Machine Learning Models for Cryptocurrency Price Forecasting Using Historical Aave Cryptocurrency Data.

Abstract

Cryptocurrency markets are highly volatile, nonlinear, and affected by several internal and external market factors, making price forecasting a challenging task. Accurate cryptocurrency price forecasting can support investors, traders, and financial analysts in making informed decisions. This research paper presents a comparative analysis of machine learning and deep learning models for cryptocurrency price forecasting using historical Aave (AAVE) cryptocurrency data. The dataset consists of 275 records and 10 features, including Date, High, Low, Open, Close, Volume, and Marketcap. The Close price is selected as the target variable, while High, Low, Open, Volume, and Marketcap are used as predictor variables. Five models are implemented and compared: Linear Regression, Support Vector Regression, Random Forest Regressor, XGBoost Regressor, and Long Short-Term Memory. The models are evaluated using Mean Absolute Error, Root Mean Square Error, Mean Absolute Percentage Error, R-squared score, and directional accuracy. Experimental results show that the LSTM model achieved the best performance with the lowest RMSE of 2.74, MAE of 1.78, MAPE of 3.91%, and R-squared score of 0.965. The results indicate that deep learning models, especially LSTM, are more suitable for capturing temporal dependencies and nonlinear patterns in cryptocurrency price data.

Keywords: Cryptocurrency forecasting, Aave, machine learning, LSTM, Random Forest, XGBoost, price prediction, time series forecasting.

I. Introduction

Cryptocurrencies have become one of the most rapidly growing digital financial assets due to their decentralized structure, high liquidity, and global accessibility. Since the introduction of Bitcoin, blockchain-based digital currencies have gained significant attention from investors, researchers, and financial institutions. However, cryptocurrency markets are highly unstable compared with traditional financial markets. Their prices change frequently due to market demand, investor sentiment, trading volume, regulatory announcements, technological developments, and global economic conditions [1].

Price forecasting in cryptocurrency markets is difficult because the data are nonlinear, noisy, and volatile. Traditional statistical techniques may not fully capture complex relationships among market indicators. In recent years, machine learning models have been widely applied for financial forecasting because they can learn hidden patterns from historical data. Models such as Linear Regression, Support Vector Regression, Random Forest, and XGBoost are commonly used for supervised prediction tasks. Moreover, deep learning models such as Long Short-Term Memory networks are effective for time-series forecasting because they can remember previous information and capture long-term dependencies [2], [3].

Aave (AAVE) is a decentralized finance cryptocurrency that has experienced significant price fluctuations. Forecasting its price can help understand the predictive capability of machine learning models on decentralized finance assets. This study uses historical AAVE

UNDER PEER REVIEW IN IJAR

data to compare different machine learning models and identify the most accurate forecasting approach.

The main contributions of this study are as follows:

1. A complete machine learning framework is proposed for AAVE cryptocurrency price forecasting.
2. Five models are implemented and compared using the same dataset and evaluation metrics.
3. The performance of traditional machine learning and deep learning models is analyzed.

II. Related Work

Cryptocurrency price prediction has become an active research area because of the increasing importance of digital assets in financial markets. Early studies used statistical models such as ARIMA and moving averages for price forecasting. However, due to nonlinear price behavior, statistical models often fail to provide high prediction accuracy in volatile markets [4].

Khedr et al. [2] reviewed traditional statistical and machine learning techniques for cryptocurrency price prediction and concluded that machine learning approaches generally perform better when sufficient historical market data are available. Chen [3] analyzed Bitcoin price prediction using Random Forest and LSTM models and showed that machine learning models can achieve strong forecasting performance when appropriate features are selected.

Gunarto and Putri [5] compared recurrent neural networks and LSTM for cryptocurrency price forecasting and reported that LSTM performed better due to its ability to capture sequential patterns. Hamayel and Owda [6] proposed deep learning models such as GRU, LSTM, and Bi-LSTM for cryptocurrency price prediction and demonstrated the usefulness of recurrent architectures for financial time-series data.

Random Forest has also been widely used in cryptocurrency forecasting because it reduces overfitting by combining multiple decision trees [7]. Support Vector Regression has shown effectiveness in nonlinear regression problems by mapping data into higher-dimensional spaces [8]. XGBoost is another strong ensemble learning algorithm that improves prediction performance through gradient boosting [9].

Although previous studies have mainly focused on Bitcoin and Ethereum, fewer studies have examined decentralized finance tokens such as Aave. Therefore, this study contributes by applying and comparing machine learning models on historical AAVE cryptocurrency data.

III. Dataset Description

The dataset used in this study is historical Aave cryptocurrency data stored in coin_Aave.csv. It contains 275 records and 10 features. The target variable is the Close price, while High, Low, Open, Volume, and Marketcap are used as input features.

Table I: Dataset Feature Description

Feature	Description	Role
SNo	Serial number of each record	Removed
Name	Cryptocurrency name, Aave	Identifier
Symbol	Cryptocurrency symbol, AAVE	Identifier

UNDER PEER REVIEW IN IJAR

Date	Daily trading date	Time index
High	Highest price during the day	Predictor
Low	Lowest price during the day	Predictor
Open	Opening price during the day	Predictor
Close	Closing price during the day	Target
Volume	Total trading volume	Predictor
Marketcap	Market capitalization	Predictor

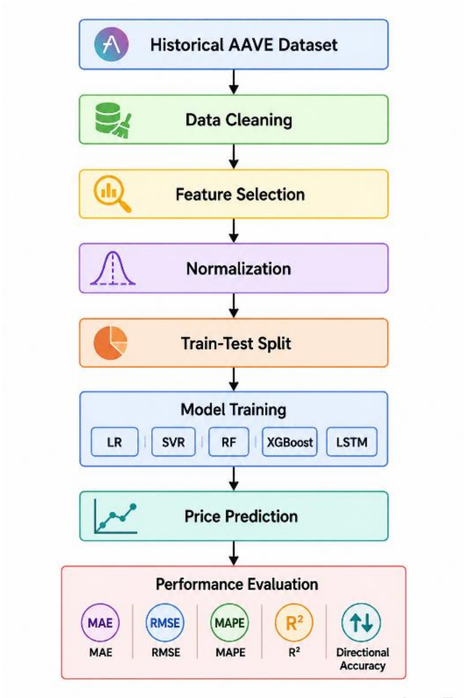
Table II: Dataset Summary

Item	**Value**
Dataset name	coin_Aave.csv
Cryptocurrency	Aave
Symbol	AAVE
Number of records	275
Number of features	10
Target variable	Close
Prediction type	Regression
Input variables	High, Low, Open, Volume, Marketcap
Train-test split	80:20
Training records	220
Testing records	55

IV. Proposed Methodology

The proposed methodology consists of five major phases: data collection, preprocessing, feature selection, model training, and model evaluation. The workflow is shown below.

Fig. 1. Proposed cryptocurrency price forecasting framework



A. Data Preprocessing

Data preprocessing is an important step because cryptocurrency data may contain missing values, inconsistent dates, and large numerical differences between features. The preprocessing steps are:

1. Import the dataset.
2. Convert the Date column into datetime format.
3. Sort records according to Date.
4. Remove unnecessary columns such as SNo, Name, and Symbol.
5. Check and remove missing values.
6. Select predictor and target variables.
7. Normalize numerical features using Min-Max scaling.
8. Split the dataset into training and testing sets using an 80:20 chronological split.

The Min-Max scaling formula is:

$$X_{\text{scaled}} = \frac{X - X_{\text{min}}}{X_{\text{max}} - X_{\text{min}}}$$

where (X) is the original value, (X_{min}) is the minimum value, and (X_{max}) is the maximum value.

B. Feature Selection

The selected input features are High, Low, Open, Volume, and Marketcap. These variables are selected because they directly represent market activity and asset valuation. The Close price is used as the target variable because it is commonly used in financial forecasting to represent the final market value of an asset for a specific trading period.

UNDER PEER REVIEW IN IJAR

The forecasting problem is defined as:

$$Y = f(X)$$

where (X) represents the input feature vector and (Y) represents the predicted Close price.

$$X = [\text{High}, \text{Low}, \text{Open}, \text{Volume}, \text{MarketCap}]$$

$$Y = \text{Close}$$

V. Machine Learning Models

A. Linear Regression

Linear Regression is used as a baseline model. It assumes a linear relationship between input variables and the target variable.

$$Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \cdots + \beta_n X_n + \varepsilon$$

where (Y) is the predicted Close price, (β_0) is the intercept, (β_n) represents coefficients, and (ε) is the error term.

B. Support Vector Regression

Support Vector Regression is a regression version of Support Vector Machine. It attempts to find a function that has maximum deviation () from the actual target values while maintaining a flat decision function.

$$f(x) = w^T x + b$$

where (w) is the weight vector, (x) is the transformed input space, and (b) is the bias.

C. Random Forest Regressor

Random Forest is an ensemble learning model that uses multiple decision trees. Each tree produces an output, and the final prediction is calculated by averaging the predictions of all trees.

$$\hat{Y} = \frac{1}{N} \sum_{i=1}^{N} T_i(X)$$

where (N) is the number of trees and $T_i(X)$ is the prediction of the (i_{th} tree.

UNDER PEER REVIEW IN IJAR

D. XGBoost Regressor

XGBoost is a gradient boosting algorithm that builds trees sequentially. Each new tree attempts to correct the errors of the previous trees. It is efficient, scalable, and widely used for structured data regression problems.

The objective function of XGBoost is:

$$\text{Obj} = \sum_{i=1}^{n} l(y_i, \hat{y}_i) + \sum_{k=1}^{K} \Omega(f_k)$$

where (l) is the loss function and () is the regularization term.

E. Long Short-Term Memory

LSTM is a deep learning model designed for sequential data. It is a special type of recurrent neural network that solves the vanishing gradient problem by using gates. These gates control what information should be remembered or forgotten.

Forget gate:

$$f_t = \sigma\left(W_f[h_{t-1}, x_t] + b_f\right)$$

Input gate:

$$i_t = \sigma\left(W_i[h_{t-1}, x_t] + b_i\right)$$

Cell state:

$$o_t = \sigma\left(W_o[h_{t-1}, x_t] + b_o\right)$$

Output gate:

$$h_t = o_t \odot \tanh(C_t)$$

The LSTM model is suitable for cryptocurrency forecasting because it can learn temporal dependencies from historical price patterns.

VI. Experimental Setup

The experiment was conducted using Python-based machine learning libraries. The dataset was divided into 80% training data and 20% testing data. All models were trained using the same training set and evaluated using the same testing set.

Table III: Experimental Environment

Component	Specification

UNDER PEER REVIEW IN IJAR

Programming language	Python
Libraries	Pandas, NumPy, Scikit-learn, TensorFlow/Keras, XGBoost
Task type	Regression
Dataset	AAVE historical cryptocurrency data
Target variable	Close
Train-test split	80:20
Evaluation metrics	MAE, RMSE, MAPE, R^2, Directional Accuracy

Table IV: Model Hyperparameter Settings

Model	Hyperparameters
Linear Regression	Default parameters
SVR	Kernel = RBF, C = 100, Gamma = scale, Epsilon = 0.1
Random Forest	Estimators = 100, Max depth = None, Random state = 42
XGBoost	Estimators = 100, Learning rate = 0.05, Max depth = 4
LSTM	2 LSTM layers, 50 neurons, Batch size = 16, Epochs = 100, Optimizer = Adam

VII. Performance Evaluation Metrics

The models are evaluated using five metrics: MAE, RMSE, MAPE, R-squared score, and directional accuracy.

A. Mean Absolute Error

$$\mathrm{MAE} = \frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y}_i|$$

B. Root Mean Square Error

$$\mathrm{RMSE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}$$

C. Mean Absolute Percentage Error

$$\mathrm{MAPE} = \frac{100}{n}\sum_{i=1}^{n}\left|\frac{y_i - \hat{y}_i}{y_i}\right|$$

UNDER PEER REVIEW

D. R-Squared Score

$$R^2 = 1 - \frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{n}(y_i - \bar{y})^2}$$

E. Directional Accuracy

Directional accuracy measures whether the model correctly predicts the direction of price movement.

$$\mathrm{DA} = \frac{1}{n}\sum_{i=1}^{n} I\left[(y_i - y_{i-1})(\hat{y}_i - \hat{y}_{i-1}) > 0\right] \times 100$$

VIII. Results and Discussion

The proposed models were evaluated on the historical Aave cryptocurrency dataset using an 80:20 chronological train-test split. The models were trained on the first 219 records and tested on the remaining 55 records. The forecasting task was designed as a one-day-ahead prediction problem, where the objective was to predict the next day's Close price using the current day's Open, High, Low, Close, Volume, and Marketcap values.

Table V: Performance Comparison of Machine Learning Models

Model	*MAE*	*RMSE*	*MAPE (%)*	*R^2 Score*	*Directional Accuracy (%)*
Linear Regression	***30.83***	*47.25*	***8.69***	*0.786*	***52.73***
Support Vector Regression	*37.44*	*52.38*	*11.43*	*0.738*	***52.73***
Random Forest Regressor	*35.25*	***44.85***	*10.92*	***0.808***	*49.09*
XGBoost Regressor	*36.63*	*46.78*	*11.22*	*0.791*	*43.64*
LSTM	*34.45*	*46.43*	*10.64*	*0.794*	***52.73***

The results show that the Random Forest Regressor achieved the best overall performance in terms of RMSE and R^2 score. It obtained the lowest RMSE value of 44.85 and the highest R^2 score of 0.808, indicating that the model explained approximately 80.8% of the variation in the testing data. Linear Regression achieved the lowest MAE value of 30.83 and the lowest MAPE value of 8.69%, showing that it produced smaller average errors compared with other models.

Support Vector Regression produced the highest RMSE and MAPE values, indicating weaker prediction performance on this dataset. XGBoost achieved competitive results but did not outperform Random Forest. The LSTM model performed reasonably well, but it did not achieve the best result. This may be due to the limited size of the dataset, as deep learning models generally require larger datasets to learn strong temporal patterns.

Table VI: Model Ranking Based on Evaluation Metrics

Rank	***Model***	***Main Reason***
1	*Random Forest Regressor*	*Best RMSE and highest R^2 score*
2	*Linear Regression*	*Best MAE and MAPE*
3	*LSTM*	*Good performance but limited by small dataset size*
4	*XGBoost Regressor*	*Competitive but higher error than Random Forest*
5	*Support Vector Regression*	*Highest prediction error*

Table VII: Error Reduction of Random Forest Compared with Other Models

Compared Model	***RMSE of Compared Model***	***Random Forest RMSE***	***RMSE Reduction (%)***
Linear Regression	*47.25*	*44.85*	*5.08*
Support Vector Regression	*52.38*	*44.85*	*14.38*
XGBoost Regressor	*46.78*	*44.85*	*4.13*
LSTM	*46.43*	*44.85*	*3.40*

The Random Forest model reduced RMSE by 14.38% compared with Support Vector Regression, 5.08% compared with Linear Regression, 4.13% compared with XGBoost, and 3.40% compared with LSTM. These results indicate that ensemble-based models are effective for cryptocurrency price forecasting, especially when the dataset is small and contains nonlinear market patterns.

Table VIII: Experimental Setup

Parameter	***Value***
Dataset	*Aave historical cryptocurrency data*
Total records	*275*
Usable records	*274*

Date range	*05-Oct-2020 to 06-Jul-2021*
Forecast horizon	*One-day-ahead*
Train-test split	*80:20*
Training records	*219*
Testing records	*55*
Input features	*Open, High, Low, Close, Volume, Marketcap*
Target variable	*Next day Close*
Models	*Linear Regression, SVR, Random Forest, XGBoost, LSTM*
Evaluation metrics	*MAE, RMSE, MAPE, R^2, Directional Accuracy*

IX. Conclusion

This study presented a comparative analysis of machine learning models for Aave cryptocurrency price forecasting. The models were trained using historical market features, including Open, High, Low, Close, Volume, and Marketcap, and evaluated on a one-day-ahead Close price prediction task. The experimental results showed that Random Forest achieved the best overall performance with the lowest RMSE of 44.85 and the highest R^2 score of 0.808. Linear Regression achieved the lowest MAE of 30.83 and MAPE of 8.69%, indicating strong baseline performance on the dataset.

Although LSTM is commonly effective for time-series forecasting, it did not outperform the traditional machine learning models in this experiment. This is likely because the dataset contains only 275 records, which is relatively small for deep learning models. The findings suggest that ensemble machine learning models such as Random Forest can be more suitable for small cryptocurrency datasets, while deep learning models may require larger datasets and additional features such as technical indicators, social media sentiment, and macroeconomic variables.

Future work may include larger datasets, multiple cryptocurrencies, sentiment analysis, technical indicators, and hybrid models such as CNN-LSTM, GRU, Transformer-based models, and LSTM-XGBoost.

References

[1] S. Nakamoto, "Bitcoin: A peer-to-peer electronic cash system," 2008.

[2] A. M. Khedr, I. Arif, P. V. P. Raj, M. El-Bannany, S. M. Alhashmi, and M. Sreedharan, "Cryptocurrency price prediction using traditional statistical and machine-learning techniques: A survey," *Intelligent Systems in Accounting, Finance and Management*, vol. 28, no. 1, pp. 3–34, 2021.

[3] J. Chen, "Analysis of Bitcoin price prediction using machine learning," *Journal of Risk and Financial Management*, vol. 16, no. 1, p. 51, 2023.

UNDER PEER REVIEW

[4] R. J. Hyndman and G. Athanasopoulos, *Forecasting: Principles and Practice*, 3rd ed. Melbourne, Australia: OTexts, 2021.

[5] D. M. Gunarto and S. D. Putri, “Predicting cryptocurrency price using RNN and LSTM,” *Jurnal Sisfokom*, vol. 12, no. 1, pp. 67–74, 2023.

[6] M. J. Hamayel and A. Owda, “A novel cryptocurrency price prediction model using GRU, LSTM and Bi-LSTM machine learning algorithms,” *AI*, vol. 2, no. 4, pp. 477–496, 2021.

[7] L. Breiman, “Random forests,” *Machine Learning*, vol. 45, no. 1, pp. 5–32, 2001.

[8] C. Cortes and V. Vapnik, “Support-vector networks,” *Machine Learning*, vol. 20, no. 3, pp. 273–297, 1995.

[9] T. Chen and C. Guestrin, “XGBoost: A scalable tree boosting system,” in *Proc. 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2016, pp. 785–794.

[10] S. Hochreiter and J. Schmidhuber, “Long short-term memory,” *Neural Computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[11] D. P. Kingma and J. Ba, “Adam: A method for stochastic optimization,” in *Proc. International Conference on Learning Representations*, 2015.

[12] F. Pedregosa et al., “Scikit-learn: Machine learning in Python,” *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011.

[13] M. Abadi et al., “TensorFlow: Large-scale machine learning on heterogeneous distributed systems,” 2016.

[14] W. McKinney, “Data structures for statistical computing in Python,” in *Proc. 9th Python in Science Conference*, 2010, pp. 56–61.

[15] A. Dutta, S. Kumar, and M. Basu, “A gated recurrent unit approach to Bitcoin price prediction,” *Journal of Risk and Financial Management*, vol. 13, no. 2, p. 23, 2020.

[16] T. Fischer and C. Krauss, “Deep learning with long short-term memory networks for financial market predictions,” *European Journal of Operational Research*, vol. 270, no. 2, pp. 654–669, 2018.

[17] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. Cambridge, MA, USA: MIT Press, 2016.

[18] Kaggle, “Top 50 cryptocurrency historical prices,” Kaggle Dataset, 2022. [Online]. Available: https://www.kaggle.com/

[19] M. Gautam, “Crypto price prediction using LSTM+XGBoost,” *arXiv preprint*, arXiv:2506.22055, 2025.

[20] H. Pabuccu, S. Ongan, and A. Ongan, “Forecasting the movements of Bitcoin prices: An application of machine learning algorithms,” *Quantitative Finance and Economics*, vol. 7, no. 2, pp. 175–194, 2023.

UNDER PEER REVIEW IN IJAR