Cryptocurrency Movement and Volatility Prediction using Deep Learning Model

Paper Report

by

Renji Earl Kurniawan	2702252134
Damien Herlnata	2702327684

[Deep Learning - COMP6826001]

[Computer Science]

[Semester 5]

School of Computer Science

UNIVERSITAS BINA NUSANTARA

2025

***Abstract*—Cryptocurrency prediction has been a complex financial time series forecasting problem, given its highly volatile, non-stationary and low signal-to-noise nature of data. Most existing cryptocurrency prediction models focus solely on predicting the price level or direction prediction. While this is essential, a complete understanding of the directional movement of the price and associated market risk/volatility are also necessary for a complete assessment. To achieve this comprehensive prediction, we propose a deep learning model architecture of Bidirectional Long Short-Term Memory (Bi-LSTM) enhanced with an Attention Mechanism. The bidirectional architecture Bi-LSTM effectively captures dependencies both in past and future context within the data, while having the Attention Mechanism dynamically assigns weights to the most significant historical timesteps, ensuring the model focuses on the most relevant information for forecasting. We approached this method using a single multi-headed model to output the movement direction and market volatility while keeping the confidence of the model prediction. Experimental results demonstrate that the proposed model achieved accuracy of 51% on future price movement direction prediction, providing a slight predictive edge over the baseline. Furthermore, the volatility head achieved low regression error for volatility prediction. The study provides a critical analysis of these results, discussing the feasibility of the dual-task approach and the specific learning dynamics observed, including the challenges and limitations faced by the model in directionality bias and avoiding convergence to the mean in volatility regression.**

Keywords—Time Series Forecasting, Cryptocurrency Movement Prediction, Risk/Volatility Prediction, Bi-LSTM, Attention mechanism

I. INTRODUCTION

Modern financial systems rely heavily on fiat money due to its advantages like transferability, divisibility, and durability [1]. However, this centralized model faces a lot of challenges such as government abuse of currency control, hyperinflation [2], and income inequality through the Cantillon effect [3], and much more.

These issues were then addressed by the introduction of the Bitcoin protocol, which implemented the first decentralized blockchain technology [4]. This innovation marked the creation of the world's first successful cryptocurrency.

Cryptocurrency is a decentralized peer-to-peer electronic cash system [4], which is a virtual medium for the exchange and transfer of assets. The decentralized control of cryptocurrencies enables secured electronic fund transfers globally without any central control or government authority, while still being transparent to the public.

Cryptocurrency has grown tremendously as a means of exchange and a store of value for the trading market. Many investors and institutions have heavily invested in cryptocurrencies nowadays. The rise of digital assets, led by cryptocurrencies, has profoundly impacted global financial markets, creating a new and highly significant asset class [5].

However, due to its decentralized and unregulated nature, the market of cryptocurrencies has proven to be much less stable and more volatile compared to traditional state-issued equity markets. The cryptocurrency market is affected by many factors that lead to this high volatility. Therefore, accurate forecasting of their price dynamics is crucial not only for investment strategy optimization but also for robust risk management within the nascent decentralized finance (DeFi) ecosystem [6].

While most existing research focuses solely on predicting the next price level or movement [7], a complete understanding requires assessment of both directional movement and market risk/volatility. This dual forecasting task remains a significant problem and relatively unexplored challenge in the literature.

To address this problem, this paper proposes a novel deep learning architecture of Bidirectional Long Short-Term Memory (Bi-LSTM) enhanced with Attention Mechanism. The Bi-LSTM leverages its recurrent and bidirectional structure to capture long-term temporal dependencies both from both forward and backward directions [8]. Meanwhile, the Attention Mechanism dynamically assigns weights to the most significant historical timesteps, ensuring the model focuses on the most relevant information for forecasting [9].

Specifically, this research primary contributions are summarized as follows:

- The proposal of deep learning architecture utilizing Bidirectional Long Short-Term Memory (Bi-LSTM) network enhanced with an Attention Mechanism for time series analysis in cryptocurrency.
- The implementation of multi-task forecasting capable of simultaneously predicting both cryptocurrency movement and market risk/volatility.
- A comprehensive evaluation demonstrating the proposed model's performance compared to established traditional architecture baselines.

The remainder of this paper is organized as follows: Section II reviews related work in time series cryptocurrency forecasting and architectures used. Section III details the proposed Bi-LSTM with Attention methodology, including data preprocessing and model architecture. Section IV presents the experimental setup, results, and comparative analysis. Finally, Section V concludes the paper and suggests avenues for future research.

II. RELATED WORK

Over the last few years, there has been a surge in research of cryptocurrencies study and prediction given its growing popularity. Initial efforts in forecasting cryptocurrency prices relied heavily on the standard model to capture linear patterns. For instance, the research [10] utilized the GARCH model to analyze Bitcoin volatility, demonstrating the presence of volatility clustering, making this the baseline for risk estimation in crypto markets. While highly effective at modelling the conditional variance, these models rely on a pre-defined function that limits their capacity to capture complex non-linear patterns between the market input and future directional movement [11].

To overcome the linearity constraints of traditional models, recent research has increasingly adopted deep learning models, especially Recurrent Neural Networks (RNN) models, leveraging their capability to act as function approximators for non-linear dependencies [12].

RNN were particularly well-suited due to their structure of internal memory [13] that allows them to process sequential data and capture non-linear temporal dependencies, making them a natural fit for financial time series.

The RNN model that has been an effective standard for time series forecasting is LSTM, which has been widely documented. For example, the paper [14] implemented a basic deep learning model to forecast the movement of price in the crypto markets. Their study validated that LSTM has the capability to learn temporal dynamics of the market, successfully capturing non-linear patterns.

Further research has also explored the task of price fluctuation classification, predicting the rise and fall of the crypto markets. The paper [15] proposed an adaptive deep learning model for classifying cryptocurrency price fluctuations. This work confirms the shift from just predicting price levels to predicting movement direction.

To capture more patterns, researchers have also explored hybrid recurrent architectures. For instance, the paper [16] proposed a hybrid LSTM-GRU model for Bitcoin price prediction. This methodology combines the strength of LSTM for better control over long dependencies and GRU for its computation efficiency to enhance feature learning.

Furthering the research, researchers found the issue of generalization and robust feature extraction has led to the adoption of powerful techniques like Transfer Learning. The study [17] investigated Hybrid Deep Transfer Learning for cryptocurrency trend prediction. Their work highlights the benefit of pre-trained models on related data to improve generalization.

To achieve state-of-the-art performance in complex financial time series, researchers have moved beyond standard unidirectional recurrent units, focusing on models that can maximize contextual information while minimizing the noise from the data.

A critical step in maximizing the temporal dependencies information is the utilization of bidirectional networks. The paper [8] investigated the performance of various recurrent architectures including GRU, LSTM, and Bi-LSTM for cryptocurrency price prediction. Their study validates that the Bi-LSTM structure achieved superior accuracy over other unidirectional architectures. This superiority is rooted from Bi-LSTM's ability to process input sequence in both forward and backward directions, successfully capturing richer contextual dependencies within the data.

Some research has also tried to address the challenge of dynamically prioritizing crucial information and minimizing noise within the data using Attention Mechanism. Introduced by [9], attention revolutionized sequence modeling by allowing the network to assign weight to each input, quantifying its relevance to the final prediction. This mechanism enables the model to focus on important events while suppressing noise in the data.

The trend toward more complex and hybrid deep learning models is further driven by institutional demand for robust forecasting. The paper [18] proposed a deep learning-based cryptocurrency prediction scheme designed for financial institutions. Their work validates the necessity of using advanced multi -layer architectures to handle massive data volume.

Furthering the field, some research tried to achieve higher accuracy by focusing on optimal feature selection. The paper [19], [20] highlighted the impact of feature engineering and selection, demonstrating the need to use engineered inputs and external multivariate data more than just raw price data. Similarly, the paper [21] performed a comparative analysis of machine learning algorithms and showed that model performance relies on the quality and relevance of the input features provided as well.

Some research also tried models incorporating external multimodal features. The paper [22] proposed a multimodal model, PreBit, which uses Twitter FinBERT embeddings to capture textual sentiment for extreme Bitcoin price movement prediction. Similarly, [23] leveraged Twitter comments to enhance their stacking ensemble deep learning model. This integration proves that a broad, feature-rich input set, using social factors and sentiments shows significant improvement to create more robust forecasting solutions.

III. Proposed Methodology

This study proposes a deep learning model designed to capture both temporal dependencies and relative importance of certain market events within the time series cryptocurrency data. The model integrates a Bidirectional Long Short-Term Memory (Bi-LSTM) network with a custom Attention Mechanism to perform dual-task forecasting of cryptocurrency directional movement and volatility regression.

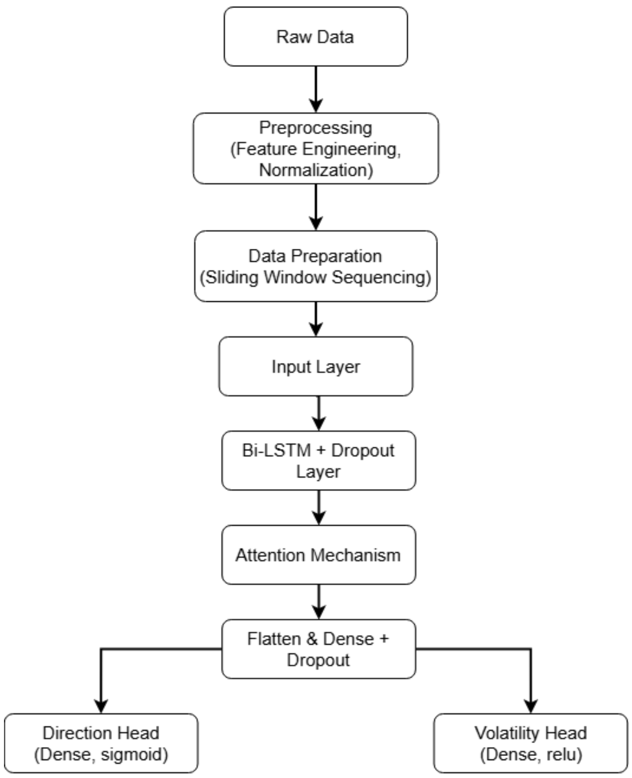

Figure 1: Proposed Multi-Task Bi-LSTM + Attention Model

A. Model Sequence Pipeline

The proposed model follows a sequential pipeline, progressing from raw data ingestion to multi-task forecasting. As illustrated in Fig.1, the architecture consists of data preprocessing, preparation, Bi-LSTM and Attention Mechanism through a shared encoder. This shared encoder then feeds into two distinct output heads dedicated to simultaneously predict the movement direction and predict the market volatility.

The component is responsible for ingesting the prepared time series data and it into a robust, high-level representation. The input to the model is a three-dimensional tensor of shape (N, Timesteps, Features), representing the batch size, the lookback window, and the number of features engineered, respectively.

The first stage of the model is a Bi-LSTM layer comprising 64 units. Unlike standard unidirectional LSTM that process data strictly forward, the Bi-LSTM employs two separate hidden layers to process the input sequence in both forward and backward directions, successfully capturing temporal dependencies within the observation window. The Bi-LSTM layer is also added with a Dropout layer with 0.3 rate to reduce overfitting and ensure the model generalizes better within the highly volatile data.

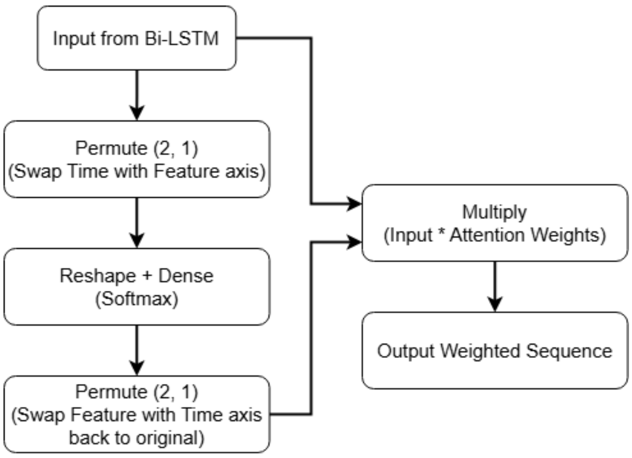

Figure 2: Attention Layer Mechanism

B. Attention Mechanism

Following the Bi-LSTM layer, an Attention Mechanism layer is applied to tackle the low signal-to-noise ratio inherent in financial data. In addition, not all timesteps in financial data contribute equally to the future price. Some timesteps that consist of sudden spikes or drops might carry more predictive weight than periods of stagnation. The attention layer allows the model to focus on these critical moments by learning to assign dynamic weights to each timestep based on its relevance to the prediction task.

As shown in Fig. 2, the Attention Mechanism uses the output from Bi-LSTM layer. The input tensor is first permuted to align the time axis with the dense layer. A trainable dense layer with a softmax activation is then applied to compute attention score for each time step. The softmax activation ensures that the sum of all attention scores equals to 1, creating a probability distribution over time sequence.

These scores are then permuted back to match the original input shape. Finally, the original input sequence is multiplied element-wise with the attention scores, amplifying timesteps with high attention scores and dampening those with low attention scores. The result is a weighted sequence that will be passed to the next layer of the model.

This weighted sequence is then passed to a Flatten layer converting the weighted 3D sequence into a 1D context vector. This context vector is passed through a Dropout layer with 0.3 rate for regularization, followed by a fully connected Dense layer with 64 units and ReLU activation, acting as the final feature transformation stage before branching into its own task head.

C. Multi-Task Output Heads

The architecture then branches into two task-specific output heads. The first branch is the Directionality Head, designed for the binary classification task of predicting future price movement. This head consists of a Dense layer with 32 units of neuron with ReLU activation and a Dense output neuron with Sigmoid activation. It outputs a probability score between 0 and 1, indicating the likelihood of the future price movement.

The second branch is the Volatility Head, designed for the regression task of predicting market volatility. This head also consists of a Dense layer with 32 units of neuron with ReLU activation and a Dense output neuron with ReLU activation to predict continuous numerical value representing the anticipated price volatility.

To train the model simultaneously for both tasks, we define a composite loss function that aggregates the errors from both output heads. For the classification task, Binary Cross-Entropy (BCE) is used to measure the divergence between predicted probabilities and actual directional labels. For the regression task, Mean Squared Error (MSE) is used to quantify the average squared difference between predicted and actual volatility values. The total loss used for backpropagation is a weighted sum of these two individual losses.

To prioritize the model's performance on directional prediction, the Directionality Head is assigned a significantly higher loss weight. Specifically, the binary cross-entropy loss is scaled by a coefficient of 50.0, while the mean squared error retains a weight of 1.0. The joint loss function is defined as:

$$L_{total} = \lambda_{dir} \cdot L_{BCE} + \lambda_{vol} \cdot L_{MSE}$$

Where λdir = 50.0 and λvol = 1.0

D. Dataset

This research utilized the historical cryptocurrency market data for Bitcoin (BTC), sourced from the Binance exchange via Kaggle data repository [24]. The selected data covers the period from January 1, 2018 to December 8, 2025, offering a comprehensive view of multiple market cycles.

While the repository consists of various time interval structures, this study specifically chose the hourly (1-hour) interval structure. This granularity offers a time series suitable for capturing short-to-medium-term price movements.

The dataset includes the standard market key feature vectors for each time step, including Open, High, Low, Close, and Volume (OHLCV).

IV. Experiments, result, and analysis

A. Implementation Details

Data Loading: For this study, we specifically extracted the essential Open, High, Low, Close, and Volume (OHLCV) columns from the dataset and discarded the other columns. The ‘Open time’ timestamps were parsed and set as the dataframe index to ensure chronological sequence.

Feature Engineering: To enhance the model’s ability to detect market patterns, we expanded the feature set beyond raw prices and created stationary data. Raw price data is often non-stationary, which can hinder deep learning convergence. Therefore, we engineered features, including logarithmic returns, logarithmic volume, candlestick shadows, technical Indicators like RSI and normalized MACD, and lastly target generation of direction and volatility.

Logarithmic returns are used to normalize hourly price changes and achieve stationarity. Similarly, logarithmic volume is the natural logarithm of volume change ratio computed to capture relative market activity independent of volume spikes.

Candlestick shadows include upper shadow and lower shadow, which were calculated relative to the closing price. These features capture intraday price rejection levels, offering insights into market sentiment often missed by closing prices alone.

Technical indicators include Relative Strength Index (RSI) and Normalized Moving Average Convergence Divergence (MACD). RSI was computed over a 14-period window to identify potentially overbought or oversold market conditions. Normalized MACD was calculated as the difference between 12-period and 26-period Exponential Moving Averages (EMA). This value was normalized by closing price to maintain scale consistency across different market cycles.

Target generation includes target direction and target volatility, both calculated in the prediction horizon of 1 hour

. Target direction is a binary classification label where 1 indicates a positive return and 0 indicates a negative return. Target volatility is a regression target derived from the absolute value of the future returns, serving as a proxy for risk magnitude.

Data Cleaning: Any rows containing missing values after preprocessing were removed. Any rows containing infinite values as a result of logarithmic calculations were replaced with zeros to ensure numerical stability during training.

Data Splitting & Sliding Window Generation: To maintain temporal integrity of the time series and simulate a realistic trading environment, the dataset was partitioned chronologically without shuffling. The data was divided into three subsets including 80% of dataset as training set used to optimize the model weights, 10% of dataset as validation set used for checking model’s performance and perform hyperparameter tuning or early stopping to prevent overfitting, 10% of dataset as testing set strictly held-out solely used for the final performance evaluation.

Following the split, Min-Max Normalization was applied to scale the feature values into the range 0 to 1. The scaler was fitted only on the training set to ensure no data from the future validation or testing set was leaked into the training process. The computed scalers were then applied to transform the whole set, including training, validation, and testing set.

To prepare for the 3D input tensors required by Bi-LSTM, a Sliding Window technique is used, where we define a sequence length of 60 hours as a lookback. For every timestep t, the model receives an input matrix containing features from 60 hours lookback (t-60 to t-1) to predict the target at the current time (t). This results in a final input shape of (N, 60, F), where N is the number of rows and F is the number of input features.

Training: The deep learning model employed was implemented using TensorFlow/Keras framework. The model was trained with Adam as the optimizer initialized with a learning rate of 5e-4 (0.0005) to ensure stable convergence.

The model was trained for a maximum of 50 epochs with a batch size of 64. To prevent overfitting and optimize training efficiency, we implemented three callback mechanisms, namely, early stopping, ReduceLROnPlateau, and model checkpoint.

Early stopping is used to stop the model training if the validation loss did not improve for 15 consecutive epochs, with the weights restored to the epoch that achieved the lowest validation loss.

ReduceLROnPlateau is used to lower the learning rate, reducing it by a factor of 0.5 with a lower bound learning rate of 1e-6, if the validation loss stagnates for 5 epochs. This allows the model to slow down on the gradient descent, possibly learning and gaining better weights.

Model checkpoint is used to save model state when a minimum validation loss was achieved, ensuring the final evaluated model represents the best generalization performance rather than the state at the final epoch.

B. Evaluation

To assess the model’s performance on the dual forecasting task, we used the standard classification report that includes accuracy, precision, recall, and f1-score for the binary classification task in directionality head. Meanwhile, for the regression task, we evaluated the deviation between predicted and actual volatility magnitude using mean absolute error (MAE), mean squared error (MSE), and root mean squared error (RMSE).

The model was evaluated on the testing set comprising the final 10% of the dataset timeline starting from August 31, 2023 to December 8, 2025.

TABLE I. Classification Report for Directionality Head

	precision	recall	f1-score	support
Turun (0)	50.10%	24.09%	32.53%	9,780
Naik (1)	51.26%	76.89%	61.51%	10,156
accuracy		50.99%		19,936
macro avg	50.68%	50.49%	47.02%	19,936
weighted avg	50.69%	50.99%	47.30%	19,936

As shown in Table I, the model achieved an accuracy of 51% on the test set. While this margin appears quite low compared to other domains, in financial time series forecasting, a consistent edge of 2-3% above the threshold 50% is considered significant and potentially profitable. The classification report further breaks down the performance of each class, revealing a weighted average of f1-score around 47.09%. This validates that the model has developed a directional bias, favoring the label 'Up', resulting in lower f1-score. The precision scores remain balanced around 50% to 51%, suggesting that while model is hesitant to predict downturns, its reliability remains consistent when a prediction is made.

TABLE II. Regression Metrics for Volatility Head

MAE	RMSE	MSE
0.00658	0.01	0.0001

As shown in Table II, the regression head of volatility prediction achieved a mean absolute error (MAE) of 0.00658, root mean squared error of 0.00998, and mean squared error (MSE) of 0.0001. However, this low error rate is highly misleading, as qualitative analysis revealed the model volatility head has suffered from lazy training, predicting values near zero every single time. This behavior indicates the model has failed to quantify market turbulence effectively and merely converged to the average volatility of the dataset.

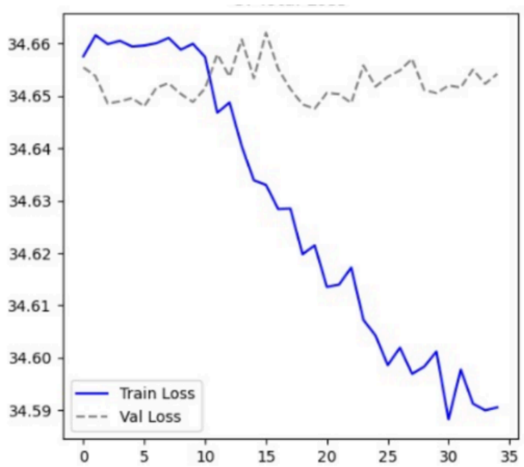

Figure 3: Training History Directional Loss

The illustration in Fig. 3 shows the plot of training and validation loss on the directionality head throughout the model training loops. The training loss exhibits a consistent downward trend, indicating the model is learning patterns within the training set.

However, the validation loss reveals significant instability. While it initially tracks with the training loss, it begins to diverge significantly. The validation loss spikes and trends upward, indicating that the model is overfitting. It suggests that while the model has the capacity to memorize training data, it struggles to generalize these patterns to unseen validation data. Therefore, the model training is stopped via early stopping and the best weight is restored and saved via model checkpoint.

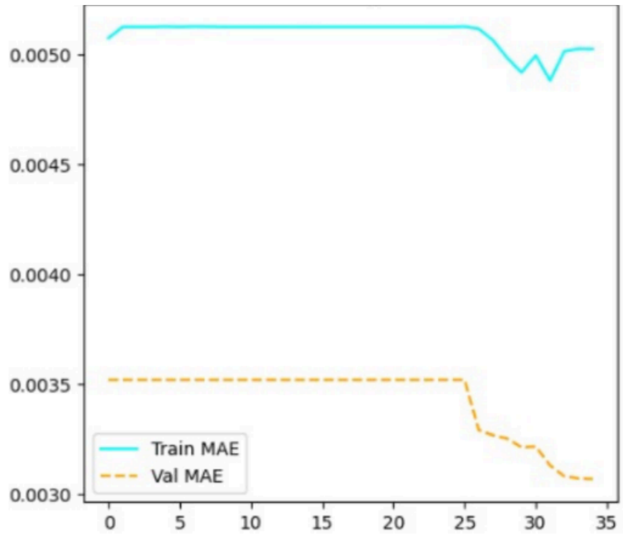

Figure 4: Training History Volatility Loss

The illustration shown in Fig. 4 is the plot for training and validation loss on the volatility head throughout the model training. The training and validation loss shows prolonged periods of stagnation, where both losses remain almost perfectly constant. This behavior strongly correlates the diagnosis of lazy training, suggesting that the model settled on a trivial solution, predicting constantly only one a static mean value for volatility rather than learning complex temporal features.

A distinct step-down in error is observed around epoch 25, which likely means the model is starting to learn, possibly from the learning rate reduction mechanism initialized for training callback, though it's stopped early because of the early stopping callback to prevent overfitting on the directionality head. Throughout the training process, the validation error remains constantly lower than the training error, suggesting that the volatility patterns in the validation set are possibly less complex or have lower variance compared to the training set, further complicating the model's ability to generalize robustly.

V. CONCLUSION

This paper presented a novel dual-task deep learning framework for cryptocurrency forecasting on future price movement direction and its volatility/risk. By integrating a Bi-LSTM model enhanced with a custom attention mechanism, we successfully created a model capable of simultaneously predicting price direction and market volatility. Experimental results demonstrate that the proposed architecture achieves accuracy of 51% on predicting future price movement. While this performance provides a marginal predictive edge, further analysis reveals a directional bias towards predicting upward trends. Regarding volatility prediction, the model achieved low regression error scores. However, these low error scores were largely due to lazy training, where the model converged to a mean-prediction strategy rather than learning dynamic volatility spikes.

To demonstrate the practical utility of this framework, the trained model has been deployed as a publicly accessible web application via Streamlit [25]. Future work will focus on mitigating the observed directional bias and addressing the lazy training on volatility head, integrating data sources such as social sentiment to improve the signal-to-noise ratio in the highly volatile cryptocurrency market.

1. Acknowledgement

The authors would like to express their deepest gratitude to the Faculty of Computer Science, Bina Nusantara University, for the support and guidance provided during the making of this research. This research would not have been possible without the academic environment and encouragement provided by the supervisors. We appreciate the opportunity and knowledge that has contributed greatly to the completion of this research

2. REFERENCES

[1] W. S. Jevons, “Money and the Mechanism of Exchange,” in *General Equilibrium Models of Monetary Economies*, Elsevier, 1989, pp. 55–65. Accessed: Dec. 12, 2025. [Online]. Available: https://www.sciencedirect.com/science/article/pii/B9780126639704500098

[2] P. Bernholz, “Monetary regimes and inflation: History, economic and political relationships,” in *Monetary Regimes and Inflation*, Edward Elgar Publishing, 2015. Accessed: Dec. 12, 2025. [Online]. Available: https://www.elgaronline.com/monobook/9781784717629.xml

[3] J. G. Hülsmann, “Fiat Money and the Distribution of Incomes and Wealth,” in *The Fed at One Hundred*, D. Howden and J. T. Salerno, Eds., Cham: Springer International Publishing, 2014, pp. 127–138. doi: 10.1007/978-3-319-06215-0_12.

[4] S. Nakamoto, "Bitcoin: A peer-to-peer electronic cash system," 2008. [Online]. Available: https://bitcoin.org/bitcoin.pdf.

[5] S. Corbet, B. Lucey, A. Urquhart, and L. Yarovaya, “Cryptocurrencies as a financial asset: A systematic analysis,” *Int. Rev. Financ. Anal.*, vol. 62, pp. 182–199, 2019.

[6] S. Kitzler, F. Victor, P. Saggese, and B. Haslhofer, “Disentangling Decentralized Finance (DeFi) Compositions,” *ACM Trans. Web*, vol. 17, no. 2, pp. 1–26, May 2023, doi: 10.1145/3532857.

[7] O. B. Sezer, M. U. Gudelek, and A. M. Ozbayoglu, “Financial time series forecasting with deep learning: A systematic literature review: 2005–2019,” *Appl. Soft Comput.*, vol. 90, p. 106181, 2020.

[8] M. J. Hamayel and A. Y. Owda, “A Novel Cryptocurrency Price Prediction Model Using GRU, LSTM and bi-LSTM Machine Learning Algorithms,” *AI*, vol. 2, no. 4, pp. 477–496, Dec. 2021, doi: 10.3390/ai2040030.

[9] A. Vaswani, “Attention is All you Need,” *Advances in Neural Information Processing Systems*, vol. 30. p. I, 2017. Accessed: Dec. 12, 2025. [Online]. Available: https://cir.nii.ac.jp/crid/1370849946232757637

[10] P. Katsiampa, “Volatility estimation for Bitcoin: A comparison of GARCH models,” *Econ. Lett.*, vol. 158, pp. 3–6, Sept. 2017, doi: 10.1016/j.econlet.2017.06.023.

[11] P. R. Hansen and A. Lunde, “A forecast comparison of volatility models: does anything beat a GARCH(1,1)?,” *J. Appl. Econom.*, vol. 20, no. 7, pp. 873–889, Dec. 2005, doi: 10.1002/jae.800.

[12] A. M. Schäfer and H. G. Zimmermann, “Recurrent Neural Networks Are Universal Approximators,” in *Artificial Neural Networks – ICANN 2006*, S. D. Kollias, A. Stafylopatis, W. Duch, and E. Oja, Eds., Berlin, Heidelberg: Springer, 2006, pp. 632–640. doi: 10.1007/11840817_66.

[13] J. L. Elman, “Finding Structure in Time,” *Cogn. Sci.*, vol. 14, no. 2, pp. 179–211, 1990, doi: 10.1207/s15516709cog1402_1.

[14] A. V., N. M.V.S., P. I., S. Reddy M., and A. K. G., “A Cryptocurrency Price Prediction Model using Deep Learning,” *E3S Web Conf.*, vol. 391, p. 01112, 2023, doi: 10.1051/e3sconf/202339101112.

[15] A. S. El-Berawi, M. A. F. Belal, and M. M. A. Ellatif, “Adaptive Deep Learning based Cryptocurrency Price Fluctuation Classification,” *Int. J. Adv. Comput. Sci. Appl.*, vol. 12, no. 12, 2021, doi: 10.14569/IJACSA.2021.0121264.

[16] Nashwan Jasim Hussein, and Adnan Mohsin Abdulazeez, “Bitcoin Price Prediction Using Hybrid LSTM-GRU Models,” *Indonesian Journal of Computer Science*, 2024, doi: 10.33022/IJCS.V13I1.3725.

[17] K. Jahanbin and M. A. Z. Chahooki, “Cryptocurrency Trend Prediction Through Hybrid Deep Transfer Learning,” *Int. J. Intell. Syst.*, vol. 2025, no. 1, p. 4211799, 2025, doi: 10.1155/int/4211799.

[18] M. M. Patel, S. Tanwar, R. Gupta, and N. Kumar, “A Deep Learning-based Cryptocurrency Price Prediction Scheme for Financial Institutions,” *J. Inf. Secur. Appl.*, vol. 55, p. 102583, Dec. 2020, doi: 10.1016/j.jisa.2020.102583.

[19] Z. Chen, C. Li, and W. Sun, “Bitcoin price prediction using machine learning: An approach to sample dimension engineering,” *J. Comput. Appl. Math.*, vol. 365, p. 112395, Feb. 2020, doi: 10.1016/j.cam.2019.112395.

[20] J. Chen, “Analysis of bitcoin price prediction using machine learning,” *J. Risk Financ. Manag.*, vol. 16, no. 1, p. 51, 2023.

[21] T. K. Samson, “Comparative Analysis of Machine Learning Algorithms for Daily Cryptocurrency Price Prediction,” *Inf. Dyn. Appl.*, vol. 3, no. 1, pp. 64–76, Mar. 2024, doi: 10.56578/ida030105.

[22] Y. Zou and D. Herremans, “PreBit—A multimodal model with Twitter FinBERT embeddings for extreme price movement prediction of Bitcoin,” *Expert Syst. Appl.*, vol. 233, p. 120838, 2023.

[23] 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, p. 1307, 2022.

[24] N. Anugrah, "BITCOIN Historical Datasets 2018-2025 Binance API," Kaggle, 2025. [Online]. Available: https://www.kaggle.com/datasets/novandraanugrah/bitcoin-historical-datasets-2018-2024. [Accessed: Dec. 12, 2025].

[25] "Cryptocurrency Movement and Volatility Prediction using Deep Learning Model," Web Application, 2025. [Online]. Available:

https://waterlemonnn-cryptocurrency-movement-and-volatili-appapp-q0e7p1.streamlit.app [Accessed: Dec. 12, 2025].

3. Appendix

Team Contribution:

- Damien Herlnata: Model experimentation, architectural diagramming, survey of related work, manuscript drafting.
- Renji Earl Kurniawan: Idea conceptualization, model experimentation, model architecture development, web application development.