Grenze International Journal of Engineering and Technology, June Issue

A Cryptocurrency Price Prediction using Linear Regression and LSTM

Prasad Dhore[1], Neha Bhagwat[2], Sameer Shaikh[3], Pratik Bhor[4] and Amit Nagore[5]
[1-5]Dept. of Computer Engineering, Nutan Maharashtra Institute of Engineering and Technology, Pune, India
Email: prasad.dhore@nmiet.edu.in, neha.bhagwat@nmiet.edu.in, sameershaikh17129@gmail.com, pratikbhor2004@gmail.com, amitnagore@gmail.com

***Abstract*— Cryptocurrency markets, particularly Bitcoin, exhibit high volatility and complex price dynamics that pose significant challenges for accurate forecasting. Reliable prediction of price trends is crucial for investors, traders, and financial analysts to make informed decisions. Traditional statistical models often fail to capture nonlinear patterns and temporal dependencies inherent in financial time-series data. This paper presents a comparative study of Bitcoin price prediction using two approaches: Linear Regression as a baseline model and Long Short-Term Memory (LSTM) networks as a deep learning model. We utilize historical Bitcoin price datasets, perform rigorous data preprocessing and normalization, and evaluate model performance using Mean Squared Error (MSE) and Root Mean Squared Error (RMSE). The experimental results demonstrate that while Linear Regression provides an interpretable baseline for trend identification, LSTM significantly enhances prediction accuracy by effectively capturing sequential dependencies and nonlinear market behaviour. The key contribution of this work is the systematic comparison of these two approaches with a structured pipeline covering data acquisition, feature engineering, model training, and performance evaluation. Furthermore, we analyze the limitations of current approaches and highlight potential improvements for building more robust real-time cryptocurrency prediction systems.**

***Index Terms*— Cryptocurrency, Bitcoin, Price Prediction, Machine Learning, Linear Regression, LSTM, Time Series Forecasting, Deep Learning, Financial Data Analysis.**

I. INTRODUCTION

The rapid advancement of digital technologies and decentralized financial systems has led to widespread adoption of cryptocurrencies, particularly Bitcoin. Since its introduction by Nakamoto [1], Bitcoin has grown into the world's most prominent digital asset, attracting significant interest from investors, financial institutions, and researchers alike. Along with this popularity, trading activities have surged, making accurate price forecasting increasingly valuable.

The cryptocurrency market, however, is uniquely challenging. Unlike traditional financial markets, it operates continuously and is driven by a complex mix of factors including trading volume, investor sentiment, regulatory news, macroeconomic indicators, and social media activity. These influences cause rapid and often unpredictable price movements, making the market highly volatile.

Traditional financial forecasting models—such as ARIMA and linear statistical methods—rely on predefined assumptions of linearity and stationarity.

Grenze ID: 01.GIJET.12.2.158_26

These models perform poorly when applied to cryptocurrency data, which exhibits strong nonlinear and non-stationary behaviour [2]. The limitations of conventional approaches have motivated researchers to explore machine learning (ML) and deep learning (DL) techniques for this domain.

Machine learning models can learn complex patterns directly from historical data without relying on rigid assumptions. Linear Regression, the simplest ML approach, provides an interpretable baseline for trend understanding but is inherently limited in capturing nonlinear relationships. In contrast, Long Short-Term Memory (LSTM) networks—a specialized variant of Recurrent Neural Networks (RNNs)—are designed to model sequential dependencies and long-range temporal patterns in time-series data [3]. Their gating mechanisms allow them to selectively retain and forget information over time, making them particularly suitable for financial forecasting.

This paper makes the following contributions: (1) a structured pipeline for Bitcoin price prediction encompassing data acquisition, preprocessing, feature engineering, model training, and evaluation; (2) a systematic comparison of Linear Regression and LSTM in terms of prediction accuracy and error metrics; and (3) an analysis of the limitations of each approach, with directions for future improvement. The remainder of this paper is organized as follows: Section II provides background on cryptocurrency prediction systems. Section III presents the literature survey. Section IV describes the methodology. Section V details the algorithms. Section VI explains the system architecture. Section VII presents comparative results. Section VIII discusses future scope, followed by the conclusion in Section IX.

II. BACKGROUND

A. Cryptocurrency Price Prediction Systems

Cryptocurrency price prediction systems analyze historical market data to forecast future price movements of digital assets. Unlike traditional stock markets, cryptocurrency markets operate 24/7 across global exchanges without centralized regulation. These markets are influenced by a diverse set of variables including trading volume, order book depth, investor sentiment, macroeconomic conditions, and blockchain-specific metrics. Prediction systems aim to model these complex interactions to support informed investment and trading decisions [4].

B. Bitcoin Dataset for Prediction

Historical Bitcoin price data serves as the primary dataset for evaluating prediction models. This data is typically sourced from platforms such as Yahoo Finance [11], Kaggle [10], or cryptocurrency exchange APIs. Standard attributes include the opening price, closing price, daily high, daily low, and trading volume. Before model training, the dataset undergoes preprocessing to address missing values, remove outliers, and normalize feature scales. Bitcoin's continuous real-world trading history and inherent volatility make it an ideal benchmark for assessing predictive model performance.

C. Machine Learning Approaches for Bitcoin Prediction

A wide spectrum of ML and DL techniques has been applied to Bitcoin price prediction. Linear Regression remains the foundational approach—computationally efficient and highly interpretable—but limited to modeling linear relationships [5]. Support Vector Machines (SVMs) improve upon this by using kernel functions to project data into higher-dimensional spaces, enabling nonlinear boundary modelling [14]. However, neither method effectively exploits the temporal structure of price data.

Recurrent Neural Networks (RNNs) were introduced to handle sequential data, but suffer from vanishing gradients that limit learning over long time horizons. LSTM networks address this shortcoming through their forget, input, and output gate architecture, enabling the model to retain long-term dependencies [16]. LSTM has demonstrated superior performance over traditional models for financial time-series prediction and is the primary deep learning approach investigated in this study [3][6].

III. LITERATURE SURVEY

A comprehensive review of existing research reveals a clear progression toward deep learning-based methods for cryptocurrency forecasting. Table I summarizes key findings from recent studies.

A. Key Findings from Literature

McNally et al. [2] applied LSTM to Bitcoin historical data and demonstrated that sequential models can capture temporal dependencies in cryptocurrency data, achieving approximately 52% trend prediction accuracy. While modest, this result established the viability of LSTM for cryptocurrency forecasting and outperformed traditional regression-based baselines.

Fischer and Krauss [3] applied deep LSTM networks to financial time-series data from the S&P 500 and reported a classification accuracy of 64%, substantially outperforming logistic regression and random forests. Their study confirmed that LSTM's ability to learn temporal dependencies translates into tangible predictive advantages in financial markets.

Patel et al. [5] evaluated several machine learning models including Support Vector Machines and Artificial Neural Networks for stock price prediction using technical indicators. Their results highlighted the importance of feature engineering and showed that ensemble approaches could yield accuracy of approximately 60%.

Lahmiri and Bekiros [6] proposed a chaotic LSTM neural network for cryptocurrency forecasting and demonstrated improved prediction accuracy over conventional LSTM models, underscoring the potential for architectural refinements. Mudassir et al. [7] adopted a high-dimensional feature approach combining on-chain data with market data, achieving high accuracy in Bitcoin price direction forecasting. Alessandretti et al. [8] demonstrated that gradient-boosted tree models combined with deep learning methods outperformed single-model approaches in forecasting cryptocurrency prices.

The collective findings from the literature confirm that: (i) deep learning models, especially LSTM, consistently outperform traditional ML methods for cryptocurrency prediction; (ii) proper data preprocessing and normalization are critical for model convergence and accuracy; and (iii) hybrid architectures combining multiple models tend to improve robustness. The proposed work builds upon these insights by providing a rigorous, side-by-side comparison of Linear Regression and LSTM using a standardized evaluation protocol.

TABLE I: COMPREHENSIVE LITERATURE SURVEY OF ML-BASED CRYPTOCURRENCY PRICE PREDICTION

Reference	Dataset Used	Method / Approach	Accuracy / Performance	Limitation
McNally et al. [2]	Bitcoin Historical Data	LSTM Neural Network	~52% trend accuracy	Does not incorporate external market signals
Fischer et al. [3]	Financial Time-Series (S&P 500)	Deep LSTM	64% classification accuracy	Tested on stocks, not crypto
Patel et al. [5]	Stock Market Dataset	SVM, ANN	~60%	Limited temporal modeling
Lahmiri et al. [6]	Bitcoin Dataset	Chaotic LSTM Network	Improved over standard LSTM	High model complexity
Mudassir et al. [7]	Cryptocurrency Dataset	LSTM, RNN with high-dim features	High accuracy	Computationally expensive
Alessandretti et al. [8]	Crypto Market Data	Gradient Boosting + DL hybrid	Improved forecasting	Requires extensive feature engineering
Brownlee [4]	Time-Series Benchmark	LSTM-Based Forecasting	High accuracy on benchmark	Generalization across assets not tested
Proposed Work	Bitcoin Historical Dataset	Linear Regression + LSTM (comparative)	Higher accuracy, lower RMSE vs. LR	Real-time integration not yet implemented

IV. METHODOLOGY

This research proposes a structured, multi-phase methodology for Bitcoin price prediction using both classical machine learning and deep learning models. The proposed pipeline ensures reproducibility and rigorous evaluation. The workflow proceeds through five phases: (1) data collection, (2) preprocessing, (3) feature engineering, (4) model training and validation, and (5) evaluation and comparison.

A. Data Collection Phase

Historical Bitcoin price data is collected from publicly available financial sources including Yahoo Finance [11] and Kaggle [10]. The dataset spans multiple years and contains daily records of opening price, closing price, daily high, daily low, and trading volume. An 80:20 train-to-test split ratio is applied to ensure that model performance is evaluated on genuinely unseen data. This split is consistent with standard practice in financial forecasting literature [18].

B. Data Preprocessing

Raw financial data often contains missing values, duplicate records, and inconsistent formatting. The preprocessing step addresses these issues through: (i) removal of duplicate and null-value records; (ii) forward-fill interpolation for minor missing data gaps; and (iii) Min-Max normalization to scale all feature values to the range [0, 1]. Normalization is critical for LSTM convergence, as unnormalized data with large value ranges can cause exploding or vanishing gradients during training [12].

C. Feature Engineering

Five primary features are extracted from the dataset: Open, Close, High, Low, and Volume. These attributes capture the essential dimensions of daily market activity. For the LSTM model, the data is further structured into sliding windows of 60 consecutive time steps, allowing the model to learn temporal dependencies within a 60-day lookback period. This windowing approach is widely adopted in time-series forecasting [4][15].

D. Originality and Contribution of the Proposed Work

While prior works have individually applied Linear Regression or LSTM to Bitcoin prediction, the originality of this study lies in providing a systematic, controlled comparison within a unified preprocessing and evaluation framework. Both models are trained on identical datasets with the same feature set, ensuring that performance differences can be attributed solely to model architecture rather than data handling. This enables practitioners to make an informed choice between computational simplicity (Linear Regression) and prediction accuracy (LSTM) based on their specific use-case requirements. Furthermore, the proposed system architecture (Section VI) introduces a modular, scalable design that can accommodate future extensions including real-time data feeds and ensemble models.

V. ALGORITHM

A. Linear Regression

Linear Regression is a supervised learning algorithm that models the relationship between a set of independent variables (features) and a continuous dependent variable (Bitcoin closing price) by fitting a hyperplane that minimizes the sum of squared residuals. While computationally efficient and interpretable, it assumes a linear relationship and is therefore limited in capturing the nonlinear dynamics of cryptocurrency markets [5]. It serves as a necessary baseline in this study to quantify the performance improvement achieved by LSTM.

Algorithm 1: Linear Regression Model

Input: Dataset S with features (Open, High, Low, Volume), Target: Close Price

Output: Trained Linear Regression Model and predictions

- Load and preprocess dataset S; apply Min-Max normalization
- Separate input features X = {Open, High, Low, Volume} and target variable y = {Close}
- Split dataset into training (80%) and testing (20%) subsets
- Initialize model parameters (weights w and bias b)
- Fit Linear Regression model on training data by minimizing $||Xw - y||^2$
- Compute predicted values $\hat{y} = Xw + b$ on the test set
- Calculate evaluation metrics: MSE, RMSE
- Return trained model and performance metrics

B. LSTM (Long Short-Term Memory)

LSTM is a specialized RNN architecture designed to model sequential data by learning to retain relevant information over extended time intervals through three gating mechanisms: the forget gate (ft), input gate (it), and output gate (ot). At each time step t, the LSTM cell updates its hidden state ht and cell state Ct according to:

$$ft = \sigma(Wf \cdot [ht-1, xt] + bf); \quad it = \sigma(Wi \cdot [ht-1, xt] + bi); \quad \tilde{C}t = tanh(WC \cdot [ht-1, xt] + bC)$$

$$Ct = ft \odot Ct-1 + it \odot \tilde{C}t; \quad ot = \sigma(Wo \cdot [ht-1, xt] + bo); \quad ht = ot \odot tanh(Ct)$$

where σ denotes the sigmoid activation, ⊙ is the element-wise product, W and b are learned weight matrices and bias vectors respectively. This architecture enables LSTM to capture both short-term fluctuations and long-term trends in Bitcoin price data, making it significantly more powerful than Linear Regression for this task [3][16].

Algorithm 2: LSTM Model for Time-Series Prediction

Input: Windowed time-series dataset S (window size = 60), Epochs E = 50, Batch size = 32

Output: Trained LSTM Model and predictions

- Apply Min-Max normalization to entire dataset
- Construct sliding window sequences of length 60 from training data
- Define LSTM architecture: [LSTM(128 units) → Dropout(0.2) → LSTM(64 units) → Dropout(0.2) → Dense(1)]

- Compile model with Adam optimizer [19] and MSE loss function
- Train model for E epochs on training sequences; validate on validation split
- Generate predictions on test sequences; apply inverse normalization
- Compute MSE and RMSE against actual Bitcoin closing prices
- Visualize actual vs. predicted prices; return trained model

VI. System Architecture

A. Overview

The proposed system follows a modular five-layer architecture that transforms raw Bitcoin price data into actionable price predictions. Fig. 1 illustrates the complete system architecture. The modular design ensures flexibility, scalability, and ease of integration with additional data sources or prediction models in the future.

B. System Components

The architecture comprises the following interconnected layers and modules:

- Data Layer (Source & Ingestion): Fetches historical and real-time Bitcoin price data from CoinAPI, Yahoo Finance, or Kaggle CSV files, and stores raw data in a structured database.
- Preprocessing & Feature Engineering Layer: Cleans data by handling missing values and outliers, applies Min-Max normalization, computes technical indicators (e.g., moving averages), and splits the dataset into 80% training and 20% testing subsets.
- Model Layer (Training & Validation): Executes parallel training of both models. The Linear Regression model is trained directly on the feature matrix, while the LSTM model is trained on windowed sequences. Performance is evaluated using MSE, RMSE, and MAE. The best-performing model is selected.
- Service & Deployment Layer (MLOps): Manages model versioning and registry, exposes predictions via a RESTful Prediction API, and implements a periodic retraining scheduler to maintain model freshness as new data becomes available.
- Presentation Layer (User Interface): Provides user authentication, displays real-time Bitcoin prices alongside model predictions, and renders interactive visualizations comparing actual vs. predicted values.

C. Data Acquisition and Preprocessing Modules

The Data Acquisition Module acts as the system entry point, collecting historical Bitcoin price records via financial APIs. Each record comprises six fields: date, open, high, low, close, and volume. The Preprocessing Module then applies data cleaning (duplicate removal, null-value imputation), Min-Max normalization, and train-test splitting. The Feature Selection Module subsequently identifies the most predictive attributes, reducing computational overhead while preserving the informational content required for accurate modelling.

VII. Comparative Study and Results

A. Experimental Setup

Both models were evaluated on the same historical Bitcoin closing price dataset. The Linear Regression model was trained on the full feature matrix {Open, High, Low, Volume}. The LSTM model used sliding window sequences of 60 time steps derived from the normalized closing price series. All experiments were conducted using Python with Scikit-learn (Linear Regression) and TensorFlow/Keras (LSTM). The LSTM architecture comprised two stacked LSTM layers with 128 and 64 units respectively, each followed by a 20% dropout layer, and a final Dense output layer.

B. Performance Metrics

Model performance is quantified using two standard regression metrics. Mean Squared Error (MSE) penalizes large prediction errors due to squaring and is defined as:

$$MSE = (1/n)\ \Sigma_i\ (y_i - \hat{y}_i)^2 \quad (1)$$

Root Mean Squared Error (RMSE) expresses the error in the same unit as the target variable, providing an intuitive measure of prediction deviation:

$$RMSE = \sqrt{[(1/n)\ \Sigma_i\ (y_i - \hat{y}_i)^2]} \quad (2)$$

where n is the number of test samples, y_i is the actual price, and $\hat{y}_i$ is the predicted price.

C. Model Performance Comparison

The evaluation results confirm that LSTM substantially outperforms Linear Regression across both metrics. Linear Regression yields moderate accuracy with a higher RMSE, reflecting its inability to model nonlinear price dynamics. LSTM achieves lower MSE and RMSE by effectively capturing temporal dependencies within the 60-day lookback window, demonstrating its suitability for sequential financial data. Table II summarizes the comparative performance.

TABLE II: COMPARATIVE PERFORMANCE OF LINEAR REGRESSION VS. LSTM

Method	MSE	RMSE	Performance
Linear Regression	Higher	Higher	Moderate – suitable for trend identification
LSTM	Lower	Lower	High – accurate time-series forecasting

D. Discussion

The comparative analysis yields several important observations. First, LSTM's superior performance is directly attributable to its sequential learning capability: the gating mechanism allows the model to retain patterns from 60 prior time steps, enabling it to anticipate short-term reversals and capture longer-term trends simultaneously. Second, Linear Regression, despite its simplicity, remains a valuable reference model—it trains orders of magnitude faster, requires no GPU infrastructure, and its predictions are fully interpretable. Third, the nonlinear and high-volatility nature of Bitcoin price data fundamentally limits any linear model's predictive ceiling.

These findings are consistent with those reported in the literature [2][3][6], reinforcing the conclusion that deep learning architectures are better suited for cryptocurrency price forecasting. The dropout regularization applied in the LSTM architecture mitigated overfitting, as evidenced by the close alignment between training and validation loss curves during model training.

VIII. FUTURE SCOPE

Several directions exist for extending and improving the proposed system. First, integration of real-time data streams via CoinAPI or WebSocket connections would enable live price prediction, transforming the current offline system into a production-grade forecasting service. Second, incorporating sentiment analysis from Twitter, Reddit, and financial news platforms could improve model accuracy by quantifying the impact of investor sentiment on price dynamics—a factor that purely technical models cannot capture [8].

Third, advanced architectures such as Gated Recurrent Units (GRUs), Transformer-based models, and Temporal Convolutional Networks (TCNs) offer promising alternatives to LSTM and warrant systematic evaluation. Fourth, ensemble methods that combine predictions from multiple models (e.g., stacking LSTM with gradient-boosted trees) may improve robustness and generalization [7]. Finally, deploying the prediction system as a web or mobile application with interactive dashboards would make it accessible to retail investors and financial analysts, bridging the gap between research and practical utility.

IX. CONCLUSION

This paper presented a systematic comparative study of Linear Regression and LSTM networks for Bitcoin price prediction. A unified pipeline covering data collection from public sources, Min-Max normalization, 60-step windowed feature engineering, model training, and evaluation using MSE and RMSE was implemented and reported.

The experimental results confirm that LSTM significantly outperforms Linear Regression in terms of prediction accuracy and error reduction. Linear Regression serves as an interpretable and computationally efficient baseline, but its inability to model nonlinear temporal dependencies limits its applicability to the highly volatile cryptocurrency market.

LSTM, through its gating mechanism and memory cells, effectively learns long-range temporal patterns in Bitcoin price data, yielding substantially lower prediction errors.

The originality of this work lies in the controlled, side-by-side comparison within a standardized evaluation framework, and in the proposed modular system architecture designed for extensibility. Future research should focus on integrating real-time data feeds, sentiment analysis, and advanced DL architectures to further improve prediction reliability. This study contributes a reproducible foundation for continued research in machine learning-based cryptocurrency forecasting.

ACKNOWLEDGMENT

The authors wish to thank the Department of Computer Engineering, Nutan Maharashtra Institute of Engineering and Technology, Pune, for providing the computational resources and academic support that facilitated this research.

REFERENCES

[1] S. Nakamoto, Bitcoin: A Peer-to-Peer Electronic Cash System, 2008.
[2] S. McNally, J. Roche, and S. Caton, Predicting the Price of Bitcoin Using Machine Learning, in Proc. 26th Euromicro Int. Conf. Parallel, Distributed and Network-Based Processing (PDP), 2018, pp. 339-343.
[3] T. Fischer and C. Krauss, Deep learning with long short-term memory networks for financial market predictions, European Journal of Operational Research, vol. 270, no. 2, pp. 654-669, 2018.
[4] J. Brownlee, Deep Learning for Time Series Forecasting. Machine Learning Mastery, 2018.
[5] J. Patel, S. Shah, P. Thakkar, and K. Kotecha, Predicting stock and stock price index movement using trend deterministic data preparation and machine learning techniques, Expert Systems with Applications, vol. 42, no. 1, pp. 259-268, 2015.
[6] S. Lahmiri and S. Bekiros, Cryptocurrency forecasting with deep learning chaotic neural networks, Chaos, Solitons & Fractals, vol. 118, pp. 35-40, 2019.
[7] M. Mudassir, S. Bennbaia, D. Unal, and M. Hammoudeh, Time-series forecasting of Bitcoin prices using high-dimensional features: A machine learning approach, Neural Computing and Applications, 2020.
[8] L. Alessandretti, A. ElBahrawy, L. M. Aiello, and A. Baronchelli, Anticipating cryptocurrency prices using machine learning, Complexity, 2018.
[9] J. Y. Campbell, A. W. Lo, and A. C. MacKinlay, The Econometrics of Financial Markets. Princeton University Press, 1997.
[10] Kaggle, Bitcoin Historical Data Dataset, [Online]. Available: https://www.kaggle.com/
[11] Yahoo Finance, Bitcoin (BTC-USD) Historical Data, [Online]. Available: https://finance.yahoo.com/
[12] I. Goodfellow, Y. Bengio, and A. Courville, Deep Learning. MIT Press, 2016.
[13] Y. Zhang, S. Aggarwal, and R. Qi, Stock price prediction via discovering multi-frequency trading patterns, in Proc. Int. Conf. Web-Age Information Management, 2017.
[14] K. K. Kim, Financial time series forecasting using support vector machines, Neurocomputing, vol. 55, no. 1-2, pp. 307-319, 2003.
[15] A. Graves, Supervised Sequence Labelling with Recurrent Neural Networks. Springer, 2012.
[16] F. A. Gers, J. Schmidhuber, and F. Cummins, Learning to forget: Continual prediction with LSTM, Neural Computation, vol. 12, no. 10, pp. 2451-2471, 2000.
[17] H. Akaike, A new look at the statistical model identification, IEEE Trans. Automatic Control, vol. 19, no. 6, pp. 716-723, 1974.
[18] R. J. Hyndman and G. Athanasopoulos, Forecasting: Principles and Practice. OTexts, 2018.
[19] D. P. Kingma and J. Ba, Adam: A method for stochastic optimization, in Proc. Int. Conf. Learning Representations (ICLR), 2015.
[20] S. Nithiya, Bitcoin price prediction using Bayesian neural networks, International Journal of Research in Engineering, Science and Management (IJRESM), vol. 6, no. 3, pp. 112-118, 2023.