IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received 17 March 2024, accepted 24 March 2024, date of publication 8 April 2024, date of current version 15 April 2024.

Digital Object Identifier 10.1109/ACCESS.2024.3386029

RESEARCH ARTICLE

HyBiLSTM: Multivariate Bitcoin Price Forecasting Using Hybrid Time-Series Models With Bidirectional LSTM

ANNY MARDJO[1] AND CHIDCHANOK CHOKSUCHAT[2], (Member, IEEE)
[1]College of Digital Science, Prince of Songkla University, Hat Yai 90110, Thailand
[2]Division of Computational Science, Faculty of Science, Prince of Songkla University, Hat Yai 90110, Thailand

Corresponding author: Chidchanok Choksuchat (chidchanok.ch@psu.ac.th)

This work was supported by the Digital Science for Economy, Society, Human Resources Innovative Development and Environment project funded by Reinventing Universities & Research Institutes under grant no. 2046735, Ministry of Higher Education, Science, Research and Innovation, Thailand.

ABSTRACT Despite their growing popularity in recent research, most hybrid models that harness the strengths of both classical time-series analysis and deep learning models have been explored within the univariate forecasting context. In the econometric domain, where exogenous factors play a crucial role; there is a pressing need for more studies focusing on multivariate forecasting. This paper introduces a novel hybrid model, HyBiLSTM. It integrates an ARIMAX GARCHX model for initial forecasting, followed by a second forecasting phase that addresses the residuals using a bidirectional long short-term memory model optimized through grey wolf optimization algorithm. The final forecast is a composite derived from both models. Three quantitative metrics (mean absolute error, root mean square error, and mean absolute percentage error) assessed the model performance using data that spanned social and economic variables from July 1, 2019, to December 31, 2022. The results revealed several key findings: 1) The addition of exogenous factors improved the performance of the ARIMA and GARCH models. 2) The BiLSTM variant outperformed other LSTM variants when combined with the ARIMAX GARCHX model. 3) An analysis using Shapley additive explanations indicated that bitcoin price was influenced by stock prices, Twitter volume, gold prices, and the Twitter sentiment index. 4) The presence of a structural break had a significant effect on the model's forecasting accuracy. Beyond expanding the academic literature on hybrid models within a multivariate context, this offers valuable practical insights for investors. Specifically, it analyzes various factors that could serve as early indicators of bitcoin price fluctuations.

INDEX TERMS LSTM, ARIMAX, GARCHX, bitcoin, SHAP, bidirectional LSTM.

I. INTRODUCTION

Cryptocurrencies, unlike stocks which are underpinned by the tangible assets and earnings of companies, lack inherent value. This absence of intrinsic worth contributes to their high volatility, as evidenced in 2021 when bitcoin's value plummeted by over half before making a remarkable recovery a few months later. Despite these fluctuations presenting a significant investment risk, the allure of decentralization, the opportunity for investment diversification, the potential for high returns, and the convenience of 24/7 trading access continue to draw an increasing number of investors to the cryptocurrency market. Since its debut in 2009 as the inaugural decentralized currency, bitcoin has maintained its status as the most prestigious cryptocurrency. Notable endorsements such as electric car giant Tesla's investment in bitcoin and El Salvador's groundbreaking move to adopt bitcoin as a legal tender in 2021 have bolstered its reputation as a digital asset and a viable currency. To minimize the risk and gain the highest return, a dependable and accurate forecasting model that

The associate editor coordinating the review of this manuscript and approving it for publication was Frederico Guimarães.

encapsulates bitcoin's unique characteristics, such as high volatility, non-stationarity, non-linearity and dependency on numerous factors, is vital.

The quest for effective price prediction models has led to increased research into multivariate bitcoin price forecasting in recent years. Traditional multivariate statistical time-series models, including the autoregressive integrated moving average (ARIMA), generalized autoregressive conditional heteroskedasticity (GARCH), and their variants [1], [2], [3], have been explored owing to their straightforward interpretation, ease of implementation, and fast computing times. However, these models often fall short in capturing the complex, non-stationary, and non-linear dynamics of bitcoin prices. By contrast, deep learning (DL) models have shown promising results in addressing these challenges. Long short-term memory (LSTM) networks, in particular, have gained popularity for their ability to overcome the diminishing gradient problem, a common issue with recurrent neural networks (RNNs). This is achieved through the inclusion of a hidden memory layer that retains long-term dependencies, allowing the model to more accurately learn patterns within sequential data sets [1]. Other studied DL models that have been investigated include adaptive neuro-fuzzy inference systems [2],; gated recurring units GRUs [3], [4], RNNs [5], [6], and Multilayer Perceptron [5], [6].

Despite its advantages, LSTM faces challenges in processing multiple features simultaneously. It treats each feature independently, assigning separate weights and biases using the same computational graph at a given timestep. Consequently, it produces a hidden state for each feature, which are then aggregated as input for the subsequent timestep. This approach complicates the model convergence and extends its processing time. To overcome these limitations and harness the strengths of classical time-series models and LSTM for predicting multivariate time-series data, we propose a hybrid model that combines ARIMAX GARCHX with various LSTM variants, including standard LSTM, bidirectional LSTM, and attention-based layer LSTM. The scarcity of studies on hybrid models in multivariate forecasting further inspired our research.

In the context of multivariate forecasting, which requires the consideration of multiple input variables, our study emphasizes social and economic factors known to influence bitcoin prices. This decision was driven by the widespread availability and comprehensibility of data related to social and economic factors, increasing the relevance and practicality of this study. Past studies have explored social-based factors such as Google search trends, tweet sentiments, tweet volume, Wikipedia search trends, online community comments [7], [8], [9], [10], [11], [12], [13], [14], [15] while economic-based factors have included variables such as the federal funds rate, stock market indexes, gold prices, oil prices, foreign exchange rates, consumer price indexes, US dollar indexes, general commodity indexes, US Dollar supply, US GDP, bonds, producer price indexes, and employment rate) [7], [8], [9], [10], [11], [16], [17], [18].

This paper makes several key contributions:

1) It offers an in-depth comparison and evaluation of hybrid multivariate time-series models, specifically ARIMAX and GARCHX combined with various LSTM variants (i.e., standard LSTM, BiLSTM, and attention-based layer LSTM). To the best of our knowledge, this is the first study to provide such an in-depth analysis of these models.

2) It utilizes Shapley additive explanation (SHAP) values to comprehend the effects of exogeneous factors on bitcoin price forecasting, thereby enriching our understanding or reinforcing existing knowledge of bitcoin's complex price dynamics.

The remaining sections of this paper are structured as follows: Section II provides a comprehensive overview of the literature reviews related to our work. In Section III, we introduce the methodology employed in our research. Section IV presents the results and discussions. Ultimately, in Section V, we offer our concluding remarks and explore potential directions for future research.

II. RELATED WORKS

A. HYBRID TRADITIONAL STATISTICAL MODEL OF TIME-SERIES DATA

Hybrid models in time-series data analysis represent a fusion of two or more forecasting methodologies, designed to leverage the strengths of various modeling approaches, while mitigating their individual limitations. These models have mainly been explored in univariate forecasting, which, despite its simplicity and relative reliability, often falls short when addressing complex econometric challenges. This is primarily attributed to its reliance on a single variable, underscoring the need for incorporating exogenous factors to enhance modeling capability and predictive accuracy [19]. Multivariate models using DL generally require more processing complexity and longer training times, rendering them less practical for certain applications. Conversely, classical time-series models such as ARIMAX and GARCHX, though simpler and quicker to implement, typically exhibit lower forecasting accuracy than their DL counterparts. A hybrid model, therefore, that leverages the advantages of both methodologies presents a desirable solution. Following [20], our hybrid model assumes that a time-series y_t consists of a linear component L_t and a non-linear component N_t, as specified in (1).

$$y_t = L_t + N_t \tag{1}$$

Using two-level forecasting procedures, the first level utilized ARIMAX GARCHX model to forecast the linear component of time-series data $\hat{L}_t$ while the second level analyzed the residuals of that model e_t (2) using LSTM variants to forecast the non-linear pattern $\hat{N}_t$. Then, both linear and non-linear forecasts were combined to obtain the final prediction (3). Fig. 1 depicts our proposed model.

$$e_t = y_t - \hat{L}_t \tag{2}$$

$$\hat{y}_t = \hat{L}_t + \hat{N}_t \tag{3}$$

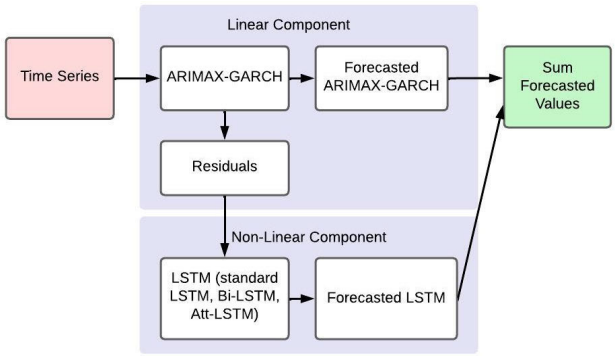

FIGURE 1. Proposed hybrid model.

B. MULTIVARIATE HYBRID TIME-SERIES FORECASTING MODEL

The "Multivariate hybrid time-series forecasting model" integrates multiple forecasting techniques and multiple variables (or series) simultaneously to predict one or more outcomes effectively. For analyzing the historical impact of external factors on bitcoin price, the ARIMAX model has been explored. The general form of ARIMAX (p,d,q) model [6] can be written as:

$$\phi\,(B)\,(1-B)^d\,Y_t = \theta_0 + \theta(B)\varepsilon_t + \sum_{i=1}^{r} a_i x_{it} \tag{4}$$

where B is the backward shift operator; $(1-B)^d$ is the differencing operator of order d to produce stationarity of the dth differenced data; $\phi\,(B) = \left(1-\phi B-\phi_2 B^2-\ldots-\phi_p B^p\right)$ is a moving average polynomial with order p; $\theta\,(B) = \left(1-\theta_1 B-\theta_2 B^2-\ldots-\theta_q B^q\right)$ is an autoregressive polynomial with order q; $\phi_1 \ldots \phi_p$ = the parameters of the autoregressive part of model; $\theta_0 \ldots \theta_q$ = the parameters of the moving average part; d is a number of times of order differencing; ε_t is error term; x_t is the value of the independent variable X at time t; a, ϕ, θ = coefficients of the factors, autoregressive terms, moving average terms; and r refers to number of factors.

As ARIMAX linearly models the data, the GARCHX model [21] was added to capture heteroscedastic variances of highly volatile bitcoin prices. The addition of X on the standard GARCH model in both the mean and variance of the model has been reported to improve model forecasting [22]. The mean and variance model can be written as (5) and (6).

$$\phi\,(B)\,(1-B)^d\,Y_t = \theta(B)\varepsilon_t + \sum_{s=0}^{l}\sum_{k=1}^{r} Y_{ks} x_{k,t-s} \tag{5}$$

$$\sigma_t^2 = a_0 + \sum_{i=1}^{p} a_i\,\epsilon_{t-1}^2 + \sum_{j=1}^{q} b_j \sigma_{t-j}^2 + \sum_{s=1}^{l}\sum_{k=1}^{r} Y_{ks} x_{k,t-s} \tag{6}$$

where r is the number of exogenous variables; l is the lag length of the exogenous variables; a, b = coefficients of the autoregressive terms, moving average terms; and Y_k are the effects of the exogenous variables on the conditional variance of the residuals.

C. DEEP LEARNING MODEL: LSTM, BiLSTM, AND ATTENTION-BASED LAYER LSTM

DL models outperform traditional statistical models in handling complex non-linear patterns and capturing long-term dependencies within data sets. However, despite their popularity in DL studies, RNNs cannot preserve long sequences of input, leading to the well-documented issue of long-term dependency loss [23]. To overcome this, LSTM networks were introduced by Hochreiter and Schmidhuber [24]. LSTMs are designed with a three-gate mechanism alongside memory cells C_t within the hidden layers, which significantly enhance their capacity to preserve information over extended sequences. The process starts with the update of the candidate for the new cell states $\tilde{C}_t$. This update is a function of the hidden state from the previous timestep h_{t-1}, and the input vector at the current timestep x_t, calculated using (7)

$$\tilde{C}_t = \tanh\left(W_c h_{t-1} + U_c x_t + b_c\right) \tag{7}$$

where W_c, U_c represent the weights of networks and b_c are bias variable value. Then, it computes forget f_t and input i_t gates to determine how much contents from the previous cell C_{t-1} will be erased and how much of values of the new candidate cell states $\tilde{C}_t$ combined into the new cell state C_t, using (8), (9) and (10) respectively.

$$f_t = \sigma\left(W_f h_{t-1} + U_f x_f + b_f\right) \tag{8}$$

$$i_t = \sigma\left(W_i h_{t-1} + U_i x_t + b_i\right) \tag{9}$$

$$C_t = f_t \cdot C_{t-1} + i_t \cdot \tilde{C}_t \tag{10}$$

where W_f, W_i, U_f, U_i are the weights of networks; and b_f, b_i represent the bias variable values. Output hidden states h_t will then be filtered by output gate o_t using (11) and (12) to decide the value of cell state C_t that will go to the output.

$$o_t = \sigma\left(W_0 h_{t-1} + U_o x_t + b_o\right) \tag{11}$$

$$h_t = o_t \otimes \tanh\left(C_t\right) \tag{12}$$

where W_0, are weights of the networks; and b_o are bias variable values. Two activation functions were used here, namely the sigmoid (σ) and the tanh. The prediction of the next event $\hat{x}_{t+1}$ can be calculated using:

$$\hat{x}_{t+1} = g\left(V.h_t\right) \tag{13}$$

where V is output layer weight matrix and g can be any activation function that matches the type of the target in data. A two-layer stacked LSTM (Fig. 2) was proposed as the first model, based on findings that a two-layer LSTM structure improves the accuracy of model predictions [25].

The second proposed model is a two-layer stacked BiLSTM (Fig. 3). The BiLSTM model utilizes two separate

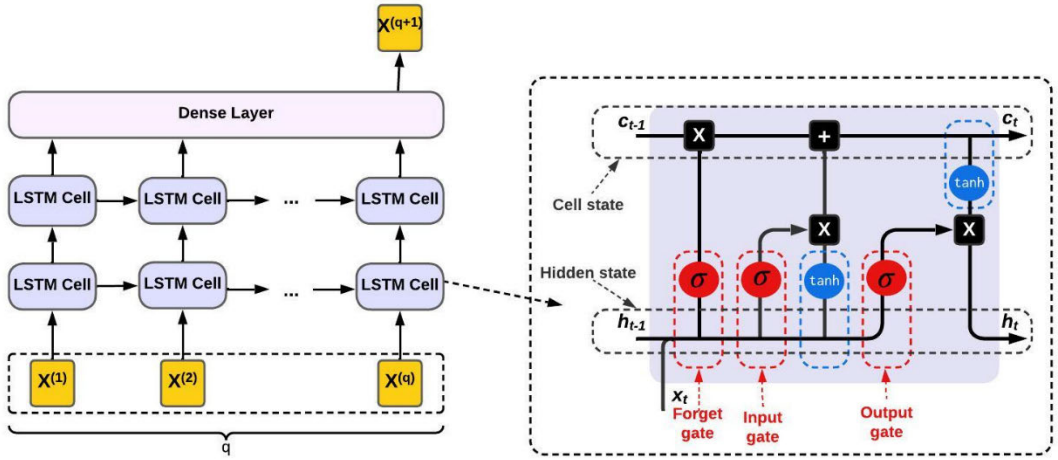

FIGURE 2. Proposed first model: A two-layer stacked standard LSTM (LSTM).

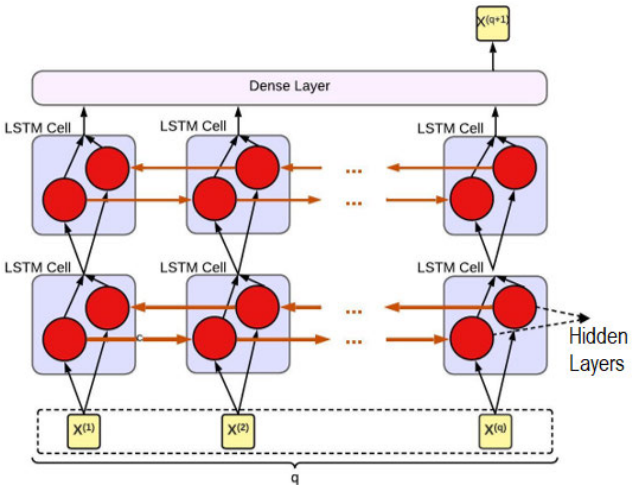

FIGURE 3. Proposed second model: A two-layer stacked bidirectional LSTM (BiLSTM).

hidden layers, each processing data in opposite directions to account for both past and future data [26]. The output values from each hidden layer are calculated independently and subsequently merged to produce the final output. Using this approach, both past and future information contained in the data set can be preserved.

The accuracy of LSTM models tends to decrease as the length of input and output sequences increases. This decline is attributed to the model's hidden state being overwritten repeatedly, which reduces the influence of initial inputs on subsequent outputs within the sequence. Attention mechanisms [27] seek to correct this by weighting the hidden states of all available time steps $h_1, \ldots, h_t$ instead of solely relying on the last hidden state h_t [28]. By assigning different weights to different hidden states, the model can prioritize the most relevant parts of the input sequence for making predictions. Consequently, the output o_t can be computed as a weighted sum of hidden states, as shown in (14):

$$o_t = \sum_{i=1}^{t} \alpha_i^t \cdot h_i \tag{14}$$

where attention weight α_i^t is computed as follows:

$$\alpha_i^t = \frac{exp(score\ (h_i, q_t))}{\sum_{j=1}^{t} exp(score\ (h_i, q_t))} \tag{15}$$

while the following (16) for the score function used Dot-product attention algorithm, selected owing to its computing efficiency and popularity

$$score\ (h_i, q_t) = h_i^t W_a q_t \tag{16}$$

where W_a represents the weight matrix and the previous timestep's output o_{t-1} serves as the query term q_t. For our proposed third (Fig. 4) and fourth models (Fig. 5), an additional LSTM and application layers were added above the first and second models. The architecture facilitates a combination of the attention-based layer o_t and the LSTM output h_{t-1} as the next layer's input for the current timestep, results in

$$X_t = concat(o_t, h_{t-1}) \tag{17}$$

D. LSTM PARAMETERS OPTIMIZATION: GREY WOLF OPTIMIZER

To optimize LSTM parameters, the grey wolf optimizer, a bio-inspired computing algorithm (BIC) influenced by the natural behavior of animals, birds, insects, and other organisms, was used. Although BIC algorithms can be resource-intensive owing to the need for parameter optimization and iterative processes, they excel at uncovering unknown patterns with less dependence on mathematical modeling or exhaustive training [29]. The grey wolf optimizer algorithm, inspired by the social structure and hunting

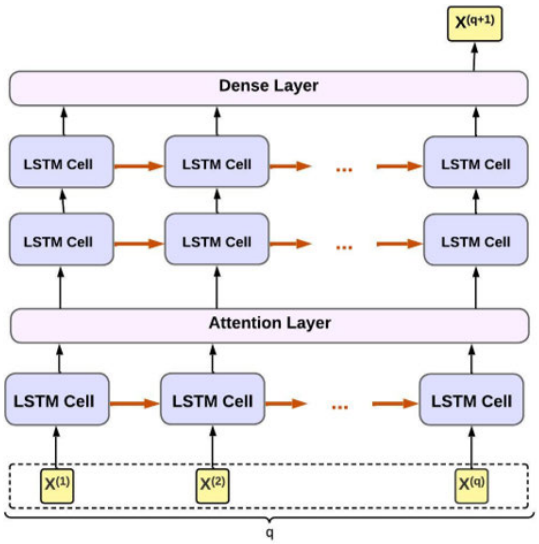

FIGURE 4. Proposed 3rd model: two layers stacked attention layer LSTM (ATT-LSTM).

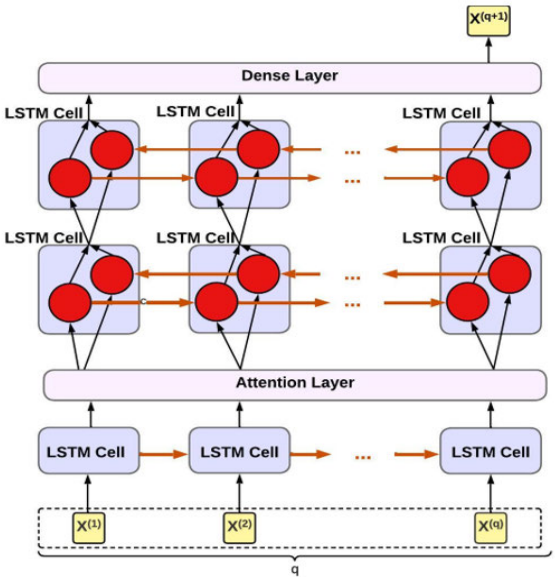

FIGURE 5. Proposed fourth model: A two-layer stacked attention BiLSTM (ATT-BiLSTM).

strategies of grey wolves, was selected from among various BIC algorithms for its proven effectiveness across different fields and its relative simplicity [30], [31], [32]. In the hierarchy of a wolf pack, the alpha (α) wolf leads and makes decisions, followed by the beta (β) wolf, who acts as the advisor and enforces discipline. The delta (δ) wolf holds a position above the omega (Ω) wolf, who is the lowest-ranking member and submits to all others. The grey wolf optimizer algorithm operates through the following steps: 1) each wolf measures its distance from α, β, and δ wolves using equations (18) to (23); and 2) updates its position according to equation (24) [33].

$$D_\alpha = |2r_2.X_\alpha - X_i| \tag{18}$$

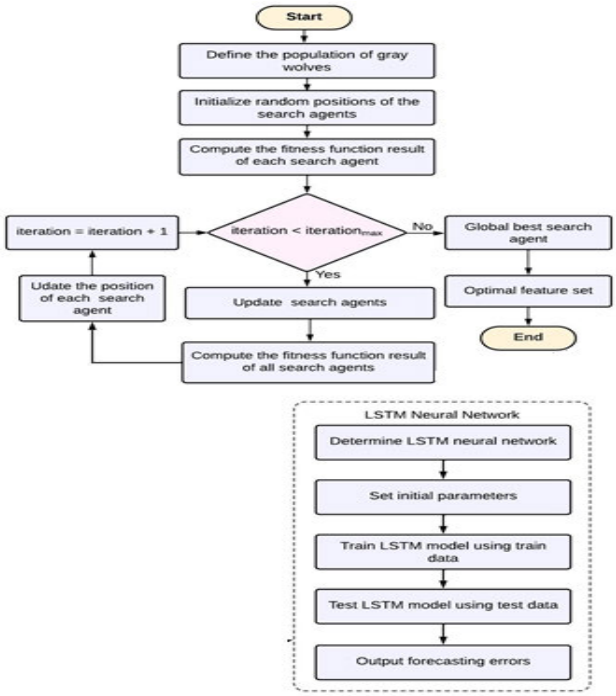

FIGURE 6. Diagram illustrating the grey wolf optimizer algorithm used in conjunction with the proposed LSTM models.

$$D_\beta = |2r_2.X_\beta - X_i| \tag{19}$$
$$D_\delta = |2r_2.X_\delta - X_i| \tag{20}$$
$$X_1 = X_\alpha - (2a \cdot r_1 - a) \cdot D_\alpha \tag{21}$$
$$X_2 = X_\beta - (2a \cdot r_1 - a) \cdot D_\beta \tag{22}$$
$$X_3 = X_\delta - (2a \cdot r_1 - a) \cdot D_\delta \tag{23}$$
$$X_1\,(t+1) = \frac{X_1 + X_2 + X_3}{3} \tag{24}$$

where X_α, X_β, and X_δ are the positions of the search agents (grey wolves); D_α, D_β, and D_δ are distances between the current position i and the search agents (α, β, δ); $r1$ and $r2$ are randomly generated numbers for boundary search space. Fig. 6 depicted GWO used in our study.

E. SHAPLEY ADDITIVE EXPLANATIONS (SHAP)

Another important contribution of this paper is the usage of SHAP for model explanation, aimed at identifying the factors affecting bitcoin price. SHAP was chosen for its three distinct advantages. First, SHAP is a local interpretable machine learning (IML) technique that offers flexibility for adaptation into constant global explanations [34]. This dual capability allows for both the analysis of specific operational points and the identification of general trends. Second, the computation of Shapley values, derived from the average marginal contribution of each feature across all possible combinations (or coalitions) including that feature, provides deep insights into how a particular feature influences the model outcome. This is invaluable for model refinement and adjustment. Third, SHAP's additive feature attribution method fosters

integration with other IML techniques, such as LIME and DeepLIFT. The SHAP model can be represented as a linear combination of the binary variables as follows (25):

$$g\left(z'\right) = \phi_0 + \sum_{i=1}^{M} \phi_i z_i' \tag{25}$$

where g is an explanatory model, $z^i \epsilon \{0, 1\}^M$ is the coalition vector, and M is the maximum number of features. The ith feature either contributes ($z' = 1$) or does not contribute ($z' = 0$) to the model. The SHAP value ϕ is of the ith feature quantifies its contribution and is calculated using the following formula (26):

$$\phi_i\left(f, x\right) = \sum_{S \subset N} \frac{|S|!\left(M - |S| - 1\right)!}{M!} \left[f_x\left(S \cup \{i\}\right) - f_x\left(S\right)\right] \tag{26}$$

where N is the set of all features, $|S|$ represents the number of features in feature subset S excluding the ith feature; and $f_x(S)$ represents the result of the machine learning model f training in feature subset S.

III. METHODOLOGY

Research methodology consists of data life cycle management, extending to the application of forecasting models.

A. DATA EXTRACTION

As in [2], historical Twitter volume was obtained from bitinfocharts.com using the keyword "Bitcoin" (case insensitive). Google search volume was gathered using the Gtrends library. The daily closing prices of Bitcoin, oil price, gold price, and U.S. stock market indexes (S&P 500, NASDAQ, and Dow Jones Industrial Average) were collected using either the *Quantmod* or *Quandl* libraries. Following prior research (Chen, 2023) to consider less than four-year date range limit and Bitcoin price bubbles, the data were collected for the period between 01/07/2019 and 31/12/2022.

Using Twitter API, we collected 2,000 tweets four times a day with keyword "Bitcoin." HyVADRF (Hybrid VADER-Random Forest) and GWO algorithms [4] (Fig. 7) was used to pre-process and classify sentiments of the tweets. The VADER algorithm eliminates the need for laborious, time intensity, error prone, and costly manual labeling while the TF-IDF implementation in this algorithm restricts the terms used for classification calculations to those relevant to the data context, resulting in a simpler and faster processing model. The GWO-Random Forest in the model is appropriate, considering the advantages of supervised over unsupervised sentiment analysis [5]. HyVADRF algorithm is divided into three distinct processes: 1) generating data labeling, 2) selecting the best machine learning model, and 3) performing model optimization using the grey wolf optimizer (Fig. 7).

These sentiments were then converted into continuous data for the daily Twitter sentiment using (27), following [36].

$$SENT_t = \frac{N_{positive}}{N_{positive} + N_{negative}} \tag{27}$$

Algorithm: HyVADRF (Hybrid VADER-Random Forest)
Input: Twitter dataset T
Output: Optimized classifying machine learning model
Process:

To generate data labelling:
  for $t \in T$
  Clean t from URL links, *hashtag* symbols, and irrelevant tweets using RegEx.
  Calculate sentiment Compound Value (cv) using VADER library.
  if $cv > 0.05$
    class = "positive"
  elseif cv <= 0.05 and cv >= -0.05
    class = "neutral"
  else
    class = "negative"
  end if
  end for

To select the best machine learning model:
Create Vector Corpus (V) from the *labelled tweets*
Clean v by removing punctuations, numbers, stopping words, and stemming
Create a Document Term Matrix (M) from V using TF-IDF.
Clean M by removing sparse terms using a 98% term sparsity threshold.
Divide M into 70% for train set (N) and 30% for the test set (D).
for $j \in$ (NB,DT,RF,SVM)
  Train N using j model.
  Test the trained model using D
end for
Select the best performance-trained model based on accuracy, precision, recall, and F-score.

To perform model optimization using Grey Wolf Optimizer GWO):
Initialize GWO parameters (i.e. *maxIteration*).
while $i=0 <$ *maxIteration*
  Get *min.node.size, num.trees, mtry and sample.fraction* values from GWO.
  Execute *ML* train model with *min.node.size, num.trees, mtry and sample.fraction* values using N.
  Execute *ML* with D to get accuracy.
end while
Get optimum result of *min.node.size, num.trees, mtry and sample.fraction* values.
Execute *min.node.size, num.trees, mtry,* and *sample.fraction* values with N and D.

FIGURE 7. Algorithm for hybrid VADER-random forest.

where $N_{positive}$ is the total positive tweets on day t and $N_{negative}$ is total negative tweets on day t. The daily Twitter sentiments were then combined with other data (Fig. 8).

B. DATA PREPROCESSING

Preprocessing of the original data set was essential to address varying data recording methods, missing values, and non-uniform scales among the features. Given that these daily data would be used for future time-series predictions, which largely rely on historical data, missing or incomplete were handled using linear interpolation from the *imputeTS* library of R. This method has been shown to yield better results than the mean substitution method for handling missing values in time-series data [37]. The data underwent a decomposition process to observe its seasonality. Outliers are commonly occurred in time-series data, especially in cryptocurrency, commodity, and finance [38]. To identify extreme outliers, descriptive statistics of the data were computed. These outliers were retained to preserve the dynamics of each series caused by exceptional events [38]. However, as dominant patterns from variances of significant outliers can hide trends, the logarithmic transformation was used to normalize the data.

The data set was split into two parts: approximately 85% for the training data set, 20% of which was used as an LSTM validation test, and 15% for the test set. Given that the data sets feature different types of measurements and contain no negative values, they were scaled to fall within the range of (0,1). Considering the significant influence of stationarity on the efficiency of a time-series model [39], it was necessary to confirm the data stationarity using the augmented Dicker–Fuller (ADF) test for unit root testing. If cases where

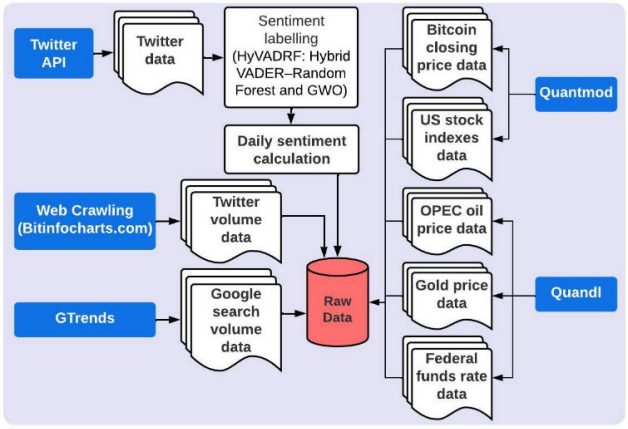

FIGURE 8. Process for data collection.

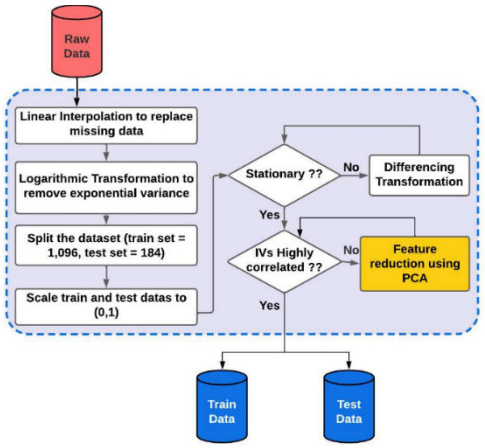

FIGURE 9. Data preprocessing.

non-stationarity was detected, differencing was conducted to de-trend the data. The degree of differencing d was determined based on the ACF plot pattern. A pattern indicative of a first-order moving average process with a parameter of -0.5 suggested over-differencing. Once stationarity was achieved, data correlations were tested using the Pearson correlation coefficient. Our analysis included factors that have been identified in previous studies as influencing bitcoin price. Principal component analysis (PCA) was utilized to evaluate the possibility of merging highly correlated factors into one factor, rather than removing them (Fig. 9).

C. FORECASTING MODELS

The development and validation of the proposed model was separated into two levels: linear and non-linear components (Fig. 10). At level 1, the process started by finding the most suitable ARIMAX (p, d, q) models. The selection of p and q parameters involved iterating through values $p \in (0, 5)$ and $q \in (0, 5)$. The best candidates ARIMAX models were then selected based on their Akaike information criterion (AIC) and Bayesian information criterion (BIC) scores. These selected models underwent further evaluation using test data, where the optimal ARIMAX model was determined by its performance, specifically the RMSE, MAE, and MAPE scores. Afterward, an ARCH Lagrange multiplier (LM) test on the residuals of the best-performing ARIMAX model was conducted. Should an ARCH effect be detected, incorporating a GARCHX model with the ARIMAX model could overcome the non-stationarity in the residuals and improve model performance.

The p and q parameters of the GARCHX (p, q) model were obtained by iterating $p \in (0, 5)$ and $q \in (0, 5)$. Level 2 utilized the fitted ARIMAX GARCHX model as inputs for LSTM models. The identification of lags for LSTM, BiLSTM, ATT_LSTM, and ATT-BiLSTM was achieved using the partial autocorrelation function (PACF) plot [25]. Different combinations of hyperparameters, namely *epoch, hidden nodes, dropout rate, learning rate*, and *batch size*, were selected through the grey wolf optimization algorithm. The

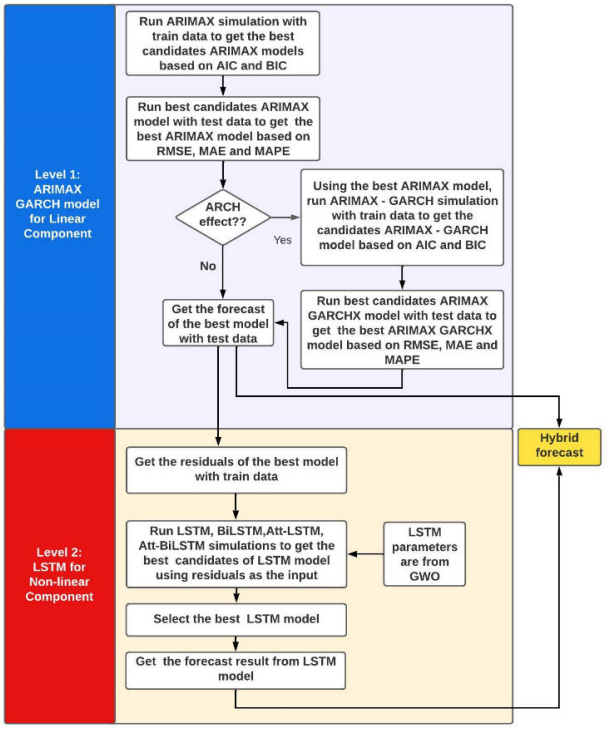

FIGURE 10. Methodology.

final forecast was obtained by combining forecasted residuals from the LSTM model with forecasted values based on the ARIMAX GARCHX model. Moreover, the SHAP algorithm was applied to provide global and local explanation of the ARIMAX GARCHX model predictions.

D. PERFORMANCE MEASUREMENTS

The preliminary data analysis began with assessing the normal distribution of the data through descriptive statistics, specifically skewness and kurtosis. Subsequently, the non-stationarity of the data was examined using ADF tests. A p-value from the ADF test less than 0.05 would indicate a stationary series. Candidates for the ARIMAX and GARCHX

models were selected based on their lowest AIC and BIC values using equations (28) and (29):

$$AIC = \frac{-2}{N} * (log{-}likelihood) + 2 * k/N \tag{28}$$

$$BIC = -2 * (log{-}likelihood) + ln(N) * k \tag{29}$$

where k is the number of model parameters; N is the number of examples in the training data set, and *log-likelihood* is a measure of model fit. The optimal ARIMAX, GARCHX, and LSTM were selected according to their lowest MAE, RMSE, and MAPE. Lower values in these metrics indicate that the model's predicted values are closer to the true values [40]. MAE, RMSE, and MAPE were calculated using the following formulas:

$$MAE = \frac{1}{n}\sum_{i=1}^{n}\left|y_i - \hat{y}_i\right| \tag{30}$$

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}\left|y_i - \hat{y}_i\right|^2} \tag{31}$$

$$MAPE = \sum_{i=1}^{n}\frac{\left|y_i - \hat{y}_i\right|}{y_i} \tag{32}$$

where y_i is the actual value, $\hat{y}_i$ is the predicted value, y_{mean} is the mean of actual value, and n is a total number of data points. For the ARIMAX model demonstrating the best performance, residual diagnostics were conducted to evaluate the assumption of non-autocorrelated random stationarity using the Ljung–Box Q–statistic test. A returned p-value greater than 0.05 indicates that the residuals are independent, suggesting a good model fit. Additionally, an ARCH LM test was performed to detect any ARCH effects. A p-value less than 0.05 would imply variance heteroscedasticity, indicating that adding a GARCHX model could improve ARIMAX performance. The residuals from the best-performing ARIMAX GARCHX models served as input data for various LSTM configurations (LSTM, BiLSTM, ATT-LSTM, and ATT-BiLSTM) to create a hybrid model. The best hybrid models were chosen based on the lowest MAE, RMSE, and MAPE scores. Finally, SHAP values were calculated to explain the model predictions. For global explanations, the average absolute SHAP values of a factor were used, with larger SHAP values indicating greater importance of the factor to the model. Local explanations used individual observational data to analyze its effect on the model predictions.

E. MACHINE SPECIFICATION

The system utilized for this study is equipped with a Lenovo IdeaPad S1145-14IIL, featuring a Processor Intel™Core™ i5-1035G1 CPU operating at a frequency of 1.00 GHz, which can reach 1.19 GHz. It boasts 20.0 GB of installed RAM (19.8 GB usable). This set-up operates on a 64-bit Windows OS, powered by an x64-based processor. The analytical framework was conducted using RStudio version 1.4.1717, leveraging various R libraries including Keras, TensorFlow, metaheuristicOpt, fastshap, dplyr, imputeTS, rugarch, forecast, Quantmod, Quandl, and Gtrends.

IV. RESULTS

The data collection spanned from July 1, 2019, to December 31, 2022. Using the Quantmod library, we collected 1,280 records of bitcoin's closing price. Similarly, records for major stock indexes were collected: 884 for Dow Jones, NASDAQ (NA), and S&P 500 (SP) each. The *Quandl* library facilitated the acquisition of 906 records of OPEC oil prices, 886 records of gold prices, and 1,280 records of the federal funds rate. Additionally, data regarding Bitcoin tweet volumes and Google search trends were compiled, amounting to 1,280 records each, sourced from Bitinfocharts.com and the *Gtrends* library, respectively. 2,000 tweets were collected for each date using the Twitter API and binary labeled using the HyVADRF methodology [30]. These binary labeled tweets for each date were then transformed into continuous values to derive a daily sentiment index. In total, 2,131,900 tweets were collected and pre-processed to construct daily sentiment index for bitcoin tweets. The final total data set was 1,280 records.

A. DATA INTERPRETATION

The decomposition of the bitcoin price time-series was conducted to analyze its seasonality and trends price (Fig.11). The monthly average bitcoin price plot revealed no discernible monthly (a) or yearly (b) seasonality. Furthermore, the plot of bitcoin's closing price indicated an absence of any significant trend. As depicted in Fig 11a, the data curve for 2019 starts in July, aligning with our data set range from July 1, 2019, to December 31, 2022.

In contrast to gold prices and stock indexes, the skewness values for most of the data were positive suggesting longer right tails compared to their left tails (Table 1). Twitter volume and the federal funds rate, exhibiting kurtosis greater than three, are characterized by leptokurtic distributions with fat tails. This suggests a relatively high occurrence of extreme values. Conversely, other variables displayed platykurtic distributions with thinner tails, indicative of fewer extreme values. Prior to splitting the data into 1,096 records for the training set and 184 records for the testing set (Fig. 12), a log-transformation was applied to achieve a more normalized data set. Given the importance of maintaining the chronological order of data points in time-series analysis, the data was not randomized. Such randomization could disrupt the temporal sequencies and dependencies, causing misleading inferences about future trends. Both data sets underwent min–max scaling to transform the values into a uniform range (0–1).

Analysis of the original bitcoin price sequence revealed a unit root across three models, while the first-order difference sequence of these models successfully passed the ADF test (Table 2). Similarly, all the exogenous factors successfully passed the ADF test after the first-order difference series (Table 3).

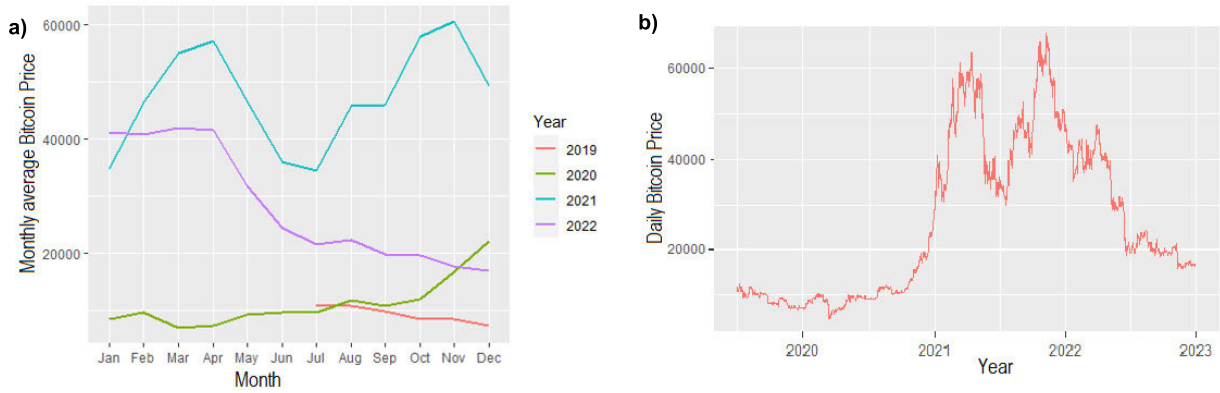

FIGURE 11. (a) Monthly average bitcoin price; (b) yearly bitcoin price.

TABLE 1. Dataset descriptive statistic.

Statistics	Twitter Sentiment Index (TSI)	Twitter Volume (TV)	Google Trends (GT)	Oil Price (OIL)	Gold Price (GO)
Mean	0.3239	84,421	52.11	69.24	1,745
Median	0.3100	88,890	49.00	66.72	1,780
Maximum	0.5500	363,566	100.00	128.27	2,067
Minimum	0.2100	445	22.00	12.22	1,389
Skewness	0.7636	0.6347	0.6339	0.1472	−0.6466
Kurtosis	2.8637	3.1988	2.6588	2.5293	2.6091
Statistics	Dow Jones stock index (DJI)	NASDAQ stock index (NA)	S&P 500 stock index (SP)	Federal Funds Rate (FFR)	Bitcoin Price (BTC)
Mean	30,679	11,687	3,741	0.8873	26,069
Median	31,063	11,622	3,822	0.1000	19,979
Maximum	36,800	16,057	4,797	4.3300	67,567
Minimum	18,592	6,861	2,237	0.0400	4,971
Skewness	−0.4079	−0.0868	−0.1404	1.2275	0.5788
Kurtosis	2.3177	1.8535	1.9022	3.5381	1.9790

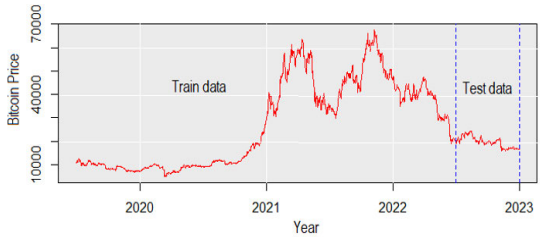

FIGURE 12. Dataset proportion.

TABLE 2. ADF test results of BTC price.

Model settings	original sequence		first-order difference sequence	
	ADF statistic	p-value	ADF statistic	p-value
No intercept term, no trend	−0.158	0.598	−10	0.01
Intercept term with no trend	−0.982	0.708	−10	0.01
Intercept term with a trend	−0.656	0.974	−10	0.01

The ACF plots for both first-order and second-order differences in the original data were analyzed to determine the appropriate d parameter (Fig. 13). The second-order difference exhibited a pattern similar to that of a first-order moving average process with a parameter of −0.5 that signifies over-difference [41]. As a result, the first-order difference was selected for further analysis.

A correlation test was conducted to explore the relationship among exogenous factors, revealing that DJI, NA, and SP stock indexes exhibit high correlation matrixes (Fig. 14). To eliminate information redundancy arising from these highly correlated data, PCA was employed to identify uncorrelated variables known as principal components (PCs), which are obtained through linear combinations of the original variables [42]. This technique allows for the identification of significant items within these PCs, offering the opportunity to consolidate them in a composite variable if deemed appropriate. The first PC (PC1) is the most significant, encapsulating nearly 32.76% of the total information contained within the data set (Fig. 15a). The variables contributing most significantly to PC1 were DJI, NA, and SP, each with loadings exceeding a threshold of 0.50. This threshold is commonly used to identify significant items in the components (Fig. 15b) [43]. Consequently, a composite variable (STO) was created using the PCA loading values for these indexes. Further PCA

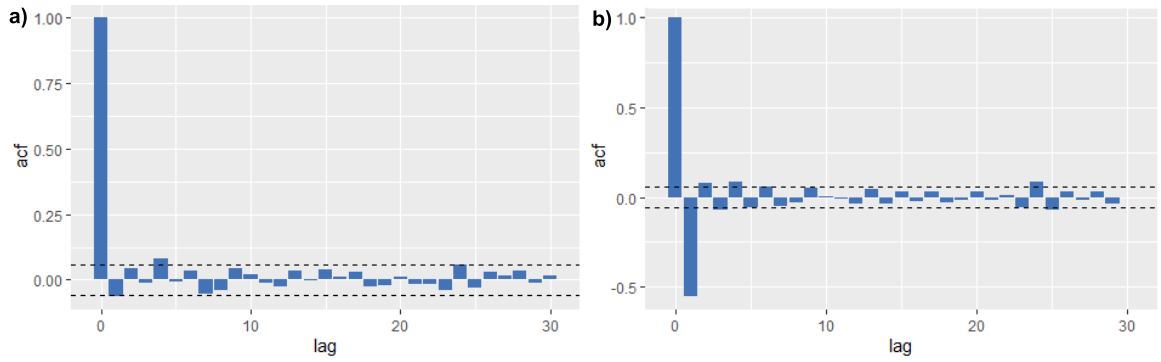

FIGURE 13. (a) First-order difference; (b) second-order difference of the ACF plot.

FIGURE 14. Correlation test before PCA.

analysis revealed that, aside from PC1 and PC2, all other PCs demonstrated higher variances than observed in the initial PCA analysis (Fig15a). This increase in variance suggests that, by assuming PCs with low variance represent noise, the new data set with the composite variable could enhance the signal-to-noise ratio. Such enhancement aids in uncovering underlying patterns within the data more effectively. Multicollinearity testing, with a variance inflation factor score below 4.0 [44] was also satisfied (Fig. 16b). The primary aim of PCA in this context was to address multicollinearity rather than to reduce its dimensionality. Therefore, once multicollinearity concerns were alleviated, further examination of other PCs was deemed unnecessary. Furthermore, all correlation test values of exogenous factors remained below 0.90, signifying no multicollinearity issues in the data (Fig. 17). The strongest correlations with bitcoin prices were observed with stock indexes (−0.33) and gold price (0.16).

B. FITTING MODELS WITH ARIMAX GARCHX

Using a function that traverses $p \in (0,5)$ and $q \in (0,5)$, various combinations of p and q orders for ARIMA and ARIMAX models were explored, with ARIMAX(1,1,1) demonstrating the best performance among these models (Table 4). This result was supported by ACF and PACF plots. Specifically, the ACF plot in Fig 18a suggested potential influences from the first- and fourth-order moving averages, while the PACF plot in Fig 18b hinted at the likelihood of the first- and fourth-order autoregression. The adequacy of the ARIMAX(1,1,1) model was further confirmed by the Ljung–Box test on the residuals, which indicated satisfactory results for lags at 10,

TABLE 3. ADF test results of exogenous factors*.

Exogenous factors	original sequence		first-order difference sequence	
	ADF statistic	p-value	ADF statistic	p-value
TSI	−3.791	0.194	−13.398	0.01
TV	−4.699	0.010	−13.357	0.01
GT	−5.783	0.010	−12.649	0.01
DJI	−1.857	0.639	−9.134	0.01
NA	−0.135	0.990	−9.537	0.01
SP	−1.319	0.867	−9.141	0.01
GO	−2.011	0.574	−11.060	0.01
OIL	−2.214	0.488	−8.500	0.01
FFR	−0.267	0.990	−9.049	0.01

*based on the intercept term with a trend

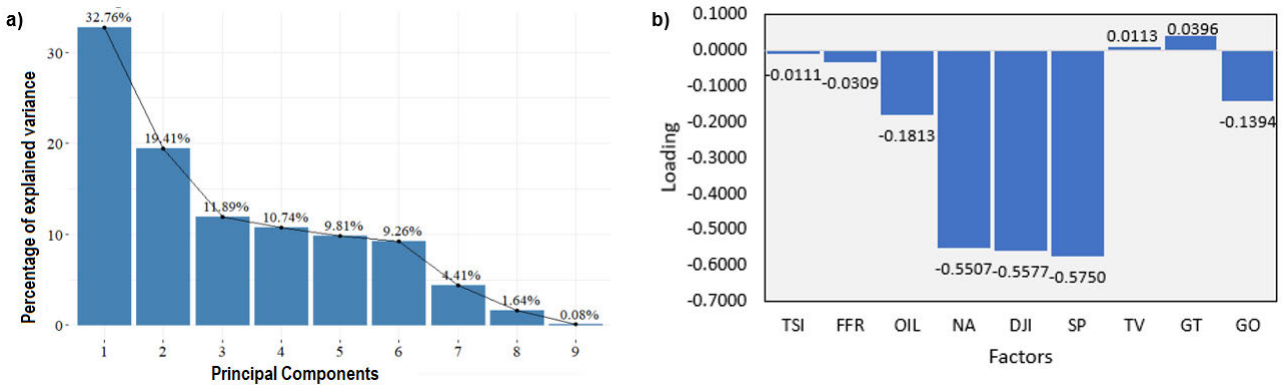

FIGURE 15. Principal component analysis (PCA) of exogeneous factors. (a) Scree plot of principal components and (b) Contribution of variables to 1st principal component observed.

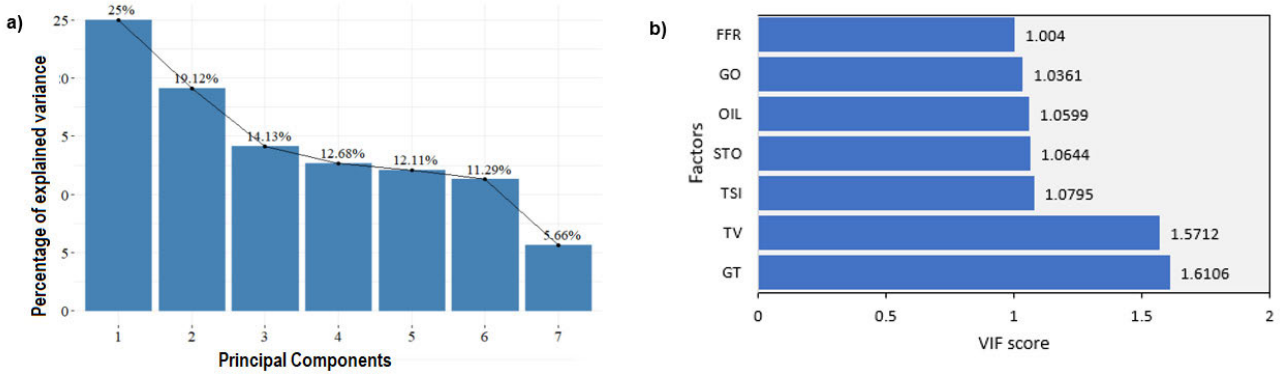

FIGURE 16. After combining factors using PCA factor loading, (a) Scree plot of principal components (b) VIF scores of exogeneous factors.

FIGURE 17. Correlation test after PCA.

15, and 20. The ARCH LM test, conducted to justify the inclusion of GARCHX in our model, revealed the presence of an ARCH effect (Table 5). This evidence prompted further examination of various GARCH and GARCHX models, integrated with the ARIMAX(1,1,1) model, by simulating p and q values within the same range (0,5). The result using train data indicated that ARIMAX(1,1,1) sGARCHX(1,3) yielded the lowest AIC and BIC scores. Similarly, when evaluated using test data, this model combination resulted in the lowest RMSE, MAE, and MAPE (Table 6). Furthermore,

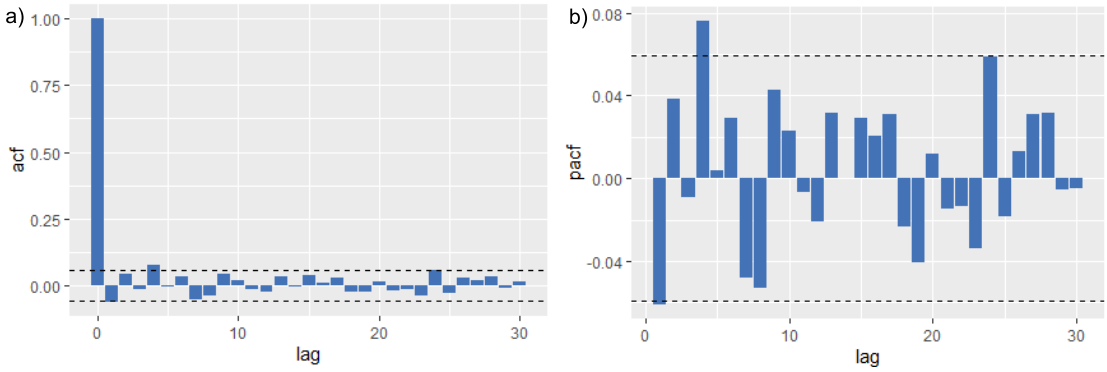

FIGURE 18. (a) Autocorrelation; (b) Partial autocorrelation plots of the bitcoin price.

TABLE 4. Tentative models.

Models	AIC	BIC	Error Forecasting		
			RMSE	MAE	MAPE
ARIMA(1,1,1)	−5,224.27	−5174.86	2894.99	2346.00	11.44
ARIMAX(1,1,1)	−6,217.40	−6,162.15	2150.79	1924.29	9.92

Note: AIC & BIC are using training data (n = 1,096); MAE, RMSE, and MAPE are using the test data (n = 184)

TABLE 5. Residual tests for arimax (1, 1, 1) model.

Lags	Ljung–Box Q Test		ARCH LM Test	
	Statistic	p-value	Statistic	p-value
Up to lag 10	9.80611	0.4574421	43.078	0.00000482
Up to lag 15	11.611583	0.7081592	45.904	0.00005508
Up to lag 20	16.150464	0.7072469	48.654	0.00034410

TABLE 6. Result of tentative models.

Models	AIC*	BIC*	Error Forecasting		
			RMSE**	MAE**	MAPE**
ARIMAX(1,1,1) sGARCH(1,1)	−5.7593	−5.7000	2202.93	2011.85	10.94
ARIMAX(1,1,1) sGARCH(1,3)	−5.7686	−5.7001	2204.96	2037.87	10.98
ARIMAX(1,1,1) sGARCHX(1,1)	−5.8208	−5.7296	2190.53	2016.68	10.61
ARIMAX(1,1,1) sGARCHX(1,3)	−5.9142	−5.8138	2129.75	1961.95	10.47

Note: AIC & BIC are using training data (n = 1,096); RMSE, MAE, and MAPE are using the test data (n = 184)

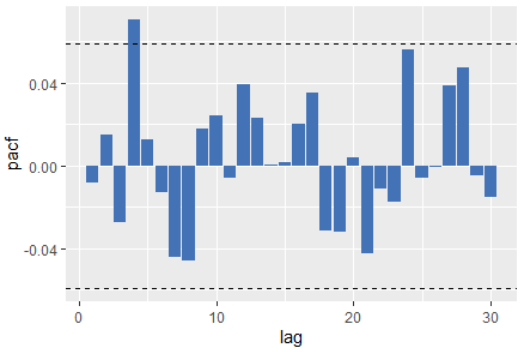

FIGURE 19. Partial autocorrelation plots of ARIMAX GARCH residuals.

confirming previous findings [42], the inclusion of exogenous factors in ARIMA and GARCH models enhances their prediction accuracy.

C. FITTING MODELS WITH LSTM

Using the residuals from the ARIMAX GARCHX model as the input layer and bitcoin daily closing price as the output layer, we constructed four LSTM models. Following the methodology outlined in [45], the window size for each LSTM model was determined based on lags that exceeded the 95% confidence interval in the PACF plot. From the PACF plot of our model residuals (Fig. 19), a lag value of 4 was used as the window size. Following [46], our LSTM model used the ReLU activation function in their hidden layers and a linear activation function in the final dense output layer, while Adam was used as LSTM optimizer.

To optimize the LSTM hyperparameters, a GWO with 80 iterations and a pack of 10 wolves was deployed [47] on the training data set. The results of this optimization process are comprehensively presented in Table 7. Compared to

TABLE 7. LSTM hyperparameters tuned using the grey wolf optimizer.

Hyperparameter	Range	Optimized value			
		LSTM	BiLSTM	Att-LSTM	Att-BiLSTM
lr.rate	[0.00001, 0.1]	0.0002	0.0003	0.0003	0.0003
batch size	[1,2,4,8,12,16, 36]	36	1	1	1
nodes	[1, 128]	50	54	42	33
epochs	[35, 50]	47	30	40	44
dropout.rt	[0.1, 0.9]	0.39	0.37	0.39	0.46

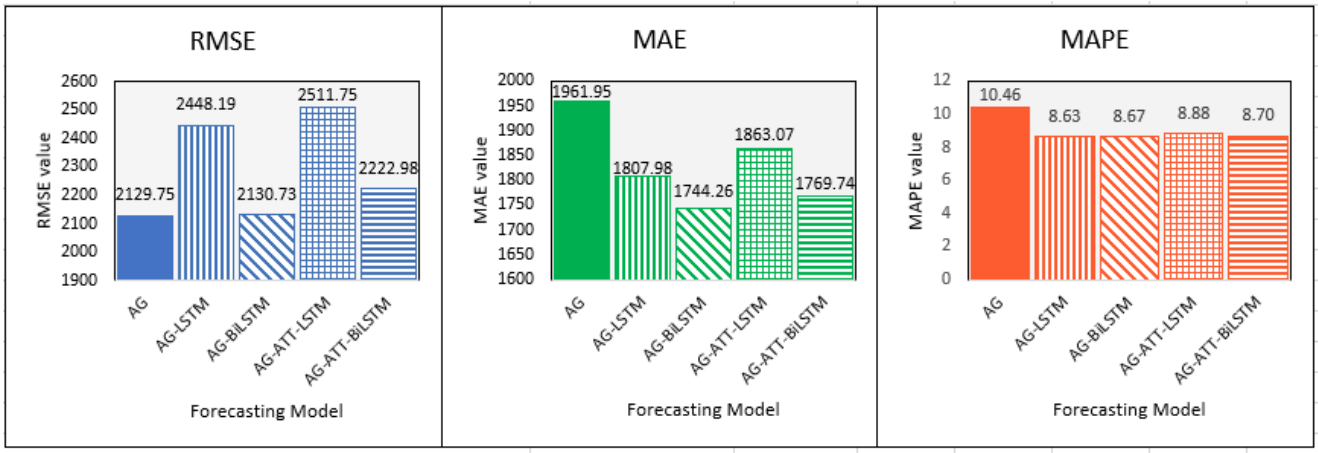

FIGURE 20. Comparison chart results using test data.

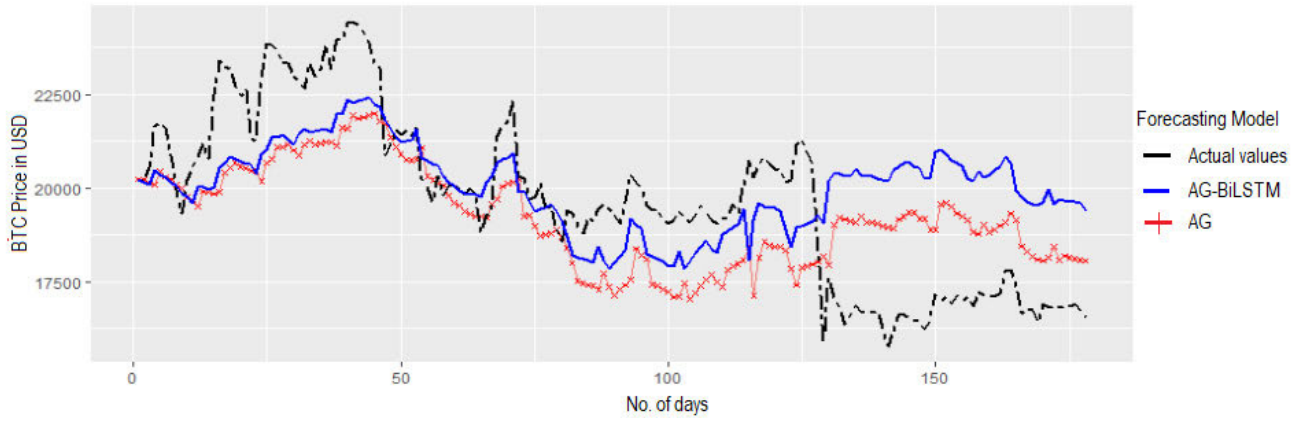

FIGURE 21. Forecasting plot using test data.

the standalone ARIMAX GARCHX (AG) model, the hybrid models, namely ARIMAX GARCHX LSTM (AG-LSTM), ARIMAX GARCHX BiLSTM (AG-BiLSTM), ARIMAX GARCHX ATT-LSTM (AG-ATT-LSTM), and ARIMAX GARCHX ATT-BiLSTM (AG-ATT-BiLSTM), demonstrated better predictive performance (Fig. 20). This signifies the ability of multivariate LSTM to extract features from non-linear components of time-series data, given that the data has undergone thorough preprocessing.

Among the evaluated models tested (Table 4, Table 6, and Fig. 20), the ARIMAX GARCHX BiLSTM model emerged as the most accurate, showcasing significantly lower RMSE and MAE values compared to its counterparts.

As depicted in Fig. 21, the forecasting accuracy deteriorated following a sudden drop around day 128 of the test data. Traditional econometric models, such as ARIMA, typically assume linear fluctuations in time-series data without structural breaks. This assumption falls short of capturing the reality of financial markets. Consequently, the performance of the ARIMAX GARCHX model, and by extension, the LSTM models that relied on its residuals, was negatively affected by extreme events.

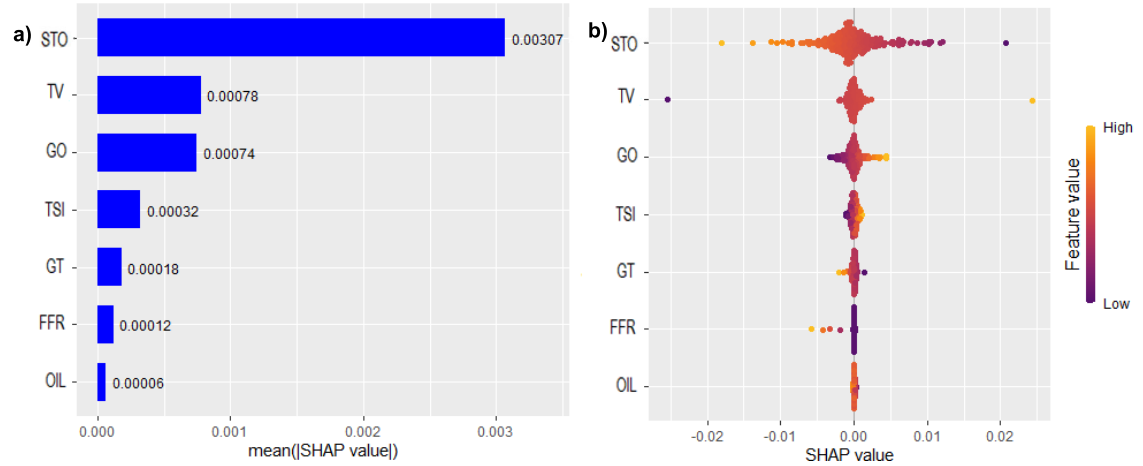

FIGURE 22. SHAP analysis of model data. (a) Overall importance of each feature on the model output, and (b) Influence of each feature value on the model output.

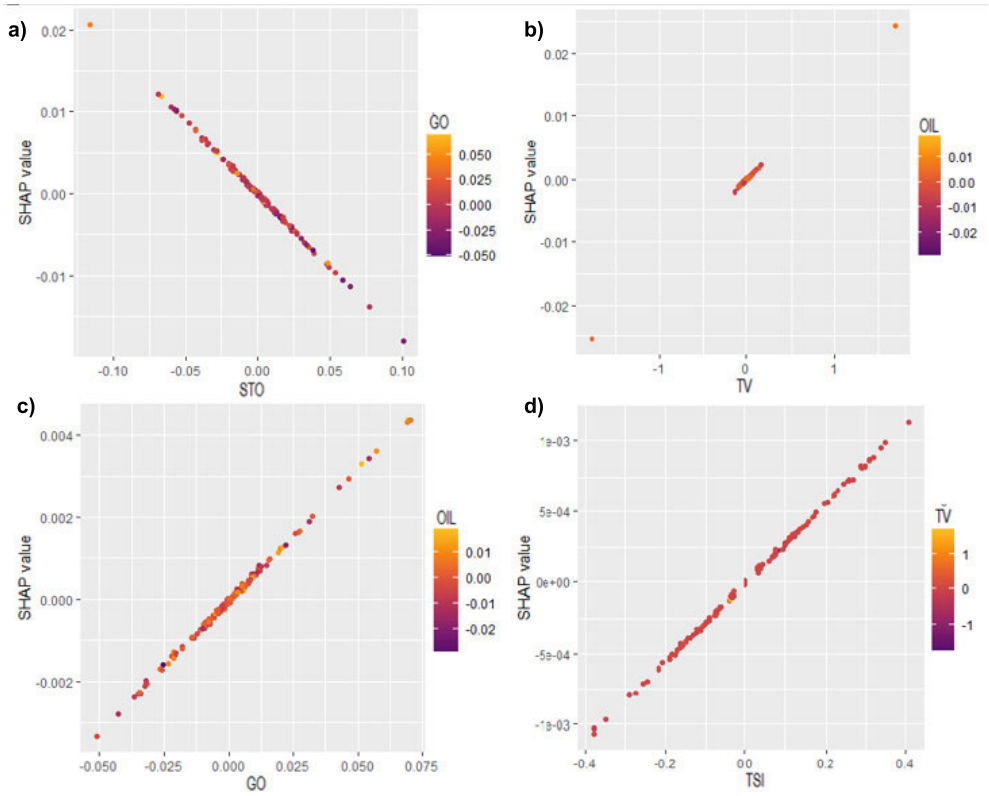

FIGURE 23. Dependence plot of (a) Stock price, (b) Twitter volume, (c) Gold price, and (d) Twitter sentiment index.

D. MODEL EXPLANATION USING SHAP

SHAP values were calculated using the ARIMAX GARCHX model, with the LSTM model utilizing the residuals. The global explanation highlighted stock indexes, Twitter volume, gold price, and Twitter sentiment index as the most influential factors (Fig 22a). Contrary to previous studies that emphasized the effect of economic factors on bitcoin price, our analysis found that federal funds rate, and oil price had minimal impact on the model output, ranking from positions 6–7 with mean SHAP values close to 0. This insight led to a model refinement proposal, suggesting the potential removal of these factors in future iterations. A detailed breakdown of how each feature's specific value influences the model predictions is illustrated in Fig 22b. For stock indexes, Google trends, and federal funds rate, high values correlate with high negative SHAP values, indicating an inverse relationship with bitcoin price. Conversely, high values of gold price and Twitter sentiment index correspond to positive SHAP values,

reflecting their positive association with bitcoin price. The trends for Twitter volume and oil price were less definitive. The observation that low values of stock index and federal funds rate lead to higher model outputs aligns with the notion that investors might turn to bitcoin when traditional investments and saving options appear less profitable. However, the positive correlation between high gold prices and high SHAP values warrants further investigation owing to its seemingly contradictory nature.

The dependence plot for a specific factor reveals the impact that a single factor has on the model predictions. Owing to its space constraints, we delve into only the four most important factors. In Fig 23a, stock index is identified as the primary factor, with its SHAP values plotted on the y-axis and GO, the secondary factor, represented on the color axis to highlight interaction effects between stock indexes and gold prices. When the stock index value is below 0, its impact on the model estimation of bitcoin price is positive. Conversely, when stock index exceeds 0, the impact turns negative, indicating a negative correlation between stock prices and bitcoin price. Furthermore, the presence of high positive gold prices alongside low stock values intensifies this relationship, and vice versa. High Twitter volume generally exhibits a positive correlation with bitcoin price, yet this trend may reverse in scenarios of significantly high oil prices (Fig. 23b). Dependence plots for both gold prices and the Twitter sentiment indexes reveal similar patterns, indicating their positive relationships with bitcoin price (Fig. 23c-d). The relationship of gold prices and prediction outputs are mostly influenced by positive oil price values, while the relationship between the Twitter sentiment index and prediction outputs are primarily influenced by negative Twitter volume values.

V. CONCLUSION

This paper presents a methodological approach aimed at predicting the closing price of bitcoin by leveraging both classical time-series models (ARIMAX, GARCHX) and advanced DL techniques (various LSTM variants). Among the tested models, the hybrid ARIMAX GARCHX BiLSTM model emerged as the most effective in forecasting bitcoin's closing price. To further validate and possible enhance the reliability of this selected model, future research could investigate alternative model selection methods, such as the model confidence set procedure. Additionally, a comprehensive evaluation of the model performance would be invaluable. Further research might compare forecasts from tentative models using the Diebold–Mariano test.

It is important to acknowledge the limitations of our model, notably its exclusion of certain exogenous factors that could potentially influence bitcoin price. Future studies could refine the model accuracy by incorporating additional factors highlighted in previous research. Despite these limitations, our study benefits from data preprocessing procedures designed to address non-stationarity, a straightforward predictive methodology, and successful tuning of LSTM hyperparameters. These strengths contribute to the model ability to generate accurate predictions, particularly in scenarios devoid of sudden, unexpected events.

The observed vulnerability of our model following a structural break event provides a clear direction for future research. It underscores the necessity to identify structural breaks in time-series data by using techniques, such as the iterative cumulative sum of squares algorithm and the Chow tests. Incorporating these structural breaks into forecasting models is crucial for enhancing their predictive accuracy. A recent study highlights the growing interest among investors in alternative cryptocurrencies, such as Ethereum, Ripple, Litecoin, Stellar, and Dash. This trend underscores the importance of developing a versatile model that is not limited to a single cryptocurrency. Future research should explore the application of systematic processes and algorithms across a broader range of cryptocurrencies. Additionally, given the effectiveness of the grey wolf optimizer in fine-tuning LSTM hyperparameters, it would be insightful to investigate its potential in optimizing other aspects of LSTM hyperparameters, such as the number of layers, window size, optimizer, activation function, and loss function). Ultimately, enhancing our understanding of how exogenous factors influence model predictions is vital. Exploring alternative techniques for assessing factor importance, such as DALEX and LIME, could provide deeper insights into the significance of these factors and their implications on model predictions.

TABLE 8. Acronyms.

Symbol	Description
ACF	AutoCorrelation Function
ADF	Augmented Dickey–Fuller
ARIMA	Autoregressive Integrated Moving Average
AIC	Akaike Information Criterion
ARMA	Autoregressive Moving Average
BIC	Bayesian Information Criterion
BiLSTM	Bidirectional LSTM
BTC	Bitcoin
DJI	Dow Jones Index
FFR	Federal Funds Rate
GARCH	Generalized Autoregressive Conditional Heteroskedasticity
GT	Google Trends
GWO	Grey Wolf Optimization
LSTM	Long Short-Term Memory
NA	NASDAQ Index
PACF	Partial AutoCorrelation Function
R	R is a programming language and open-source environment that is utilized for traditional statistical, machine learning, and deep learning computing and graphics
SHAP	Shapley Additive exPlanations

APPENDIX

See Table 8.

REFERENCES

[1] A. A. Abdelhamid, E. M. El-Kenawy, B. Alotaibi, G. M. Amer, M. Y. Abdelkader, A. Ibrahim, and M. M. Eid, "Robust speech emotion recognition using CNN+LSTM based on stochastic fractal search optimization algorithm," *IEEE Access*, vol. 10, pp. 49265–49284, 2022, doi: 10.1109/ACCESS.2022.3172954.

[2] W. Chen, H. Xu, L. Jia, and Y. Gao, "Machine learning model for Bitcoin exchange rate prediction using economic and technology determinants," *Int. J. Forecasting*, vol. 37, no. 1, pp. 28–43, Jan. 2021, doi: 10.1016/j.ijforecast.2020.02.008.

[3] A. Dutta, S. Kumar, and M. Basu, "A gated recurrent unit approach to Bitcoin price prediction," *J. Risk Financial Manage.*, vol. 13, no. 2, p. 23, Feb. 2020, doi: 10.3390/jrfm13020023.

[4] S. Hansun, A. Wicaksana, and A. Q. M. Khaliq, "Multivariate cryptocurrency prediction: Comparative analysis of three recurrent neural networks approaches," *J. Big Data*, vol. 9, no. 1, p. 50, Apr. 2022, doi: 10.1186/s40537-022-00601-7.

[5] F. Valencia, A. Gómez-Espinosa, and B. Valdés-Aguirre, "Price movement prediction of cryptocurrencies using sentiment analysis and machine learning," *Entropy*, vol. 21, no. 6, p. 589, Jun. 2019, doi: 10.3390/e21060589.

[6] H. Jang and J. Lee, "An empirical study on modeling and prediction of Bitcoin prices with Bayesian neural networks based on blockchain information," *IEEE Access*, vol. 6, pp. 5427–5437, 2018, doi: 10.1109/ACCESS.2017.2779181.

[7] X. Li and C. A. Wang, "The technology and economic determinants of cryptocurrency exchange rates: The case of Bitcoin," *Decis. Support Syst.*, vol. 95, pp. 49–60, Mar. 2017, doi: 10.1016/j.dss.2016.12.001.

[8] O. Poyser, "Exploring the determinants of Bitcoin's price: An application of Bayesian structural time series," 2017, *arXiv:1706.01437*.

[9] S. Palamalai, B. Maity, and K. Kumar, "Macro-financial parameters influencing Bitcoin prices: Evidence from symmetric and asymmetric ARDL models," *Rev. Econ. Anal.*, vol. 13, no. 3, pp. 143–175, Oct. 2021, doi: 10.15353/rea.v13i3.3585.

[10] T. Panagiotidis, T. Stengos, and O. Vravosinos, "On the determinants of Bitcoin returns: A LASSO approach," *Finance Res. Lett.*, vol. 27, pp. 235–240, Dec. 2018, doi: 10.1016/j.frl.2018.03.016.

[11] T. Panagiotidis, T. Stengos, and O. Vravosinos, "The effects of markets, uncertainty and search intensity on Bitcoin returns," *Int. Rev. Financial Anal.*, vol. 63, pp. 220–242, May 2019, doi: 10.1016/j.irfa.2018.11.002.

[12] S. Dastgir, E. Demir, G. Downing, G. Gozgor, and C. K. M. Lau, "The causal relationship between Bitcoin attention and Bitcoin returns: Evidence from the copula-based Granger causality test," *Finance Res. Lett.*, vol. 28, pp. 160–164, Mar. 2019, doi: 10.1016/j.frl.2018.04.019.

[13] D. Shen, A. Urquhart, and P. Wang, "Does Twitter predict Bitcoin?" *Econ. Lett.*, vol. 174, pp. 118–122, Jan. 2019, doi: 10.1016/j.econlet.2018.11.007.

[14] J. Abraham, D. W. Higdon, J. Nelson, and J. Ibarra, "Cryptocurrency price prediction using tweet volumes and sentiment analysis," *SMU Data Sci. Rev.*, vol. 1, no. 3, pp. 1–22, 2018.

[15] Y. B. Kim, J. G. Kim, W. Kim, J. H. Im, T. H. Kim, S. J. Kang, and C. H. Kim, "Predicting fluctuations in cryptocurrency transactions based on user comments and replies," *PLoS ONE*, vol. 11, no. 8, Aug. 2016, Art. no. e0161197.

[16] E. Bouri, P. Molnár, G. Azzi, D. Roubaud, and L. I. Hagfors, "On the Hedge and safe haven properties of Bitcoin: Is it really more than a diversifier?" *Finance Res. Lett.*, vol. 20, pp. 192–198, Feb. 2017.

[17] Y. Zhu, D. Dickinson, and J. Li, "Analysis on the influence factors of Bitcoin's price based on VEC model," *Financial Innov.*, vol. 3, no. 1, pp. 1–13, Dec. 2017, doi: 10.1186/s40854-017-0054-0.

[18] S. Pyo and J. Lee, "Do FOMC and macroeconomic announcements affect Bitcoin prices?" *Finance Res. Lett.*, vol. 37, Nov. 2020, Art. no. 101386, doi: 10.1016/j.frl.2019.101386.

[19] J. Wang, F. Ma, E. Bouri, and Y. Guo, "Which factors drive Bitcoin volatility: Macroeconomic, technical, or both?" *J. Forecasting*, vol. 42, no. 4, pp. 970–988, Nov. 2022, doi: 10.1002/for.2930.

[20] G. P. Zhang, "Time series forecasting using a hybrid ARIMA and neural network model," *Neurocomputing*, vol. 50, pp. 159–175, Jan. 2003, doi: 10.1016/s0925-2312(01)00702-0.

[21] R. J. Brenner, R. H. Harjes, and K. F. Kroner, "Another look at models of the short-term interest rate," *J. Financial Quant. Anal.*, vol. 31, no. 1, pp. 85–107, Mar. 1996.

[22] M. Yeasin, K. Singh, A. Lama, and R. Paul, "Modelling volatility influenced by exogenous factors using an improved GARCH-X model," *J. Indian Soc. Agricult. Statistic*, vol. 74, pp. 209–216, Dec. 2020.

[23] M. M. Patel, S. Tanwar, R. Gupta, and N. Kumar, "A deep learning-based cryptocurrency price prediction scheme for financial institutions," *J. Inf. Secur. Appl.*, vol. 55, Dec. 2020, Art. no. 102583, doi: 10.1016/j.jisa.2020.102583.

[24] L. Yu, S. Liang, R. Chen, and K. K. Lai, "Predicting monthly biofuel production using a hybrid ensemble forecasting methodology," *Int. J. Forecasting*, vol. 38, no. 1, pp. 3–20, Jan. 2022, doi: 10.1016/j.ijforecast.2019.08.014.

[25] A. N.-L. Huynh, R. C. Deo, D.-A. An-Vo, M. Ali, N. Raj, and S. Abdulla, "Near real-time global solar radiation forecasting at multiple time-step horizons using the long short-term memory network," *Energies*, vol. 13, no. 14, p. 3517, Jul. 2020, doi: 10.3390/en13143517.

[26] M. Schuster and K. K. Paliwal, "Bidirectional recurrent neural networks," *IEEE Trans. Signal Process.*, vol. 45, no. 11, pp. 2673–2681, Nov. 1997, doi: 10.1109/78.650093.

[27] D. Bahdanau, K. Cho, and Y. Bengio, "Neural machine translation by jointly learning to align and translate," 2014, *arXiv:1409.0473*.

[28] J. S. Bridle, "Probabilistic interpretation of feedforward classification network outputs, with relationships to statistical pattern recognition," in *Neurocomputing*, F. F. Soulié and J. Hérault, Eds. Berlin, Germany: Springer, 1990, pp. 227–236.

[29] A. A. Sekh, D. P. Dogra, S. Kar, P. P. Roy, and D. K. Prasad, "ELM-HTM guided bio-inspired unsupervised learning for anomalous trajectory classification," *Cogn. Syst. Res.*, vol. 63, pp. 30–41, Oct. 2020, doi: 10.1016/j.cogsys.2020.04.003.

[30] A. Mardjo and C. Choksuchat, "HyVADRF: Hybrid VADER–random forest and GWO for Bitcoin tweet sentiment analysis," *IEEE Access*, vol. 10, pp. 101889–101897, 2022, doi: 10.1109/ACCESS.2022.3209662.

[31] J. Hu, T. Zhou, S. Ma, D. Yang, M. Guo, and P. Huang, "Rock mass classification prediction model using heuristic algorithms and support vector machines: A case study of chambishi copper mine," *Sci. Rep.*, vol. 12, no. 1, p. 928, Jan. 2022, doi: 10.1038/s41598-022-05027-y.

[32] J. Batra, R. Jain, V. A. Tikkiwal, and A. Chakraborty, "A comprehensive study of spam detection in e-mails using bio-inspired optimization techniques," *Int. J. Inf. Manage. Data Insights*, vol. 1, no. 1, Apr. 2021, Art. no. 100006, doi: 10.1016/j.jjimei.2020.100006.

[33] J.-S. Pan, P. Hu, and S.-C. Chu, "Novel parallel heterogeneous meta-heuristic and its communication strategies for the prediction of wind power," *Processes*, vol. 7, no. 11, p. 845, Nov. 2019, doi: 10.3390/pr7110845.

[34] R. I. Hamilton and P. N. Papadopoulos, "Using SHAP values and machine learning to understand trends in the transient stability limit," 2023, *arXiv:2302.06274*.

[35] J. Chen, "Analysis of Bitcoin price prediction using machine learning," *J. Risk Financial Manage.*, vol. 16, no. 1, p. 51, Jan. 2023, doi: 10.3390/jrfm16010051.

[36] H. Mao, S. Counts, and J. Bollen, "Predicting financial markets: Comparing survey, news, Twitter and search engine data," 2011, *arXiv:1112.1051*.

[37] N. M. Noor, M. M. A. B. Abdullah, A. S. Yahaya, and N. A. Ramli, "Comparison of linear interpolation method and mean method to replace the missing values in environmental data set," *Mater. Sci. Forum*, vol. 803, pp. 278–281, Aug. 2014, doi: 10.4028/www.scientific.net/msf.803.278.

[38] I. E. Livieris, N. Kiriakidou, S. Stavroyiannis, and P. Pintelas, "An advanced CNN-LSTM model for cryptocurrency forecasting," *Electronics*, vol. 10, no. 3, p. 287, Jan. 2021, doi: 10.3390/electronics10030287.

[39] I. E. Livieris, S. Stavroyiannis, E. Pintelas, and P. Pintelas, "A novel validation framework to enhance deep learning models in time-series forecasting," *Neural Comput. Appl.*, vol. 32, no. 23, pp. 17149–17167, Dec. 2020.

[40] L. Luo and C. Qi, "An analysis of the crucial indicators impacting the risk of terrorist attacks: A predictive perspective," *Saf. Sci.*, vol. 144, Dec. 2021, Art. no. 105442, doi: 10.1016/j.ssci.2021.105442.

[41] M. C. Chang and D. A. Dickey, "Recognizing overdifferenced time series," *J. Time Ser. Anal.*, vol. 15, no. 1, pp. 1–18, Jan. 1994, doi: 10.1111/j.1467-9892.1994.tb00173.x.

[42] M. Waqar, H. Dawood, P. Guo, M. B. Shahnawaz, and M. A. Ghazanfar, "Prediction of stock market by principal component analysis," in *Proc. 13th Int. Conf. Comput. Intell. Secur. (CIS)*, Dec. 2017, pp. 599–602.

[43] E. Nzayisenga and Y. Zhu, "The import trade forecasting model based on PCA: Evidence from Rwanda," *Open J. Statist.*, vol. 10, no. 4, pp. 678–693, Jan. 2020, doi: 10.4236/ojs.2020.104042.
[44] J. F. Hair, G. T. M. Hult, C. M. Ringle, and M. Sarstedt, *A Primer on Partial Least Squares Structural Equation Modeling (PLS-SEM)*, 2nd ed. Thousand Oaks, CA, USA: Sage, 2017.
[45] T. I. Okedi and A. C. Fisher, "Time series analysis and long short-term memory (LSTM) network prediction of BPV current density," *Energy Environ. Sci.*, vol. 14, no. 4, pp. 2408–2418, Apr. 2021, doi: 10.1039/d0ee02970j.
[46] M. Roondiwala, H. Patel, and S. Varma, "Predicting stock prices using LSTM," *Int. J. Sci. Res.*, vol. 6, pp. 1754–1756, Apr. 2017, doi: 10.21275/ART20172755.
[47] M. M. Eid, E.-S.-M. El-Kenawy, N. Khodadadi, S. Mirjalili, E. Khodadadi, M. Abotaleb, A. H. Alharbi, A. A. Abdelhamid, A. Ibrahim, G. M. Amer, A. Kadi, and D. S. Khafaga, "Meta-heuristic optimization of LSTM-based deep network for boosting the prediction of monkeypox cases," *Mathematics*, vol. 10, no. 20, p. 3845, Oct. 2022, doi: 10.3390/math10203845.

ANNY MARDJO received the bachelor's degree in business administration from RMIT University, Melbourne, Australia, and the master's degree in information technology from Swinburne University, Melbourne. She is currently pursuing the Ph.D. degree with the College of Digital Science, Prince of Songkla University, Hat Yai Campus, Thailand. Presently, she is a Lecturer with the Faculty of Commerce and Management, Prince of Songkla University, Trang Campus, Thailand. Her research interests include internet-based applications and data analysis.

CHIDCHANOK CHOKSUCHAT (Member, IEEE) received the Ph.D. degree in computer and information science from Silpakorn University. She is currently an Assistant Professor with the Division of Computational Science, Faculty of Science, Prince of Songkla University, Hat Yai Campus, Thailand. She is also with the Office of Digital Innovation and Intelligent Systems, Prince of Songkla University. Her research interests include parallel processing, ontology engineering, web services, linked open data, data science toolkits, and the Internet of Things.