

Comparison of State-of-art Machine Learning Models' Performances: Evidence from BTC, BNB, ETH, USDC and USDT

Zimeng Yi^{1,a,*}

¹Department of Mathematical Science, Nanyang Technological University, Singapore

a. Yizi0002@e.ntu.edu.sg

**corresponding author*

Abstract: As a matter of fact, cryptocurrency price prediction has become a major trend for financial research especially based on the state-of-art machine learning techniques. This study explores the effectiveness of six machine learning (ML) models in predicting the daily prices of five major cryptocurrencies, i.e., Bitcoin (BTC), Ethereum (ETH), Tether (USDT), BNB (BNB), and USD Coin (USDC). The ML models applied include the Ordinary Least Squares (OLS) model, Random Forest, Long Short-Term Memory (LSTM), XGBoost, LightGBM, and Transformer. The results demonstrate that traditional regression and tree-based ensemble models, particularly OLS and LightGBM, are more effective in predicting cryptocurrency prices, especially for BTC, BNB, and ETH. LightGBM consistently outperforms OLS in minimizing prediction errors, while Random Forest and XGBoost also yield strong, though slightly less accurate, results. Conversely, LSTM and Transformer models underperform, indicating their limitations with the specific characteristics of cryptocurrency data. These findings suggest that traditional and tree-based models are better suited for cryptocurrency price prediction. Future research should investigate the incorporation of additional market indicators and advanced ML architectures to enhance predictive accuracy.

Keywords: Machine learning prediction, cryptocurrency, price forecasting.

1. Introduction

Machine learning (ML) has experienced significant evolution since its early days, progressing from simple algorithms to sophisticated models capable of addressing complex tasks. Initially, the focus was on pattern recognition and data classification, with foundational algorithms like decision trees and support vector machines paving the way for today's advanced techniques. The emergence of deep learning, particularly through neural networks such as Convolutional Neural Networks (CNNs) [1] and Recurrent Neural Networks (RNNs) [2], marked a substantial advancement, enabling notable breakthroughs in areas like image recognition and natural language processing [3].

Over the past few years, the introduction of attention-based models, particularly Transformers, has brought about a transformative change in fields like natural language processing and computer vision [4]. Models such as BERT and GPT, which utilize self-attention mechanisms, have proven to be highly effective in handling large-scale data with unparalleled efficiency [5]. Moreover, the increasing prominence of ensemble methods, which combine several learning algorithms to improve

predictive accuracy, emphasizes the constant innovation in the field of machine learning. Machine learning (ML) is now widespread, affecting various industries like healthcare, finance, and technology. In finance, ML models are being increasingly used for important tasks such as fraud detection, algorithmic trading, and risk assessment [6]. The growing use of ML in financial markets shows its potential to bring significant changes, especially in areas like asset prediction, portfolio management, and other key financial functions. As ML continues to advance, its role in these sectors is expected to grow, leading to more innovation and improved efficiency.

The application of ML in financial asset prediction has garnered significant attention in recent years. Traditional econometric models, while useful, often fall short in capturing the non-linear relationships present in financial data [7]. With its capacity to learn from large datasets and reveal hidden patterns, ML models, have shown promise in this area. Research has demonstrated the efficacy of ML techniques like random forests, gradient boosting, and neural networks in predicting stock prices, cryptocurrency returns, and other financial metrics. For instance, studies using LSTM networks, which excel at time series forecasting, have achieved notable success in predicting stock market trends [8]. Additionally, ensemble methods combining multiple ML models have been effective in improving prediction accuracy [9, 10]. The cryptocurrency markets, known for their extreme volatility and quick price fluctuations, present unique challenge and opportunities for ML-based prediction models. Recent studies have shown that models integrating sentiment analysis, macroeconomic indicators, and technical analysis outperform traditional approaches in forecasting crypto prices.

This study aims to evaluate and compare the performance of state-of-the-art ML models in predicting the prices of five key cryptocurrencies: Bitcoin (BTC), Ethereum (ETH), Tether (USDT), BNB (BNB), and USD Coin (USDC). These cryptocurrencies were selected for their significant market influence and varying characteristics, which present a unique opportunity to test the robustness and adaptability of different ML models. Each of these cryptocurrencies is applied by 6 models which are Ordinary Least Squares model, Random Forest model, Long Short-Term Memory model, XGBoost model, LightGBM model and Transformer.

2. Data and Method

The data for this study comprises daily price information for five major cryptocurrencies: Bitcoin (BTC), Ethereum (ETH), Tether (USDT), BNB (BNB), and USD Coin (USDC). The dataset spans from July 1, 2020, to July 1, 2024, providing 1,462 observations for each cryptocurrency. The data was sourced from investing (<https://www.investing.com/>), ensuring accuracy and reliability. The dataset includes both independent variables (features) and dependent variables (targets). The independent variables consist of historical price data, trading volume, market sentiment indicators, and macroeconomic factors, while the dependent variable is the daily closing price of each cryptocurrency. To prevent overfitting and ensure the generalizability of the models, the data was split into training and testing sets using sci-kit-learn's cross-validation and train-test split functions, with an 80-20 split ratio.

This study evaluates the performance of 6 ML models, which are OLS, Random Forest, Long Short-Term Memory, XGBoost, LightGBM, and transformer. Ordinary Least Squares (OLS) regression is a core method in statistical modelling used to estimate the relationship between a dependent variable and one or more independent variables. The goal of OLS is to identify the best-fitting line by minimizing the sum of squared differences between observed and predicted values. This straightforward approach assumes a linear relationship between variables, making it a useful benchmark for comparing more sophisticated models. Due to its simplicity and effectiveness, OLS is widely used in both academic research and practical applications. OLS aims to minimize the sum of squared residuals (the difference between observed and predicted values):

$$\min_{\beta} \sum_{i=1}^n (y_i - \hat{y}_i)^2 = \min_{\beta} \sum_{i=1}^n (y_i - (\beta_0 + \sum_{j=1}^p \beta_j x_{ij}))^2 \quad (1)$$

where $\hat{y}_i$ is the predicted values of the dependent variable.

Random Forest is a powerful ensemble learning method widely used for both classification and regression tasks. It is an extension of decision trees, designed to improve prediction accuracy and control overfitting by creating a "forest" of decision trees. Each tree in the forest is built using a random subset of the data and features, making Random Forest robust against noise and capable of handling large datasets with high dimensionality. For regression tasks, the ultimate forecast is calculated as the mean of the predictions made by all the trees. In contrast, for classification tasks, the prediction is chosen through a process of majority voting. The randomization in data and features makes Random Forest robust, reduces the correlation among trees, and enhances predictive performance.

Long Short-Term Memory (LSTM) is a specialized form of Recurrent Neural Network (RNN) that excels at capturing and modelling long-term dependencies in sequential data. Traditional RNNs often face challenges in retaining information over extended sequences, primarily due to the vanishing gradient problem. However, LSTMs are designed with mechanisms that allow them to store and retrieve information over longer periods. This capability makes LSTMs highly effective for tasks such as time series forecasting, natural language processing, and speech recognition.

XGBoost (Extreme Gradient Boosting), is a highly efficient and powerful machine learning algorithm. It builds decision trees sequentially, where each tree corrects errors from the previous one. XGBoost is known for its speed, accuracy, and ability to prevent overfitting through advanced regularization. It works well with large datasets and is widely used in both competitions and practical applications due to its versatility in handling various tasks like classification and regression. XGBoost employs second-order derivatives (Hessian) for more precise updates during training. Regularization is built into the framework to prevent overfitting. It also includes advanced features like handling missing values and tree pruning:

$$Objective = \sum_{i=1}^n l(y_i, \hat{y}_i^{(t-1)} + f_t(x_i)) + \Omega(f_t) \quad (2)$$

where l is the loss function, y_i is the true values, $\hat{y}_i^{(t-1)}$ is the prediction from the previous iteration $f_t(x_i)$ is the current tree's prediction and $\Omega(f_t)$ is the regularization term.

LightGBM (Light Gradient Boosting Machine) is a high-performance gradient boosting framework, which designed to be fast, efficient, and scalable, especially for large datasets with many features. LightGBM uses a unique algorithm that grows tree leaf-wise rather than level-wise, which helps reduce loss more effectively and leads to faster training times. It also supports categorical features natively and is well-suited for tasks like classification, regression, and ranking. LightGBM is popular in machine learning due to its ability to handle large-scale data with high accuracy and low memory usage. LightGBM is optimized for large-scale datasets, with a focus on reducing memory usage and speeding up training times. It uses a leaf-wise growth strategy, choosing leaves with the highest loss reduction for splitting, which often results in deeper trees with fewer leaves. This approach allows LightGBM to achieve faster training and better performance on large datasets compared to XGBoost:

$$Split\ Gain = \frac{1}{2} \left[\frac{G_L^2}{H_L + \lambda} + \frac{G_R^2}{H_R + \lambda} + \frac{(G_L + G_R)^2}{H_L + H_R + \lambda} \right] - \gamma \quad (3)$$

where G_L and G_R are the gradients of the left and right leaves; H_L and H_R are the Hessians of the left and right leaves; λ is the regularization term and γ is the minimum loss reduction required for a split.

Transformer models and diffusion models represent two advanced approaches in the field of machine learning and artificial intelligence, each excelling in different tasks. Transformer models, originally developed for natural language processing, have revolutionized the way sequential data is processed, thanks to their self-attention mechanisms. On the other hand, diffusion models, which are

used for data generation, have gained attention for their ability to generate high-quality synthetic data, particularly in fields like image generation and drug discovery.

When evaluating the performance of machine learning models, it is crucial to use appropriate metrics to assess how well the models predict outcomes. Four common evaluation metrics used are R-squared (R^2), Mean Squared Error (MSE), Mean Absolute Error (MAE), and Mean Absolute Percentage Error (MAPE). Here's a brief introduction to each. R-squared (R^2) measures the proportion of the variance in the dependent variable that is predictable from the independent variables. It provides an indication of how well the model fits the data. The value of R^2 ranges from 0 to 1, where 1 indicates a perfect fit and 0 indicates that the model does not explain any of the variability in the data:

$$R^2 = 1 - \frac{SS_{res}}{SS_{tot}} \quad (4)$$

where SS_{res} is the sum of squares of residuals and SS_{tot} is the total sum of squares. Mean Squared Error (MSE) calculates the average of the squared differences between the actual and predicted values. It is more sensitive to large errors, as it penalizes larger deviations more heavily than MAE.

$$MSE = \frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2 \quad (5)$$

where y_i are the actual values and $\hat{y}_i$ are the predicted values. Mean Absolute Error (MAE) represents the average absolute difference between the actual and predicted values. It provides a straightforward measure of prediction accuracy. A lower MAE indicates a better fit as it shows that predictions are closer to the actual values on average:

$$MAE = \frac{1}{n} \sum_{i=1}^n |y_i - \hat{y}_i| \quad (6)$$

where y_i are the actual values and $\hat{y}_i$ are the predicted values. Mean Absolute Percentage Error (MAPE): Expresses the average absolute error as a percentage of the actual values, offering a relative measure of prediction accuracy that is easy to interpret. A lower MAPE indicates better model performance. However, it is sensitive to small actual values, which can result in large percentage errors:

$$MAPE = \frac{100\%}{n} \sum_{i=1}^n \left| \frac{y_i - \hat{y}_i}{y_i} \right| \quad (7)$$

where y_i are the actual values and $\hat{y}_i$ are the predicted values.

3. Results and Discussion

3.1. BTC

Seen from the results given in Table. 1, the OLS model exhibited exceptional performance, with an R^2 of 0.9995 and a very low MAPE of 0.8401%. This indicates that the model was highly accurate in predicting BTC prices, with minimal errors, making it a reliable choice for this dataset. The Random Forest model also performed well, with an R^2 of 0.9992 and a MAPE of 0.9026%. Although slightly less precise than the OLS model, it still demonstrated strong predictive capabilities. However, its higher MSE suggests a slight decrease in accuracy compared to OLS. The LSTM model underperformed significantly, with an R^2 close to 0 and a negative adjusted R^2 . The high MAPE of 55.5634% indicates that the LSTM struggled to accurately predict BTC prices, making it unsuitable for this dataset. The Transformer model also showed poor performance, with an R^2 of -0.0058 and a MAPE of 55.8420%. This suggests that the model was not able to capture the patterns in the BTC price data effectively, resulting in inaccurate predictions. The XGBoost model achieved similar results to Random Forest, with an R^2 of 0.9992 and a MAPE of 0.9408%. While it performed well, it was slightly less accurate than OLS and LightGBM, but still provided robust predictions for BTC prices. The LightGBM model emerged as one of the best performers, with an R^2 of 0.9994 and the

lowest MAPE of 0.7863%. This indicates that LightGBM was highly effective in predicting BTC prices, offering strong accuracy and reliability.

In conclusion, among the different machine learning models applied to BTC price prediction, OLS and LightGBM stood out as the most accurate and reliable, with LightGBM slightly surpassing OLS in minimizing errors. Random Forest and XGBoost also delivered strong performance but were slightly less precise. On the other hand, the LSTM and Transformer models failed to provide accurate predictions, indicating that they may not be suitable for this dataset. Overall, traditional regression-based and tree-based ensemble models proved to be the most effective for predicting BTC prices in this context.

Table 1: Different ML models applied on BTC

ML model	R^2	Adj- R^2	MSE	MAE	MAPE
OLS	0.9995	0.9995	133817.0497	248.9599	0.8401%
Random Forest	0.9992	0.9992	210207.9137	301.5024	0.9026%
LSTM	0.0021	-0.0160	264320377.1796	13722.0868	55.5634%
Transformer	-0.0058	-0.0240	265556412.3933	13760.3125	55.8420%
XGBoost	0.9992	0.9992	203741.8887	310.1703	0.9408%
LightGBM	0.9994	0.9994	163092.7595	266.4019	0.7863%

3.2. BNB

Seen from Table 2, the OLS model demonstrated remarkable accuracy in predicting BNB prices, with an R^2 of 0.9990 and a low MAPE of 2.4857%. This model effectively captured the underlying trends, resulting in minimal errors and high reliability. The Random Forest model also performed well, with an R^2 of 0.9988 and a notably low MAPE of 1.1628%. It provided strong predictive accuracy, though it was slightly less precise than OLS and LightGBM. The LSTM model performed poorly, with an R^2 close to 0 and a negative adjusted R^2 , indicating that it was unable to capture the patterns in the BNB price data. The extremely high MAPE of 171.9076% reflects its significant inaccuracy, making it unsuitable for this task. The Transformer model, like LSTM, struggled with BNB price predictions. It resulted in a negative R^2 of -0.0293 and a high MAPE of 163.5752%, indicating that the model was unable to effectively learn the patterns in the data, leading to substantial prediction errors. The XGBoost model delivered strong performance, with an R^2 of 0.9986 and a MAPE of 1.4573%. While it was slightly less accurate than Random Forest and LightGBM, it still provided reliable predictions with relatively low error margins. The LightGBM model emerged as the best performer, with an R^2 of 0.9991 and the lowest MAPE of 1.2403%. This model offered the most accurate predictions and demonstrated superior ability to capture the nuances in the BNB price data.

Table 2: Different ML models applied on BNB

ML model	R^2	Adj- R^2	MSE	MAE	MAPE
OLS	0.9990	0.9990	25.8636	3.1608	2.4857%
Random Forest	0.9988	0.9988	30.6515	3.3285	1.1628%
LSTM	0.0040	-0.0141	26475.9719	123.0358	171.9076%
Transformer	-0.0293	-0.0479	27296.4818	124.2170	163.5752%
XGBoost	0.9986	0.9985	37.5313	3.4314	1.4573%
LightGBM	0.9991	0.9991	23.7505	3.0255	1.2403%

Among the different machine learning models applied to BNB price prediction, LightGBM and OLS stood out as the most accurate and reliable, with LightGBM slightly outperforming OLS in minimizing prediction errors. Random Forest and XGBoost also demonstrated strong performance, though they were marginally less precise. On the contrary, the LSTM and Transformer models failed to provide accurate predictions, indicating that they may not be suitable for this dataset. Overall, tree-based ensemble models, particularly LightGBM, proved to be the most effective for predicting BNB prices.

3.3. ETH

As shown in Table 3, the OLS model exhibited high accuracy in predicting ETH prices, achieving an R^2 of 0.9991 and a low MAPE of 1.4652%. This indicates that the model was very effective in capturing the price patterns of ETH, resulting in minimal errors. The Random Forest model performed well with an R^2 of 0.9989 and a MAPE of 1.1777%. Although slightly less accurate than OLS, it still provided strong predictive performance with relatively low errors, making it a reliable model for ETH price prediction. The LSTM model struggled significantly, with an R^2 of 0.0030 and a negative adjusted R^2 . The high MAPE of 79.3526% indicates that the model failed to effectively predict ETH prices, rendering it unsuitable for this task. The Transformer model, similar to LSTM, was unable to accurately predict ETH prices. It had a negative R^2 of -0.0029 and a MAPE of 73.2334%, showing that it did not capture the underlying trends in the data, leading to substantial prediction errors. The XGBoost model showed strong performance, with an R^2 of 0.9989 and a MAPE of 1.2561%. While slightly less accurate than LightGBM and OLS, it still delivered reliable predictions with low error margins, making it a competent choice for ETH price prediction. The LightGBM model emerged as the best performer, achieving an R^2 of 0.9992 and the lowest MAPE of 1.0987%. This model was the most effective in capturing the nuances of ETH price movements, offering the most accurate and reliable predictions.

To conclude, in predicting ETH prices, LightGBM and OLS models proved to be the most accurate, with LightGBM slightly outperforming OLS in minimizing prediction errors. Both Random Forest and XGBoost also delivered strong results, albeit with marginally higher error rates. In contrast, the LSTM and Transformer models performed poorly, indicating their inadequacy for this dataset. Overall, tree-based models, particularly LightGBM, demonstrated superior performance in forecasting ETH prices.

Table 3: Different ML models applied on ETH

ML model	R^2	Adj- R^2	MSE	MAE	MAPE
OLS	0.9991	0.9991	952.5466	20.1911	1.4652%
Random Forest	0.9989	0.9988	1106.5973	21.8130	1.1777%
LSTM	0.0030	-0.0151	975506.6758	806.1528	79.3526%
Transformer	-0.0029	-0.0211	981338.9171	782.9536	73.2334%
XGBoost	0.9989	0.9989	1082.5925	22.1806	1.2561%
LightGBM	0.9992	0.9992	802.2746	18.8565	1.0987%

3.4. USDC

As listed in Table 4, the OLS model exhibited exceptional performance, achieving an R^2 of 0.9957 and a remarkably low MAPE of 0.0040%. This indicates that the model almost perfectly captured the relationship between the features and USDC prices, leading to highly accurate predictions. The Random Forest model demonstrated moderate predictive ability, with an R^2 of 0.4648 and a MAPE

of 0.0249%. While the model provided reasonable predictions, it was significantly less accurate than the OLS model, indicating some difficulty in capturing the full complexity of the data. The LSTM model performed poorly, with a negative R^2 of -0.0080 and a high MAPE of 0.0774%. This suggests that the model was unable to effectively learn the underlying patterns in the USDC price data, resulting in considerable prediction errors. The Transformer model yielded the worst performance among all models, with an R^2 of -0.9430 and a MAPE of 0.2909%. This indicates that the model not only failed to capture the patterns in the data but also produced highly inaccurate predictions, making it unsuitable for forecasting USDC prices. The XGBoost model showed modest predictive capability, with an R^2 of 0.4225 and a MAPE of 0.0288%. While it was better than LSTM and Transformer models, its performance was still relatively weak compared to OLS, reflecting its limitations in accurately predicting USDC prices. The LightGBM model performed similarly to XGBoost and Random Forest, with an R^2 of 0.4455 and a MAPE of 0.0251%. Although it was more accurate than the LSTM and Transformer models, it still lagged behind the OLS model in terms of prediction accuracy.

Among the models tested for predicting USDC prices, the OLS model emerged as the clear frontrunner, delivering near-perfect accuracy with minimal errors. In contrast, tree-based models like Random Forest, XGBoost, and LightGBM provided only moderate predictive power, with substantial room for improvement. LSTM and Transformer models, on the other hand, performed poorly, failing to capture the underlying patterns in the data. Overall, the OLS model stands out as the most reliable and accurate method for forecasting USDC prices in this context.

Table 4: Different ML models applied on USDC

ML model	R^2	Adj- R^2	MSE	MAE	MAPE
OLS	0.9957	0.9957	0.0000	0.0000	0.0040%
Random Forest	0.4648	0.4555	0.0000	0.0002	0.0249%
LSTM	-0.0080	-0.0263	0.0000	0.0008	0.0774%
Transformer	-0.9430	-0.9780	0.0000	0.0029	0.2909%
XGBoost	0.4225	0.4124	0.0000	0.0003	0.0288%
LightGBM	0.4455	0.4359	0.0000	0.0002	0.0251%

3.5. USDT

As given in Table 5, the OLS model performed strongly, with an R^2 of 0.8994 and a very low MAPE of 0.0114%. This indicates that the model was highly effective in capturing the relationship between the predictors and USDT prices, resulting in accurate and reliable predictions. The Random Forest model also showed good performance, though slightly less accurate than OLS, with an R^2 of 0.8179 and a MAPE of 0.0149%. While it was effective in modeling the data, it did not achieve the same level of precision as the OLS model. The LSTM model had limited success in predicting USDT prices, with an R^2 of 0.0102 and a higher MAPE of 0.0476%. This suggests that the model struggled to learn the underlying patterns in the data, leading to less accurate predictions. The Transformer model performed poorly, with a negative R^2 of -0.0814 and the highest MAPE of 0.0514% among all models. This indicates that the model was unable to capture the relevant patterns in the data and produced inaccurate predictions, making it unsuitable for this task. The XGBoost model demonstrated moderate predictive power, with an R^2 of 0.7804 and a MAPE of 0.0177%. Although it performed reasonably well, it fell short of the accuracy levels seen with the OLS and Random Forest models. The LightGBM model performed similarly to XGBoost, with an R^2 of 0.7817 and a MAPE of 0.0167%.

While it was better than the LSTM and Transformer models, it still did not reach the high accuracy levels achieved by the OLS model.

The result shows that the OLS model is the most effective method for predicting USDT prices, providing high accuracy and low error rates. While tree-based models like Random Forest, XGBoost, and LightGBM also demonstrated good performance, they were not as precise as OLS. The LSTM and Transformer models, however, struggled significantly, with poor performance and higher error rates. Overall, OLS emerges as the most reliable model for forecasting USDT prices, with the tree-based models offering reasonable alternatives and the neural network-based models being less suitable in this context.

Table 5: Different ML models applied on USDT

ML model	R^2	Adj- R^2	MSE	MAE	MAPE
OLS	0.8994	0.8990	0.0000	0.0001	0.0114%
Random Forest	0.8179	0.8147	0.0000	0.0001	0.0149%
LSTM	0.0102	-0.0077	0.0000	0.0005	0.0476%
Transformer	-0.0814	-0.1009	0.0000	0.0005	0.0514%
XGBoost	0.7804	0.7766	0.0000	0.0002	0.0177%
LightGBM	0.7817	0.7779	0.0000	0.0002	0.0167%

3.6. Limitations and Prospects

While this study provides valuable insights into the effectiveness of various machine learning models for cryptocurrency price prediction, several limitations need to be acknowledged. First, the analysis was confined to five major cryptocurrencies, i.e., BTC, ETH, USDT, BNB, and USDC, excluding many other digital assets that might behave differently. This narrow focus limits the generalizability of the findings to a broader range of cryptocurrencies. Additionally, the dataset covered a four-year period, which, although comprehensive, may not fully capture longer-term market trends or rare events that could impact model performance. Another limitation lies in the reliance on daily closing prices as the primary predictor, neglecting other potentially influential factors such as trading volume, market sentiment, and macroeconomic indicators that could improve the accuracy of the models.

Furthermore, the study revealed that deep learning models like LSTM and Transformer performed poorly in this context, suggesting that these models may not be well-suited for predicting the highly volatile and potentially non-stationary nature of cryptocurrency prices. However, this finding does not dismiss the possibility that these models could perform better with different configurations or additional preprocessing steps, such as data normalization or augmentation. Additionally, the study focused on evaluating model performance in terms of accuracy and error rates but did not address the interpretability of these models, which is crucial for practical applications in finance where understanding the reasoning behind predictions is essential.

Looking forward, future research could overcome these limitations by broadening the range of cryptocurrencies analyzed, incorporating a wider variety of predictors, and extending the time frame of the data. There is also potential to enhance model performance through the use of ensemble techniques, more refined hyperparameter tuning, or hybrid models that combine the strengths of multiple approaches. Another promising direction is the exploration of explainable AI techniques, which could make machine learning models more transparent and accessible to financial professionals who need not only accurate predictions but also a clear understanding of the factors driving those predictions.

4. Conclusion

In conclusion, the results highlight that traditional regression-based models, particularly OLS, and tree-based ensemble models, like LightGBM, are more effective for predicting cryptocurrency prices, especially for BTC, BNB, and ETH. Among these models, LightGBM consistently outperforms OLS in reducing prediction errors, while Random Forest and XGBoost also deliver strong, even though slightly less accurate, results. In contrast, LSTM and Transformer models struggled to produce reliable predictions, suggesting that they may not be well-suited to the specific characteristics of cryptocurrency data. These results indicate that for the cryptocurrencies studied, traditional and tree-based models offer superior predictive capabilities. Looking ahead, future research could explore incorporating additional market indicators and more advanced ML model architectures to further improve prediction accuracy.

References

- [1] Shiri, F.M., Perumal, T., Mustapha, N. and Mohamed, R. (2023). *A Comprehensive Overview and Comparative Analysis on Deep Learning Models: CNN, RNN, LSTM, GRU*. ArXiv: 2305.17473
- [2] Schmidt, R.M. (2019). *Recurrent Neural Networks (RNNs): A gentle Introduction and Overview*. ArXiv: abs/1912.05911
- [3] Torfi, A., Shirvani, R.A., Keneshloo, Y., Tavaf, N. and Fox, E.A. (2020). *Natural Language Processing Advancements By Deep Learning: A Survey*. ArXiv: 2003.01200
- [4] Han, K., Wang, Y., Chen, H., et al. (2022). *A survey on vision transformer*. *IEEE transactions on pattern analysis and machine intelligence*, 45(1), 87-110.
- [5] Yenduri, G., Ramalingam, M., Selvi, G.C., et al. (2024). *Gpt (generative pre-trained transformer)–a comprehensive review on enabling technologies, potential applications, emerging challenges, and future directions*. *IEEE Access*, 12, 54608-54649.
- [6] Bogojevic, A.V. (2021). *Challenges of Financial Risk Management: AI Applications*. *Management:Journal of Sustainable Business and Management Solutions in Emerging Economies*, 26(3), 27–34.
- [7] Raza, F. (2023). *Machine Learning for Financial Forecasting*. *Research Gare*, 1, 96483
- [8] Sunki, A., SatyaKumar, C., Narayana, G., Koppera, V. and Hakeem, M. (2024). *Time series forecasting of stock market using ARIMA, LSTM and FB prophet*. *MATEC Web of Conferences*. 392.
- [9] Ngo, N.T., Pham, A.D., Truong, T.T.H., Truong, N.S., Huynh, N.T. and Pham, T.M. (2022). *An ensemble machine learning model for enhancing the prediction accuracy of energy consumption in buildings*. *Arabian Journal for Science and Engineering*, 1-13.
- [10] Gao, X.Y., Amin, A.A., Shaban, H.H. and Anwar, E.M. (2021). *Improving the accuracy for analyzing heart diseases prediction based on the ensemble method*. *Complexity*, 2021(1), 6663455.