Frontiers in Health Informatics
ISSN-Online: 2676-7104
www.healthinformaticsjournal.com
2024; Vol 13: Issue 7
Open Access

Generative Pre-Trained Transformer Deep Learning And Sentimentanalysis-Based Bitcoin Cryptocurrency Price Prediction

Jay Krishna Joshi[1], Dr. R.P.Sharma[2], Dr.Aboo Bakar Khan[3]
[1]Department of Data Science, 'Mukesh Patel School of Technology Management & Engineering, SVKM's NMIMS, Mumbai- 400056, India', and Research Scholar, Chhatrapati Shivaji Maharaj University, Panvel410206, India
[2]Registrar, Chhatrapati Shivaji Maharaj University, Panvel410206, India
[3]Head of Electrical and Electronic Engineering Department, Chhatrapati ShivajiMaharaj University, Panvel 410206, India)

Cite this paper as: Jay Krishna Joshi, Dr. R.P.Sharma , Dr.Aboo Bakar Khan (2024). Generative Pre-Trained Transformer Deep Learning And Sentimentanalysis-Based Bitcoin Cryptocurrency Price Prediction. *Frontiers in Health Informatics*, *13 (7) 1027-1039*

Abstract
Predictability in the Bitcoin (BTC) prices can allay the fears of small investors. Existing techniques are yet to give importance to large language models and transformer architectures in developing the price prediction models. To fulfill the research gap, the current research paper has developed a model using Generative Pre-trained Transformer (GPT)architecture integrated with a linear artificial neural network (ANN) for making the finalprice prediction of BTCs in USD. The model is trained on BTC news articles dated June1, 2023, to June 1, 2024 from coindesk website collected through web scraping. Thenthe model was trained and after hyper-parameter tuning the results on the out-of-sample dataset were 5% mean absolute percentage error. Compared to state of the art models the performance of the proposed model on the other metrics like residual plots, meansquared error and prediction accuracy were higher by 2-5 %.

Keywords: Transformers, Deep learning, Bitcoin, Price predictions.

1. Introduction

BTC by Satoshi Nakamoto introduced use of blockchain and cryptocurrencies to the world in 2009. The uniqueness of bitcoin was that there was no central regulating entity to monitor its demand and supply. Initial adopters were tech-enthusiasts, researchers and privacy advocates. However, within a few years serious players such as industries, investors, capitalists (and even cyber-criminals) too started accepting the technology. Ever since BTC was introduced till today several new cryptocurrencies have emerged and yet even today BTC continues to lead the pack (Dash, Ethereum, Litecoin, Ripple and Monero) both in terms of price per unit and market capitalization. Hence, understanding the unpredictable nature of BTC cryptocurrency has become a subject of extensive research ever since the cryptocurrency became public (3).

Compared with the regular listed shares in stock markets the price of BTCs has seen high fluctuations as it rose from $1 in 2009 to $20,000 in 2017. A $1 dollar investment in BTC in Nov 2015 (1 BTC = USD 315) would fetch 200 times returns on investment (RoI) in nine years as of June 2024 (1 BTC = USD 70000). In spite of the high RoI retail investors especially small investors are jittery of investing in BTCs as compared to regular listed shares as BTCs lack the transparency, tangibility, entity of ownership and regulation mechanism. In such a scenario, if an element of predictability is introduced in the form of BTC price prediction it will allay the doubts of investors.

Major works in the literature have focused on time series analysis, machine learning (ML), deep learning (DL) approaches and even sentiment analysis for BTC risk management through price prediction (15). BTC price prediction is a multifaceted problem benefiting from various analytical methodologies. Time series analysis employs historical price data to identify patterns and trends, offering insights into future movements. Machine learning (ML) algorithms enhance predictive accuracy by uncovering complex, non-linear relationships within vast datasets. Deep learning (DL) further refines these predictions by leveraging neural networks to model intricate dependencies. Economic models integrate macroeconomic variables, leveraging theoretical frameworks to interpret market dynamics influencing BTC. Each approach contributes uniquely, ensuring robust, comprehensive predictions, thus enabling investors and stakeholders to make well-informed decisions in the highly volatile cryptocurrency market. Within DL, long short term memory networks (LSTM) and other variants of recurrent neural networks (RNN) like gated recurrent unit (GRU) or 1D convolutional neural networks (1D-CNN) are known for their efficacy in the literature for time series prediction. In general, ML has been accepted as the first line of techniques for BTC price prediction (9). It is not uncommon to find research on the utility of classical time series models for price predicting. The difference between the ML/DL approach and the time series approach is the presence of additional features viz., news articles, social media data. These additional features are considered important in BTC price prediction by the ML/DL research whereas the time series research uses only the historical BTC price for predicting the future price. Although ML/DL or economic models are well established and empirically tested, both streams have their pros and cons. And the current paper prefers to use the ML/DL approach over classical economic modeling as it offers flexibility and data driven analysis.

Crypto-currencies like other financial markets are significantly impacted from current affairs like wars, elections, etc. Such factors contribute to the volatility and non linearity. It is challenging to capture the inherent uncertainties connected with cryptocurrency using the currently used conventional methodologies. Our approach is to learn a pattern between news articles and crypto prices through ML. The main motivation for such an approach is due to known events in the past where this trend was first noticed. Crypto-enthusiasts would recall the events of May 2021, when the BTC market price saw a drop of nearly 20%. Analysts associated the whole catastrophe with Chinas announcement of a cryptocurrency crackdown and Tesla CEO Musks announcement a week earlier saying that the company would be refusing payments in BTCs for its vehicles (2). This premise clearly tells us that news articles would have significant impact on the price of BTCs.

The primary contributions of the current paper are as follows:

1. Data collection of news articles and BTC prices for creating a dataset. Further, the data will be preprocessed, cleaned, tokenized and vectorized before analysis. Nonlinear data visualization and clustering techniques will be used to understand the different groups.
2. Develop a DL methodology for learn the hidden patterns in the word embeddings and the BTC price. Apply data reduction techniques using feature selection to retain important features only.
3. Compare and evaluate the ML methodology vis-a-vis state of the art nonlinear and linear ML (bagging, boosting, ensemble)/DL models (LSTM, GRU, RNN) and evaluate the predictive power of the ML methodology.

To the best of our knowledge based on the literature review of latest research, ours is the first work on several counts such as: use of LLM based word embeddings, news articles as features, and use of open source tools for development of the ML methodology and further open sourcing the entire codebase and data for research purposes. Even though there areresearch papers sharing some common points with our research especially related to the ML methodology adopted (7; 10) the field of BTC price prediction is dynamic and an time updated study shall always be beneficial for research.
The remaining sections of the paper are divided as follows: Section 2 describes thepreliminaries and review of literature, Section 3 explains the methodology and the datacollection procedure and finally the paper terminates with results and key learning in Section 4 followed by concluding remarks and future scope in Section 5.

2. Related works

In this section, we describe the major works in the field of research. Not only we find conventional ML techniques but also utility of reinforcement learning and meta-heuristic optimization.

2.1. Preliminaries

Linear models are fundamental techniques in statistical and machine learning domains. They establish a linear relationship between input variables and the output variable by fitting a linear equation to observed data. Common examples include linear regression (LR) for continuous outputs and logistic regression for binary outcomes. The simplicity of linear models allows for ease of implementation and interpretability. Key assumptions include linearity, independence, homoscedasticity, and normality of residuals. Although they are constrained by their linear hypothesis, linear models often serve as baselines and are particularly effective in scenarios where the underlying relationships between variables can be approximated by linear functions.

Non-linear machine learning models overcome the limitations of linear models by capturing complex patterns in data. Examples include decision trees, support vector machines with non-linear kernels, and neural networks. These models are capable of modeling intricate relationships between inputs and outputs, allowing for higher accuracy in prediction tasks. However, they often require more data and computational resources. The interpretability of non-linear models can be challenging, as they do not provide simple coefficients like linear models. Techniques such as feature importance, SHAP, and LIME are employed to gain insights into their decision-making processes.

RNN are specialized neural architectures designed for sequential data. Unlike feedforward networks, RNNs maintain hidden states that capture temporal dependencies. However, they suffer from vanishing and exploding gradient problems during training. LSTM networks address these issues with gating mechanisms input, output, and forget gatesthat regulate information flow, making LSTMs effective for long-range dependencies. GRU is a streamlined version of LSTM, combining the forget and input gates into a single update gate. GRUs maintain competitive performance with reduced computational complexity, making them efficient for sequence-based tasks.

Random Vector Functional Link Networks (RVFLN) enhance simple neural networks by incorporating randomization in the input weights, facilitating rapid training and effective performance on non-linear problems. Extreme Learning Machines (ELM) operate on a similar principle but focus on single-layer feedforward networks with hidden nodes having randomly assigned weights. Both RVFLN and ELM are known for their fast learning speeds and generalization capabilities. Variational Mode Decomposition (VMD) is a signal processing technique that decomposes a signal into its constituent modes, each representing an intrinsic mode function. VMD is utilized in applications requiring noise reduction and feature extraction, providing robust decomposition capabilities.

Economic models such as Generalized Autoregressive Conditional Heteroskedasticity (GARCH) and Exponential GARCH (EGARCH) are pivotal in the analysis of financial time series data. GARCH models address volatility clustering by specifying that current conditional variance depends on past squared returns and past variances, making it valuable for forecasting financial markets' volatility. EGARCH, an extension of GARCH, accounts for asymmetric responses of volatility to positive and negative shocks, a crucial feature for accurately capturing financial market behavior. These models are extensively utilized in risk management, option pricing, and determining Value at Risk (VaR), thereby providing robust quantitative tools for financial economists.

The AutoRegressive Integrated Moving Average (ARIMA) model is a widely utilized statistical technique in time series forecasting. ARIMA integrates three main components: autoregression (AR), differencing (I), and moving average (MA). The autoregressive part leverages dependencies in the dataset, the differencing aspect addresses non-stationarity, and the moving average component accounts for the error of the model as a linear combination of error terms. The model's parameters, denoted as ARIMA(p,d,q), correspond to the orders of these components. ARIMA is versatile and robust for short-term forecasting, making it an invaluable tool in various sectors, including finance, economics, and environmental science.

2.2. BTC pricing

BTC pricing is significantly influenced by the dynamics of supply and demand, trader sentiment, macroeconomic factors, and blockchain technology developments (14) (Figure 3).Cryptoexchanges, acting as intermediaries, offer platforms for trading BTC against fiat currencies or other cryptocurrencies. Initially, BTC pricing was minuscule, reflecting limited acceptance and liquidity. Over the years, increased mainstream adoption, institutional interest,and regulatory frameworks have contributed to price volatility and upward trends. Developments such as halving events and technological advancements (e.g., Lightning Network) also impact pricing. Analysis of historical data reveals a pattern of rapid valuation shifts, underscoring the complex interplay of intrinsic and extrinsic factors shaping BTCs market price.

2.3. State of the art: Literature survey

Generally DL based RNN approaches have been popular and one such work in the recent past that uses LSTM and feature extraction from BTC blockchain is by G Kim et al. (9). The authors argue that a self-attention-based multiple long short-term memory (SAM-LSTM) approach has found the highest mean absolute error (MAE), root mean square error (RMSE), mean square error (MSE), and mean absolute percentage error (MAPE) values of 0.3462, 0.5035, 0.2536, and 1.3251, respectively. For training the model, the authors have performed extensive feature extraction of 254 variables related to BTC price, market, distribution and transaction data. The authors have accepted that the key drawback of their work is the lack of comparison will state of the art techniques. However, the authors agreed that social media data can be used as a source of information related to price prediction but this was left by the authors as a task for others interested in carrying forward their research.

G Hongze et al. too have recognized that no single technique has accurately identified the price of BTCs and advocated the use of ARIMA for price prediction (5). The authors have not used any external features other than the price of the BTC itself for prediction. Thus, this line of research is different from the classical ML or DL research that have used features such as on-chain or off-chain for price prediction. The main criticism of economic models like vector autoregressive models (VAR), or ARIMA or error correcting models (ECM) is that they assume that the trend shall be stable and repeatable. However, BTC pricing is volatile, non-linear and non-stationary and hence models will have limited predictive power.

Ensemble modeling is also a popular technique used in ML/DL to improve prediction power by using multiple models and averaging or voting their results to get a final prediction. Here, the work of X Du et al. (4) is a good example of research where a hybrid ensemble model is built on top of a pipeline of multiple models like ELM and VMD. The authors optimized the parameters of their hybrid model using an optimization strategy. Their model could achieve a R^2 of around 0.92 for short term forecasting. In their concluding remarks, the authors noted that there were complex influencing factors which could have been included in the price prediction analysis.

Complex neural networks like RVFLN trained using an elitist artificial electric field algorithm (eAEFA) has been empirically evaluated against popular ML/DL techniques and found to be a useful financial forecasting tool for BTCs (7). We observe a common thread in the BTC price prediction literature - in feasibility of comparing against models proposed by other researchers due to differences in the datasets. Usually, authors have preferred to compare against standard ML/DL models or variants of their own models. A reason for this could be lack of transparency in data and code sharing observed by us in the research.

Previously discussed literature focused focused on current price prediction based on previous prices. There is research, that completely ignores past prices for predicting future prices. Like M Saad et al. (13) studied how on-chain features afftects the BTC price. The authors collected features like user activity on price, wallets and unique addresses, difficulty

and hash rate, cost per transaction, mining revenue, total transactions, unique addresses, fee, and price. Further the authors empirically tested and found the LR model trained on data from April 2016 to January 2018, and predicted results from January 2018 to May 2018 was the best. Another research performed on the same lines is by S Syed et al. (16). The authors aimed to identify the optimal model for predicting the price of BTC using Deephaven forData curation. Their study involves extracting data from BTCs by Deephaven and selecting the most correlating parameters through time lag adjustment. The authors used correlating cryptocurrency data to train models ANN, LSTM and GRU. For BTC, the results showed that the LSTM model outperform ANN and GRU models in both training and testing data with MAE, RMSE, and MAPE average values of 0.079, 1.16, and 0.0006, respectively.
Clearly, today's world relies on news and social media for information dissemination and BTC price prediction models are getting developed that rely on social media data for classification. American Institute for Economic Research shows that Bitcoin prices fluctuated substantially between 2016 and 2017 as a result of global news and emotions (11). However, the noise in the social media is an impediment to its direct use in the ML/DL pipeline. S Otabek et al. (11) used twitter data and sentiment analysis using the Valence Aware Dictionary and 362 Sentiment Reasoner (VADER) python library. Further the authors have used reinforcement learning (RL) strategies (such as Q-learning) for accurate BTC price prediction. Another line of recommendation has been action recommendation models that classify buying decisions as 'buy', 'wait' and 'sell' based on patterns learned from BTCs. The work of J Park et al. (12) sheds light on how current classification models can incorporate off-chain twitter data to improve their prediction accuracy. The authors performed extensive data cleaning on the kaggle dataset of twitter tweets collected for sentiment analysis. The sentiment analysis data was used for improving the prediction accuracy of the model.

According to the hypothesis (8) the price of an asset instantly acknowledges new information and reacts to it accordingly. Therefore, it is possible to observe how the market reacts to information and develop a pattern that exhibits the behavior of the market when new information is presented.

3. Materials and Methods

3.1. Transformer based DL architecture for BTC price prediction

Figure 2 illustrates the architecture transformer based DL architecture for BTC price prediction.

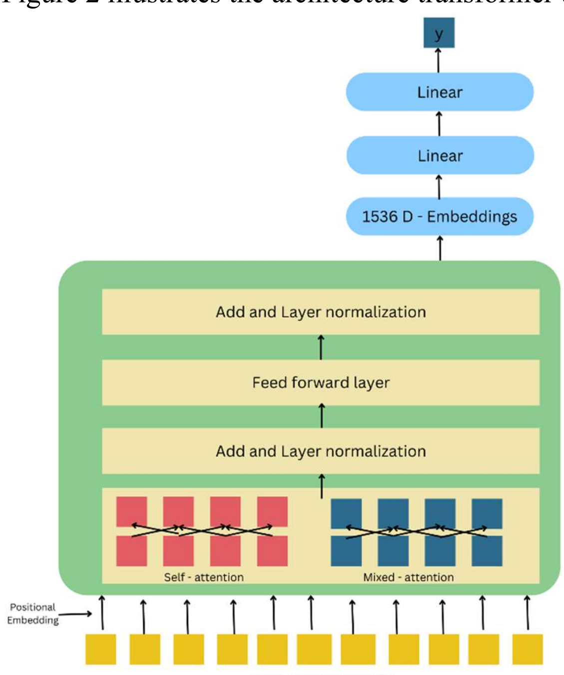


Figure 1. Proposed model: BTC pricing prediction using DL

Let D be a dataset containing news articles alongside historical Bitcoin prices. Each news article is processed using the model to produce a 1536-dimensional embedding vector e_t for the article published at time t. The first step in the model involves converting input text into a representation suitable for processing. For a given input sequence X = $\{x1, x2, \ldots, xn\}$, where x_i represents tokens in the sequence, the input is embedded into a continuous vector space using an embedding matrix W_e.

Given the token x_i, its corresponding embedding e_i can be expressed as:

$$\mathbf{e}_i = W_e \cdot \mathbf{v}_{x_i}$$

where v_{xi} is a one-hot vector representing the token x_i and $W_e \in R^{d \times V}$ is the embedding matrix with d being the dimensionality of the embeddings and V the size of the vocabulary.

Since the model is based on transformers, which are not inherently sequential, it utilizes positional encoding to retain the order of tokens in the sequence. The sinusoidal positional encodings pi are computed as follows:

$$\mathbf{p}_i = \begin{cases} \sin\left(\frac{i}{10000^{2j/d}}\right) & \text{if } j \text{ is even} \\ \cos\left(\frac{i}{10000^{2j/d}}\right) & \text{if } j \text{ is odd} \end{cases}$$

where j is the dimension index and i is the position of the token in the sequence. Positional encoding vectors are added to the embedding vectors to form the final input representation:

$h_i = e_i + p_i$

Each layer l of the model consists of a multi-head self-attention mechanism followed by a feed-forward neural network. The self-attention mechanism computes attention scores αij for token i focusing on token j:

The attention scores are calculated as follows:

$$\alpha_{ij} = \frac{\exp\left(\frac{\mathbf{h}_i \cdot \mathbf{h}_j^T}{\sqrt{d_k}}\right)}{{}_{k=1}^{n} \exp\left(\frac{\mathbf{h}_i \cdot \mathbf{h}_k^T}{\sqrt{d_k}}\right)}$$

where dk is the dimensionality of the keys.

The output of the self-attention sub-layer for a given token can then be calculated as:

$$\mathbf{z}_i = {}_{j=1}^{n} \alpha_{ij} \mathbf{h}_j$$

The output is then passed through a feed-forward network (FFN) which consists of two linear transformations with a ReLU activation in between:

$$\mathbf{y}_i = \text{FFN}(\mathbf{z}_i) = \text{ReLU}(W_1 \mathbf{z}_i + b_1) W_2 + b_2$$

Layer normalization is applied, and a residual connection is included after each sub-layer (self-attention and feed-forward) to facilitate the training of deep networks:

$$\mathbf{h}_i^{(l)} = \text{LayerNorm}\left(\mathbf{y}_i + \mathbf{h}_i^{(l-1)}\right)$$

The output representations from the last transformer layer are processed to generate predictions or embeddings for downstream tasks:

$$\mathbf{h}_i^{(L)} \in \mathbb{R}^d$$

where L is the number of layers in the architecture. The output representations are passed to a two layered ANN model that predicts the BTC price. The closing price of Bitcoin at time t is denoted as p . The set of features X for day t is constructed from the embedding of the previous day's news article e_{t-1} and the closing price of Bitcoin from the previous day p_{t-1}:

$$\mathbf{X}_t = [\mathbf{e}_{t-1}, p_{t-1}]$$

The target variable y_t is the closing price at time t:

$y_t = p_t$

The architecture is mathematically represented as follows:

$$z_1 = \sigma(W_1\mathbf{X} + b_1) \quad (1)$$

$$z_2 = \sigma(W_2 z_1 + b_2) \quad (2)$$

$$\hat{y} = W_3 z_2 + b_3 \quad (3)$$

Where: - $\mathbf{X}$ is the input vector containing news embeddings and previous price, - $W1$, $W2$, $W3$ are weight matrices for each layer, - $b1$, $b2$, $b3$ are the biases, - $z1$, $z2$ are the outputs from the hidden layers, - $\hat{y}$ is the predicted price, and - σ is Rectified Linear Unit (ReLU) activation function for hidden layers and linear for output layer.

The ANN is trained using the Adam optimizer with a mean squared error loss function. The dataset is split into training, validation, and test sets, ensuring a robust evaluation of the model's predictive capability.

The computational complexity of the proposed model primarily arises from the forward pass and the backpropagation processes. Let n be the number of input features (1537 in this case) and m be the number of neurons in the hidden layers. The time complexity of forward propagation in a single hidden layer is $O(nm)$ and can be generalized for k hidden layers as:

$$O\left(k \cdot (nm + m^2)\right)$$

Where m^2 accounts for the computations involving weight updates during backpropagation.

The overall time complexity for training the model over E epochs with a batch size of b can be expressed as:

$$O\left(E \cdot \frac{N}{b} \cdot \left(k \cdot (nm + m^2)\right)\right)$$

Where N is the total number of data points. Consequently, this model is scalable to larger datasets, contingent upon the dimensions of the dense layers and the embedding size.

3.2 Data Collection

The dataset encompasses a comprehensive collection of Bitcoin price data from June 1, 2023, to June 1, 2024, providing a detailed overview of the cryptocurrency's market behavior over this period (Figure 2). Key price metrics, including Open, High, Low, Close, and Adjusted Close values, are documented alongside trading Volume, allowing for an in-depth analysis of price fluctuations and trading activity. The inclusion of a 'Difference' column further aids in assessing daily volatility in the market. Complementing these quantitative measures is a qualitative component, represented by news articles sourced from the Coindesk website, which offers valuable contextual insights into the factors influencing Bitcoin price dynamics.

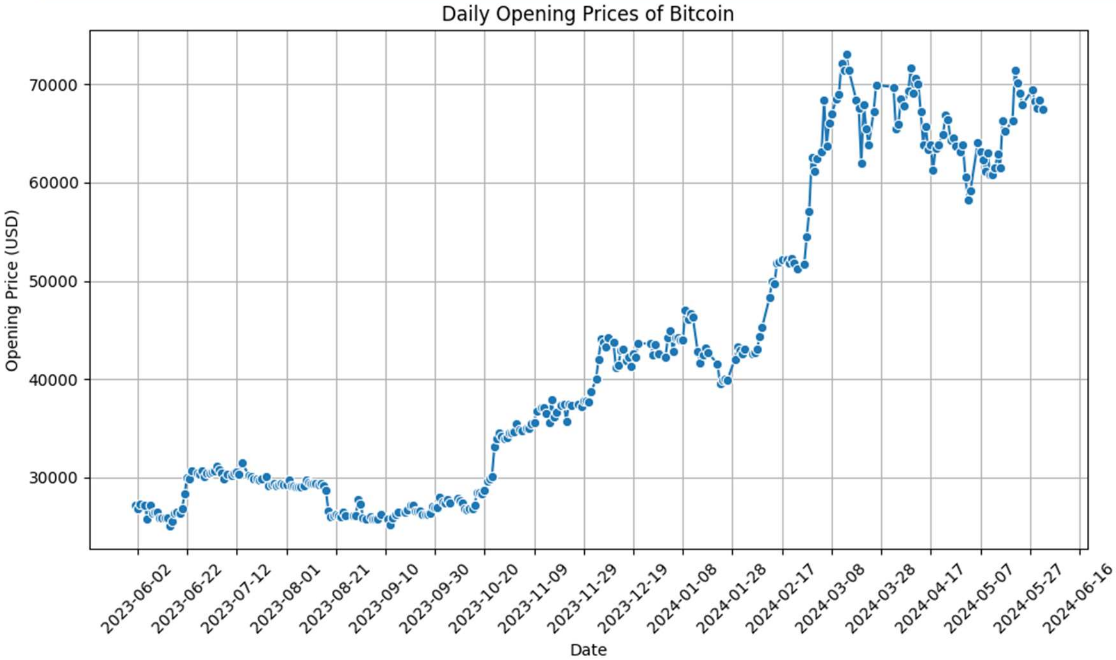


Figure 2. BTC pricing fluctuations in collected data

3.3. System implementation

In this study, we employed several key libraries to facilitate the conversion of news articles into semantic embeddings. First, we utilized Pandas, NumPy and we also made use of tiktoken for efficient text encoding and tokenization, ensuring optimal performance during the embedding process. Data analysis and machine learning/deep learning modeling were conducted using Visual Studio Code on a laptop equipped with an Intel(R) Core(TM) i5-1035G1 CPU operating at 1.00GHz with a maximum frequency of 1.19 GHz, featuring a 64-bit operating system, a 64-bit processor, 8.00 GB of RAM, and running Windows 11 Home Single Language. The implementation can be cloned from the Github repo (https://github.com/JayKrishnaJoshi/Blockchain_news_analysis).

3. Results and Discussion

The performance of the proposed DL model is assessed and contrasted using three widelyrecognized metrics.

The first metric is the Mean Absolute Error (MAE), computed usingEquation (4), which offers a simple indication of the average absolute deviations. Anothervariant of MAE is the mean absolute percent error (MAPE) which indicates the percent of theaverage absolute deviations.

$$(4)\text{MAE} = \frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y}_i|$$

The second metric is the Mean Squared Error (MSE), determined using Equation (5),which places greater emphasis on larger errors by squaring the differences.

$$(5)\ \text{MSE} = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2$$

Lastly, the Coefficient of Determination (R^2) is used, as shown in Equation (6), to quantify the amount of variance in the outcome that the model accounts for.

$$(6)\text{R} = 1 - \frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}$$

In these equations (1), (2), and (3), yi denotes the actual values, $\hat{y}_i$ signifies the values predicted by the model, and $\bar{y}$ represents the average of the actual values. Analyzing MAE, MSE, and R^2 together provides a thorough evaluation of the model's effectiveness.

The figure 3 illustrates the progression of mean squared error (MSE) for both the training and validation sets during the training of the proposed model. Initially, the training loss shows some fluctuations, which are expected as the model continues to learn and adjust its parameters. However, as the training progresses, both the training loss and the validation loss exhibit a steady decline.

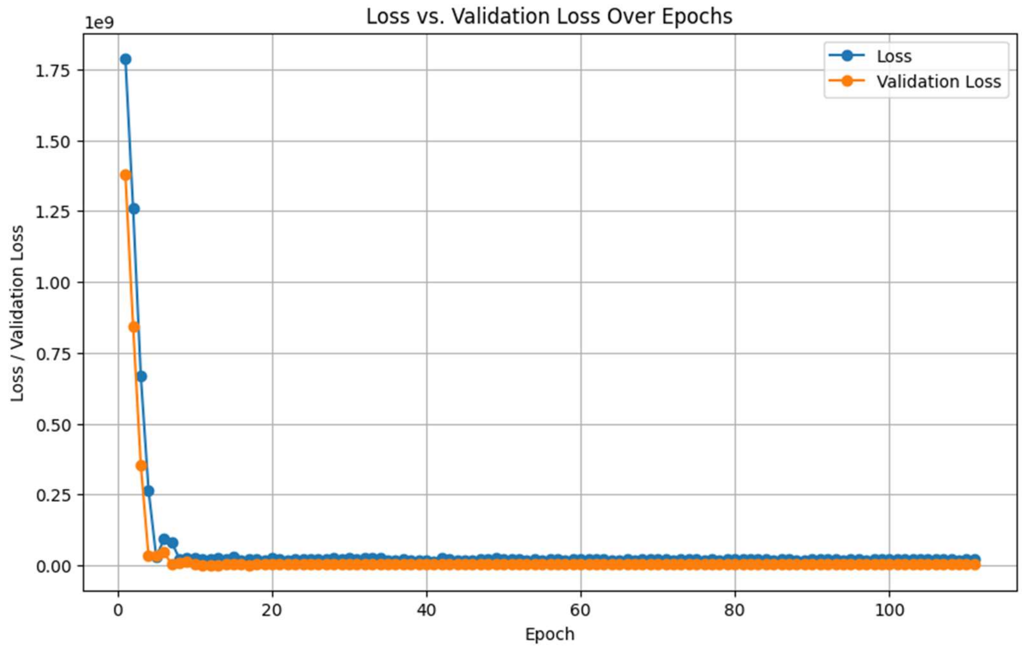


Figure 3.Training set mean squared error and validation set mean squared error with epochs for proposed model

This downward trend in both losses indicates that the model is successfully learning to minimize errors and generalizing well to unseen data. The reduction in validation loss, particularly, suggests that the model is not overfitting, as the performance on the validation set continues to improve alongside the training set.

These results demonstrate that the proposed model effectively learns from the training data while maintaining its predictive accuracy on the validation set, as evidenced by the converging loss values. The consistent reduction in both training and validation losses indicates the robustness of the model in terms of minimizing prediction errors over time.

Figure 4 provides a true vs predicted values by the proposed model predictions on test set for randomly selected 150 data points.

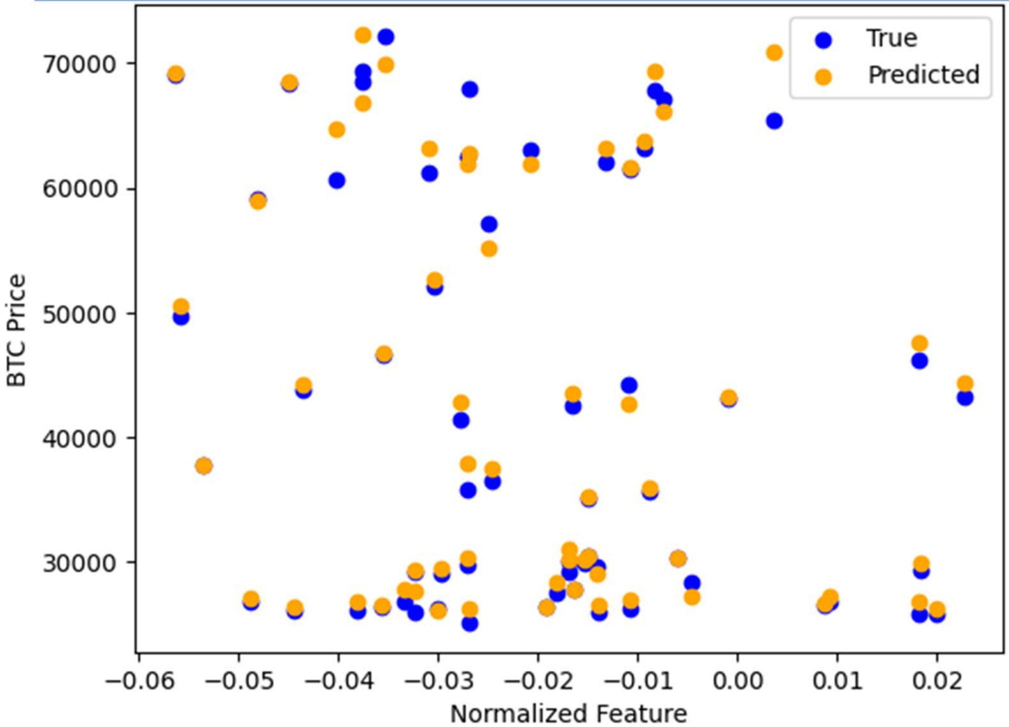
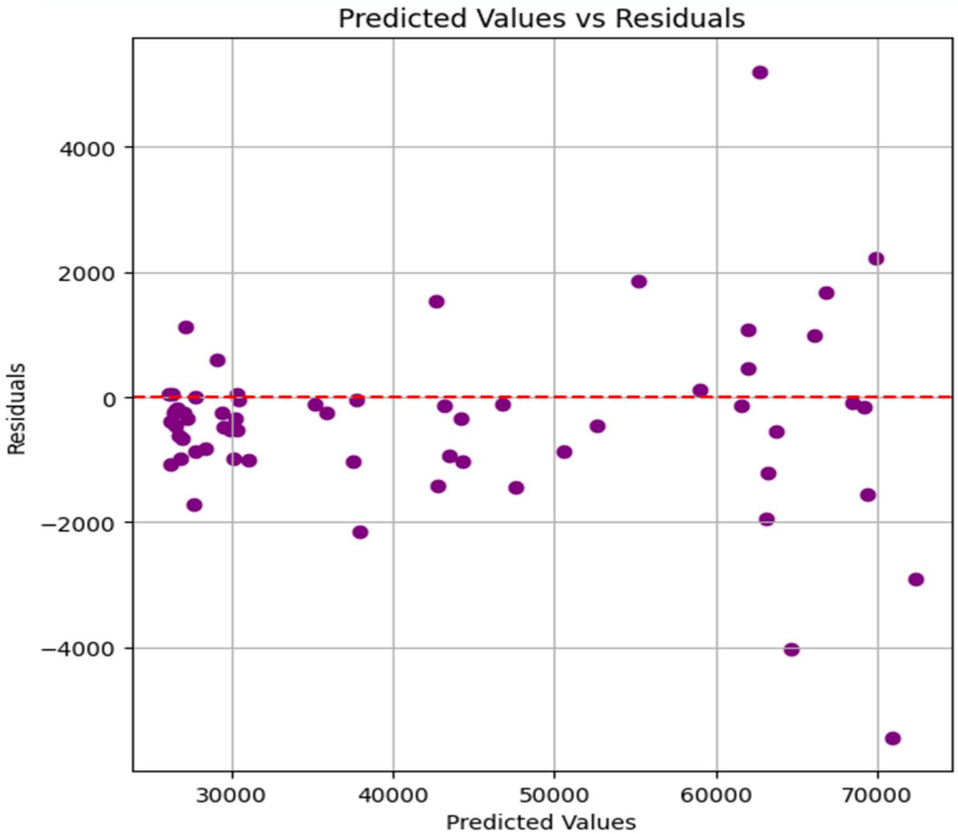


Figure 4.True vs Predicted values - Proposed model predictions on test set

Figure 5 presents a scatter plot that depicts the alignment between predicted values (on the y-axis) and actual values (on the x-axis). The predicted values from the proposed model are observed to be closely aligned with the red line, indicating a strong correlation.


Predicted Values vs Residuals
4000
2000
0
−2000
−4000
Residuals
30000
40000
50000
60000
70000
Predicted Values

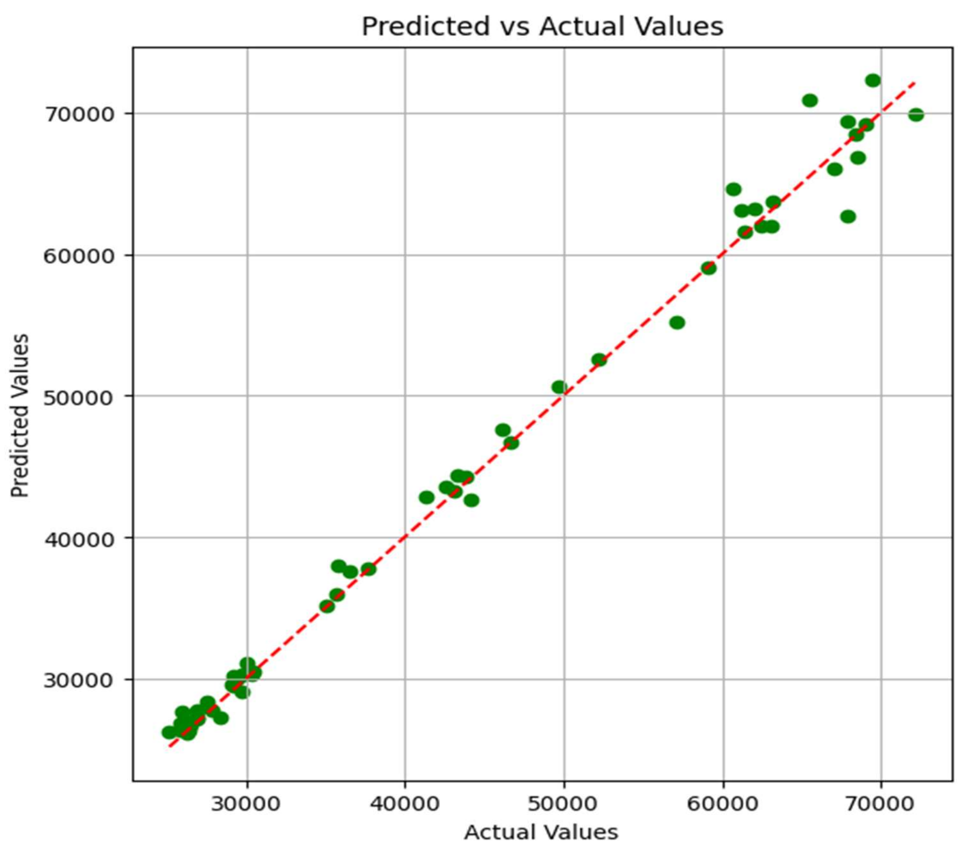

Predicted vs Actual Values
70000
60000
50000
40000
30000
Predicted Values
30000
40000
50000
60000
70000
Actual Values

Figure 5 . Proposed model: Prediction on test set (residual plots)

The analysis presented in the bar chart in Figure 6 provides a comparative evaluation of the MSE for different predictive models, including G Hongze et al., X Du et al, M Saad et al., S Syed et al, and the proposed model.

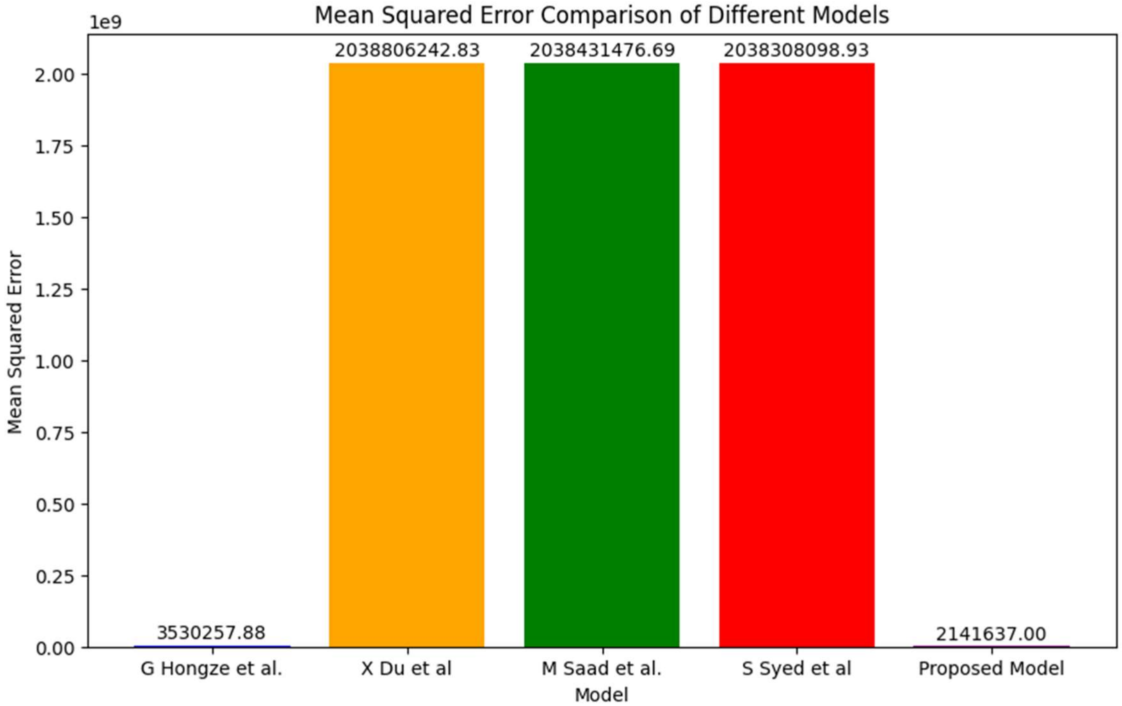


Figure 6 ..Comparison of predictive performance on test of Proposed model and comparable models

Lower MSE values indicate better predictive accuracy, as they represent smaller discrepancies between the predicted and actual values. The proposed model, however, demonstrated superior performance with an MSE of approximately 2,141,637.00, outperforming the G Hongze et al. by a margin of over 1.3 million in the MSE metric. This suggests that the proposed model, incorporates advanced methodologies or optimizations that allow it to more accurately capture the underlying patterns in the dataset, thus providing predictions with greater precision.

4. Concluding remarks and future scope

While traditional machine learning models like G Hongze et al. still provide competitive performance, the proposed model has shown to offer significant improvements in prediction accuracy. The high MSE values associated with the X Du et al, M Saad et al., and S Syed et al models suggest that these neural network architectures may not be well-suited for the specific characteristics of this dataset, potentially due to the challenges in tuning their hyperparameters or the nature of the data itself.The proposed model's ability to achieve thelowest MSE highlights its potential utility in applications requiring high precision in predictive modeling.

Conflict of Interest:

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

References

[1] Akba F., Medeni I.T., Guzel M.S., Askerzade I.: Manipulator detection in cryptocurrency markets based on forecasting anomalies. In: IEEE Access, vol. 9, pp. 108819–108831, 2021.

[2] Aljojo N., Alshutayri A., Aldhahri E., Almandeel S., Zainol A.: A nonlinear autoregressive exogenous (NARX)

neural network model for the prediction of timestamp influence on bitcoin value. In: IEEE Access, vol. 9, pp. 148611–148624, 2021.
[3] Azamjon M., Sattarov O., Cho J.: Forecasting Bitcoin Volatility through On-Chain and Whale-Alert Tweet Analysis using the Q-Learning Algorithm. In: IEEE Access, 2023.
[4] Du X., Tang Z., Wu J., Chen K., Cai Y.: A new hybrid cryptocurrency returns forecasting method based on multiscale decomposition and an optimized extreme learning machine using the sparrow search algorithm. In: IEEE Access, vol. 10, pp. 60397–60411, 2022.
[5] Guo H., Gao K., Yu Y., Liu Y., Fu L.: A Diluted Bitcoin-Dollar-Gold Mean Prediction Scheme Based on Periodic Prediction Method. In: IEEE Access, vol. 10, pp. 105319– 105327, 2022.
[6] Jagannath N., Barbulescu T., Sallam K.M., Elgendi I., Mcgrath B., Jamalipour A., AbdelBasset M., Munasinghe K.: An on-chain analysis-based approach to predict ethereum prices. In: IEEE Access, vol. 9, pp. 167972–167989, 2021.
[7] Jang H., Lee J.: An empirical study on modeling and prediction of bitcoin prices with bayesian neural networks based on blockchain information. In: IEEE access, vol. 6, pp. 5427–5437, 2017.
[8] Jay P., Kalariya V., Parmar P., Tanwar S., Kumar N., Alazab M.: Stochastic neural networks for cryptocurrency price prediction. In: Ieee access, vol. 8, pp. 82804–82818, 2020.
[9] Kim G., Shin D.H., Choi J.G., Lim S.: A deep learning-based cryptocurrency price prediction model that uses on-chain data. In: IEEE Access, vol. 10, pp. 56232–56248, 2022.
[10] Nayak S.C., Das S., Dehuri S., Cho S.B.: An elitist artificial electric field algorithm based random vector functional link network for cryptocurrency prices forecasting. In: IEEE Access, 2023.
[11] Otabek S., Choi J.: Twitter attribute classification with q-learning on bitcoin price prediction. In: IEEE Access, vol. 10, pp. 96136–96148, 2022.
[12] Park J., Seo Y.S.: Twitter sentiment analysis-based adjustment of cryptocurrency action recommendation model for profit maximization. In: IEEE Access, 2023.
[13] Saad M., Choi J., Nyang D., Kim J., Mohaisen A.: Toward characterizing blockchainbased cryptocurrencies for highly accurate predictions. In: IEEE Systems Journal, vol. 14(1), pp. 321–332, 2019.
[14] Shah A.S., Karabulut M.A., Akhter A.S., Mustari N., Pathan A.S.K., Rabie K.M., Shongwe T.: On the vital aspects and characteristics of cryptocurrencyA survey. In: Ieee Access, vol. 11, pp. 9451–9468, 2023.
[15] Sujatha R., Mareeswari V., Chatterjee J.M., Abd Allah A.M., Hassanien A.E.: A Bayesian regularized neural network for analyzing bitcoin trends. In: IEEE Access, vol. 9, pp. 37989–38000, 2021.
[16] Syed S., Iqbal A., Mehmood W., Syed Z., Khan M., Pau G.: Split-Second Cryptocurrency Forecast Using Prognostic Deep Learning Algorithms: Data Curation by Deephaven. In: IEEE Access, vol. 11, pp. 128644–128654, 2023.