Journal of Applied Informatics and Computing (JAIC)
Vol.9, No.5, October 2025, pp. 2089~2101
e-ISSN: 2548-6861

A Hybrid Data Science Framework for Forecasting Bitcoin Prices using Traditional and AI Models

Puguh Hiskiawan [1]*, Jovan William [2]*, Louis Feliepe Tio Jansel [3]*
Data Science, Faculty of Technology and Design, Bunda Mulia University
phiskiawan@bundamulia.ac.id [1], s36240018@student.ubm.ac.id [2], s36240023@student.ubm.ac.id[3]

Article Info

Article history:

Received 2025-08-02
Revised 2025-08-24
Accepted 2025-09-03

Keyword:

Bitcoin,
Data Science Framework,
Artificial Intelligence,
Intermodel.

ABSTRACT

Bitcoin, a highly volatile and decentralized digital asset, presents considerable challenges for accurate price forecasting. This study proposes an applied data science framework that compares traditional statistical approaches with modern Artificial Intelligence (AI)-based models to predict Bitcoin's daily closing price. Using BTC-USD historical data from January 2020 to December 2024, we converted prices into Indonesian Rupiah (IDR) to increase local relevance. Our forecasting horizon is 30 days, based on a 60-day lookback window. We evaluate six models: Linear Regression, ARIMA, and Prophet as traditional techniques, alongside Random Forest, XGBoost, and Long Short-Term Memory (LSTM) networks as AI approaches. All models were trained using lag-based or sequence-based time series features and evaluated using MAE, RMSE, R^2, MAPE, and SMAPE. Results show that AI models, particularly LSTM and XGBoost, offer better performance in capturing short-term non-linear dynamics compared to traditional models. LSTM provides high accuracy, though with greater computational demand, while XGBoost strikes a balance between speed and precision. Prophet and ARIMA remain effective for quick and interpretable forecasts but struggle with abrupt trend shift common in cryptocurrency markets. In addition to performance metrics, we include a robustness analysis based on median absolute error and outlier detection to assess model stability under extreme variations. Visual analytics—including forecast curves, error distributions, and uncertainty bounds—help interpret and communicate model behavior. This comprehensive evaluation offers practical insights for investors, analysts, and fintech practitioners, and the pipeline can be extended to other volatile assets

I. INTRODUCTION

The quick rise of digital banking has elevated bitcoin to the status of one of the most influential and volatile assets in today's global economy. precisely an autonomous digital currency with no oversight, Bitcoin is significantly affected by market sentiment, international occurrences, regulations, and technological disruptions. Due to these features, predicting the price regarding bitcoin is an essential but difficult issue for researchers, financiers, and regulators [1] [2] [3].

Throughout the preceding era, several studies have made efforts to model Bitcoin fluctuations in prices utilizing both traditional time series approaches and artificial intelligence (AI) techniques. Traditional models, especially linear regression, autoregressive integrated moving average (ARIMA), and prophet, are straightforward and effortless to comprehend [4] [5][6]. Linear Regression, for instance, reveals an explicit association between lagged attributes and future prices [7][8], whereas ARIMA is a well-known statistical method for modelling autocorrelated and steady data [9]. Prophet, developed through Facebook, optimizes time series forecasting by handling variability and missing data, making it excellent for business and finance applications [6]. Nevertheless, these classical models regularly manage to accurately represent the non-linear and chaotic behaviour of

Bitcoin's price, particularly following high-volatility events. As a result, the research has moved toward AI-based techniques like Random Forest, XGBoost, and Long Short-Term Memory (LSTM) networks. Random Forest [10] [11] and XGBoost [12] [13] are tree-based ensemble algorithms that can reliably forecast and robustly represent intricate feature interpersonal interactions [1], [14][15]. LSTM, a deep learning model designed for sequential data, exhibited exceptional results in capturing temporal dependencies and long-term trends in financial time series [16][17].

A few investigations have methodically evaluated both families of models within an identical, verifiable data science pipeline, and while individual studies have demonstrated that AI models often outperform traditional ones under specific circumstances, comprehensive comparisons remain limited. In this study, we advance the literature by introducing a hybrid forecasting framework, where "hybrid" denotes the integration of traditional statistical approaches and AI-based methods into a single, unified pipeline. This framework not only allows direct comparison between paradigms but also leverages their complementary strengths, offering a more holistic view of forecasting performance. Additionally, much of the current research merely targets prediction accuracy, omitting model robustness, interpretability, and inter-model analysis—all of which are essential for practical application. [2][18][19][20].

The main advantage of conducting research on Bitcoin price prediction is being able to assist in determining actions in the midst of uncertainty. Detailed projections can influence trading tactics, reduce risk, along with enhance financial planning for investors [21][22]. Additionally, prediction models can be deployed by regulatory organizations and governments for maintaining their sights on systemic potential risks in cryptocurrency markets. But the field also has significant obstacles to overcome. Due to Bitcoin is vulnerable to abrupt external shocks, consisting of regulatory announcements or geopolitical events, it is difficult for any model to consistently achieve long-term accuracy [23][24]. Moreover, regardless of whether AI models are strong, they are frequently seen as "black boxes" which require to be thoroughly evaluated for interpretability, overfitting, and fairness [25][26].

A hybrid forecasting conduct that combines traditional statistical and artificial intelligence (AI)-based models in a structured data science pathway is recommended in the current research to overcome these deficiencies. The three major advantages of this research consider are outlined as follows: (1) it presents an equitable assessment of numerous models under comparable conditions; (2) its improvements intermodel interpretability and robustness beyond traditional metrics; and (3) it assembles the whole process in a data science framework which can be modular and extensible. This makes the mechanism verifiable or adjustable to other volatile assets like Ethereum, oil prices, or stock indexes by other professionals and academics [27].

The research's projected result is an in-depth evaluation of AI and traditional models for Bitcoin forecasting, as well as information on which approaches best balance interpretability, accuracy, perseverance, and computational efficiency. The main objectives of this project are to contribute a reusable forecasting architecture to the financial data science ecosystem and a data-driven framework for decision-making support in volatile financial scenarios. To enhance their practical relevance, the projections are offered in Indonesian Rupiah (IDR). While the USD-IDR conversion is a linear transformation that has no influence on model performance, it presents the results in the local financial context, where Rupiah exchange-rate volatility provides interpretive value for Indonesian investors and policymakers.

II. METHODS

The procedure is divided into numerous stages, including feature engineering, hybrid model training and prediction, data collection, pre-processing utilizing an ETL pipeline, and comprehensive performance assessments. Figure 1 depicts the data science paradigm as an operational procedure for enhancing model robustness and interpretability in the context of financial time series forecasting [28][29].

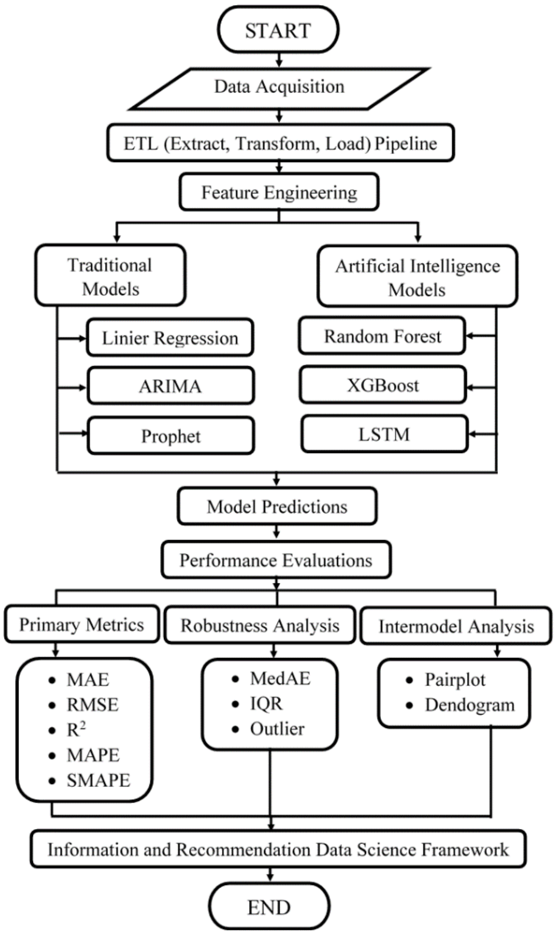

Figure 1. Data Science Framework

A. Data Acquisition

The first and fundamental stage of the hybrid prediction framework is the data acquiring procedure, as shown in Figure 1. Yahoo Finance, an acclaimed and reliable source of financial market data, contributed the dataset for this research project. In order to ensure automation, reproducibility, and immediate retrieval, daily historical data for Bitcoin (BTC-USD) was physically extracted using the Python-based yfinance package [30] [31][32].

The compilation encompasses 1,827 daily records during a five-year period, from January 1, 2020, to December 31, 2024. Nevertheless, a filtered subset of 517 recent entries—corresponding to data from January 1, 2024 to January 1, 2025—is utilized throughout model training, validation, and evaluation in order to conform with the restrictions of the modelling framework and effectively represent the practical forecasting range. This one-year timeframe captures short-term volatility and price trends that are applicable to actual market activity while facilitating a targeted, high-frequency time series forecasting endeavour [33].

The five main aspects of the dataset for every trading day are as follows:

- The open price of Bitcoin is its starting trading value at the beginning of the day, while the high price is its highest value throughout the trading session.
- Low Price: the lowest price ever noted;
- Close Price: the total amount of Bitcoin moved during the day;
- Volume: the final worth of trading at the end of the session

Out of them, the Close Price has been selected as the forecasting the objective variable. The choice of terminology was made due to its captures the complete effect of intraday price swings and trader sentiment, is straightforward to interpret, has market significance, and is commonly employed in financial forecast literature [34].

All price data originally denominated in US dollars (USD) is converted to Indonesian rupiah (IDR) to retain both global relevance and local economic context. For Indonesian investor applications and domestic financial research, this currency conversion is very critical. In order to maintain temporal alignment between Bitcoin prices and foreign exchange rates, the conversion will be carried out using the historical daily exchange rates that correspond to each exchange date. As stated in the following layer of Figure 1, the outcomes of this data acquiring stage can be utilized as the input for the subsequent step, ETL pre-processing.

B. ETL (Extract, Transform, Load) Pipeline

The estimation framework's second essential step is the ETL (Extract, Transform, Load) pipeline. As an element of data pre-processing, this phase provides ensure that the unprocessed Bitcoin price data gathered in the Data Acquisition step is cleaned, planned, and processed into a format that allows it to be utilized to perform sophisticated analytical processing and modelling.

The yfinance Python package has been utilized to import the raw dataset that was gathered from Yahoo Finance in order to starting the extraction process [35]. Each trading day's Open, High, Low, Close, and Volume are the five primary financial indicators comprised of this dataset. Full temporal alignment retains the close price, which is the target variable, while additional parameters are available to receive additional engineering. Additionally, to maintain contextual relevance for the Indonesian financial circumstances, daily exchange rates are extracted to convert prices denominated in USD into IDR. This conversion represents a linear rescaling that has little impact on the statistical properties of the data or the relative performance of the models, but it enhanced interpretability for domestic investors by expressing the local financial environment more realistically by means of Rupiah's own exchange-rate fluctuations.

The ETL pipeline's transformation stage is its most demanding part. Among them are:

- Handling Missing Values: interpolating or reducing rows with null or unusual entries to ensure completeness.
- Outlier Detection and Smoothing: Identifying aberrant spikes or dips that can interfere with model training and flattening them applying historical context or statistical thresholds.
- Currency Conversion: The time series is kept economically proper by converting close prices from USD to IDR using the associated daily exchange rates.
- Resampling and Alignment: validating that all observations have periodic intervals and match legitimate trading days.
- Lookback Window Construction: The dataset is organized using a 60-day sliding window trend, with each input sequence (X) encompassing 60 daily values beyond the past that are used in predicting the target period (Y), which is 30 days from presently.

This stage may also involve the implementation of normalization or scaling (e.g., Min-Max scaling or Z-score standardization), especially with models that are sensitive to scale, involving neural networks and tree ensembles.

The cleaned and organized data must then be loaded into certain information structures for modelling as the last stage. The generated datasets are structured as supervised learning sequences, as shown in the following branch of Figure 1, with each sample composed of engineered input features and corresponding future target values. These datasets are generated for input into traditional or artificial intelligence-based predictive models.

This ETL procedure provides an essential connection between the collection of raw data and the emergence of significant characteristics by ensuring the coherence, integrity, and sustainability of the incoming data. The study assures the modelling process's consistency, resilience, and adaptability for various financial time series by simplifying and standardizing this pipeline.

C. Feature Engineering

The forecasting pipeline's feature engineering stage plays an important role considering it transforms preprocessed time series data into structured inputs the fact can be used by models. According to the type of predictive model, the feature engineering process diverges into two concurrent tracks, as represented in Figure 1: artificial intelligence (AI)-based models and traditional statistical approaches. Feature synthesis for each class of model requires distinct approaches based on its capabilities, assumptions, and data handling procedures [36].

Feature engineering for traditional models like Linear Regression, ARIMA, and Prophet emphasizes interpretability and statistical assumptions. In this context, developed features include time-derived components (e.g., weekday, month, or quarter) to capture seasonality and calendar implications, which are particularly advantageous for Prophet. Lag variables have been generated from previous closing prices (e.g., lag-1 to lag-60) to capture autoregressive patterns throughout time [4][5][6][37]. These lagged properties enable linear models like ARIMA to detect direct temporal associations in the series. Moving averages and standard deviations are among the rolling statistics intended to illustrate short-term trends and volatility. ARIMA, in particularly, leverages differencing to figure out stationarity, which is essential for meaningful parameter estimation in time series models. The resulting feature set is inadequate, interpretable, and in agreement with traditional forecasting theory [9][17].

AI-based models, such as Random Forest, XGBoost, and Long Short-Term Memory (LSTM) networks, necessitate a more flexible and multimodal feature engineering method. These models are designed to automatically learn multifaceted nonlinear relationships from data, allowing for more detailed resource representations. A 60-day rolling window is used for generating input sequences in both tree-based models and LSTMs, which are then utilized for forecasting the following 30-day horizon [12][38][39]. Tree-based models flatten this window into fixed-length feature vectors, whilst LSTM networks preserve the sequence structure as a three-dimensional array for retaining temporal order. In the absence of lag sequences, the AI-based feature set includes derived indicators such as daily returns, percentage changes, volatility measurements, and volume-based signals such as trade volume moving averages and volume-price ratios. Although machine learning models are capable of processing multivariate inputs and high dimensionality, additional technical indicators have been integrated as part of the learning process. Furthermore, feature implications analysis in tree-based models processes the feature space, whereas LSTMs automatically capture latent temporal associations using internal storage the procedures [40].

These two complemented feature engineering approaches, which are represented in Figure 1 as parallel pathways, collaborate in order to enhance each model type's ability to forecast for the Bitcoin price forecasting working by making certain that each model type obtains the most informative and functionally relevant information.

D. Model Predictions

Following feature engineering, the produced datasets will be used to train an extensive variety of forecasting models, which are divided into two categories: traditional models and artificial intelligence (AI) models. This hybrid modeling procedure has been designed to take advantage of the unique features of both model groups. Traditional models provide interpretability and statistical rigor, whereas AI-based models excel in identifying complicated, nonlinear patterns in data.

We ensure fairness across all approaches; the models were validated using a chronological hold-out split, with the earliest 80% of observations used for training and the most recent 20% reserved for testing. This prevents temporal leakage and better reflects a real-world forecasting scenario. Hyperparameter tuning was not the primary focus of this research, as the study emphasizes benchmarking rather than optimization. Accordingly, default configurations were employed for Random Forest, XGBoost, and Prophet, while ARIMA was fixed at an order of (5,1,0). For the LSTM model, limited manual iterations were conducted to balance accuracy and efficiency, resulting in a two-layer architecture with 50 units each, a dropout rate of 0.2, a batch size of 32, and 100 epochs with EarlyStopping (patience=10). Before applying ARIMA, the stationarity of the series was verified using the Augmented Dickey–Fuller (ADF) test, with differencing applied when necessary; Prophet and ARIMA were not fully optimized, as the focus was on methodological benchmarking rather than parameter fine-tuning [1][2].

The models are all trained and examined on a consistent data split to provide comparable performance comparisons. Repeatability and scale investigations are made possible by the single pipeline the fact that automates the training technique. Throughout training, each model predicts the predicted horizon using out-of-sample data. Beyond this, the predictions undergo the evaluation phase, when their accuracy, consistency, and comparative behavior are assessed [41][42].

E. Performance Evaluations

Performance evaluation is the next essential aspect in the framework, particularly is displayed in Figure 1, after expectations have been generated employing both traditional and AI-based models. This phase measures each model's suggested accuracy, dependability, and comparative behavior along a variety of analytical parameters. This evaluation is organized into three primary elements in order to attain this: (a) Primary Metrics, (b) Robustness Analysis, and (c) Intermodel Analysis. Collectively, these three elements constitute an in-depth evaluation of each model's performance in terms of both practical effectiveness and precision in forecasting.

Primary metrics are used to evaluate predictive accuracy, providing direct comparisons between the predicted values yˆ_t and the actual observed values y_t over the forecasting horizon t=1,2,…,n. The following standard error-based formulations are employed.

- Mean Absolute Error (MAE) calculates the average size of a series of forecasts' mistakes without implementing consideration of their direction. The mean of the absolute discrepancies between the expected and actual outcomes will be utilized to compute it. The MAE is relatively easy to understand and provides a linear score, which indicates that each individual error is given an equal weight [3].

$$MAE = \frac{1}{n}\sum_{t=1}^{n} |y_t - \hat{y}_t| \qquad (1)$$

- Root Mean Squared Error (RMSE), in contrast, discourages greater deviations more severely by squaring each mistake before averaging. Because it is sensitive to outliers, RMSE is especially helpful in predicting applications where big errors are undesirable [4].

$$RMSE = \sqrt{\frac{1}{n}\sum_{t=1}^{n} |y_t - \hat{y}_t|} \qquad (2)$$

- Coefficient of Determination (R^2) determines the percentage of the observed data's variation that the model can account for. Better model fit is indicated by an R^2 value around 1, whereas low explanatory power can be determined by a value close to 0. It is an advantageous supplement to absolute error metrics such as RMSE and MAE [5].

$$R^2 = 1 - \frac{\sum_{t=1}^{n}(y_t - \hat{y}_t)^2}{\sum_{t=1}^{n}(y_t - \bar{y}_t)^2} \qquad (3)$$

- Mean Absolute Percentage Error (MAPE) depicts the average absolute inaccuracy in comparing with actual values and represents predicting accuracy as a percentage. Although MAPE is scale-independent, it may be utilized for comparing models across other currencies or datasets when y_t is close to zero [6].

$$MAPE = \frac{100\%}{n}\sum_{t=1}^{n}\left|\frac{y_t - \hat{y}_t}{y_t}\right| \qquad (4)$$

- Symmetric Mean Absolute Percentage Error (SMAPE) adjusts MAPE to account for asymmetry in over- and under-forecasting. It is especially valuable in financial forecasting, where percentage deviations can be misleading if not normalized symmetrically when actual or predicted values are near zero [7].

$$SMAPE = \frac{100\%}{n}\sum_{t=1}^{n}\frac{y_t - \hat{y}_t}{(|y_t| + |\hat{y}_t|)/2} \qquad (5)$$

All of these metrics, which each describe an individual component of model error behavior, operate well together to provide an accurate foundation for evaluating the average prediction performance among models.

The following evaluation component tackles this by focusing on robustness analysis, which analyses how robust and accurate model predictions are regardless of noise, volatility, or outliers, all of which are common in financial time series such as Bitcoin. This stage of analysis revolves on non-parametric, distribution-sensitive measures:

- Median Absolute Error (MedAE) is similar to MAE, except it estimates the median rather than the mean of absolute errors. This makes it less susceptible to extreme values and more appropriate for capturing the core pattern of prediction errors in volatile settings [8][9].

$$MedAE = median(|y_t - \hat{y}_t|) \qquad (6)$$

- Interquartile Range (IQR) measures the spread of errors by calculating the range between the 25th and 75th percentiles. A lower IQR indicates that the model produces more consistent predictions, even if its mean error is not the lowest [10][11].

$$IQR = Q_3 - Q_1 \qquad (7)$$

Where Q_1 and Q_3 are first and third quartiles of the absolute errors. A smaller IQR indicates that most errors fall within a narrow, predictable range as an indicator of model stability

- Outlier Threshold or Sensitivity refers to a model's tendency to produce large errors when encountering anomalous or extreme market behaviors. This is assessed by identifying the frequency and severity of prediction errors that lie beyond statistical thresholds (e.g., above 1.5× IQR). Models that maintain stable performance despite such events are considered more robust [12][13].

$$Outlier\ Threshold = Q_3 + 1.5 \times IQR \qquad (8)$$

Robustness analysis is essential in financial applications where occasional large errors can have significant real-world implications, especially in risk-sensitive decision-making.

The last assessments component is intermodel analysis, allowing analysts to better understand how models connect to one another in terms of predictive behavior. In contrast to focusing solely upon absolute performance, this phase focuses for similarities, differences, and clustering patterns between models, revealing meta-level insights into how forecasting systems compare as ensemble.

- Pairplots are Intended to display pairwise correlations between model predictions. Pairplots assist to figure out whether particular predictions consistently agree or diverge through contrasting predicted values against one other. Robust Pearson correlations across models can imply redundancy, but orthogonal behavior can point to complementing modeling approaches ideal for assembling [14][15].

- Dendrograms are produced by hierarchical clustering and group models based on the Euclidean distance (or other similarity measures) between their prediction vectors. The dendrogram structure assists in identifying natural groupings among models, which is vital to figuring out model families, selecting fluctuated ensembles, and understanding which modelling methodologies provide unique forecasting defining features [16][17].

These intermodel tools not only enable performance benchmarking, but they additionally provide pertinent tactical understanding into the interactions between different forecasting paradigms in science.

The research effort confirms a thorough and equitable evaluation of forecasting performance through the integration of these three levels of assessment: intermodel analysis, robustness checks, and primary accuracy metrics. This multimodal assessment directly contributes to the data science framework's last phase, when model outcomes are converted into practical recommendations for decision-makers.

F. Information and Recommendation Pathways

The synthesis of findings and creation of technical recommendations constitute the last phase of the suggested technique, which is depicted at the bottom of Figure 1. For practitioners, analysts, and decision-makers intending at applying forecasting approaches in practical financial contexts, especially in the volatile and high-impact field of cryptocurrencies, this phase serves as essential for integrating complex model outputs into helpful guidance.

This Information and Recommendation Framework provides structured, technical pathways for growing the forecasting system in both academic research and real-world financial applications, in spite of analyzing model performance. It completes the end-to-end hybrid forecasting methodology recommended in this study and functions as the pipeline's last output.

III. Results and Discussions

A. Overview Observations

The current investigation applies a consistent setting for experiments for all tests in order to determine the forecasting capacity utilization provided by various models. Models are trained on a 60-day historical window and handed the task of forecasting the upcoming 30 days. The prediction task has been structured using a rolling-window framework. Through modeling the sequential nature of financial data, this windowing approach enables a realistic simulation of time series forecasting.

The dataset covers 517 daily Bitcoin price evaluations from January 1, 2020, to December 31, 2024. To determine the implications of data distribution on forecasting accuracy, the models were trained using two differing feature representations: the raw feature set (in absolute price scale) and a log-transformed version.

Three traditional statistical techniques—Linear Regression, ARIMA, and Prophet—as well as four artificial intelligence (AI)-based techniques—Random Forest, XGBoost, and LSTM—were deployed to develop and analyse seven forecasting models. A total of 14 experimental configurations are generated by applying each model separately to both feature sets. As mentioned in the following sections, a thorough set of assessment evaluates and visual analyses have been employed to evaluate each model's performance.

B. Predictive Visualization

Two important graphs are shown in this part to give a visual evaluation of the model's performance. Both in terms of trend alignment and predictions for the future, these visualizations serve as proof of the way precisely each forecasting model represents the actual structure of the Bitcoin price.

Figure 2 shows a plot matching the actual Bitcoin prices over the evaluation period with the projected figures obtained from each model. By superimposing forecasts from every model on a single timeline, this line diagram provides it possible to compare the extent to which each model tracks the actual market moves. Effectively tracking both upward and downward movements with little latency, models like LSTM and XGBoost indicate a high degree of alignment with the real price curve. In contrast, simpler linear techniques, such as Linear Regression, tend to smooth out variations and are unable to adjust to abrupt market movements.

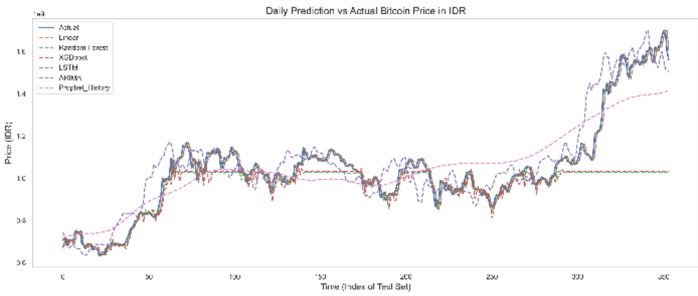

Figure 2. Prediction alignment across models

The potential of each model for forecasting 30 future data points based on the most recent 60-day input window can be observed more thoroughly in figure 3. The pattern and spatial distribution of forecasts across numerous models has been highlighted in this figure. The best-performing models, LSTM and XGBoost in particular, show subtle sensitivity to the current trend and generate predictions that correspond to the underlying momentum of the price data. Prophet tends to generate conservative projections and regularly ignores unexpected directional alterations, despite being typically reliable at discerning trend and seasonality. Beyond consistent conditions, Random Forest and ARIMA indicate limited extrapolation power, which might contribute to predictions becoming either flat or lag.

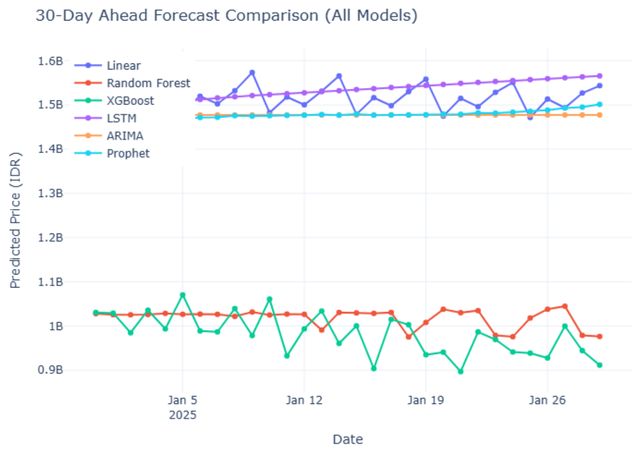

Figure 3. Forecasting trajectory all models

All things considered, LSTM and XGBoost appears to be more efficient at detecting both the broader trend and the short-term volatility. Their natural ability for representing intricate, nonlinear temporal relationships may have been a consequence of this. Whereas ARIMA and Linear Regression have difficulty able to adapt to irregular price behavior, Prophet does an adequate task of capturing smoother trends [4][33][38].

C. Primary Metrics Evaluation

Model performance had been assessed using five error metrics: Mean Absolute Error (MAE), Root Mean Squared Error (RMSE), Coefficient of Determination (R^2), Mean Absolute Percentage Error (MAPE), and Symmetric Mean Absolute Percentage Error. The above indicators combine for evaluating prediction accuracy, error magnitude, and consistency across all models.

TABLE 1.
PERFORMANCE METRICS SUMMARY (ABSOLUTE FEATURE SET)

Model	MAE (↓)	RMSE (↓)	R^2 (↑)	MAPE (%) (↓)	SMAPE (%) (↓)
Linear	21.37 M	29.93 M	0.984	2.022	2.031
Random Forest	101.92 M	196.50 M	0.294	7.598	8.586
XGBoost	105.86 M	198.03 M	0.283	7.988	8.998
LSTM	24.58 M	33.64 M	0.979	2.278	2.288
ARIMA	21.31 M	29.72 M	0.984	2.020	2.026
Prophet	90.68 M	106.75 M	0.791	8.559	8.447

The models in Table 1 with the lowest MAE and RMSE values in the above table are the Linear Regression and ARIMA models, which both have remarkably comparable accuracy scores. A strong match to the data is indicated by the similarities in their best R2 score of 0.984. Interestingly, LSTM gets a high R2 of 0.979 and low percentage-based errors (MAPE and SMAPE about 2.28%), indicating that it reflects the pattern well, but with some divergence in magnitude, although experiencing slightly larger absolute erroneous. In the present research, Random Forest and XGBoost, on the other hand, behaved poorly. Both have significant absolute and percentage errors, as well as low R^2 values, implying that they are unable to adequately express the temporal structure of the data, presumably due to its nonsequential nature. Prophet outperforms the tree-based models, but it is still behind the top three.

Figure 3 demonstrates the absolute prediction errors for all models. Models such as Linear, ARIMA, and LSTM have smaller and more centered error distributions, which facilitates the numerical evaluation. XGBoost and Random Forest show greater spreads and higher average mistakes.

Figure 4 depicts the prediction error distributions for each model in normal scale, enabling an obvious comparison of how each model's forecasts differ from actual values. Linear Regression and ARIMA again stand out, with error distributions that are narrower and more symmetric, centered at zero. LSTM follows with a little larger but still constrained distribution.

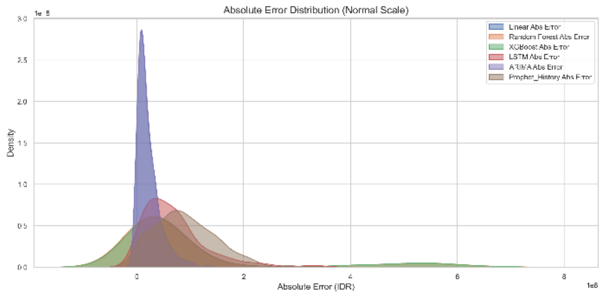

Figure 4. Absolute Error Distribution

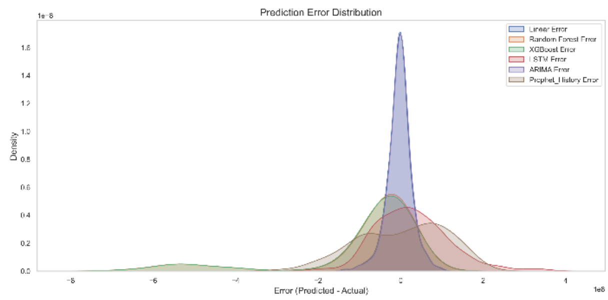

Figure 5. Prediction Error Distribution

Tree-based models, such as Random Forest and XGBoost, reveal substantial bias and variance, causing lower R^2 values. The calculations and graphics clearly suggest that Linear Regression, ARIMA, and LSTM are the most dependable models for forecasting Bitcoin values in this study, with LSTM providing a competitive nonlinear alternative to AI algorithms [6][56].

D. Robustness Analysis

Traditional evaluation metrics provide valuable insights into overall model correctness, whereas robustness indicators provide a more in-depth understanding of the stability and consistency of each model's predictions under numerous circumstances. This section evaluates the Median Absolute Error (MedAE), Interquartile Range (IQR), and number of prediction outliers, featuring three relation error distribution visualizations to support the analysis.

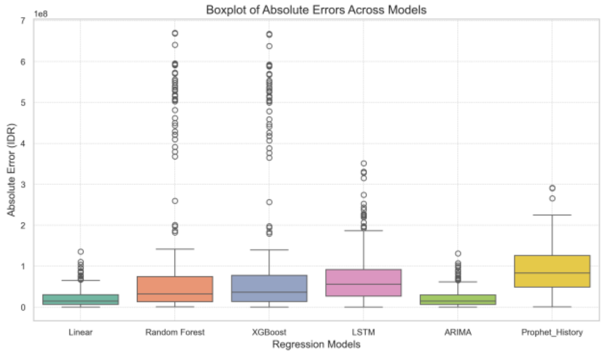

Figure 6. Absolute Error (normal scale)

The distribution of absolute errors for each model on the original (linear) scale can be observed in this boxplot in figure 6. With modest median values and comparatively limited error bands, the Linear and ARIMA models tend to function steadily with few dramatic departures. On the other hand, the error ranges of Random Forest, XGBoost, and Prophet are greater, indicating more variability and either overfitting or undergeneralization.

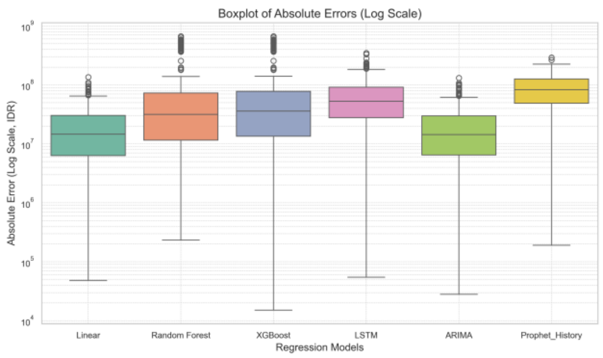

Figure 7. Absolute Error (log scale)

Figure 7 demonstrates changes in performance across small and large error ranges by displaying errors on a logarithmic scale. The log transformation compounds disparities between models with significant outliers. The compactness of ARIMA and Linear models becomes more apparent here, whereas the distributions of tree-based models such as XGBoost and Random Forest show a strong rightward skew, indicating a higher proportion of high-error circumstances.

This enhanced version incorporates median lines (Figure 8), allowing for a direct visual comparison of central error patterns between models. Notably, ARIMA has the lowest MedAE (14.37M), followed by Linear Regression (14.55). These findings are in accordance with their consistent predictive performance. In contrast LSTM has a fairly high MedAE (56.65M) despite great average metric performance, indicating some significant mispredictions that interfere with error consistency. Prophet, however having a high MedAE (83.12M), has a very low number of outliers (3), signaling that its oversights are regularly enormously but less irregular.

The most reliable models, as shown by the results of our research of Table 2, are ARIMA and Linear Regression, that maintain a low median error with a narrow IQR and few outliers. On the other hand, Random Forest and XGBoost are less predictable under volatile market situations due to their high unpredictability and frequent outliers. Although LSTM exhibits great promise, it might need to be further improved to lower exceptionally large prediction errors [48][56].

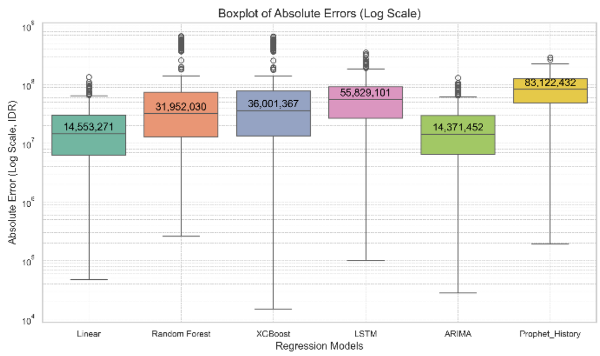

Figure 8. Absolute Error with Median

TABLE 2.
SUMMARIZES THE ROBUSTNESS ANALYSIS

Model	MedAE	IQR	Outliers Threshold
Linear	14.37 M	23.37 M	19
Random Forest	14.55 M	24.11 M	17
XGBoost	33.45 M	63.98 M	44
LSTM	36.00 M	64.34 M	44
ARIMA	56.65 M	61.96 M	19
Prophet	83.12 M	77.40 M	3

E. Intermodel Correlation and Agreement

An intermodel research has been conducted in order to get a greater comprehension of the forecasting models' interactions beyond each measurement. This provides structural grouping, error behaviors, and prediction alignment comparisons. These results have significance when determining whenever particular frameworks could be absorbed satisfactorily and whether they behave similarly, neither of which are needed to generate dependable ensemble or hybrid systems.

The Pearson correlation coefficients between each model's anticipated values and the actual closing prices of Bitcoin are presented in Figure 9. remarkably, the models that perform greatest across critical evaluation metrics are the ARIMA and Linear Regression models, which additionally exhibit the strongest correlations with the actual information. Furthermore, LSTM demonstrates a substantial correlation, revealing that it might detect widespread trend patterns in spite of its complex architecture. Random Forest and XGBoost, whereas exhibit a slightly lower correlation with the real series, which could indicate overfitting to local noise or a less effective understanding of the underlying market direction.

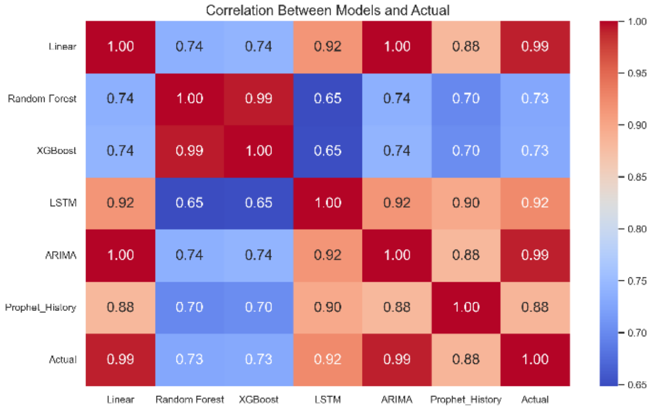

Figure 9. Correlation of Predicted and Actual Values

The correlation of absolute error magnitudes among models can be observed in Figure 10, which provides additional insight into the behavior of model mistakes. Models are likely to make mistakes on the same observations if there are several high correlations in this matrix. As expected, given their similar tree-based architecture, XGBoost and Random Forest, for example, reveal a substantial degree of error correlation. However, because both ARIMA and linear models depend upon trend-driven statistical concepts, they also have important error similarities. Interestingly, Prophet and LSTM seem to have less overlap in their error profiles and are more independent, which shows that they could potentially be effective when recognizing various aspects of the data when applied in a mixed ensemble arrangement.

Figure 11 illustrates a visual comparison of model predictions against the actual values and each other utilising a matrix of scatter plots and distributions to support the results presented. Each model's prediction distributions are represented by the diagonal elements, and the associations among each model's predictions are indicated by the off-diagonal elements. Strong visual alignment with the real series has been shown by ARIMA, Linear, and LSTM, which generate narrow, focused patterns. The more scattered patterns generated by Random Forest and XGBoost, on the contrary hand, emphasize their greater variability and possible instability in specific circumstances.

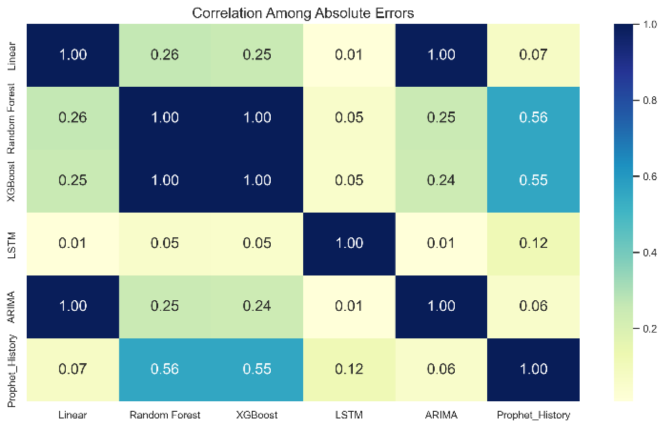

Figure 10. Correlation of Model Error Magnitudes

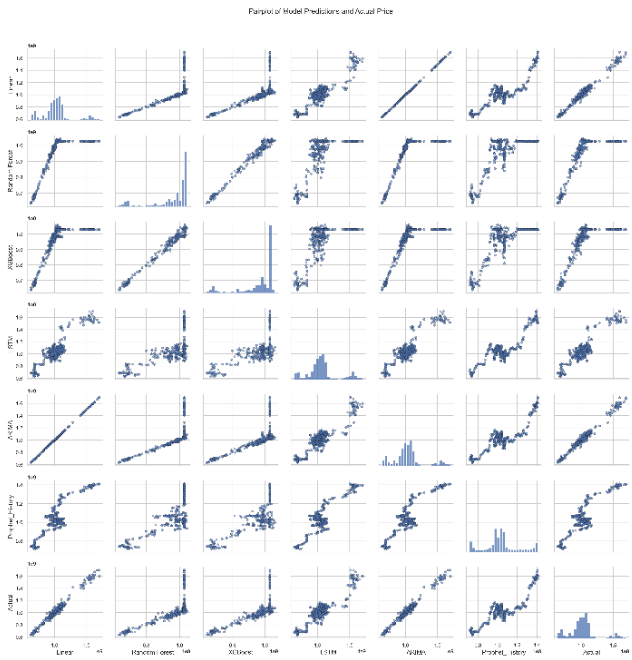
Figure 11. Pairplot of Model Prediction and Actual Prices

A scatter matrix of absolute errors across models can be seen in Figure 12 to better examine this variability. It is readily apparent that Linear Regression and ARIMA consistently produce more focused and lower error values, indicating more reliable performance. XGBoost and Random

Forest, on the reverse side, exhibit important co-variability and broader distributions, emphasizing their tendency to make similar types of mistakes. Prophet's appropriately compact and uniform dispersion, notwithstanding its huge error numbers, reflects underfitting rather than failure to perform.

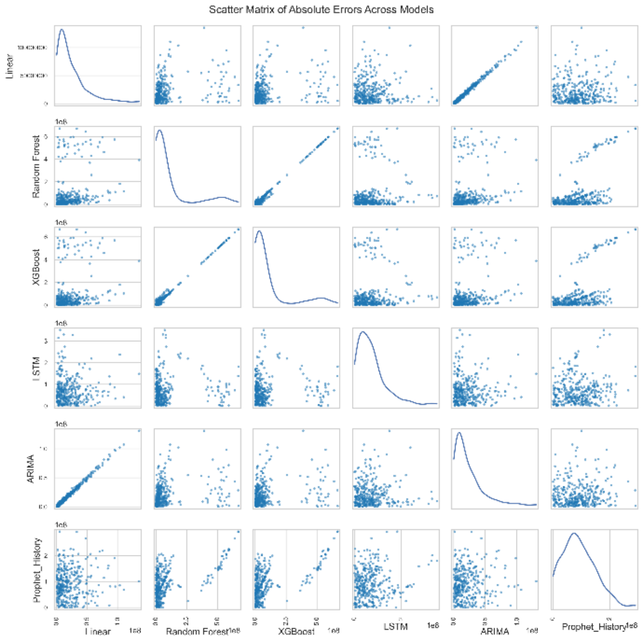

Figure 12. Scatter Matrix of absolute Error Models

The models ultimately are grouped hierarchically according to their behavior in Figure 13. The way models cluster based on error structures or prediction similarity is illustrated by this dendrogram. The restricted grouping of Linear Regression and ARIMA confirms their similar performance patterns and similar methodological foundations. Furthermore, Random Forest and XGBoost indicate a parallel in their ensemble-learning approach and high-variance results, constituting a separate cluster. With the integration of temporal dynamics and trend-capturing influence, LSTM is located in the middle of those classifications. But Prophet is set apart from the others, which emphasizes its distinctive behavior and slightly distinct forecast profile.

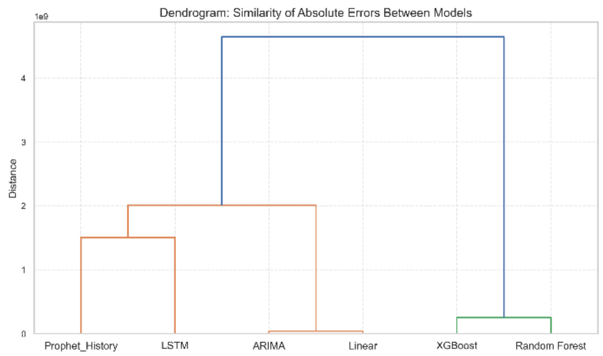

Figure 13. Similarity Dendogram among Absolute Error between Models

In the final analysis, this intermodel research not only validates the higher dependability of ARIMA, Linear, and LSTM models, but also demonstrates their complementing tendencies, making them excellent candidates for ensemble integration. Models such as XGBoost and Random Forest, by contrast, are different but frequently overlap and require careful calculation because to their volatility and shared constraints. These structural insights are useful for enhancing forecasting systems using model selection or hybridization methods [57].

F. Interpretation and Insights

The comprehensive evaluation across numerous performance characteristics indicates ARIMA and Linear Regression are the most consistently dependable models for short-term Bitcoin price forecasting in this study. Both models perform adequately across all key factors, with low MAE and RMSE, high R^2 values, and insignificant percentage-based errors (MAPE, SMAPE). Their performance is further supported by excellent error distribution patterns (as depicted in the boxplots and absolute error visualizations), characterized by low median absolute errors, narrow interquartile ranges (IQR), and few outliers. These properties reveal that both ARIMA and Linear Regression generate dependable forecasts with little volatility, making them ideal for applications required interpretability and stability, such as financial monitoring or risk-sensitive trading approaches.

LSTM provides significant strength in capturing non-linear patterns and short-term temporal dynamics, although slightly lagging beyond ARIMA and Linear in terms of raw accuracy. In both current and upcoming circumstances, its prediction curves nearly match actual prices. Nevertheless, there are some trade-offs associated with its performance. LSTM is more sensitive to the volume and quality of data, requiring more intricate hyperparameter adjustment, and takes longer to train. Due to this, it is more suitable for circumstances in which there are ample computational resources and prevalent datasets available, particularly for modeling more volatile or non-linear market segments. Random Forest and XGBoost, on the other hand, indicate low reliability despite their potential advantage in dealing with high-dimensional, non-linear data. These models underperform in practically each measurement and produce greater, more unpredictable errors. They additionally exhibit a significant inter-model error correlation, showing that deficiencies are comparable across data points. This redundancy, along with excessive volatility, lowers their standalone worth. However, they may still be useful in ensemble settings where each particular modeling architecture states unique perspectives. Whereas Prophet was constructed for trend-seasonality decomposition, it behaved inadequately in the majority of evaluations in this study. It provides relatively high MAE, RMSE, and error percentages, but its correlation with actual data and other models is limited. Nonetheless, Prophet demonstrates consistent behavior and minimal variance in some robustness indicators (e.g., low outlier count), rendering it potentially acceptable for fundamental trend forecasting tasks in less time-sensitive or high-resolution domains.

In the final analysis, ARIMA and Linear Regression provide the optimum integrate of accuracy, durability, and simplicity, whereas LSTM adds value when modelling complexity or temporal depth must be considered. Meanwhile, tree-based models and Prophet could require more cautious deployment, preferably in hybrid or ensemble systems where their distinctive qualities can be carefully employed. These findings offer straightforward direction for practical implementation and prospective model development procedures.

G. Implications and Recommendations

The results imply that the determination of model should be in line with the objectives of forecasting and the characteristics of the data. Although it needs a lot of data and processing power, LSTM is best suited for capturing intricate, short-term dynamics. XGBoost is perfect for real-world deployment because it strikes a compromise between precision and efficiency. For rapidly, comprehensible results, traditional models like ARIMA and Linear Regression are still helpful, particularly in less volatile circumstances.

Practitioners should consider log transformations to improve stability—especially for Prophet and XGBoost—and leverage robustness analysis to identify outlier-sensitive models. Given the diverse strengths across models, ensemble strategies could further enhance prediction reliability, particularly in volatile markets like cryptocurrency.

IV. CONCLUSIONS

The current research presents a comparative forecasting framework that predicts Bitcoin's daily closing price in IDR using classic statistical models (ARIMA, Linear Regression, Prophet) and AI-based approaches (Random Forest, XGBoost, LSTM). Our outcomes illustrate that AI models, particularly LSTM and XGBoost, excel at catching short-term, non-linear patterns, with LSTM providing the highest accuracy despite higher computing costs. Traditional models such as ARIMA and Linear Regression, while less adaptable to volatility, are nonetheless competitive due to their simplicity and resilience. Robustness analyses and visualization tools (for example, forecast curves, error distributions, and intermodel correlations) highlight necessary trade-offs and identify potential for ensemble approaches.

Finally, the research study not only depicts the intricate trade-offs between various forecasting methods, but also provides actionable advice for practitioners and academics. The suggested methodology can be applied to other highly volatile financial products, enabling better informed and robust decision-making in the rapidly transforming digital economy. Moreover, the comparative performance metrics reported in this study provide convincing numerical evidence of the framework's reliability, reinforcing its value as a credible reference for both scholarly research and practical applications. Nonetheless, this study is limited to historical price data and does not account for external drivers such as macroeconomic variables or sentiment factors, which may significantly influence Bitcoin prices. Future extensions of the framework could incorporate these fundamental aspects to enhance predictive realism and broaden its applicability.

REFEFRENCES

[1] T. Phaladisailoed and T. Numnonda, "Machine learning models comparison for bitcoin price prediction," *Proc. 2018 10th Int. Conf. Inf. Technol. Electr. Eng. Smart Technol. Better Soc. ICITEE 2018*, no. January, pp. 506–511, 2018, doi: 10.1109/ICITEED.2018.8534911.

[2] S. Saraswathi, J. S. Sridhala, A. Elavazhagan, J. S. Sabharwal, and S. I. Mohammad, "An Ensemble Machine Learning Technique for Bitcoin Price Prediction," *J. Trends Comput. Sci. Smart Technol.*, vol. 6, no. 2, pp. 153–167, 2024, doi: 10.36548/jtcsst.2024.2.005.

[3] J. T. Kjellstadli, E. Bering, M. Hendrick, S. Pradhan, and A. Hansen, "Can local stress enhancement induce stability in fracture processes? Part I: Apparent stability," *Front. Phys.*, vol. 7, no. JULY, pp. 1–8, 2019, doi: 10.3389/fphy.2019.00098.

[4] R. Manivannan, S. Subashree, G. Venkateshwaran, G. Pushpa, E. Saranya, and M. Rajakumaran, "An Intelligent and Accurate Bitcoin Price Forecasting using ARIMA Model," *SSRN Electron. J.*, 2025, doi: 10.2139/ssrn.5091288.

[5] I. Suryani and H. Harafani, "Predicting the Bitcoin Price Using Linear Regression Optimized With Exponential Smoothing," *J. Ris. Inform.*, vol. 3, no. 3, pp. 277–282, 2021, doi: 10.34288/jri.v3i3.238.

[6] Suresh N, Priya B, Sai Teja M, Sai Krishna B, and Lakshmi G, "Historical analysis and forecasting of stock market using fbprophet," *South Asian J. Eng. Technol.*, vol. 12, no. 3, pp. 152–157, 2022, doi: 10.26524/sajet.2022.12.43.

[7] M. F. Arfa, M. R. AlFathan, H. B. Lumbantobing, and R. Rahmadenni, "Prediksi Harga Cryptocurrency Dengan Metode Linier Regresi," *SENTIMAS Semin. Nas. Penelit. dan Pengabdi. Masy.*, vol. 1, no. 1, pp. 8–15, 2023, [Online]. Available: https://journal.irpi.or.id/index.php/sentimas/article/view/609%0A https://journal.irpi.or.id/index.php/sentimas/article/download/609/332

[8] C. Pebriyanti and G. M. Suranegara, "Evaluation of Scalability and Resilience of Hyperledger Fabric in Blockchain Implementation for Diploma Management," vol. 9, no. 3, 2025.

[9] Y. Hua, "Bitcoin price prediction using ARIMA and LSTM," *E3S Web Conf.*, vol. 218, pp. 1–5, 2020, doi: 10.1051/e3sconf/202021801050.

[10] F. Orte, J. Mira, M. J. Sánchez, and P. Solana, "A random forest-based model for crypto asset forecasts in futures markets with out-of-sample prediction," *Res. Int. Bus. Financ.*, vol. 64, no. August 2022, 2023, doi: 10.1016/j.ribaf.2022.101829.

[11] S. A. Basher and P. Sadorsky, "Forecasting Bitcoin price direction with random forests: How important are interest rates, inflation, and market volatility?," *Mach. Learn. with Appl.*, vol. 9, no. June, p. 100355, 2022, doi: 10.1016/j.mlwa.2022.100355.

[12] Y. He, "Prediction of Bitcoin Daily Returns Based on OLS , XGBoost , and CNN Machine Learning Models," no. Ecai 2024, pp. 181–187, 2025, doi: 10.5220/0013212400004568.

[13] A. Hafid, M. Ebrahim, M. Rahouti, and D. Oliveira, "Cryptocurrency Price Forecasting Using XGBoost Regressor and Technical Indicators," *Conf. Proc. IEEE Int. Performance, Comput. Commun. Conf.*, pp. 1–9, 2024, doi: 10.1109/IPCCC59868.2024.10850357.

[14] S. Zhang, M. Li, and C. Yan, "The Empirical Analysis of Bitcoin Price Prediction Based on Deep Learning Integration Method," *Comput. Intell. Neurosci.*, vol. 2022, 2022, doi: 10.1155/2022/1265837.

[15] A. R. Kamila, J. F. Andry, A. Wahyu, C. Kusuma, W. Prasetyo, and G. H. Derhass, "Analysis Comparison of K-Nearest Neighbor , Multi-Layer Perceptron , and Decision Tree Algorithms in Diamond Price Prediction," *COGITO Smart J.*, vol. 10, no. 2, pp. 298–311, 2024.

[16] Ferdiansyah, S. H. Othman, R. Zahilah Raja Md Radzi, D. Stiawan, Y. Sazaki, and U. Ependi, "A LSTM-Method for Bitcoin Price Prediction: A Case Study Yahoo Finance Stock Market," *ICECOS 2019 - 3rd Int. Conf. Electr. Eng. Comput. Sci. Proceeding*, no. June, pp. 206–210, 2019, doi: 10.1109/ICECOS47637.2019.8984499.

[17] F. Ding, "The Research on the Prediction of Cryptocurrency Based on Linear Regression and LSTM," pp. 1472–1481, 2023, doi: 10.2991/978-94-6463-102-9_154.

[18] Z. Ye, Y. Wu, H. Chen, Y. Pan, and Q. Jiang, "A Stacking Ensemble Deep Learning Model for Bitcoin Price Prediction Using Twitter Comments on Bitcoin," *Mathematics*, vol. 10, no. 8, 2022, doi: 10.3390/math10081307.

[19] P. R. Panggabean and H. A. Aidilof, "Sentiment Analysis of Public Comments on X Social Media Related to Israeli Product Boycotts Using The Long Short-Term Memory (LSTM) Method," vol. 9, no. 3, pp. 775–783, 2025.

[20] B. Hakim, "Analisa Sentimen Data Text Preprocessing Pada Data Mining Dengan Menggunakan Machine Learning," *JBASE - J. Bus. Audit Inf. Syst.*, vol. 4, no. 2, pp. 16–22, 2021, doi: 10.30813/jbase.v4i2.3000.

[21] Y. Yang, J. Xiong, L. Zhao, X. Wang, L. Hua, and L. Wu, "A Novel Method of Blockchain Cryptocurrency Price Prediction Using Fractional Grey Model," *Fractal Fract.*, vol. 7, no. 7, pp. 1–19, 2023, doi: 10.3390/fractalfract7070547.

[22] Sagaino, T. M. S. Mulyana, I. G. N. Suryantara, J. A. Ginting, and F. Adikara, "Pemetaan Kejadian Bencana Alam Di Jawa Barat Menggunakan Algoritma K-Means," 2022.

[23] D. Mao and J. R. Edwards, *Simulation of chemically-reacting gas-solid flowfields using a preconditioning strategy*. 2003. doi: 10.2514/6.2003-4110.

[24] S. D. Amalia, M. A. Barata, and P. E. Yuwita, "Optimization of Random Forest Algorithm with Backward Elimination Method in Classification of Academic Stress Levels," *J. Appl. Informatics Comput.*, vol. 9, no. 3, pp. 633–641, 2025.

[25] P. Madhumathy, M. Vinoth Kumar, and R. Umamaheswari, *Machine Learning and Iot for Intelligent Systems and Smart Applications*. 2021. doi: 10.1201/9781003194415.

[26] P. Boozary, S. Sheykhan, and H. GhorbanTanhaei, "Forecasting the Bitcoin price using the various Machine Learning: A systematic review in data-driven marketing," *Syst. Soft Comput.*, vol. 7, no. February, p. 200209, 2025, doi: 10.1016/j.sasc.2025.200209.

[27] Y. Si, "Using ARIMA model to analyse and predict bitcoin price," *BCP Bus. Manag.*, vol. 34, pp. 1210–1216, 2022, doi: 10.54691/bcpbm.v34i.3161.

[28] J. Brasil *et al.*, *Advances in Data Science*, vol. 11, no. 1. 2020. doi: 10.1002/9781119695110.

[29] Y. Lou, *Advances in Data Science*. 2020. doi: 10.1002/9781119695110.

[30] P. Overview *et al.*, "• Data Science General • Business Science".

[31] M. Clinton, T. Manullang, A. Z. Rakhman, H. Tantriawan, and A. Setiawan, "Comparative Analysis of CNN, Transformers, and Traditional ML for Classifying Online Gambling Spam Comments in Indonesian," *J. Appl. Informatics Comput.*, vol. 9, no. 3, pp. 592–602, 2025, [Online]. Available: https://jurnal.polibatam.ac.id/index.php/JAIC/article/view/9468

[32] A. Christopher and T. M. S. Mulyana, "Klasifikasi Tumbuhan Angiospermae Menggunakan Algoritma K-Nearest Neighbor Berdasarkan Pada Bentuk Daun," *JIPI (Jurnal Ilm. Penelit. dan Pembelajaran Inform.*, vol. 7, no. 4, pp. 1233–1243, 2022, doi: 10.29100/jipi.v7i4.3211.

[33] Z. Yuan, "Gold and Bitcoin Price Prediction based on KNN, XGBoost and LightGBM Model," *Highlights Sci. Eng. Technol.*, vol. 39, pp. 720–725, 2023, doi: 10.54097/hset.v39i.6635.

[34] D. C. A. Mallqui and R. A. S. Fernandes, "Predicting the Direction, Maximum, Minimum and Closing Prices of Daily Bitcoin Exchange Rate Using Machine Learning Techniques," *Appl. Soft Comput. J.*, 2018, doi: https://doi.org/10.1016/j.asoc.2018.11.038.

[35] D. M. Beazley, *Python Warning and Disclaimer Bulk Sales*. 2006. [Online]. Available: http://dlx.b-ok.org/genesis/79000/73027e552f3bfa7bf634a71bab0744b3/_as/[Beazley]_Python_Essential_Reference(b-ok.org).pdf

[36] D. Cielen, A. Meysman, and M. Ali, *Introducing data science : big data, machine learning, and more, using Python tools*. 2016.

[37] A. Dimitriadou and A. Gregoriou, "Predicting Bitcoin Prices Using Machine Learning," *Entropy*, vol. 25, no. 5, pp. 1–10, 2023, doi: 10.3390/e25050777.

[38] W. Lu, J. Li, Y. Li, A. Sun, and J. Wang, "A CNN-LSTM-based model to forecast stock prices," *Complexity*, vol. 2020, 2020, doi: 10.1155/2020/6622927.

[39] B. Johnston and I. Mathur, *Applied Supervised Learning with Python*. 2019.

[40] Y. Ding, Y. Zhu, J. Feng, P. Zhang, and Z. Cheng, "Interpretable spatio-temporal attention LSTM model for flood forecasting," *Neurocomputing*, vol. 403, 2020, doi: 10.1016/j.neucom.2020.04.110.

[41] Z. Fang, Y. Wang, L. Peng, and H. Hong, "Integration of convolutional neural network and conventional machine learning classifiers for landslide susceptibility mapping," *Comput. Geosci.*, vol. 139, no. October 2019, p. 104470, 2020, doi: 10.1016/j.cageo.2020.104470.

[42] P. Hiskiawan, C.-C. Chen, and Z.-K. Ye, "Processing of electrical resistivity tomography data using convolutional neural network in ERT-NET architectures," *Arab. J. Geosci.*, vol. 16, no. 10, pp. 1–14, 2023, doi: 10.1007/s12517-023-11690-w.

[43] A. Botchkarev, "A new typology design of performance metrics to measure errors in machine learning regression algorithms," *Interdiscip. J. Information, Knowledge, Manag.*, vol. 14, no. January, pp. 45–76, 2019, doi: 10.28945/4184.

[44] M. Amos, "Perbandingan Algoritma Linear Regression, Neural Network, Deep Learning, Dan K-Nearest Neighbor (K-NN) Untuk Prediksi Harga Bitcoin," *JSI J. Sist. Inf.*, vol. 14, no. 1, pp. 2450–2464, 2022, doi: 10.18495/jsi.v14i1.16561.

[45] S. Nakagawa, P. C. D. Johnson, and H. Schielzeth, "The coefficient of determination R2 and intra-class correlation coefficient from generalized linear mixed-effects models revisited and expanded," *J. R. Soc. Interface*, vol. 14, no. 134, 2017, doi: 10.1098/rsif.2017.0213.

[46] D. Chicco, M. J. Warrens, and G. Jurman, "The coefficient of determination R-squared is more informative than SMAPE, MAE, MAPE, MSE and RMSE in regression analysis evaluation," *PeerJ Comput. Sci.*, vol. 7, pp. 1–24, 2021, doi: 10.7717/PEERJ-CS.623.

[47] S. Das, M. Billah, and S. A. Mumu, "A Hybrid Approach for Predicting Bitcoin Price Using Bi-LSTM and Bi-RNN Based Neural Network," *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 13113 LNCS, no. December 2024, pp. 223–233, 2021, doi: 10.1007/978-3-030-91608-4_23.

[48] S. Fakharchian, "Designing a forecasting assistant of the Bitcoin price based on deep learning using market sentiment analysis and multiple feature extraction," *Soft Comput.*, vol. 27, no. 24, pp. 18803–18827, 2023, doi: 10.1007/s00500-023-09028-5.

[49] S. Chan, J. Chu, S. Nadarajah, and J. Osterrieder, "A Statistical Analysis of Cryptocurrencies," *J. Risk Financ. Manag.*, vol. 10, no. 2, p. 12, 2017, doi: 10.3390/jrfm10020012.

[50] P. D. Modi, K. Arshi, P. J. Kunz, and A. M. Zoubir, "A Data-driven Deep Learning Approach for Bitcoin Price Forecasting," *Int. Conf. Digit. Signal Process. DSP*, vol. 2023-June, pp. 8–11, 2023, doi: 10.1109/DSP58604.2023.10167930.

[51] N. G. S. Lakshmi, V. Ajaykumar, B. Ashish, K. Hemalatha, T. Manohar, and V. Sivashankerreddy, "Comparative Bitcoin Price Prediction Using Multiple Machine Learning Techniques," in *Proceedings of the International Conference on Computational Innovations and Emerging Trends (ICCIET 2024*, Atlantis Press

International BV, 2024, pp. 1248–1258. doi: 10.2991/978-94-6463-471-6_120.

[52] I. Letteri, "A Comparative Analysis of Statistical and Machine Learning Models for Outlier Detection in Bitcoin Limit Order Books," in *Outlier Detection in Bitcoin Limit Order Books*, 2025. [Online]. Available: http://arxiv.org/abs/2507.14960

[53] Paul J. Deitel and Harvey Deitel, *Intro to Python for Computer Science and Data Science*. 2020.

[54] N. I. Boyko and O. A. Tkachyk, "Hierarchical clustering algorithm for dendrogram construction and cluster counting," *Informatics Math. methods Simul.*, vol. 13, no. 1–2, pp. 5–15, 2023, doi: 10.15276/imms.v13.no1-2.5.

[55] M. Forina, C. Armanino, and V. Raggio, "Clustering with dendrograms on interpretation variables," *Anal. Chim. Acta*, vol. 454, no. 1, pp. 13–19, 2002, doi: 10.1016/S0003-2670(01)01517-3.

[56] N. S. S. Kiranmai Balijepalli and V. Thangaraj, "Prediction of cryptocurrency's price using ensemble machine learning algorithms," *Eur. J. Manag. Bus. Econ.*, 2025, doi: 10.1108/EJMBE-08-2023-0244.

[57] Y. Z. D. Z. S. C. S. G. Y. Ma, "Single-Image Crowd Counting via Multi-Column Convolutional Neural Network," *CVPR Pap.*, vol. 2, no. 35, pp. 589–597, 2016, doi: 10.1002/slct.201701956.