Empirical Forecasting Analysis of Bitcoin Prices: A Comparison of Machine Learning, Deep Learning, and Ensemble Learning Models

Original Scientific Paper

Nrusingha Tripathy
Department of Computer Science and Engineering,
Siksha 'O' Anusandhan Deemed to be University,
Bhubaneswar, India
nrusinghatripathy654@gmail.com

Pranati Satapathy
Department of IMCA,
Utkal University,
Bhubaneswar, Odisha, India
satapathy.pranati@gmail.com

Sarbeswara Hota
Department of Computer Application,
Siksha 'O' Anusandhan Deemed to be University,
Bhubaneswar, India
sarbeswarahota@soa.ac.in

Subrat Kumar Nayak
Department of Computer Science and Engineering,
Siksha 'O' Anusandhan Deemed to be University,
Bhubaneswar, India
subratsilicon28@gmail.com

Debahuti Mishra
Department of Computer Science and Engineering,
Siksha 'O' Anusandhan Deemed to be University,
Bhubaneswar, India
debahutimishra@soa.ac.in

Abstract *– Bitcoin has drawn a lot of interest recently as a possible high-earning investment. There are significant financial risks associated with its erratic price volatility. Therefore, investors and decision-makers place great significance on being able to precisely foresee and capture shifting patterns in the Bitcoin market. However, empirical studies on the systems that support Bitcoin trading and forecasting are still in their infancy. The suggested method will predict the prices of all key cryptocurrencies with accuracy. A number of factors are going to be taken into account in order to precisely predict the pricing. By leveraging encryption technology, cryptocurrencies may serve as an online accounting framework and a medium of exchange. The main goal of this work is to predict Bitcoin price. To address the drawbacks of traditional forecasting techniques, we use a variety of machine learning, deep learning, and ensemble learning algorithms. We conduct a performance analysis of Auto-Regressive Integrated Moving Averages (ARIMA), Long-Short-Term Memory (LSTM), FB-Prophet, XGBoost, and a pair of hybrid formulations, LSTM-GRU and LSTM-1D_CNN. Utilizing historical Bitcoin data from 2012 to 2020, we compared the models with their Mean Absolute Error (MAE) and Root Mean Squared Error (RMSE). The hybrid LSTM-GRU model outperforms the rest with a Mean Absolute Error (MAE) of 0.464 and a Root Mean Squared Error (RMSE) of 0.323. The finding has significant ramifications for market analysts and investors in digital currencies.*

Keywords*: Bitcoin, Cryptocurrency, Arima, LSTM, Prophet, XGBOOST, Hybrid model, Prediction*

1. INTRODUCTION

Using freely available software and peer-to-peer networks, the digital currency known as Bitcoin was developed to serve as a private, untraceable payment system because Bitcoin is not backed by any institution or government and has no physical existence. The concept of cryptocurrencies, or digital currency, served as the inspiration for Bitcoin. Cryptocurrencies are digital, computer-generated money that only exists online. Cryptocurrency uses cryptographic methods to secure transactions and is designed to be used as a means of exchange [1]. Today's financial systems employ cryptocurrencies, which are virtual or digital money. All cryptocurrencies are built on very complicated blockchain technology, which promises to stock data in a manner that is dreadful to hack and modify. It is difficult to create phony cryptocurrencies because of the additional security provided by cryptography for these currencies. El Salvador became the first country to recognize Bitcoin under the US dollar as legal money in November 2021 [2], even though cryptocurrencies are still difficult to forecast if they will ever be extensively used in international markets. The most appealing market for financial speculation at the moment is the one for cryptocurrency. By making use of digital market speculators and marketing investment schemes that incorporate Bitcoins, Ethereum, and other cryptocurrencies yet are replete with unforeseen traps, many people have gained significant sums of money.

However, cryptocurrencies are extremely volatile. The element that raises the volatility of cryptocurrencies is their smaller market cap than that of equities. Due to the still-small size of the cryptocurrency market, even a tiny investment can have a significant impact on a cryptocurrency's price. Most trading algorithms include various parameters, largely rely on historical data, and fail to handle high volatility [3, 4]. There are several methods for them such as ARIMA and LSTM-based on Recurrent Neural Networks (RNNs). Nevertheless, understanding their behaviour is difficult because each of them requires a lot of parameters.

These algorithms work well for patterns that persist, but in the world of cryptocurrencies, trends and patterns may change quickly. To make more accurate forecasts, an algorithm can handle all of these problems including holidays that are identified in advance, lacking observations, substantial outliers, and seasonal impacts carried on by humans. The main contributions of the paper are summarised as follows:

- We take the historical Bitcoin data of the past eight years, from January 2012 to September 2020. Initially, the data is pre-processed and we make the data balanced.
- The hybrid LSTM-GRU model gives the best result by taking RMSE and MAE as parameters, after that with the help of our best model we go for future prediction.
- Various machine learning and deep learning methods were investigated, but the hybrid LSTM-GRU model performs remarkably well.

2. LITERATURE SURVEY

Given the fact that Bitcoin is still a relatively new technology, there aren't many models available for predicting its price. Time series models interact with information gathered from regular time sequences, 10-minute intervals, and intervals of ten seconds. These models produced three sets of time series data for periods of 30, 60, and 120 minutes, and then three linear models utilizing GLM/Random Forest were produced from the datasets. For predicting the price of Bitcoin, all three of these models are linearly constructed from information obtained from daily time sequences, 10-minute periods, and 10-second intervals [5].

Forecasting the stock's trend instead of predicting its price in the years, a pattern may be deduced from the trend. These might offer both quick forecasts (for a day or a week) and extended ones (for months). The longer projections were shown to be more accurate (79%) and to have better outcomes. The network's achievement of the evaluation standard, which is based on expected output, is also taken into consideration in the research. For Bitcoin prediction, the authors employ machine learning strategies that include gradient descent, linear search, and both deep learning and regression approaches. They are not founded on the same data-related assumptions as ARIMA and Prophet, including time series stationary motion or the availability of a Date field. The disadvantage of LSTM-based RNNs is that it is difficult to comprehend and anticipate how they will behave. To acquire good results, meticulous hyperparameter adjustment is also required. The accuracy of the LSTM is significantly impacted by substantial simultaneous seasonality in the context of cryptocurrency [6].

The cryptocurrency trading market changes rapidly over only a brief amount of time even though little is known about its characteristics, periodicity can occur on a daily, weekly, or even every-hour basis on average, data contains large outliers, and it doesn't just depend on historical data. Each of them could have a detrimental influence on how well the ARIMA model performs. For techniques like ARIMA to provide decent results, further tinkering may be necessary, which is typically out of scope for many people who lack the necessary understanding. In this study, a special model based on Facebook Prophet is created to alleviate the limitations of the ARIMA model in terms of Bitcoin trading [7].

Prophet's hyperparameters don't need much tweaking because their main goal is to find correlations in commercial time series. Outliers and trend shifts based on new goods and market events are often manageable by the Prophet. Prophet resists missing data and changes in trend. In opposition to Auto-Arima, Prophet shows a seasonal pattern that is more accurate, despite the fact the absolute values are considerably off from the real information. Prophet is exceptional in that it doesn't necessitate any prior expertise or expertise in forecasting time series data due to its capacity to instinctively recognize the fundamental seasonal patterns within the data and give a set of "simple to understand" criteria.

Prophet was created to handle planned holidays, missing data, and significant outliers [8]. It may therefore be utilized by non-statisticians to produce conclusions that are generally on par with, if not superior to, those produced by experts. The most efficient method for learning from sequential data is probably LSTM-based neural network, of which time series are a special example. Using 4857377 the digital currency Bitcoin data source.

We undertake a fair study on the Auto-Regressive Integrated Moving Average (ARIMA), Long-Short-Term Memory (LSTM), FB-Prophet, XGBoost, and two hybrid formulations, LSTM-GRU and LSTM-1D_CNN.

3. MATERIAL AND METHODS

3.1. DATASET

The major focus of this study is the time-series forecast of BTC prices using different machine learning and deep learning models. An accumulation of data values for a sequence of time points is referred to as a time series. We collected the "Bitcoin Historical Data" from Kaggle competitions. Here, historical market data for a few exchanges that accept Bitcoin is made available at 1-minute intervals. The OHLC (Open, High, Low, Close) minute-by-minute updates, volume in BTC and the designated currency, and adjusted Bitcoin price, covers the period from January 2012 to September 2020. The Open and Close columns display the daily opening and closing prices. Utilizing historical Bitcoin data from 2012 to 2020, we contrasted the models.

3.2. METHODOLOGY

We used the Bitcoin data to apply current machine learning and front-line deep learning models. We used well-known models including the ARIMA, LSTM, Prophet, XGBoost, LSTM-GRU and LSTM-1D CNN model. Because the data were spatial-temporal, models that utilized LSTM and GRU approaches produced superior results. We provide a framework for better analysis, as shown in Fig. 1 The following phases are used to explain the framework in detail.

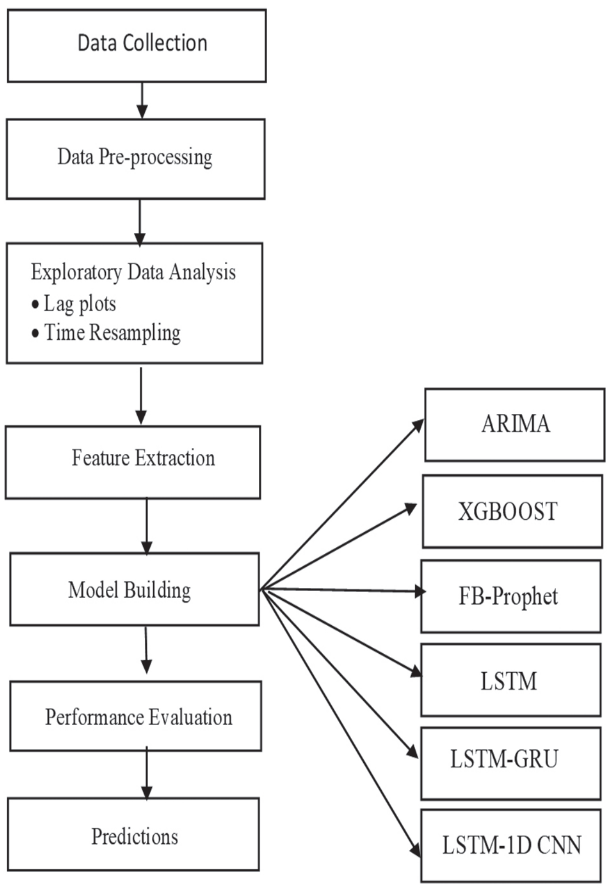

Fig. 1. Proposed Methodology

3.2.1. HANDLING MISSING VALUES

Missing data occurs when no information is provided for a couple of elements. In the actual world, inadequate data is a big problem. NA (Not Available) values may also be referred to as missing values in pandas. Several datasets in data frames sometimes contain missing values, either as a result of the data never being collected or as a result of the data being available but not being recorded.

Table 1. Attribute names with missing values

Attribute Name	Total Missing Values	Missing %
Timestamp	0	0.00000
Open	1243608	25.60246
High	1243608	25.60246
Low	1243608	25.60246
Close	1243608	25.60246
Volume_(BTC)	1243608	25.60246
Volume_(BTC)	1243608	25.60246
Weighted Price	1243608	25.60246

To overcome this situation, we use Fill values forward shortly called 'ffill' or 'pad', and Fill values backward similarly called 'bfill' or 'backfill', and the linear interpolations method was used as three imputation strategies to replace the missing data. The observed value is used to replace NaNs in the 'ffill' technique. The next observed value is used to replace NaNs in the 'bfill' procedure. These two techniques allow for the filling of a sizable amount of the missing information. The linear interpolation technique is used to fill in the missing data. It is a restoration strategy that, on the premise that there is an immediate relationship between the data points, substitutes a missing data point with non-missing values from nearby data points [9]. No null values are discovered when we use these three methods on our dataset. Table 1 describes different attribute names and their missing values.

3.2.2. Exploratory Data Analysis

a. Visualizing the weighted price

Visualizing time-series data may reveal a lot when dealing with it. To assist in highlighting certain observations or specific occurrences in the time series, markers can be added to the plot. Since the cryptocurrency market is open around the clock and cryptocurrencies are prone to wild price fluctuations, traders are always seeking methods to take advantage of openings [10]. Visualizing the weighted price enables traders to decide when to purchase or sell. Fig. 2 shows the weighted price vs. years plotting.

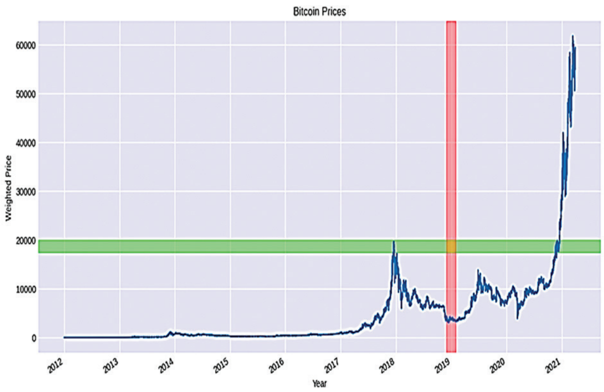

Fig. 2. Weighted price vs. Years

b. Visualizing using Lag Plots

To establish whether the outcomes in a collection of data or a series of values are random or not, lag graphs are utilized. If the information being analyzed is random, the lag plot fails to show any discernible patterns.

Is there going to be more unwanted mail because of user risky behavior (e.g. questionable registrations around the World Wide Web) [10]?

- Is there going to be more unwanted mail because users leave their addresses around on the Internet [11]?
- Does it hold that even a careful e-mail user is obligated to eventually start receiving unwanted mail?

If the facts are not randomly distributed, the lag plot will clearly show a pattern. The type of pattern can assist the user in identifying the irregular structure in the data. Outliers can also be found using lag graphs. The minutely, hour, making it, and day lag graphs all exhibit a positive correlation, as can be shown [11, 12]. We discovered no relationship for month lag plots. We replicated all of our information to the everyday level maximum to preserve autocorrelation. Fig. 3 shows the lag plots with different time intervals.

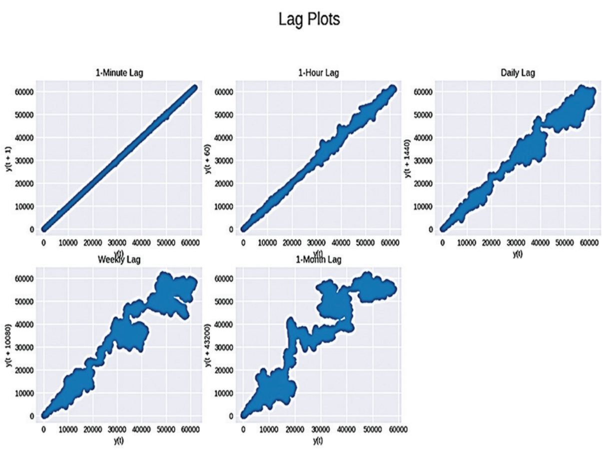

Fig. 3. Lag plots

c. Time Series Decomposition

Trend, seasonal, and other components can be separated into a time series. The series may be created by multiplying or adding the starting point, trend, periodic index, and residual [13]. Then, using statistical analyses such as the KPSS and enhanced Dickey-Fuller tests, we verified stationarity. After decomposing the time data, we discover no indication of seasonality. Because the variance, covariance, and mean are not constant, the series is also non-stationary. Fig. 4 shows the time series decompositions plotting.

Fig. 4. Time series Decompositions

d. Auto Correlation Function (ACF)

Autocorrelation is the relationship between a given time series and an extended version of itself. The time series' relationship with itself is the initial lag of the automatic correlation function (ACF), which results in a correlation of 1.

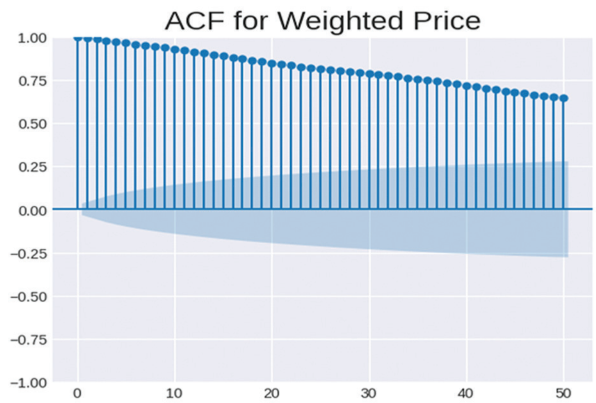

Fig. 5. Auto Correlation Function (ACF) for Weighted Price

A computed number called Auto Correlation Function (ACF) is used to show how closely a value in a time series resembles a prior value. Python's Statsmoldels package greatly simplifies the process of computing autocorrelation [14, 15]. Fig. 5 shows the Auto Correlation Function (ACF) for Weighted Price.

e. Partial Auto Correlation Function (PACF)

In time series analysis, the partly autocorrelation function (PACF) regresses the time series' components at all shorter lags to determine the complete relationship between a stationary time series and its individual independently delayed values. Unlike the autocorrelation function, it ignores additional delays. The partial autocorrelation at lag k is the synchronization between X_t_t and X_(t-k) that is ignored by delays 1 through k1. By setting the method option to "ols" (regression of time series dependent on it and on constant lags), we will use the plot_pacf function of the statsmodels. graphics.tsaplots package [16]. Fig. 6 shows the Partial Auto Correlation Function (PACF).

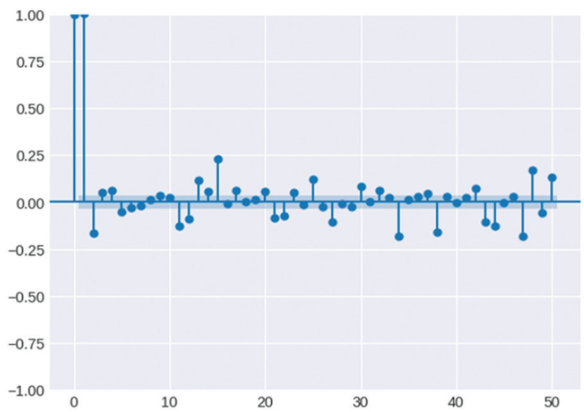

Fig. 6. Partial Auto Correlation Function (PACF)

3.2.3. Feature Extraction

Time series data could be noisy as a result of the substantial market movements. It might be difficult to spot a pattern or trend in the data. There is a tonne of noise when evaluating everyday data. Our data's noise level has significantly decreased, and the upward trend is now more obvious than the real data. There is an excessive amount of noise in the daily figures that we are looking at. We use a rolling mean to average this out by one week, which is handy. The rolling mean, sometimes called the moving average, is a data processing method that helps remove noise from data by averaging [17]. Simple division and agglomeration of the data into windows based on factors like mean, median, count, etc. The trend is now more obvious than the actual data since we use a rolling mean for 3, 7, and 30 days.

4. MODEL BUILDING

4.1. AUTO-REGRESSIVE INTEGRATED MOVING AVERAGE (ARIMA) MODEL

To better comprehend the data set or forecast future changes, time series data are employed in a statistical analysis model called ARIMA. ARIMA stands for Auto-Regressive Integrating Moving Average. It's a collection of models that may be applied to time series data to depict a range of frequent temporal patterns [18, 19]. This acronym is descriptive and encapsulates the key model characteristics. The model forecasts upcoming securities or changes in financial markets by focusing on discrepancies between values compared to actual values across the series.

4.2. FACEBOOK PROPHET

Facebook's Prophet is a freely available approach for creating time-series models that include both classic ideas and cutting-edge advances. It succeeds at simulating time series with various seasonality and does not share the drawbacks of other algorithms. It depends on an element of the model in which seasonality (weekly, yearly, plus vacation) and non-linear patterns are fitted together. Even in the context of incomplete data, Prophet can identify major outliers and variations in the trend. Furthermore, it precisely calculates the mixed data without requiring manual labor. Prophet was created to benefit from Facebook business predictions. Examples include trends with non-linear growth curves, significant outliers, trend changes, and daily, monthly, and yearly historical observations [20, 21].

4.3. XGBOOST

A universal gradient boosting library called XGBoost designed to be rapid and scalable for training machine learning models. XGBoost, which stands for Extreme Gradient Boosting, is a popular and powerful machine learning algorithm used for both regression and classification tasks. It belongs to the ensemble learning family of algorithms and is known for its exceptional predictive performance. The ensemble learning approach is used to aggregate results from a variety of weak models to get a more accurate forecast. The "ds" column among them is used to store date-time series. The values for the relevant time series in the data frame are kept in the "y" column, which is the other column. As a result, the outline works fairly well with seasonal time series and offers a few options for handling the dataset's seasonality.

To employ libraries like xgboost in Python, which provide user-friendly APIs for building and training XGBoost models. This library offers extensive documentation and various tuning options to help and get the most out of the algorithm.

4.4. LONG SHORT-TERM MEMORY (LSTM)

Long Short-Term Memory (LSTM) were introduced by Sepp Hochreiter and Jurgen Schmidhuber in 1997 to address the vanishing gradient problem that can occur when training traditional Recurrent Neural networks (RNNs). LSTMs are designed to capture long-range dependencies in sequential data by maintaining a memory cell that can store information over long sequences. They achieve this by using a more complex structure than simple RNNs, which allows them to learn and remember information over longer time intervals [22]. RNNs come in a variety of types, with LSTM networks being the most popular. The input and forget gates modulate the inner contents of the memory cell. Since both segue are closed, the information in the memory cell will not change from one time step to the next. This allows groups of information to flow across many time steps. As a result, the LSTM model can correctly overcome the vanishing gradient issue that most RNN models have [23, 24].

4.5. LSTM-GRU MODEL

We originally used a composite LSTM-GRU model. LSTM is used to overcome the problem of vanishing gradients in backpropagation. The three gates that make up an LSTM are the input gate (IG), forget gate (fg), and output gate (og). Gates are used in memory to store information. Information is kept there in analog format. Sigmoid function ranges between 0 and 1 are multiplied element-by-element on these gates. If the gate's value is zero, the data is ignored or deleted.

Tan(h) is a popular non-linear activation function that has a range of 1 to +1. The information does not fade when a second derivative is used. A sigmoid function has values between 0 and 1. In simple terms, the method is used to suggest to memory components called gates whether data should be destroyed or maintained. Equations discuss input gate, forget gate, and output gate (og) mathematical equations that were taken from the literature and changed. The update gate (ug) and reset gate (rg) are GRU's two gates, in contrast. This approach required feeding the LSTM the output of the GRU. At a specific time and location, the input set of characteristics exit contains hybrid feature space (start-node, end-node, way-id, week, hour, agg-minutes, quarter, holiday, peak hour, etc).

4.6. LSTM-1D_CNN MODEL

Nine layers make up the suggested algorithm for predicting bitcoin prices. In Fig. 7, the model architecture is shown.

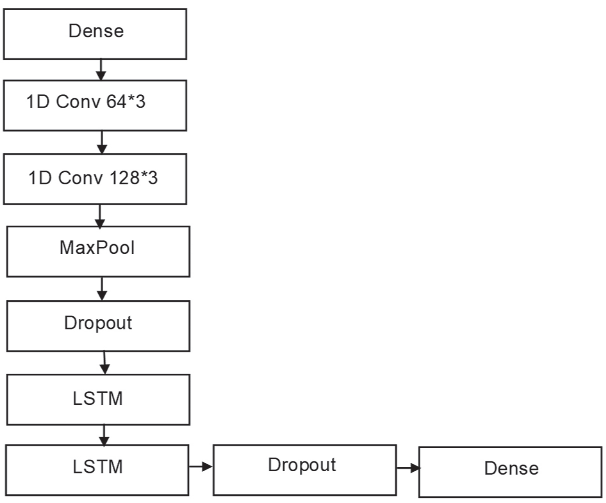

Fig. 7. Architecture of LSTM-1D_CNN

There are 2 LSTM layers and 2 1D CNN layers. Relu serves as the activation function for each convolutional layer. To prevent overfitting, two dropout layers have also been included. Both 0.5 and 0.4 are dropout rates. The output unit has the number 5. There are 200 in the batch.

The most dependable performance is determined by examining a range of input circumstances. Two hybrid deep learning models work incredibly well for forecasting time series. It has two key benefits, namely: (1) In order to make accurate predictions, the model does away with the time series decomposition procedure and (2) strengthens its high-level temporal representation.

5. PERFORMANCE MEASURE OF HYBRID DEEP LEARNING MODELS

The approach's performance and pattern are evaluated using well-known and current performance evaluation measures. Additionally, it suggests the ideal model for achieving output label efficiency. We have utilized established performance assessment measures like RMSE (Root Mean Square Error) and MAE (Mean Absolute Error) to assess the solution.

The information and formulae for the current performance evaluation measures are obtained. The abbreviation RMSE stands for root-mean-square deviation. The mean squared variance between intended output and projected output is what this term represents.

MSE = $\sqrt{(\sum_i(Y_{dsi} - Y_{psi})^2 / n}$ (number of observations)

Where,

Y_{dsi} = desired, Y_{psi} = predicted speed, n= number of observations.

The MAE (Mean-Absolute-Error) in our example measures how far away from the required output speed our anticipated output speed is. We can get a mathematical explanation for this.

$MAE = 1/n \sum_i |(Y_{dsi} - Y_{psi})|$

6. RESULTS AND ANALYSIS

The graphical depiction of the results and analysis on the test dataset is demonstrated in this part. Fig. 8 displays the ARIMA projected Bitcoin price, where the x and y axes represent the months and the sample price rate. Fig. 9 and Fig. 10 illustrate the XG Boost and FB-Prophet predicted Bitcoin price, similarly, Fig. 11, Fig. 12 and Fig.13 illustrate the LSTM, hybrid LSTM-1D_CNN and LSTM-GRU predicted Bitcoin price respectively. In this collection, the test data is taken from January 2020 through September 2020.

Table 2, shows a comparison analysis of proposed method and previous method in Bitcoin price prediction. we quickly contrast the outcomes of earlier research in the field with our newly proposed method. Table 3 shows that the RMSE value for the ARIMA is 18100.14, whereas the values for the FB-Prophet, XG Boost, and LSTM are 1309.49, 13571.91, and 421.29 respectively but when we hybridize the combinations of LSTM-GRU and LSTM-1D_CNN the hybrid model performs better than others. The RMSE value of LSTM-1D_CNN is 83.408 and LSTM-GRU is 0.323.

Overall, the suggested hybrid deep learning based prediction model LSTM-GRU result is significant as compared to others.

Both individual investors and asset managers must be able to predict the price of bitcoin. Despite the fact that Bitcoin is a currency, it cannot be analysed similarly to other traditional currencies where economic theories about future cash flow models, purchasing power parity, and uncovered interest rate parity are relevant. This is due to the fact that the market for digital currencies like Bitcoin does not allow for the application of certain traditional criteria relating to the connection between supply and demand. Contrarily, Bitcoin has a number of characteristics that make it useful for investors, including transaction speed, dispersion, decentralisation, and the sizable online community of people interested in talking about and sharing important information concerning digital currencies, especially Bitcoin.

Table 2. Comparison of proposed method and previous method

Reference	Dataset	Method	Results
Yan Li et al. [18]	Bitcoin	BP, CNN, LSTM, CNN-LSTM	RMSE of BP = 515.35 RMSE of CNN = 261.90 RMSE of LSTM = 297.97 RMSE of CNN-LSTM= 258.31
Kervanci et al. [19]	Bitcoin	4-layer LSTM, LSTM and BO, 4-layer GRU	RMSE of 4-LSTM = 32.98 RMSE of LSTM & BO= 4260 RMSE of 4-GRU= 251.60
Hashish et al. [20]	Bitcoin	ARIMA, LSTM, HMM-LSTM	RMSE of ARIMA = 141.96 RMSE of LSTM = 7.006 RMSE of HMM-LSTM= 5.82
Proposed work (ensemble deep model)	Bitcoin	ARIMA, LSTM, FB Prophet, XGBoost, LSTM-1D_CNN LSTM-GRU	RMSE of ARIMA = 18100 RMSE of LSTM = 421 RMSE of Prophet= 1309 RMSE of XGBoost= 13571 RMSE of LSTM-1D_CNN = 83.408 RMSE of LSTM-GRU= 0.323

Fig. 8. ARIMA Predicted BTC Price

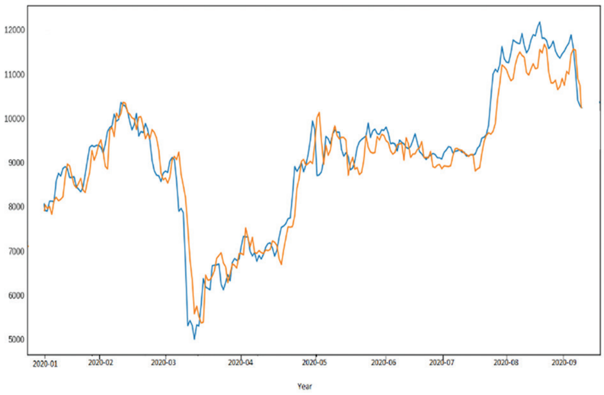

Fig. 9. XGBoost Predicted BTC Price

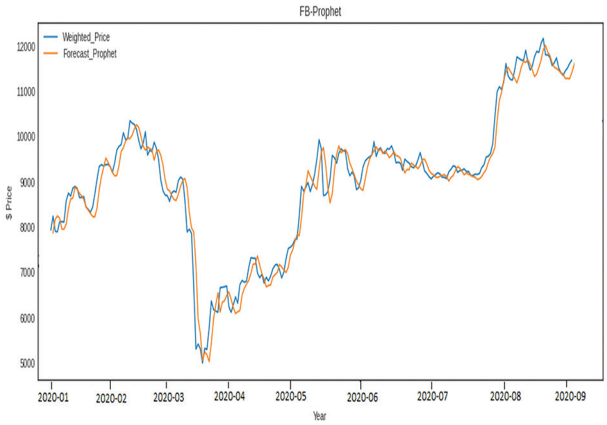

Fig. 10. FB Prophet Predicted BTC Price

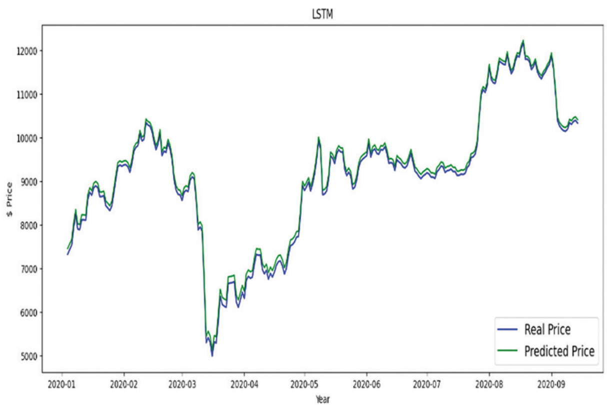

Fig. 11. LSTM Predicted BTC Price

Fig. 12. LSTM-1D_CNN Predicted BTC Price

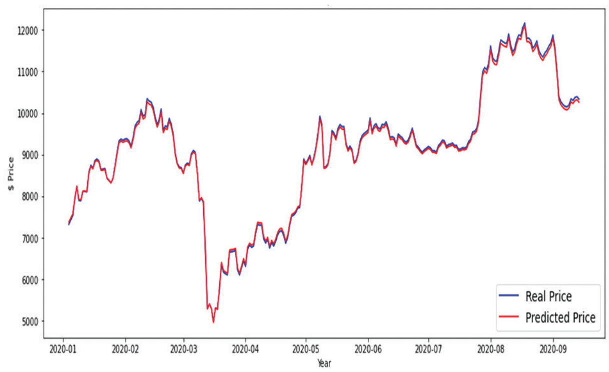

Fig. 13. LSTM-GRU Predicted BTC Price

Table 3. RMSE, MAE values of models

Model Name	RMSE Value	MAE Value
ARIMA	18100.14	10733.36
XG_Boost	13571.91	6500.36
FB-Prophet	1309.49	690.08
LSTM	421.29	20.518
LSTM-1D_CNN	83.408	9.140
LSTM-GRU	0.323	0.464

7. CONCLUSION

With the swings of the financial trading market and the precision of forecasting models, the market for Bitcoin mechanics alters dynamically. A comparison analysis must be done to properly comprehend the parallels and contrasts between various financial enterprises using cryptocurrency. Several deep-learning prediction methods are introduced in this work for forecasting Bitcoin's market behaviour. we use two ensemble models, both of which give an eye-catching result. The hybrid LSTM-1D_CNN gives an RMSE value of 83.408 and an MAE value of 9.140, while the LSTM-GRU gives an RMSE score of 0.323 and an MAE score of 0.464. The hybrid LSTM-GRU model performed well as compared to others.

We employed time series forecasting since output in the past was unpredictable. Deep learning techniques can recognize and benefit from the links and patterns contained in a data set through a self-learning process. Contrary to conventional approaches, algorithms based on deep learning can analyse the linkages and undetected trends within the data to effectively represent this type of data and offer a reliable forecast. Multivariate time series research employs several deep learning methods. The models might potentially be trained using small datasets and short-term predictions in the future. This study might be enhanced by using other hybrid deep learning techniques to forecast Bitcoin performance.

8. REFERENCE

[1] R. Schmidt, M. Möhring, D. Glück, R. Haerting, B. Keller, C. Reichstein, "Benefits from Using Bitcoin: Empirical Evidence from a European Country", International Journal of Service Science, Management, Engineering, and Technology, Vol. 7, 2016, pp. 48-62.

[2] Z. Shahbazi, Y. C. Byun, "Knowledge Discovery on Cryptocurrency Exchange Rate Prediction Using Machine Learning Pipelines", Sensors, Vol. 22, 2022, p. 1740.

[3] G. Prashanth, G. V. S. S. K. R. Naganjaneyulu, M. Revanth, A. V. Narasimhadhan, "Multi Indicator based Hierarchical Strategies for Technical Analysis of Crypto market Paradigm", International Journal of Electrical and Computer Engineering Systems, Vol. 14, No. 7, 2023, pp. 765-780.

[4] W. Chen, Z. Zheng, M. Ma, J. Wu, Y. Zhou, J. Yao, "Dependence structure between Bitcoin price and its influence factors", International Journal of Computational Science and Engineering, Vol. 21, 2020, pp. 334-345.

[5] S. F. Wamba, J. R. K. Kamdjoug, R. E. Bawack, J. G. Keogh, "Bitcoin Blockchain and Fintech: A systematic review and case studies in the supply chain", Production Planning & Control, Vol. 31, 2019, pp. 115-142.

[6] B. Manujakshi, M. G. Kabadi, N. Naik, "A Hybrid Stock Price Prediction Model Based on PRE and Deep Neural Network", Data, Vol. 7, 2022, p. 51.

[7] N. Tripathy, S. Hota, D. Mishra, "Performance analysis of bitcoin forecasting using deep learning techniques", Indonesian Journal of Electrical Engineering and Computer Science, Vol. 31, No. 3, 2023, pp. 1515-1522.

[8] E. Loh, S. Ismail, "Emerging Trend of Transaction and Investment: Bitcoin Price Prediction using Machine Learning", International Journal of Advanced Trends in Computer Science and Engineering, Vol. 9, 2020, pp. 100-104.

[9] M. J. Hamayel, A. Y. Owda, "A Novel Cryptocurrency Price Prediction Model Using GRULSTM and bi-LSTM Machine Learning Algorithms", AI, Vol. 2, No. 4, 2021, pp. 477-496.

[10] T. Shintate, L. Pichl, "Trend Prediction Classification for High Frequency Bitcoin Time Series with Deep Learning", Journal of Risk and Financial Management, Vol. 12, No. 1, 2019, p. 17.

[11] V. Derbentsev, V. Babenko, K. Khrustalev, H. Obruch, S. Khrustalova, "Comparative Performance of Machine Learning Ensemble Algorithms for Forecasting Cryptocurrency Prices", International Journal of Engineering, Transactions A: Basics, Vol. 34, 2021, pp. 140-148.

[12] S. Yogeshwaran, M. J. Kaur, P. Maheshwari, "Project Based Learning: Predicting Bitcoin prices using deep learning", Proceedings of the IEEE Global Engineering Education Conference, Dubai, United Arab Emirates, 8-11 April 2019, pp. 1449-1454.

[13] N. Tripathy, S. Hota, S. Prusty, S. K. Nayak, "Performance Analysis of Deep Learning Techniques for Time Series Forecasting", Proceedings of the International Conference in Advances in Power, Signal, and Information Technology, Bhubaneswar, India, 9-11 June 2023, pp. 639-644.

[14] L. G. N. De Leon, R. C. Gomez, M. L. G. Tacal, J. V. Taylar, V. V. Nojor, A. R. Villanueva, "Bitcoin Price Forecasting using Time-series Architectures," Proceedings of the International Conference on ICT for Smart Society, Bandung, Indonesia, 10-11 August2022, pp. 1-6.

[15] M. A. Fauzi, N. Paiman, Z. Othman, "Bitcoin and cryptocurrency: Challenges opportunities and future works", Journal of Asian Finance Economics and Business, Vol. 7, No. 8, 2020, pp. 695-704.

[16] S. Lahiri, S. Bekiros, "Deep Learning Forecasting in Cryptocurrency High-Frequency Trading", Cognitive Computation, Vol. 13, No. 2, 2021, pp. 485-487.

[17] P. D. DeVries, "An analysis of cryptocurrency Bitcoin and the future", International Journal of Business Management and Commerce, Vol. 1, No. 2, 2016, pp. 1-9.

[18] Y. Li, W. Dai, "Bitcoin price forecasting method based on CNN-LSTM hybrid neural network model", The Journal of Engineering, Vol. 2020, No. 13, 2020, pp. 344-347.

[19] I. S. Kervanci, M. F. Akay, E. Özceylan, "Bitcoin price prediction using LSTM, GRU and hybrid LSTM-GRU with bayesian optimization, random search, and grid search for the next days", Journal of Industrial and Management Optimization, 2023.

[20] I. A. Hashish, F. Forni, G. Andreotti, T. Facchinetti, S. Darjani, "A hybrid model for bitcoin prices prediction using hidden Markov models and optimized LSTM networks", IEEE International Conference on Emerging Technologies and Factory Automation, Zaragoza, Spain, 10-13 September 2019, pp. 721-728.

[21] N. Tripathy, S. K. Nayak, J. F. Godslove, I. K. Friday, S. S. Dalai, "Credit Card Fraud Detection Using Logistic Regression and Synthetic Minority Oversampling Technique (SMOTE) Approach", International Journal of Computer and Communication Technology, 2022, 8(4), 4.

[22] J. Liu, A. Serletis, (2019), "Volatility in the cryptocurrency market", Open Economies Review, 2019, 30, 779-811.

[23] Z. Ftiti, W. Louhichi, H. Ben Ameur, "Cryptocurrency volatility forecasting: What can we learn from the first wave of the COVID-19 outbreak?", Annals of Operations Research, 2021, 1-26.

[24] N. Tripathy, I. K. Friday, D. Rath, D. S. K. Nayak, S. Nayak, "Deep Learning-based Gated Recurrent Unit Approach to Stock Market Forecasting: An Analysis of Intel's Stock Data", International Journal of Smart Sensor and Adhoc Network, 2023, 4 (1).