Journal of Social Research

P-ISSN:2827-9832
E-ISSN: 2828-335x
Vol. 5, No. 8, July 2026
http://ijsr.internationaljournallabs.com/index.

A Data Science-Based Forex Prediction System (EUR/USD) With Integrated Machine Learning and a Web Dashboard

Rafli Maulana*, Sufajar Butsianto, Elkin Rilvani
Universitas Pelita Bangsa, Indonesia
Email: raffliilaana@gmail.com*, Sufajar@pelitabangsa.ac.id, elkin.rilvani@pelitabangsa.ac.id

Keywords:
Forex;
Data Science;
Machine Learning;
LSTM;
Web Dashboard

Abstract

The rapid development of information technology has driven significant transformations in various sectors, including the global financial sector. One of the most dynamic financial instruments is foreign exchange (forex), the trading of foreign currencies, which plays a crucial role in international economic activity. However, the high volatility and complexity of factors influencing exchange rates make predicting forex price movements a significant challenge. Conventional approaches such as fundamental and technical analysis are limited by their subjective nature and inability to recognize non-linear patterns in price data. This research aims to develop a data science-based forex prediction system with the integration of machine learning algorithms and an interactive web dashboard. The models used include Long Short-Term Memory (LSTM) and Gated Recurrent Unit (GRU), designed to analyze time series data and capture long-term patterns in the EUR/USD currency pair's price movements. The development process follows the CRISP-DM (Cross Industry Standard Process for Data Mining) stages, including data understanding, data preparation, modeling, evaluation, and system implementation. Thus, this research provides two main contributions: (1) academically, strengthening the application of Machine Learning methods in the field of digital finance based on Data Science, and (2) practically, producing a prototype of an interactive forex prediction system that can be used as a decision support system for traders and financial analysts.

INTRODUCTION

Information technology has advanced significantly over the past two decades, transforming several sectors including global finance. Forex, the movement of foreign currencies, is a key financial instrument supporting international trade, investment, and monetary policy (Carney 2019; Cofnas 2015; Ogunbiyi-Badaru et al. 2024). It is the largest and most liquid financial market in the world, with the Bank for International Settlements (BIS) reporting a daily global forex turnover of approximately USD 7.5 trillion in April 2022 (Banco de México, 2022).

Many investors rely on forex given its importance to the global financial system, though exchange rates fluctuate rapidly due to factors such as inflation, interest rates, geopolitical conditions, and market sentiment (Hu et al., 2021).

These factors make forex price prediction particularly challenging, as inaccurate forecasts can cause significant financial losses for traders, investors, and analysts. Forex analysis has traditionally relied on two approaches: fundamental analysis, which focuses on macroeconomic issues such as monetary policy, inflation, and official economic reports, and technical analysis (Gamaliy et al. 2018; Kadiri et al. 2015; Walter et al. 2020; Yıldırım et al. 2021).

Technical analysis uses historical price data and tools such as Moving Averages, the Relative Strength Index (RSI), and Moving Average Convergence Divergence (MACD) to identify patterns. While both approaches provide useful insight, fundamental analysis is poorly suited to short-term prediction, and technical analysis cannot capture the complexity and non-linearity of forex price movements (Lamani et al. 2025; Ozdemir et al. 2026; Saberironaghi et al. 2025).

Data science has therefore become a new paradigm in financial data analysis, enabling large-scale data processing, application of Machine Learning algorithms, and objective quantitative performance measurement (Nguyen et al. 2023; Nwaimo et al. 2019; Wilson et al. 2024). In forex, this involves preprocessing and feature engineering of historical price data, building predictive models using algorithms such as LSTM, GRU, and ARIMA, and evaluating prediction accuracy with metrics including RMSE, MAE, and MAPE (Islam & Hossain, 2021).

LSTM, a variant of Recurrent Neural Networks (RNNs), is popular for its ability to learn long-term dependencies in time series data and has been shown to outperform traditional methods such as ARIMA on non-linear, dynamic financial data (Bormpotsis et al., 2023). Because model accuracy alone is not sufficient, prediction results must also be presented clearly; interactive web dashboards address this by allowing traders and analysts to visualize historical trends, forecasts, and real-time price movements in a single view. Integrating Machine Learning with web dashboards offers further advantages, including:

In this context, Deep Learning approaches such as LSTM have become an increasingly popular alternative for predicting non-linear, volatile financial time series such as forex.

As a development of Recurrent Neural Networks (RNNs), LSTM is designed to capture long-term dependencies in time series data and has been shown in various studies to outperform traditional statistical models such as ARIMA on dynamic, non-stationary financial data.

Although LSTM offers advantages in capturing non-linear patterns, a benchmark model is still needed to evaluate its performance gains. ARIMA is therefore used as the baseline model in this study to compare predictions from the Deep Learning approach against a conventional statistical method (Bousqaoui et al. 2021; Kontopoulou et al. 2023; Wilson et al. 2024).

Advantages of this integration include clear, intuitive visualization of predictions, presentation of new data via Application Programming Interface (API) development, and potential use as an intelligent decision support system (DSS) in practice (Saghafi et al., 2025). Most previous studies have focused on improving Machine Learning model accuracy without integrating results into an accessible visualization platform (Hu et al., 2021; Yıldırım et al., 2021).

Therefore, based on this background, this study proposes developing a “Data Science-Based Forex Prediction System with Machine Learning Integration and a Web Dashboard,” designed to process historical forex data (primarily EUR/USD) and build a Machine Learning-based predictive model, present results in an interactive web dashboard for ease of understanding, and assess system performance in terms of model accuracy and dashboard functionality. Based on this approach, the study has two objectives:

Academic Relevance → Contributing to the literature in Data Science, Machine Learning, and Software Engineering related to financial data. Practical Input → Demonstrating a prototype system providing computational and analytical tools for traders, investors, and financial analysts in the digital age (Odunaike 2025; Tuarob et al. 2021; Wang et al. 2025).

To keep the research focused, several limitations are established, including the use of Long Short-Term Memory (LSTM) as the primary prediction model and ARIMA as the comparison baseline, the employment of historical forex price data (open, high, low, close) at specific time intervals, and the web dashboard serving exclusively to visualize historical data and forecast results without direct integration with trading platforms. Based on the background explained above, the research problems are formulated to address how the EUR/USD currency pair can be predicted using the Long Short-Term Memory (LSTM) approach based on Data Science, how the LSTM model's performance compares with the ARIMA statistical baseline in predicting EUR/USD forex price movements, and how to design and implement a web dashboard to present forex prediction results visually and intuitively for users. Accordingly, this research aims to design a forex prediction model for the EUR/USD currency pair using the LSTM method with a Data Science approach, compare the performance of LSTM models with ARIMA statistical models as a baseline using quantitative evaluation metrics, and develop an interactive web dashboard integrated with a prediction model to display historical data and forex forecasting results. The benefits of this research are twofold: academically, it strengthens the application of Machine Learning methods in digital finance based on Data Science, while practically, it produces a prototype of an interactive forex prediction system that can serve as a decision support system for traders and financial analysts in navigating the volatile forex market.

METHOD

Types and Approaches of Research

This research was a quantitative study using a computational experimental approach based on data science. This approach was chosen because the analysis and evaluation rely on numerical data that can be objectively measured and analyzed using statistical methods and machine learning.

The computational experiment involved building, training, and testing predictive models using Long Short-Term Memory (LSTM) as the primary model and Autoregressive Integrated Moving Average (ARIMA) as the baseline, allowing objective evaluation of model performance through specific error metrics.

This research also adopts a Predictive Analytics approach, as its main focus is predicting future EUR/USD exchange rates based on historical data.

Research Object

The object of this study is historical EUR/USD forex exchange rate data, chosen for its high liquidity, large trading volume, and availability of complete, continuous historical data.
The data used is **time series data** with the following main attributes:

1. *Open* (opening price)
2. *High* (highest price)
3. *Low* (lowest price)
4. *Close* (closing price)

Table 1. Description of Research Objects

Aspect	Information
Research Object	Historical data of EUR/USD forex exchange rates
Data Types	Secondary data
Data Form	Time series
Data Attributes	Date, Open, High, Low, Close, Change
Time Interval	Daily, Weekly, Monthly, Yearly
Data Period	Period 2012 - 2025
Data source	Kaggle and Oanda
Purpose of Use	Forex exchange rate prediction

Source: Authors' elaboration based on research design (2026)

The data represents forex price movements over a specific time interval and is used as the primary input in the modeling and prediction process.

Data Collection Methods

The data used is secondary data, comprising historical EUR/USD exchange rate data obtained from online financial market data sources that provide open, structured forex price data, such as international financial data provider platforms.

The historical data includes Open, High, Low, and Close (OHLC) prices arranged chronologically at daily intervals, chosen to balance price movement detail with data stability during modeling.

Table 2. Example of Dataset Used

Date	Price	Open	High	Low	Vol.	Change %
12/01/2016	10,513	10,589	10,874	10,352		-0.68%
11/01/2016	10,585	10,980	11,300	10,517		-3.59%
10/01/2016	10,979	11,236	11,245	10,849		-2.30%
01/09/2016	11,238	11,158	11,328	11,121		0.74%
01/08/2016	11,156	11,177	11,367	11,045		-0.13%
07/01/2016	11,170	11,106	11,198	10,951		0.59%
06/01/2016	11,104	11,132	11,435	10,912		-0.22%
01/05/2016	11,129	11,446	11,616	11,098		-2.84%
04/01/2016	11,454	11,380	11,466	11,215		0.67%

Source: Authors' compilation from Kaggle and Oanda historical forex data (2026)

Table 3. Dataset Structure

No	Column Name	Data Type	Information
1	Date	Date	Transaction date
2	Price	Float	Closing price (Close Price)
3	Open	Float	Opening price
4	High	Float	Highest price
5	Low	Float	Lowest price
6	Vol.	Numeric	Transaction volume
7	Change %	Percentage	Percentage change in price

Source: Authors' elaboration based on dataset structure (2026)

The data period used is 2012 to 2025, selected to cover diverse market conditions including stable periods and periods of high volatility so the prediction model can learn more diverse, representative price movement patterns.

The data collection stages are as follows:

1. Determine the currency pair being studied, namely EUR/USD.
2. Determine the time period of historical data used, namely 2012 – 2025.
3. Download historical forex price data according to the research period from Kaggle and Oanda.
4. Checking data completeness and consistency.
5. Store data in a structured format (e.g. CSV) to facilitate data processing.

Data Pre-processing

Data preprocessing ensures data quality before modeling and includes data cleaning, normalization, time series data generation, and data segmentation.

1. Data Cleaning

Data cleansing addresses missing values, duplicate data, and outliers; missing values are handled via interpolation or deletion depending on analysis needs.

The data cleaning stages this time include:

- Remove spaces from column names.
- Change the date format to datetime type.
- Change the **Price column** to a numeric (float) data type.
- Sort data by date chronologically.

```python
def load_data(path):

    df = pd.read_csv(path)

    df.columns = df.columns.str.strip()

    df["Date"] = pd.to_datetime(df["Date"])

    df = df.sort_values("Date")

    df["Price"] = (
        df["Price"]
        .astype(str)
        .str.replace(",", "")
        .astype(float)
    )

    return df
```

Figure 1. Data Cleaning Process

Source: Authors' documentation based on data preprocessing implementation (2026)

Table 4. Data After Cleaning Process

Date	Price	Date	Price
2016-04-01	11,454	12/01/2016	11,454
2016-05-01	11,129	11/01/2016	11,129
2016-06-01	11,104	10/01/2016	11,104
2016-09-01	11,238	07/01/2016	11,238
2016-10-01	10,979	06/01/2016	10,979
2016-11-01	10,585	01/05/2016	10,585
2016-12-01	10,513	04/01/2016	10,513
AFTER		BEFORE	

Source: Authors' calculation based on data preprocessing results (2026)

2. Data Normalization

Because forex price data has a wide range of values that varies across time periods, normalization using scaling techniques is applied to keep values within a specific range, typically 0 to 1, improving the stability and speed of LSTM model training.

```python

def normalize_data(data):

    scaler = MinMaxScaler(
        feature_range=(0,1)
    )

    scaled = scaler.fit_transform(
        data.reshape(-1,1)
    )

    return scaled, scaler
```

Figure 2. Data Normalization Process

Source: Authors' documentation based on data preprocessing implementation (2026)

Table 5. Data After Normalization Process

No	Original Price	Price Normalization
1	11,454	1,000
2	11,238	0.770
3	11,170	0.698
4	11,156	0.683
5	11,129	0.655
6	11,104	0.628
7	10,979	0.495
8	10,585	0.077
9	10,513	0.000

Source: Authors' calculation based on Min-Max Scaling normalization results (2026)

The normalization equation used is

$$X_{norm} \frac{X - X_{min}}{X_{max} - X_{min}} =$$

Example : $X_{norm} \frac{11.454 - 10.513}{11.454 - 10.513} = \frac{0.941}{0.941} = 1.000$

3. Time Series Data Formation

Normalized historical data is then transformed into time series form using the sliding window method, where a number of past data points are used as input to predict values in the next period, allowing the model to learn temporal patterns.

4. Data Sharing

The processed data was split chronologically into training (80%) and testing (20%) sets to preserve time order and prevent data leakage.

Predictive Modeling

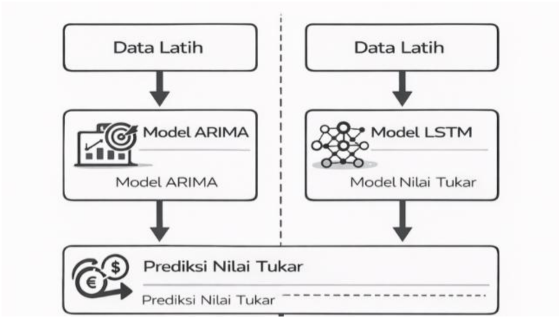

Figure 3. Prediction Modeling
Source: Authors' elaboration based on research methodology (2026)

1. Modeling Using LSTM

The Long Short-Term Memory (LSTM) model is the primary model in this study. LSTM modeling includes designing the network architecture, training the model on training data, and predicting test data, with the training objective of minimizing prediction error between actual and predicted values.

The LSTM model was built using TensorFlow and Keras.

The model architecture consists of:

- 2 Layer LSTM
- 2 Layer Dropout
- 1 Dense Layer

Table 6. LSTM Model Parameters

Parameter	Mark
Window Size	12
Epoch	50
Batch Size	8
Optimizer	Adam
Loss Function	Mean Squared Error
Dropout	0.2
Early Stopping	Patience = 10

Source: Authors' elaboration based on LSTM model configuration (2026)

Figure 4. LSTM Model Parameter Program
Source: Authors' documentation based on LSTM implementation using TensorFlow/Keras (2026)

2. Modeling Using ARIMA

The Autoregressive Integrated Moving Average (ARIMA) model is used as the baseline. ARIMA modeling involves testing data stationarity, differencing, and determining parameters p, d, and q. It is trained on the same data as the LSTM model to ensure an objective comparison.

Table 7. ARIMA Model Parameters

Parameter	Mark
p	1–5
d	1
q	0–5

Source: Authors' elaboration based on ARIMA model configuration (2026)

Figure 5. ARIMA Model Parameter Program

Source: Authors' documentation based on ARIMA implementation using Statsmodels (2026)

Model Evaluation Method

Model performance evaluation assessed the accuracy and reliability of predictions using the following metrics:

1. *Root Mean Squared Error (RMSE)*

   Measuring the magnitude of the prediction error: $RMSE = \sqrt{\frac{1}{n}\sum(yi - yi)\text{^}2}$

   The smaller the RMSE value, the better the model.

2. *Mean Absolute Error (MAE)*

   Measures the average absolute error. $MAE = \frac{1}{n}\sum | yi - yi |$

3. *Mean Absolute Percentage Error (MAPE)*

   Measuring the error rate in percentage form $MAPE = \frac{1}{n}\sum | \frac{yi-yi}{yi} |$

Figure 6. Metrix LSTM program

Source: Authors' documentation based on LSTM evaluation implementation (2026)

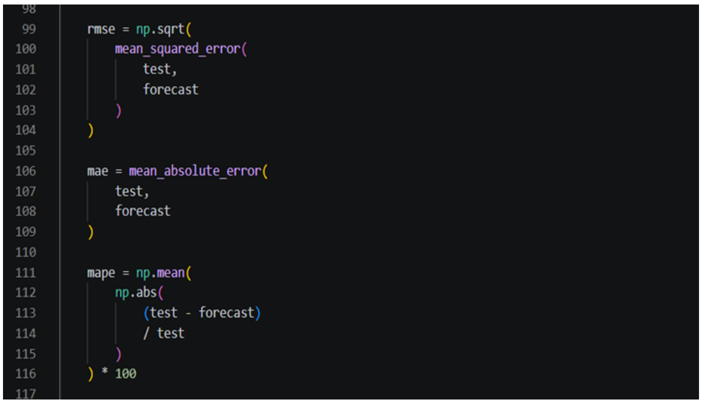

Figure 7. ARIMA Metrix Program

Source: Authors' documentation based on ARIMA evaluation implementation (2026)

The evaluation results are used to compare the performance of the LSTM and ARIMA models.

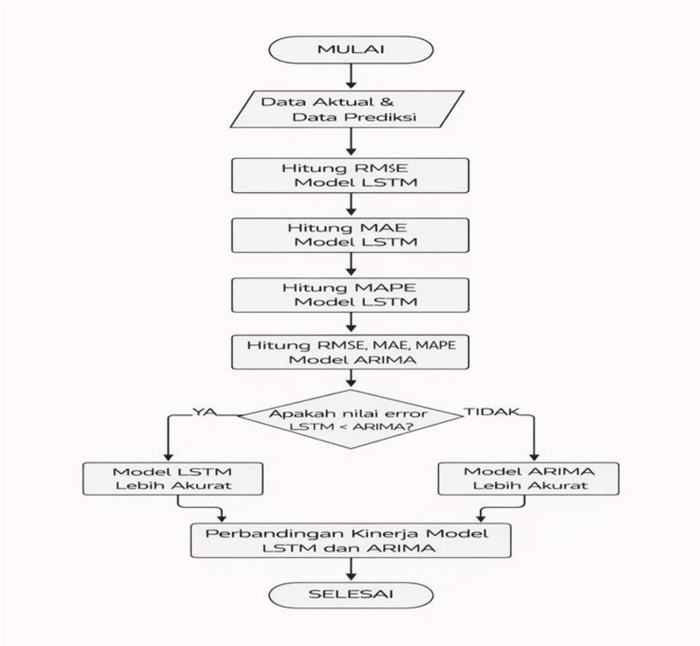

Figure 8. Evaluation Model

Source: Authors' elaboration based on research methodology (2026)

Web Dashboard Implementation

Prediction results from the LSTM and ARIMA models are integrated into a web dashboard for visualization, displaying historical data, prediction results, and a comparison between actual and predicted values in an easy-to-understand graphical form.

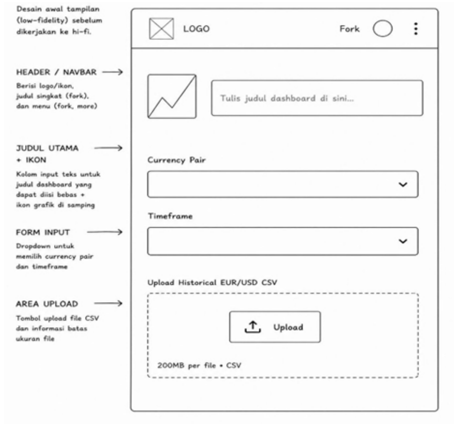

Figure 9. Dashboard Visualization I

Source: Authors' documentation based on web dashboard implementation using Streamlit (2026)

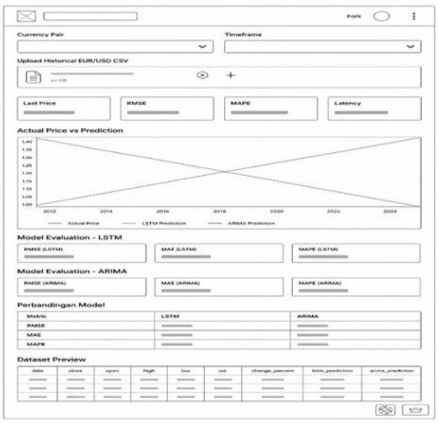

Figure 10. Dashboard II Visualization

Source: Authors' documentation based on web dashboard implementation using Streamlit (2026)

The dashboard displays key information, including time series graphs comparing actual prices with LSTM and ARIMA predictions, and model evaluation metrics (RMSE, MAE, MAPE) used to assess accuracy and reliability. It is intended not as a trading system but as a tool for visualizing and analyzing research results.

Software and Tools Used

The development of the forex prediction system in this research is supported by several software and tools as follows.

1. **Visual Studio Code (VS Code)**
   Used as *an Integrated Development Environment (IDE)* for system design and development processes.
2. **Python Programming Language**
   Used as the primary programming language in data processing, Machine Learning modeling, and dashboard development.
3. **TensorFlow / Keras**
   Used to build and train Long Short-Term Memory (LSTM) models.

4. **Stats models**
   Used to build Autoregressive Integrated Moving Average (ARIMA) models.
5. **Pandas and NumPy**
   Used for processing and manipulating forex time series data.
6. **Scikit-learn**
   Used for data normalization processes as well as calculating evaluation metrics such as RMSE, MAE, and MAPE.
7. **Streamlit**
   Used to build simple and interactive web-based forex prediction dashboards, including CSV data upload features.
8. **CSV Data Format**
   Used as an input data format to load historical forex data into the system.

RESULTS AND DISCUSSION

Implementation of the EUR/USD Forex Prediction System

This research produces a web-based EUR/USD exchange rate prediction system built using Python and the Streamlit framework, developed to implement the LSTM and ARIMA methods for predicting the EUR/USD exchange rate based on monthly historical data.

System implementation includes data collection, preprocessing, model training, evaluation, and visualization of results via a web dashboard. The dataset used is historical EUR/USD data on a monthly timeframe, cleaned and normalized before being used in model training.

The dashboard that was built provides several main features, namely:

1. Upload EUR/USD dataset.
2. Displays the prediction results using the LSTM model.
3. Displays prediction results using the ARIMA model.
4. Displays a comparison chart of actual prices and predicted results.
5. Shows the Training Loss and Validation Loss graphs of the LSTM model.
6. Displays the Residual Analysis graph on the ARIMA model.
7. Displays model evaluation values in the form of RMSE, MAE, and MAPE.
8. Displays the best model conclusion based on the evaluation results.

Figure 11. Web Dashboard I
Source: Authors' documentation based on web dashboard implementation (2026)

Figure 12. Web Dashboard I I
Source: Authors' documentation based on web dashboard implementation (2026)

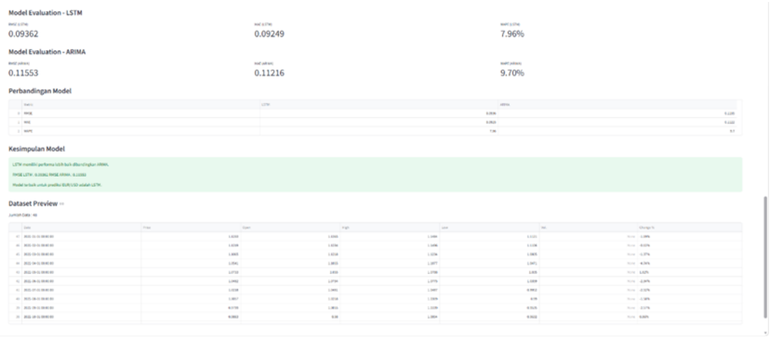

Figure 13. Web Dashboard I II
Source: Authors' documentation based on web dashboard implementation (2026)

This implementation simplifies analyzing and interpreting prediction results visually, helping users understand each model's performance more easily.

Data Preprocessing Results

The preprocessing stage improves data quality before model training and involves the following processes:

1. Data cleaning (data cleaning).
2. Date data type conversion.
3. Sorting data by time.
4. Convert price column to numeric.
5. Data normalization using the Min-Max Scaling method.

Normalization uses a value range of 0 to 1, making LSTM model training more stable and faster to converge.

The data was split into training (80%) and testing (20%) sets so the model could be evaluated on previously unseen data.

Test Results for the 2012–2016 Period

The first test used historical EUR/USD data from January 2012 to December 2016, totaling 60 monthly data points.

Figure 14. Actual Price vs Prediction Graph 2012 – 2016
Source: Authors' visualization based on model prediction results (2026)

Table 8. Test Results for the 2012–2016 Period

Method	RMSE	MAE	MAPE
LSTM	0.04068	0.03798	3.51%
ARIMA	0.03658	0.03133	2.31%

Source: Authors' calculation based on model evaluation results (2026)

Based on the evaluation results, the ARIMA model achieved lower RMSE, MAE, and MAPE values than LSTM, indicating that ARIMA produced more accurate predictions than LSTM during the 2012–2016 period.

This indicates the EUR/USD pattern during this period was relatively stable and linear, making it easier to capture with the ARIMA model.

Test Results for the 2017–2021 Period

The second test used historical EUR/USD data from January 2017 to December 2021, totaling 60 monthly data points.

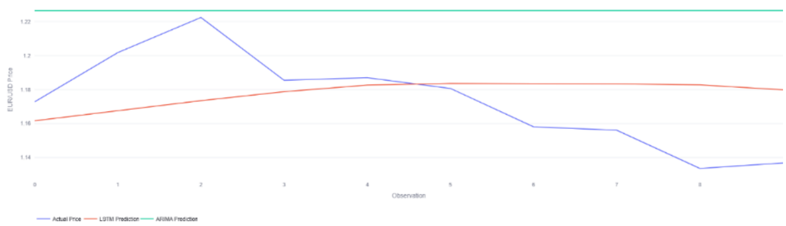

Figure 15. Actual Price vs Prediction Graph 2017 – 2021
Source: Authors' visualization based on model prediction results (2026)

Table 9. Test Results for the 2017–2021 Period

Method	RMSE	MAE	MAPE
LSTM	0.03826	0.03201	2.76%
ARIMA	0.05459	0.04684	4.03%

Source: Authors' calculation based on model evaluation results (2026)

The test results show that the LSTM model is still able to maintain better performance than the ARIMA model.

Error values increased compared to the previous period, indicating a more complex EUR/USD pattern in 2017–2021 that made prediction more challenging.

However, the MAPE value of 2.78% still shows that the LSTM model is capable of producing predictions with a good level of accuracy.

Test Results for the 2022–2025 Period

The third test used historical EUR/USD data from January 2022 to December 2025, totaling 48 monthly data points.

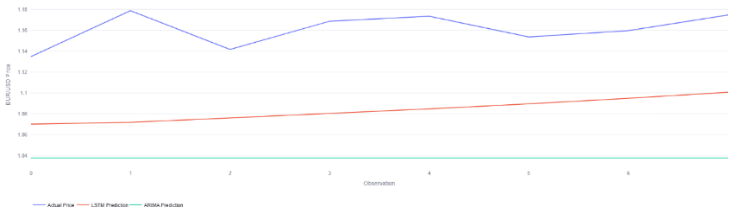

Figure 16. Actual Price vs Prediction Graph 2022 – 2025
Source: Authors' visualization based on model prediction results (2026)

Table 10. Test Results for the 2022–2025 Period

Method	RMSE	MAE	MAPE
LSTM	0.08790	0.08660	7.45%
ARIMA	0.11553	0.11216	9.70%

Source: Authors' calculation based on model evaluation results (2026)

Both models showed reduced performance compared with earlier testing, though LSTM still produced lower error values than ARIMA.

This increase in errors may reflect more volatile global market conditions, making historical patterns harder for both models to learn.

System Functional Test Results

Table 11. System Functional Test Results

NO	Feature	Results
1	Upload Dataset	Succeed
2	Data Preprocessing	Succeed
3	LSTM Prediction	Succeed
4	ARIMA Prediction	Succeed
5	Actual vs Prediction Graph	Succeed
6	Training Loss Graph	Succeed
7	ARIMA Residual Graph	Succeed
8	RMSE Evaluation	Succeed
9	MAE Evaluation	Succeed
10	MAPE Evaluation	Succeed

Source: Authors' observation based on system functional testing (2026)

Functional testing confirmed that all dashboard features worked as required, with no functional errors detected.

Analysis of LSTM and ARIMA Prediction Results

Based on the three tests conducted, the research results show that model performance is greatly influenced by the characteristics of the data used.

In 2012–2016, ARIMA performed better than LSTM, indicating the EUR/USD pattern was still fairly linear and suited to a statistical approach.

Conversely, in the 2017–2021 and 2022–2025 periods, LSTM produced more accurate predictions than ARIMA. This reflects LSTM's ability to handle time series data with non-linear patterns and long-term dependencies relevant to forex data, where exchange rate changes are influenced by both short-term conditions and longer-term historical patterns.

This shows that LSTM's ability to capture non-linear relationships provides an advantage when market patterns become more complex.

Analysis of Actual Price vs Prediction Chart

The Actual Price vs Prediction chart is used to evaluate the model's ability to follow the actual EUR/USD exchange rate movement pattern.

In the 2012–2016 period, the ARIMA prediction curve stayed close to actual data, consistent with its lower error rate for that period.

In the 2017–2021 and 2022–2025 testing periods, however, the pattern changed: the LSTM prediction curve followed the direction of actual data movement more closely than ARIMA. Although differences between predicted and actual values remained, the LSTM trend was relatively closer to the actual data pattern.

In contrast, ARIMA models tend to produce flatter curves and are less responsive to rapid price changes. This leads to increased ARIMA error values in data with higher volatility.

This visualization supports the RMSE, MAE, and MAPE results, confirming LSTM's better ability to follow EUR/USD movement under dynamic market conditions.

Training Loss and Validation Loss Analysis

The Training Loss and Validation Loss graphs are used to evaluate the training process of the LSTM model over several epochs.

In all tests, training loss decreased gradually in early epochs before stabilizing, indicating the model successfully learned patterns in the training data.

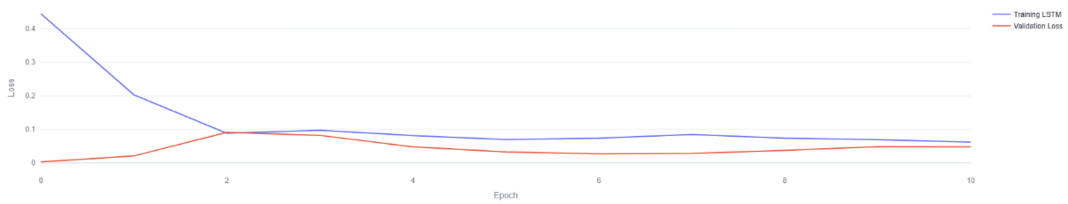

Figure 17. Training vs Validation Loss Graph 2012-2016

Source: Authors' visualization based on LSTM model training results (2026)

Furthermore, the validation loss values also showed a stable trend without any significant increase. This indicates that the model is not experiencing excessive overfitting.

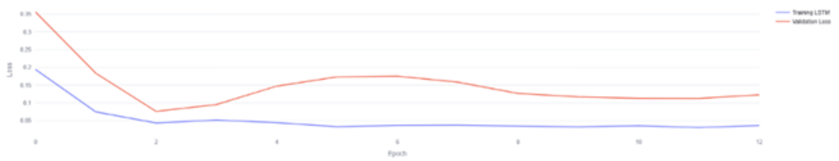

Figure 18. Training vs Validation Loss Graph 2017-2021
Source: Authors' visualization based on LSTM model training results (2026)

In the 2017–2021 period, the small gap between training and validation loss indicated good generalization to unseen data.

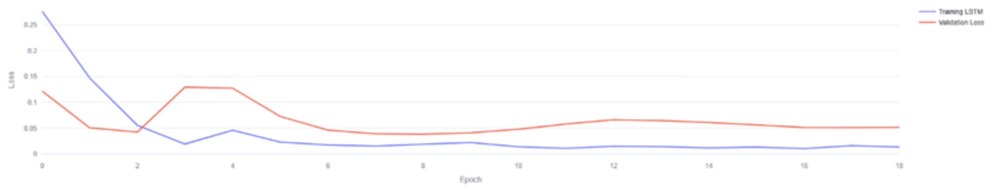

Figure 19. Training vs Validation Loss Graph 2022-2025
Source: Authors' visualization based on LSTM model training results (2026)

In the 2022–2025 period, validation loss rose slightly in several epochs, reflecting more complex data patterns that made learning more challenging.

Overall, the Training Loss and Validation Loss results show LSTM learned EUR/USD historical patterns well and produced a stable model.

ARIMA Residual Analysis

The residual graph shows residuals fluctuating around zero, with some notably large positive and negative values.

Figure 20. Residual Graph 2012-2016
Source: Authors' visualization based on ARIMA model residual analysis (2026)

In 2012–2016, smaller residuals produced lower error values, indicating ARIMA successfully captured most data patterns.

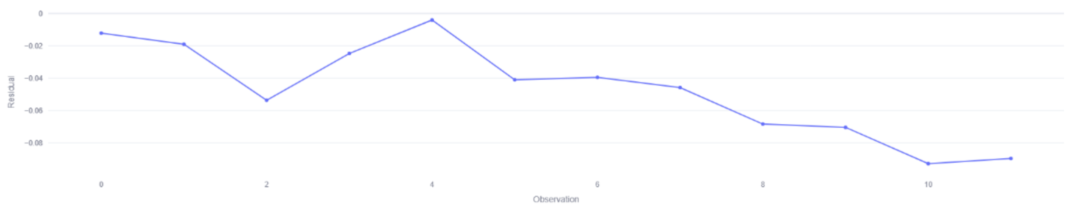

Figure 21. Residual Graph 2017-2021

Source: Authors' visualization based on ARIMA model residual analysis (2026)

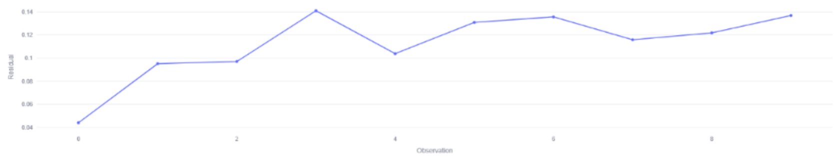

Figure 22. Residual Graph 2022-2025

Source: Authors' visualization based on ARIMA model residual analysis (2026)

In 2017–2021 and especially 2022–2025, residuals fluctuated more, indicating certain patterns could not be optimally modeled by ARIMA.

The larger the residuals, the greater the difference between predicted and actual data. The increase in residuals during the last two periods is therefore one reason for the increase in RMSE and MAE values in the ARIMA model.

Comparative Analysis of Evaluation Metrics

Model performance was compared using three evaluation metrics: RMSE, which imposes a larger penalty on extreme errors; MAE, which measures average absolute error; and MAPE, which measures the error rate as a percentage.

Based on the test results, the RMSE values obtained are as follows:

Table 12. Results of RMSE Testing Between Periods

Period	LSTM RMSE	ARIMA RMSE
2012 – 2016	0.04068	0.03658
2017 – 2021	0.03856	0.05459
2022 - 2025	0.09362	0.11553

Source: Authors' calculation based on model evaluation results (2026)

The average RMSE of the LSTM model is:

$$\frac{0{,}04068 + 0{,}03856 + 0{,}09362}{3} = 0{,}05762$$

Meanwhile, the average RMSE of the ARIMA model is:

$$\frac{0{,}03658 + 0{,}05459 + 0{,}11553}{3} = 0{,}06890$$

The average RMSE shows LSTM has a lower overall error rate than ARIMA.

All MAPE values remained below 10%, which by general forecast-accuracy classification indicates excellent model accuracy.

Analysis of Differences in Results Between Periods

The difference in model performance in each period shows that data characteristics have a large influence on the prediction results.

The 2012–2016 period yielded the lowest error values due to relatively stable EUR/USD movement, conditions under which ARIMA performed well given its near-linear data pattern.

Data complexity increased in 2017–2021 due to various global economic factors, allowing LSTM to begin outperforming ARIMA.

The 2022–2025 period yielded the highest error values for both models, reflecting more volatile market conditions and harder-to-predict historical patterns.

Achievement of Research Objectives

This research succeeded in achieving all the objectives set out in Chapter I.

The first objective building a web dashboard-based EUR/USD prediction system was achieved via implementation using Python and Streamlit.

The second objective applying LSTM and ARIMA to historical EUR/USD data was also achieved, with both models performing training, prediction, and evaluation automatically through the dashboard.

The third objective, namely comparing the performance of the two methods, was successfully achieved through three different test scenarios.

Results show ARIMA outperformed during 2012–2016, while LSTM outperformed during 2017–2021 and 2022–2025. LSTM therefore appears more suitable for predicting EUR/USD movements under complex, dynamic market conditions, while ARIMA is more suitable for relatively stable data.

These findings indicate that model choice depends not only on the algorithm but significantly on the characteristics of the data analyzed. LSTM is therefore recommended as the primary approach for developing a EUR/USD forex prediction system based on monthly historical data.

CONCLUSION

Based on the research conducted on this Data Science-based EUR/USD prediction system integrating Machine Learning and a web dashboard, it can be concluded that this research successfully built a web-based EUR/USD exchange rate prediction system using Python and the Streamlit framework, with both LSTM and ARIMA methods effectively applied to historical monthly data to generate exchange rate predictions based on historical patterns. Test results reveal that model performance is significantly influenced by data characteristics, where ARIMA performed better during the 2012–2016 period with an RMSE of 0.03658 compared to LSTM's 0.04068, indicating ARIMA's effectiveness on relatively stable data patterns, while LSTM demonstrated superior performance in the 2017–2021 and 2022–2025 periods with RMSE values of 0.03856 versus 0.05459 and 0.09362 versus 0.11553 respectively, confirming LSTM's ability to better capture non-linear patterns and complex trend changes in forex data. Overall, LSTM can be considered superior to ARIMA, having produced the best performance in two of the three experiments conducted, while the dashboard successfully displays Actual Price vs. Prediction graphs, Training Loss and Validation Loss graphs, ARIMA Residual Analysis graphs, RMSE/MAE/MAPE values, and model performance comparisons, serving as an effective analytical tool for evaluating EUR/USD exchange rate predictions. This research demonstrates that Machine Learning methods combined with web dashboards can effectively build a forex prediction system that provides predictive information in an accessible way for users. For future research, it is recommended to explore hybrid models combining LSTM with other deep learning architectures such as GRU or Transformer, incorporate additional fundamental and macroeconomic indicators to improve prediction accuracy, extend the prediction timeframe to intraday or weekly intervals, and enhance the dashboard with real-time data integration and automated retraining capabilities for more adaptive and responsive forex prediction systems.

REFERENCE

Banco de México. (2022). *Bank for International Settlements (BIS) 2022 triennial central bank survey of foreign exchange and OTC derivatives markets, April 2022* (Press release). https://www.banxico.org.mx/markets/central-bank-surveys-of-foreign-exchange-and-deriv/%7B27700CB5-1211-0722-096A-40D2E04CCB7F%7D.pdf

Bormpotsis, C., Sedky, M., & Patel, A. (2023). Predicting forex currency fluctuations using a novel bio-inspired modular neural network. *Big Data and Cognitive Computing, 7*(3). https://doi.org/10.3390/bdcc7030152

Bousqaoui, H., Slimani, I., & Achchab, S. (2021). Comparative analysis of short-term demand predicting models using ARIMA and deep learning. *International Journal of Electrical and Computer Engineering, 11*(4), 3319. https://doi.org/10.11591/ijece.v11i4.pp3319-3330

Carney, M. (2019). The growing challenges for monetary policy in the current international monetary and financial system. In *Proceedings of the Jackson Hole Economic Symposium* (pp. 377–411).

Cofnas, A. (2015). *The forex trading course: A self-study guide to becoming a successful currency trader*. John Wiley & Sons.

Gamaliy, V., et al. (2018). Exchange rates: The influence of political and economic events. A fundamental analysis approach. *Banks and Bank Systems, 13*(4), 131–142. https://doi.org/10.21511/bbs.13(4).2018.12

Hu, Z., Zhao, Y., & Khushi, M. (2021). A survey of forex and stock price prediction using deep learning. *Applied System Innovation, 4*(1), 9. https://doi.org/10.3390/asi4010009

Kadiri, E., & Alabi, O. (2015). Importance of technical and fundamental analysis in the foreign exchange market. *British Journal of Economics, Management & Trade, 5*, 181–194.

Kontopoulou, V. I., Panagopoulos, A. D., Kakkos, I., & Matsopoulos, G. K. (2023). A review of ARIMA vs. machine learning approaches for time series forecasting in data driven networks. *Future Internet, 15*(8), 255. https://doi.org/10.3390/fi15080255

Lamani, L., Leka, E., Peposhi, I., & Aliti, A. (2025). A Bi-LSTM and technical analysis-based framework for financial forecasting on Albanian forex markets. In *International Conference on Intelligence-Based Transformations of Technology and Business Trends* (pp. 282–299). Springer.

Nguyen, D. K., Sermpinis, G., & Stasinakis, C. (2023). Big data, artificial intelligence and machine learning: A transformative symbiosis in favour of financial technology. *European Financial Management, 29*(2), 517–548. https://doi.org/10.1111/eufm.12386

Nwaimo, C. S., Oluoha, O. M., & Oyedokun, O. (2019). Big data analytics: Technologies, applications, and future prospects. *Iconic Research and Engineering Journals, 2*(11), 411–419.

Odunaike, A. (2025). Integrating real-time financial data streams to enhance dynamic risk modeling and portfolio decision accuracy. *International Journal of Computer Applications Technology and Research, 14*(8), 1–16.

Ogunbiyi-Badaru, O., Alao, O. B., Dudu, O. F., & Alonge, E. O. (2024). *The impact of FX and fixed income integration on global financial stability: A comprehensive analysis* (Unpublished manuscript).

Ozdemir, O., Biehl, M., & Petkov, N. (2026). A survey of fundamental indicators, machine learning techniques and feature extraction techniques for stock price prediction. *Discover Artificial Intelligence*.

Saberironaghi, M., Ren, J., & Saberironaghi, A. (2025). Stock market prediction using machine learning and deep learning techniques: A review. *AppliedMath, 5*(3), 76.

Saghafi, A., Bagherian, M., & Shokoohi, F. (2025). Forecasting Forex EUR/USD closing prices using a dual-input deep learning model with technical and fundamental indicators. *Mathematics, 13*(9), 1472. https://doi.org/10.3390/math13091472

Tuarob, S., et al. (2021). DAViS: A unified solution for data collection, analyzation, and visualization in real-time stock market prediction. *Financial Innovation, 7*(1), 56. https://doi.org/10.1186/s40854-021-00268-8

Walter, T., & Wansleben, L. (2020). How central bankers learned to love financialization: The Fed, the Bank, and the enlisting of unfettered markets in the conduct of monetary policy. *Socio-Economic Review, 18*(3), 625–653. https://doi.org/10.1093/ser/mwz054

Wang, T., & Tobias, G. R. (2025). Research on intelligent optimization mechanisms of financial process modules through machine learning-enhanced collaborative systems in digital finance platforms. *Future Technology, 4*(4), 240–254.

Wilson, A., & Anwar, M. R. (2024). The future of adaptive machine learning algorithms in high-dimensional data processing. *International Transactions on Artificial Intelligence, 3*(1), 97–107.

Yıldırım, D. C., Toroslu, I. H., & Fiore, U. (2021). Forecasting directional movement of forex data using LSTM with technical and macroeconomic indicators. *Financial Innovation, 7*(1), 1–36. https://doi.org/10.1186/s40854-020-00220-2