Neural Processing Letters (2025) 57:77
https://doi.org/10.1007/s11063-025-11787-1

Detecting Bitcoin Sentiment: Leveraging Language Model Applications in Sentiment Analysis for Bitcoin Price Prediction

Hae Sun Jung[1] · **Haein Lee**[2] · **Jang Hyun Kim**[3]

Accepted: 12 July 2025 / Published online: 19 August 2025

Abstract

As Bitcoin continues to establish itself as a global asset and discussions around relevant regulations become more active, there is an increasing demand for a comprehensive price prediction framework. To address this necessity, this study aims to enhance the accuracy of Bitcoin price predictions by integrating sentiment information with technical indicators, on-chain data, and cryptocurrency price data. Recognizing Bitcoin's sensitivity to market sentiment, the proposed framework incorporates sentiment features derived from both lexicon-based methods and large language models. As unsupervised sentiment tools can introduce label noise particularly in domain-specific or ambiguous financial contexts, this study combines the outputs of multiple sentiment models at the feature level to construct a more stable representation. This design improves the robustness of downstream regression performance and distinguishes the framework from previous hybrid models that relied on a single sentiment source without component-wise evaluation. Experimental results using a dataset spanning 2700 days showed that the long short-term memory (LSTM) model with a 3-day window achieves the best performance with mean absolute percentage error (MAPE) of 3.93% and R-squared value of 0.99106. Feature importance analysis further demonstrates sentiment index as the most impactful feature, as excluding it resulted in the largest decline in predictive accuracy. Additionally, the model's performance was evaluated under four major volatility periods, revealing MAPE values ranging from 1.49 to 4.03%, highlighting the framework's practical capability in rapidly adapting to sudden market shifts. In summary, integrating sentiment information attained from multiple language models significantly enhanced prediction accuracy compared to single source approaches. These findings

Hae Sun Jung and Haein Lee have contributed equally to this work.

✉ Jang Hyun Kim
alohakim@skku.edu

[1] Advanced Institute of Convergence Technology, Seoul National University, Gyeonggi-do, Suwon-si 16229, Korea

[2] Department of Applied Artificial Intelligence/Department of Human-Artificial Intelligence Interaction, Sungkyunkwan University, Seoul 03063, Korea

[3] Department of Interaction Science/Department of Human-Artificial Intelligence Interaction, Sungkyunkwan University, Seoul 03063, Korea

highlight the framework's practical value for sentiment-informed investment strategies and risk alerts, with a modular design that enables flexible adaptation and potential integration into automated trading systems.

Keywords Large language models · Sentiment analysis · Time series analysis · Bitcoin · Price prediction

1 Introduction

The approval of exchange-traded funds (ETFs) has strengthened Bitcoin's recognition as an asset class as it becomes increasingly integrated into the global financial ecosystem [1]. ETF approval marks a significant turning point, making Bitcoin more accessible to institutional investors and facilitating its incorporation into traditional financial markets. This development not only underscores Bitcoin's growing role and investment value in the market but also highlights its gradual integration into established economic systems. Bitcoin's integration introduces innovative technologies and new investment opportunities to traditional financial systems, thereby reshaping the structure and operations of financial markets.

However, Bitcoin's extreme volatility remains a critical concern [2]. The high price fluctuation not only complicates investment strategies but also poses risks to financial stability. Accurate price prediction is essential to mitigate these risks, prevent speculative bubbles, and inform sound regulatory decisions [3].

While traditional price prediction approaches rely heavily on historical price and volume data, growing evidence suggests that Bitcoin is uniquely sensitive to investor sentiment, more so than many conventional financial assets [4–6]. By extracting psychological and emotional signals from unstructured textual data such as news articles and social media, sentiment analysis offers valuable insights for capturing the non-quantitative drivers of price movements [7, 8].

This study seeks to enhance the accuracy and robustness of Bitcoin price prediction by integrating sentiment analysis with technical indicators, on-chain metrics, and market data. Specifically, the objectives of this study are as follows:

- To propose a hybrid framework that integrates sentiment, technical indicators, on-chain metrics, and market data.
- To evaluate the comparative effectiveness of various sentiment analysis methods in capturing Bitcoin market sentiment.
- To examine the contribution of each feature category through ablation testing and feature importance analysis.
- To assess the robustness of predictive performance during periods of extreme market volatility.

By applying diverse sentiment analysis methods to a single structured source, this study addresses methodological limitations in previous research that relied on a single sentiment model. The proposed framework provides a robust and scalable approach to Bitcoin price prediction under volatile conditions, offering actionable insights for investors, regulators, and researchers.

To implement this framework, sentiment features from both lexicon-based and large language model (LLM)-based models were integrated with multiple time series models, including AutoRegressive Integrated Moving Average with eXogenous variables (ARIMAX), convolutional neural networks (CNN), recurrent neural networks (RNN), gated recurrent units (GRU), long short-term memory (LSTM), bidirectional LSTM (Bi-LSTM), temporal convolutional networks (TCN), CNN-LSTM hybrids, and Autoformer. Model performance was evaluated using mean absolute percentage error (MAPE) and R-squared (R^2) value, with additional ablation testing and stress-testing under extreme volatility to ensure robustness.

The remainder of this paper is organized as follows. Section 2 reviews relevant literature on sentiment analysis and its application in financial forecasting. Section 3 presents the dataset, preprocessing pipeline, and modeling strategies. Section 4 details the experimental setup and results. Finally, Sect. 5 discusses the implications, limitations, and directions for future research.

2 Literature Review

2.1 Advancement in Sentiment Analysis Methods

Sentiment Analysis is a field within natural language processing (NLP) and data analysis that involves extracting sentiment information from a given text [9]. It is actively applied across various domains, primarily used to extract and analyze emotions from unstructured textual data such as social media, reviews, and news articles [10, 11]. As the field has evolved, various approaches have been developed to improve sentiment analysis, starting with early lexicon-based methods.

Early research in sentiment analysis largely relied on lexicon-based methods. Lexicon-based sentiment analysis refers to approaches that use predefined dictionaries of words and their associated sentiment scores to analyze and determine the emotional tone. Methods such as TextBlob, Pattern, Valence Aware Dictionary and sEntiment Reasoner (VADER), and Linguistic Inquiry and Word Count (LIWC) are the representative examples of this approach [12, 13]. While this method focuses on individual words and has limitations in fully capturing the overall nuances of sentiment, it is cost-effective and performs well with real-world data.

However, lexicon-based methods often struggle to account for contextual relationships between words, limiting their applicability in capturing complex emotional expressions. This gap motivated the shift toward machine learning and deep learning-based approaches, which better handle contextual understanding and semantics in text.

Recent advancements in computing power and deep learning have further enabled the development of large pre-trained language models such as BERT, Generative Pre-trained Transformer (GPT), and LLM Meta Artificial Intelligence (LLaMA). BERT significantly enhances sentiment analysis by bidirectionally capturing text context, thereby enabling a more precise understanding of word meanings and their relationships within a sentence [14]. Additionally, GPT excels in understanding and predicting complex emotional expressions [15]. The LLaMA series, including LLaMA 2 and LLaMA 3, is designed to be both highly efficient and scalable, offering advanced performance across various NLP tasks [16]. These models are particularly beneficial in analyzing the nuanced and dynamic sentiment patterns in social contexts. The adoption of these sophisticated models has substantially led to more precise and nuanced interpretations of text data across various applications.

2.2 Utilization of Sentiment Analysis in Financial Analysis

Sentiment analysis has increasingly become a crucial tool for providing insights into financial markets and investor behavior, while enhancing the accuracy of predictive models through integration with traditional financial indicators [17–21]. In particular, it enables effective analysis of the impact of sentiment on financial decision-making and market dynamics by identifying trends in investor sentiment.

Previous studies have demonstrated the application of sentiment analysis in financial prediction. For example, Jin et al. [17] addressed the challenges of stock market prediction, such as noise and volatility, by proposing a deep learning-based model that incorporates sentiment. The study confirmed that sentiment trends significantly improve prediction accuracy. Similarly, Jing et al. [18] employed CNN to classify investor sentiment extracted from stock forums and combined it with LSTM for prediction. The results showed that hybrid models integrating sentiment analysis outperform single-model approaches.

Sentiment analysis has also been utilized in the cryptocurrency market to predict price movements. Naeem et al. [19] proposed a machine learning-based approach for currency exchange rate prediction utilizing sentiment analysis from Twitter and achieved an accuracy of 82.14% using logistic regression. Wolk [20] also proposed a method for predicting cryptocurrency prices using sentiment analysis, specifically leveraging Twitter and Google Trends to forecast short-term prices of major cryptocurrencies. The study confirmed that social media sentiment significantly impacts cryptocurrency prices. Critien et al. [21] analyzed the relationship between sentiment extracted from Twitter and future prices across various time intervals and presented a multi-class classification model to predict the magnitude of price changes. The experiments demonstrated the ability to classify price direction and magnitude with 63% accuracy.

These studies collectively highlight that incorporating sentiment analysis into financial prediction models can enhance accuracy by providing unique insights into market dynamics. However, most previous studies have relied on a single sentiment source and integrated sentiment information without explicitly distinguishing the contributions of each component, which made it hard to assess the relative value of different input features. To address these limitations, the proposed framework integrates multiple sentiment signals derived from diverse language models, along with technical indicators and on-chain data, within a modular architecture. This design allows for more transparent and adaptable forecasting, distinguishing the proposed model from recent hybrid approaches.

3 Materials and Methods

The experimental process is structured into several key steps to ensure a comprehensive analysis (Fig. 1). First, data collection and preprocessing were conducted to prepare the dataset for modeling. Subsequently, window sizes were determined, followed by train-test splitting to establish the experimental setup. Time series models were then employed for predictive analysis. The model's performance was evaluated, and further analyses included feature importance assessment, ablation testing, and a robustness test to evaluate model reliability under extreme volatility.

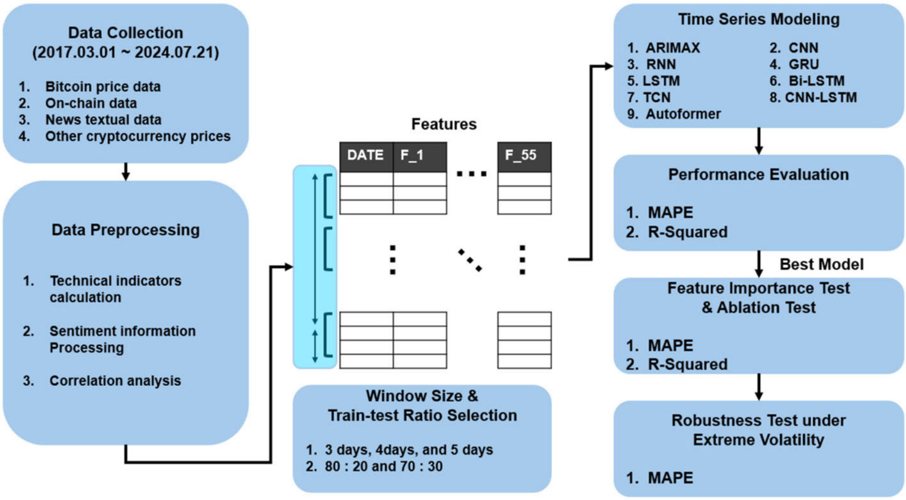

Fig. 1 Experiment workflow diagram

3.1 Data Collection

Data employed in the experiment was collected from March 1, 2017, to July 21, 2024. The collected data were categorized into four types.

The first data category was historical Bitcoin price data, which included the daily closing price, high, low, opening price in US dollars (USD), price fluctuations, and trading volume. This data was downloaded in.xlsx format from Investing.com after setting the desired date range.

The second data category was Bitcoin on-chain data. On-chain data refers to data generated and stored within the blockchain network and reflects the state of the blockchain network [22]. 15 different on-chain data were collected for the experiment: Bitcoin confirmation time, hash rate, bitcoin market cap, median transaction fee, mining profitability, average transaction fee, number of transactions in blockchain, average fee percentage in total block reward, median transaction value, number of unique (from) addresses per day, amount of sent Bitcoin, average block size, median transaction value, top 100 richest addresses to total coins %, average transaction value. This data was retrieved in.csv format from Bitinfocharts.com using a Python scraper using Beautiful Soup library [23].

The third data category was news text data from LexisNexis, which was collected to consider sentiment information related to Bitcoin. The query used was "Bitcoin," and instead of collecting news from all newspapers, news from the "Major World Newspapers" category was collected. This category is generally recognized as providing the most comprehensive and reliable coverage of international topics to readers worldwide [24]. This data was collected in.docx format from the Lexis+ database using a Python scraper and then parsed into a.csv file using R Studio parser [25].

The final data category includes price data for other cryptocurrencies. Existing literature indicates a significant correlation between Bitcoin and other major cryptocurrencies [26–28], providing valuable insights into Bitcoin price movements. To capture these dynamics, the authors incorporated price data for Ethereum, Dogecoin, Litecoin, and Ripple [29]. These cryptocurrencies were selected based on their high market capitalization, substantial trading

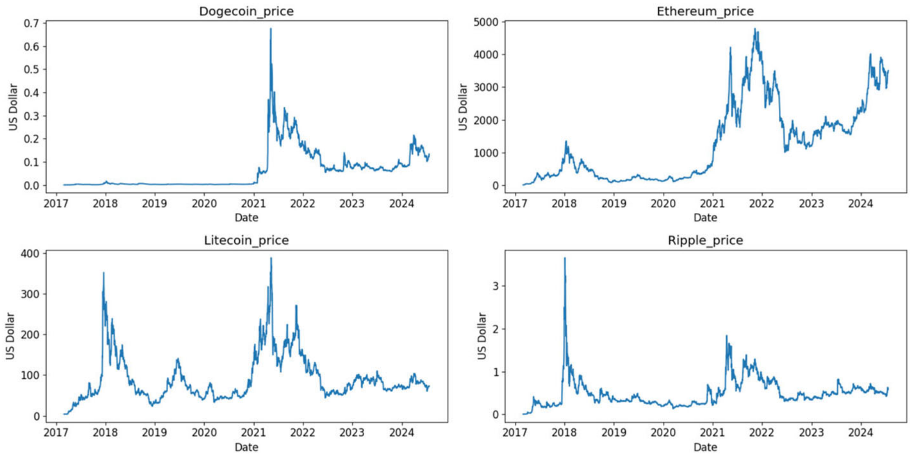

Fig. 2 Graphical representation of collected cryptocurrency price data

volumes, and reliable historical data [30]. The data was collected in.csv format from Bitinfocharts.com using a Python scraper with the Beautiful Soup library. The visualization of the collected cryptocurrencies' price data in USD can be seen in Fig. 2. For all the visualizations, the x-axis represents the date, and the y-axis indicates the value of a single unit of the respective coin at the corresponding time.

3.2 Technical Indicators Calculation

The next step involved calculating technical indicators from the collected historical Bitcoin price data. These indicators help capture trends, momentum, and potential reversal points in the market, enabling a more comprehensive analysis of Bitcoin price behavior.

In this study, the authors utilized commonly used technical indicators, including the simple moving average (SMA), exponential moving average (EMA), relative strength index (RSI), momentum, moving average convergence divergence (MACD), stochastic RSI, stochastic oscillator, Williams %R, and rate of change (ROC). These indicators were selected for their proven effectiveness in financial analysis, as they assist in identifying key market signals such as overbought/oversold conditions, trend direction, and volatility [31–35].

The calculations were conducted using the Python technical analysis (TA) library, a tool specifically designed for technical analysis tasks. For the RSI, SMA, and EMA indicators, values were calculated across multiple time frames to account for both short-term and mid-term market trends. This multi-period approach provides for a more detailed representation of the dynamic price behavior of Bitcoin prices.

3.3 Sentiment Information Processing

Subsequently, a sentiment index was derived from the collected news text to incorporate market sentiment into the analysis. The authors employed various unsupervised learning-based sentiment analysis models, ranging from traditional lexicon-based methods to LLMs.

These models were selected to capture different aspects of sentiment expressed in the news, allowing for a comparative analysis.

3.3.1 Textual Data Preprocessing

To refine the dataset, duplicate articles were removed to avoid redundancy, and the text was standardized through tokenization, lowercase conversion, and stemming, ensuring that words with the same base form are treated equivalently. Punctuation and special characters were removed to clean the text, while lemmatization further normalized the data. Stop words were excluded to focus on informative content, and records containing null values were eliminated to prevent skewed results or inaccurate sentiment analysis. Through these preprocessing steps, the textual data was refined and standardized for more accurate and meaningful analysis in the subsequent sentiment labeling.

3.3.2 Unsupervised Sentiment Labeling

This section provides an overview of the sentiment analysis libraries utilized in this study. A total of seven sentiment analysis tools were employed, each offering unique approaches to extracting sentiment from textual data.

LIWC LIWC is a widely used text analysis tool designed to analyze the emotional, cognitive, and structural components of language [36]. LIWC leverages a proprietary dictionary of words to categorize text into various psychological and linguistic dimensions. In the context of cryptocurrency analysis, LIWC was chosen for its domain-agnostic lexicon, which offers broad emotional coverage suitable for general financial discourse, although it may not fully capture informal or domain-specific expressions commonly seen in crypto texts. LIWC is employed in a variety of fields, including psychology, social science, and communication studies. It is used to study emotional expression in personal writing, understand psychological traits in social media, and analyze communication patterns in various contexts. LIWC returns various emotional values, and the authors extracted the positive and negative tone scores. For the experiment, if the positive tone score was greater than the negative tone score, the text was labeled as positive. Conversely, if the negative tone score was higher, the text was labeled as negative. If the positive and negative tone scores were equal, the text was labeled as neutral.

VADER VADER is a lexicon-based sentiment analysis tool specifically designed and optimized for analyzing social media text [37]. Although originally optimized for short social media texts, VADER was included to serve as a lightweight lexicon-based tool capable of capturing clear sentiment signals in headline-style or succinct financial news content. The authors utilized the compound score provided by VADER to classify the sentiment of each news post. The compound score is a normalized metric that combines the positive, negative, and neutral scores into a single value, ranging from -1 (most negative) to $+1$ (most positive). The authors set the threshold for classifying sentiments at 0.05 based on Hutto et al. [8]. Specifically, posts with a compound score greater than or equal to 0.05 were classified as positive. Posts with a compound score less than -0.05 were classified as negative. The rest were classified as neutral.

Flair Flair utilizes contextualized embeddings to capture the nuanced meanings of words based on their surrounding context [38]. This capability is particularly valuable in the cryptocurrency domain, where news articles often feature mixed or speculative tones. Flair was selected for its strength in identifying subtle, context-dependent sentiment cues that are often overlooked by simpler models. This contextual understanding is crucial for analyzing news articles where sentiment can be influenced by nuanced expressions and varying contexts. The model outputs sentiment labels that classify text into positive, negative, or neutral categories.

FinBERT FinBERT is a domain-specific adaptation of the BERT architecture designed for sentiment analysis in financial texts, including news articles, reports, and corporate disclosures [39]. Trained on an extensive financial dataset, it is optimized for capturing the specific linguistic nuances and terminology of financial contexts. In the context of cryptocurrency analysis where formal financial narratives and institutional reports significantly influence market sentiment, this targeted training makes FinBERT particularly effective in identifying sentiment trends. The model classifies text into positive, negative, or neutral categories.

CryptoBERT CryptoBERT is a BERT-based sentiment analysis model specifically trained on cryptocurrency-related texts [40]. It is designed to analyze social media posts, forums, and other user-generated content within the cryptocurrency domain. By posttraining the BERTweet model on domain-specific datasets, CryptoBERT effectively captures informal expressions and linguistic patterns unique to cryptocurrency communities. This targeted training significantly improves its sentiment classification accuracy compared to general-purpose BERT models. The model categorizes sentiment into positive (Bullish), neutral, and negative (Bearish) labels.

Llama 2 and Llama 3 LLaMA 2 and LLaMA 3 are general-purpose LLMs designed for robust performance across a wide range of text types [16]. In this study, these models were included to evaluate the zero-shot effectiveness of non-specialized models in extracting sentiment from cryptocurrency-related texts, serving as a baseline for comparison with domain-specific models. These models perform sentiment analysis through API-based, prompt-driven interactions. Unlike specialized models that excel in narrow domains, the LLaMA series provides broad linguistic coverage, making them suitable for unsupervised sentiment analysis tasks in diverse contexts. Specifically, the versions used were "meta-llama/Llama-2-7b-hf" and "meta-llama/Meta-Llama-3-8B-Instruct", which classify text into positive, negative, or neutral categories.

3.3.3 Sentiment Index Calculation

The daily sentiment index was calculated for the labeled text data using the Eq. 1 based on existing literature [2, 41].

$$Sentiment\ Index = \frac{N_{Pos} - N_{Neg}}{N_{Pos} + N_{Neg} + N_{Neu}} \tag{1}$$

In the equation above, N_{Pos} refers to the number of positive news articles on a given date, N_{Neg} refers to the number of negative posts, and N_{Neu} represents the number of neutrally labeled posts. When the sentiment index is positive, it indicates that the majority of the textual data on a given date exhibits a positive tone, whereas a negative index signifies a predominantly negative tone. This index serves as a tool for not only distinguishing between positive and negative sentiments but also evaluating their intensity.

Figure 3 provides a comprehensive illustration of the sentiment dynamics and market trends over time, visualized through multiple subplots derived from different data sources

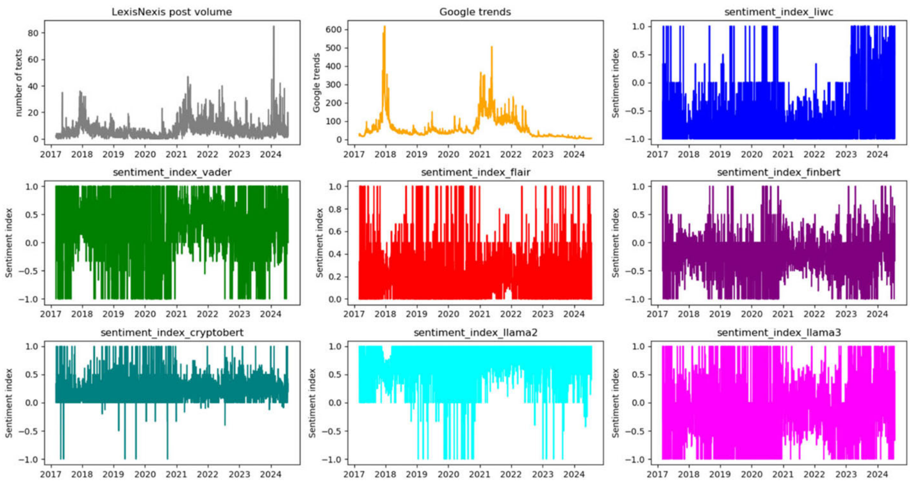

Fig. 3 Data visualization for considering sentiment related to bitcoin

and sentiment analysis methods. The top-left panel shows the LexisNexis post volume, representing the frequency of posts in media over time, while the middle panel displays Google Trends data, indicating the relative interest in search queries. Sentiment index derived from various models, including LIWC, Vader, Flair, FinBERT, CryptoBERT, Llama2, and Llama3, is presented in their respective panels. These visualizations reflect the temporal variations in market sentiment, with each method offering unique perspectives on the underlying emotional trends in textual data. Overall, these subplots provide a comprehensive overview of sentiment fluctuations across different models and data sources.

3.4 Correlation Analysis with Other Cryptocurrencies

After calculating sentiment index, the authors examined the price correlations between Bitcoin's closing prices and those of Ripple, Litecoin, Ethereum, and Dogecoin. To quantify these relationships, Pearson's correlation coefficients were employed which statistically evaluates the linear relationship between two variables [42]. The equation for Pearson's correlation coefficients can be expressed as follows:

$$r = \frac{\sum_{i=1}^{n}\left(X_i - \overline{X}\right)\left(Y_i - \overline{Y}\right)}{\sqrt{\sum_{i=1}^{n}\left(X_i - \overline{X}\right)^2} \cdot \sqrt{\sum_{i=1}^{n}\left(Y_i - \overline{Y}\right)^2}} \tag{2}$$

where X_i and Y_i represent the i-th value of variable X and Y, respectively; $\overline{X}$ and $\overline{Y}$ denotes the means of each variable, and n is the number of data points. Pearson's coefficient ranges from − 1 to 1, where a value closer to 1 indicates a strong positive correlation, a value closer to − 1 indicates a strong negative correlation, and a value around 0 suggests no linear correlation. This analysis allowed identifying the degree of correlation between Bitcoin and these selected cryptocurrencies, providing insight into how price movements in Bitcoin may influence or be influenced by these other assets. Figure 4 illustrates the Pearson correlation coefficients between Bitcoin's closing price and the prices of selected cryptocurrencies, highlighting the varying degrees of linear relationships among these assets.

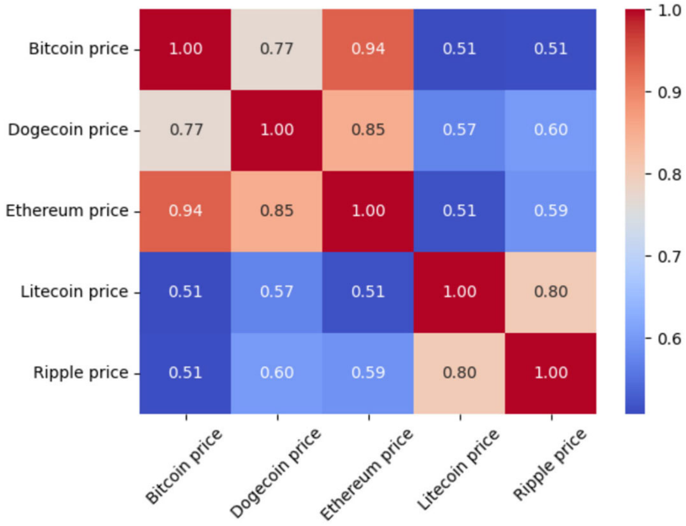

Fig. 4 Visualization of the correlation coefficients between Bitcoin and other cryptocurrencies

The analysis results indicated that Ethereum has the highest correlation with Bitcoin's closing price, with a correlation coefficient of 0.94. In contrast, Litecoin and Ripple showed relatively low correlations with Bitcoin, both at 0.51, while Dogecoin exhibits a comparatively higher correlation of 0.77. These results may be attributed to the market roles, technical similarities, and behavioral patterns of investors for each cryptocurrency.

The strong correlation between Bitcoin and Ethereum could be due to the fact that both assets boast the highest market capitalization in the cryptocurrency market and attract significant attention from investors simultaneously. Ethereum is the second-largest cryptocurrency by market capitalization and considered a key indicator in the cryptocurrency market alongside Bitcoin [43, 44].

On the other hand, the relatively low correlation between Litecoin and Bitcoin (0.51), despite their technical similarities, may stem from differences in their use cases and positions in the market. While Litecoin shares a similar structure with Bitcoin, it emphasizes its role as "digital silver," focusing on improving transaction speed and reducing fees [45]. These characteristics lead to Litecoin being preferred by different market participants or reacting significantly to different market conditions than Bitcoin.

Similarly, the low correlation between Ripple and Bitcoin (0.51) may also be due to fundamental differences in their purposes and functions [46]. Ripple is designed as a protocol for fast and cost-effective international money transfers between financial institutions and is more centralized in structure, unlike Bitcoin. As a result, Ripple is often more sensitive to news or regulatory changes related to the financial industry, which may cause it to behave differently from Bitcoin in terms of price fluctuations.

In the case of Dogecoin, the relatively high correlation of 0.77 may reflect the tendency for its price to align with Bitcoin's overall market trends, despite its meme culture and speculative nature [47]. Although Dogecoin is not as widely used as Bitcoin, its price can still be influenced by the overall investor sentiment in the market.

In conclusion, Ethereum and Dogecoin exhibited high correlations with Bitcoin while the prices of Ripple and Litecoin exhibited lower correlations.

Table 1 Computational complexity of the preprocessing steps

Data loading	$O(n)$
Data merging	$O(k \cdot n + k \cdot m)$
Missing value interpolation	$O(n \cdot p)$
Min–max scaling	$O(n \cdot p)$
Time series reformatting	$O(n \cdot p \cdot s)$
Total complexity	$O(n) + O(k \cdot n) + O(n \cdot p) + O(n \cdot p \cdot s)$

3.5 Data Integration, Window Size and Train-Test Ratio

To utilize the preprocessed datasets as input features for the model, all collected data were combined into a unified dataset. This integration process involved aligning the data by date and applying necessary transformations to ensure consistency across all data sources. The computational complexity associated with this process is summarized in Table 1.

In the table above, n represents the total number of data points included in the dataset, s denotes the number of variables input into the model, and p refers to the sequence length (i.e., window size). k indicates the number of categorical types of merged datasets, and m is the number of rows in the smaller dataset considered during the data merging process.

Given this unified and preprocessed dataset, the next crucial step in time series analysis is defining the "look-back" period, often referred to as the sliding window or window size. In time series analysis, the sliding window or window size refers to the number of consecutive data points used for making predictions. It defines the "look-back" period during which the model considers the previous observations to forecast future values [48]. The selection of an appropriate window size is critical, as it directly influences the model's capacity to capture underlying patterns and trends within the data. Bitcoin is a highly volatile asset, and most existing literature demonstrates that short-term windows tend to have advantages [49, 50]. Based on this, the authors employed window sizes of 3 days, 4 days, and 5 days for the experiment.

Since time series data is inherently sequential, the order of data points is crucial. A well-executed train-test split ensures that the model is tested on data it has not encountered during training, providing a realistic evaluation of its predictive capabilities [51–53]. In the experiment, the authors used train-test split ratios of 70:30 and 80:20 to explore the effects of different splits on model performance.

3.6 Time Series Prediction Framework

In this section, a brief overview of the features and advantages of eight time series prediction models employed for the experiment is provided: ARIMAX, CNN, RNN, GRU, LSTM, Bi-LSTM, TCN, CNN-LSTM, and Autoformer.

3.6.1 ARIMAX

ARIMAX is a statistical model for univariate time series forecasting that incorporates external features. It combines autoregressive terms, differencing for stationarity, and moving average components, while allowing multiple exogenous inputs. In this study, ARIMAX serves as a non-deep learning baseline to assess the incremental value of complex neural architecture when sentiment features are integrated.

3.6.2 CNN

CNNs are primarily used for image data analysis, but they are also effective for extracting features from time series data [54]. A 1-Dimensional CNN is applied to detect local patterns in time series data and generate feature maps through fixed-length filters, which are then passed to subsequent layers. This approach is particularly useful for detecting periodic patterns or identifying local characteristics in the data.

3.6.3 RNN

RNNs are particularly effective at capturing the temporal dependencies inherent in sequential data, such as time series [55]. RNN processes data in a sequence, retaining information from prior time steps and using it to inform predictions at subsequent steps. However, RNNs may encounter challenges with long-term dependencies, which can result in information loss over extended sequences.

3.6.4 GRU

GRU is a variant of RNN designed to mitigate the shortcomings of RNNs, with a structure similar to LSTM but simpler [56]. GRUs use gating mechanisms to selectively retain important information, making them faster to train and requiring fewer computational resources than LSTMs. GRUs provide efficient and effective performance in the time series prediction tasks.

3.6.5 LSTM

LSTM networks were developed to address the long-term dependency problems of RNNs [57]. LSTMs utilize input gate, output gate, and forget gate to selectively remember and forget information. This architecture allows LSTMs to maintain important information over long sequences, making them well-suited for learning complex patterns in time series data.

3.6.6 Bi-LSTM

Bi-LSTM is an extension of LSTM that learns sequences in both forward and backward directions, allowing it to leverage more contextual information [58]. This is particularly useful in the time series prediction problems where both future and past information are important. By gathering information from both directions, Bi-LSTM enables more precise predictions.

3.6.7 TCN

TCN is a CNN-based model for time series processing that can handle sequential data without a recurrent structure [59]. TCNs use dilated convolutions and skip connections to access all past information and enable parallel processing regardless of sequence length. They are particularly effective in learning long sequences.

3.6.8 CNN-LSTM

The CNN-LSTM is a hybrid model that combines the strengths of CNNs and LSTMs. CNNs are first used to extract local patterns from time series data, followed by LSTMs to process the extracted features as sequences and learn temporal dependencies. This model is particularly useful for time series data that exhibits both local patterns and long-term dependencies.

3.6.9 Autoformer

Autoformer is a Transformer-based model designed specifically for time series forecasting [60]. It incorporates an auto-Correlation mechanism to capture long-term dependencies and periodic patterns in sequential data, outperforming traditional attention mechanisms. Additionally, it uses a decomposition block to separate trend and residual components, reducing noise and improving prediction accuracy. Autoformer excels in handling long time series efficiently while maintaining lower computational complexity.

4 Experiment and Result Analysis

4.1 Experimental Setup

The experiments were conducted on the following hardware configurations: Time series modeling was executed on a system equipped with an Intel Core i9-13900 processor, 64 GB of RAM, and an RTX 4090 GPU. Sentiment analysis was performed using an NVIDIA A100 GPU on Google Colaboratory, supported by an Intel Xeon processor with 2 virtual CPUs and 53 GB of RAM. The primary software packages utilized included Torch version 2.3.1, TensorFlow version 2.17.0, scikit-learn version 1.3.2, and Transformers version 4.42.4.

4.2 Performance Evaluation Metrics

This section provides a brief overview of the characteristics and descriptions of the two evaluation metrics used for time series prediction performance: MAPE, and R^2 value.

4.2.1 MAPE

MAPE is the mean of the absolute percentage differences between the predicted and actual values. MAPE expresses error as a percentage of the actual values, which simplifies understanding the model's performance relative to the magnitude of the data (Eq. (3)).

$$MAPE = \frac{100}{n} \times \sum_{i=1}^{n} \left| \frac{y_i - \widehat{y_i}}{y_i} \right| \tag{3}$$

In the Eq. (3), n means the total number of observations, y_i refers to the actual value for the $i - th$ observation, and $\widehat{y_i}$ is the predicted value for the $i - th$ observation. A lower MAPE indicates better accuracy. Its unitless nature is particularly advantageous, as it facilitates comparison across different datasets and measurement units. These characteristics make MAPE especially valuable in fields like business and economics, where relative errors are more interpretable than absolute errors.

4.2.2 R^2 Value

R^2 value (i.e., coefficient of determination) measures the proportion of the variance in the dependent variable that is predictable from the independent variables. It essentially provides a measure of how well the observed outcomes are replicated by the model (Eq. (4)).

$$R^2 = 1 - \frac{\sum_{i=1}^{n} (y_i - \widehat{y_i})^2}{\sum_{i=1}^{n} (y_i - \overline{y})^2} \tag{4}$$

In the Eq. (4), y_i denotes the actual value at time i, $\widehat{y_i}$ represents the predicted value, $\overline{y}$ is the mean of the actual values, and n indicates the total number of observations. The value ranges from 0 to 1, where a value closer to 1 indicates a better fit of the model to the data, while a value of 0 suggests that the model explains none of the variability in the target variable. R^2 value is a widely utilized metric in regression analysis for determining how well a model captures the overall trend in the data.

4.3 Experimental Results

Hyperparameter optimization is crucial for improving model performance, as it involves systematically adjusting parameters to enhance predictive accuracy [61, 62]. It can be broadly categorized into meta-heuristic methods and grid search approaches. While meta-heuristic algorithms are effective in exploring large and complex parameter spaces, they often suffer high computational costs. Considering the clearly defined parameter space and the need for reproducibility, this study adopted a grid search approach for systematic and efficient hyperparameter exploration. The search included learning rate configurations using an exponential decay scheduler, batch sizes of 16, 32, and 64, number of hidden units for RNN-based models set to 32, 64, and 128, and dropout rates of 0.1, 0.2, and 0.3, with the Adam optimizer. The experimental results for the time series models, conducted with varying window sizes (3, 4, and 5 days) and different train test splits (70:30 and 80:20), are described in Table 2.

Analysis results showed that the LSTM model achieved the best performance in terms of both MAPE and R^2 value when using a 3-day window size with 3.93% and 0.99106, respectively. This result highlights the effectiveness of LSTM in capturing the temporal dependencies in Bitcoin price movements, particularly when shorter window sizes are used. Shorter windows allow the model to focus on immediate price movements, enabling better adaptation to rapid market changes, which is a critical factor in highly volatile assets like Bitcoin.

As expected, ARIMAX showed significantly lower prediction accuracy compared to deep learning models across all window sizes. This is primarily due to its limited capacity to capture non-linear temporal dependencies and dynamic feature interactions. While ARIMAX is a robust statistical benchmark, its structural simplicity makes it less suitable for complex financial time series like Bitcoin, especially when incorporating high-dimensional sentiment and technical indicators.

Even though TCN and Autoformer are among the latest models, they underperformed in predicting Bitcoin prices compared to LSTM due to architectural limitations. TCN, which relies on convolutional operations to model temporal dependencies, is effective at capturing local patterns within its receptive field but struggles with long-range sequential dependencies that are crucial in financial time series like Bitcoin. While TCN employs dilated convolutions to expand its receptive field, it lacks mechanisms for maintaining long-term memory, unlike LSTM's gating mechanisms that dynamically retain or discard information over extended

Table 2 Experimental results by time series model

Time series algorithm	Window size	80: 20		70: 30	
		MAPE (%)	R^2	MAPE (%)	R^2
ARIMAX	3	35.20%	− 1.0237	31.17	0.0083
	4	34.30%	− 0.9165	28.97	0.1117
	5	33.80%	− 0.8583	29.14	0.1216
CNN	3	4.91	0.97493	5.43	0.97691
	4	5.45	0.95541	6.07	0.96742
	5	5.91	0.94432	5.45	0.97703
RNN	3	8.17	0.89696	8.91	0.92674
	4	6.38	0.9417	7.43	0.94153
	5	6.59	0.9366	7.66	0.96176
GRU	3	5.06	0.9490	5.11	0.96978
	4	4.85	0.96991	5.17	0.97501
	5	4.59	0.96851	5.18	0.97938
LSTM	3	4.66	0.96523	**3.93**	**0.99106**
	4	5.42	0.9451	5.18	0.96701
	5	5.30	0.9424	4.96	0.97779
Bi-LSTM	3	5.29	0.94595	5.25	0.96823
	4	6.73	0.91321	5.19	0.96955
	5	5.98	0.93874	7.06	0.96193
TCN	3	11.85	0.79354	14.34	0.84691
	4	11.10	0.81986	14.31	0.83117
	5	11.18	0.83671	13.24	0.86003
CNN-LSTM	3	5.19	0.97314	6.06	0.97943
	4	5.84	0.94511	6.47	0.95316
	5	7.33	0.9055	6.71	0.96113
Autoformer	3	8.97	0.90371	11.24	0.85011
	4	9.13	0.89877	11.93	0.83918
	5	9.81	0.87651	12.71	0.81384

The bold values represent the best-performing results for each metric, specifically the lowest MAPE and the highest R-squared value

periods. This limitation results in TCN smoothing over abrupt price movements, leading to the loss of critical information in volatile markets where sudden spikes or drops are significant.

Similarly, Autoformer, a Transformer-based model designed for time series data, also faced challenges with Bitcoin price prediction. In the present context, its underperformance can be explained by a structural mismatch between the model's assumptions and the characteristics of cryptocurrency markets. Autoformer presumes the existence of stable and decomposable temporal structures, such as recurring trends and periodic seasonality. However, Bitcoin prices frequently undergo regime shifts and exhibit irregular volatility patterns driven by sentiment shocks and macroeconomic events. These properties violate the assumptions of temporal stationarity and periodic regularity that Autoformer relies on. Additionally, its decomposition process may amplify high-frequency noise in short- and mid-term windows, where market responses are often driven by abrupt changes in sentiment rather than by consistent seasonal patterns. As a result, Autoformer shows limited predictive effectiveness in this highly volatile and sentiment-sensitive domain.

On the other hand, CNN demonstrated the second-best performance after LSTM, attributed to its ability to effectively capture local patterns in the data. While CNN lacks the sequential processing capabilities of RNN-based models like LSTM, its convolutional layers are adept at identifying short-term trends or anomalies in time series data. This capability is particularly useful in highly volatile markets where rapid responses to recent changes are crucial. CNN's effectiveness in detecting short-term patterns in smaller windows makes it a competitive model for Bitcoin price prediction, though it does not match the overall adaptability of LSTM.

In summary, the findings underscore the superior performance of LSTM in managing temporal sequences, especially with shorter window sizes, making it particularly well-suited for Bitcoin price prediction. The underperformance of TCN and Autoformer highlights the importance of selecting models that align closely with the unique demands of financial time series data. CNN's relatively strong performance further emphasizes the value of local pattern recognition in volatile markets. These insights offer actionable guidance for selecting and optimizing models to predict cryptocurrency prices in highly dynamic and uncertain conditions.

4.3.1 Feature Importance Assessment

The following section outlines the process of evaluating feature importance for the LSTM with a window size of 3, in the context of an explainable time series model. To achieve this, feature importance was quantified through the computation of input gradients. This method measures the impact of each feature on the loss function, providing insights into how much each feature contributes to the model's predictions.

The visualization of feature importance is presented in Fig. 5, where different categories are plotted using distinct colors to enhance visibility.

The visualization results indicate that the Bitcoin transaction-related data, highlighted in pink, had the most significant impact on the predictions, while trading volume had a surprisingly minimal effect. Subsequently, technical indicators had a substantial influence on price prediction. On-chain data and the prices of other cryptocurrencies also contributed to the model's predictions. Lastly, it was observed that sentiment analysis results, particularly those derived using Flair and Llama 2, 3, had a notable impact.

While feature importance indicates the extent to which individual features contribute to the model's predictions, it does not directly account for the interaction effects between features or how the removal of specific features might affect model performance. Therefore, an ablation test is necessary to further investigate these aspects.

4.3.2 Ablation Test

The ablation test is a method used to evaluate how the performance of a model changes when specific features or groups of features are removed [63], allowing for a more practical understanding of the relative importance of features, as it directly assesses the effect of removing a feature on the overall model performance.

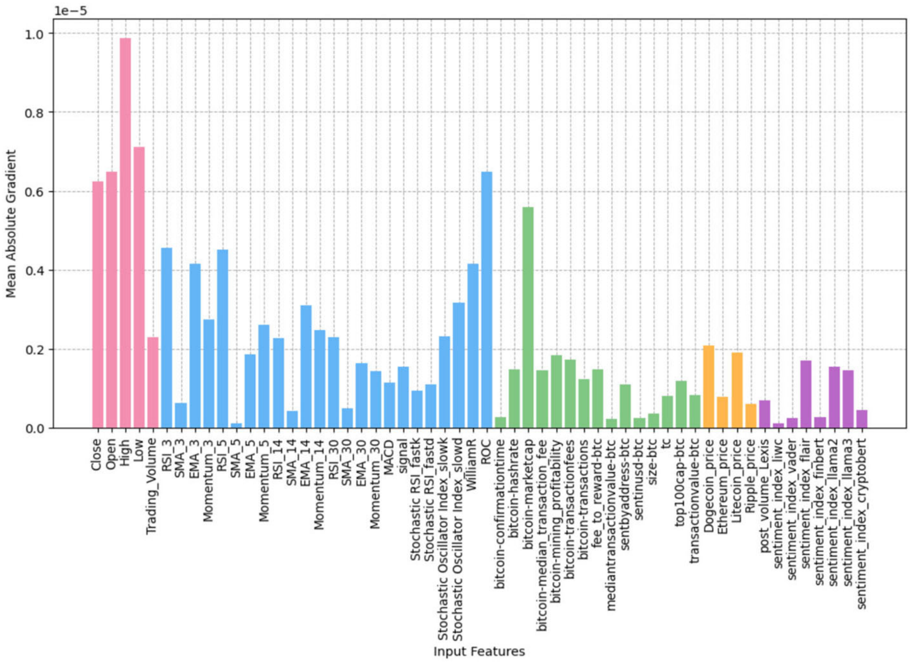

Fig. 5 Feature importance assessment through computation of input gradients

Table 3 Results of ablation test categorized by input features

	ALL input	Without technical indicators	Without on-chain data	Without other cryptocurrency	Without sentiment information
MAPE (%)	3.93	6.27	5.33	6.23	6.61
R^2 value	0.99106	0.98421	0.98531	0.98231	0.98081

Ablation Test by Input Feature Categories To assess the importance of different feature categories, an ablation study by selectively removing each category was conducted (Table 3, Fig. 6).

Firstly, when the technical indicators were removed from the input features, the MAPE increased to 6.27%, and the R^2 value decreased to 0.98421, indicating a significant drop in model performance. Subsequently, when the prices of other cryptocurrencies were excluded, the MAPE rose to 6.23%, and the R^2 value fell to 0.98231, showing a noticeable impact on performance, though less significant compared to the removal of technical indicators. When the on-chain data was excluded, the MAPE was 5.33%, and the R^2 value was 0.98421. Although the performance decline was not as pronounced as when the previous two input features were removed, it still demonstrates that on-chain data significantly contributes to the model. Lastly, when sentiment information was removed, the MAPE increased to 6.61%, and the R^2 value dropped to 0.98081, marking the most substantial decline in performance among all tested categories. This suggests that sentiment-related features are crucial to the predictive

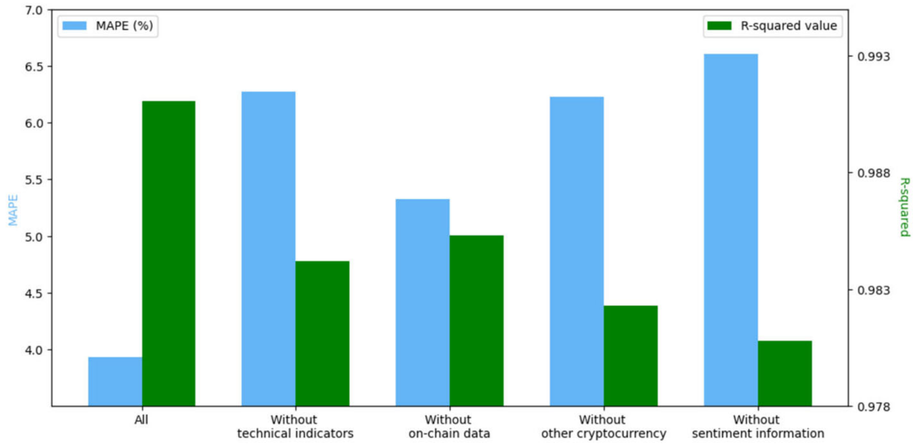

Fig. 6 Visualization of ablation test results by input features

Table 4 Results of ablation test by sentiment information

	LIWC	VADER	FLair	FinBERT	LLAMA2	LLAMA3	CryptoBERT
MAPE (%)	6.61	6.67	6.41	6.56	6.71	6.79	6.81
R^2 value	0.97997	0.9801	0.98125	0.98093	0.98015	0.97991	0.97813

ability of the Bitcoin price prediction model, and their removal leads to a significant decrease in the model's accuracy and explanatory power.

Ablation Test by Sentiment Information from Different Language Models In order to evaluate the impact of various sentiment analysis tools on the performance, an ablation study by sentiment information was performed (Table 4, Fig. 7).

The results indicate that the sentiment analysis tools, LIWC, VADER, LLAMA2, LLAMA3, and CryptoBERT led to a decrease in accuracy compared to the baseline model, which included all other inputs except sentiment analysis, with a MAPE of 6.61% and an R^2 value of 0.98081.

However, the model using Flair showed an improvement in MAPE by approximately 0.2% and an increase in the R^2 value. Similarly, the model using FinBERT showed a slight increase in MAPE by 0.05% and an improvement in the R^2 value. These results suggest that FLAIR and FinBERT are the most effective tools for improving prediction accuracy. Furthermore, it can be observed that combining sentiment analysis information from multiple tools, rather than using them individually, could lead to significant performance improvements.

4.3.3 Performance Evaluation of the Model Under Extreme Volatility

The evaluation of the model's performance during periods of extreme volatility or sudden market changes is essential to ensure its robustness and reliability under challenging conditions [64]. Such volatility serves as a stress test, providing insights into the model's ability

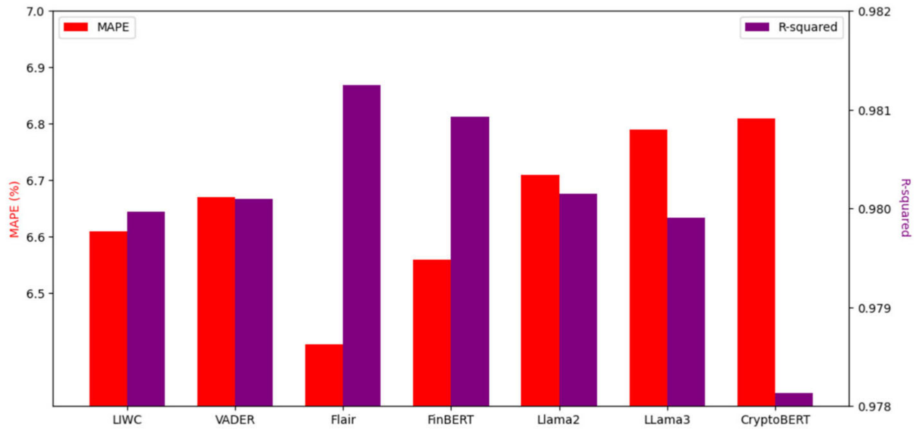

Fig. 7 Visualization of ablation test results by input features

to handle unexpected changes and anomalies that deviate significantly from normal patterns. In practical applications like financial forecasting or risk management, these volatile periods often disproportionately influence decision-making outcomes, making it critical to ensure the model's accuracy and stability in such scenarios. By examining performance under these conditions, valuable insights into the model's strengths, limitations, and potential areas for improvement can be gained, ultimately enhancing its practical relevance and predictive reliability. Figure 8 illustrates the daily volatility and 30-day rolling volatility of Bitcoin across the entire experimental dataset, offering a comprehensive view of overall volatility trends. The daily volatility captures short-term price fluctuations, highlighting immediate market reactions to events or news. In contrast, the 30-day rolling volatility smooths daily changes over a month, revealing sustained trends and broader market stability. Together, these metrics provide insights into both immediate and long-term volatility patterns.

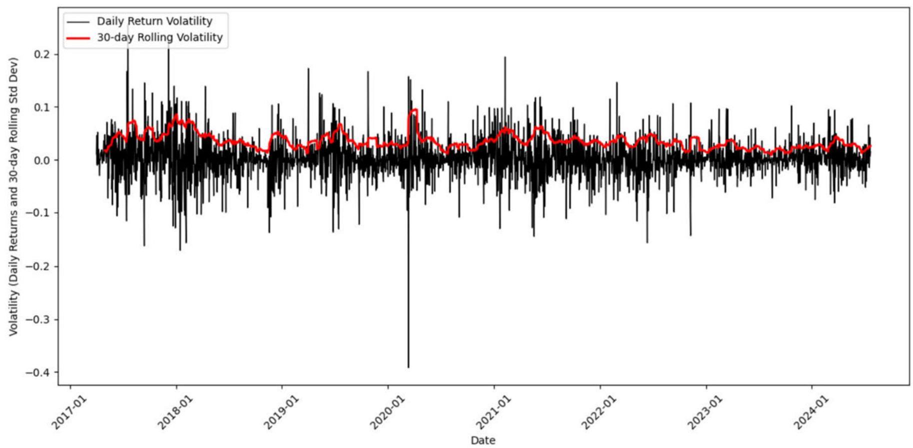

Fig. 8 Daily and 30-day rolling volatility of Bitcoin across the experimental dataset

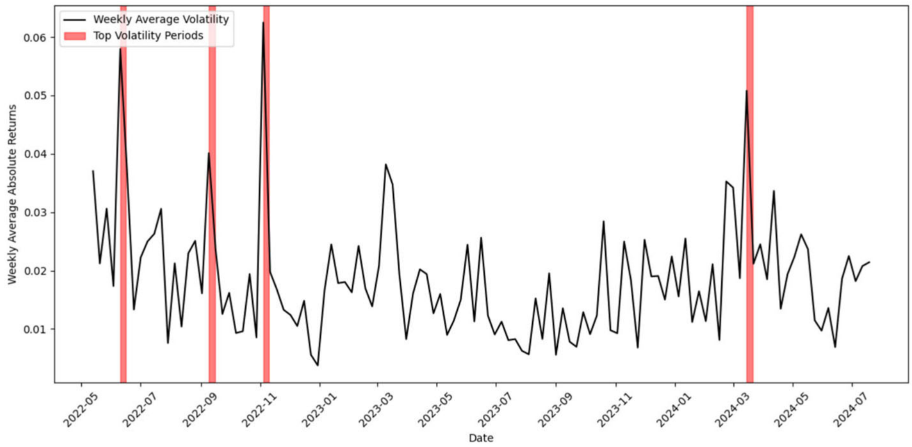

Fig. 9 Top four weekly volatility periods highlighted in red in Bitcoin

From this visualization, the analysis further identified the top four non-overlapping weekly periods of extreme volatility within the test dataset. Volatility was calculated using absolute daily returns, resampled into weekly intervals to compute average values. These periods reflect extreme market conditions and were selected for evaluating the model's predictive performance (Fig. 9):

The LSTM with a 70:30 split and a window size of 3 was used to predict prices for the four identified periods. Prediction performance was evaluated using MAPE, with normalized values for more intuitive visualization (Fig. 10, Table 5). Fig. 9 visualizes the results, where each graph represents a period, where the blue line indicates actual values, and the yellow line represents predicted values.

Table 5 summarizes the model's performance across the identified periods. The MAPE values range from 1.49 to 4.03% across the periods. While Period 1, characterized by the highest level of volatility, shows a relatively higher MAPE, the overall low values indicate that the model provides reliable predictions even under extreme volatility conditions.

5 Discussion and Conclusions

This study focused on factors influencing Bitcoin prices, particularly focusing on sentiment analysis using various language models. The results indicated that with a train/test ratio of 70:30, an LSTM model with a window size of 3 achieved the best performance, with an MAPE of 3.93% and an R^2 value of 0.99106. This suggests that short-term predictions align better with Bitcoin's volatile nature, and a 70:30 split performs better than an 80:20 split for this period. While direct comparisons with other studies are challenging due to differences in evaluation metrics, timeframes, and scaling methods, this study demonstrates superior performance over previous research using Bitcoin datasets spanning more than 2000 days. For instance, Wen and Ling [65] tried to predict Bitcoin prices from January 2018 to December 2022 reported an R^2 value of 0.988, while Nair et al. [66] employed data from September 2014 to January 2022 achieved an R^2 value of 0.94383.

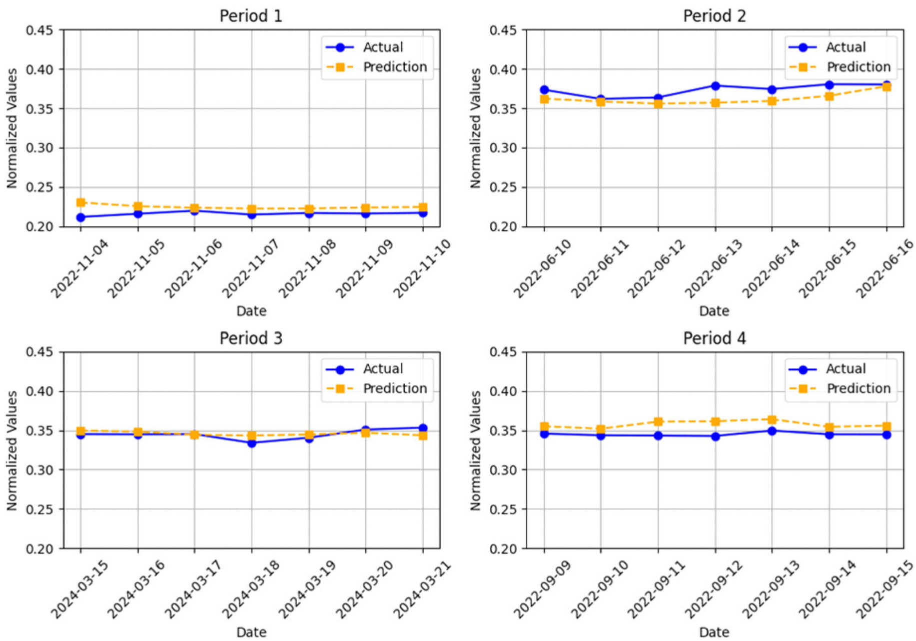

Fig. 10 Prediction on top four weekly volatility periods.
• Period 1: November 4, 2022–November 10, 2022 (Average Volatility: 0.0654)
• Period 2: June 10, 2022–June 16, 2022 (Average Volatility: 0.0543)
• Period 3: March 15, 2024–March 21, 2023 (Average Volatility: 0.0482)
• Period 4: September 9, 2022–September 15, 2022 (Average Volatility: 0.0467)

Table 5 Prediction performance evaluation on top four weekly volatility periods

	Period 1	Period 2	Period 3	Period 4
MAPE (%)	4.02568	2.90082	1.49462	3.70792

A key methodological contribution of this study lies in its integration of sentiment features from multiple analysis tools, including both lexicon-based and large language model-based approaches. Rather than relying on a single sentiment model which may be limited in capturing the linguistic nuances of cryptocurrency-related financial texts, the framework combines outputs from multiple sentiment tools at the feature level. This design choice was motivated by the potential for label noise in unsupervised sentiment labeling, particularly in domain-specific or ambiguous contexts. Such noise may degrade downstream regression performance if left unaddressed. By aggregating diverse sentiment signals, the proposed approach mitigates model-specific biases, enhances the stability of sentiment features, and improves the reliability of price prediction under volatile conditions. This strategy not only differentiates the framework from previous hybrid models but also contributes to its robustness in practical trading environments.

Feature importance and ablation test results further confirmed that sentiment information played a pivotal role in predictive accuracy, outperforming technical indicators, on-chain data, and other cryptocurrency prices when evaluated in isolation. Among the sentiment tools, Flair and FinBERT yielded the most substantial improvements in model performance, while combining all sentiment features consistently led to the lowest error rates. The model's reliability was also validated under periods of extreme volatility, with MAPE values ranging from 1.49 to 4.03% across four high-volatility intervals. These results demonstrate the model's robustness under sudden market shifts.

Beyond predictive accuracy, the proposed framework offers practical utility by integrating sentiment signals from multiple language models, making it applicable to sentiment-informed investment strategies, market risk alerts, and the design of rule-based trading systems. Its modular structure also facilitates targeted analysis of feature contributions, allowing flexible adaptation to different data configurations and serving as a foundation for potential integration with automated trading environments.

Despite its contributions, this study has several limitations. First, although the predictive contribution of sentiment information was systematically assessed, the classification accuracy of each sentiment model was not directly evaluated. This decision was made because the primary objective of the study is to assess the contribution of sentiment-derived features to price forecasting, rather than to measure the classification accuracy of sentiment labels. Second, comparative analysis with previous studies was limited by the reproducibility of data, as time series data can vary based on length and period. Third, the analysis focused solely on Bitcoin and did not consider other cryptocurrency assets. Broader applicability of the framework would require adapting sentiment inputs and technical indicators to other asset-specific market behaviors. Finally, although this study employed grid search for hyperparameter optimization due to its reproducibility and simplicity, future research may consider employing meta-heuristic algorithms to enhance model tuning efficiency in complex forecasting settings [67–69].

Acknowledgements We would like to thank Editage (www.editage.co.kr) for English language editing.

Author Contributions The author contributions to the paper are as follows: study conception and design: HSJ, HL; data collection: HSJ, HL; model analysis and interpretation of results: HSJ, HL; draft manuscript preparation: HSJ, HL, JHK; All authors reviewed the results and approved the final version of the manuscript.

Funding This research was supported by the SungKyunKwan University and the BK21 FOUR(Graduate School Innovation) funded by the Ministry of Education(MOE, Korea) and National Research Foundation of Korea(NRF). This research was supported by the 2025 government (Ministry of Education) funding through the Seoul RISE Project (S-2025-1610-000).

Data Availability The data that support the findings of this study are available from the corresponding author upon reasonable request.

Declarations

Ethical Approval Not applicable.

Conflict of interest The authors declare no competing interests.

References

1. Olabanji SO, Oladoyinbo TO, Asonze CU, Adigwe CS, Okunleye OJ, Olaniyi OO (2024) Leveraging fintech compliance to mitigate cryptocurrency volatility for secure us employee retirement benefits: Bitcoin ETF case study. Preprint https://doi.org/10.2139/ssrn.4739190
2. Jung HS, Lee SH, Lee H, Kim JH (2023) Predicting bitcoin trends through machine learning using sentiment analysis with technical indicators. Comput Syst Sci Eng. https://doi.org/10.32604/csse.2023.034466
3. Wang J, Ma F, Bouri E, Guo Y (2023) Which factors drive Bitcoin volatility: macroeconomic, technical, or both? J Forecast 42(4):970–988. https://doi.org/10.1002/for.2930
4. Khedr AM, Arif I, El BM, Alhashmi SM, Sreedharan M (2021) Cryptocurrency price prediction using traditional statistical and machine-learning techniques: a survey. Intell Syst Account Finance Manag 28(1):3–34. https://doi.org/10.1002/isaf.1488
5. Kraaijeveld O, De Smedt J (2020) The predictive power of public Twitter sentiment for forecasting cryptocurrency prices. J Int Finan Markets Inst Money 65:101188. https://doi.org/10.1016/j.intfin.2020.101188
6. Jung HS, Kim JH, Lee H (2024) Decoding Bitcoin: leveraging macro and micro factors in time series analysis for price prediction. PeerJ Comput Sci 10:e2314. https://doi.org/10.7717/peerj-cs.2314
7. Lee H, Kim JH, Jung HS (2024) Deep-learning-based stock market prediction incorporating ESG sentiment and technical indicators. Sci Rep 14(1):10262. https://doi.org/10.1038/s41598-024-61106-2
8. Mishev K, Gjorgjevikj A, Vodenska I, Chitkushev LT, Trajanov D (2020) Evaluation of sentiment analysis in finance: from lexicons to transformers. IEEE Access 8:131662–131682. https://doi.org/10.1109/ACCESS.2020.3009626
9. Lee H, Lee SH, Nan D, Kim JH (2022) Predicting user satisfaction of mobile healthcare services using machine learning: confronting the COVID-19 pandemic. J Organ End User Comput (JOEUC) 34(6):1–17. https://doi.org/10.4018/JOEUC.300766
10. Lee H, Lee SH, Park H, Kim JH, Jung HS (2024) ESG2PreEM: automated ESG grade assessment framework using pre-trained ensemble models. Heliyon 10(4):e26404. https://doi.org/10.1016/j.heliyon.2024.e26404
11. Lee H, Jung HS, Lee SH, Kim JH (2023) Robust sentiment classification of metaverse services using a pre-trained Language Model with soft voting. KSII Trans Internet Inf Syst (TIIS) 17(9):2334–2347. https://doi.org/10.3837/tiis.2023.09.002
12. Mahmoudi A, Jemielniak D, Ciechanowski L (2024) Assessing accuracy: a study of Lexicon and rule-based packages in R and Python for sentiment analysis. IEEE Access 12:20169–20180. https://doi.org/10.1109/ACCESS.2024.3353692
13. Vashishtha S, Susan S (2019) Fuzzy rule based unsupervised sentiment analysis from social media posts. Expert Syst Appl 138:112834. https://doi.org/10.1016/j.eswa.2019.112834
14. Devlin J (2018) Bert: pre-training of deep bidirectional transformers for language understanding. arXiv prerint arXiv:1810.04805
15. Brown et al (2020) Language models are few-shot learners. Adv Neural Inf Process Syst 33:1877–1901
16. Touvron H et al (2023) Llama: open and efficient foundation language models. arXiv preprint arXiv:2302.13971
17. Jin Z, Yang Y, Liu Y (2020) Stock closing price prediction based on sentiment analysis and LSTM. Neural Comput Appl 32:713–9729. https://doi.org/10.1007/s00521-019-04504-2
18. Jing N, Wu Z, Wang H (2021) A hybrid model integrating deep learning with investor sentiment analysis for stock price prediction. Expert Syst Appl 178:115019. https://doi.org/10.1016/j.eswa.2021.115019
19. Naeem S et al (2021) Machine learning-based USD/PKR exchange rate forecasting using sentiment analysis of Twitter data. Comput Mater Contin 67(3):3451–3461. https://doi.org/10.32604/cmc.2021.015872
20. Wołk K (2020) Advanced social media sentiment analysis for short-term cryptocurrency price prediction. Expert Syst 37(2):e12493. https://doi.org/10.1111/exsy.12493
21. Critien JV, Gatt A, Ellul J (2022) Bitcoin price change and trend prediction through twitter sentiment and data volume. Financ Innov 8(1):1–20. https://doi.org/10.1186/s40854-022-00352-7
22. Zhu LH, Zheng BK, Shen M, Gao F, Li HY, Shi KX (2020) Data security and privacy in bitcoin system: a survey. J Comput Sci Technol 35:843–862. https://doi.org/10.1007/s11390-020-9638-7

23. Richardson L (2007) Beautiful soup documentation. April. https://readthedocs.org/projects/beautiful-soup-4/downloads/pdf/latest/. Accessed 18 Jan 2025
24. Jung HS, Lee H, Kim JH (2023) Unveiling cryptocurrency conversations: insights from data mining and unsupervised learning across multiple platforms. IEEE Access 11:130573–130583. https://doi.org/10.1109/ACCESS.2023.3334617
25. Jung HS, Lee H, Woo YS, Baek SY, Kim JH (2024) Expansive data, extensive model: investigating discussion topics around LLM through unsupervised machine learning in academic papers and news. PLoS ONE 19(5):e0304680. https://doi.org/10.1371/journal.pone.0304680
26. Akbulaev N, Mammadov I, Hemdullayeva M (2020) Correlation and regression analysis of the relation between Ethereum price and both its volume and Bitcoin price. J Struct Finance 26(2):46–56. https://doi.org/10.3905/jsf.2020.1.099
27. Ghorbel A, Jeribi A (2021) Investigating the relationship between volatilities of cryptocurrencies and other financial assets. Decis Econ Finance 44(2):817–843. https://doi.org/10.1007/s10203-020-00312-9
28. Bouri E, Gabauer D, Gupta R, Tiwari AK (2021) Volatility connectedness of major cryptocurrencies: the role of investor happiness. J Behav Exp Financ 30:100463. https://doi.org/10.1016/j.jbef.2021.100463
29. Bhosale J, Mavale S (2018) Volatility of select crypto-currencies: a comparison of Bitcoin, Ethereum and Litecoin. Annu Res J SCMS Pune 6(1):132–141
30. Deniz A, Teker D (2020) Determinants of cryptocurrency market: an analysis for bitcoin, Ethereum and Ripple. Int J Bus Soc Sci 11(11):36–45. https://doi.org/10.30845/ijbss.v11n11p5
31. Oriani FB, Coelho GP (2016) Evaluating the impact of technical indicators on stock forecasting. In: 2016 IEEE symposium series on computational intelligence (SSCI). IEEE, pp 1–8. https://doi.org/10.1109/SSCI.2016.7850017
32. Ayala J, García TM, Noguera JLV, Gómez VF, Divina F (2021) Technical analysis strategy optimization using a machine learning approach in stock market indices. Knowl-Based Syst 225:107119. https://doi.org/10.1016/j.knosys.2021.107119
33. Kang BK (2021) Improving MACD technical analysis by optimizing parameters and modifying trading rules: evidence from the Japanese Nikkei 225 futures market. J Risk Financ Manag 14(1):37. https://doi.org/10.3390/jrfm14010037
34. Dongrey S (2022) Study of market indicators used for technical analysis. Int J Eng Manag Res 12(2):64–83
35. Kirkpatrick CD II, Dahlquist JR (2010) Technical analysis: the complete resource for financial market technicians. FT Press, New Jersey
36. Tausczik YR, Pennebaker JW (2010) The psychological meaning of words: LIWC and computerized text analysis methods. J Lang Soc Psychol 29(1):24–54. https://doi.org/10.1177/0261927X09351676
37. Hutto C, Gilbert E (2014) Vader: a parsimonious rule-based model for sentiment analysis of social media text. In: Proceedings of the international AAAI conference on web and social media, vol 8, no 1, pp 216–225. https://doi.org/10.1609/icwsm.v8i1.14550
38. Akbik A et al (2019) FLAIR: an easy-to-use framework for state-of-the-art NLP. In: Proceedings of the 2019 conference of the North American chapter of the association for computational linguistics (demonstrations), pp 54–59
39. Araci D (2019) FinBERT: financial sentiment analysis with pre-trained language models. arXiv prerint arXiv:1908.10063
40. Kulakowski M, Frasincar F (2023) Sentiment classification of cryptocurrency-related social media posts. IEEE Intell Syst 38(4):5–9. https://doi.org/10.1109/MIS.2023.3283170
41. Wu S, Liu Y, Zou Z, Weng TH (2022) S_I_LSTM: stock price prediction based on multiple data sources and sentiment analysis. Connect Sci 34(1):44–62. https://doi.org/10.1080/09540091.2021.1940101
42. Pearson K (1895) VII. Note on regression and inheritance in the case of two parents. Proc R Soc Lond 58(347–352):240–242
43. Beneki C, Koulis A, Kyriazis NA, Papadamou S (2019) Investigating volatility transmission and hedging properties between Bitcoin and Ethereum. Res Int Bus Finance 48:219–227. https://doi.org/10.1016/j.ribaf.2019.01.001
44. Ante L (2022) The non-fungible token (NFT) market and its relationship with Bitcoin and Ethereum. FinTech 1(3):216–224. https://doi.org/10.3390/fintech1030017
45. Yu H, Sun Y, Liu Y, Zhang L (2024) Bitcoin gold, Litecoin silver: an introduction to cryptocurrency valuation and trading strategy. In: Future of information and communication conference. Springer, Cham, pp 573–586
46. Armknecht F, Karame GO, Mandal A, Youssef F, Zenner E (2015) Ripple: overview and outlook. In: Trust and trustworthy computing: 8th international conference, TRUST 2015, Heraklion, Greece, August 24–26, 2015, proceedings 8. Springer, pp 163–180

47. Zhang S, Mani G (2021) Popular cryptoassets (Bitcoin, Ethereum, and Dogecoin), gold, and their relationships: volatility and correlation modeling. Data Sci Manag 4:30–39. https://doi.org/10.1016/j.dsm.2021.11.001
48. Rajabi S, Roozkhosh P, Farimani NM (2022) MLP-based learnable window size for Bitcoin price prediction. Appl Soft Comput 129:109584. https://doi.org/10.1016/j.asoc.2022.109584
49. Jaquart P, Dann D, Weinhardt C (2021) Short-term bitcoin market prediction via machine learning. J Finance Data Sci 7:45–66. https://doi.org/10.1016/j.jfds.2021.03.001
50. Wirawan IM, Widiyaningtyas T, Hasan MM (2019) Short term prediction on bitcoin price using ARIMA method. In: 2019 International seminar on application for technology of information and communication (iSemantic). IEEE, pp 260–265
51. Hyndman RJ (2018) Forecasting: principles and practice. OTexts, 2018, Australia
52. Makridakis S, Hibon M (2000) The M3-Competition: results, conclusions and implications. Int J Forecast 16(4):451–476. https://doi.org/10.1016/S0169-2070(00)00057-1
53. Bergmeir C, Benítez JM (2012) On the use of cross-validation for time series predictor evaluation. Inf Sci 191:192–213. https://doi.org/10.1016/j.ins.2011.12.028
54. LeCun Y et al (1989) Backpropagation applied to handwritten zip code recognition. Neural Comput 1(4):541–551
55. Rumelhart DE, Hinton GE, Williams RJ (1986) Learning representations by back-propagating errors. Nature 323(6088):533–536
56. Cho K (2014) Learning phrase representations using RNN encoder–decoder for statistical machine translation. arXiv preprint arXiv:1406.1078
57. Hochreiter S (1997) Long short-term memory. In: Neural computation. MIT-Press, Cambridge
58. Schuster M, Paliwal KK (1997) Bidirectional recurrent neural networks. IEEE Trans Signal Process 45(11):2673–2681
59. Bai S, Kolter JZ, Koltun V (2018) An empirical evaluation of generic convolutional and recurrent networks for sequence modeling. arXiv preprint arXiv:1803.01271
60. Wu H, Xu J, Wang J, Long M (2021) Autoformer: decomposition transformers with auto-correlation for long-term series forecasting. Adv Neural Inf Process Syst 34:22419–22430
61. Moshayedi AJ, Nasab STM, Khan ZH, Khan AS (2024). Engineering applications of AI and swarm intelligence. In: Meta-heuristic algorithms as an optimizer: prospects and challenges (part II), pp 155–180. https://doi.org/10.1007/978-981-97-5979-8_8
62. Moshayedi AJ, Nasab STM, Khan ZH, Khan AS (2024) Engineering applications of ai and swarm intelligence. In: Meta-heuristic algorithms as an optimizer: prospects and challenges (part I), pp 131–154. https://doi.org/10.1007/978-981-97-5979-8_7
63. Zeiler MD, Fergus R (2014) Visualizing and understanding convolutional networks. In: Computer vision–ECCV 2014: 13th European conference, Zurich, Switzerland, September 6–12, 2014, proceedings, part I 13. Springer, pp 818–833
64. Naeem MA (2024) Navigating median and extreme volatility in stock markets: implications for portfolio strategies. Int Rev Econ Finance 95:103507. https://doi.org/10.1016/j.iref.2024.103507
65. Wen NS, Ling LS (2023) Evaluation of cryptocurrency price prediction using LSTM and CNNs models. JOIV Int J Inf Vis 7(32):2016–2024. https://doi.org/10.30630/joiv.7.3-2.2344
66. Nair M, Marie MI, Abd-Elmegid LA (2023) Prediction of cryptocurrency price using time series data and deep learning algorithms. Int J Adv Comput Sci Appl 14(8):338–347
67. Khan AT, Cao X, Li Z, Li S (2021) Enhanced beetle antennae search with zeroing neural network for online solution of constrained optimization. Neurocomputing 447:294–306. https://doi.org/10.1016/j.neucom.2021.03.027
68. Khan AT, Cao X, Liao B, Francis A (2022) Bio-inspired machine learning for distributed confidential multi-portfolio selection problem. Biomimetics 7(3):124. https://doi.org/10.3390/biomimetics7030124
69. Khan AT, Wang Y, Wang T, Hua C (2025) Neural dynamics for computing and automation: a survey. IEEE Access 13:27214–27227. https://doi.org/10.1109/ACCESS.2025.3539374