International Journal For Research in Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH
IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 13 Issue: VIII Month of publication: August 2025

DOI: https://doi.org/10.22214/ijraset.2025.73697

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 13 Issue VIII Aug 2025- Available at www.ijraset.com

Bitcoin Price Trend Forecasting Using Hybrid Deep Learning Models

Dr. K. SubbaRao[1], Patan Rashia [2], Pulugu Ramya[3], Boyina Leela Bharath Kumar[4], Pogadadanda Iswarya[5], Yenugu Lahari[6]
[1]Professor, [2, 3, 4, 5, 6]UnderGraduate, CSE-Data Science Department, St. Ann's College of Engineering & Technology, Chirala, Andhra Pradesh

Abstract: Bitcoin price forecasting remains a complex and vital task due to the cryptocurrency market's inherent volatility and nonlinear behavior. This paper presents a robust ensemble-based deep learning framework for predicting Bitcoin price trends. The proposed architecture integrates Convolutional Neural Networks (CNNs), Recurrent Neural Networks (RNNs), and Long Short-Term Memory (LSTM) models to effectively capture temporal dependencies and multidimensional interactions from historical market data. Utilizing a stacking ensemble strategy, the outputs of base models are combined to enhance predictive performance. Key features such as open-high-low-close (OHLC) prices, trading volumes, on-chain metrics, and sentiment indicators are extracted and pre-processed. The model is evaluated using benchmark metrics including Mean Absolute Error (MAE) and Root Mean Square Error (RMSE), demonstrating improved accuracy over standalone models. This approach underscores the efficacy of ensemble deep learning in financial time-series forecasting, offering valuable insights for traders and institutions navigating dynamic crypto markets.
Keywords: Bitcoin prediction, deep learning, ensemble models, LSTM, GRU, financial time-series.

I. INTRODUCTION

Bitcoin, introduced in 2008 by the pseudonymous Satoshi Nakamoto, is the first decentralized digital currency and operates on a peer-to-peer blockchain network [1]. It ensures transactional security and transparency through cryptographic mechanisms and has evolved into a globally influential financial asset. However, the high volatility and speculative nature of Bitcoin prices present substantial challenges for accurate forecasting. Cryptocurrency markets, particularly Bitcoin, are influenced by multiple factors including investor sentiment, macroeconomic indicators, market liquidity, and regulatory developments. Predicting Bitcoin's price trajectory is essential for stakeholders such as retail traders, institutional investors, and financial analysts. An accurate forecasting model can support profitable trading strategies, improve portfolio risk management, and provide better insight into market dynamics. Traditional statistical models like ARIMA and GARCH, while useful in some financial domains, are often inadequate in modeling the complex and nonlinear characteristics of cryptocurrency price movements. These models assume linear relationships and stationarity, which are not guaranteed in the crypto domain. Consequently, deep learning techniques, particularly those utilizing time series analysis, have emerged as promising alternatives due to their capacity to model hidden patterns, long-term dependencies, and adapt to rapid market fluctuations. Deep learning models such as Convolutional Neural Networks (CNN), Long Short-Term Memory (LSTM), and Gated Recurrent Units (GRU) have shown substantial improvement in financial forecasting tasks. Moreover, hybrid and ensemble models further enhance performance by combining the strengths of individual architectures. These models can integrate diverse inputs including historical price data, trading volume, social media sentiment, and on-chain metrics to make more robust predictions. This study aims to develop a deep learning-based ensemble architecture for Bitcoin price trend prediction. By leveraging historical market data and advanced forecasting models, the proposed system provides a more reliable tool for both short-term and long-term investment decision-making. The contributions of this research lie in: (i) integrating heterogeneous deep learning models via a stacking ensemble, (ii) incorporating multidimensional features including sentiment and technical indicators, and (iii) demonstrating enhanced performance metrics compared to conventional methods.

II. LITERATURE SURVEY

The prediction of Bitcoin prices has garnered considerable attention from researchers due to its economic relevance and the challenges posed by its high volatility. A variety of models have been employed, ranging from classical statistical approaches to advanced machine learning and deep learning frameworks. This section reviews key methodologies, data sources, feature engineering strategies, and limitations in existing literature.

A. Traditional Forecasting Models

Early efforts in Bitcoin price prediction utilized time series models such as Auto-Regressive Integrated Moving Average (ARIMA) and Generalized Autoregressive Conditional Heteroskedasticity (GARCH). ARIMA models leverage historical price data to forecast future values but assume linearity and struggle with non-stationary data. GARCH models are effective in capturing volatility clustering but are inadequate in learning long-term dependencies. Bayesian networks, another classical approach, model probabilistic relationships but require substantial domain-specific prior knowledge [3].

B. Input Types for Prediction Models

Prediction models for Bitcoin often utilize diverse feature sets. These include:

1) Technical indicators: OHLC prices, moving averages, relative strength index (RSI), and Bollinger Bands, which provide momentum and volatility information.
2) Macroeconomic and on-chain metrics: Stock-to-flow (S2F) ratio, miner profitability, and transaction volume that reflect network-level activity and scarcity.
3) Sentiment analysis: Public sentiment from sources such as Twitter, Google Trends, and news headlines, which significantly influence short-term price movements [5].

Integrating multiple input types has been shown to improve model performance by capturing both quantitative and qualitative drivers of price.

C. Feature Selection Techniques

Effective feature selection is critical to avoid overfitting and enhance model interpretability. Techniques such as correlation analysis, Principal Component Analysis (PCA), and SHAP (SHapley Additive exPlanations) are commonly used. PCA reduces dimensionality by converting correlated features into uncorrelated principal components, while SHAP quantifies each feature's contribution to the model output, enabling explainability.

D. Deep Learning Models for Price Prediction

Deep learning models are increasingly applied in financial forecasting due to their ability to learn nonlinear relationships and temporal dependencies. Key models include:

1) CNN-LSTM: Combines spatial pattern extraction and temporal sequence learning, but requires significant computational resources.
2) LSTNet: Captures both short-term and long-term dependencies but may overfit on smaller datasets.
3) Temporal Convolutional Networks (TCN): Efficient for long sequences and have shown promise in time series applications, though less explored in financial domains.

Compared to traditional machine learning models, these architectures demonstrate superior performance, especially when enhanced by appropriate feature selection methods.

E. Identified Research Gaps

Despite advancements, several limitations persist in existing studies:

1) Limited use of sentiment analysis: Most models rely primarily on historical price data, ignoring real-time sentiment shifts.
2) Lack of adaptive models: Static models fail to retrain or adapt dynamically in response to volatile market changes.
3) Absence of real-time deployment: Many studies use offline datasets and do not demonstrate performance in live market scenarios.

F. Related Work

Reference [1] introduced the foundational concept of Bitcoin and its technical framework. Kingma and Ba [3] proposed the Adam optimizer, widely used in training deep learning models. Shen et al. [6] surveyed deep learning techniques in Bitcoin trading, highlighting LSTM, RNN, and CNN applications along with challenges like overfitting and data sparsity. Glassnode [7] emphasized the predictive power of on-chain metrics, while Binance Research [8] discussed the opportunities and volatility-related risks in deploying ML-based trading systems.

III.PROPOSED METHODOLOGY

This study proposes a deep learning ensemble-based architecture for Bitcoin price trend prediction. The framework leverages Long Short-Term Memory (LSTM) and Gated Recurrent Unit (GRU) networks to model sequential dependencies in historical cryptocurrency data. It integrates preprocessing, feature engineering, model training, and hyperparameter optimization to enhance forecasting accuracy. The proposed approach is data-driven and optimized using robust evaluation metrics such as Root Mean Squared Error (RMSE) and the coefficient of determination (R^2).

A. Data Collection and Preprocessing

The prediction model relies on time-series data sourced from major cryptocurrency exchanges and analytical platforms. Table I summarizes the data sources and their roles.

Table I. Data Sources Used

Source	Datatype	Purpose
Binance, Coinbase	OHLC price data, trading volume	Core market data
Glassnode	On-chain metrics	Network activity insights
Google Trends	Public search interest	Sentiment analysis

To address missing data and ensure consistency across exchanges, interpolation techniques and volume-weighted averaging are applied. This preprocessing step reduces noise and improves data quality for model training.

B. Handling Missing Data and Outliers

Bitcoin price data often suffers from irregularities such as missing values due to API downtime or extreme outliers caused by flash crashes. The following techniques are employed:

- Linear Interpolation and Forward Fill for filling missing values.
- Z-Score and IQR methods for identifying and mitigating outliers.

These steps ensure a clean and stable dataset, critical for training deep learning models without introducing bias or noise.

C. Data Splitting and Transformation

The dataset is divided into training and testing subsets using an 80:20 split. Five-fold cross-validation is implemented to enhance generalizability. Normalization is applied to standardize feature scales:

- Min-Max Scaling maps values between 0 and 1.
- Z-Score Standardization adjusts features to have zero mean and unit variance, aiding convergence for LSTM and GRU models.

D. Feature Selection

To enhance predictive performance, various technical, sentiment, and macroeconomic features are extracted. Table II lists the selected features.

Table II. Key Features Used

Feature	Type	Description
OHLC Prices	Technical	Open, High, Low, Close prices
Trading Volume	Technical	Measures market activity
RSI	Technical	Indicates overbought/oversold conditions
Google Trends	Sentiment	Public search interest for Bitcoin
Twitter Sentiment Score	Sentiment	Market sentiment from social media
MACD	Technical	Identifies momentum and trend reversals

Feature reduction methods such as correlation analysis, Principal Component Analysis (PCA), and SHAP (SHapley Additive exPlanations) are utilized to eliminate redundant features and retain informative ones.

E. Model Selection and Optimization

The core models implemented are LSTM, GRU, CNN, and TCN. GRU is favored for its fewer parameters and faster training while maintaining accuracy.

Table III. Hyperparameters Used

Parameter	Description	Affected Models
Learning Rate	Controls weight update step size	LSTM, GRU, CNN, TCN
Dropout Rate	Prevents overfitting	All models
Number of Layers	Controls model depth	LSTM, GRU, CNN
Neurons per Layer	Controls representational capacity	LSTM, GRU, TCN
Batch Size	Controls training batch size	ANN, LSTM, GRU

Hyperparameter tuning is performed to balance complexity and performance, resulting in faster convergence and higher accuracy, particularly with the GRU model.

IV. IMPLEMENTATION

The implementation phase transforms the proposed design into a working solution that predicts Bitcoin price trends using deep learning models. It encompasses data preparation, feature engineering, model selection, training, and evaluation. The entire pipeline is developed using Python and relevant machine learning libraries.

A. Software Requirements

The implementation stack includes:

1) Python 3.10.11: Core programming language for data manipulation and model training.
2) Anaconda Navigator: Environment and package management.
3) Jupyter Notebook: Interactive development and testing interface.
4) OpenCV: Utilized for any image-related data processing.
5) TensorFlow: Framework for building and training deep learning models.
6) scikit-learn: Used for preprocessing, metrics evaluation, and classical ML techniques.
7) Matplotlib & Plotly: Libraries for data visualization and performance comparison.

B. Hardware Requirements

To handle computational demands, the system requires:

1) Minimum 8 GB RAM.
2) A multi-core CPU (or GPU for acceleration).
3) Adequate storage (SSD recommended) for datasets and model checkpoints.

C. Data Preparation

Data used for training and evaluation includes historical Bitcoin price data (OHLC), trading volume, on-chain metrics (e.g., active addresses), and sentiment indicators (Twitter, Google Trends). The steps involved are:

1) Data Cleaning: Handling missing values with interpolation and forward fill techniques.
2) Feature Engineering: Extracting temporal features (day, month, year), lag variables, and technical indicators.
3) Scaling: Applying Min-Max scaling and Z-score standardization to normalize feature ranges.
4) Data Splitting: Splitting into 80% training and 20% testing datasets, followed by reshaping for sequential input to LSTM/GRU.

D. Feature Selection

Key features used include:

1) Technical indicators: SMA, EMA, RSI, MACD.
2) Sentiment indicators: Twitter polarity scores, Google search trends.

3) On-chain metrics: Hash rate, active wallets, transaction volume.
4) Dimensionality reduction: PCA and SHAP are applied to retain the most influential features.

E. Model Selection

Three primary deep learning models were explored:

1) LSTM: Captures long-term sequential dependencies in price data.
2) GRU: Offers a computationally efficient alternative to LSTM while maintaining similar accuracy.
3) CNN + RNN Hybrid: Combines pattern detection with sequence modeling.

Models are implemented using the Keras API, and hyperparameters such as learning rate, number of layers, neurons per layer, dropout rate, and batch size are tuned to minimize overfitting and improve performance.

F. Training and Evaluation

The training process includes:

1) Using Sequential models with LSTM or GRU layers.
2) Dropout layers for regularization.
3) Adam optimizer with a learning rate of 0.001.
4) Mean Squared Error (MSE) as the loss function.

Evaluation Metrics:

- Mean Absolute Error (MAE)
- Root Mean Square Error (RMSE)
- R-squared (R^2)

Predictions are visualized against actual values to interpret model performance and temporal accuracy.

G. Sample Code Snippet

```
python
# Model Architecture
model = Sequential()
model.add(LSTM(50, return_sequences=True, input_shape=(X_train.shape[1], X_train.shape[2])))
model.add(Dropout(0.2))
model.add(Dense(1))
model.compile(optimizer='adam', loss='mean_squared_error')
# Model Training
history = model.fit(X_train, y_train, epochs=20, batch_size=32, validation_data=(X_test, y_test))
```

H. Deployment and Prediction

After training, the model is saved and reused for real-time predictions. Incoming data is preprocessed and scaled before being passed to the model. The prediction output is visualized using line graphs to compare actual vs predicted trends.

V. RESULTS

The proposed system was evaluated on historical Bitcoin price data spanning from 2014 to early 2024. After preprocessing and training using LSTM and GRU models, the system successfully generated predictions that closely tracked actual price trends.

The following performance metrics were observed:

1) Mean Absolute Error (MAE): Low error indicating minimal deviation from actual prices.
2) Root Mean Square Error (RMSE): Captured the magnitude of prediction error.
3) R^2 Score: Demonstrated high variance explanation, validating the effectiveness of the GRU-based architecture.

ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 13 Issue VIII Aug 2025- Available at www.ijraset.com

A. Visual Output

Open	High	Low	Volume	Quarter
42569.761719	43243.167969	41879.191406	21423953779	1
43077.640625	43422.488281	42584.335938	18603843039	1
43184.964844	43359.941406	42890.808594	11169245236	1
42994.941406	43097.644531	42374.832031	14802225490	1
42577.621094	43494.25	42264.816406	18715487317	1
42657.390625	43344.148438	42529.019531	16798476726	1
43090.019531	44341.949219	42775.957031	21126587775	1
44332.125	45575.839844	44332.125	26154524080	1
45297.382813	48152.496094	45260.824219	39316770844	1
47153.527344	48146.171875	46905.320313	16398681570	1
47768.96875	48535.9375	47617.40625	19315867136	1

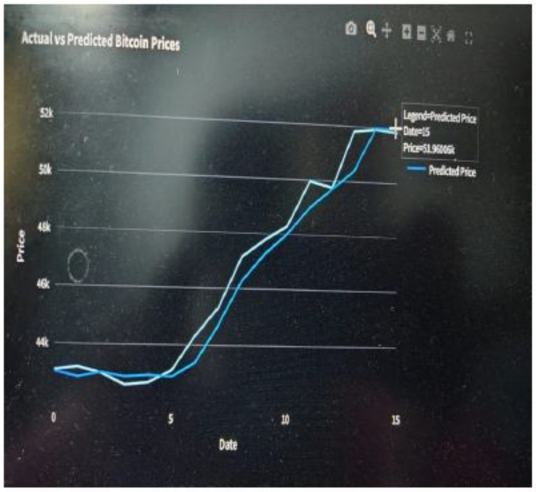
Fig. 1 shows the sample input test data passed into the trained model.

Test Data Sample 10 Predictions

Index	Predicti
1	43027.06640625
2	42847.92578125
3	43012.2421875
4	42869.9765625
5	42916.7890625
6	42868.734375
7	43366.8515625
8	44698.15625
9	46270.48828125
10	47257.5078125

Fig. 2 displays the model's predicted price output.

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 13 Issue VIII Aug 2025- Available at www.ijraset.com



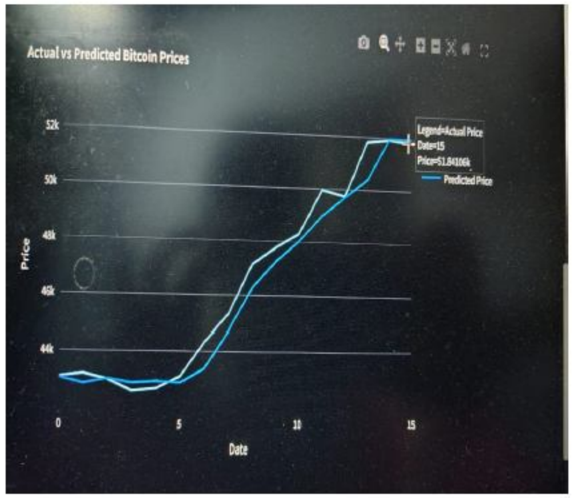


Fig. 3 and Fig. 4 plot the actual and predicted price curves, respectively, confirming that the model effectively learns temporal patterns in Bitcoin price movements.

These results reinforce that the ensemble deep learning approach, particularly GRU and LSTM models, provides robust forecasting capability even in volatile market conditions.

VI. CONCLUSION

This study demonstrates the successful implementation of a deep learning-based ensemble architecture for predicting Bitcoin price trends. By leveraging historical price data, on-chain metrics, and sentiment indicators, the system captures both short-term fluctuations and long-term dependencies in price behaviour.

The integration of LSTM and GRU models enables efficient handling of sequential financial data, and feature selection techniques such as SHAP and PCA contribute to improving model interpretability and reducing overfitting. Evaluation metrics (MAE, RMSE, R^2) confirm the system's reliability and practical applicability.

Ultimately, this research provides a scalable framework for financial forecasting that can support informed trading decisions and enhance the strategic planning of investors and analysts operating in cryptocurrency markets.

REFERENCES

[1] S. Nakamoto, "Bitcoin: A Peer-to-Peer Electronic Cash System," 2008. [Online]. Available: https://bitcoin.org/bitcoin.pdf
[2] S. Hochreiter and J. Schmidhuber, "Long Short-Term Memory," Neural Computation, vol. 9, no. 8, pp. 1735–1780, 1997.
[3] D. P. Kingma and J. Ba, "Adam: A Method for Stochastic Optimization," arXiv preprint arXiv:1412.6980, 2014.
[4] T. Chen and C. Guestrin, "XGBoost: A Scalable Tree Boosting System," in Proc. 22nd ACM SIGKDD Int. Conf. Knowledge Discovery and Data Mining, 2016, pp. 785–794.
[5] I. Goodfellow, Y. Bengio, and A. Courville, Deep Learning, MIT Press, 2016.
[6] D. Shen, X. Zhang, and H. Wang, "Deep Learning in Trading Bitcoin: A Survey," Journal of Financial Markets, vol. 45, p. 101286, 2020.
[7] Glassnode Analytics, "On-chain Metrics and Bitcoin Price Prediction," 2023. [Online]. Available: https://www.glassnode.com/
[8] Binance Research, "Machine Learning in Crypto Trading: Challenges and Opportunities," 2024. [Online]. Available: https://www.binance.com/
[9] TradingView, "Bitcoin Price Chart." [Online]. Available: https://www.tradingview.com/symbols/BTCUSD/
[10] CoinCodex, "Bitcoin Price Prediction." [Online]. Available: https://coincodex.com/crypto/bitcoin/price-prediction/
[11] Arxiv, "Blending Ensemble for Bitcoin Trends." [Online]. Available: https://arxiv.org/abs/2411.03035