POLY). The results show that linear methods produce more successful forecasts. This study makes an important contribution to understanding future financial market movements.

Foreign currency markets, commodities markets, and stock market indices of emerging nations that affect BIST100 were investigated by Akbulut and Adem (2023). According to the authors, the economies of countries are closely interrelated, so it is vital to understand and predict the interactions in financial markets. Closing prices from January 2017 through October 2021 make up the data set utilized in the analysis. To ensure the study's findings were objective, researchers used cross-validation techniques using k = 5 and 10-folds. Models were compared using MAE, RAE, RMSE, and LSTM. The LSTM model outperformed the competition in terms of prediction accuracy, as shown by the experiments. Specifically, the LSTM model's MAE value is 10.27, RMSE value is 14.15, and RAE value is 6.06. This research significantly advances our knowledge of the intricate relationships and price projections seen in financial markets.

The intricacy of financial time series data was discussed by Erden (2023), who also looked at the use of various techniques to forecast the performance of EREGL stock, which is traded on Borsa Istanbul's major metal market. In the research, deep learning algorithms including LSTM, GRU, and RNN were utilized to forecast data, while the ARIMA model was used for conventional time series analysis. Different time frames, feature extraction, and data preparation are some of the techniques used to enhance the deep learning model's performance. The RNN algorithm beat the others and obtained an accuracy of 93% on the test data set, according to the study's experiments with time delays in various circumstances. This study provides an effective approach to understanding the complex interactions and price forecasts in financial markets.

METHODOLOGY

Research Design

This section provides an overview of the dataset and the machine learning and deep learning techniques used to the euro price forecasting process. The Central Bank of the Republic of Turkey's official website provided the historical information utilized in this research to predict the average monthly Euro pricing. Therefore, official data are used in this study. The data set consists of monthly average Euro prices between January 2000 and October 2023. The dataset is split into training and test datasets so that the approaches may be tested. In this case, 20% of the dataset is used for testing and the remaining 80% for training. Table 1 displays a portion of the study’s data set.

Table 1. A Portion of The Dataset Used for The Research

Date	Euro Price	Date	Euro Price
2000-01	0.55	….	….
2000-02	0.55	….	….
2000-03	0.56	2023-05	21.43
2000-04	0.56	2023-06	24.94
2000-05	0.56	2023-07	29.20
2000-06	0.58	2023-08	29.43
….	….	2023-09	28.82
….	….	2023-10	29.08

Resource: evds2.tcmb.gov.tr

During data pre-processing, an input pricing dataset is created by converting the Euro data into a dataset originating from the NumPy format. Furthermore, MinMaxScaler (Equation 1) is used to express the values of all features on the same scale and to normalize the dataset. This makes the calculations faster and more efficient. Subsets for testing, evaluation, and training are then created from the dataset. The evaluation data set is used to compare the prediction mistakes made during the optimization phase, while the training data is utilized to train the model. Ultimately, predictions are produced, the model's performance is assessed on the test dataset, and outcomes are determined.

$$x_{scaled} = \frac{x - x_{min}}{x_{max} - x_{min}} \quad (1)$$

Evaluation Matrix

Forecasting past Euro prices provides important information by evaluating the performance of different methods. An in-depth comparison of SVM, XGBoost, LSTM, and GRU models' results is performed here. Statistical indicators such as MAPE, RMSE, MAE, and R^2 are used to compute the accuracy of the models' prediction outcomes. The formulae for determining these statistical values are shown in Equations 2, 3, and 5.

$$\mathrm{MAPE} = \frac{1}{N} \sum_{i=1}^{N} \frac{|\widehat{X_i} - X_i|}{X_i} \quad (2)$$

$$\mathrm{RMSE} = \sqrt{\frac{1}{n} \sum_{i}^{n} (X_i - \widehat{X_i})^2} \quad (3)$$

$$\mathrm{MAE} = \frac{1}{n} \sum_{i}^{n} |X_i - \widehat{X_i}| \quad (4)$$

$$R^2 = 1 - \frac{\sum_i (y_i - \hat{y}_i)^2}{\sum_i (y_i - \mu)^2} \quad (5)$$

The following sub-headings explain in detail how each algorithm works and which features it focuses on.

Data Analysis

Long and Short Term Memory (LSTM) Algorithms

When compared to conventional RNNs, LSTM is the superior ANN design. LSTM has a structure that can better handle long-distance connections. Basically, LSTM uses special memory cells that can store and transmit the outputs of units at different time steps. These memory cells are designed to preserve historical information and manage the flow of information through various control gates. In addition, the memory cells of the LSTM use a cell state to store important information about past events. The forget gate, entrance gate, and exit gate are all components of this cell's regulatory network. In this way, LSTM stands out with its ability to capture complex relationships over time and its capacity to provide more control over data (Yurtsever, 2022: 1485).

Cells in LSTM are more complex information processing units compared to typical conventional neurons. Numerous gates of various lengths are included in each cell with the goal of preserving and arranging the information flow along rows. This characteristic allows LSTM to determine whether information is more significant in the short or long term, which makes it ideal for a wide range of sequential issues. The following is a definition for an LSTM cell:

$$i_t = \sigma(W_{ii} x_t + b_{ii} h_{t-1} + b_{hi}) \quad (6)$$

$$f_t = \sigma(W_{if}x_t + b_{if} + W_{hf}h_{t-1} + b_{hf}) \quad (7)$$

$$g_t = tanh\,(W_{ig}x_t + b_{ig} + W_{hg}h_{t-1} + b_{hg}) \quad (8)$$

$$o_t = \sigma(W_{io}x_t + b_{io} + W_{ho}h_{t-1} + b_{ho}) \quad (9)$$

$$c_t = f \odot c_{t-1} + i_t \odot g_t \quad (10)$$

$$h_t = o_t \odot \tanh\,(c_t) \quad (11)$$

In this case, "x" is the input state and "h" is the concealed state. The current time increment is denoted by t. Multiplication by elements (Hadamard multiplication) is represented by "$\odot$", while sigmoid activation is indicated by. Gates in their present configuration are denoted as g_t (cell gate), "f_t" (forget gate), "i_t" (input gate), and "o_t" (output gate). The weights and bias terms between gates may be learned, and they are represented by "Wxy" and "bxy" respectively. Figure 1 is a graphic representation of the information transfer that takes place inside of an LSTM cell.

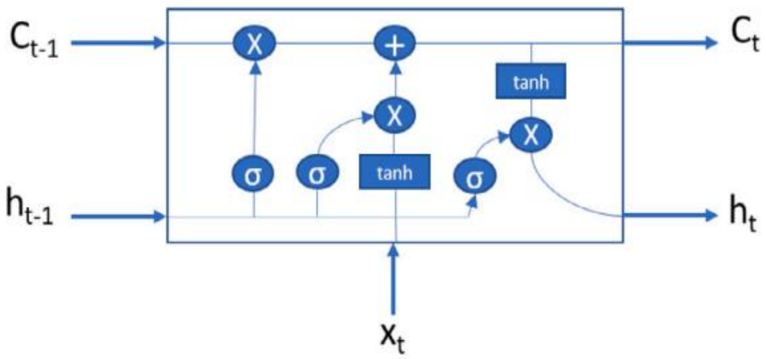

Figure 1. The Cell of LSTM
Resource: Elmaz et al. (2021)

Gated Recurrent Unit (GRU) Algorithms

GRU, as introduced by Cho et al. in 2014, stands as a variation of Recurrent Neural Networks (RNNs). One of the primary challenges faced by traditional RNNs lies in effectively capturing and processing long-range dependencies within sequences. This challenge is addressed by integrating gating mechanisms into the architecture. To determine the fraction of the current input (x_t) and the previous output (h_{t-1}) to transmit on to the next cell, GRU uses two crucial gates: the update gate (z_t) and the reset gate (r_t). This stands in contrast to LSTM's more intricate structure. GRU boasts a reduced count of learnable parameters compared to LSTM, mainly attributable to the absence of LSTM's output layers (Nosouhian et al., 2021: 2). Figure 2 provides a visual depiction of a GRU unit, while Equation 12 encapsulates the entire update mechanism within GRU units.

$$\binom{r}{z} = \binom{\text{sig}\,m}{\text{sigm}}\, W^l \binom{h_t^{l-1}}{h_{t-1}^l} \quad (12)$$

$$\tilde{h}_t^l = \tanh\left(W_x^l h_t^{l-1} + W_g^l\left(r \odot h_{t-1}^l\right)\right) \quad (13)$$

$$h_t^l = (1-z)\odot h_{t-1}^l + z \odot \tilde{h}_t^l \tag{14}$$

On the other hand, the reset gate plays a pivotal role in assigning significance to the previously obtained information. Weighted by "W" the data retained in the memory is selectively dispatched, ensuring only essential details advance to the next step. As we tackle equations 15 and 16, we lay out the fundamental operations that underpin the Gated Recurrent Unit (GRU).

Update Gate:

$$\mathrm{z_t} = \sigma(\mathrm{W_z} * [\mathrm{h_{t-1}}, \mathrm{x_t})] \tag{15}$$

Reset Gate:

$$\mathrm{r_t} = \sigma(\mathrm{W_r} * [\mathrm{h_{t-1}}, \mathrm{x_t})] \tag{16}$$

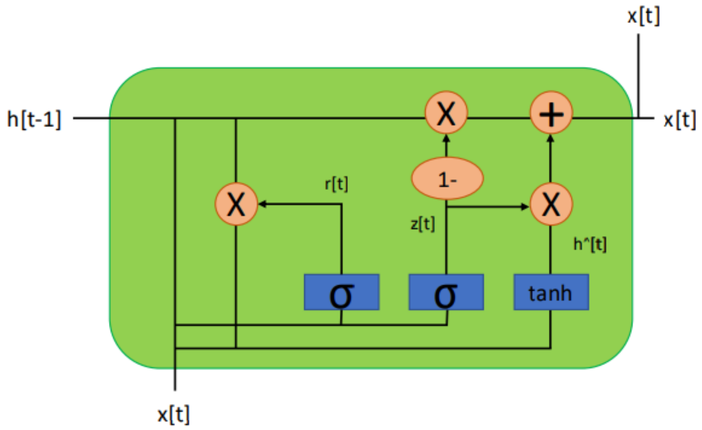

Figure 2. An Illustration of A Single-Unit GRU Diagram
Resource: Nosouhian et al., 2021: 2

Support Vector Machines (SVM) Kernel Algorithms

The statistical learning theory that has developed over the last several decades is where SVM got its start. Time series forecasting, object categorization, pattern recognition, and other machine learning tasks are just a few of the many uses for support vector machines (SVM), which were invented in 1995 by Vapnik and his colleagues. One of SVM's branches, Support Vector Regression, takes a distinct approach in forecasting. Instead of rigidly adhering to a predefined 'model,' it leverages observed data to predict the outcomes of a function, following a path less conventional in traditional time series forecasting methodologies. The goal of a time series prediction method is to create a function $f(x)$ given a dataset containing time series $x(t)$, where t is a sequence of N discrete samples expressed as $t = \{0,1,2,3, \dots, N-1\}$, and $y(t+\triangle)$ is an expected future value (t higher than or equal to N). This function is responsible for producing an output that corresponds to the projected value within a specified prediction horizon. Through the means of regression analysis, equations (17) and (18) serve the purpose of establishing such prediction functions, catering to both linear and non-linear regression scenarios (Sapankevych and Sankar, 2009: 27).

$$f(x) = (w \cdot x) + b \tag{17}$$

$$f(x) = (w \cdot \phi(x)) + b \tag{18}$$

When the input data are non-linear, the goal is to map them via "$\varphi(x)$" (the Kernel Function) to a higher-dimensional feature space. Then, linear regression on higher-dimensional feature space is the goal. In addition to establishing the ideal weight w and threshold b, "*optimal*" weight criteria are created to accomplish this aim. The Euclidean norm (minimizing $||w||^2$) is a useful tool for assessing the "*smoothness*" of the weights, the first criteria. The second criterion is empirical risk, which seeks to reduce estimate error as much as possible. The objective is to get$R_{reg}(f)$, where f is a function of $x(t)$, as little as possible.

$$R_{reg}(f) = R_{\text{emp}}(f) + \frac{\lambda}{2}||w||^2 \tag{19}$$

The term "λ" refers to a scaling factor that serves as a regularization constant or a capacity control term. Its purpose is to protect data against overfitting and generalization. Evidence-based risk is:

$$R_{\text{emp}}(f) = \frac{1}{N}\sum_{i=0}^{N=1} L\left(x(i), y(i), f(x(i), w)\right) \tag{20}$$

For a given training set, where $t = \{0,1,2,3, \dots, N-1\}$, the truth data, or predicted value, is indicated by $y(i)$, the index " i " signifies a discrete time series. A loss function or cost function, $L(\cdot)$, must be defined. The e-insensitive loss function, created by Vapnik, and the quadratic loss function, often used with the Least Squares SVM, are popular. A quadratic programming problem is formulated using the e-insensitive loss function in order to determine the proper weights and minimize regularized risk (Mellit et al., 2013: 299):

minimize $\frac{1}{2}||w||^2 + C\sum_{i=1}^{n} L\big(y(i), f(x(i), w)\big)$

where

$$L\big(y(i), f(x(i), w)\big)$$
$$= \begin{cases} |y(i), f(x(i), w)| - \varepsilon & if |y(i), f(x(i), w)| \geq \epsilon \\ 0 & otherwise \end{cases} \tag{21}$$

To find the optimal weights, the Karush-Kuhn-Tucker conditions must be met, which say that the product of the variables and constraints is zero at that time. Multiplying the optimum weights by the dot products between data points characterizes the estimate of the function $f(x)$:

$$f(x) = \sum_{i=1}^{N}(a_i - a_i^*)\langle x, x(i)\rangle + b. \tag{22}$$

The locations on or off the "ε" tube that have Lagrange multipliers "a" greater than zero are called "*support vectors*". This exemplifies how a subset of the whole dataset is generally necessary to get the optimal weights for these non-zero Lagrange multipliers. In other words, $f(x)$ may be defined without using the whole dataset. One of the numerous benefits of this method is that the solution is quite small. The space $x(i)$ is transformed into a (possibly) higher-dimensional feature space called "$\varphi(x(i))$" for use in SVR-based non-linear regression. A Mercer-conforming kernel function may be created as follows, given that the solution in SVR is based on the dot products of input data:

$$k(x, x') = \langle \phi(x), \phi(x') \rangle \tag{23}$$

To compute the ideal weights w inside the feature space, substitute this value into equation (22). The process remains the same. Many kernel functions—such as the Gaussian, polynomial, and hyperbolic tangent—satisfy Mercer's requirements. Because they allow non-linear data to be mapped into "feature" spaces, which are linear by nature, kernels are essential to SVM/SVR applications. The optimization procedure seen in the linear example may now be replicated. Even though the Gaussian kernel is a popular option, empirical evaluations are often needed to choose an acceptable kernel function. The architecture of the resultant SVR model is shown in Figure 3.

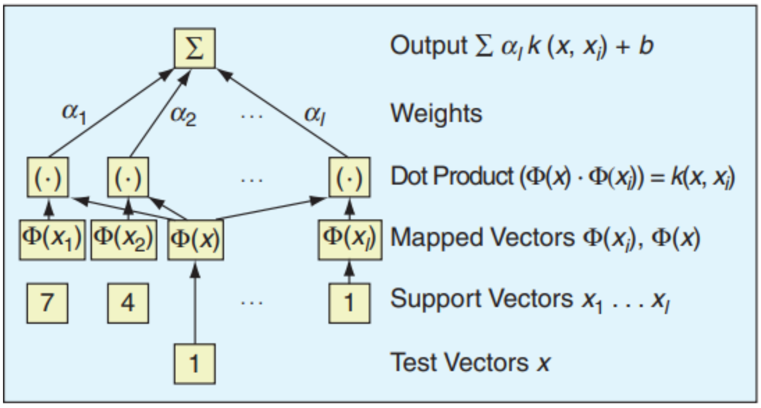

Figure 3. Support Vector Machines Architecture
References: Sapankevych and Sankar, 2009: 28

Extreme Gradient Boosting (XGBoosting) Algorithms

The open-source machine learning method XGBoost uses gradient boosting to handle a variety of prediction tasks, including regression and classification. In the context of financial forecasting, XGBoost plays a crucial role in improving the accuracy of predictions related to stock prices, market trends, and risk assessment. Its ability to handle large and complex financial datasets, along with its efficient optimization techniques, makes it a preferred choice for financial analysts and researchers seeking robust predictive models in the world of finance (Yan et al., 2022: 5).

$$obj(f) = \sum_{i=1}^{n} L(y_i, \ \hat{y}_i) + \Omega(f) \tag{24}$$

The number of training samples is indicated by "n" in this case, the real label is "y_i," the predicted label is "$\hat{y}_i$," the training loss function is "L," and a regularization term is "Ω". The regularization term prevents overfitting, while the training loss function evaluates how well the model predicts. The individual predictions made by XGBoost's K regression trees are denoted by the notation "$\left(f_k(x_i)\right)$". The ultimate prediction is the aggregate sum of all these trees outcomes.

$$\hat{y}_i = \sum_{k=1}^{K} f_k(x_i) \tag{25}$$

Taking equation (25) into account in relation to equation (24), the Kth tree's objective function is expressed as follows.

$$obj(f) = \sum_{i=1}^{n} L\left(y_i,\ \hat{y}_i^{K-1} + f_K(x_i)\right) + \Omega(f_K) + const \tag{26}$$

The regularization term of the first K-1 trees is where the equation's constant originates. Equation (26) may be transformed into the following form by using Taylor expansion.

$$obj(f) = \sum_{i=1}^{n} \left[L\left(y_i, \hat{y}_i^{K-1}\right) + g_k f_k(x_i) + \frac{1}{2} h_i f_k^2\ (x_i)\right] + \Omega(f_K) + const \tag{27}$$

In order to decrease model complexity, one derives the regularization term through the following process:

$$\Omega(f) = yT + \frac{1}{2}\lambda \|w\|^2 \tag{28}$$

T stands for the total number of leaves, ω for the leaf weights, and λ and γ, which have default values of 1 and 0, respectively, are the coefficients. Machine learning models play a pivotal role in forecasting and adapting to new data, making their capacity for generalization and model accuracy crucial factors to be considered. Figure 4 depicts the XGBoost algorithm's design.

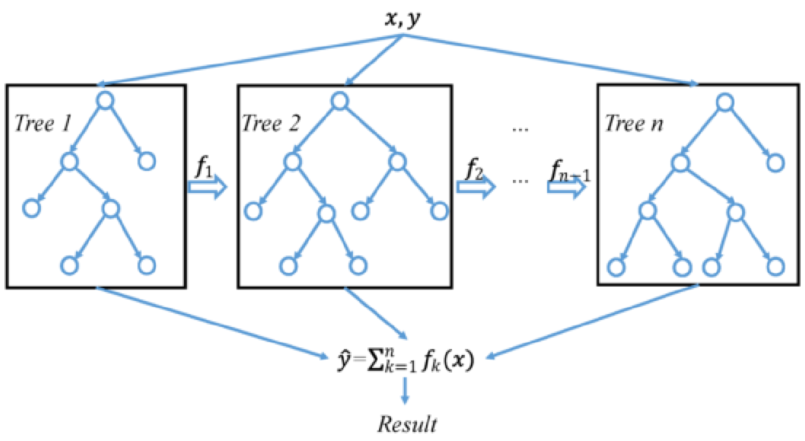

Figure 4. The Architecture of The Xgboost
Resource: Wang et al., 2019: 138

Hyperparameter Settings of Algorithms

For the LSTM method, a sequential model was developed in this work. There is just one layer employed in this LSTM model estimate technique. 200 neurons and a hyperbolic tangent activation function are features of the LSTM layer. Then, one output unit is used to add a thick layer. The "adam" optimization process was then used to construct the model, with MSE serving as the loss function. For 100 epochs, the model was trained using batches of 64 data points each from the training set. Test data were also used to assess the model's performance. In conclusion, the LSTM model's hyperparameters are as follows:

- Number of neurons in the LSTM layer: 200
- Activation function: Hyperbolic Tangent (tanh)
- Optimization algorithm: Adam
- Loss function: Mean Square Error (MSE)
- Training periods (epoch): 100
- Batch size: 64

A GRU cell of 200 units makes up the first layer of the GRU model. There is just one output unit in the second layer, which is dense. Models use mean squared error (MSE) loss and the "adam" optimizer. The model underwent 100 epochs of training using the training set of data. We utilized 64 samples in each epoch. In a nutshell, hyperparameters:

- Number of GRU cells: 200 units
- Activation function: Hyperbolic Tangent (tanh)
- Optimization algorithm: Adam
- Loss function: Mean Square Error (MSE)
- Duration of the training process (epochs): 100
- Number of samples used in each training step (batch size): 64

When building the SVM model, the Radial Basis Function kernel with kernel "rbf" is used. This allows the SVR model to make non-linear predictions. The model fits the training data and generates the optimal prediction function. When constructing the XGBoost regression model, the XGBRegressor class is used. The objective of the model is to predict Euro prices using input features. Training data is used by the model to determine the link between the input characteristics and the target Euro prices. It then makes predictions using test data.

FINDINGS

The study explores the application of LSTM, GRU, XGBoost, and SVM models for the task of forecasting euro prices. The sole independent variable used across these models is the time series data representing the average monthly exchange rate between the euro and the Turkish lira. The dataset was sourced from the official website of the Central Bank of the Republic of Turkey, evds.mb.gov.tr, spanning a temporal range of 286 months, commencing from January 2000 and extending to October 2023. The study uses five different error indicators to assess how well the forecasting models perform. These error metrics encompass RMSE, MSE, MAE, MAPE, and R^2. Each model undergoes a sequence of development, training, and subsequent assessment using a dedicated dataset. The error metrics resulting from the model outcomes are presented in Table 2 for reference.

Table 2. Test Results of LSTM, GRU, XGBoost, and SVM Algorithms

Model	RMSE	MAE	MAPE	R^2	MSE
LSTM	0.2438	0.1648	0.0604	0.9982	0.0594
SVM	2.0895	2.0207	0.9277	0.8688	4.3662
XGBoosting	0.2579	0.1106	0.0148	0.9980	0.0665
GRU	**0.0829**	**0.0552**	**0.0131**	**0.9997**	**0.0068**

The average deviation between the model's predictions and the actual values is shown by the RMSE value. Better forecasts are shown by a lower RMSE. The average absolute deviation of the forecasts from the actual data is measured by the MAE. A low MAE means better predictions. MAPE measures the percentage deviation of predictions from actual values. A lower MAPE means better predictions. The R^2 value measures how much of the data variance the model explains. A high R^2 value indicates better model performance. The MSE statistic calculates the average squared error between forecasts and reality. Better prediction accuracy is shown by a smaller MSE. The better the forecast, the lower the MAE number should be. The more precise a prediction is, the lower the RMSE number should be. Values of R^2 might be anything from zero to one. Higher accuracy in predictions is shown by reduced margins of error (MAE) and RMSE (the difference between the expected and actual values). When R^2 is near to 1, it indicates that the values are very close to each other (Sulistio et al., 2023: 180-181). The findings show that the GRU model has the lowest error rates across the board. In particular, low values of RMSE, MAE, MAPE, and MSE indicate that this model makes better predictions than the others. Furthermore, the high R^2 value indicates that the model provides a good explanation for the data. When compared to other models, the SVM performs poorly because to its high RMSE and MAE values. The R^2 value is also lower than the other models, indicating that it explains the data poorly. Although XGBoost and LSTM models also perform quite well, the GRU model explains the data in the best way. Therefore, the GRU model gave the best result in this study. Figure 5, however, is a chart depicting the results of the LSTM, GRU, XGBoost, and SVM models in comparison to their predictions.

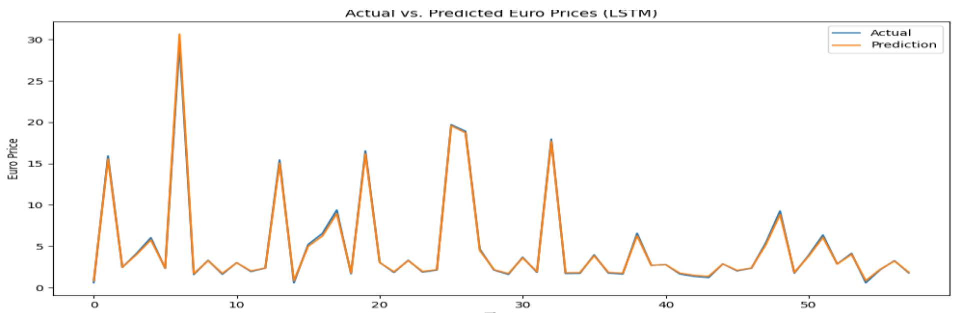

Figure 5a. Actual and Predicted Values of LSTM Algorithm

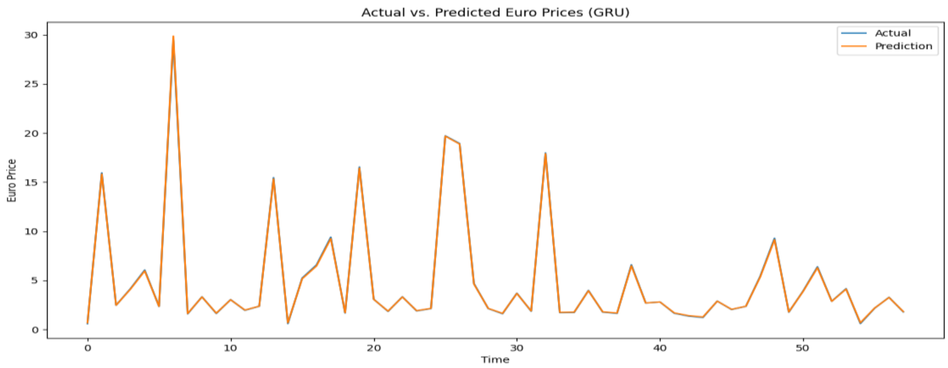

Figure 5b. Actual and Predicted Values of GRU Algorithm

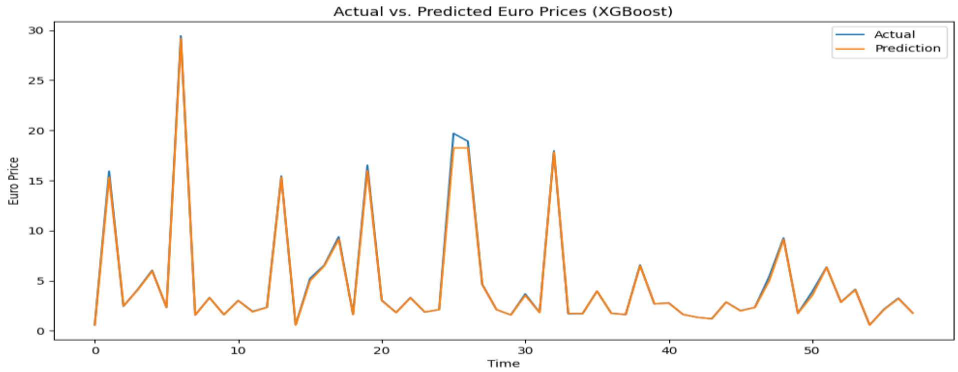

Figure 5c. Actual and Predicted Values of XGBoost Algorithm

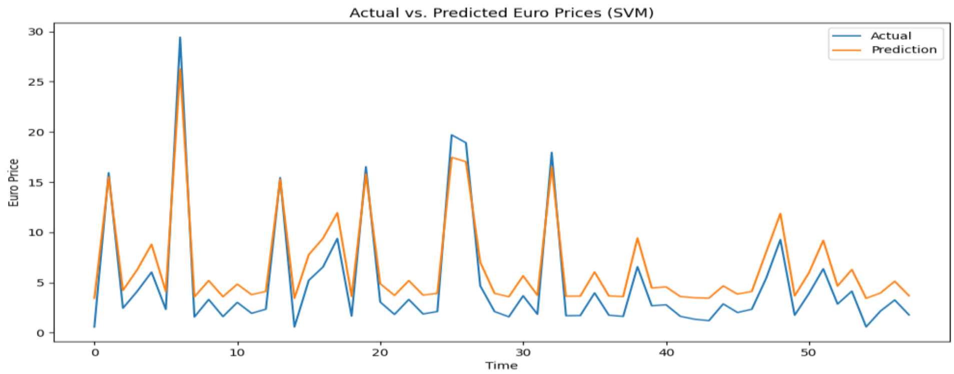

Figure 5d. Actual and Predicted Values of SVM Algorithm

As can be seen from the graphs of the test results of the models shown in Figure 5, according to the comparison results, the GRU model has the lowest RMSE value, which means that its predictions are closer to the actual data than the other models. SVM has the highest RMSE, which indicates the lowest prediction accuracy. Again, the GRU model has the lowest MAE, indicating that its predictions are closer to the actual values. SVM has the highest MAE. The GRU model has the lowest MAPE compared to the other models, indicating the lowest error rate in percentage terms. SVM has the highest MAPE. The GRU model has the highest R^2, indicating that it best explains the data. SVM has the lowest R^2. The RU model has the lowest MSE, meaning that it performs best in terms of squared error. SVM has the highest MSE. In conclusion, according to all error statistics obtained, the GRU model shows the best performance and makes better predictions than the other models. The SVM model, on the other hand, generally has a lower prediction accuracy than the other models.

After it was determined that the GRU model was the best model in terms of test performance in the monthly Euro forecasting process, the next 12-month forecasting process was started. The reason for choosing the GRU algorithm in the future forecasting process is that it shows the best performance in the training and testing phases.

The average Euro rate forecast for the next 12 months was made by performing the following steps: For future prediction with the GRU model, feature generation is performed. In order to use monthly data in particular, a sequence is created. This array contains the past 12 months of data. This creates the features that the model will use to predict future monthly prices. Next, the GRU model is built with a sequential Keras model using Adam optimization, as in the training and testing phases. The first layer of the model is a GRU layer and contains 200 units. The hyperbolic tangent is used as an activation function. A dense layer is then added. The model is used to forecast Euro prices for the next 12 months. Initially, 12 months of data at the end of the current data series are taken. Then, a forecast for each future month is made. These forecasts are returned to their original scales and printed.

Future forecastInitially, we use the data from the last 12 months to forecast future prices based on the model's predictions. This data is taken from the most recent part of the X array and corresponds to the “look_back” created in the previous step. This array represents the current time series that the model will use to generate its forecasts. Next, forecasts for the next 12 months are made. For each month, the model generates a forecast using “current_sequence”. This forecast is stored in a variable called “next_month”. These forecasts are used to generate future forecasts in the next step. After the future forecasts have been calculated, “current_sequence” is updated. This shifts the last 12 months of data to obtain a new monthly forecast. This shifting makes room for the last month’s forecast to be used in the calculation of the next forecast. That is, for each new month, the oldest month's forecast is subtracted and replaced by the new forecast. The forecasts for each month are returned to the original scales as predicted by the model using This is done specifically to make prices more understandable and comparable. Finally, the Euro price forecasts for the next 12 months are printed. For each month, the month the forecast represents is indicated along with the forecast value. This can be used to understand future price movements and make decisions. Forecast results for the next 12 months with the GRU model are presented in Table 3 and Figure 6.

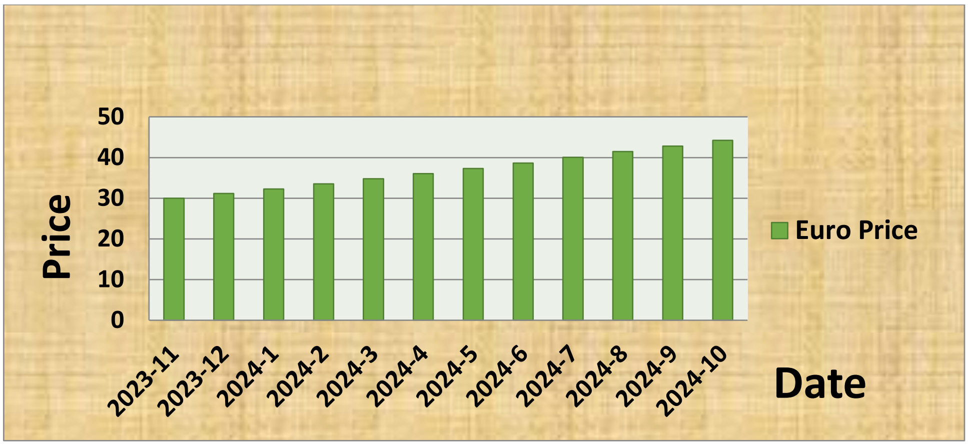

Figure 6. GRU Model's Euro Forecast for the Next 12 Months

Table 3. Euro Price Forecasts of GRU Model for the Next 12 Months

Date	Euro	Date	Euro
2023-11	29.98	2024-5	37.33
2023-12	31.12	2024-6	38.67
2024-1	32.29	2024-7	40.05
2024-2	33.5	2024-8	41.43
2024-3	34.74	2024-9	42.8
2024-4	36.01	2024-10	44.21

The model forecasts the likely course of Euro prices for the period from November 2023 to October 2024. The forecasts indicate that the Euro price will follow a monthly average trend between 29.98 Turkish lira and 44.21 Turkish lira and will tend to increase continuously. These forecasts can be an important source of information on future Euro price movements and provide a valuable guide for financial decisions.

CONCLUSION, DISCUSSION AND RECOMMENDATIONS

This study was carried out to examine the performance of LSTM, GRU, XGBoosting, and SVM algorithms for financial time series forecasting and to evaluate the ability of these algorithms to predict Euro prices. For this purpose, after evaluating the model performances, it was found that the best performance belongs to the GRU model. Euro price forecasts for the next 12 months were obtained using the GRU model.

The results of this study show that the GRU model exhibits superior performance in financial time series forecasting. In particular, the RMSE (root mean squared error) value is 0.0829, the MAE (mean absolute error) value is 0.0552, and the R^2 (R-squared) value is 0.9997. The results demonstrate the high accuracy and precision levels of the GRU model in price forecasts. Moreover, these results offer the potential to provide financial analysts and investors with more reliable information about future Euro prices. Therefore, it can be concluded that deep learning techniques such as the GRU model are an effective tool for financial market analysis.

At the same time, the performance of the GRU model is compared with similar previous studies in the literature. These comparisons reveal the importance and usability of deep learning algorithms for predicting future price movements in financial markets. For example, Ranjit et al. (2018) used LSTM models in their study and obtained successful results. However, in this study, higher accuracy values were obtained by using the GRU model. In addition, some machine learning techniques in other studies, especially XGBoost and MARS methods (Abar, 2020), have high accuracy rates.

The results show that the GRU model has significant forecasting ability. For the 12-month period, Euro price forecasts have higher accuracy rates than other methods. The model is particularly useful for capturing the dynamic nature of the data and can offer financial analysts better insights into future price movements. These results offer a number of suggestions for the future. Firstly, researchers developing forecasting models in financial markets may want to consider the further use of deep learning techniques and their comparison with traditional methods. Continuous monitoring and improvement of the performance of the model is important to achieve greater success in financial market analysis. Secondly, GRU can be hybridized with other deep learning and machine learning models to test its effectiveness on real-time financial data streams. Third, this model can be applied to various financial instruments, such as stocks, bonds, and cryptocurrencies, further optimizing the algorithm and adjusting the parameters. Fourth, external factors such as geopolitical events, policy changes, and economic indicators can be integrated into the models. Fifth, interdisciplinary collaborations with experts in economics, finance, and data science can be established to develop educational resources and training modules based on the study findings. Finally, these advanced techniques can be compared with traditional econometric models to examine their long-term forecasting capabilities. It is believed that these various suggestions will expand the impact of the research and improve the accuracy and reliability of financial forecasts.

The results of this study show that the GRU model is an effective tool for Euro price forecasts. These results are expected to make a significant contribution to the literature and guide financial analysts, investors, and researchers in predicting future price movements. For future research, it is suggested to evaluate the performance of GRU and other deep learning algorithms in the price prediction of various financial assets. Analyses on extended data sets and different time periods may contribute to the improvement of these techniques. In conclusion, it is expected that this study will inspire further research on the use and effectiveness of deep learning techniques in financial analysis and that these approaches will promote the understanding of financial markets and the development of more effective investment strategies.

REFERENCES

Abar, H. (2020). Estimation of gold prices with Xgboost and Mars methods. *Ekev Academy Journal, (83)*, 427-446.

Akbulut, S., & Adem, K. (2023). BIST 100 prediction with the interaction of financial instruments of developing countries using deep learning and machine learning methods. *Niğde Ömer Halisdemir University Journal of Engineering Sciences, 12(1)*, 1-1.

Alpay, Ö. (2020). USD/TRY price forecasting using LSTM architecture. *European Journal of Science and Technology*, 452-456.

Ata, O., & Erbudak, A. E. (2022), Exchange rate forecasting application with data mining and machine learning. *Fırat University Journal of Engineering Sciences, 34(2)*, 553-563.

Bağcı, B. (2020). Increasing the forecasting power of moving averages and exponential smoothing methods: forecasting the exchange rate in Turkey. *Turkuaz International Journal of Socio-Economic Strategic Research, 2(1)*, 1-12.

Bircan, H., & Karagöz, Y. (2003). An application on monthly exchange rate forecasting with Box-Jenkins models. *Kocaeli University Journal of Social Sciences, (6)*, 49-62.

Central Bank of the Republic of Turkey, https://evds2.tcmb.gov.tr/index.php?/evds/serieMarket Access Date: 04.10.2023.

Cho K., Van Merrienboer B., Gulcehre C., Bahdanau, D., Bougares, F., Schwenk, H. & Bengio, Y. (2014). Learning phrase representations using RNN encoder-decoder for statistical machine translation, *arXiv preprint*, *arXiv:1406.1078*, 2014.

Elmaz, F., Eyckerman, R., Casteels, W., Latré, S., & Hellinckx, P. (2021). CNN-LSTM architecture for predictive indoor temperature modelling. *Building and Environment, 206*, 108327.

Erden, C. (2023). Comparison of forecasting performance of deep learning and arima methods: the case of a borsa İstanbul stock. *Journal of Management and Economics, 30(3)*, 419-438.

Ketboğa, M. (2019). Characteristicso exchange rate forecasting methods, their superiority and weaknesses in determining exchange rate movements. *Giresun University Journal of Economics and Administrative Sciences, 5(2)*, 44-67.

Köse, Ü. B. (2021). *Exchange rate forecasting with artificial neural networks and grey model (*Unpublished Master's Thesis), Marmara University, Institute of Social Sciences, Istanbul.

Mellit, A., Pavan, A. M., & Benghanem, M. (2013). Least squares support vector machine for short-term prediction of meteorological time series. *Theoretical and applied climatology, 111*, 297-307.

Nosouhian, S., Nosouhian, F., & Kazemi Khoshouei, A. (2021). A review of recurrent neural network architecture for sequence learning: comparison between lstm and gru. *Preprints*. 1-7, https://doi.org/10.20944/preprints202107.0252.v1.

Özcan, K. A. (2023), Prediction of stock exchange index direction with machine learning methods: the case of BIST 100. *Gümüşhane University Journal of Social Sciences, 14(3)*, 1001-1018.

Özdemir, A. K., Tolun, S. & Demirci, E. (2011). Estimation of index return direction by binary classification method: The case of ISE 100 index. *Journal of Niğde University Faculty of Economics and Administrative Sciences, 4(2)*, 45-59.

Özkan, F. (2011). Alternative Approach with artificial neural networks in exchange rate forecasting. *Eskişehir Osmangazi University Journal of the Faculty of Economics and Administrative Sciences, October 2011, 6(2)*, 185-200

Ranjit, S., Shrestha, S., Subedi, S., & Shakya, S. (2018). Comparison of algorithms in foreign exchange rate prediction. *In 2018 IEEE 3rd International Conference on Computing, Communication and Security (ICCCS)*, 9-13, IEEE.

Sapankevych, N. I., & Sankar, R. (2009). Time series prediction using support vector machines: a survey. *IEEE Computational Intelligence Magazine, 4(2)*, 24-38.

Sulistio, B., Warnars, H. L. H. S., Gaol, F. L. & Soewito, B. (2023). Energy sector stock price prediction using the CNN, GRU & LSTM hybrid algorithm. *In 2023 International Conference on Computer Science, Information Technology and Engineering (ICCoSITE)*, 178-182, IEEE.

Ustalı, N. K., Tosun, N. & Tosun, Ö. (2021). Stock price prediction with machine learning techniques. *Eskişehir Osmangazi University Journal of Economics and Administrative Sciences, 16(1)*, 1-16.

Wang, Y., Pan, Z., Zheng, J., Qian, L. & Li, M. (2019). A hybrid ensemble method for pulsar candidate classification. *Astrophysics and Space Science, 364*, 1-13.

Yan, H., Yan, K. & Ji, G. (2022). Optimisation and prediction in the early design stage of office buildings using genetic and XGBoost algorithms. *Building and Environment, 218,* 109081. https://doi.org/10.1016/j.buildenv.2022.109081.

Yurtsever, M. (2022). Automobile sales forecasting using economic ındicators with lstm method. *Journal of Nevşehir Hacı Bektaş Veli University Institute of Social Sciences, 12(3),* 1481-1492.