Cryptocurrency Price Forecasting Using XGBoost Regressor and Technical Indicators

Abdelatif Hafid[1], Maad Ebrahim[2], Ali Alfatemi[3], Mohamed Rahouti[3], and Diogo Oliveira[4]
[1]*ESISA Analytica, Higher School of Engineering in Applied Sciences, Fez, Morocco*
abdelatif.hafid@yahoo.com
[2]*GAIA, Ericsson, Montreal, Canada*
maad.ebrahim@ericsson.com
[3]*Computer and Information Science, Fordham University, NY 10023, USA.*
mrahouti@fordham.edu; aalfatemi@fordham.edu
[4]*College of IST, Penn State University, Monaca, PA 15601, USA.*
dko5179@psu.edu

arXiv:2407.11786v1 [cs.LG] 16 Jul 2024

***Abstract*—The rapid growth of the stock market has attracted many investors due to its potential for significant profits. However, predicting stock prices accurately is difficult because financial markets are complex and constantly changing. This is especially true for the cryptocurrency market, which is known for its extreme volatility, making it challenging for traders and investors to make wise and profitable decisions. This study introduces a machine learning approach to predict cryptocurrency prices. Specifically, we make use of important technical indicators such as Exponential Moving Average (EMA) and Moving Average Convergence Divergence (MACD) to train and feed the XGBoost regressor model. We demonstrate our approach through an analysis focusing on the closing prices of Bitcoin cryptocurrency. We evaluate the model's performance through various simulations, showing promising results that suggest its usefulness in aiding/guiding cryptocurrency traders and investors in dynamic market conditions.**

***Index Terms*—Artificial intelligence, Bitcoin, Machine learning, Market forecasting, Price prediction, Regression analysis, XGBoost.**

I. Introduction

Over the past few years, the rapid expansion of the stock market has made it an appealing option for investors seeking high returns and easy access. However, investing in stocks carries inherent risks, underscoring the need for a well-defined investment strategy. Traditionally, investors relied on empirical methods such as technical analysis, guided by financial expertise. With the widespread adoption of financial technology (FinTech), statistical models incorporating machine learning techniques have emerged for forecasting stock price movements. This shift has demonstrated significant success across various markets, including the S&P 500, NASDAQ [1], and the cryptocurrency market [2], [3]. In this research, our emphasis is on the cryptocurrency market, a dynamic force in finance, with a particular focus on Bitcoin price prediction [4].

Furthermore, Blockchain technology, the backbone of cryptocurrencies, has gained substantial attention in the banking and financial industry due to its secure and transparent decentralized database [5]. Despite the advantages of abundant market data and continuous trading, the cryptocurrency market faces challenges such as high price volatility and relatively smaller capitalization. Success in cryptocurrency financial trading hinges on the careful analysis and selection of data, making the development of machine learning models crucial for extracting meaningful insights. Models such as Long Short Term Memory (LSTM) and Random Forest (RF) are instrumental in predicting cryptocurrency prices by leveraging historical data and patterns, thereby aiding effective decision-making in this volatile market. Despite the potential, there have been limited studies attempting to create successful trading strategies in the cryptocurrency market.

With the advent of FinTech, machine learning models have been increasingly adopted to forecast stock price movements, transforming the landscape of financial analysis and trading. These models leverage large datasets and complex algorithms to identify patterns and predict future price trends, which has led to notable success across various markets, including the S&P 500 and NASDAQ [1]. In the cryptocurrency market, which is characterized by its high volatility and rapid price fluctuations, machine learning techniques have proven particularly valuable. Studies have demonstrated the efficacy of deep learning methods, such as Stacked Denoising Autoencoders (SDAE) and LSTM networks, in predicting Bitcoin prices with high accuracy [2], [3]. These models utilize a variety of inputs, including historical price data, trading volume, public sentiment, and macroeconomic indicators, to generate predictions that can guide investment decisions. The integration of machine learning into FinTech has thus provided investors with powerful tools to navigate the complexities of financial markets, enhancing their ability to make informed and strategic trading decisions.

Despite the advantages of the cryptocurrency market, such as abundant market data and continuous trading, it faces significant challenges like high price volatility and relatively smaller capitalization. Successful trading in this market depends on careful data analysis and selection, making the development of machine learning models crucial for extracting meaningful insights. Models like LSTM and RF are instrumental in predicting cryptocurrency prices by utilizing historical data and patterns, thus aiding effective decision-making in this volatile landscape. While there have been limited studies on developing successful trading strategies in the cryptocurrency market, our research aims to bridge this gap by introducing a novel machine learning strategy using the XGBoost regressor model, which incorporates essential technical indicators and historical data to enhance financial trading strategies.

This research introduces an efficient machine learning approach for forecasting cryptocurrency prices, specifically focusing on Bitcoin. The motivation behind this study stems from the inherent volatility and complexity of the cryptocurrency market, which pose significant challenges for traders and investors. Traditional methods of technical analysis and empirical strategies are often insufficient in predicting price movements in such a dynamic environment. To address this, we propose using the XGBoost regressor model, a powerful machine learning technique known for its robustness and accuracy. Our methodology integrates a comprehensive set of technical indicators, including the Exponential Moving Average (EMA), Moving Average Convergence Divergence (MACD), Relative Strength Index (RSI), and other relevant metrics derived from historical market data. The data is sourced from Binance via its API, covering a detailed time span with high-frequency intervals, which allows for capturing rapid market changes.

The proposed model undergoes extensive preprocessing and feature engineering to enhance its predictive capabilities. By employing regularization techniques, we mitigate the risk of overfitting and fine-tune the model parameters through a grid search for optimal performance. Our results demonstrate that the XGBoost regressor model significantly improves prediction accuracy, evidenced by low Mean Absolute Error (MAE) and Root Mean Squared Error (RMSE) values, as well as a near-perfect R-squared value. This study contributes to the state-of-the-art by providing a robust and scalable solution for cryptocurrency price prediction, leveraging advanced machine learning techniques to navigate the complexities of financial markets and aiding in informed decision-making for traders and investors.

The key contributions of this paper are summarized as follows:

- Introduce an efficient machine learning strategy using the XGBoost regressor model for cryptocurrency price prediction.
- Integrate a comprehensive set of technical indicators, including EMA, MACD, RSI, and others, with historical market data.
- Employ regularization techniques to mitigate overfitting and fine-tuned model parameters through grid search.
- Demonstrate significant improvements in prediction accuracy with low MAE, RMSE, and a near-perfect R-squared value.
- Provide a robust and scalable solution for navigating the complexities of financial markets, aiding informed decision-making for traders and investors.

The rest of the paper is organized as follows. Section II reviews the most relevant existing works. Section III explains how we collected and prepared the data. Section IV proposes the machine learning model and its mathematical formulation. In Section V, we evaluate/assess our proposed model. Section V also provides a comparison between the proposed work and existing studies in the literature. Finally, Section VI concludes the paper.

II. Related Work

The effort to forecast cryptocurrency prices has garnered significant interest in recent years, leading to the development of various methods to address this complex problem [6]. This section reviews advanced studies that employ machine learning for predicting cryptocurrency prices, with a particular focus on Bitcoin due to its dominant position and the extensive availability of data.

Among these advancements, machine learning has significantly impacted cryptocurrency price forecasting by providing models that adeptly navigate the complex and volatile digital currency market [7]. These methods range from simple regression models to advanced deep learning networks, each capable of detecting patterns and predicting future prices based on historical data [8].

Cryptocurrency value fluctuations are influenced by numerous factors, which has prompted the adoption of machine learning for price prediction [9], [10]. For instance, studies by Greaves and AU [11] have investigated using network attributes and machine learning to predict Bitcoin prices. Similarly, Jang and Lee [12] combined blockchain-related features, time series analysis, and Bayesian neural networks (BNNs) for Bitcoin price analysis.

Building on this foundation, further research by [13], [14], and [15] has applied machine learning to Bitcoin price forecasting. Saad et al. [15] not only predicted prices but also identified critical network attributes and user behaviors influencing price variations in Bitcoin and Ethereum [16], alongside the supply and demand dynamics of cryptocurrencies. Additionally, Sin and Wang [17] utilized neural networks for price predictions, leveraging blockchain data features.

Continuing this trend, Christoforou et al. [18] developed a Bitcoin price prediction model using neural networks, focusing on factors affecting price volatility and utilizing blockchain data and network activity metrics for forecasting. Furthermore, Chen et al. [19] and Akyildirim et al. [20] demonstrated the application of machine learning in forecasting Bitcoin prices and mid-price movement of Bitcoin futures, respectively. These studies highlight the ability of machine learning to harness vast datasets and identify complex patterns, enhancing predictive accuracy beyond traditional statistical approaches.

Moreover, some studies have demonstrated the effectiveness of combining machine learning techniques with blockchain data for cryptocurrency price forecasting. For example, Martin et al. [21] introduced a hybrid method that merges diverse data and analytical techniques, enhancing accuracy in this complex field. Liu et al. [22] focused on optimizing performance and interpretability in financial time series, showcasing the benefits of combining various machine learning approaches. He et al. [23] developed a deep learning ensemble model for financial time series forecasting, applicable to cryptocurrencies, illus-

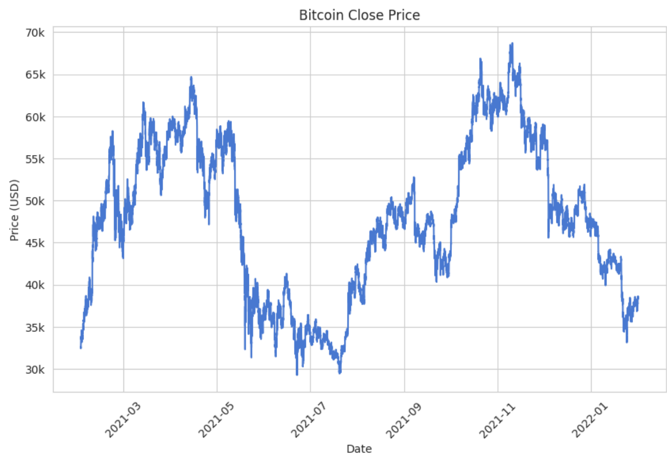

Fig. 1: Bitcoin close price over time.

trating the increased reliability and accuracy of multiple deep learning [24] strategies.

Additionally, Nazareth and Reddy [8] reviewed machine learning in finance, highlighting hybrid models' effectiveness in handling financial market complexities. Further research by Nagula and Alexakis [25], Petrovic et al. [26], Gupta and Nalavade [27], and Luo et al. [28] underscores the success of diverse computational techniques in improving Bitcoin price predictions, advancing sophisticated, accurate models for cryptocurrency investments.

In conclusion, machine learning not only excels in predictive accuracy but also in adaptability and scalability, both of which are essential as the cryptocurrency market evolves. With the capacity to update models with new data, machine learning remains a vital tool for cryptocurrency trading and investment, ensuring timely and precise forecasts [19], [20].

Unlike existing studies, our work introduces a novel machine learning strategy that leverages the XGBoost regressor model, combining a range of technical indicators such as EMA and MACD with historical data for Bitcoin price prediction. This approach emphasizes the use of regularization techniques to prevent overfitting and fine-tuning model parameters for enhanced accuracy. Our methodology stands out by effectively integrating diverse datasets and analytical techniques, ensuring robust and precise predictions in the highly volatile cryptocurrency market.

III. Data

This section covers the essential notations and abbreviations, explains the data collection process, details the preprocessing steps, and discusses the engineering of additional features. Table I provides the definitions of the parameters and abbreviations used in this paper.

A. Data Collection and Preprocessing

We obtained Bitcoin historical market data from Binance via the Binance API [29]. The dataset spans from February 1, 2021, to February 1, 2022, with a time interval of 15 minutes ($\Delta t = 15$ minutes). This interval was chosen for its balance between capturing detailed market fluctuations and maintaining accuracy. The data is split into 80% for the training set and 20% for the testing set. The choice of a shorter time interval is particularly important due to the high volatility of the Bitcoin market, where rapid changes are frequent. In such highly volatile markets, shorter intervals are essential for accurately capturing these swift price movements, unlike in less volatile markets where longer intervals might suffice.

Figure 1 illustrates the Bitcoin close price over time in USD. The x-axis represents dates, while the y-axis represents the price in USD. The plot provides a visual representation of the fluctuation in Bitcoin's closing price over the observed period, enabling insights into the cryptocurrency's price trend and volatility.

We take advantage of `StandardScaler` from `sklearn.preprocessing` module to scale the data. Let's denote the elements of the matrix X as x_{ij}, where i represents the row index (sample) and j represents the column index (feature). The transformation applied by the `StandardScaler` to each feature j is outlined as follows:

1) Compute the mean ($\mu_j = \frac{1}{m}\sum_{i=1}^{m} x_{ij}$) and standard deviation ($\sigma_j = \sqrt{\frac{1}{m}\sum_{i=1}^{m}(x_{ij} - \mu_j)^2}$) of feature j, where m is the number of samples (rows), x_{ij} is the element at the i-th row and j-th column of X.
2) Apply the transformation to each element of feature j:

$$x'_{ij} = \frac{x_{ij} - \mu_j}{\sigma_j}$$

where x'_{ij} is the scaled value of x_{ij}.

B. Feature Engineering

In this section, we elaborate on the various features incorporated in this case study, employing both historical market data and technical indicators.

1) Historical Data: In historical data analysis, we utilize various metrics to understand the behavior of Bitcoin prices within specific time periods. These metrics include:

- Open price ($\mathcal{O}_p$): The initial price of Bitcoin at the beginning of a specific time period.
- Highest price ($\mathcal{H}_p$): The maximum price of Bitcoin recorded during a time period.
- Lowest price ($\mathcal{L}_p$): The minimum price of Bitcoin recorded during a time period.
- Close price ($\mathcal{C}_p$): The final price of Bitcoin at the end of a time period.
- Trading volume ($\mathcal{V}$): The total number of Bitcoin traded within a time period.
- Quote Asset Volume (QAV): The total trading value of Bitcoin within a time period.
- Number of Trades (NOT): The total number of trades executed during a time period.
- Total Buy Base Volume (TBBV): The total volume of Bitcoin bought during a time period.
- Total Buy Quote Volume (TBQV): The total value of Bitcoin bought during a time period.

TABLE I: Notations and abbreviations used in the paper.

Notation	Description
m	Total number of samples or observations
m_{train}	Number of training samples
m_{test}	Number of testing samples
n	Number of features
$\mathcal{C}_p^{(i)}$	Close price at time t_i
$\mathcal{O}_p^{(i)}$	Opening price at time t_i
$\mathcal{H}_p^{(i)}$	High price at time t_i
$\mathcal{L}_p^{(i)}$	Low price at time t_i
$\mathcal{V}^{(i)}$	Volume of the cryptocurrency being traded at time t_i
$\text{QAV}^{(i)}$	Total trading value at time t_i
$\text{NOT}^{(i)}$	Number of trades at time t_i
$\text{TBBV}^{(i)}$	Total volume of Bitcoin bought at time t_i
$\text{RSI}_\alpha^{(i)}$	Relative strength index at time t_i within a time period α
$\text{MACD}^{(i)}$	Moving average convergence divergence at time t_i
$\text{EMA}_\alpha^{(i)}$	Exponential moving average at time t_i within a period of time α
$\text{PROC}_\alpha^{(i)}$	Price rate of change at time t_i within a period of time α
$\%K_\alpha^{(i)}$	Stochastic oscillator at time t_i within a period of time α
$\text{MOM}_\alpha^{(i)}$	Momentum at time t_i within a period of time α
η	Learning rate
λ, α	Regularization parameters
N	Number of trees
Δt	Time interval

2) Technical Indicators: Technical analysis indicators represent a trading discipline utilized to assess investments and pinpoint trading opportunities through the analysis of statistical trends derived from trading activities, including price movements and volume [30]. In this study, we explore indicators to feed our machine learning model, such as EMA, MACD, relative strength index, momentum, price rate of change, and stochastic oscillator.

We employ EMA with different periods, where EMA_{10}, EMA_{30}, and EMA_{200} represent the average price of Bitcoin over the last 10, 30, and 200 periods, respectively. To measure the magnitude of recent price changes and evaluate overbought or oversold conditions, we use RSI. Specifically, RSI_{10}, RSI_{14}, RSI_{30}, and RSI_{200} assess price changes over 10, 14, 30, and 200 periods, respectively. In addition, we apply Momentum (MOM) indicators to gauge the rate of change in Bitcoin prices, with MOM_{10} and MOM_{30} reflecting changes over the last 10 and 30 periods, respectively.

Furthermore, we incorporate MACD, a trend-following momentum indicator that illustrates the relationship between two moving averages of Bitcoin prices. Additionally, we use %K10, %K30, and %K200 as components of the stochastic oscillator, which compare the current price of Bitcoin to its price range over the last 10, 30, and 200 periods, respectively. Finally, we include the Percentage Rate of Change with 9 periods (PROC_9), measuring the percentage change in Bitcoin prices over the last 9 periods.

IV. Methodology

This section details the proposed methodology for our machine learning approach to cryptocurrency price forecasting. Algorithm 1 outlines our machine learning approach for cryptocurrency price forecasting using the XGBoost regressor model combined with various technical indicators such as EMA, MACD, RSI, and more. The process includes data collection and preprocessing, feature engineering, model training with hyperparameter tuning, and model evaluation. Details of this methodology are discussed in the following subsections.

Let $(x^{(i)}, y^{(i)})$ denotes a single sample/observation, and the set of samples is represented by:

$$\mathcal{S} = \left\{(x^{(1)}, y^{(1)}), (x^{(2)}, y^{(2)}), \ldots, (x^{(m)}, y^{(m)})\right\}$$

where $x^{(i)} \in \mathbb{R}^n$ and $y^{(i)} = \mathcal{C}_p^{(i)}$.

Considering both technical indicators and historical data for price prediction necessitates the integration of diverse datasets. To achieve this, we combine technical indicators and historical data as inputs to our model. The feature vector at a given time t can be expressed as follows:

Algorithm 1 Cryptocurrency price forecasting using XGBoost regressor and technical indicators.

Input: Historical market data $H = \{h_1, h_2, ..., h_T\}$, Technical indicators $T = \{t_1, t_2, ..., t_N\}$, Target variable $Y = \{y_1, y_2, ..., y_T\}$
Output: Trained XGBoost regressor model $\mathcal{M}$
procedure DATA PREPARATION
  // Collect and preprocess data
  $D \leftarrow$ Collect data from Binance API
  $D_{\text{train}}, D_{\text{test}} \leftarrow$ Split data into training and testing
  // Scale features
  $D_{\text{train}} \leftarrow$ StandardScaler.fit_transform(D_{train})
  $D_{\text{test}} \leftarrow$ StandardScaler.transform(D_{test})
end procedure
procedure FEATURE ENGINEERING
  // Extract historical data features
  $H \leftarrow$ Extract historical data features from D
  // Calculate technical indicators: TE
  $T \leftarrow$ Calculate TE (EMA, MACD, RSI, etc.)
  // Combine features
  $X \leftarrow H \cup T$
end procedure
procedure MODEL TRAINING
  // Initialize XGBoost regressor
  $\mathcal{M} \leftarrow$ XGBoost Regressor
  // Define hyperparameter grid
  $G \leftarrow \{\eta, D_{\text{max}}, N, \lambda, \alpha, \gamma, S, C\}$
  // Perform grid search with cross-validation
  $\theta \leftarrow$ GridSearchCV($\mathcal{M}, G,$ scoring=RMSE)
  // Train model with best hyperparameters
  $\mathcal{M} \leftarrow \mathcal{M}.fit(X_{\text{train}}, Y_{\text{train}})$
end procedure
procedure MODEL EVALUATION
  // Predict on test set
  $\hat{Y} \leftarrow \mathcal{M}.predict(X_{\text{test}})$
  // Calculate evaluation metrics
  MAE $\leftarrow \frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y}_i|$
  RMSE $\leftarrow \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}$
  $R^2 \leftarrow 1 - \frac{\sum_{i=1}^{n}(y_i-\hat{y}_i)^2}{\sum_{i=1}^{n}(y_i-\bar{y})^2}$
  // Display results
  Display (MAE, RMSE, R^2)
end procedure
return Trained model $\mathcal{M}$

$$\mathbf{x}^{(i)} = \begin{bmatrix} \mathcal{C}_p^{(i)} \\ \mathcal{V}^{(i)} \\ \text{QAV}^{(i)} \\ \text{NOT}^{(i)} \\ \text{TBBV}^{(i)} \\ \text{RSI}_{14}^{(i)} \\ \text{RSI}_{30}^{(i)} \\ \text{RSI}_{200}^{(i)} \\ \text{MOM}_{10}^{(i)} \\ \text{MOM}_{30}^{(i)} \\ \text{MACD}^{(i)} \\ \text{PROC}_{9}^{(i)} \\ \text{EMA}_{10}^{(i)} \\ \text{EMA}_{30}^{(i)} \\ \text{EMA}_{200}^{(i)} \\ \%K_{10}^{(i)} \\ \%K_{30}^{(i)} \\ \%K_{200}^{(i)} \end{bmatrix}, \quad \mathbf{x}^{(i)} \in \mathbb{R}^n \tag{1}$$

To extend the generality of our model, we stack all feature vectors into a matrix $\mathbf{X}$, which can be expressed as follows:

$$\mathbf{X} = \begin{array}{c} \begin{array}{cccc} \mathcal{C}_p & \mathcal{V} & \cdots & \%K_{200} \end{array} \\ \begin{bmatrix} x_{11} & x_{12} & \cdots & x_{1m} \\ x_{21} & x_{22} & \cdots & x_{2m} \\ x_{31} & x_{32} & \cdots & x_{3m} \\ \vdots & \vdots & \cdots & \vdots \\ x_{n-11} & x_{n-12} & \cdots & x_{n-1m} \\ x_{n1} & x_{n2} & \cdots & x_{nm} \end{bmatrix} \end{array} \tag{2}$$

Where:

$$\mathcal{C}_p = \begin{bmatrix} \mathcal{C}_p^{(1)} \\ \mathcal{C}_p^{(2)} \\ \vdots \\ \mathcal{C}_p^{(m)} \end{bmatrix}, \mathcal{V} = \begin{bmatrix} \mathcal{V}^{(1)} \\ \mathcal{V}^{(2)} \\ \vdots \\ \mathcal{V}^{(m)} \end{bmatrix}, \ldots, \%K_{200} = \begin{bmatrix} \%K_{200}^{(1)} \\ \%K_{200}^{(2)} \\ \vdots \\ \%K_{200}^{(m)} \end{bmatrix}$$

The output matrix can then be expressed as follows:

TABLE II: Parameter grid for GridSearchCV.

Parameter	Values
N	300, 400
η	0.01, 0.1, 0.2
$D_{\max}$	3, 4
W_{min}	1, 3
S	0.8, 1.0
C	0.8, 1.0
γ	0, 0.1
α	0.5, 1
λ	0.5, 1

$$Y = \begin{bmatrix} y^{(1)} \\ y^{(2)} \\ \vdots \\ y^{(m)} \end{bmatrix}$$

In this case study, the problem is to minimize the cost function for XGBoost regressor, which is a regularized finite-sum minimization problem defined as:

$$\min_{\Theta} J(\Theta) := \sum_{i=1}^{m_{\text{train}}} L(y_i, \hat{y}_i) + \sum_{k=1}^{K} \mathcal{R}(f_k) \tag{3}$$

Where:

- Θ represents the set of parameters to be learned during training.
- $L(y_i, \hat{y}_i)$ is the loss function that measures the difference between the true target value y_i and the predicted target balue $\hat{y}_i$ for the i-th instance. In the context of this case study, we employ the mean squared error (MSE) loss function, which is expressed as follows:

$$L(y_i, \hat{y}_i) = \sum_{i=1}^{n} (y_i - \hat{y}_i)^2 \tag{4}$$

  Here, y_i is the true target value for sample i, and $\hat{y}_i$ is the predicted target value for sample i.
- $\mathcal{R}(f_k)$ represents the regularization term for each tree to control its complexity. It typically includes both L_1 and L_2 regularization. Assuming T is the number of leaves in tree f_k and $w_{j,k}$ is the weight for leaf j in tree f_k, the regularization term for tree f_k is:

$$\mathcal{R}(f_k) = \gamma T + \frac{1}{2}\lambda \sum_{j=1}^{T} w_{j,k}^2 + \alpha \sum_{j=1}^{T} |w_{j,k}| \tag{5}$$

  The regularization terms ($\mathcal{R}(f_k)$) help control the complexity of individual trees in the ensemble, preventing overfitting.

During training, XGBoost regressor aims to find the set of parameters (Θ) that minimizes the overall cost function. The optimization is typically performed using techniques like gradient boosting, which involves iteratively adding weak learners to the ensemble to reduce the residual errors [31].

Table II presents a parameter grid used in GridSearchCV, a technique for hyperparameter tuning in machine learning models. Hyperparameters are predefined settings that control the learning process of algorithms. The table lists various hyperparameters commonly used in the XGBoost regressor model, a popular gradient boosting framework [31]. Each hyperparameter is accompanied by its corresponding values that are explored during the grid search process. For instance, N represents the number of estimators (trees) in the XGBoost model, with values of 300 and 400 being considered. Similarly, η denotes the learning rate, with potential values of 0.01, 0.1, and 0.2.

Other hyperparameters include $D_{\max}$ for maximum depth of trees, $W_{\min}$ for minimum child weight, S for subsampling ratio, C for column subsampling ratio, γ for minimum loss reduction required to make further splits, α for L1 regularization term on weights, and λ for L2 regularization term on weights.

This parameter grid serves as a roadmap for systematically exploring various combinations of hyperparameters to identify the optimal configuration for the XGBoost model, thereby enhancing its predictive performance. The best combination of hyperparameters for the XGBoost model was selected based on the smallest RMSE, resulting in enhanced predictive performance. The chosen parameters are as follows:

$$\Theta = \begin{pmatrix} C & \gamma & \eta & D_{\max} & W_{\min} & N & \alpha & \lambda & S \\ 0.8 & 0 & 0.2 & 4 & 3 & 300 & 1 & 0.5 & 1.0 \end{pmatrix}$$

Finally, the RMSE achieved with this parameter combination is the smallest observed during the hyperparameter tuning process.

V. Results and Analysis

In this section, we provide simulations-based evaluations of the proposed machine learning model. In particular, we compute the Mean Absolute Error (MAE), RMSE, and R-squared (R^2).

$$\text{MAE} = \frac{1}{n} \sum_{i=1}^{n} |y_i - \hat{y}_i| \tag{6}$$

MAE provides a simple and straightforward interpretation of the average absolute deviation between the predicted and actual values. It is easy to understand and is less sensitive to outliers compared to other metrics like RMSE.

$$\text{RMSE} = \sqrt{\frac{1}{n} \sum_{i=1}^{n} (y_i - \hat{y}_i)^2} \tag{7}$$

RMSE provides a measure of the average magnitude of prediction errors in the same units as the target variable. It penalizes larger errors more heavily than MAE, making it particularly useful when large errors are undesirable.

$$R^2 = 1 - \frac{\sum_{i=1}^{n} (y_i - \hat{y}_i)^2}{\sum_{i=1}^{n} (y_i - \bar{y})^2} \tag{8}$$

where $\bar{y}$ is the mean of the actual values of the target variable.

R^2 Score provides an indication of how well the model fits the data relative to a simple baseline model (e.g., a model that always predicts the mean). It ranges from 0 to 1, where

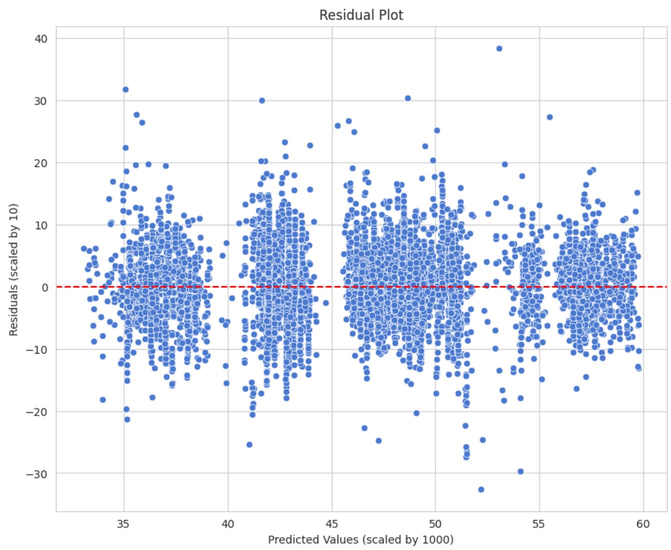

Fig. 2: Scatter plot showing the residuals against the predicted values.

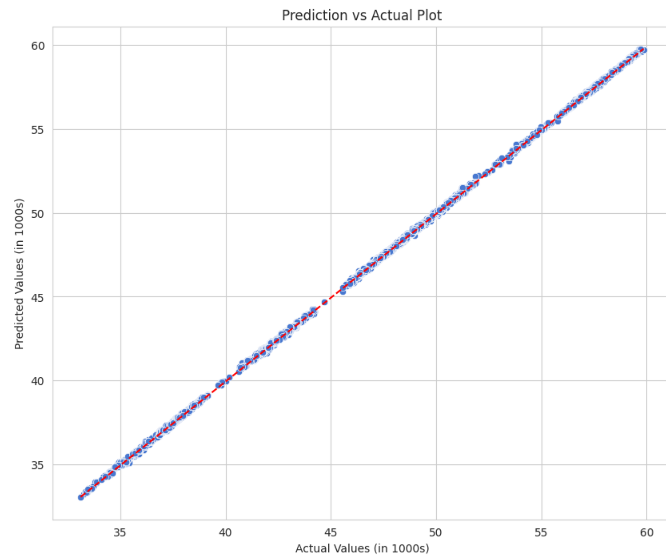

Fig. 3: Scatter plot of actual vs. predicted values.

higher values indicate a better fit. R^2 score is widely used for comparing different models and assessing overall model performance.

Metric	Value
RMSE	59.9504
MAE	46.2229
R^2	0.9999

TABLE III: Model evaluation metrics.

Table III presents key evaluation metrics for our regression model. The RMSE is 59.9504, indicating the square root of the average squared difference between predicted and actual values. The MAE is 46.2229, indicating the average absolute difference between predicted and actual values. The model's R^2 Score is 0.9999, reflecting an exceptionally strong fit to the data. Overall, the model demonstrates high accuracy and predictive capability, with low errors and a near-perfect R^2 score.

Another way to assess the performance of the XGBoost Regressor model is to analyze the relationship between the predicted values and the residuals. Let y_{test} be the true target values from the test dataset, $\hat{y}_{\text{pred}}$ be the predicted target values from the model, and ε be the residuals calculated as $\varepsilon = y_{\text{test}} - \hat{y}_{\text{pred}}$.

Figure 2 shows a scatter plot of the residuals against the predicted values. The plot displays the relationship between the predicted values (scaled by 1000) and the residuals (scaled by 10). A horizontal dashed line at $y = 0$ indicates perfect prediction, where residuals are centered around zero. The plot illustrates the model's ability to predict accurately across the range of predicted values.

Furthermore, Figure 3 presents a scatter plot depicting the comparison between predicted values (in 1000s) and actual values (in 1000s). The diagonal dashed red line represents ideal prediction, where actual values align perfectly with predicted values. This plot offers insight into the model's efficacy across the spectrum of actual values, showcasing its predictive performance.

A. State-of-the-Art Comparison

Lastly, this subsection provides a comparison between the work proposed in this paper and existing studies in the literature.

Table IV provides a comprehensive comparison of various machine learning approaches in financial forecasting and trading. Shynkevich et al. [32] leverage machine learning algorithms on daily stock price time series, achieving optimal performance by analyzing different forecast horizons and input window lengths. Similarly, Liu et al. [2] employ SDAE deep learning models utilizing historical data, public attention, and macroeconomic factors, which result in superior prediction accuracy. In addition, Jaquart et al. [33] implement ensemble machine learning models on cryptocurrency market data (streamed from CoinGecko [35]), producing statistically significant predictions and incorporating a long-short portfolio strategy. Furthermore, Hafid et al. [3] use a Random Forest classifier with historical data and a few technical indicators to achieve high accuracy in market trend prediction, effectively signaling buy and sell moments.

Saad et al. [15] integrate economic theories with machine learning, analyzing user and network activity to attain high accuracy in price prediction and offer insights into network dynamics. Moreover, Akyildirim et al. [34] apply SVM, LR, ANN, and RF algorithms on historical price data and technical indicators, demonstrating consistent predictive accuracy and trend predictability. In contrast, this paper introduces a novel approach using an XGBoost regressor with technical indicators and historical data, achieving low MAE, RMSE, and an R^2 value close to 1, thereby contributing a new machine learning strategy to the field.

TABLE IV: Comparison of machine learning approaches in financial forecasting and trading.

Paper	Methodology	Data Utilization	Model Performance	Contribution to Field
Shynkevich et al. [32]	Machine learning algorithms	Daily price time series for stocks	Optimal performance with varied metrics	Impact of forecast horizon and input window length
Liu et al. [2]	SDAE, deep learning	Historical data, public attention, macroeconomic environment	Superior performance in predictions	Improved prediction with SDAE, RMAE, and DA
Jaquart et al. [33]	Ensemble machine learning models	Cryptocurrency market data	Statistically significant predictions	Long-short portfolio strategy
Hafid et al. [3]	RF classifier	Historical data, few technical indicators	High accuracy in market trend prediction	Effective market trend prediction (buy & sell)
Saad et al. [15]	Integration of economic theories with machine learning	User and network activity	High accuracy in price prediction	Understanding network dynamics
Akyildirim et al. [34]	SVM, LR, ANN, RF	Historical price data, technical indicators	Predictive accuracy in price trends	Evidence of predictability in trends
This paper	XGBoost regressor	Technical indicators, historical data	Low MAE, RMSE, $R^2 \approx 1$	Novel machine learning strategy

VI. Conclusion

Our research highlights the efficacy of the XGBoost regressor model in forecasting Bitcoin prices using a combination of technical indicators and historical market data. The model's performance, as evidenced by the low Mean Absolute Error (MAE) and Root Mean Squared Error (RMSE) along with a near-perfect R^2 value, underscores its potential in providing accurate and reliable predictions in the highly volatile cryptocurrency market. By incorporating regularization techniques to mitigate overfitting and fine-tuning model parameters through an extensive grid search, we have achieved a robust predictive model. Furthermore, the use of various technical indicators such as the Exponential Moving Average (EMA), Moving Average Convergence Divergence (MACD), Relative Strength Index (RSI), and others, in conjunction with historical prices and volume data, has proven effective in enhancing the model's predictive capabilities. This approach not only offers a comprehensive analysis of market trends but also facilitates better decision-making for traders and investors.

This work contributes to the field of financial forecasting, particularly in the domain of cryptocurrency price prediction. The findings suggest that machine learning models, when properly calibrated and integrated with relevant technical indicators, can serve as powerful tools for navigating the complexities of financial markets. Future research could further explore the integration of additional data sources and advanced machine learning techniques to continue improving the accuracy and applicability of such models in dynamic trading environments.

References

[1] Y.-L. Hsu, Y.-C. Tsai, and C.-T. Li, "Fingat: Financial graph attention networks for recommending top-k k profitable stocks," *IEEE Transactions on Knowledge and Data Engineering*, vol. 35, no. 1, pp. 469–481, 2021.

[2] M. Liu, G. Li, J. Li, X. Zhu, and Y. Yao, "Forecasting the price of bitcoin using deep learning," *Finance research letters*, vol. 40, p. 101755, 2021.

[3] A. Hafid, A. S. Hafid, and D. Makrakis, "Bitcoin price prediction using machine learning and technical indicators," in *International Symposium on Distributed Computing and Artificial Intelligence*. Springer, 2023, pp. 275–284.

[4] S. Nakamoto, "Bitcoin: A peer-to-peer electronic cash system," *Decentralized business review*, 2008. [Online]. Available: http://dx.doi.org/10.2139/ssrn.3440802

[5] A. Hafid, A. S. Hafid, and M. Samih, "Scaling blockchains: A comprehensive survey," *IEEE access*, vol. 8, pp. 125 244–125 262, 2020.

[6] H. Sebastião and P. Godinho, "Forecasting and trading cryptocurrencies with machine learning under changing market conditions," *Financial Innovation*, vol. 7, no. 1, pp. 1–30, 2021.

[7] A. M. Khedr *et al.*, "Cryptocurrency price prediction using traditional statistical and machine-learning techniques: A survey," *Intelligent Systems in Accounting, Finance and Management*, vol. 28, no. 1, pp. 3–34, 2021.

[8] N. Nazareth and Y. V. R. Reddy, "Financial applications of machine learning: A literature review," *Expert Systems with Applications*, vol. 219, p. 119640, 2023.

[9] S. Tanwar *et al.*, "Machine learning adoption in blockchain-based smart applications: The challenges, and a way forward," *IEEE Access*, vol. 8, pp. 474–488, 2019.

[10] J. B. Awotunde *et al.*, "Machine learning algorithm for cryptocurrencies price prediction," in *Artificial Intelligence for Cyber Security: Methods, Issues and Possible Horizons or Opportunities*. Springer, 2021, pp. 421–447.

[11] A. Greaves and B. Au, "Using the bitcoin transaction graph to predict the price of bitcoin," *No Data*, 2015.

[12] 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, 2017.

[13] S. McNally *et al.*, "Predicting the price of bitcoin using machine learning," in *2018 26th euromicro international conference on parallel, distributed and network-based processing (PDP)*. IEEE, 2018, pp. 339–343.

[14] S. Velankar *et al.*, "Bitcoin price prediction using machine learning," in *2018 20th International Conference on Advanced Communication Technology (ICACT)*. IEEE, 2018, pp. 144–147.

[15] M. Saad *et al.*, "Toward characterizing blockchain-based cryptocurrencies for highly accurate predictions," *IEEE Systems Journal*, vol. 14, no. 1, pp. 321–332, 2019.

[16] G. Wood *et al.*, "Ethereum: A secure decentralised generalised transaction ledger," *Ethereum project yellow paper*, vol. 151, no. 2014, pp. 1–32, 2014.

[17] E. Sin and L. Wang, "Bitcoin price prediction using ensembles of neural networks," in *International Conference on Natural Computation, Fuzzy Systems and Knowledge Discovery (ICNC-FSKD)*. IEEE, 2017, pp. 666–671.

[18] E. Christoforou *et al.*, "Neural networks for cryptocurrency evaluation and price fluctuation forecasting," in *Mathematical Research for Blockchain Economy*. Springer, 2020, pp. 133–149.

[19] Z. Chen, C. Li, and W. Sun, "Bitcoin price prediction using machine learning: An approach to sample dimension engineering," *Journal of Computational and Applied Mathematics*, vol. 365, p. 112395, 2020. [Online]. Available: https://doi.org/10.1016/j.cam.2019.112395

[20] E. Akyildirim *et al.*, "Forecasting mid-price movement of bitcoin futures using machine learning," *Annals of Operations Research*, vol. 330, no. 1, pp. 553–584, 2023.

[21] K. Martin *et al.*, "Combining blockchain and machine learning to forecast cryptocurrency prices," in *International Conference on Blockchain Computing and Applications (BCCA)*. IEEE, 2020, pp. 52–58.

[22] S. Liu *et al.*, "Financial time-series forecasting: Towards synergizing performance and interpretability within a hybrid machine learning approach," *arXiv preprint arXiv:2401.00534*, 2023.

[23] K. He *et al.*, "Financial time series forecasting with the deep learning ensemble model," *Mathematics*, vol. 11, no. 4, p. 1054, 2023.

[24] A. Alfatemi, M. Rahouti, R. Amin, S. ALJamal, K. Xiong, and Y. Xin, "Advancing ddos attack detection: A synergistic approach using deep residual neural networks and synthetic oversampling," *arXiv preprint arXiv:2401.03116*, 2024.

[25] P. K. Nagula and C. Alexakis, "A new hybrid machine learning model for predicting the bitcoin (BTC-USD) price," *Journal of Behavioral and Experimental Finance*, vol. 36, p. 100741, 2022.

[26] A. Petrovic *et al.*, "Cryptocurrency price prediction by using hybrid machine learning and beetle antennae search approach," in *Telecommunications Forum (TELFOR)*. IEEE, 2021, pp. 1–4.

[27] R. Gupta and J. E. Nalavade, "Metaheuristic assisted hybrid classifier for bitcoin price prediction," *Cybernetics and Systems*, vol. 54, no. 7, pp. 1037–1061, 2023.

[28] C. Luo *et al.*, "Bitcoin price forecasting: an integrated approach using hybrid lstm-elm models," *Mathematical Problems in Engineering*, vol. 2022, 2022.

[29] "Data from binance api," https://www.binance.com/

[30] S. B. Achelis, "Technical analysis from a to z," 2001.

[31] T. Chen and C. Guestrin, "Xgboost: A scalable tree boosting system," in *Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining*, 2016, pp. 785–794.

[32] Y. Shynkevich, T. M. McGinnity, S. A. Coleman, A. Belatreche, and Y. Li, "Forecasting price movements using technical indicators: Investigating the impact of varying input window length," *Neurocomputing*, vol. 264, pp. 71–88, 2017.

[33] P. Jaquart, S. Köpke, and C. Weinhardt, "Machine learning for cryptocurrency market prediction and trading," *The Journal of Finance and Data Science*, vol. 8, pp. 331–352, 2022.

[34] E. Akyildirim, A. Goncu, and A. Sensoy, "Prediction of cryptocurrency returns using machine learning," *Annals of Operations Research*, vol. 297, pp. 3–36, 2021.

[35] "CoinGecko Methodology," https://www.coingecko.com/en/methodology, accessed: February 7, 2024.