Volume: 13 Issue: I Month of publication: January 2025

DOI: https://doi.org/10.22214/ijraset.2025.66472

Predicting Ethereum Price Using Machine Learning Models: A Comparative Analysis

Shriya Tyagi

Abstract: The cryptocurrency market, known for its high volatility and immense data availability, provides an excellent opportunity for predictive modeling. This paper explores the prediction of Ethereum's price using four distinct models: Random Forest, Logistic Regression, Long Short-Term Memory Networks (LSTM), and CNN-LSTM hybrid models. The study evaluates the performance of these models based on metrics such as Mean Squared Error (MSE), Root Mean Squared Error (RMSE), R-squared (R^2), and Accuracy (%). The findings highlight that Logistic Regression outperformed the other models with the lowest MSE (6741.12) and highest accuracy (98.66%) [Table 1]. This research demonstrates the potential of combining traditional and advanced machine learning techniques to achieve robust price prediction in the cryptocurrency domain.

I. INTRODUCTION

Ethereum, the second-largest cryptocurrency by market capitalization, has garnered significant attention in financial markets. Predicting Ethereum's price is crucial for traders, investors, and financial analysts. Its decentralized nature, combined with factors like trading volumes, market sentiment, and global adoption, presents both opportunities and challenges.

Cryptocurrencies, unlike traditional stocks, lack the centralized regulations and standard valuation models that guide financial markets. For instance, while traditional stock markets rely on models like CAPM [2] or DCF [3] for price predictions, cryptocurrencies are influenced by factors such as network activity, blockchain metrics, and external events like regulatory announcements. This lack of standard valuation necessitates innovative approaches such as machine learning for accurate predictions.

According to CoinGecko, Ethereum's daily trading volume in March 2024 averaged approximately
$2.25 billion, reflecting its significant market activity [4]. Such a high trading volume highlights the need for predictive models that can help traders optimize entry and exit points, potentially maximizing profits. Given its volatility, even marginal improvements in prediction accuracy can translate into significant financial gains for traders and institutions.

This study aims to shed light on how machine learning models can effectively capture these complex- ities to predict Ethereum's price accurately. Additionally, recent research has demonstrated that integrating advanced deep learning methods with traditional models significantly enhances prediction accuracy [5].

The paper focuses on four models:

1) Random Forest [6]
2) Logistic Regression [7]
3) LSTM [8]
4) CNN-LSTM Hybrid [9]

II. DATA DESCRIPTION

A. Dataset

The dataset spans Ethereum's historical price data, including:

- Open, High, Low, Close prices (OHLC)
- Volume traded
- Moving averages (MA50, MA200)
- Sentiment analysis data sourced from cryptocurrency-related news and social media [10, 20].

Source: Data was aggregated from CoinMarketCap [11], Yahoo Finance [12], and a custom sentiment analysis tool leveraging Twitter API. Sentiment data was further validated using a secondary dataset from CryptoCompare [13] to ensure consistency.

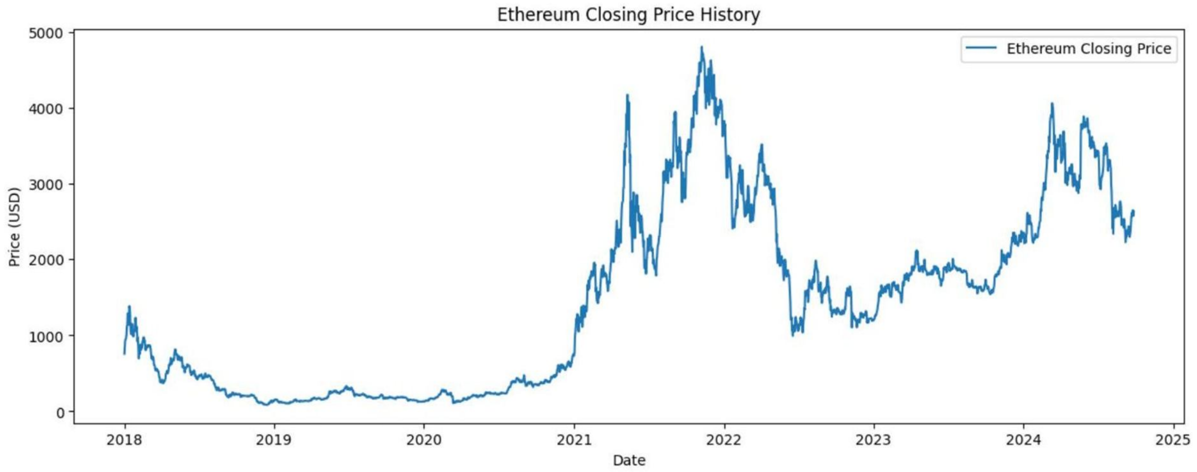

Figure 1: Ethereum Closing Price History

Figure 1 displays Ethereum's historical closing price trends, highlighting the volatility and significant market events over time. This dataset serves as the basis for feature extraction and model training.

B. Preprocessing

The raw data underwent comprehensive preprocessing steps:

- Handling missing values by linear interpolation.
- Feature engineering, including adding lagged variables for temporal analysis.
- Scaling the features using MinMaxScaler to normalize data for machine learning models.
- Data was split into training (80%) and testing (20%) sets, ensuring no data leakage.

C. Sentiment Analysis Integration

Sentiment scores were computed using the VADER (Valence Aware Dictionary and sEntiment Reasoner) tool [10]. Tweets and news articles were preprocessed to extract sentiment polarity, later integrated as a feature in the dataset. Additionally, financial news sentiment classification was cross-validated using Google Cloud Natural Language API, which provided confidence scores for sentiment analysis accuracy.

D. Market Context

On average, Ethereum's market cap fluctuates between $100 billion to $500 billion, depending on broader market trends. For instance, Ethereum reached its highest market capitalization of approximately $571.67 billion in November 2021 [14, 21]. As of January 4, 2025, Ethereum's market cap was around $434.29 billion [1]. These figures illustrate the dynamic nature of Ethereum's market value, influenced by shifting market conditions.

Traditional financial markets benefit from well-established predictive tools, such as Bollinger Bands and the Relative Strength Index (RSI), which are widely used for forecasting price movements [15]. The cryptocurrency market, in contrast, operates in an unregulated and 24/7 trading environment, making predictive modeling even more critical due to its susceptibility to sudden price swings [16]. Additionally, the lack of regulation in the crypto market can lead to market manipulation, further increasing volatility and the necessity for effective predictive tools [17].

Algorithmic trading accounts for approximately 60-75% of transactions in traditional financial mar- kets, such as U.S. equities [18]. This makes automation in predictive modeling highly valuable, with potential applications in automated trading strategies and hedging mechanisms.

III. METHODOLOGY

A. Random Forest

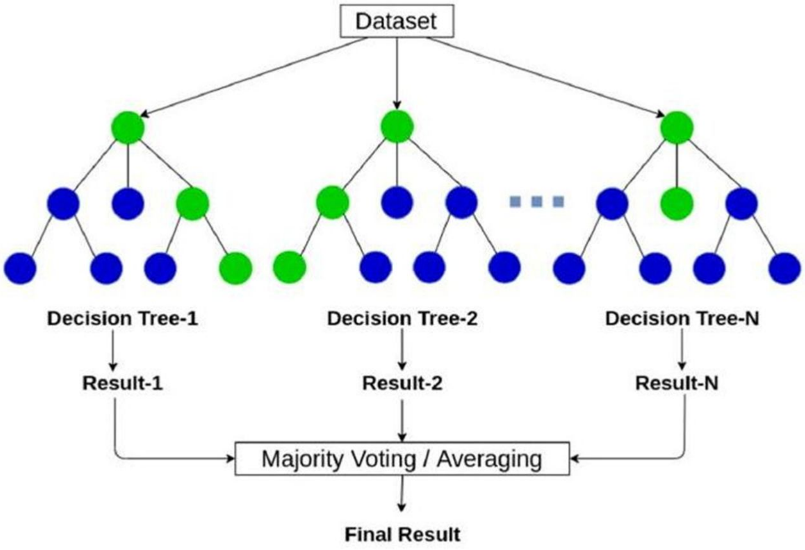

Figure 2: Random Forest algorithm’s ensemble approach

Random Forest is an ensemble learning method combining multiple decision trees. Its ability to handle non-linear relationships and noise in data makes it a strong candidate for this task. Hyperparameter tuning was performed for the number of estimators, maximum depth, and minimum samples per split.

B. Logistic Regression

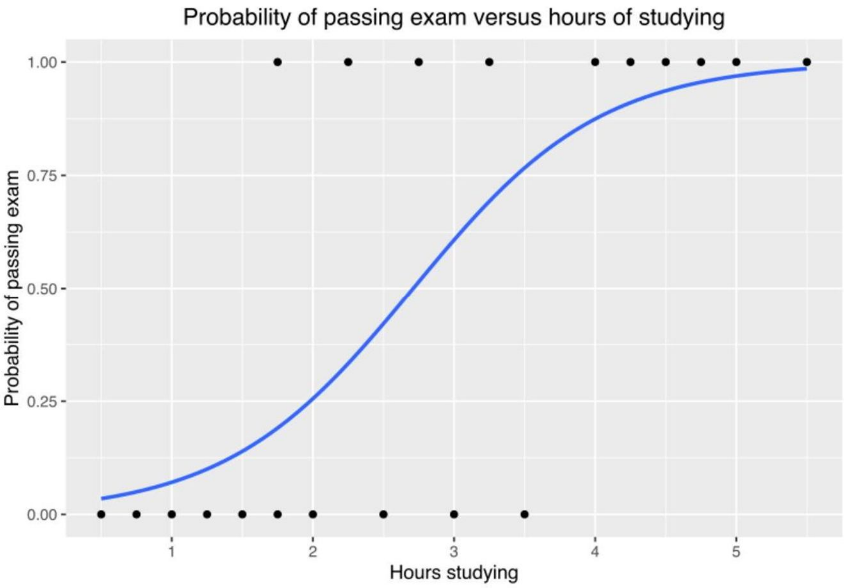

Figure 3: Example graph of a logistic regression curve fitted to data.

While traditionally a classification algorithm, Logistic Regression was adapted for regression by treat- ing price movement direction as a probabilistic outcome. It was chosen for its simplicity and interpretabil- ity, offering a baseline comparison against more complex models.

C. Long Short-Term Memory (LSTM)

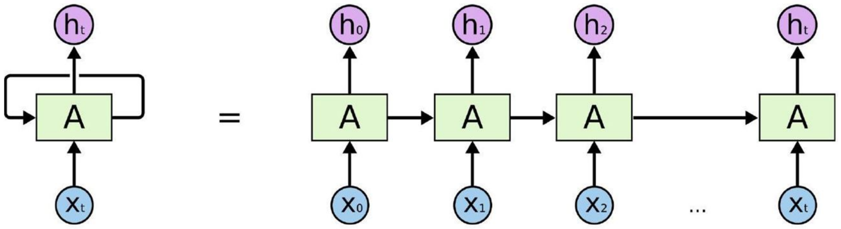

Figure 4: Architecture of LSTM networks

LSTMs are a type of Recurrent Neural Network (RNN) designed to capture long-term dependencies. The model architecture included:

```
# LSTM Model
    lstm_model = Sequential([
        Input (shape=(sequence_length, 1)), LSTM(50, return_sequences=True), Dropout(0.2),
        LSTM(50, return_sequences=False), Dropout(0.2),
        Dense(25), Dense(1)
    ])
    lstm_model.compile( optimizer='adam', loss='mean_squared_error'
    )
    lstm_model.fit(
        X_train_lstm, y_train_lstm, epochs=50,
        batch_size=32, validation_data=(X_test_lstm, y_test_lstm)
    )
    lstm_predictions   =   lstm_model.predict(X_test_lstm)
```

Figure 5: Code snippet of the LSTM model configurations used.

- 2 LSTM layers with 50 units each
- Dropout layers to prevent overfitting
- A dense layer with a linear activation function for final predictions.

D. CNN-LSTM Hybrid

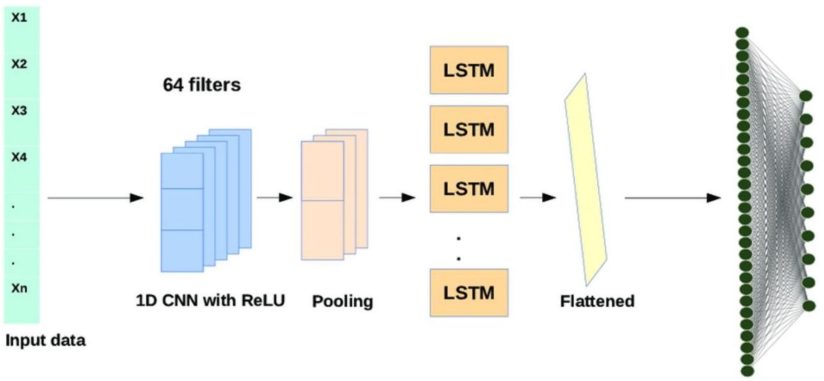

Figure 6: Architecture of CNN + LSTM network

The CNN-LSTM hybrid model combines:

- Convolutional layers for extracting spatial features from input sequences.
- LSTM layers to handle temporal dependencies.

Recent research highlights the effectiveness of such hybrid models in time series forecasting [9]. The architecture included:

```
# CNN-LSTM Model
cnn_lstm_model = Sequential([ # CNN Layers
    Conv1D(
        filters=64, kernel_size=3, activation='relu',
        input_shape=(sequence_length, 1)
    ),
    MaxPooling1D(pool_size=2), # LSTM Layers
    LSTM(50, return_sequences=True), Dropout(0.2),
    LSTM(50, return_sequences=False), Dropout(0.2),
    # Dense Layers
    Dense(25, activation='relu'),
    Dense(1)  # Output layer for price prediction
])
```

Fig 7: CNN-LSTM model architecture with configurations used.

- A convolutional layer with 64 filters and a kernel size of 3 .
- MaxPooling layer for feature scaling.
- LSTM layers with 50 units, followed by dense layers for output.

IV. RESULTS

A. *Performance Metrics*

The models were evaluated using:

- Mean Squared Error (MSE): Measures average squared errors between actual and predicted values.
- Root Mean Squared Error (RMSE): Square root of MSE, providing error in the same units as the target variable.
- R-squared ($\mathbf{R}^2$): Proportion of variance explained by the model.
- Accuracy (%): Percentage of correct predictions within a predefined threshold.

±	Model	MSE	RMSE	R^2	Accuracy (%)
	Random Forest	9395.24	96.9291	0.981341	98.1341
	Logistic Regression	6741.12	82.1043	0.986612	98.6612
	LSTM	10136.9	100.682	0.979868	97.9868
	CNN-LSTM	14018.5	118.4	0.97216	97.216

Table 1: Performance Metrics for Machine Learning Models

Table 1 provides a comparative evaluation of the models based on key metrics such as MSE, RMSE, R^2, and accuracy. Furthermore, leveraging neural architecture search has been shown to improve model performance in dynamic environments [19]. Logistic Regression achieved the best performance with an MSE of 6741.12 and an accuracy of 98.66%.

B. *Visualization*

- Actual vs Predicted Prices for All Models: All models demonstrated an ability to track Ethereum's price trends. Logistic Regression showed the most consistent performance.
- Hybrid Model Performance vs Actual Price: While CNN-LSTM captured complex patterns, its higher errors suggest overfitting to the training data.

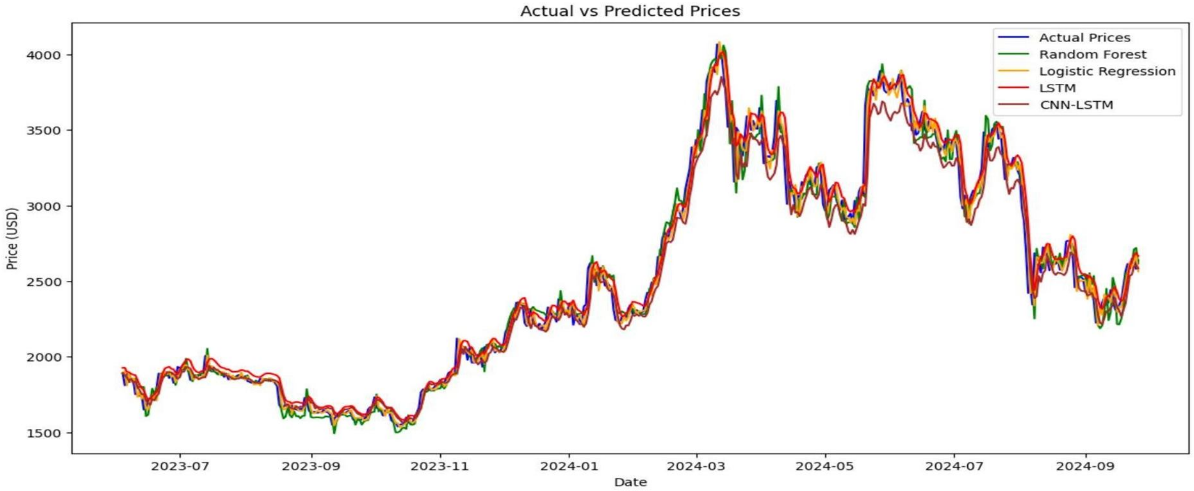

Figure 8: Actual vs. Predicted Prices for All Models

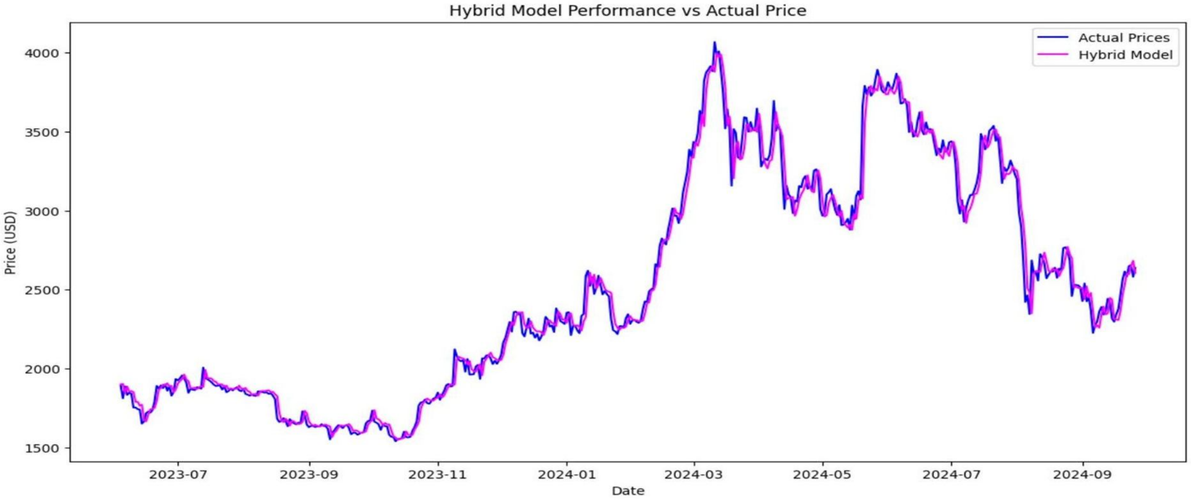

Figure 9: Hybrid Model Performance vs. Actual Price

Figure 8 compares the actual Ethereum prices with predictions from all four models. Logistic Re- gression exhibited the closest alignment with actual values, followed by Random Forest.

Figure 9 highlights the performance of the hybrid model compared to actual prices. While the hybrid model captured intricate patterns and exhibited higher errors compared to Logistic Regression, it offers several advantages:

1) Robustness Across Scenarios: By leveraging multiple models, the hybrid approach mitigates the risk of over-reliance on a single model's limitations, ensuring better adaptability to varied data patterns.
2) Error Compensation: The weighted averaging mechanism allows stronger models to compensate for weaker ones, providing a balanced prediction that reduces the impact of outliers or specific model weaknesses.
3) Complex Pattern Recognition: Incorporating advanced models like CNN-LSTM enables the hybrid model to capture both spatial and temporal patterns, which single models like Logistic Regression may overlook.
4) Future-Proofing: As market or time-series patterns evolve, the hybrid model's diverse components make it more resilient and likely to generalize better to unseen data.
5) Dynamic Environments: In real-world applications where data is often non-linear and volatile, the hybrid model's ability to blend the strengths of multiple paradigms ensures consistent performance.

These benefits make the hybrid model a promising choice for long-term deployment, despite its slightly higher error in the current evaluation.

V. DISCUSSION

A. Insights

1) Logistic Regression consistently outperformed more complex models, suggesting that simpler mod- els can be highly effective when engineered features are carefully chosen.

2) Random Forest demonstrated robustness but struggled with temporal dependencies compared to LSTMs.

3) LSTM and CNN-LSTM models highlight the challenges of training deep learning models, such as overfitting and high computational requirements.

B. Challenges

1) Data Quality: The accuracy of sentiment analysis directly impacts model performance. Future work should focus on advanced NLP techniques to improve sentiment extraction.

2) Computational Complexity: Deep learning models require significant resources, making them less accessible for real-time predictions.

C. Practical Implications

High trading volumes in Ethereum signify market opportunities for predictive modeling. For example, if a model improves prediction accuracy by just 1%, traders handling $10 million daily volumes could see profit increases of $100, 000, assuming efficient execution strategies.

Additionally, integrating predictive models into trading algorithms can enhance decision-making and reduce human error. Studies in traditional markets suggest that algorithmic trading improves execution speeds and reduces costs, a finding increasingly relevant to cryptocurrencies.

VI. CONCLUSION AND FUTURE WORK

This study demonstrates the effectiveness of machine learning models in predicting Ethereum’s price, with Logistic Regression emerging as the top performer. Future research directions include:

1) Exploring alternative features: On-chain metrics, such as gas fees and transaction volumes, can provide additional predictive power.

2) Improving deep learning models: Techniques like attention mechanisms and transfer learning could enhance performance.

3) Real-time system deployment: Building end-to-end pipelines for real-time cryptocurrency price prediction.

The integration of advanced analytics and machine learning into cryptocurrency trading has the potential to revolutionize decision-making processes, offering traders and investors a competitive edge.

REFERENCES

[1] CoinGecko, "Ethereum Historical Data." [Online]. Available: https://www.coingecko.com/en/coins/ethereum/historical_data. [Accessed: Jan. 10, 2025].

[2] Financial Modeling Prep, "Understanding WACC and CAPM in DCF Valuations: Identifying Dis- count Rates." [Online]. Available: https://site.financialmodelingprep.com/discounted-cash-flow-blogs/Understanding-WACC-and-C APM- in-DCF-Valuations-Identifying-Discount-Rates. [Accessed: Jan. 10, 2025].

[3] Wall Street Mojo, "DCF: Discounted Cash Flow." [Online]. Available: https://www.wallstreetmojo.com/dcf-discounted-cash-flow/. [Accessed: Jan. 10, 2025].

[4] CoinGecko, "Blockchain Trading Volume Market Share." [Online]. Available: https://www.coingecko.com/research/publications/blockchain-trading-volume-market-share. [Accessed: Jan. 10, 2025].

[5] M. Sipper, "Combining Deep Learning with Good Old-Fashioned Machine Learning," arXiv preprint, arXiv:2207.03757, 2022. [Online]. Available: https://arxiv.org/abs/2207.03757

[6] L. Breiman, "Random Forests," Machine Learning, vol. 45, no. 1, pp. 5-32, 2001. DOI: 10.1023/A:1010933404324. Available: https://link.springer.com/journal/10994/volumes-and-issues/45-1

[7] P. Peng, K. Lee, and G. Ingersoll, "An Introduction to Logistic Regression Analysis and Reporting," The Journal of Educational Research, vol. 96, no. 1, pp. 3-14, 2002. DOI: 10.1080/00220670209598786.

[8] D. Hopp, "Economic Nowcasting with Long Short-Term Memory Artificial Neural Networks (LSTM)," arXiv preprint, arXiv:2106.08901, 2021. [Online]. Available: https://arxiv.org/abs/2106.08901.

[9] X. Jin et al., "Prediction for Time Series with CNN and LSTM," in Proc. Int. Conf. Modelling, Identification and Control (ICMIC), vol. 582, pp. 631-641, 2019. DOI: 10.1007/978-981-15-0474-7_59.

[10] C. Hutto and E. Gilbert, "VADER: A Parsimonious Rule-Based Model for Sentiment Analysis," in Proc. Int. AAAI Conf. Web and Social Media, 2014.

[11] CoinMarketCap, "Ethereum Historical Data." [Online]. Available: https://coinmarketcap.com. [Ac- cessed: Jan. 10, 2025].

[12] Yahoo Finance. [Online]. Available: https://finance.yahoo.com. [Accessed: Jan. 10, 2025].

[13] CryptoCompare. [Online]. Available: https://www.cryptocompare.com. [Accessed: Jan. 10, 2025].

[14] GlobalData, "Ethereum's Market Capitalization History." [Online]. Available: https://www.globaldata.com/data-insights/financial-services/ethereums-market-capitalization- history/. [Accessed: Jan. 10, 2025].

[15] Investopedia, "Trading Strategy with Bollinger Bands and RSI." [Online]. Available: https://www.investopedia.com/ask/answers/121014/how-do-i-create-trading-strategy-bollinger- b ands-and-relative-strength-indicator-rsi.asp. [Accessed: Jan. 10, 2025].

[16] Bybit Learn, "What is Crypto Volatility Trading?" [Online]. Available: https://learn.bybit.com/en/crypto/what-is-crypto-volatility-trading/. [Accessed: Jan. 10, 2025].

[17] Economics Design, "Understanding Volatility in Crypto: A Comprehensive Guide." [Online]. Available: https://economicsdesign.com/featured-insights/defi/understanding-volatility-in-crypto-a-compreh ensive-guide/.

[18] Quantified Strategies, "What Percentage of Trading is Algorithmic?" [Online]. Available: https://www.quantifiedstrategies.com/what-percentage-of-trading-is-algorithmic/. [Accessed: Jan. 10, 2025].

[19] W. Liu, H. Wang, J. Zhang, and J. Huang, "DDPNAS: Efficient Neural Architecture Search via Dy- namic Distribution Pruning," International Journal of Computer Vision, vol. 131, no. 4, pp. 1034-1053, 2023. DOI: 10.1007/s11263-023-01753-6.

[20] Statista, "Cryptocurrency Sentiment Analysis Trends." [Online]. Available: https://www.statista.com. [Accessed: Jan. 10, 2025].

[21] CoinGecko, "2021 Year-End Report." [Online]. Available: https://assets.coingecko.com/reports/2021-Year-End-Report/CoinGecko-2021-Report.pdf. [Ac- cessed: Jan. 10, 2025].