Expert Systems With Applications 261 (2025) 125515

Contents lists available at ScienceDirect

Expert Systems With Applications

journal homepage: www.elsevier.com/locate/eswa

MFB: A Generalized Multimodal Fusion Approach for Bitcoin Price Prediction Using Time-Lagged Sentiment and Indicator Features

Ping Han [a,b], Hui Chen [c,a,*], Abdur Rasool [a,*], Qingshan Jiang [a,1], Min Yang [a,2]

[a] Shenzhen Key Laboratory for High Performance Data Mining, Shenzhen Institute of Advanced Technology, Chinese Academy of Sciences, Shenzhen 518055, China
[b] Shenzhen College of Advanced Technology, University of Chinese Academy of Sciences, Shenzhen 518055, China
[c] Shenzhen Polytechnic University, Shenzhen 518055, Guangdong, China

ARTICLE INFO

Keywords:
Bitcoin price prediction
Sentiment analysis
Multimodal fusion
Feature optimization
BiLSTM and BiGRU
Tweets data

ABSTRACT

Bitcoin's volatile nature has made its price prediction a sought-after mathematical model in the FinTech industry. Existing studies, however, need to look into the critical aspect of time-lagged sentiment in Bitcoin price forecasting. This omission is significant because time-lagged sentiment captures delayed market reactions that are not immediately apparent in price movements. Moreover, the correlation between time-lagged sentiment and technical indicators and the limitations of individual machine learning and deep learning models necessitates a comprehensive approach for accurate and reliable Bitcoin price predictions. This paper introduces the multimodal fusion Bitcoin (MFB), an innovative generalized multimodal fusion approach that effectively integrates BiLSTM and BiGRU layers for complex feature extraction. The model employs the BorutaShap algorithm for feature selection and utilizes attention mechanisms and spatial dropout for optimization and generalization. MFB's training and validation use news and tweet data combined with Bitcoin technical indicators to explore the impact of time-lagged sentiment on price movements, leading to more accurate and timely market predictions. The MFB performs superior Bitcoin prediction performance, achieving 97.63% accuracy and an MAE of 0.0065. Experiments highlight MFB's capability to outperform existing models, offering significant insights for investors in making informed decisions. MFB's innovative methodology, particularly in next-hour Bitcoin price forecasting, marks an advancement in financial forecasting. By capturing the nuanced dynamics of market sentiment and its delayed effects, MFB is a pioneering multimodal fusion approach in the FinTech domain, revolutionizing Bitcoin price prediction.

1. Introduction

Market sentiments, reflecting investors' collective attitudes and behaviors towards financial markets like Bitcoin, are crucial in influencing cryptocurrency prices. Positive market sentiment often correlates with a rise in cryptocurrency prices; conversely, negative sentiment can lead to a decline (Abid et al., 2019). Sentiment analysis, mainly focusing on Bitcoin, the most prominent cryptocurrency with a market capitalization of approximately $125 billion (Gyamerah, 2021), has become a vital tool for price prediction in FinTech. As a decentralized, anonymous, and inflation-resistant currency, Bitcoin dominates the market, accounting for over 40 % of the total cryptocurrency market capitalization (Koo & Kim, 2024). Research has consistently shown that Bitcoin price predictions are significantly influenced by sentiment analysis, drawing on text data from diverse sources such as tweets and news articles (Abid et al., 2019; Basiri et al., 2021; Fakharchian, 2023). Integrating this sentiment-laden text data with Bitcoin price data has recently gained traction as a novel approach to cryptocurrency price prediction (Jahanbin & Chahooki, 2023).

Numerous studies (Kapar & Olmo, 2021; Oikonomopoulos et al., 2022; Passalis, 2022) have explored price prediction using sentiment data, revealing the effectiveness of such integration. For instance, integrating sentiment classification from English tweets with S&P 500 index movements using data mining techniques achieved an 80 % accuracy

* Corresponding authors.
E-mail addresses: p.han@siat.ac.cn (P. Han), huichen@szpu.edu.cn (H. Chen), abdurrasoolphd@gmail.com (A. Rasool), qs.jiang@siat.ac.cn (Q. Jiang), min.yang@siat.ac.cn (M. Yang).
[1] https://orcid.org/0000-0002-8040-0308.
[2] https://orcid.org/0000-0001-7345-5071.

https://doi.org/10.1016/j.eswa.2024.125515
Received 3 February 2024; Received in revised form 10 August 2024; Accepted 4 October 2024
Available online 19 October 2024

rate (Kapar & Olmo, 2021). Another research (Oikonomopoulos et al., 2022) utilizing machine learning for Bitcoin prediction reported accurate identification of 43.9 % price increases and 61.9 % price decreases using logistic regression. Furthermore, a multisource financial sentiment analysis using deep learning was applied to forecast Bitcoin prices (Passalis, 2022). Despite these promising developments in sentiment-based prediction models (Fakharchian, 2023; Kapar and Olmo, 2021; Passalis, 2022; Jakubik et al., 2023), challenges persist, particularly in the delay of predictions aligning with real-time market prices. This delay often results in limited predictive performance for Bitcoin price models that rely on sentiment data, which is less than ideal (Serafini et al., 2020).

This research delves into the critical question of how time-delayed text sentiment impacts Bitcoin prices. Imagine a scenario where a tweet positively depicting Bitcoin emerges on Twitter (current name X). Initially, it might not noticeably affect the market, yet over time, its influence could sway public opinion, potentially leading to an uptick in Bitcoin purchases and an increase in its price. Conversely, a tweet with negative connotations might not immediately affect the market but could eventually deter potential buyers, thereby reducing the prices (Baroiu & Ene, 2022; Han et al., 2022). Our research primarily aims to uncover and scrutinize the correlation patterns between the delays in sentiment, as reflected in textual data, and the consequent fluctuations in Bitcoin prices. By meticulously mapping these correlations, we aim to elucidate the direct and immediate impacts of sentiment on Bitcoin's market value and then analyze Bitcoin's price fluctuations by comparing sentiments.

In addition to this endeavor, we also concentrate on how financial stakeholders can leverage these identified correlation patterns between text data and Bitcoin prices for more reliable price predictions. This aspect of our study involves a detailed analysis of Bitcoin's price indicator features. It is complemented by insights from other studies (Basiri et al., 2021; Fakharchian, 2023; Ortu et al., 2022; Deveikyte et al., 2022; Xu et al., 2023), which have explored the interplay between these indicators and broader market trends. For example, a notable approach by researchers (Fakharchian, 2023) involved using convolutional neural networks (CNN) and long short-term memory (LSTM) networks, augmented by feature selection techniques like mutual information regression and correlation-based methods. The MAE value of this model reached 0.025, which is far better than the eight models compared together. Another study (Ortu et al., 2022) utilized technical, trading, and social indicators to predict Ethereum and Bitcoin prices, achieving an accuracy increase from 51 %-55 % to 67 %-84 % in daily classification tasks, thereby emphasizing the benefits of a comprehensive approach.

In 2023, a significant method (Deveikyte et al., 2022) was introduced, leveraging natural language processing (NLP) to analyze sentiment in financial news and tweets and its correlation with FTSE100 market movements. This research discovered that news headline sentiment could predict market returns, while tweet sentiment strongly correlated with market volatility. Incorporating topic modeling via latent Dirichlet allocation, this method attained a directional prediction with an accuracy of 63 % for market volatility, highlighting the efficacy of combining sentiment analysis with topic modeling. Apart from previous existing studies (Basiri et al., 2021; Chen, 2021; Kang et al., 2022; Zhang et al., 2022) utilizing deep neural networks for price prediction, a recent work (Xu et al., 2023) trained bidirectional LSTM and bidirectional gate recurrent unit (BiGRU) models for stock market prediction in an encoder-decoder setup. This approach significantly improves existing FinTech methods and further consolidates the motivation of deep learning techniques in the financial market.

Although the preceding studies mentioned earlier have significantly advanced Bitcoin price prediction, they overlooked the crucial aspect of time-lagged sentiment in their analyses. This omission is noteworthy as integrating time-lagged sentiment with correlational analysis and indicator features is vital for achieving the highest accuracy in Bitcoin price forecasting (Liu et al., 2022; Joon, 2022). Considering time-lagged sentiment data is essential because it captures the delayed market reactions to various stimuli, which are often not immediately reflected in price movements (Liu et al., 2022). In summary, previous studies (Fakharchian, 2023; Jakubik et al., 2023; Hong et al., 2022) have not adequately considered time-lagged sentiment, which is essential for accurately capturing delayed reactions (Ghafoori et al., 2023). Moreover, the correlation between time-lagged sentiment and indicator features has been largely overlooked. Additionally, the limited performance of individual machine learning and deep learning models poses another significant challenge (Jakubik et al., 2023; Ortu et al., 2022; Deveikyte et al., 2022). These issues emphasize the necessity of establishing a comprehensive model that can effectively integrate and address these challenges, thereby improving the accuracy and reliability of Bitcoin price predictions.

This paper introduces the multimodal fusion Bitcoin (MFB), a novel generalized multimodal fusion approach addressing crucial challenges in Bitcoin price prediction. MFB uniquely integrates BiLSTM and BiGRU layers within a multimodal fusion framework. It employs the Boruta-Shap algorithm (Kursa & Rudnicki, 2010) for feature selection, combining it with advanced multimodal techniques like attention mechanisms and spatial dropout for comprehensive model optimization and generalization. This combination enables effective processing and interpretation of complex time-lagged sentiment data. The model undergoes rigorous training and validation using a thorough dataset, which includes news and tweet sentiment data integrated with Bitcoin technical indicators features, facilitating a nuanced understanding of market dynamics. MFB adeptly synthesizes sentiment data with Bitcoin's technical indicators, demonstrated by its notable performance in metrics like MAE, R^2, and F1 scores. The incorporation of the Boruta-Shap algorithm enhances the model's ability to select the most relevant features, contributing to its superior predictive capability. Unlike traditional models in the FinTech realm, MFB's multimodal exploration of time-lagged sentiment and technical indicators offers accurate and timely Bitcoin price predictions crucial for investors and market analysts. This innovative multimodal approach, especially practical in forecasting next-hour Bitcoin prices, positions MFB as a groundbreaking model in financial forecasting, adept at capturing the intricacies of market sentiment and its delayed effects. The data and code mentioned in this article are available at https://github.com/YukikiHan/MFB.

The significant contributions are as follows:

1. MFB multimodal integrates BiLSTM and BiGRU layers sequentially; BiLSTM layers first capture temporal dependencies in the data, then refined by BiGRU layers for contextual analysis to enhance the approach's ability to interpret complex patterns in time-lagged sentiment data.
2. MFB employs the BorutaShap algorithm for feature selection, alongside advanced techniques like attention mechanisms and spatial dropout, to prioritize and select the most impactful features, leading to improved accuracy and prediction efficiency.
3. MFB's training and validation using a diverse dataset that includes news and tweets sentiment data and Bitcoin technical indicator features enable it to predict next-hour Bitcoin prices for short-term traders and outperform traditional models.

The remaining paper is structured as follows: Section 2 reviews existing Bitcoin price prediction methods. Section 3 provides preliminaries, Section 4 introduces our proposed methodology, Section 5 details experiments and results, and Section 6 concludes the study.

2. Literature review

2.1. Traditional feature optimization for price prediction

Feature optimization is a critical aspect of price-prediction models, focusing on extracting pertinent information while eliminating noise and irrelevant features. Traditional feature optimization methods, including random forest (RF), principal component analysis (PCA), and autoencoder (AE), have been widely explored in the context of price prediction. However, their applicability to our proposed study reveals certain limitations.

In a study examining daily open prices of 11 stocks over the past 10 days (Aloraini, 2015), the ensemble feature selection method, incorporating Pearson (Pearson, 1895) and Spearman (Spearman, 1961) correlations, demonstrated improved predictive results. However, the temporal scope of 10 days may not adequately capture the nuanced time-lagged relationships our proposed theme seeks to explore. Similarly, the application of RF to predict the one-day-ahead direction of 12 different indexes in the international market (Kumar et al., 2016) showed its efficacy. The study is found to have more than 75 % of the relative frequency of selection by proposed hybrid models for all stock indices considered in this study. Nonetheless, focusing on short-term predictions and specific indexes may not align with our aim of forecasting Bitcoin prices with longer time horizons.

PCA, a feature extraction method converting high-dimensional feature vectors into low-dimensional components (Htun et al., 2023), has been integrated with neural network models for improved predictions. For example (Das et al., 2019), the author compared the Firefly algorithm with an evolutionary framework for online sequential extreme learning machine (OSELM) with the factor analysis (FA)-based prediction model and found that compared with the genetic algorithm-extreme learning machine (GA-ELM), the improvement exceeded 79 %, 75 %, 40 %, 68 %, 36 %, 36 % (1, 3, 5, 7, 15, 30 days in advance respectively). While PCA reduces dimensionality, its performance may be limited in capturing the intricate relationships and temporal dependencies inherent in Bitcoin price data. Autoencoder (Kramer, 1991), when combined with LSTM models for stock return prediction based on fundamental features (Dami & Esterabi, 2021), the average MAE for all ten shares of the LSTM while using AutoEncoder (LSTM-E) model is 0.034, better than other models. However, the unique characteristics of extending these methods to predict Bitcoin prices must be carefully considered, as the factors affecting the cryptocurrency market may differ significantly from traditional financial methods.

Boruta is a wrapper feature selection method that can help identify all the relevant features for a supervised learning task (Kursa & Rudnicki, 2010). Boruta is based on adding random shadow features to the original data and comparing their importance with the real features using a tree-based model. (Manikandan et al., 2024) compares the performance of three methods with and without Boruta feature selection. The investigation used the Cleveland Clinic cardiology dataset containing 14 features and 303 instances. It was found that the Boruta feature selection algorithm selected the 6 most relevant features, improving the algorithm's results. For example, the accuracy of the decision tree produced by applying Boruta feature selection increased from 75.41 % to 80.33 %. However, the disadvantage of Boruta is that it may ignore some features that are weakly related to the target variable but have strong interaction effects when combined.

Collectively, these studies underscore the significance of feature optimization in improving the accuracy, efficiency, and interpretability of price-prediction models. By discerningly selecting and extracting features, models can achieve superior performance by eliminating irrelevant or redundant information. The next section delves into deep neural networks' principles, advantages, limitations, and practical cases in different fields, building on the foundation laid by feature optimization techniques.

2.2. Deep learning techniques in Bitcoin forecasting

The paradigm shift toward deep learning has ushered in a new era in price prediction, with deep neural networks (DNNs) emerging as the preferred choice over traditional statistical and machine learning models. This section explores the recent advances and challenges posed by DNNs in the context of our proposed study. LSTM networks, renowned for capturing temporal dependencies without succumbing to the vanishing gradient problem, have become instrumental in price prediction tasks (Aslam et al., 2021). The MRC-LSTM method is proposed by (Guo et al., 2021) combining a multi-scale residual convolutional neural network (MRC) and an LSTM to predict the closing price of Bitcoin. The method achieved an MAE of 166.52 and an RMSE of 261.44, significantly better than other baseline methods. However, the short-term focus of the study prompts consideration of its applicability to our proposed research, which involves forecasting Bitcoin prices over longer time horizons.

The optimized LSTM prediction model explored (Zhang et al., 2022) demonstrates high accuracy in predicting the future values of Bitcoin and gold based on historical price series. While promising, the temporal scope and specific assets analyzed may differ from the characteristics inherent in cryptocurrency markets, raising questions about the model's generalizability (Zhan et al., 2018). Hybrid models, such as the 1DCNN-GRU model proposed (Kang et al., 2022), leverage the strengths of one-dimensional convolutional neural networks (1DCNN) and stacked gated recurrent units (GRU). The proposed hybrid model was evaluated on three different cryptocurrency datasets. The experimental results showed that the 1DCNN-GRU model outperformed the existing methods, achieving the lowest RMSE values of 43.933 on the Bitcoin dataset, 3.511 on the Ethereum dataset, and 0.00128 on the Ripple dataset. However, careful consideration is needed to assess its suitability for our study, which involves the fusion of lagged data correlations and optimized features (Bishop, 2006).

As presented (Ye et al., 2022), stacked ensemble prediction models offer a holistic approach by combining distinct base models to enhance overall performance and reduce biases. While effective in utilizing mixed cryptocurrency data, further exploration is required to evaluate their effectiveness within the context of our proposed theme. The integrated deep learning model proposed (Ye et al., 2022) integrates LSTM and GRU with stacked integration technology to predict Bitcoin prices in near real-time. Focusing on short-term prediction raises questions about its applicability to our theme, which involves forecasting prices over longer intervals.

Supplementary Table S1 provides insights into recent research on Bitcoin price prediction, including datasets, feature optimization methods, prediction models, limitations, and future research directions. The feature sets considered encompass a, including open (O), high (H), low (L), close (C), technical indicators (T), sentiment indicators (S), financial news (N), and event-related features (E).

3. Preliminaries

3.1. Deep learning models

A DNN is a collection of neurons organized in a sequence of multiple layers (Montavon et al., 2018). In the DNN model, the neurons of each layer receive the activated neurons of the previous layer and connect with the neurons of the subsequent layers to form a network structure. Each neuron performs simple calculations, such as a weighted sum of inputs, and applies an activation function to each neuron's output, thereby introducing nonlinearity and enabling complex calculations (Zhang et al., 2023). In the context of Bitcoin price prediction, DNNs have been used to analyze historical price data and identify patterns that can be used to make predictions about future prices. Recurrent neural networks (RNNs) have various variants, such as LSTM, GRU, BiLSTM, BiGRU, BRNN, etc., which can handle sequential data, such as text, speech, time series, etc. They have a common feature of introducing memory mechanisms in the network, allowing it to store past information and use it to influence future outputs (Fakharchian, 2023). Because Bitcoin price is a kind of time series data with characteristics such as dynamism, nonlinearity, and high dimensionality, these models can use past price information to predict future price trends and help investors make decisions. We will use these models to build a Bitcoin price prediction system and compare their advantages and disadvantages.

3.2. Data description

Our data is primarily composed of text data and numerical data. The text data consists of Tweets and News data, while the numerical data comprises Bitcoin price data. The Tweets data was collected from Kaggle and included tweets containing the keywords 'Bitcoin' or 'BTC' from January 1, 2021, to May 4, 2021. The News data was obtained from the FINBERT-SIMF platform and includes news about Bitcoin and BTC. The Bitcoin price data from the CoinMarketCap platform includes hourly Open, High, Low, Close, and Volume data from January 1, 2021, to May 4, 2021. Table 1 exhibits the data set's source, sample number, and period information.

3.3. Evaluation metrics

Five evaluation metrics, including mean absolute error (MAE), mean squared error (MSE), coefficient of determination (R^2), root mean squared error (RMSE), and mean absolute percentage error (MAPE), along with four performance metrics, including Recall, Precision, F1, and Accuracy are used to evaluate the performance of the proposed approach. These metrics are widely used in Bitcoin price prediction. Supplementary S2 describes how these evaluation metrics are defined and calculated.

4. Proposed MFB approach

The proposed method's architecture (illustrated in Fig. 1) consists of the following four key stages:

i. *Data Preprocessing*: In the multimodal context, financial tweets and news are cleaned and sentiment-analyzed to prepare the foundational data.
ii. *Lagged Data Correlation*: In a multimodal approach, sentiment scores are cross-correlated with Bitcoin price data to identify time-delayed market effects.
iii. *Feature Optimization*: Leveraging multimodal data, features are meticulously selected and refined using the BorutaShap algorithm and SPB optimizer for precision.
iv. *Fusion Model Generalization*: Our multimodal fusion model harnesses the strengths of BiLSTM and BiGRU layers, incorporating attention mechanisms for nuanced predictive analysis of Bitcoin prices.

The details of these substantial stages are explicitly given in the next sub-sections.

4.1. Problem definition

Given a set of features $X = \{X^1, X^2, \cdots, X^m\}$, where each $X^i = \{x^i_1, x^i_2, \cdots, x^i_n\}$ represents a time series, the task is to predict Bitcoin's closing price. Here, m denotes the number of distinct time series (or feature dimensions), and n is the number of sampling points within each time series. These time series encompass technical indicators such as EMA, MACD, RSI, and sentiment scores computed using Vader. Our goal is to forecast the closing price of Bitcoin for the next hour, denoted as $Y = \{y_1, y_2, \cdots, y_p\}$, where p represents the number of prediction points in the continuous time series. The model aims to generate predictions $\widehat{Y} = \{\widehat{y}_1, \widehat{y}_2, \cdots, \widehat{y}_p\}$, where $\widehat{y}_i$ is the predicted closing price at the i_{th} prediction point. The primary objective is to optimize the model such that the error between the predicted closing prices $\widehat{Y}$ and the actual closing prices Y is minimized. This error is quantified using metrics such as the RMSE, MAE, or MAPE.

Table 1
Data source, number of samples, and period of utilized dataset in this study.

Data set	Source	Number	Period
Tweets	https://www.kaggle.com/datasets/hiraddolatzadeh/bitcoin-tweets-2021–2022	134,509	January 1, 2021, to May 4, 2021
News	https://figshare.com/articles/dataset/MarketData_for_MarketPredict_RESTFul_API_including_News_and_Market_Data/14754966	1826	
Bitcoin price	https://coinmarketcap.com/	2813	

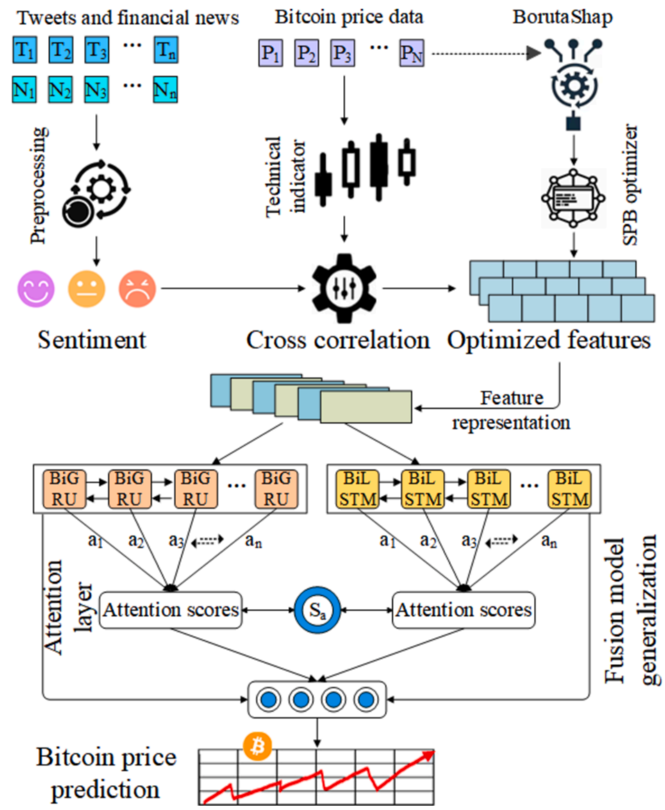


Fig. 1. The framework of our proposed MFB approach.

4.2. Data preprocessing

To preprocess the noisy text data from the web, we initiate by reading a CSV file containing the raw text.

The following cleansing process for each news item and tweet involves a series of streamlined steps, transforming the unstructured data into a structured format for further analysis:

- Convert text to lowercase.
- Replace the URL with the word "URL".
- Replace user mentions with "USER_MENTION".
- Separate hashtags from words.
- Replace multiple dots with a space.
- Remove extra spaces, quotes, and double quotes.
- Replace the emoji with "EMO_POS" or "EMO_NEG".
- Handle duplicate characters.
- Remove hyphens and apostrophes.

Post-cleaning, each word undergoes a validity check to eliminate spelling errors or illegal characters. We then apply the Porter stemming algorithm to reduce words to their base forms. The processed tweets are subsequently compiled into a single, space-separated string of words for analysis. See (Fig. 2).

4.3. Lagged data correlation

In the multimodal fusion approach of MFB, we merge cleaned news and tweets data, aligning text and Bitcoin price data by their date-time indices. Utilizing VADER (Hutto & Gilbert, 2014), we calculate hourly 'compound' sentiment scores and aggregate them to identify overall sentiment trends. Simultaneously, we average hourly Bitcoin closing prices, preparing for a detailed analysis of the correlation between market sentiment and Bitcoin price fluctuations. We define a *Crosscorr* function to calculate the Lag-N cross-correlation between text data and Bitcoin price data. This function has two optional parameters: *Lag* and *Method*. *Lag* specifies the number of lags to consider in cross-correlation calculations, from -20 h to 20 h. *Method* specifies the correlation method to be used. We calculate the cross-correlation between text data and Bitcoin price data at each lag using the Pearson (Pearson, 1895), Kendall (Kendall, 1938), and Spearman (Spearman, 1961) methods, presented in Table 2.

Mathematically, the cross-correlation between two features, x and y, can be defined as follows (Maschotta et al., 2007):

$$R_{xy}[n] = \sum_{m=-\infty}^{\infty} x[m]y[m+n] \tag{1}$$

where m is the index, which can take any integer value, n is the lag, x and y are the two signals, and R_{xy} is the cross-correlation function.

The cross-correlation function measures the similarity between two features as a function of the lag between them. For instance, in our study, if two features of the Bitcoin market are identical, the cross-correlation function peaks at $n = 0$. However, if one feature is temporally shifted relative to another, the peak of the cross-correlation function indicates the extent of this lag. We gain insights into the lag relationships between different market indicators by calculating and analyzing these cross-correlation values. These relationships reflect the dynamic changes and causal relationships of the market and have a direct guiding role in the subsequent process of constructing feature vectors (Liu et al., 2022).

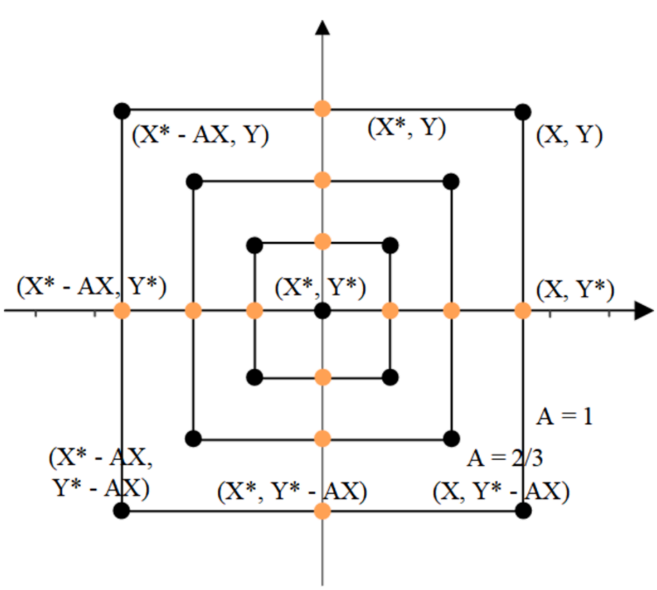


Fig. 2. SPB optimizer search pattern, with agents' positions converging towards the optimal solution X* in a fluctuating feature space, enhancing adaptation to time series data correlations for Bitcoin price prediction.

4.4. Feature optimization

Capitalizing on the cross-correlation analysis outcomes, the MFB approach progresses to constructing feature vectors, crucial for the multimodal framework. This process transforms the existing feature set $O = \{\text{feature}_n\}, n = \{1, 2, \cdots, N\}$ into new feature set $F = \{\text{o\_feature}_w\}, w = \{1, 2, \cdots, W\}$ by combining or modifying original features and representing these new features as numeric or symbolic feature vectors. These vectors are then utilized in machine learning applications to enhance the analysis and prediction of Bitcoin price movements. This approach allows for extracting and applying relevant features, including those technical indicators shaped by the temporal dynamics identified in the cross-correlation analysis. After refining feature vectors derived from Bitcoin price data, the MFB model utilizes these in market analysis, emphasizing technical indicators. These indicators, categorized into the following types, are integral to training the multimodal Bitcoin price prediction model.

i. *Trend Indicators*: Moving averages, exponential moving averages, and MACD help identify price trends.
ii. *Oscillators*: Including RSI, Stochastic Oscillator (SO), and Bollinger Bands, used for gauging price fluctuations.
iii. *Volume Energy Indicators*: Like trading volume and on-balance volume, these measure market participant interest and confidence.

Supplementary S3 describes the definitions and formulas of these technical indicators. A critical aspect of using these technical indicator features is maintaining a quantifiable correlation with the target variable, like the Bitcoin price, while being cautious to avert including an excessive number of indicators to avoid overfitting the model.

For optimal selection of technical indicators in the MFB model, a correlation matrix filter is used as a strategic feature selection method. This method effectively identifies the most relevant indicators by calculating the correlation between each pair of features within the set *O*. Specifically, we compute the Pearson correlation coefficient $P = \{p_1, p_2, \cdots, p_k, \cdots, p_K\}$ for each pair of features in *O*. Here, p_k represents the correlation coefficient for the k_{th} feature pair while p_K is the coefficient for the last, or K_{th} pair. Features exhibiting a correlation coefficient exceeding ± 0.9 are deemed highly correlated, indicating redundancy. Redundant features, offering minimal unique information, are candidates for removal. This process not only diminishes dataset dimensionality but also bolsters the efficacy of the machine-learning algorithms by reducing noise and focusing on the most informative indicators. Consequently, by judiciously filtering and selecting technical indicators based on their correlation, we ensure that the model remains robust, avoiding overfitting while retaining critical predictive capabilities.

Building upon the foundation laid by the correlation matrix filter, which helps us identify and eliminate redundant features, we further refine our feature selection process using the BorutaShap algorithm (Kursa & Rudnicki, 2010). This advanced technique, situated at the intersection of the Boruta feature selection algorithm and Shapley value analysis, aligns with the recent advancements in explainable artificial intelligence. BorutaShap excels in meticulously evaluating and ranking the importance of features, offering a dual advantage: it not only selects an optimal subset of features but also provides a consistent global ranking of these features, aiding in model interpretability. Mathematically, the definition of BorutaShap is as follows (Ghosh & Chaudhuri, 2022):

$$C_i = \sum_{S \subseteq M\{i\}} \frac{|S|!(m - |S| - 1)!}{m!} [n(S \cup \{i\}) - n(S)] \tag{2}$$

where C_i represents the contribution of the i_{th} feature, $M\{i\}$ is the feature

Table 2
Benchmark correlation coefficient methods.

Type	Definition	Formula	Measure
Pearson (Pearson, 1895)	A dimensionless ratio of the product of the covariance and the standard deviation between two variables.	$\gamma = \frac{\sum(x_i - \bar{x})(y_i - y)}{\sqrt{\sum(x_i - \bar{x})^2 \sum(y_i - \bar{y})^2}}$	Values range between −1 and 1, where 1 represents a perfect positive correlation, −1 represents a perfect negative correlation, and 0 represents no correlation.
Kendall (Kendall, 1938)	A measure of the correlation between the ranks of two variables.	$\tau = \frac{n_c - n_d}{n(n-1)/2}$	
Spearman (Spearman, 1961)	Similar to the Kendall correlation coefficient, it uses the rank of the variable rather than the original value of the variable.	$\rho = 1 - \frac{6\sum d_i^2}{n(n^2-1)}$	

set with base m, S is the subset of M with feature i, and $n(S)$ is the prediction result considering the i_{th} feature.

BorutaShap, leveraging TreeSHAP in Python, optimizes feature selection post-correlation matrix filtering. By employing custom sampling to evaluate minimal subsamples, the method isolates key features through iterative comparison of shadow features derived from perturbed eigenvalues with the original dataset. This yields a robust, explanatory feature set that enhances model accuracy and reduces overfitting. However, BorutaShap's static nature limits its adaptability in dynamic markets like Bitcoin, where feature relevance constantly evolves.

In advancing our multimodal MFB approach, we integrate the Surrounding Prey Behavior Optimizer (SPB) to strategically address the dynamic optimization challenges in Bitcoin market prediction. SPB, inspired by the intricate hunting strategies of wolves and depicted in Fig. 3, is a paradigm of swarm intelligence algorithms designed to adapt to the fluctuating nature of financial data.

Fig. 3 illustrates the SPB's search strategy, where each agent's position in the feature space is adjusted relative to the current optimal solution, denoted as X^*. The agents explore the search space by moving in a pattern that emulates wolves' surroundings and converging behavior toward their prey. The positions (X, Y), (X', Y'), and their intermediate adjustments (AX, AY) represent potential solutions within the search space. The parameter A, varying between $2/3$ and 1, modulates the agents' movement, ensuring a balance between exploration and exploitation- a key to addressing the non-static nature of lagged data correlations in financial time series. The figure further embodies the mathematical framework of SPB, with Eq. (3) defining the iterative

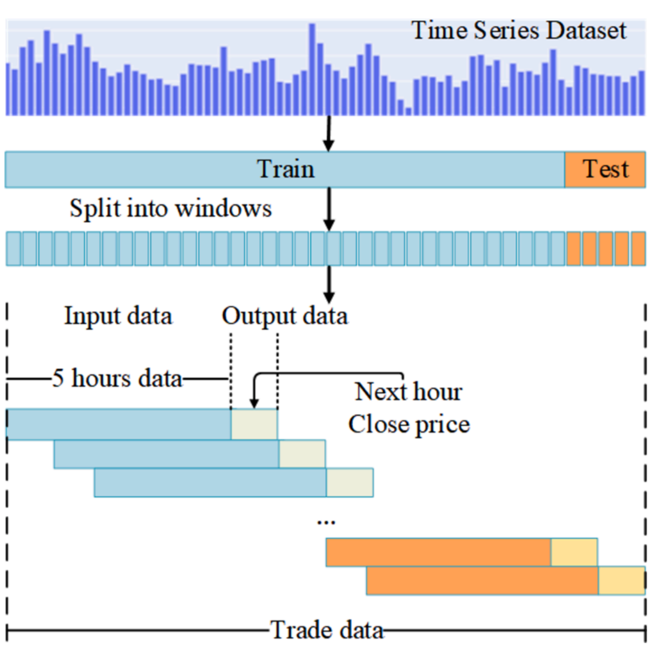


Fig. 3. Mechanism of predicting Bitcoin price for the next hour using time-lag windows.

update of each agent's position (Nadimi-Shahraki et al., 2023):

$$X_i^{t+1} = X^{*t} - A \bullet D \tag{3}$$

$$\mathrm{D} = \left|\mathrm{C} \times X^{*t} - X_i^t\right| \tag{4}$$

where t is the current iteration, X_i^t is the position of the i_{th} wolf for the current iteration, X^{*t} represents the position vector of the best-obtained solution thus far that is updated in each iteration if there is a better solution. D denotes the distance between the prey X^{*t} and the wolf X_i^t.

Algorithm 1. Dynamic feature optimization with correlation-based selection and SPB enhancement.

```
Input: Original feature set O, Pearson correlation coefficient P
Output: Optimized feature set F
Calculate P
for k in 1 to K do
  if p_k > 0.9 or p_k < −0.9 do
    Remove one feature of the feature pairs
Set the maximum number of iterations I
for i in 1 to I do
  Create the shadow feature S, fit the Random Forest model with the new matrix B = [O, S]
  if Z-score of O > maximum Z-score of S
    ACCEPT the original feature, calculate the feature importance using Eq. (2)
Set the maximum number of iterations L, initialize the population
while t < L do
    for each search agent do
      Update the position of current search agent X_i^t, a, A and C using Eqs. (3) to (7)
      Calculate the fitness of all search agents, Update X^{*t}
t = t + 1, F ← X^{*t}
Return F
```

A and C are coefficient vectors, and the calculation formula is as shown in Eqs. (5) and (6) (Nadimi-Shahraki et al., 2023):

$$\mathrm{A} = 2 \times \mathrm{a} \times \mathrm{r}_1 - \mathrm{a} \tag{5}$$

$$\mathrm{C} = 2 \times \mathrm{r}_2 \tag{6}$$

$$\mathrm{a} = 2 - \mathrm{t} \times \left(\frac{2}{\mathrm{MaxIter}}\right) \tag{7}$$

where a is the convergence factor. As the number of iterations decreases linearly from 2 to 0, the random number between 0 and 1 is taken modulo of r_1 and r_2. The parameter MaxIter denotes the total number of iterations. The pseudocode of the feature optimization algorithm is shown in Algorithm 1.

Incorporating domain-specific feature knowledge into the feature set refined by BorutaShap allows SPB to dynamically adjust to real-time market shifts. This adaptability addresses the inherent lag in data correlation and boosts the model's predictive accuracy. By integrating SPB's adaptive search pattern and convergence properties with BorutaShap's static selection, the model becomes more robust against overfitting and more sensitive to Bitcoin's temporal price dynamics.

4.5. Fusion model generalization

The multimodal MFB approach employs trainable and frozen embeddings to harness pre-trained knowledge and feature adaptability from the refined feature sets after feature optimization. This multimodal approach synergizes recurrent layers for intricate feature extraction, augmented by attention mechanisms to focus on crucial parts of the data sequence. At the beginning, the input text sequences are transformed into word indices.

$$\mathrm{x} = (\mathrm{x}_1, \mathrm{x}_2, \cdots, \mathrm{x}_i, \cdots, \mathrm{x}_n), i = 1, 2, 3, \cdots, n \tag{8}$$

where x_i is the index of the i_{th} word.

We utilize two distinct embedding layers: Embd1 (trainable, initialized with pre-trained GloVe word embeddings) and Embd2 (non-trainable, frozen to preserve GloVe embeddings). These layers convert word indices into word embeddings (Rasool et al., 2021). The embedding layer uses the following formula to compute the embedding vector for a word:

$$u_i = W \times v_i \tag{9}$$

where u_i is the embedding vector for word i, W is the embedding matrix, v_i is the word vector for word i.

We use a pre-trained GloVe embedding matrix $W_g \in \mathbb{R}^{|V| \times e}$, where V is the vocabulary and e is the embedding dimension. to prepare data for the fusion generalization model. First, the vocabulary is loaded, and the GloVe vectors of the words in the vocabulary are retrieved; then, the embedding matrix is created by initializing it with random values; finally, the text data is padded to equal lengths and shuffled to train the model. Then we get the text matrix $\mathrm{T_e} \in \mathbb{R}^{\mathrm{n} \times \mathrm{p} \times \mathrm{e}}$, where n is the number of tweets or news, p is the sequence padding length. Gaussian noise is added to the embeddings to introduce regularization and likely improve generalization. Spatial dropout is a regularization technique that differs from standard dropout, which randomly discards individual neurons rather than the entire channel. By randomly discarding the entire channel, the correlation between features in the network is reduced, which helps to prevent overfitting. Usually, neural networks may overly rely on specific features during training, leading to overfitting. By randomly removing the entire feature channel through Spatial dropout, the network is propelled to focus on different features on all channels, thereby reducing dependence on a single feature.

MFB employs two sets of parallel feature extraction layers with varying kernel sizes (2 and 3) to detect diverse n-gram patterns within the multimodal data. The layers use the following formula to compute the output:

$$y = f(W \times x + b) \tag{10}$$

where y is the output, f is the activation function "ReLu", W is the kernel, x is the input, b is the bias.

Each group consists of two feature extraction layers with the same kernel size, followed by max pooling Pooling $= \{GlobalMax1D, GlobalAvg1D\}$ for dimensionality reduction (Kamyab et al., 2022). After the max-pooling layer, we obtain the feature map p_n.

$$p_n = Max[h_n], n = 1, n = 2 \tag{11}$$

The outputs of the max-pooling layer are concatenated and fed into a bidirectional LSTM or GRU layer ($\overrightarrow{br}$and$\overleftarrow{br}$), capturing long-range dependencies h_{tbr}. *AttentionWithContext* layers are applied to the outputs of both LSTM and GRU layers to focus on the most relevant parts of the sequences.

The recurrent layers use the following formula to compute the output of a recurrent layer (Basiri et al., 2021):

$$h_t = f(W_h \times h_{t-1} + W_x \times x_t + b) \tag{12}$$

$$\overrightarrow{h}_{t_{br}} = \overrightarrow{br}(p_n), n \in [1, p] \tag{13}$$

$$\overleftarrow{h}_{t_{br}} = \overleftarrow{br}(p_n), n \in [p, 1] \tag{14}$$

where h_t is the output of the recurrent layer at time t, f is the activation function, LSTM layers use the "tanh" activation function and GRU layers use the "sigmoid" activation function, W_h is the recurrent weight matrix, W_x is the input weight matrix, x_t is the input at time t, b is the bias.

MFB employs an attention mechanism, utilizing attention weights to calculate the weighted average of multimodal input sequences. The mechanism formula is as follows (Wen & Li, 2018):

$$\alpha_i = softmax(W \times h_i) \tag{15}$$

$$c = sum(\alpha_i \times h_i) \tag{16}$$

where α_i is the attention weight for input i, W is the attention weight matrix, h_i is the output of the recurrent layer at time i. where c is the attention output.

The outputs of the attention layers are concatenated and passed through a series of dense layers with dropout and batch normalization for further processing and feature extraction. The dense layers use the following formula to compute the output of a dense layer (Liu & Guo, 2019):

$$h_d = f(W_d \times x + b) \tag{17}$$

where h_d is the output of the dense layer, f is the activation function "ReLu", W_d is the dense weight matrix, x is the input, b is the bias.

Algorithm 2. Fusion model generalization for text data embedding and sequence processing.

Input: Text matrix T, the pre-trained GloVe embedding matrix W_g.
Output: Texts classification vector $\mathrm{C} = \{\mathrm{c_i} \in \{0, 1\} : \mathrm{i} \in [1, \mathrm{n}]\}$.
Construct the word embedding matrix $\mathrm{T_e}$, the feature map p_n using Eqs. (9) to (11)
branches = {Bi-LSTM, Bi-GRU}
for br ∈ branches **do**
  Apply br on $\mathrm{T_e}$ to obtain both future and preceding contexts $\overrightarrow{h}_{t_{br}}$, $\overleftarrow{h}_{t_{br}}$ using Eqs. (12) to (14)
  Construct h_{br}, $u_{t_{br}}$
  num $= \exp\left(\mathrm{u}_{\mathrm{t_{br}}}^{\mathrm{T}} u_{w_{br}}\right)$
  sum←0
  for n ∈ [1, p] **do**
    sum←sum $+ \exp\left(\mathrm{u}_{\mathrm{t_{br}}}^{\mathrm{T}} u_{w_{br}}\right)$
    $\alpha_{t_{br}} = \frac{num}{sum} S_{br} \leftarrow 0$
  for n ∈ [1, p] **do**
    $S_{br} \leftarrow S_{br} + \exp(\alpha_{t_{br}} h_{t_{br}})$ using Eqs. (15) and (16)
Apply (S_{br}, batch normalization)
Construct h_d using Eq. (17)
Feed h_d into a sigmoid function for binary classification
Update parameters of the model using the binary cross-entropy loss function with the Adam method
Return C

In our MFB approach, the final dense layer utilizes a sigmoid activation function to generate a single output, indicative of the likelihood of the text correlating with a specific class in the multimodal dataset. The model was compiled using binary cross-entropy loss and the Adam optimizer with learning rate "*lr*" and learning rate decay "lr_d". We adjust these parameters to achieve the best performance of our proposed MFB approach.

Additionally, validation splits are implemented to monitor performance on unseen data during training. We selected 10 % of the data for validation, and by testing the model on the validation set, we can choose the optimal model configuration and adjustment strategy to avoid the model overfitting on the training set. This synthesis method in MFB is encapsulated in Algorithm 2.

Capitalizing on the refined multimodal feature sets and the advanced generalized multimodal fusion, our approach to forecasting Bitcoin prices for the next hour strategically applies transfer learning techniques. Utilizing the pre-trained weights of the MFB, we explore two primary methods: freezing certain layers of the MFB to retain learned patterns and fine-tuning the entire model to adapt it more closely to Bitcoin price prediction. Fine-tuning can be mathematically expressed as (Xu, et al., 2109):

$$\theta_{fine-tuned} = argminL(\theta, D_{train}) + \lambda R(\theta) \tag{18}$$

where θ are the weights of the pre-trained model, L is the loss function, D_{train} is the training dataset, *argmin* is the value of the argument that minimizes a given function, R is the regularization term, and λ is the regularization parameter. Fine-tuning aims to minimize the loss function while preventing overfitting by adding a regularization term.

Our multimodal approach's input features include text sentiment and technical indicators of Bitcoin prices, reflecting market sentiment and trends, respectively. The time-series input data is processed via the BiLSTM layer and then gets the output, $\mathrm{BiLSTM_i} = \left\{\mathrm{blstm}^i_1, \mathrm{blstm}^i_2, \cdots, \mathrm{blstm}^i_t, \cdots, \mathrm{blstm}^i_n\right\}$, which can capture the before and after dependencies in sequence data. The output is then further processed through a bidirectional GRU layer, and the output of the BiGRU layer is $\mathrm{BiGRU_j} = \left\{\mathrm{bgru}^j_1, \mathrm{bgru}^j_2, \cdots, \mathrm{bgru}^j_t, \cdots, \mathrm{bgru}^j_n\right\}$, which is capable of capturing contextual dependencies in sequence data. Pre-trained embeddings from the MFB are used to initialize the embedding layers, ensuring that the semantic insights from the general text are transferred to our specific domain of Bitcoin price prediction. In the recurrent layers, weights are initialized from the pre-trained MFB, enabling the transfer of learned sequence processing patterns.

The final step involves a fully connected layer employing linear regression to output the predicted Bitcoin price. This layer integrates the insights gained from the lagged data correlation and the feature optimization process to produce a nuanced and accurate prediction for the next hour's Bitcoin price.

Building on this robust framework, we initiate the model training by partitioning the dataset into training and testing sets, respectively, for training and evaluating the model's generalization ability, focusing on the Bitcoin closing price as our target variable. To standardize the data, we apply a zero-base normalization, scaling the values within a [0, 1] range for consistency. Subsequently, the data is segmented into windows, each spanning 5 h, to align with our window_len parameter.

Algorithm 3. Next-hour Bitcoin price prediction using MFB layers with fine-tuning.

Input: Bitcoin Technical Indicators and Sentiment scores X, Bitcoin Closing Price Y
Output: Predicted Bitcoin Closing Price $\widehat{Y}$
Data_windows ← windows (Y, sequence_length)
Normalised_data = []
for k **in** Data_windows **do**
  for i **in** p **do**
    Normalised_window =(X^i/X^i.iloc[0]) − 1
    Normalised_data.append (Normalised_window)
row ← 80 % of the shape of Normalised data
X_train, y_train ← [: row,: −1], X_test, y_test ← [row:,: −1]
Set epoch_size: N = 50
for epoch **in** 1 to N **do**
  Select M training sample with training size with the random state.
  for i **in**1 to M **do**
    Feed training sample to BiLSTM and get the output $\mathrm{BiLSTM_i}$
    Connect BiGRU layer with the output from BiLSTM layer $\mathrm{BiGRU_j}$
    Connect dense layer and sigmoid function with the dropout output
    Model.fit (X_train, y_train, validation_data = (X_test, y_test), epochs, batch_size)
    Model.compile (loss ='MSE', optimizer='Adam', metrics = [MAE, MSE, RMSE, R^2, MAPE])
Return $\widehat{Y}$

This structuring is pivotal for transforming the data into a format conducive to model input. Training is conducted using the *Model.fit (X_train, y_train, validation_data = (X_test, y_test), epochs, batch_size)* function, where X_train and y_train supply the training data and labels, and validation_data provides the testing equivalents. The epochs define the number of training cycles, while batch_size dictates the volume of data processed in each step. Notably, we shuffle the entire training dataset at each epoch's onset, dividing it into several batches per the batch-size specification. Fig. 3 shows the preparation and training process that underpins our next-hour Bitcoin price prediction capability, as further elaborated in Algorithm 3.

5. Experiment and results

The experimental workflow of the MFB, aligning with our multimodal fusion approach and detailed in the results section, is concisely outlined in the subsequent steps, which are also depicted in Fig. 4:

- *Data Collection*: Gather Bitcoin prices, financial tweets, and news data.
- *Preprocessing*: Apply various preprocess techniques for structured data and apply VADER to extract sentiment scores.
- *Lagged Data Correlation*: Correlate sentiment scores with Bitcoin price to identify temporal relationships.
- *Fusion Model*: Process the data through a sequence of layers within the model, including embedding, recurrent with attention, and dense layers.
- *Feature Optimization*: Utilize correlation matrix filtering, BorutaShap feature selection, and SPB optimization to refine model features.
- *Model Training*: Compile and train the model, tuning it to optimize performance.
- *Performance Evaluation*: Assess the model's effectiveness using various metrics such as recall, precision, F1 score, and accuracy.
- *Bitcoin Price Forecasting*: Deploy the trained model to predict the next hour's Bitcoin price, with outcomes evaluated against metrics like MAE, MSE, R2, RMSE, and MAPE.

The details of these substantive steps will be clearly outlined in the

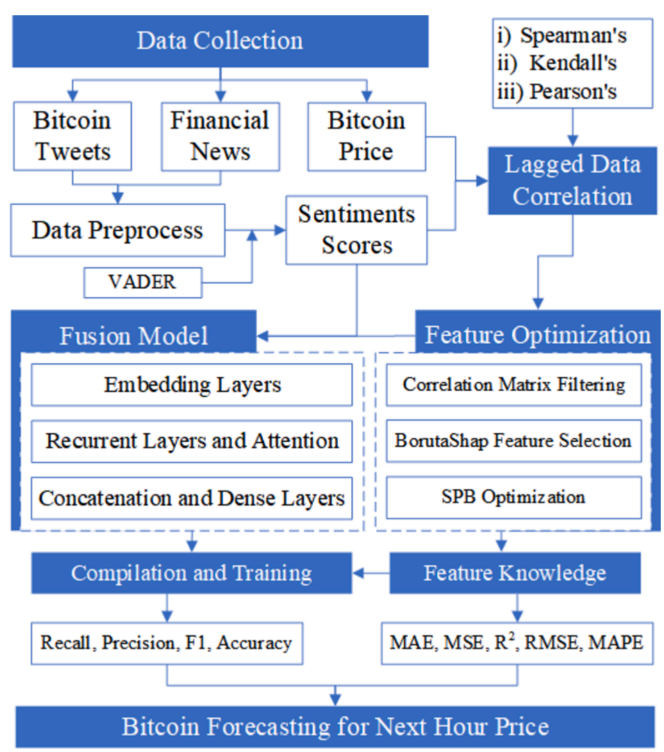


Fig. 4. Experimental flowchart of the proposed multimodal fusion approach for next-hour Bitcoin price prediction.

following subsections. The experiment was conducted on an Ubuntu 20.04 system in a high-performance environment equipped with a 2.60 GHz Intel Xeon Gold 6240 CPU and 256 GB of running memory. The graphics card model was NVIDIA GeForce RTX 2080Ti SUPER. In addition, the experiment was conducted in Python 3.8.13v language, utilizing TensorFlow and Keras frameworks to assist in the development and operation of neural networks and utilizing many packages and libraries such as NumPy, Pandas, Matplotlib, and Seaborn for data processing, visualization, and model evaluation.

5.1. *Correlation analysis*

The sentiment analysis conducted using VADER on news and tweets data provides an explicit dichotomy in sentiment distribution as illustrated in Fig. 5. News data exhibits a high positive sentiment of 74.23 %, and tweets data presents a near-equal split between positive (43.70 %) and negative (42.23 %) sentiments, with a smaller neutral sentiment at 14.07 %. This variation is critical to our model as it may influence the predictive accuracy for Bitcoin prices, considering that sentiment is a significant driver of market behavior, which can reflect investors' expectations.

The balanced sentiment in tweets points to a divided public opinion and potential market volatility, whereas the predominantly positive sentiment in the news may suggest a bullish market trend.

5.1.1. *Between Bitcoin price and text data*

A correlation analysis between Bitcoin price and sentiment score derived from text data is depicted in Fig. 6. Utilizing the Pearson correlation coefficient via the Pandas *corr*() method, we observe that Bitcoin's price movements tend to align with shifts in sentiment scores.

Generally, a positive relationship is evident, where increases in sentiment scores coincide with rising Bitcoin price rates. This pattern suggests a solid link between market sentiment and Bitcoin's valuation trends.

The cross-correlation patterns between Bitcoin price fluctuations and sentiment scores from textual data are exhibited in Fig. 7 using three correlation metrics. Fig. 7(a) provides a measure of linear relationship, while Fig. 7(b) and (c) are based on rank correlation and offer a broader understanding of the dependency structure. The positive correlation across metrics underscores sentiment analysis as a credible predictive element in our MFB model. The correlation peaks among all metrics at lag = 0, suggesting that the sentiment and Bitcoin price changes are synchronized; shifts in Bitcoin prices immediately mirror changes in sentiment scores. This synchronization underscores the real-time impact of market sentiment on Bitcoin prices, a key consideration for our proposed MFB approach. The real-time correlation confirmed by the analysis substantiates our hypothesis that properly processed sentiment data is a potent predictor of market behavior. Changes in sentiment can lead to changes in investor behavior, thereby affecting the Bitcoin price trend. This reinforces its integration into our MFB to boost the model's accuracy in forecasting Bitcoin prices.

5.1.2. *Among technical indicators*

Furthering our analysis from sentiment correlation, Fig. 8 highlights the relationships within Bitcoin's trading metrics. It confirms a strong linear correlation between the 'Close' price and Open, High, and Low prices, establishing the closing price as a critical predictor for our model. This relationship underscores the closing price's reflection of market trends within the trading cycle.

The Volume's correlation with the 'Close' price, characterized by dense clustering at lower volumes and notable outliers at higher volumes, suggests a nuanced impact on price changes, meriting further examination. These patterns and sentiment insights inform the data-driven sophistication of our proposed MFB approach.

The correlation matrix in Fig. 9 reveals significant relationships among the technical indicators used for Bitcoin price prediction. Notably, a high correlation is observed between the Stochastic and Williams indicators and between the EMA and the Bollinger Bands (Hong et al., 2022), as indicated by the lighter shades in the heatmap. After refining the indicator set through Pearson correlation analysis to minimize redundancy, the number of technical indicators was pared down to 7, as shown in Fig. 10(a). This revised matrix illustrates significantly reduced inter-feature correlation, facilitating a clearer analysis. In Fig. 10(b), the filtered technical indicators display a varied correlation with the 'Close' feature. Specifically, the RSI shows a noticeable correlation with the 'Close' price, suggesting its utility in capturing momentum that may influence the closing price of Bitcoin.

This level of correlation is essential to ensure that our MFB model leverages the most informative and independent predictors to enhance its forecast accuracy for Bitcoin's closing price.

5.2. *Feature validation*

Continuing from the refined correlations and patterns identified, we applied the BorutaShap algorithm and SPB optimizer, as introduced in our proposed methodology, to discern each technical indicator's importance further.

Table 3 illustrates the outcomes of the BorutaShap and SPB optimizer feature selection process. The Average Feature Importance (AFI) and Standard Deviation Importance (SDI) were calculated for each feature, and a decision-making step to accept or reject the feature based on its Z-score was calculated as the AFI divided by the SDI.

The Z-score can be used to evaluate the importance of features. The higher the Z-score, the greater the importance of the feature; therefore, it can be considered a more influential feature. The Z-score can help determine which features are much more important than random

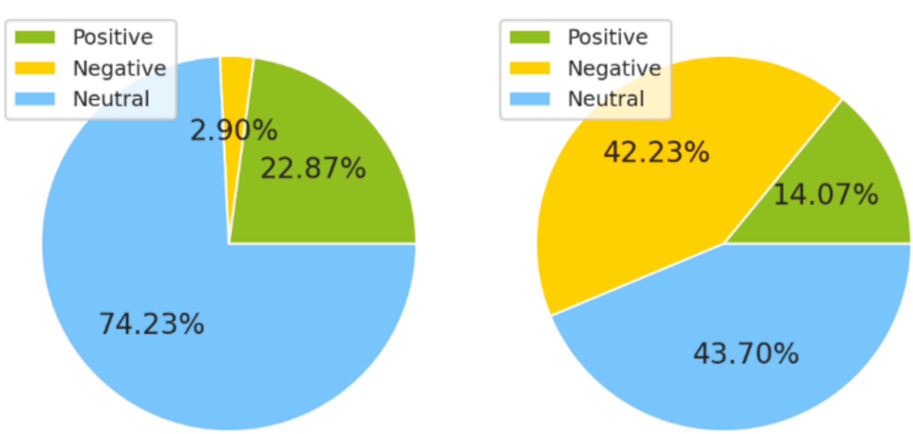


Fig. 5. Sentiment distribution in news and tweets via VADER analysis.

Fig. 6. Correlation of Bitcoin price and sentiment score derivatives.

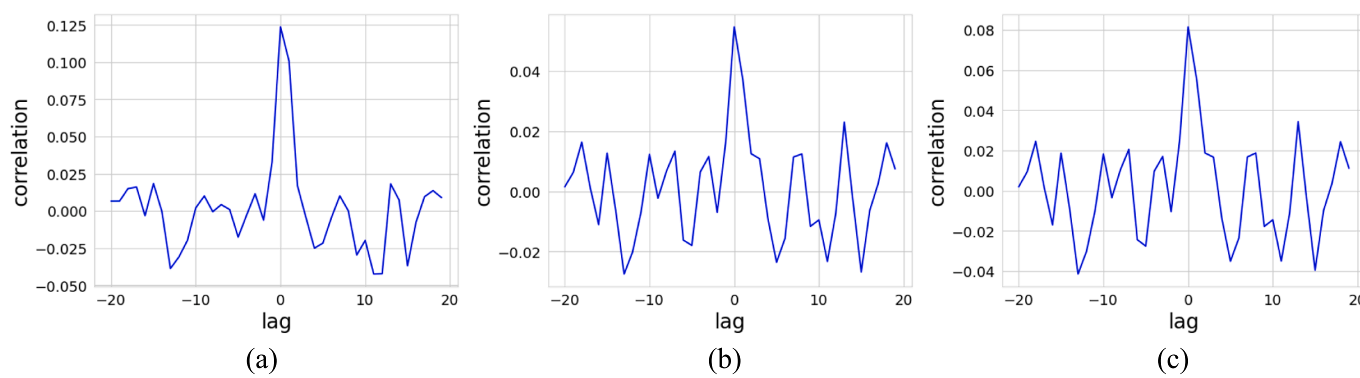


Fig. 7. Cross-correlation analysis between text data and Bitcoin price with lagged variables, (a) Pearson cross-correlation, (b) Kendall cross-correlation, (c) Spearman cross-correlation.

features. All seven features were deemed 'Accepted', underscoring their relevance in predicting Bitcoin prices. Comparing the results in Table 3, the EMA indicator stands out as it scores much higher than the other indicators, further emphasizing the importance of EMA in the model. These analyses validate our feature selection and confirm the utility of each technical indicator. These indicators, now rigorously vetted, provide a robust foundation for our Bitcoin price prediction model.

5.3. MFB price prediction on Bitcoin

Due to the significance of the validated features of technical indicators, we advanced to Bitcoin price prediction to assess the forecasting ability of our multimodal fusion Bitcoin (MFB) approach. We partitioned the data into a 90/10 split for training and testing, utilizing random sampling to maintain representativeness. The training set was employed to fit the model, while the test set served to assess its performance.

Key hyperparameters, such as learning rate and batch size, were fine-tuned, as detailed in Table 4, to optimize the training process. The division of the dataset is depicted in Fig. 11, illustrating the temporal split used for training and testing the model. We set the ratio of the training and testing sets to 8:2, which helps ensure that the model uses different datasets during the training and testing processes, providing an objective evaluation of the model's performance. The 'window_len' parameter was set to 5, providing the model with a compact yet comprehensive view of temporal dependencies to predict future Bitcoin prices effectively.

Table 5 illustrates the architecture of the BRNN model. The model comprises two bidirectional recurrent neural network layers with 512 hidden units. A 20 % dropout layer is added afterward to prevent overfitting. All RNN units use linear functions as activation functions. The model was trained for 50 epochs, with a batch size of 8 samples per epoch. The model was trained using the mean squared error (MSE) loss function. In this model, the fully connected layer has only one output, so it converts the output of the hidden units into a prediction value. During the model optimization, we conducted a thorough comparative analysis using five different optimizers (Aslam et al., 2401)—Adam, AdaDelta, RMSprop, AdaGrad, and SGD—to enhance the prediction accuracy of our MFB model for Bitcoin price forecasting. This comparison, conducted through three iterative runs to ensure reliability, aimed to identify the optimizer that most effectively minimized prediction errors. The Adam optimizer demonstrated the lowest MAE and MSE, indicating highly accurate and consistent predictions.

Furthermore, it achieved the highest R^2, signifying a robust fit to the Bitcoin price data, and the lowest RMSE and MAPE, confirming the precision of the model. In contrast, the alternative optimizers showed relatively poorer performance, as evidenced by some metrics displaying negative R^2 values. An analysis of the R^2 calculation formula provided in Supplementary S2 reveals that a negative R^2 value fundamentally indicates the model's failure to capture the underlying trend of the data. Consequently, the predictive power of the model is inferior to that of simply using the mean of the observed values for all predictions. This is likely due to the optimizers' less effective handling of the complex volatility and noise inherent in the Bitcoin price series.

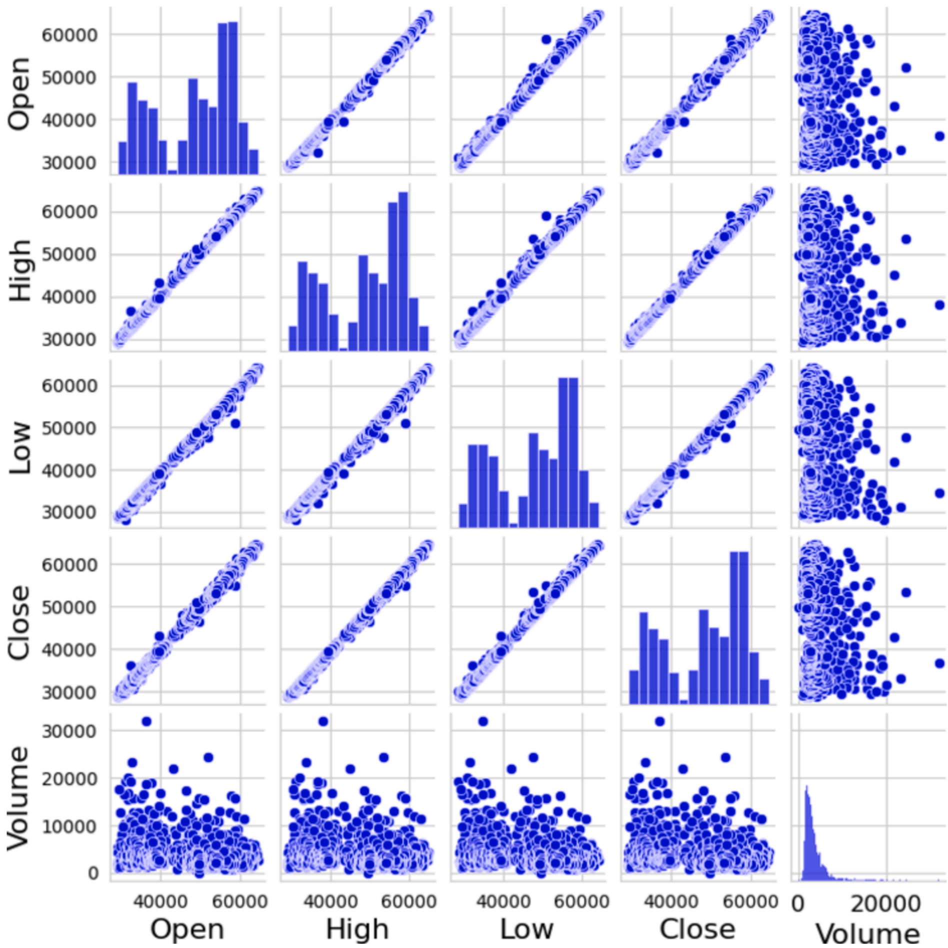


Fig. 8. Correlation analysis of market dynamics and volume impact on Bitcoin trading parameters.

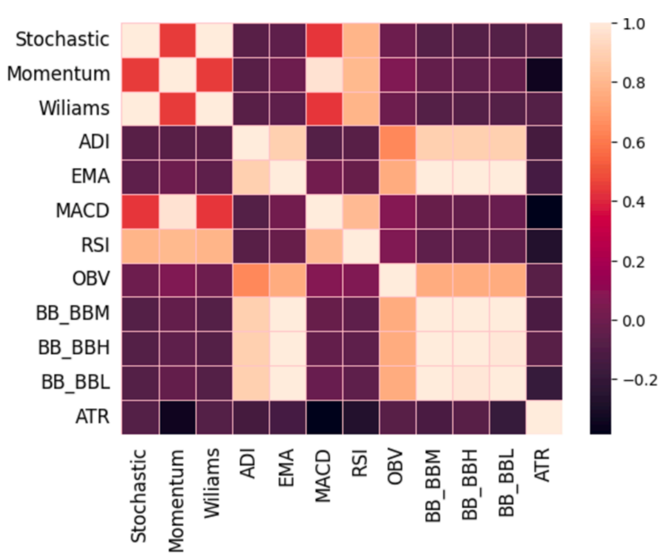


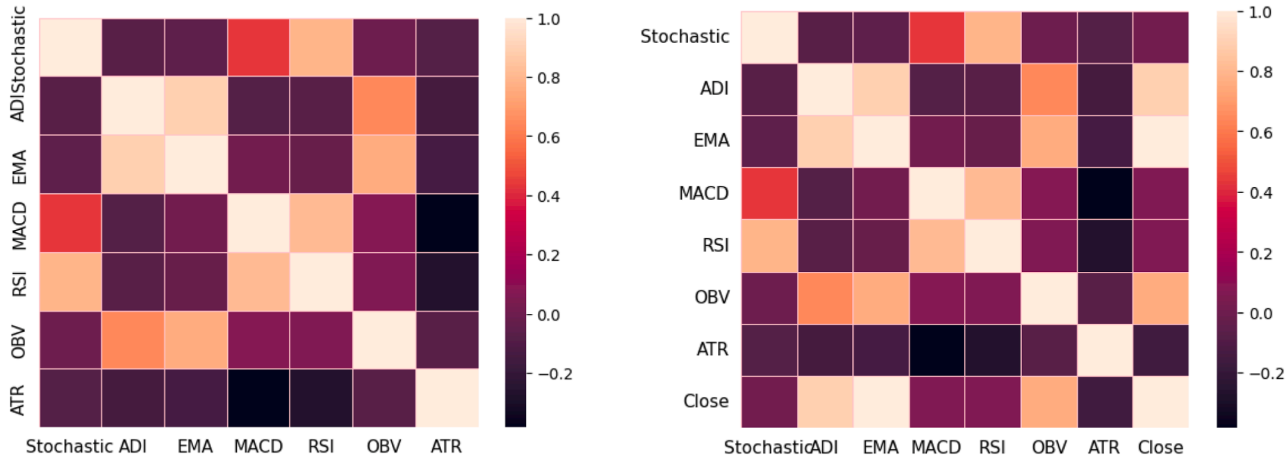
Fig. 9. Correlation matrix of the 12 technical indicators on Bitcoin.

These optimizers did not converge as efficiently to minimize errors nor capture the nuanced patterns present in the dataset as adeptly as the Adam optimizer. Its superior performance can be attributed to its adaptive learning rate mechanism, which excels in adjusting to the volatile cryptocurrency market conditions. The optimizer's fine-tuned results, detailed in Table 6 and visualized in Fig. 12, underscore its effectiveness and have led us to select it as the preferred optimizer for training our predictive model. With Adam, our MFB model is expected to offer more reliable and precise Bitcoin price predictions, an essential tool for stakeholders in the cryptocurrency domain.

Fig. 13 provides a visual evaluation of the predictive performance of the MFB in forecasting Bitcoin prices. Fig. 13(a) demonstrates the MFB's ability to closely track the actual Bitcoin prices, indicating a robust predictive performance of the model. The congruence between actual and forecasted prices suggests that the multimodal fusion Bitcoin (MFB) approach has successfully deciphered the foundational patterns and trends in the Bitcoin market. In Fig. 13(b), we delve deeper into the model's predictive accuracy by comparing actual prices with one-hour-ahead forecasts derived from the training and testing phases.

The close alignment of predicted and actual prices during the testing phase, where the data was previously unseen, highlights the MFB's multimodal generalization proficiency and ability to provide reliable short-term Bitcoin price forecasts. The MFB model's capacity to accurately forecast Bitcoin prices for the upcoming hour is crucial for real-world applications like algorithmic trading, where the multimodal approach enhances the timeliness and precision of predictions, driving effective investment strategies.

Together, Fig. 13 validates the effectiveness of the MFB in not only fitting the historical data but also in predicting future price movements. The multimodal MFB model's precise tracking and short-term price forecasting capabilities are bolstered by incorporating time-lagged sentiment data, enriching its understanding of dynamic market sentiments. Feature optimization refines the model's inputs for impactful predictions, while correlation analysis clarifies the interplay of market indicators, solidifying the predictive base.



(a) (b)

Fig. 10. Correlation matrix among (a) the filtered seven technical indicators and (b) between the filtered seven technical indicators and the 'Close' feature.

Table 3
Evaluation of feature relevance for Bitcoin price prediction using BorutaShap and SPB algorithms.

Features	AFI	SDI	Decision
EMA	3.602684639	3.74E-05	Accepted
RSI	−0.158810831	0.001813246	Accepted
Stochastic	−0.216105369	0.001648946	Accepted
OBV	−0.260812917	0.001810746	Accepted
ADI	−0.283823426	0.001028633	Accepted
MACD	−0.285457331	0.000846852	Accepted
ATR	−0.29253772	0.000629814	Accepted

Table 4
Hyperparameters utilized in model training.

Hyperparameter	Value	Hyperparameter	Value
lr	1e-3	np.random.seed	42
lr_d	1e-10	window_len	5
units	128	zero_base	True
spatial_dr	0.5	neurons	256
kernel_size1	2	epochs	50
kernel_size2	3	batch_size	8
dense_units	32	loss	MSE
dr	0.2	dropout	0.2
conv_size	32	optimizer	Adam

Fig. 14 offers a comparative analysis of diverse neural network models concerning actual Bitcoin price movements, underscoring the performance of our proposed multimodal fusion Bitcoin (MFB) approach. It reveals that while traditional models like the GRU and Bi-GRU show noticeable deviations from the actual price trends, our MFB model maintains a closer trajectory to the practical Bitcoin prices. This comparison underscores the MFB's advanced capability to model and predict complex patterns within the cryptocurrency market. The MFB's accurate predictions, closely mirroring market trends, indicate its efficacy in capturing temporal market dynamics. This fusion model outperforms tested models, offering investors and analysts reliable short-term market forecasts.

Fig. 15 captures the training and validation loss trajectories across various neural network architectures employed in our study, including LSTM, GRU, Bi-LSTM, Bi-GRU, and our proposed MFB. The initial decrease in both training and validation loss across the models indicates effective learning from the data, while the stability of the validation loss in the MFB model points to its robust generalization capabilities. Notably, the MFB's validation loss (Fig. 15(e)) does not show the upswing trend in the other models, suggesting it is better at avoiding overfitting. This characteristic of the MFB, evidenced by its steady validation loss curve, emphasizes the model's potential to accurately capture the Bitcoin market's underlying patterns for reliable price prediction.

Table 7 presents the averaged results of ten experimental runs, comparing the performance indices of various models. The MFB model demonstrates the lowest MAE at 0.0065 and the lowest MSE at approximately 8.29e-05.

These values indicate that, on average, our model has the least prediction error and can more accurately capture and reflect market changes and trends. Additionally, the highest R^2 value of 0.7377 shows that our MFB approach can explain a more significant proportion of the variance in Bitcoin prices than its counterparts. Moreover, the RMSE is the lowest at 0.0091, highlighting the model's accuracy in predicting Bitcoin prices. Although the LSTM model shows a slightly better MAPE at 2.5983, the overall performance of the MFB is more robust across the other key metrics, making it the preferred model for forecasting Bitcoin prices accurately.

To validate the robustness of the MFB model, we conduct a comparative experiment using the GroupTimeSeriesSplit (GTSS) library, designed for performing cross-validation on time series data while respecting the temporal order and group dependencies within the datasets. This library is useful for datasets where traditional cross-validation techniques might lead to data leakage or fail to account for the inherent temporal structure. In our experiment, we set the parameter "test_size".

to a value of 562, specifying that each test set will contain 562 samples, and the value of "n_splits" to 500, which means that cross-validation will be performed 500 times to create 500 different training and testing partitions. This extensive segmentation helps to obtain robust estimates of the model's performance by exposing the model to various training and testing scenarios.

As presented in Table 8, the comparison of MFB performance metrics with and without GTSS indicates that the performance of the MFB model is consistent across different datasets. It has good generalization ability and robustness, highlighting its reliability in practical applications of time series prediction.

5.4. Baseline comparison

The performance of the MFB model relative to other state-of-the-art methods on the same datasets utilized in this study is shown in Table 9.

MFB exhibits a notably high accuracy of 81.52 % on the news and an even more impressive 97.63 % on tweet data. The superior performance

Fig. 11. Training and test dataset split for Bitcoin price prediction.

Table 5
The architecture of the bidirectional recurrent neural network model.

Layer (type)	Output Shape	Param #
bidirectional_2 (Bidirectional)	(None, 5, 512)	538,624
dropout_5 (Dropout)	(None, 5, 512)	0
bidirectional_3 (Bidirectional)	(None, 512)	1,182,720
dropout_6 (Dropout)	(None, 512)	0
dense_5 (Dense)	(None, 1)	513

Table 6
Comparative analysis of optimizer performance.

Optimizer	MAE	MSE	R^2	RMSE	MAPE
AdaDelta	0.0127	0.0004	−0.1462	0.0190	3.0751
RMSprop	0.0090	0.0002	0.2966	0.0149	4.9069
AdaGrad	0.0098	0.0002	0.3487	0.0143	3.3099
SGD	0.0108	0.0003	−0.0944	0.0186	4.2679
Adam	**0.0065**	**9.1855e-05**	**0.7094**	**0.0096**	**2.9546**

with tweet data could be due to MFB's sophisticated architecture, adept at handling the platform's informal and concise content. However, the performance of news data is slightly lower, which could be due to the complex and formal nature of such texts, often requiring a deeper understanding of context and language nuances that present a more significant challenge for any text analysis model. This is a common trend across models, which roughly perform less effectively on news data.

Compared to the T-BiLSTM-CNN model, the MFB shows an improvement in accuracy of approximately 4.47 % on news data and 8.17 % on Tweet data. This substantial outperformance is likely due to MFB's fusion approach, which combines bidirectional LSTM and GRU layers, enabling it to capture both forward and backward dependencies in data more comprehensively. The effectiveness of the proposed MFB multimodal in accurately predicting Bitcoin prices can be linked to its ability to process and learn from time-lagged sentiment data and correlations within the market. Therefore, the outperformance of MFB not only marks an advancement in text analysis but also contributes to more accurate and timely predictions in the volatile Bitcoin market.

Fig. 16 shows the ROC curves of MFB and other models, emphasizing MFB's higher AUC values for both (a) news and (b) tweets data. MFB's superior ability to discern sentiment, credited to its bidirectional LSTM and GRU layers, and attention mechanism, drives its performance. Particularly for tweets, the MFB's proficiency likely benefits from the abundant and varied data, enhancing sentiment classification and

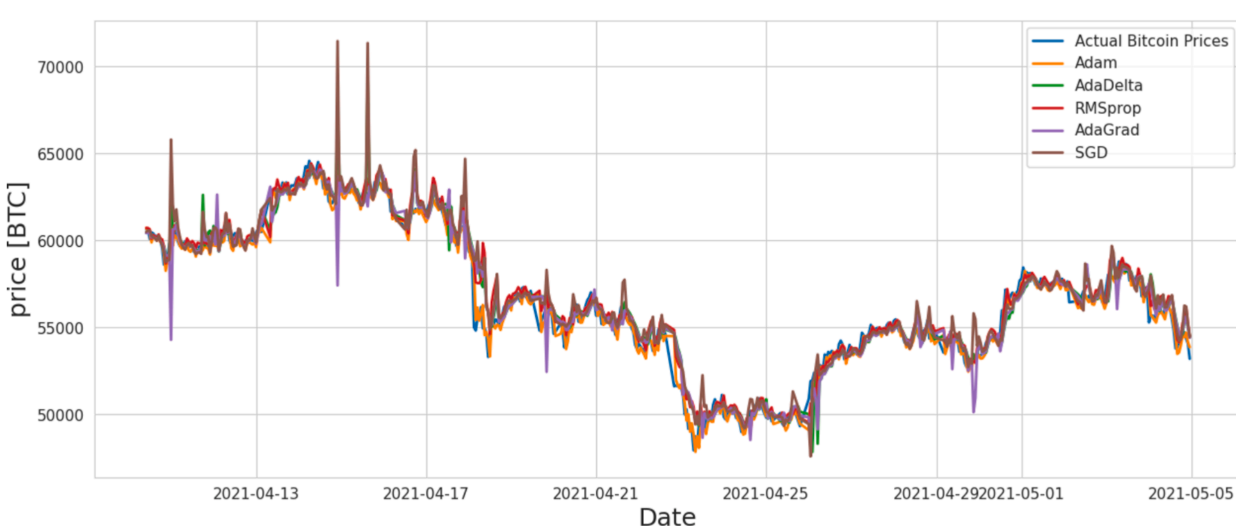


Fig. 12. Predictive analysis visualization with various optimizers.

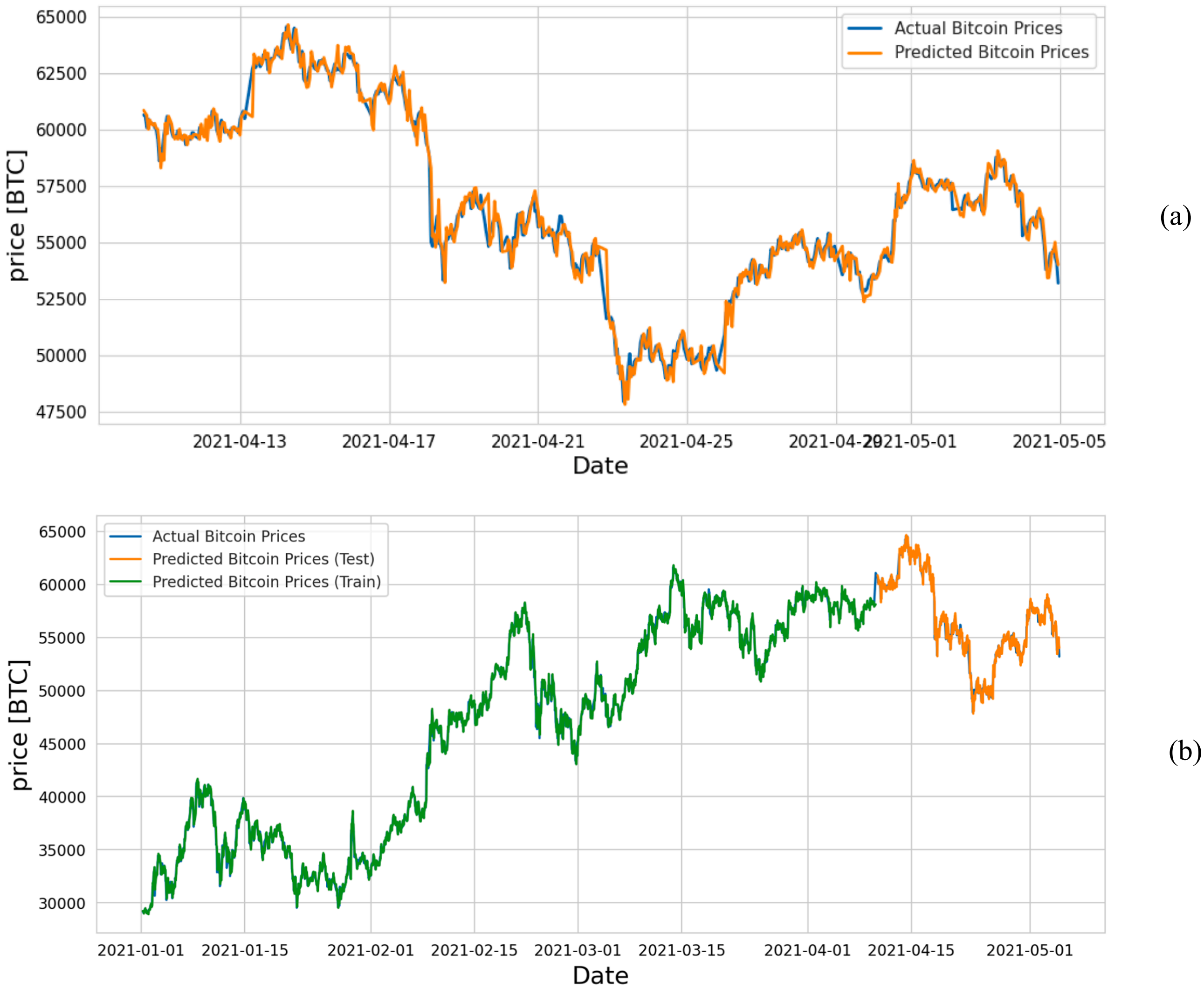


Fig. 13. MFB's prediction (a) actual vs. predicted Bitcoin prices, (b) comparison of actual Bitcoin prices with next-hour predictions.

Fig. 14. MFB prediction performance comparison with different models.

Bitcoin price prediction effectiveness. This success is also supported by the model's adept use of time-lagged sentiment data, feature optimization, and correlation analysis, reinforcing MFB's role as a potent tool for financial forecasting.

To further demonstrate the effectiveness and accuracy of the MFB model in the real-time prediction of Bitcoin prices, we select 8 price

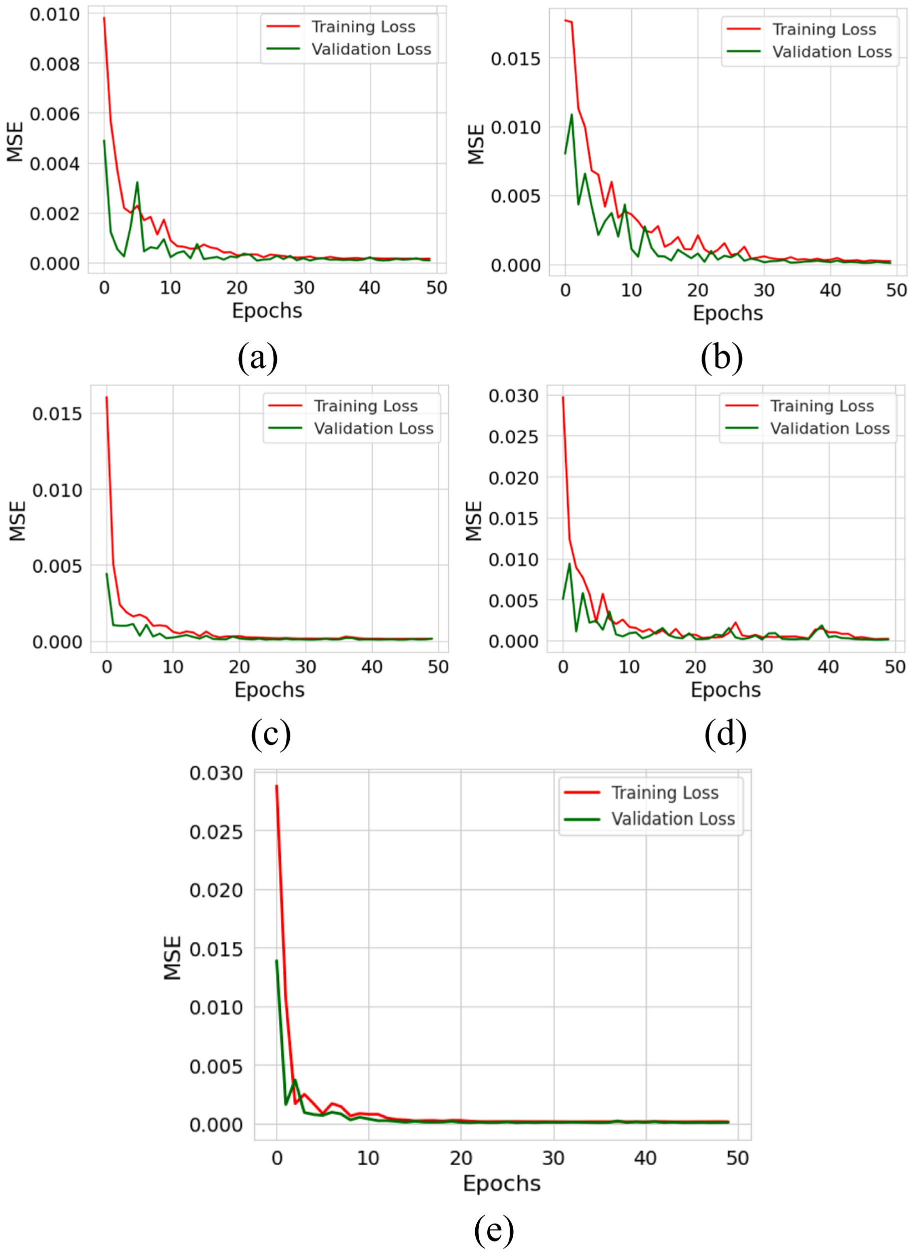


Fig. 15. Loss trends during model training for (a) LSTM, (b) GRU, (c) Bi-LSTM, (d) Bi-GRU, and (e) proposed MFB approach.

Table 7
Comparative performance metrics of MFB variants.

Metrics	LSTM	GRU	Bi-LSTM	Bi-GRU	MFB
MAE	0.0070	0.0067	0.0070	0.0071	**0.0065**
MSE	9.6547e-05	9.2945e-05	0.0001	0.0001	**8.2925e-05**
R^2	0.6946	0.7060	0.5992	0.6787	**0.7377**
RMSE	0.0098	0.0096	0.0113	0.0101	**0.0091**
MAPE	**2.5983**	3.0445	2.7636	2.8028	2.7157

Table 8
Comparison of performance metrics for MFB with and without GTSS.

Metrics	MAE	MSE	R^2	RMSE	MAPE
MFB with GTSS	0.0083	0.0001	0.7226	0.0093	2.6673
MFB without GTSS	0.0065	8.2925e-05	0.7377	0.0091	2.7157

prediction models from 2020 to 2024 as baseline models for comparison, test them using our dataset, and evaluate them using three metrics: MAE, MSE, and RMSE. The results of the MFB are shown in Table 10. From the results, it can be seen that the MFB model has an MAE of 0.0065, an MSE of 8.2925e-05, and an RMSE of 0.0091, surpassing recent research and underscoring the robustness and accuracy of the MFB model in real-time Bitcoin price prediction.

Compared with our proposed MFB model, the shortcomings of other methods can be attributed to several factors. Firstly, many of these methods lack social media data in their datasets, which is crucial for capturing market sentiment and trends in the cryptocurrency field. By applying time-lagged sentiment analysis, our model leverages valuable insights from social media to improve its predictive accuracy. Secondly, a significant drawback of other methods is the lack of appropriate feature engineering and optimization techniques. Our MFB model integrates the BorutaShap algorithm for feature optimization, enabling it to recognize and utilize the most relevant features for prediction. This strategic approach ensures that the model is not overwhelmed by

Table 9
Performance of MFB compared to baseline methods.

Methods & Ref.	Year	Data	Class	Recall	Precision	F1	Accuracy
ARC (Wen & Li, 2018)	2018	News	N	0.1220	0.2941	0.1724	0.7377
			P	0.9155	0.7831	0.8442	
		Tweets	N	0.9475	0.9415	0.9445	0.9361
			P	0.9209	0.9289	0.9249	
AC-BiLSTM (Liu & Guo, 2019)	2019	News	N	0.1707	0.3501	0.2295	0.7432
			P	0.9085	0.7914	0.8459	
		Tweets	N	0.9387	0.9077	0.9229	0.9101
			P	0.8718	0.9137	0.8922	
GloveRNNCNN (Abid et al., 2019)	2019	News	N	0.2683	0.3793	0.3143	0.7377
			P	0.8732	0.8052	0.8378	
		Tweets	N	0.9711	0.9746	0.9728	0.9689
			P	0.9660	0.9614	0.9637	
Word Embedding CNN (Alharbi & de Doncker, 2019)	2020	News	N	0.0976	0.4444	0.1739	0.7923
			P	**0.9930**	0.7921	0.8812	
		Tweets	N	0.9567	0.9515	0.9541	0.9472
			P	0.9345	0.9415	0.9379	
ABCDM (Basiri et al., 2021)	2021	News	N	0.1207	0.2531	0.2165	0.7417
			P	0.9185	0.7764	0.8738	
		Tweets	N	0.9707	0.9762	0.9735	0.9697
			P	0.9683	0.9609	0.9646	
ACR-SA (Kamyab et al., 2022)	2022	News	N	0.2927	0.4286	0.3478	0.7541
			P	0.8873	0.8129	0.8485	
		Tweets	N	0.9691	0.9804	0.9747	0.9712
			P	0.9740	0.9592	0.9665	
T-BiLSTM-CNN (Shukla & Kumar, 2023)	2023	News	N	0.1220	0.4545	0.1923	0.7705
			P	0.9577	0.7907	0.8662	
		Tweets	N	0.9303	0.8907	0.9101	0.8946
			P	0.8467	0.9004	0.8728	
Proposed MFB	2024	News	N	**0.3590**	**0.6087**	**0.4516**	**0.8152**
			P	0.9379	**0.8447**	**0.8889**	
		Tweets	N	**0.9748**	**0.9832**	**0.9790**	**0.9763**
			P	**0.9782**	**0.9674**	**0.9727**	

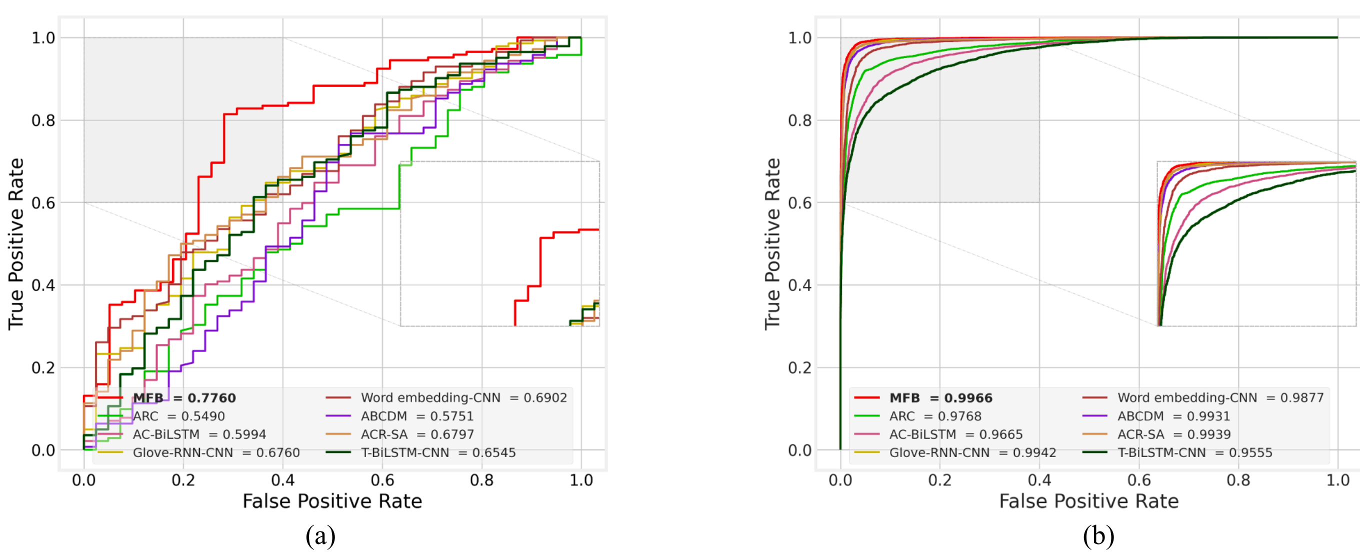


Fig. 16. Performance comparison of ROC curve of different methods on (a) News data and (b) Tweets data.

irrelevant or redundant data, thus improving its performance. Finally, the dependence on unidirectional LSTM and GRU layers in certain methods limits their ability to capture complex temporal patterns effectively.

By contrast, our MFB model includes bidirectional layers that enable it to learn from past and future information simultaneously. This bidirectional architecture enables our model to understand the context and correlation in the data and has excellent predictive performance compared to methods that only use unidirectional layers. The MFB fusion of BiLSTM and BiGRU layers, along with the BorutaShap algorithm for feature optimization, has significantly enhanced its accuracy in predicting Bitcoin prices, making it potent in the fluctuating cryptocurrency market.

6. Conclusion & future work

Our study presents MFB, a groundbreaking model in the domain of Bitcoin price prediction. The comprehensive correlation analysis between Bitcoin price and sentiment scores, as shown in Fig. 6, underscores the pivotal role of market sentiment in influencing Bitcoin's valuation. The positive correlation, especially at lag = 0 presented in Fig. 7, validates the immediate impact of sentiment on Bitcoin prices,

Table 10
Performance of MFB compared to existing methods.

Author & Ref.	Year	Method	Performance Metrics
Serafini et al. (Serafini et al., 2020)	2020	ARIMAX, LSTM-based RNN	MAE:0.247 MSE: 0.140 RMSE: 0.375
Ji et al. (Ji et al., 2021)	2021	Doc-WT-LSTM	MAE: 0.019 MSE:0.012 RMSE: 0.110
J. Hong et al. (Hong et al., 2022)	2022	Correlational strategy with GRU, LSTM, BiLSTM	MAE:0.031 MSE: 0.001 RMSE: 0.042
Ye, Z., et al. (Ye et al., 2022)	2022	Stacking ensemble deep model of 2 base models: LSTM&GRU	MAE:0.021 MSE: 0.008 RMSE: 0.089
Jakubik et al. (Jakubik et al., 2023)	2023	LSTM-CNN-RF	MAE:0.012 MSE: 0.0093 RMSE: 0.0965
Fakharchian et al. (Fakharchian, 2023)	2023	Deep Model based CNN & LSTM	MAE:0.025 MSE: 0.001 RMSE: 0.039
A. Bâra et al. (Bâra & Oprea, 2024)	2024	ELM	MAE:0.024 MSE: 0.001 RMSE: 0.031
M. S. Devi et al. (Devi et al., 2024)	2024	DQNRL	MAE:0.0247 MSE: 0.061 RMSE: 0.2469
Proposed MFB	2024	MFB-Bi-RNN	**MAE:0.0065** **MSE: 8.2925e-05** **RMSE: 0.0091**

reinforcing the importance of real-time data analysis in financial forecasting.

This correlation analysis reveals crucial relationships between the closing price and other trading variables in Fig. 8, highlighting the closing price as a vital predictor in our model. This insight and the nuanced impact of trading volume on price changes form the basis of our sophisticated predictive model. MFB closely tracks actual Bitcoin prices and excels in forecasting next-hour prices with high fidelity (in Fig. 13) – an essential capability for real-world applications like algorithmic trading. Compared to other contemporary models, MFB shows superior performance, achieving an accuracy of 81.52 % on news and an impressive 97.63 % on tweets data in Table 9. This success is due to our model's ability to process complex time-lagged sentiment data and draw insightful correlations within the market. Moreover, the MFB model's high AUC values in ROC curve comparisons illustrated in Fig. 16 and its outperformance in vital metrics like MAE, MSE, and RMSE in baseline and existing models' comparison given in Table 10 establish it as a leading model in Bitcoin price forecasting.

Our proposed approach represents a significant leap forward in Bitcoin price prediction. By effectively capturing the intricate dynamics of market sentiment and its delayed effects, MFB emerges as a highly reliable model for financial forecasting. Its innovative methodology seamlessly integrates time-lagged sentiment data with technical market indicators and sets a new standard in the FinTech domain, offering invaluable insights for investors and market analysts.

Building on our comprehensive research in Bitcoin price prediction using the MFB, future work could be expanded from three key aspects: Firstly, the MFB model was initially developed with a focus on Bitcoin price prediction, but its design has a flexible architecture that can adapt to different categories of datasets. By fine-tuning model parameters and training on specific domain data, unique challenges and characteristics of distant financial markets, including stocks, bonds, commodities, and even indices, can be addressed. Future work can extend the application of this model to other financial markets to test its versatility and robustness under different market conditions. Secondly, large language models and transformers (Wahidur et al., 2024) are known for their deep understanding of context and language nuances, which can offer more sophisticated sentiment interpretations. Introducing these enhanced sentiment analysis tools can identify the emotional driving forces behind market behavior and analyze the emotional factors behind market reactions and fluctuations more deeply, significantly improving sentiment analysis in MFB. This study mainly focuses on the emotional impact of social media without considering the influence of external market factors or unforeseeable events, such as regulatory changes and macroeconomic trends of price forecasting. Future work can further integrate larger-scale data to support model optimization and testing, which can further enhance the applicability and robustness of the model and produce better prediction results. In addition, exploring robust transfer learning or continual learning (Aslam et al., 2024) methods can further improve the adaptability and performance of the model on various financial datasets. Lastly, integrating blockchain network indicators with the MFB could offer a more comprehensive understanding of market dynamics, potentially uncovering new predictive insights for Bitcoin and other cryptocurrencies.

Funding

This work is supported by the Shenzhen Polytechnic Research Fund No. 6023310010 K, the National Key R&D Program of China under Fund No. 2021YFF1200100 and 2021YFF1200104, and the National Social Science Foundation of China under Fund No. 22BGL084.

CRediT authorship contribution statement

Ping Han: Data curation, Methodology, Software, Validation, Visualization, Writing – original draft. **Hui Chen:** Methodology, Funding acquisition, Supervision. **Abdur Rasool:** Conceptualization, Methodology, Formal analysis, Supervision, Validation, Writing – review & editing. **Qingshan Jiang:** Project administration, Validation, Supervision, Resources. **Min Yang:** Investigation, Formal analysis, Validation, Resources.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

The data and code mentioned in this article are available at https://github.com/YukikiHan/MFB.

Acknowledgments

The authors would like to thank all the anonymous reviewers for their insightful comments and constructive suggestions that have obviously upgraded the quality of this manuscript.

Appendix A. Supplementary data

Supplementary data to this article can be found online at https://doi.org/10.1016/j.eswa.2024.125515.

References

Abid, F., Alam, M., Yasir, M., & Li, C. (2019). Sentiment analysis through recurrent variants latterly on convolutional neural network of Twitter. *Future Generation Computer Systems, 95*, 292–308.

Gyamerah, S. A. (2021). Two-stage hybrid machine learning model for high-frequency intraday bitcoin price prediction based on technical indicators, variational mode decomposition, and support vector regression. *Complexity, 2021*, 1767708. https://doi.org/10.1155/2021/1767708

Koo, E., & Kim, G. (2024). Centralized decomposition approach in LSTM for Bitcoin price prediction. *Expert Systems with Applications, 237*, Article 121401. https://doi.org/10.1016/j.eswa.2023.121401
Basiri, M. E., Nemati, S., Abdar, M., Cambria, E., & Acharya, U. R. (2021). ABCDM: An attention-based bidirectional CNN-RNN deep model for sentiment analysis. *Future Generation Computer Systems, 115*, 279–294.
Fakharchian, S. (2023). Designing a forecasting assistant of the Bitcoin price based on deep learning using market sentiment analysis and multiple feature extraction. *Soft Computing, 27*(24), 18803–18827.
Jahanbin, K., & Chahooki, M. A. Z. (2023). Aspect-based sentiment analysis of twitter influencers to predict the trend of cryptocurrencies based on hybrid deep transfer learning models. *IEEE Access, 11*, 121656–121670. https://doi.org/10.1109/ACCESS.2023.3327060
Kapar, B., & Olmo, J. (2021). Analysis of Bitcoin prices using market and sentiment variables. *The World Economy, 44*(1), 45–63.
Oikonomopoulos, S., Tzafilkou, K., Karapiperis, D., & Verykios, V. (2022). Cryptocurrency Price Prediction using Social Media Sentiment Analysis, in 2022 13th International Conference on Information, Intelligence, Systems & Applications (IISA), 18-20 July 2022 2022, pp. 1-8, doi: 10.1109/IISA56318.2022.9904351.
Passalis, N., et al. (2022). Multisource financial sentiment analysis for detecting Bitcoin price change indications using deep learning. *Neural Computing and Applications, 34*(22), 19441–19452. https://doi.org/10.1007/s00521-022-07509-6
Jakubik, J., Nazemi, A., Geyer-Schulz, A., & Fabozzi, F. J. (2023). Incorporating financial news for forecasting Bitcoin prices based on long short-term memory networks. *Quantitative Finance, 23*(2), 335–349.
Serafini, G., Yi, P., Zhang, Q., & Brambilla, M. (2020). Sentiment-Driven Price Prediction of the Bitcoin based on Statistical and Deep Learning Approaches," in 2020 International Joint Conference on Neural Networks (IJCNN), 19-24 July 2020 2020, pp. 1-8, doi: 10.1109/IJCNN48605.2020.9206704.
Baroiu, A.-C., & Ene, G. D. (2022). Twitter Sentiment and Bitcoin Price–Is there a connection?," in 2022 26th International Conference on System Theory, Control and Computing (ICSTCC), 2022: IEEE, pp. 258-262.
Han, P., Hong, J., Chen, H., & Pan, Y. (2022). A Hybrid Recommendation Model for Social Network Services Using Twitter Data. In *International Conference on Web Services* (pp. 122–129). Springer.
Ortu, M., Uras, N., Conversano, C., Bartolucci, S., & Destefanis, G. (2022). On technical trading and social media indicators for cryptocurrency price classification through deep learning, Expert Systems with Applications, vol. 198, Jul 2022, Art no. 116804, doi: 10.1016/j.eswa.2022.116804.
Deveikyte, J., Geman, H., Piccari, C., & Provetti, A. (2022). A sentiment analysis approach to the prediction of market volatility. *Frontiers in Artificial Intelligence, 5*, Article 836809. https://doi.org/10.3389/frai.2022.836809
Xu, L. X., Xu, W., Cui, Q. Z., Li, M. Y., Luo, B., & Tang, Y. Y. (2023). Deep heuristic evolutionary regression model based on the fusion of BiGRU and BiLSTM. *Cognitive Computation, 15*(5), 1672–1686. https://doi.org/10.1007/s12559-023-10135-6
Chen, Y., et al. (2021). Stock price forecast based on CNN-BiLSTM-ECA model. *Scientific Programming, 2021*, 1–20.
Kang, C. Y., Lee, C. P., & Lim, K. M. (2022). Cryptocurrency price prediction with convolutional neural network and stacked gated recurrent unit. *Data, 7*(11), 149.
Zhang, X., Zhang, L., Zhou, Q., & Jin, X. (2022). A novel bitcoin and gold prices prediction method using an LSTM-P neural network model. *Computational Intelligence and Neuroscience, 2022*.
Liu,W. W., Zhang, C., Qiao, G. X., & Xu, L. (2022). Impact of network investor sentiment and news arrival on jumps, North American Journal of Economics and Finance, vol. 62, pp. 1780-1780, Nov 2022. [Online]. Available: <Go to ISI>://CCC:000880054800007.
Joon, K. I. M. Y. (2022). An analysis of the effect of social media sentiment on consumer sentiment. *Journal of Market Economy, 51*(1), 71–102. https://doi.org/10.38162/jome.51.1.4
Hong, J., Han, P., Hong, Z., Tan, Z., Lin, F., & Wei, S. X. (2022). A Correlational Strategy for the Prediction of High-Dimensional Stock Data by Neural Networks and Technical Indicators. In *International Conference on Big Data and Security* (pp. 405–419). Springer.
Ghafoori, M., Clevenger, C., Abdallah, M., & Rens, K. (2023). Heart rate modeling and prediction of construction workers based on physical activity using deep learning, Automation in Construction, vol. 155, Nov 2023, Art no. 105077, doi: 10.1016/j.autcon.2023.105077.
Kursa, M. B., & Rudnicki, W. R. (2010). Feature selection with the Boruta package. *Journal of Statistical Software, 36*, 1–13.
Aloraini, A. (2015). Penalized ensemble feature selection methods for hidden associations in time series environments case study: Equities companies in saudi stock exchange market. *Evolving Systems, 6*(2), 93–100.
Pearson, K. (1895). VII. Note on regression and inheritance in the case of two parents, Proceedings of the Royal Society of London, vol. 58, no. 347-352, pp. 240-242, 1895.
Spearman, C. (1961). General intelligence. *Objectively Determined and Measured*.
Kumar, D., Meghwani, S. S., & Thakur, M. (2016). Proximal support vector machine based hybrid prediction models for trend forecasting in financial markets. *Journal of Computational Science, 17*, 1–13.
Htun, H. H., Biehl, M., & Petkov, N. (2023). Survey of feature selection and extraction techniques for stock market prediction. *Financial Innovation, 9*(1), 26.
Das, S. R., Mishra, D., & Rout, M. (2019). Stock market prediction using Firefly algorithm with evolutionary framework optimized feature reduction for OSELM method. *Expert Systems with Applications: X, 4*, Article 100016.
Kramer, M. A. (1991). Nonlinear principal component analysis using autoassociative neural networks. *AIChE Journal, 37*(2), 233–243.
Dami, S., & Esterabi, M. (2021). Predicting stock returns of Tehran exchange using LSTM neural network and feature engineering technique. *Multimedia Tools and Applications, 80*, 19947–19970.
Manikandan, G., Pragadeesh, B., Manojkumar, V., Karthikeyan, A., Manikandan, R., & Gandomi, A. H. (2024). Classification models combined with Boruta feature selection for heart disease prediction, Informatics in Medicine Unlocked, p. 101442, 2024.
Aslam, S., Jiang, Q., & Qu, Q. (2021). LSTM based model for real-time stock market prediction on unexpected incidents, in 2021 IEEE International Conference on Real-time Computing and Robotics (RCAR), 2021: IEEE, pp. 1149-1153.
Guo, Q., Lei, S., Ye, Q., & Fang, Z.(2021). MRC-LSTM: a hybrid approach of multi-scale residual CNN and LSTM to predict bitcoin price, in 2021 International Joint Conference on Neural Networks (IJCNN), 2021: IEEE, pp. 1-8.
Zhan, X., Li, Y., Li, R., Gu, X., Habimana, O., & Wang, H. (2018). Stock price prediction using time convolution long short-term memory network. In *Knowledge Science, Engineering and Management: 11th International Conference, KSEM 2018, Changchun, China, August 17–19, 2018, Proceedings, Part I 11* (pp. 461–468). Springer.
Bishop, C. (2006). Pattern recognition and machine learning. *Springer Google Schola, 2*, 35–42.
Ye, Z., Wu, Y., Chen, H., Pan, Y., & Jiang, Q. (2022). A stacking ensemble deep learning model for bitcoin price prediction using Twitter comments on bitcoin. *Mathematics, 10*(8), 1307.
Montavon, G., Samek, W., & Müller, K.-R. (2018). Methods for interpreting and understanding deep neural networks. *Digital Signal Processing, 73*, 1–15.
Zhang, C., Sjarif, N. N. A., & Ibrahim, R. (2023). Deep learning models for price forecasting of financial time series: A review of recent advancements: 2020–2022, Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery, p. e1519, 2023.
Kendall, M. G. (1938). A new measure of rank correlation. *Biometrika, 30*(1/2), 81–93.
Hutto, C., & Gilbert, E. (2014). Vader: A parsimonious rule-based model for sentiment analysis of social media text. *Proceedings of the International AAAI Conference on Web and Social Media, 8*(1), 216–225.
Maschotta, R., Boymann, S., & Hoppe, U. (2007). Comparison of feature-list cross-correlation algorithms with common cross-correlation algorithms. *EURASIP Journal on Advances in Signal Processing, 2007*, 1–15.
Ghosh, I., & Chaudhuri, T. D. (2022). Integrating Navier-Stokes equation and neoteric iForest-BorutaShap-Facebook's prophet framework for stock market prediction: An application in Indian context. *Expert Systems with Applications, 210*, Article 118391.
Nadimi-Shahraki, M. H., Zamani, H., Asghari Varzaneh, Z., & Mirjalili, S. (2023). A Systematic Review of the Whale Optimization Algorithm: Theoretical Foundation, Improvements, and Hybridizations, Archives of Computational Methods in Engineering, pp. 1-47, 2023.
Rasool, A., Qu, Q., & Ji, C. (2021). WRS: a novel word-embedding method for real-time sentiment with integrated LSTM-CNN model, in 2021 IEEE International Conference on Real-time Computing and Robotics (RCAR), 2021: IEEE, pp. 590-595.
Kamyab, M., Liu, G., & Adjeisah, M. (2022). ACR-SA: Attention-based deep model through two-channel CNN and Bi-RNN for sentiment analysis. *PeerJ Computer Science, 8*, e877.
Wen, S., & Li, J. (2018). Recurrent convolutional neural network with attention for twitter and yelp sentiment classification: ARC model for sentiment classification. In *in Proceedings of the 2018 International Conference on Algorithms, Computing and Artificial Intelligence* (pp. 1–7).
Liu, G., & Guo, J. (2019). Bidirectional LSTM with attention mechanism and convolutional layer for text classification. *Neurocomputing, 337*, 325–338.
Xu, R., Zhang, Z., Luo, F., & Tan, C. (2021). Raise a child in large language model: Towards effective and generalizable fine-tuning, arXiv preprint arXiv:2109.05687, 2021.
Aslam, S., Aslam, H., Manzoor, A., & Hui, C. (2024). AntiPhishStack: LSTM-based Stacked Generalization Model for Optimized Phishing URLs Detection, arXiv preprint arXiv:2401.08947, 2024.
Alharbi, A. S. M., & de Doncker, E. (2019). Twitter sentiment analysis with a deep neural network: An enhanced approach using user behavioral information. *Cognitive Systems Research, 54*, 50–61.
Shukla, M., & Kumar, A. (2023). An experimental analysis of deep neural network based classifiers for sentiment analysis task. *IEEE Access*.
Ji, X., Wang, J., & Yan, Z. (2021). A stock price prediction method based on deep learning technology. *International Journal of Crowd Science, 5*(1), 55–72.
Bâra, A., & Oprea, S.-V. (2024). An ensemble learning method for Bitcoin price prediction based on volatility indicators and trend. *Engineering Applications of Artificial Intelligence, 133*, Article 107991.
Devi, M. S., Pandian, J. A., Umanandhini, D., Surekha, B., & Sree, K. B. (2024). Deep Q-Network Based Reinforcement Learning for Bitcoin Future Price Prediction, in 2024 International Conference on Advancements in Smart, Secure and Intelligent Computing (ASSIC), 2024: IEEE, pp. 1-6.
Wahidur, R. S., Tashdeed, I., Kaur, M., & Lee, H.-N. (2024). Enhancing Zero-shot crypto sentiment with fine-tuned language model and prompt engineering. *IEEE Access*.
Aslam, S., Wu, H., & Li, X. (2024). CEL: A Continual Learning Model for Disease Outbreak Prediction by Leveraging Domain Adaptation via Elastic Weight Consolidation, bioRxiv, p. 2024.01. 13.575497, 2024.