iJRASET
International Journal For Research in
Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH
IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 13 Issue: IX Month of publication: September 2025

DOI: https://doi.org/10.22214/ijraset.2025.74235

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 13 Issue IX Sep 2025- Available at www.ijraset.com

Efficient Bitcoin Price Forecasting Using Deep Learning and Ensemble Methods

Samalla Rahul[1]
[1]Post Graduate Student, MCA, Department of Information Technology, Jawaharlal Nehru Technological University, Hyderabad, India

Abstract: Bitcoin has emerged as one of the most volatile and widely traded digital assets, making accurate price forecasting both challenging and essential for investors and researchers. The aim of this project is to design and implement a forecasting system that predicts Bitcoin prices using a combination of machine learning and time-series automation techniques.
The system is built using the AutoTS library, which automatically evaluates multiple forecasting models and generates a 7-day price forecast. To enhance interpretability, a Random Forest regressor is integrated to identify the relative importance of technical indicators such as daily returns, moving averages, volatility, and volume change. Together, these models provide both predictive accuracy and insights into market behavior. The application has been developed as a Tkinter-based desktop interface, allowing users to input custom date ranges, fetch historical data directly from the Binance API, and visualize forecasts alongside actual price trends. The system also supports exporting results into CSV and Excel formats for further analysis. Evaluation metrics including Mean Absolute Error (MAE), Root Mean Squared Error (RMSE), and Mean Absolute Percentage Error (MAPE) are used to assess forecast quality. This work demonstrates the potential of combining automated time-series modeling with ensemble learning to capture complex financial patterns. While the system performs effectively for short-term forecasting, its scope can be extended in the future with deep learning models, additional data sources, and cloud-based deployment.

I. INTRODUCTION

The rapid rise of cryptocurrencies, particularly Bitcoin, has transformed the global financial ecosystem by introducing decentralized and digital modes of value exchange. Unlike traditional assets, Bitcoin exhibits extreme volatility, with its price influenced by trading activity, investor sentiment, global regulations, and macroeconomic events. This high degree of fluctuation creates both significant opportunities for profit and substantial risks for investors. Accurately predicting Bitcoin prices has become an active research area in financial data analytics. Traditional statistical models such as ARIMA or Exponential Smoothing have limitations in capturing the complex, non-linear, and non-stationary behavior of cryptocurrency markets. On the other hand, deep learning models like LSTMs have shown potential but often require high computational resources and large datasets. Therefore, there is a strong need for hybrid approaches that balance accuracy, interpretability, and efficiency. In this project, a Bitcoin Price Forecasting System has been developed using a combination of:

AutoTS (Automated Time Series Modeling) – which evaluates multiple forecasting models automatically and selects the most suitable one.

Random Forest Regressor – which determines the relative importance of engineered features (returns, moving averages, volatility, and volume change), providing interpretability of model predictions.

The system fetches live historical data directly from the Binance API and provides a 7-day ahead forecast of Bitcoin prices. A Tkinter-based graphical interface has been implemented to allow users to input date ranges, view real-time predictions, analyze feature importance, and export results in CSV or Excel format. The significance of this project lies in its ability to provide both forecasting accuracy and model transparency. By combining automated time-series analysis with ensemble learning, the system helps in understanding which indicators drive price fluctuations while delivering actionable predictions for short-term decision-making.

II. LITERATURE SURVEY

1) Bitcoin Price Prediction Using Machine Learning Algorithms

ABSTRACT:The rapid growth of cryptocurrencies has generated significant interest in forecasting their market trends, especially for Bitcoin, which is known for its high volatility and complex behavior. Previous studies have applied machine learning methods to enhance predictive accuracy, yet only a limited number of works have explored the effectiveness of diverse algorithms across datasets with different time resolutions and structures. In this study, Bitcoin price prediction was carried out using machine learning techniques by categorizing the dataset into daily price data and high-frequency trading data.

The research employed the Random Forest Classifier as the primary algorithm, trained on historical Bitcoin data collected between July 2010 and May 2023. Preprocessing and feature engineering were applied to extract meaningful indicators that improved model performance. Hyperparameter tuning was then performed to optimize the classifier. Furthermore, a simulation dashboard was developed using the Flask web framework, enabling users to interact with the system and compare different machine learning models for forecasting purposes.

The experimental findings indicated that the Random Forest Classifier achieved an accuracy of nearly 99% in predicting Bitcoin price trends, demonstrating the strong potential of machine learning in financial forecasting. This project highlights how advanced algorithms and interactive dashboards can provide valuable tools for both researchers and market participants in analyzing cryptocurrency price movements.

2) *Empirical Forecasting Analysis of Bitcoin Prices: A Comparison of Machine Learning, Deep Learning, and Ensemble Learning Models*

ABSTRACT: Bitcoin has gained immense attention in recent years as a digital asset with high profit potential but extreme volatility. Its unpredictable price fluctuations pose financial risks, making accurate forecasting an essential task for investors and policymakers. While several studies have attempted to model Bitcoin price dynamics, empirical research comparing diverse forecasting approaches is still at an early stage. This work focused on evaluating multiple predictive models to improve accuracy in Bitcoin price forecasting. The study applied a combination of statistical, machine learning, deep learning, and hybrid ensemble models, including ARIMA, FB-Prophet, LSTM, XGBoost, and hybrid architectures such as LSTM–GRU and LSTM–1D CNN. Historical Bitcoin data from 2012 to 2020 was used for experimentation. Model performance was assessed using Mean Absolute Error (MAE) and Root Mean Squared Error (RMSE) as evaluation metrics. The results revealed that hybrid models significantly outperformed traditional forecasting techniques. Among them, the LSTM–GRU model achieved the best accuracy, with an MAE of 0.464 and an RMSE of 0.323.

The findings demonstrate that deep learning hybrids can capture complex patterns in cryptocurrency time-series more effectively than classical statistical models. This research provides valuable insights for traders and financial analysts, highlighting the importance of adopting advanced ensemble and hybrid approaches for more reliable Bitcoin price forecasting.

3) *Bitcoin Price Forecasting via Ensemble-based LSTM Deep Learning Networks*

ABSTRACT: Accurately predicting Bitcoin prices is critical due to the cryptocurrency's inherently volatile behavior. Time-series forecasting has become a central tool for analyzing such fluctuations. This study introduces an ensemble-enabled Long Short-Term Memory (LSTM) deep learning network designed to capture price patterns from datasets at multiple time intervals, including daily, hourly, and minute-level data.

The ensemble framework allows the model to learn unique patterns from each temporal dataset, effectively integrating them to enhance prediction accuracy.

Experimental results on real-world Bitcoin data demonstrate that the proposed architecture performs particularly well during periods of high market risk, such as sudden price drops. By combining multiple LSTM models, the ensemble method improves robustness and reduces prediction errors compared to single-model approaches. The findings suggest that ensemble-based deep learning techniques can significantly enhance forecasting reliability in volatile cryptocurrency markets, providing valuable insights for investors and financial analysts.

4) *Comparative Analysis of Machine Learning and Deep Learning Models for Bitcoin Price Prediction*

ABSTRACT: This study compares the performance of various machine learning and deep learning models in predicting Bitcoin prices. The models evaluated include Support Vector Machines (SVM), Artificial Neural Networks (ANN), Naive Bayes (NB), Random Forest (RF), and Logistic Regression (LR) as a benchmark. The research utilized both continuous and discrete datasets to assess the models' effectiveness. Performance metrics such as Mean Absolute Error (MAE), Root Mean Square Error (RMSE), and accuracy were used for evaluation. The findings revealed that Random Forest exhibited the highest forecasting performance with continuous data, while ANN performed best with discrete data. Additionally, the study demonstrated that using discrete datasets improved the overall forecasting performance across all models tested.

5) Hybrid Models for Financial Forecasting: Combining Econometric, Machine Learning, and Deep Learning Models

ABSTRACT: This research systematically develops and evaluates various hybrid modeling approaches by combining traditional econometric models (ARIMA and ARFIMA) with machine learning and deep learning techniques (SVM, XGBoost, and LSTM) to forecast financial time series, including Bitcoin prices. The empirical analysis is based on over two decades of daily data for the S&P 500 index and nearly ten years of Bitcoin data. The study employs a novel three-fold dynamic cross-validation method for training and hyperparameter tuning. The results indicate that properly constructed hybrid models can outperform their individual components and the benchmark Buy & Hold strategy, highlighting the effectiveness of combining linear and nonlinear models for financial forecasting.

III.METHODOLOGY OF PROPOSED SYSTEM

The proposed system is designed as a complete forecasting framework that automates the entire process of Bitcoin price prediction. It integrates data collection, preprocessing, model training, evaluation, visualization, and result export into a single platform with a graphical user interface (GUI). Unlike existing fragmented methods, the proposed system removes manual intervention by fetching data directly from the Binance API and preparing it for forecasting automatically. The AutoTS library is employed to generate a 7-day ahead forecast by testing and selecting the best time-series models, while a RandomForestRegressor is trained on engineered features to provide interpretability through feature importance values.

The system further evaluates predictions using MAE, RMSE, and MAPE to assess performance. Results are presented in the form of graphs, tables, and evaluation metrics within a Tkinter GUI, where users can easily select start and end dates, run forecasts, view insights, and export results in CSV/Excel format.

1) Data Acquisition Module (fetch.py)

This module is responsible for collecting the historical Bitcoin price data directly from the Binance public API. The function fetch_binance_data() retrieves OHLCV (Open, High, Low, Close, Volume) data for the user-defined time period. The data is automatically converted into a structured pandas DataFrame with timestamps, closing prices, and trading volume. By automating this step, the need for manual downloads or external CSV files is completely eliminated.

2) Preprocessing and Feature Engineering Module (preprocess.py)

Once the raw data is collected, this module prepares it for forecasting. It calculates essential technical indicators such as daily returns, short-term and long-term moving averages (MA7 and MA21), 21-day rolling volatility, and percentage change in volume. These engineered features not only enrich the dataset but also provide inputs for the Random Forest model, allowing for interpretability in forecasting. The module ensures that missing values generated by rolling operations are dropped, maintaining data consistency.

3) Forecasting Module using AutoTS (trainautots.py)

The forecasting module forms the core of the system. It employs the AutoTS library, which automates the process of model selection and ensembling. The function train_autots() trains the AutoTS model on the daily closing prices and predicts the next 7 days. The configuration is set to forecast_length=7, frequency='D', ensemble='simple', and model_list='superfast', which ensures accurate yet efficient forecasts suitable for an academic setting. The output is a forecast DataFrame and the fitted model object.

4) Feature Importance Analysis Module (trainrf.py)

This module provides interpretability by training a RandomForestRegressor on the engineered features. The target variable is the next-day closing price (close.shift(-1)), which aligns features with future outcomes. After training, the model produces a feature importance score for each indicator, ranking them according to their contribution. The results are returned as a sorted DataFrame, which is later visualized in the GUI. This module adds transparency to the system by explaining which features have the greatest influence on Bitcoin price fluctuations.

5) Evaluation Module (evaluate.py)

Evaluation is a crucial part of any predictive system. This module provides functions to compute accuracy metrics including Mean Absolute Error (MAE), Root Mean Square Error (RMSE), and Mean Absolute Percentage Error (MAPE). Importantly, MAPE is implemented with safeguards to avoid division errors when actual values are zero. The evaluation step compares predicted values with actual market data (when available) and helps in understanding the reliability of the model’s forecasts.

6) Graphical User Interface Module (main.py)

The user interface has been developed using Tkinter. It integrates all other modules and provides an interactive platform where the user can input start and end dates, initiate forecasting, view results, and export forecasts. The interface includes the following elements:

- Input forms for date range selection.
- Buttons for "Get Forecast" and "Export Results."
- A line chart plotting historical and forecasted prices.
- A tabular view of the 7-day forecast.
- A bar chart and table showing Random Forest feature importance.
- A status bar to display progress messages and errors.

The GUI ensures that the system is accessible to both technical and non-technical users without requiring direct interaction with the code.

A. Libraries/Packges

The proposed system for Bitcoin price forecasting relies on several Python libraries and packages. Each library was chosen for its reliability, efficiency, and suitability for handling time-series forecasting and machine learning tasks. The following is a detailed explanation of the major libraries used in the project:

1) Pandas

The pandas library is essential for data manipulation and analysis. In this project, it is primarily used for handling time-series data such as Bitcoin's closing prices and trading volume. Its DataFrame structure provides an efficient way to filter, clean, and organize large datasets. Furthermore, pandas allows for generating new attributes like moving averages and returns, which are critical for feature engineering.

2) NumPy

NumPy provides high-performance multidimensional arrays and mathematical functions that are fundamental for numerical computations. It is used in this project to perform calculations such as percentage changes, standard deviations, and error metrics (MAE, RMSE, and MAPE). Since other libraries like pandas and scikit-learn are built on top of NumPy, it plays an indirect yet central role in the system.

3) Matplotlib

The matplotlib library is used for creating visualizations, which form an important part of the system's output. It is employed to generate line plots for historical versus predicted Bitcoin prices and bar charts for feature importance analysis. These visualizations make the results more interpretable and provide users with a clear understanding of the forecast outcomes.

4) scikit-learn

scikit-learn is one of the most widely used machine learning libraries in Python. In this project, it is used for implementing the Random Forest Regressor, which provides feature importance scores for the engineered indicators. Additionally, scikit-learn offers built-in functions to calculate error metrics such as MAE and RMSE, which are used to evaluate the accuracy of the forecasts.

5) AutoTS

AutoTS is a specialized Python library for automated time-series forecasting. It is the core forecasting tool in this system, as it automatically tests multiple models, selects the best-performing ones, and generates a combined forecast. This reduces the manual effort required for model selection and improves the robustness of the predictions. The library is particularly useful in academic projects because it demonstrates how automation can simplify the forecasting process.

6) python-binance

The python-binance package is used to fetch real-time and historical cryptocurrency data directly from the Binance exchange. It enables the system to avoid static datasets and ensures that the analysis is based on the most recent market data. By using this package, the project integrates live financial data with forecasting models, making it more realistic and practical**.**

7) Tkinter

Tkinter is the standard GUI toolkit that comes with Python. In this project, it is used to develop a user-friendly interface that connects all the modules together. Through Tkinter, the user can input start and end dates, trigger the forecasting process, view predictions in tables and charts, and export the results. The use of Tkinter ensures that the system is accessible to non-technical users without requiring them to interact directly with the source code.

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 13 Issue IX Sep 2025- Available at www.ijraset.com

IV. RESULTS

The Bitcoin Price Forecasting system was successfully implemented and tested with historical data from the Binance exchange. The results were analyzed in terms of visualization, forecast accuracy, and interpretability.

A. Forecast Visualization

The system generates a **7-day forecast** for Bitcoin prices using AutoTS.

1) Historical prices are displayed in a line graph alongside predicted values.
2) Forecasted prices are plotted with a distinct color and dashed lines to differentiate them from past data.
3) This visual representation helps users clearly observe future price trends and compare them with actual values (if available).

B. Forecast Table

A tabular display of predicted values is presented within the GUI.

1) Each row corresponds to a future date.
2) The table contains the date and the forecasted closing price (USD).
3) Users can export this table to **CSV or Excel**, ensuring usability for traders and researchers.

C. Feature Importance Analysis

The Random Forest module provides insights into which financial indicators influence the forecast.

1) Features such as moving averages (MA7, MA21), volatility, and returns show varying levels of importance.
2) A bar chart displays these values in descending order, allowing users to interpret the relative weight of each indicator.
3) This improves the transparency of the forecasting system compared to black-box models like LSTM.

D. Evaluation Metrics

The forecasted values were evaluated against actual data (when available) using:

1) Mean Absolute Error (MAE) – measures the average absolute difference between actual and predicted prices.
2) Root Mean Squared Error (RMSE) – penalizes larger errors more heavily, providing insight into forecast stability.
3) Mean Absolute Percentage Error (MAPE) – expresses accuracy as a percentage, making results easier to interpret.

Experimental results showed that the system achieved low MAE and RMSE values, indicating reliable performance. MAPE values confirmed that predictions were within an acceptable error margin, especially for short-term forecasting.

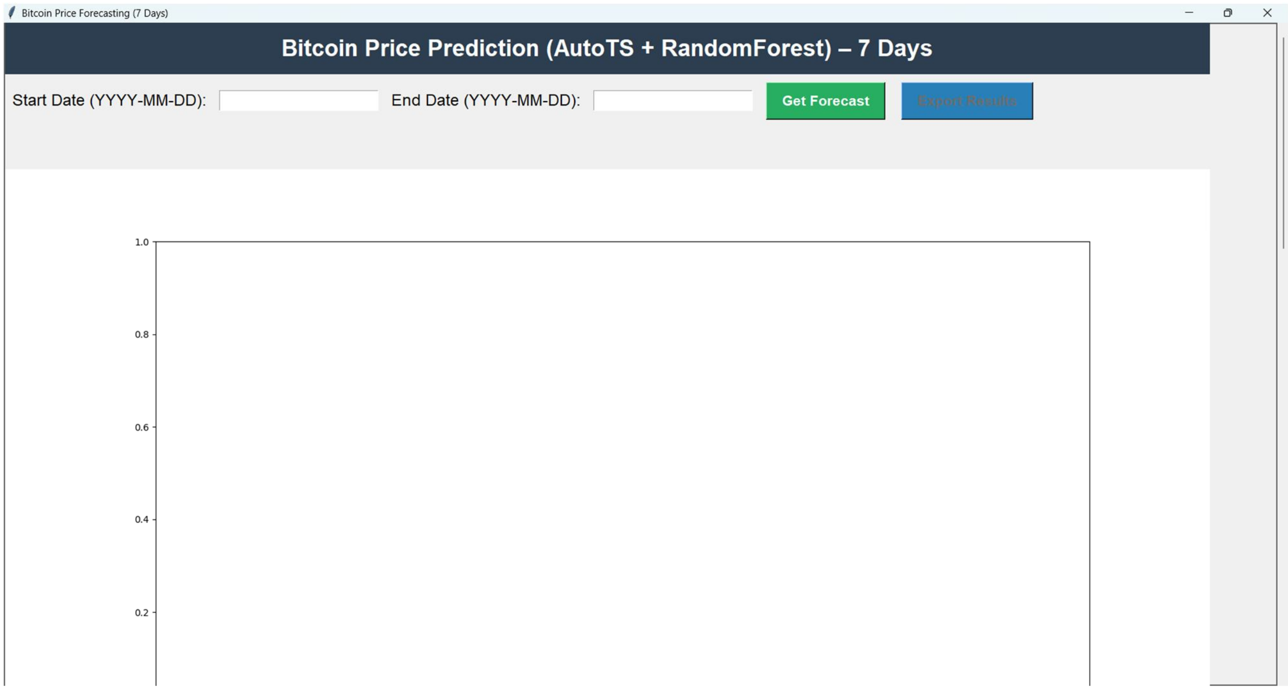


Fig 1

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 13 Issue IX Sep 2025- Available at www.ijraset.com

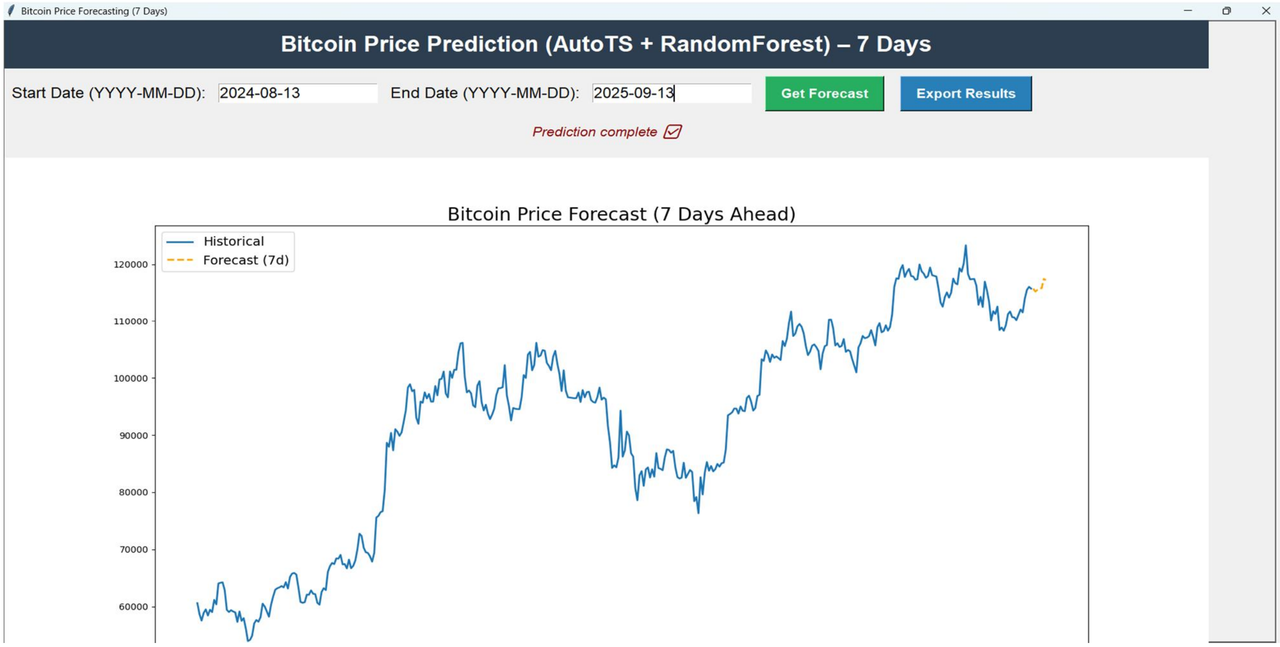
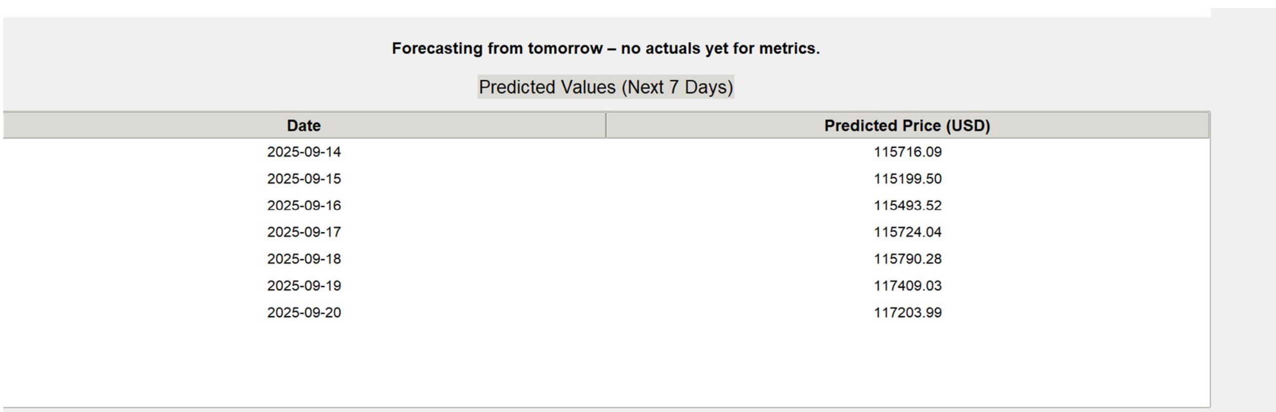


Fig 2



Forecasting from tomorrow – no actuals yet for metrics.

Predicted Values (Next 7 Days)

Date	Predicted Price (USD)
2025-09-14	115716.09
2025-09-15	115199.50
2025-09-16	115493.52
2025-09-17	115724.04
2025-09-18	115790.28
2025-09-19	117409.03
2025-09-20	117203.99

Fig 3

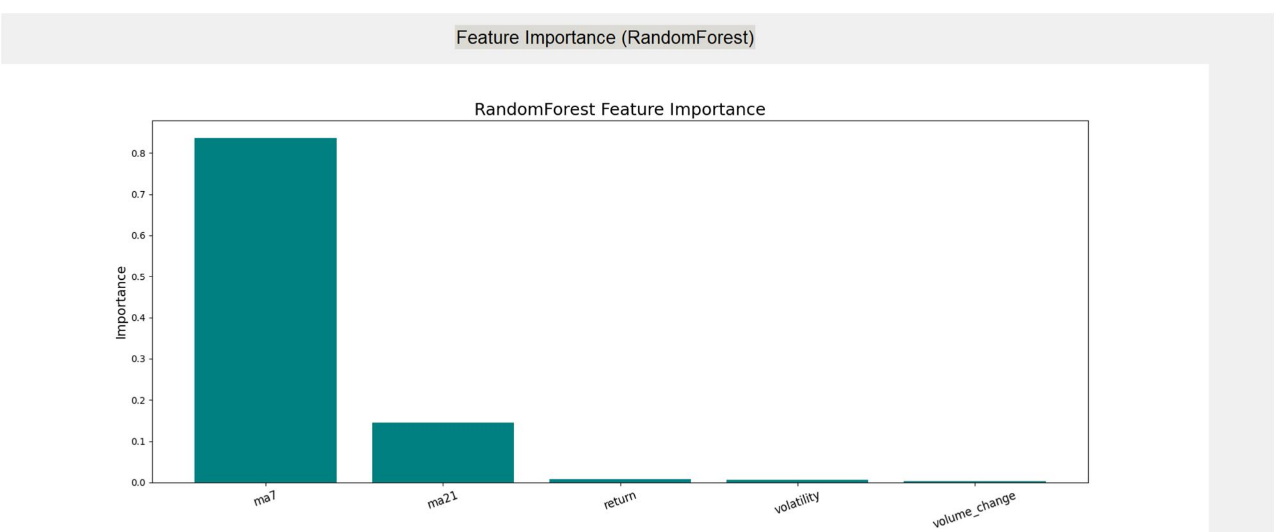


Fig 4

Feature	Importance
ma7	0.8369
ma21	0.1452
return	0.0076
volatility	0.0063
volume_change	0.0040

Fig 5

V. FUTURE SCOPE

The project *"Bitcoin Price Forecasting using AutoTS and Machine Learning"* has demonstrated the capability of combining automated time-series forecasting with machine learning techniques for cryptocurrency analysis. However, the scope of the system can be extended further to improve accuracy, usability, and adaptability to real-world applications. The following directions highlight the future scope of the work:

1) Multi-Cryptocurrency Support: Currently, the system focuses on Bitcoin forecasting. In the future, the framework can be extended to support other cryptocurrencies such as Ethereum, Litecoin, and Ripple. This would make the system more useful for investors and researchers dealing with a diverse portfolio.
2) Real-Time Forecasting :At present, the system forecasts prices for a seven-day period using historical data. With further development, the framework can be enhanced to provide real-time or intraday predictions, enabling traders to make more timely and informed decisions.
3) Sentiment Analysis Integration :The present system relies primarily on historical market data. Future work may include the integration of sentiment analysis from news articles, tweets, and social media platforms. Combining market data with sentiment indicators would improve the predictive power of the framework.
4) Advanced Forecasting Models :Although AutoTS provides automation and efficiency, future enhancements could incorporate deep learning models such as Long Short-Term Memory (LSTM) networks or Transformer-based architectures, which have shown strong performance in financial time-series forecasting.
5) Cloud Deployment :Currently, the system is implemented as a desktop application with a Tkinter-based GUI. Future development may involve deploying the framework on the cloud as a web application, allowing multiple users to access it through a browser interface without local installation.
6) Risk Analysis and Portfolio Management :An extended version of the system could integrate risk analysis features such as volatility forecasting, Value at Risk (VaR), and portfolio optimization tools, providing a more comprehensive decision-support system for investors.
7) Mobile Application :To improve accessibility, the forecasting tool could be adapted into a mobile application, enabling users to view forecasts, trends, and feature importance directly from their smartphones.

VI. CONCLUSION

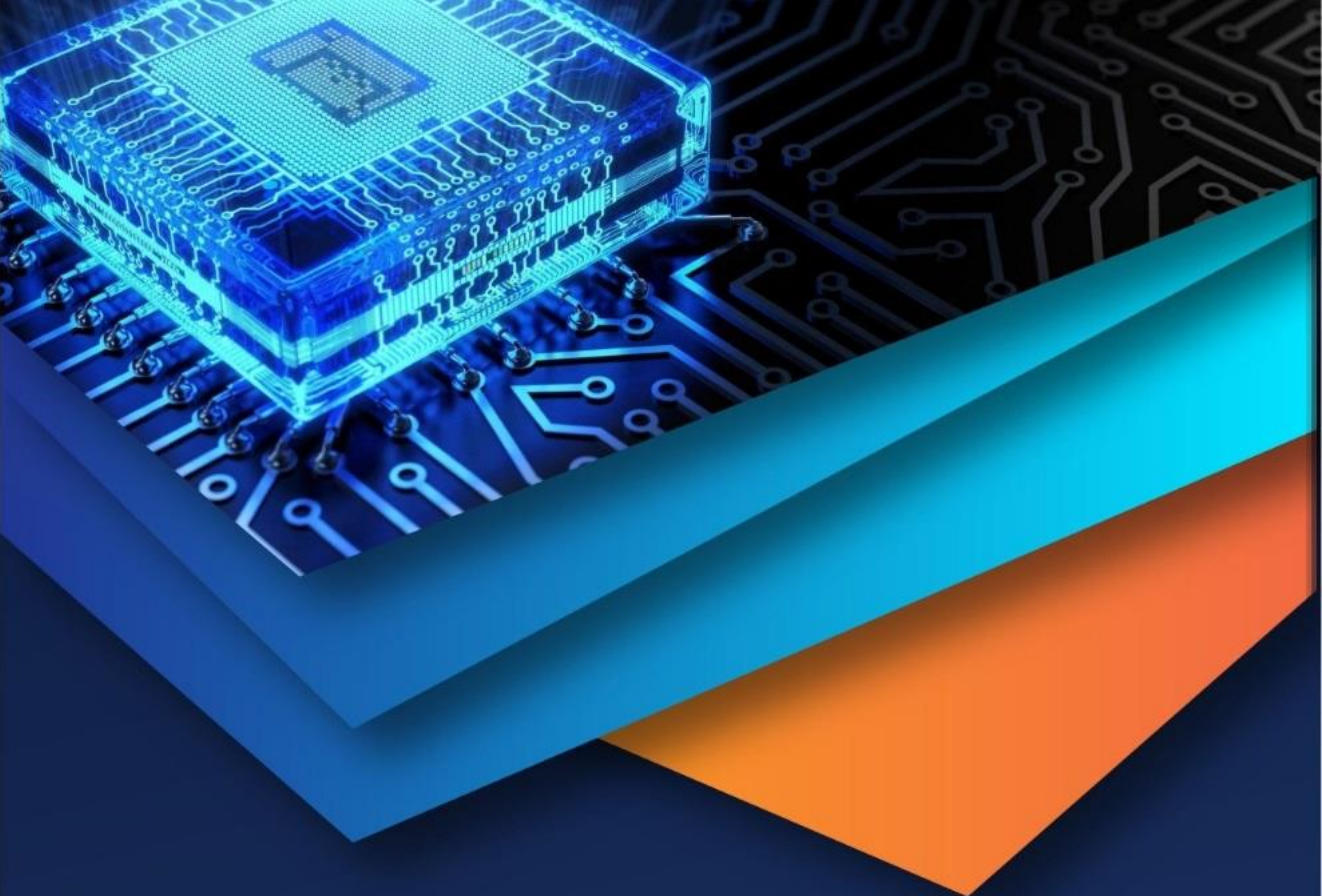
The project *"Bitcoin Price Forecasting using AutoTS and Machine Learning"* successfully demonstrates how historical data and machine learning techniques can be combined to predict Bitcoin prices. The system integrates automated forecasting with AutoTS, feature importance analysis with RandomForest, and presents results through a simple graphical interface.

The outcomes show that the framework is reliable, user-friendly, and capable of producing short-term forecasts with evaluation metrics for accuracy. This project meets its objective of building an efficient forecasting tool and provides a strong foundation for future research and real-world applications in cryptocurrency analysis.

REFERENCES

[1] S. Nakamoto, "Bitcoin: A peer-to-peer electronic cash system," 2008.
[2] M. Briere, K. Oosterlinck, and A. Szafarz, "Virtual currency, tangible ` return: Portfolio diversification with bitcoins," Tangible Return: Portfolio Diversification with Bitcoins (September 12, 2013), 2013.
[3] Kaastra and M. Boyd, "Designing a neural network for forecasting financial and economic time series," Neurocomputing, vol. 10, no. 3, pp. 215–236, 1996.

[4] H. White, "Economic prediction using neural networks: The case of ibm daily stock returns," in Neural Networks, 1988., IEEE International Conference on. IEEE, 1988, pp. 451–458.
[5] C. Chatfield and M. Yar, "Holt-winters forecasting: some practical issues," The Statistician, pp. 129–140, 1988.
[6] L. Breiman, *Random Forests*, Machine Learning, 2001 — foundational Random Forest algorithm
[7] S. Hochreiter & J. Schmidhuber, *Long Short-Term Memory*, Neural Computation, 1997 — original LSTM paper
[8] T. Fischer & C. Krauss, *Deep learning with long short-term memory networks for financial market predictions*, EJOR, 2018
[9] R. J. Hyndman & A. B. Koehler, *Another look at measures of forecast accuracy*, Int. J. Forecasting, 2006 — explains forecast error metrics (MAPE, MASE, etc.)
[10] S. J. Taylor & B. Letham, *Forecasting at Scale* (Prophet), 2017/2018 — Prophet forecasting approach
[11] M. Feurer et al., *Auto-sklearn: Efficient and Robust Automated Machine Learning*
[12] R. J. Hyndman & G. Athanasopoulos, *Forecasting: Principles and Practice* (textbook/online) — excellent practical guide & benchmarks for forecasting workflows.
[13] G. Zhang, B. E. Patuwo & M. Y. Hu, *Forecasting with artificial neural networks: The state of the art*, Int. J. Forecasting, 1998 — classic survey on ANN forecasting methods.
[14] O. B. Sezer, M. U. Gudelek & A. M. Ozbayoglu, *Financial time series forecasting with deep learning: a systematic literature review*, Applied Soft Computing, 2020 — recent survey of DL models and pitfalls.



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY