IIETA International Information and Engineering Technology Association
Advancing the World of Information and Engineering

Mathematical Modelling of Engineering Problems
Vol. 10, No. 6, December, 2023, pp. 2255-2264
Journal homepage: http://iieta.org/journals/mmep

Predictive Modeling of NEAR Cryptocurrency Pricing Using Deep Learning: Influence of Bitcoin Market Movements

Mohammed Gollapalli[1], Atta-ur Rahman[2*], Omran Hakami[3], Mohammed Alhashim[3], Basheer Arab[3], Faisal Almashharwai[3], Mustafa Youldash[3], Ashraf Saadeldeen[4], Abdullah Alturkey[5], Dania Alkhulaifi[5]

[1] Department of Computer Information Systems (CIS), College of Computer Science and Information Technology (CCSIT), Imam Abdulrahman Bin Faisal University (IAU), P.O. Box 1982, Dammam 31441, Saudi Arabia
[2] SAUDI ARAMCO Cybersecurity Chair, Department of Computer Science, College of Computer Science and Information Technology (CCSIT), Imam Abdulrahman Bin Faisal University (IAU), P.O. Box 1982, Dammam 31441, Saudi Arabia
[3] Department of Computer Engineering (CE), College of Computer Science and Information Technology (CCSIT), Imam Abdulrahman Bin Faisal University (IAU), P.O. Box 1982, Dammam 31441, Saudi Arabia
[4] Department of Computer Science, Faculty of Computer Science and Information Technology (CCSIT), Al-Baha University, Al-Baha 65528, Saudi Arabia
[5] Department of Computer Science (CS), College of Computer Science and Information Technology (CCSIT), Imam Abdulrahman Bin Faisal University (IAU), P.O. Box 1982, Dammam 31441, Saudi Arabia

Corresponding Author Email: aaurrahman@iau.edu.sa

https://doi.org/10.18280/mmep.100641

Received: 25 July 2023
Revised: 15 September 2023
Accepted: 22 October 2023
Available online: 21 December 2023

Keywords:
cryptocurrency, Long Short-Term Memory, Gated Recurrent Unit, NEAR, Bitcoin, cybersecurity, deep learning

ABSTRACT

The emergent domain of cryptocurrency represents a significant development in financial markets, with a profound impact on investment strategies. This study seeks to develop a predictive model for forecasting the future prices of the NEAR cryptocurrency, a relatively unexplored asset within this volatile market. The objective is to furnish investors with a reliable tool for informed decision-making by anticipating market fluctuations. Focusing exclusively on the NEAR coin, the research endeavors to ascertain the correlative impact of Bitcoin market trends on its valuation. To date, extensive research has been conducted on Bitcoin price volatility; however, investigations into NEAR coin price predictions remain conspicuously absent. The dataset underpinning this analysis comprises two years of price and volume data for both NEAR and Bitcoin. Employing Long Short-Term Memory (LSTM) and Gated Recurrent Unit (GRU) deep learning architectures—algorithms renowned for their efficacy in time-series forecasting—the model is constructed to account for historical values when predicting future prices. The efficacy of LSTM and GRU algorithms in processing sequential datasets is well-documented in existing literature, reinforcing their selection for this study. The anticipated result is a predictive model that not only forecasts the future price of the NEAR coin with a high degree of accuracy but also elucidates the extent to which Bitcoin's market behavior influences NEAR's value. This contribution is poised to enhance the precision of cryptocurrency trading strategies, ultimately benefiting the broader financial community.

1. INTRODUCTION

Currency, an essential medium for the exchange of goods and services, has undergone a transformative evolution. This evolution has traversed from the rudimentary barter system to the sophisticated digital transactions of the modern era, propelled by a succession of technological advancements that have reshaped global payment systems [1]. Historically associated with Bitcoin, blockchain technology has emerged from its enigmatic origins to become a cornerstone of the volatile yet burgeoning field of cryptocurrencies [2]. As a novel payment mechanism, cryptocurrency is swiftly gaining traction on a global scale [1]. Among these, Bitcoin, Ethereum, and Litecoin have shown remarkable growth, with their values appreciating over time. Notably, Ethereum's surge in popularity has positioned it as the world's second-largest cryptocurrency [3].

Cryptocurrency markets are characterized by their instability, a stark contrast to the traditional commodity markets. Influenced by an amalgamation of technological, emotional, and regulatory factors, these markets are dynamic, ambiguous, and unpredictable [4]. From a trader's perspective within the cryptocurrency domain, the direction of price movement is secondary to the predictability of market trends [5]. Accurate prediction of cryptocurrency prices can revolutionize trading practices and exert a significant influence on the digital economy. The persistent increase in value and popularity of Bitcoin and other cryptocurrencies over recent years underscores the utility of such predictive systems for individuals and organizations alike [6].

Consequently, an extensive body of research has been dedicated to the study of digital currencies [6].

Extant research on cryptocurrency pricing has predominantly focused on deep learning algorithms adept at processing time-series data. Long Short-Term Memory (LSTM) and Gated Recurrent Units (GRU) are the algorithms most frequently employed for such data. Nonetheless, alternative methodologies have been explored to enhance predictive performance. For instance, the study [6] utilized machine learning algorithms to predict daily cryptocurrency price fluctuations. Meanwhile, research detailed in the study [7] adopted the random walk theory, a prevalent financial market model for stock valuation. In the case of Bitcoin, Ethereum, and Litecoin, methods were developed to discern market response patterns using Multi-Layer Perceptron (MLP) and LSTM models. Other investigations have concentrated on forecasting the pricing of specific cryptocurrencies, such as Bitcoin, the pioneer of decentralized payment systems, which differs markedly from centralized counterparts like PayPal [8]. This characteristic renders Bitcoin's payment system inherently unstable and subject to rapid change. The study [9] conducted a comprehensive examination and comparison of advanced deep learning techniques, including deep neural network (DNN) LSTM models, deep residual networks, and convolutional neural networks, and their combinations to predict Bitcoin prices. A hybrid neural network model, integrating a convolutional neural network (CNN) with an LSTM, was proposed in reference [10] to address the drastic fluctuations in Bitcoin's price and improve prediction accuracy. Moreover, the research presented in reference [11], employing GRU, LSTM, and bi-LSTM machine learning algorithms, demonstrated the efficacy of three types of recurrent neural network (RNN) algorithms in forecasting the values of cryptocurrencies such as Bitcoin, Ethereum, and Litecoin, exhibiting exceptional predictive capabilities. The present investigation is centered upon the NEAR cryptocurrency, with the objective of devising a predictive model for its valuation. Such a model would be a significant aid to investors, enhancing the decision-making process in trading activities. Furthermore, this study seeks to elucidate the influence of Bitcoin's price volatility on the NEAR token. Despite the extensive body of research focusing on Bitcoin price dynamics, it is surprising to note the absence of scholarly inquiry into the pricing mechanisms of the NEAR coin. The dataset underpinning this research comprises price and volume data for NEAR and Bitcoin over a two-year span (2020-2022).

To construct the predictive model, deep learning algorithms such as Long Short-Term Memory (LSTM) and Gated Recurrent Unit (GRU) are employed, selected for their capacity to incorporate historical data in forecasting future values—a critical feature for time-series data analysis. Such methodologies have been reliably effective in analogous research contexts. Prior to model training, meticulous data visualization techniques are applied to ensure data reliability and consistency, thereby facilitating the data cleaning and preprocessing stages. The dataset comprises "candles," indicative of the NEAR coin price movements, with each row representing a one-hour interval of price data. The prediction target is the closing price for the subsequent hour. Data retrieval for the study is conducted utilizing the Binance API, encompassing a two-year duration (2020-2022) for the experimental analysis.

The ambition of the model is not only to forecast the future price of NEAR but also to achieve an acceptable level of precision, whether the prediction is exact or approximate.

The structure of the remainder of the paper is as follows: Section 2 provides a comprehensive review of the literature pertinent to the topic; Section 3 describes the methodologies adopted in the study; Section 4 details the empirical analyses undertaken; Section 5 presents the results; Section 6 discusses these findings; and Section 7 offers concluding remarks.

2. LITERATURE REVIEW

A review of the extant literature reveals a variety of approaches to modeling the volatility prevalent in cryptocurrency markets. In the study conducted by Tanwar et al. [4], neural networks were enhanced by the incorporation of layer-wise randomization within the activation functions of observed features. The research employed Multi-Layer Perceptron (MLP) and Long Short-Term Memory (LSTM) models to capture the price movements of prominent cryptocurrencies: Bitcoin, Ethereum, and Litecoin. For the purpose of these models, a seven-day window with 23 distinct features was utilized. To standardize the dataset, two normalization techniques were employed, both contingent upon the mean values of the respective features. The MLP was found to yield a Mean Absolute Percentage Error (MAPE) of 4.017 for Bitcoin, 3.207 for Ethereum, and 3.836 for Litecoin, while the LSTM models exhibited MAPEs of 4.485 for Bitcoin, 4.012 for Ethereum, and 3.855 for Litecoin. The authors suggested that future research could profitably explore optimal hyperparameter tuning techniques to ascertain the most efficacious values [4].

Another research endeavor [5] examined the predictability of price movements across twelve cryptocurrency exchanges on both daily and minute-to-minute timeframes, employing an array of machine learning algorithms including Support Vector Machines (SVM), Logistic Regression, Artificial Neural Networks, Random Forests, and additional SVMs. The findings indicated that four algorithms consistently classified cryptocurrencies with an accuracy exceeding 50% for all studied currencies. The accuracy of most machine learning algorithms hovered between 55% and 65%. Notably, SVMs emerged as the most robust and high-performing models for predicting subsequent day returns, demonstrating a consistent fit above 50%, minimal variation across a broad range of products and timescales, and substantial generalization capabilities across various subperiods. Data sourced from the Bitfinex exchange was posited to be representative of the broader cryptocurrency market, thereby enabling the extrapolation of the findings to the market at large [5].

The objective of the study delineated presented by Mittal et al. [6] was to forecast cryptocurrency market prices based on historical trends. The researcher sought to discern daily market trends by examining features pertinent to the pricing of digital currencies. The CryptocoinHistoricPrice dataset, obtained from the renowned repository Kaggle, served as the basis for the analysis. This dataset, consisting of 659360 entries and nine columns, included a diverse set of descriptive variables such as date, open price, high price, low price, close price, volume, market cap, type of coin, and price delta. A dataset was selected that allowed for an examination of the market price behavior and structure of cryptocurrencies. Remarkably, a 99% accuracy rate was achieved on the test dataset when all columns were utilized in the analysis [6]. In the realm of cryptocurrency forecasting, a sophisticated deep-learning methodology comprising Gated Recurrent Units (GRU) and Long Short-Term Memory (LSTM) networks has been developed to predict the prices of Litecoin and Zcash, taking into account the influence of parent coins. Data for this analysis, consisting of 3408 points—1737 for Litecoin and 1671 for Zcash—were sourced from the financial platform Investing.com. The efficacy of the proposed model was evaluated over varying temporal windows (1 day, 3 days, 7 days, and 30 days), with the 1-day window yielding a Mean Squared Error (MSE) of 0.02038 for Litecoin and 0.00461 for Zcash. The researchers envisage extending their methodology to encompass a broader spectrum of cryptocurrencies with complex interdependencies [7].

A distinct investigation aimed at predicting Bitcoin's closing price was conducted, involving a dataset corresponding to 2590 days. The dataset was partitioned into an 80% training subset, with the remaining 20% allocated for testing. Through the application of 5-fold cross-validation, various algorithms were scrutinized. To enhance model performance, features exhibiting correlations below 0.75 or above 0.95 were excluded from the training process. Furthermore, the MinMaxScaler technique was employed to normalize the training set. Results indicated that the LSTM algorithm excelled in predicting Bitcoin's closing price with a root mean square error (RMSE) of 0.087 when based on data from the preceding 7 days. Conversely, the Deep Neural Network (DNN) algorithm surpassed others in forecasting significant price movements [9].

In a study conducted by Li and Dai [10], a hybrid network comprising a convolutional neural network (CNN) and an LSTM neural network was proposed for Bitcoin price prediction. The CNN component served to extract salient features from the data, which were then fed into the LSTM for temporal analysis. The dataset was bifurcated into two categories: one including Bitcoin's price parameters—opening, closing, highest, and lowest prices—and the other encompassing macroeconomic factors along with investor attention metrics. The model's performance was gauged using mean absolute error (MAE), RMSE, and mean absolute percentage error (MAPE), with the latter yielding a result of 2.35% [10].

In recent research [12], a predictive model for Bitcoin's closing price was proposed, leveraging three prominent time series algorithms to manage the cryptocurrency's inherent volatility effectively. The dataset, comprising 1460 daily observations with seven features, was divided, allocating 80% for training and the remaining 20% for evaluation. Normalization of the training data was accomplished using the MinMax scaler to enhance model performance. Further refinement was achieved by adjusting the weights within the model, assigning greater importance to salient features. It was observed that the Gated Recurrent Unit (GRU) algorithm outperformed both the Long Short-Term Memory (LSTM) and the Recurrent Neural Network (RNN) in terms of prediction accuracy for a 7-day horizon, achieving a Root Mean Square Error (RMSE) of 0.051. This was accomplished within a 12-window size context, also demonstrating a reduced computational demand.

The synthesis of extant literature underscores the burgeoning relevance of Bitcoin within online commerce globally, particularly under the aegis of blockchain technology in cloud-based frameworks [13, 14]. The emphasis of such research pivots on the secure facilitation of financial transactions worldwide, especially in scenarios with multiple stakeholders, including cloud-based resource provisioning [15, 16]. Furthermore, the healthcare sector is experiencing a surge in telemedicine research, where technologies such as Bitcoin, blockchain, Internet of Things (IoT), and cloud computing are becoming increasingly pivotal [17, 18]. These technologies are particularly instrumental in ensuring the security and integrity of healthcare data [19, 20].

The compilation of studies presented herein affirms the critical role of advanced computational models in understanding and predicting cryptocurrency trends. The integration of such models with emerging technologies showcases a multidisciplinary approach that spans from financial security to healthcare, emphasizing the expansive utility of blockchain technology.

3. DESCRIPTION OF THE PROPOSED TECHNIQUES

This section provides technical description for the algorithms included in the experiment. Three different algorithms will be implemented, convolutional neural network [21, 22], Gated Recurrent Unit, and Long Short-Term Memory [23]. Based on previous studies from literature reviews, it is observed that these algorithms achieved the highest performance [24, 25].

3.1 Convolutional neural network

CNN is one of the deep learning algorithms. Convolutional layers are connected to multiple pooling layers, and at the end, flattening layers or global max layers are used to reduce the dimensions of the features. Approximately, CNN takes images as input. Applying convolution layers to the image will extract the features of the input image and reduce the size of the image while keeping the relationship between the parts. Basically, it takes a small matrix called the kernel and a part of the input tensor that is the same size as the kernel. After the tensor and kernel are combined, create a feature map by summing up the result, and keep shifting the tensor across the input until it covers the entire input image. Convolution layers are followed by pooling layers to reduce the complexity of the computation by minimizing the size of the representation. Last, CNN flattens the feature to make it one-dimensional and sends it to a deep neural network. Activation functions remove negative values from convolution layers by replacing them with zeros.

$$\mathrm{ReLU}\,(x) = \max\,(0, x) \tag{1}$$

3.2 Gated recurrent unit

GRU is a recurrent network type. It aims to fix the standard recurrent neural network, which is vanishing gradient. GRU uses two gates which are the update gate and reset gate to decide what should be passed to the output [26]. These gates help the network keep the relevant information and cut out the irrelevant ones. The way the update gate works is when a feature is inputted to neural first multiply the past information with its weight same goes for the input next sum them both together and apply sigmoid activation function, so the result becomes between 0 and 1. The update gate gives the model the ability to pick the information to pass with the input to the output. The reset gate is like the update gate mathematically but differs in weight and purpose. It is used to decide how much of the saved information to drop.

3.3 Long Short-Term Memory

An LSTM is one of the types of recurrent neural networks. It is used in many applications that work repeatedly, such as speech recognition, text generation, captioning videos, and sentiment analysis. It identifies the associations between elements and inputs. As well as containing the memory, the LSTM contains gates, such as input gates, output gates, and forget gates. As a result of this gate updating the state by sending the input to the sigmoid function, it determines which weights are to be considered; if the output is 0, it will be ignored, if it is 1, it will be saved. Sigmoid functions are referred to as output gates because they take the information from previous inputs and classify the information from the current state into what the next state is supposed to be. By applying the Sigmoid function, the forget gate either deletes unnecessary data or saves it for the next process; if the result is almost 0, then it is deleted, otherwise it is considered and saved. Sigmoid function output is multiplied by tanh function output in the input and output gates so the sigmoid can select the required values from tanh function output to update weights and adjust the neurons.

Cell states are transferred from the memory to the next states of neural networks through memory. Due to this, the RNN algorithm has a vanishing gradient problem, which occurs when the learning rate drops to the point that the weights can't be updated enough to produce the required output for recurrent learning, as the training data for the previous layers aren't saved. With LSTMs, this problem is overcome by allowing the cells to perform the required computations for each iteration, thus allowing the model layers to be trained without losing the required results.

4. EMPIRICAL STUDIES

In this section, a description of the collected dataset will be detailed and how the preprocessing was on the data. Also, this section contains a statistical analysis visual analysis of the dataset. Followed by a detailed description of the experiment setup and how the models are built. In the last section, a performance measure of the model would be used as part of this research that has been comprehensively detailed.

4.1 Description of dataset

The data is about the candles that represent the price movements for NEAR cryptocurrency. Each row stands for one candle, and each candle represents one hour of price movement for NEAR cryptocurrency. The data was collected using Binance API. The price records start from 14/10/2020 12:00:00 and ends at 02/03/2022 12:00:00 which gives us a total of 12098 samples. The data has five features, the first one is the open price which refers to the start price for the candle. The next feature refers to the price the candle is closed at. The last two features are high and low. These two refer to the lowest and highest prices that the candle retched during its movement. The data is divided into three sets train with 70%, validation with 20%, and test holding 10% of the data. Be figure below shows the price movement.

4.2 Statistical analysis of the dataset

A statistical analysis of the data is shown in the two tables below. The summary of the results includes the mean, median, 25%, 75%, standard deviation, maximum, minimum, and the correlation of the datasets. Table 1 provides the statistical analysis of the dataset classes in terms of their mean, median, quartile, minimum and maximum values, respectively.

Similarly, Table 2 presents the correlation between the classes of the dataset which are normalized values between 0 and 1. Here 0 represents zero or no-correlation while 1 represents the maximum possible correlation [27, 28].

Table 1. Statistical analysis

	Mean	Median	25%	75%	Std	Min	Max
Open	5.737	5.0083	0.526	5.0083	4.2445	0.526	20.295
High	5.813	5.0740	0.535	5.0740	4.297	0.535	20.597
Low	5.663	4.9284	0.525	4.9284	4.194	0.525	20.042
Vol	419356	280858	0.000	280858	515699	0.000	20849410
Close	5.738	5.0070	0.527	5.0070	4.2446	0.527	20.297

Table 2. Correlation

	Open	High	Low	Volume	Close
Open	1.000000	0.999563	0.999762	0.169556	0.999589
High	0.999563	1.000000	0.999432	0.187136	0.999696
Low	0.999762	0.999432	1.000000	0.162039	0.999756
Volume	0.169556	0.187136	0.162039	1.000000	0.173785
Close	0.999589	0.999696	0.999756	0.173785	1.000000

4.3 Pre-processing

A key step in any kind of ML problem is pre-processing the data. There are many pre-processing techniques involved, such as normalizing the data, slicing the window, simple moving average, and converting the data into stationarity, especially when dealing with time series problems. In this section, all the techniques used in the study are explained respectively.

i. Normalization

When dealing with time series problems, normalization is important so that the data are in the same range [29, 30]. In detail, each feature in the dataset has a different scale. Hence, it is better to apply normalize as it achieves higher accuracy in less time. For normalizing the data, the MinMaxScaler function from Scikit-Learn was used. All the algorithms are normalized between 0 and 1 except Gated Recurrent Unit where it performed better with -1 to 1. This was accomplished after splitting the dataset into training, validation, and testing sets. Because this fitting should only be on the training data. After fitting the scalar on the training set, each set is transformed using the transform function.

$$x - scaled = x - std\,(\max - min)/min \qquad (2)$$

ii. Slicing Window

To predict the future value of the cryptocurrency closing price, it is necessary to consider the past values. The dataset was reshaped to consider one day for predicting the next hour.

4.4 Experimental setup

The experiment was conducted using the Python programming language. In this experiment, Keras, TensorFlow, and Scikit-learn libraries were used. The dataset was divided into training, validation, and testing sets with 70%, 20%, and 10%, respectively. This division is purely based on the author's own intuition and mainly based on hit and trial methods used in the past. As part of the split, the first segment of the dataset was for training, the second segment was for validation, and the rest was for testing. The experiment was performed using the LSTM, GRU, CNN, and CNN-LSTM techniques. All the steps for building the model will be explained in this section.

4.4.1 Long-short term memory and gate recurrent unit

A sequential model was created using Keras in TensorFlow. The structure of the LSTM layers is 80, 100 with return sequence, 1 respectively. In contrast, GRU layers are 100,100 with return sequence 1, respectively. Input shape was (24, 5), where 24 represents the next 24 hours and 5 represents the number of features. After the first layer, RepeatedVectore was applied, and the n parameter was set to be 1. Moreover, TimeDistribution was used as the last layer with the Dense layer. To avoid overfitting problems and optimize the model, there are some important configurations that were used during the experiment. All the important configurations are listed in Table 3, Table 4, and Table 5 for avoiding overfitting, parameters for compile method and fit method, respectively.

Table 3. Avoid overfitting

Methods	Parameters	Values
ModelCheckpoint	Save_best_only	True
EarlyStopping	Monitor, patience, mode, restore_best_weights	'val_loss', 30, 'min', True respectively
ReduceLROnPlateau	Monitor, factor, mode, patience, min_lr	'val_loss', 0.2, 'min', 3, 0.001 respectively

Table 4. Compile method

Parameters	Values
Optimizer	Adam with 0.003 learning rate
Loss	Mean Squared Error
Metrics	Root Mean Squared Error

Table 5. Fit method

Parameters	Values
Batch_size	64
Epochs	100
Shuffle	False

4.4.2 Convolutional neural network (CNN)

TensorFlow's Keras library was used to create the sequential model. The structure of CNN layers is 100, 100, MaxPolling, flatten, 1 respectively. Convid1D was used for the first and second layers. The kernel size is set to 2 for both layers, padding is the same, and ReLu is the activation function. To avoid overfitting problems and optimize the model, there are some important configurations that were applied during the experiment. All the important configurations are listed in Table 3, Table 4 and Table 5 for avoiding overfitting, parameters for compile method and fit method, respectively.

4.4.3 CNN-LSTM encoder

FunctionalAPI in TensorFlow and the Keras library were used to create the model. The structure is made up of two parts. In the first part, we have the CNN layers, where Conv1D is used to create the first two layers with 100 and 100, respectively. Each layer has a kernel size of 2, padding is set to the same value, and ReLu is used as the activation function. Next, the two layers were concatenated and passed to the LSTM layers. The structure of LSTM layers is 100, and 100 respectively. Both layers will return sequence. After that, the previous two layers were merged and passed to the last LSTM layer and the normal Dense layer [31-36]. They consist of 50 and 1 respectively. To avoid overfitting problems and optimize the model, there are some important configurations that were used during the experiment. All the important configurations are listed in Table 3, Table 4, and Table 5 for avoiding overfitting, parameters for compile method and fit method, respectively.

5. RESULTS AND COMPARISON

In this section, the results of the proposed method and a comparison with other algorithms are comprehensively discussed. The experiments have been conducted with four algorithms which are LSTM, GRU, CNN, and CNN-LSTM. Also, the results of these algorithms are gone into a comparison to have the best result among all experiments. Results shown below are measured based on the loss which is Mean Squared Error (MSE) and the metric which is Root Mean Squared Error (RMSE). All measurements are calculated after the optimization phase and data pre-processing. The data which are used in testing set is 20% of the dataset and the results are gained based on the model's performance with this set. Table 6 shows the results of the evaluation of all four algorithms. All the results are measured based on the performance of the algorithms with the test set. The MSE shows the loss function value where the RMSE is acting like an accuracy of the prediction. Several optimization techniques have been used before evaluating any of the mentioned models. The lowest MSE and RMSE is obtained when using GRU. However, some algorithms have obtained noticeable results such as LSTM and CNN-LSTM. They both got near to 0.035 RMSE, where for MSE values LSTM achieved 0.0012 and CNN-LSTM achieved 0.0013.

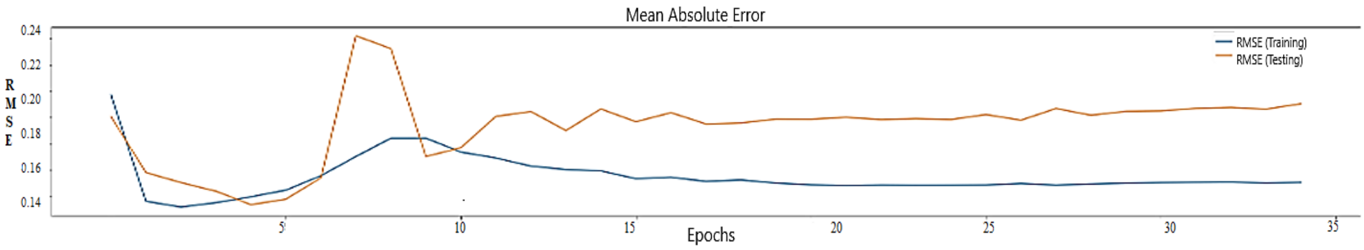

Figure 1. RMSE of GRU with training set

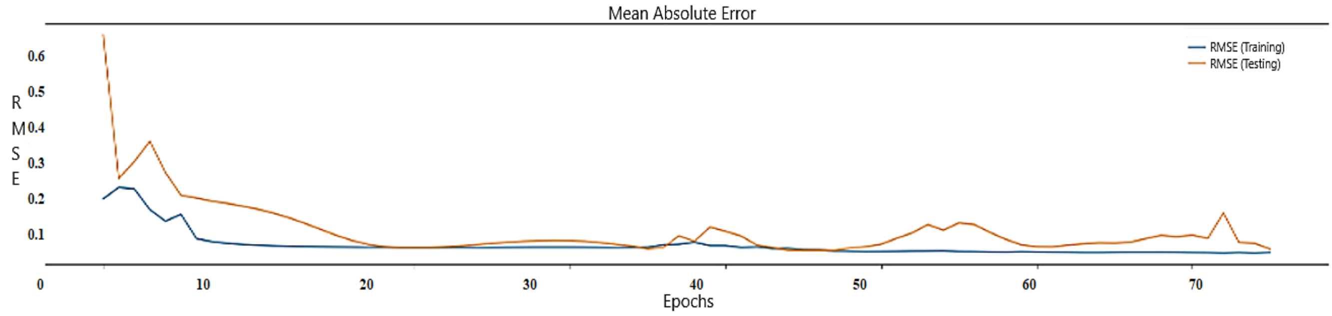

Figure 2. RMSE of LSTM with training set

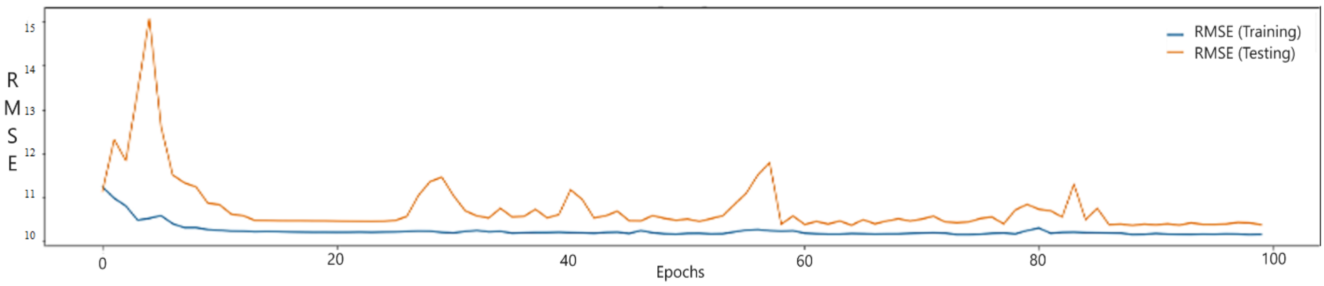

Figure 3. RMSE of CNN with training set

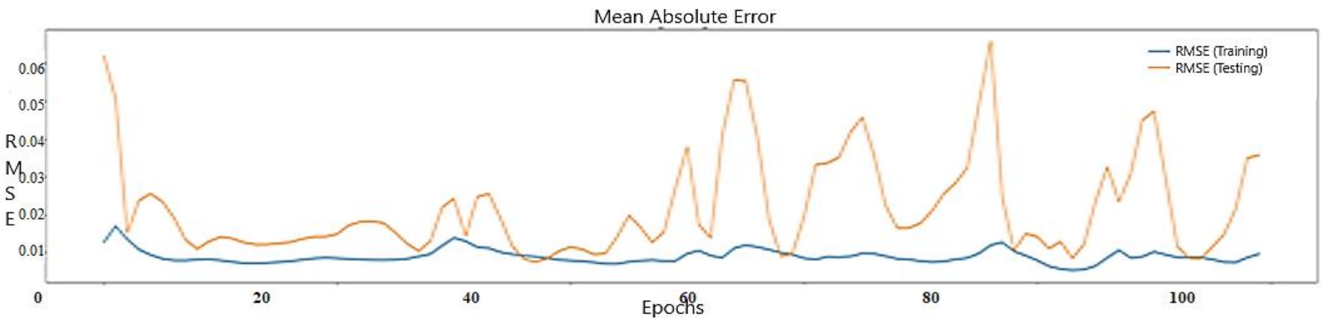

Figure 4. RMSE of CNN-LSTM with training set

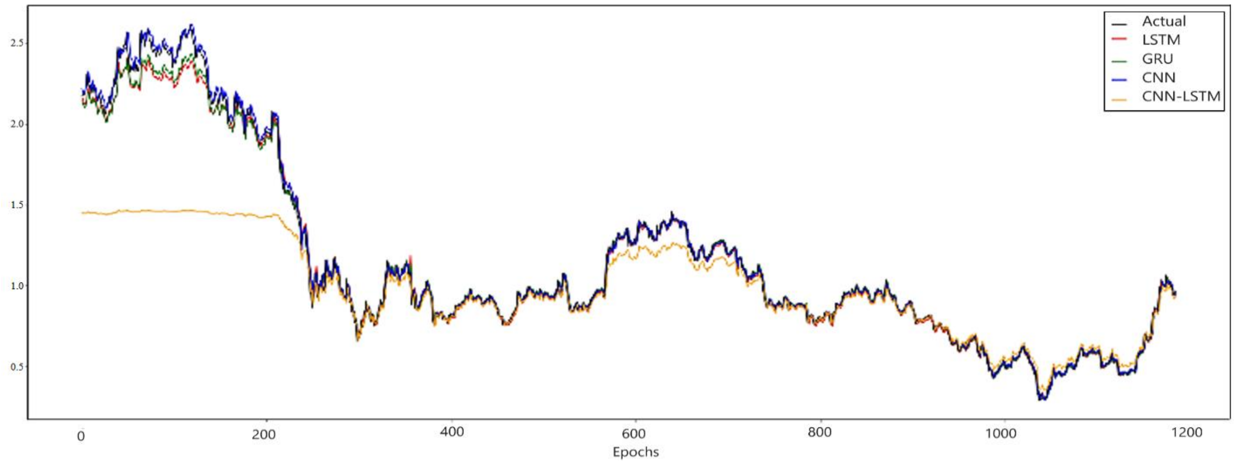

Figure 5. Comparison the prediction of all models against the actual value

Table 6. RMSE & MSE comparison

Algorithm	RMSE	MSE (Loss)
LSTM	0.0352	0.0012
GRU	0.0336	0.0011
CNN	0.0384	0.0015
CNN-LSTM	0.0357	0.0013

Based on these results the best GRU performed better than the other algorithms. GRU has achieved root Mean Squared Error (RMSE) of 0.0336 and Mean Squared Error (MSE) of 0.0011. The obtained results by GRU can be improved later which can be marked as a future work. As shown in the table, the RMSE and MSE that the GRU achieved is mostly predictable because of the capabilities that the GRU have. The used data requires a network that can memorize the previous steps and the assigned weight so it can learn from history.

Figure 1 to Figure 4 display the values of RMSE while training the algorithms. It can be noticed that all the algorithms started with high loss except CNN. That is because CNN passes the first values as it is without any modification or weights adjusting and CNN does not take history into account. The figures proof that GRU obtained the best RMSE and performed better than the other models. The RMSE for training is less than RMSE for validation in all experiments. Also, it is smoother which means there are no huge ups and downs while the number of epochs is increasing. For better demonstration another comparison based on the prediction of all models against the actual value of the target. This comparison can show how the prediction of the model is close to the actual value. Additionally, it illustrates model's sustainability and consistently. Figure 5 shows the comparison between each model prediction and the actual value. The x-axis represents the number of days, or it can be considered as the date, where the y-axis holds the normalized price of NEAR.

The prediction is performed using the testing set. The black line represents the actual value of the target where each algorithm is represented with a different color. CNN-LSTM was very far from the actual value at the beginning. However, GRU and CNN were very close to the actual value and at some point, getting similar values. From the figure above it's confirmed that the GRU model performs.

6. DISCUSSION

The results of the application of the GRU model for the NEAR prediction are displayed in Table 7. It shows how the closing price differs with the changing of other attributes. The relation is considered as categorization intervals and non-interval for the classes.

Table 7. Open price & target

Open Price (Intervals)	Target Class: Close Price (Interval)
1.812 – 2.377	1.825 – 2.381
0.446 – 0.533	0.446 – 0.933
0.832 – 0.813	0.832 – 0.401
0.634 – 0.446	0.550 – 0.551
2.424 – 1.946	2.424 – 1.525
0.813 – 0.634	0.888 – 0.851
2.424 – 1.946	2.459 – 1.485
1.946 – 1.032	1.856 – 1.804
2.377 – 2.424	2.481 – 2.284

The table above shows an interval relation between the Open attribute and the target attribute which is the Closing price. The relation between the opening price and closing price shows that they are close to each other when opening and closing in their intervals. For example, if the price opens between 0.813 – 0.634, the close price is going to be 0.888 – 0.851, the two attributes are near to each other in their values.

The graph given in Figure 6 shows a non-linear relation between opening and closing prices. The graph illustrates that the prices are close to each other in each candle. The x-axis shows candles, and the y-axis shows the opening and closing prices, respectively. Table 8 shows that when the coin is getting the highest price in a specific candle, the range of the closing price is getting in intervals, for example, if the highest price is 0.448 in candle 1051 in graph, the closing price of the coin is going to be 0.365. Likewise, Table 9 shows the relation between the High price of a specific candle and the closing price of the same candle.

Figure 6. Open & close

Figure 7. Low & close

Table 8. Highest price & target

High	Target Class: Close Price
1.745	1.702
2.240	2.295
2.315	2.642
1.862	1.697
1.038	1.021
1.049	1.076
0.876	0.963
1.000	1.061
0.448	0.365

Table 9. Lowest price & target

Low	Target Class: Close Price
1.770	1.851
2.115	2.196
1.114	1.263
0.9684	0.9684
1.695	1.862
1.265	1.361
0.154	0.155
1.2176	1.2176
1.3614	1.3622

The relation between the lowest price and the closing price is represented in the table. The table shows a specific low and closing price for some candles. For example, if the price was low in candle 253 in the graph, the closing price is 1.263. Figure 7 shows the relation between the low and close prices for each candle for the dataset. Also here, the prices are close to each other. For example, if the lowest price in candle 1177 is 0.453, the closing price is 0.463, which is almost close.

7. CONCLUSION

Introducing cryptocurrency, a new form of payment closely related to blockchain Cryptocurrency is a new way to pay that is rapidly gaining attention all over the world. The value of certain cryptocurrencies such as Bitcoin is increasing gradually while their growth rate is rapid. If the trend is predictable, it does not matter whether prices are rising or falling from the perspective of a cryptocurrency trader. Cryptocurrency price prediction will have a huge influence on the digital market and could generate an evolution in trading. Organizations would benefit from creating a system for protection, not only as a whole but as well as individuals. In addition to these factors, the digital currency has also been subject to extensive research. The correct pricing of various cryptocurrencies has been studied extensively through applying deep learning algorithms to time series data. Most of these studies are concerned with time series and have used different deep learning algorithms. In the case of this type of data, the most used algorithms were LSTM and GRU, respectively. Data used in this research was derived from the candles that showed the price changes of the NEAR coin. In this study, the data was collected based on the "Binance" API for a period of two years. Every row represented a sample data point, which represented one candle in the data. Every candle in our research depicted the fluctuation in NEAR coin prices over one hour. Each candle's targeted prices for the next hour’s close. As a last step, the model should forecast future coin prices as a function of either the exact or estimated price. The outcomes of this research proved that the GRU performed better than the other algorithms that are used in this study. The MSE was 0.0011 for GRU, also it had a RMSE of 0.0336. By using GRU, results could be improved further. As part of the future work, the data could be modified to use the moving average instead of the actual price and add more factors that affect the price. Further, transfer learning techniques can be investigated for better prediction of the currency value [37-40].

ACKNOWLEDGEMENTS

The authors would like to acknowledge SAUDI ARAMCO Cybersecurity Chair at College of Computer Science and Information Technology (CCSIT), Imam Abdulrahman Bin Faisal University (IAU), Dammam, Saudi Arabia for funding this research.

REFERENCES

[1] Hamukuaya, N.H. (2021). The development of cryptocurrencies as a payment method in South Africa. Potchefstroom Electronic Law Journal, 24(24): 1-23. https://doi.org/10.17159/1727-3781/2021/v24i0a9364

[2] Jubb, R. (2015). The real value of equality. The Journal of Politics, 77(3): 679-691. https://doi.org/10.1086/681262

[3] Wang, R. (2021). Multiple regression tests and prediction for Ethereum transaction value. SSRN Electronic Journal. https://doi.org/10.2139/ssrn.3879952

[4] Tanwar, S., Patel, N.P., Patel, S.N., Patel, J.R., Sharma, G., Davidson, I.E. (2021). Deep learning-based cryptocurrency price prediction scheme with inter-dependent relations. IEEE Access, 9: 138633-138646. https://doi.org/10.1109/ACCESS.2021.3117848

[5] Akyildirim, E., Goncu, A., Sensoy, A. (2021). Prediction of cryptocurrency returns using machine learning. Annals of Operations Research, 297: 3-36. https://doi.org/10.1007/s10479-020-03575-y

[6] Mittal, R., Arora, S., Bhatia, M.P.S. (2018). Automated cryptocurrencies prices prediction using machine learning. International Journal of Soft Computing, (8)4: 1758-1761.

[7] Jay, P., Kalariya, V., Parmar, P., Tanwar, S., Kumar, N., Alazab, M. (2020). Stochastic neural networks for cryptocurrency price prediction. IEEE Access, 8: 82804-82818. https://doi.org/10.1109/ACCESS.2020.2990659

[8] Qureshi, M.M., Yunus, F.B., Li, J.Q., Ur-Rahman, A., Mahmood, T., Ali, Y.A.A. (2023). Future prospects and challenges of on-demand mobility management solutions. IEEE Access, 11(1): 114864-114879. https://doi.org/10.1109/ACCESS.2023.3324297

[9] Ji, S., Kim, J.M., Im, H. (2019). A comparative study of bitcoin price prediction using deep learning. Mathematics, 7(10): 898. https://doi.org/10.3390/math7100898

[10] Li, Y., Dai, W. (2020). Bitcoin price forecasting method based on CNN-LSTM hybrid neural network model. The Journal of Engineering, 2020(13): 344-347. https://doi.org/10.1049/joe.2019.1203

[11] Hamayel, M.J., Owda, A.Y. (2021). A novel cryptocurrency price prediction model using GRU, LSTM and bi-LSTM Machine Learning Algorithms. AI, 2(4): 477-496. https://doi.org/10.3390/ai2040030

[12] Awoke, T., Rout, M., Mohanty, L., Satapathy, S.C. (2021). Bitcoin price prediction and analysis using deep learning models. In Communication Software and Networks, 134. Springer, Singapore.

[13] Alhaidari, F., Rahman, A., Zagrouba, R. (2023). Cloud of things: Architecture, applications and challenges. Journal of Ambient Intelligence and Humanized Computing, 14: 5957-5975. https://doi.org/10.1007/s12652-020-02448-3

[14] Ahmad, M., Qadir, M.A., Rahman, A., Zagrouba, R., Alhaidari, F., Ali, T., Zahid, F. (2023). Enhanced query processing over semantic cache for cloud based relational databases. Journal of Ambient Intelligence and Humanized Computing, 14: 5853-5871. https://doi.org/10.1007/s12652-020-01943-x

[15] Rahman, A., Ibrahim, N.M., Musleh, D., Khan, M.A.A., Chabani, S., Dash, S. (2022). Cloud-based smart grids: opportunities and challenges. In Biologically Inspired Techniques in Many Criteria Decision Making, Smart Innovation, Systems and Technologies, 271, Springer, Singapore.

[16] Rahman, A., Dash, S., Ahmad, M., Iqbal, T. (2021). Mobile cloud computing: A green perspective. In

Intelligent Systems, Lecture Notes in Networks and Systems, 185, Springer, Singapore.
[17] Khan, M.A., Abbas, S., Atta, A., Ditta, A., Alquhayz, H. Khan, M.F., Rahman, A., Naqvi, R.A. (2020). Intelligent cloud based heart disease prediction system empowered with supervised machine learning. Computers, Materials & Continua, 65(1): 139-151. https://doi.org/10.32604/cmc.2020.011416
[18] Musleh, D., Alotaibi, M., Alhaidari, F., Rahman, A., Mohammad, R.M. (2023). Intrusion detection system using feature extraction with machine learning algorithms in IoT. Journal of Sensor and Actuator Networks, 12(2): 29. https://doi.org/10.3390/jsan12020029
[19] Ahmed, M.S., Rahman, A., AlGhamdi, F., AlDakheel, S., Hakami, H., AlJumah, A., AlIbrahim, Z., Youldash, M., Alam Khan, M.A., Ahmed, M.I.B. (2023). Joint diagnosis of pneumonia, COVID-19, and tuberculosis from chest X-Ray images: A deep learning approach. Diagnostics, 13(15): 2562. https://doi.org/10.3390/diagnostics13152562
[20] Gollapalli, M., Rahman, A., Alkharraa, M., Saraireh, L., AlKhulaifi, D., Salam, A.A., Krishnasamy, G., Alam Khan, M.A., Farooqui, M., Mahmud, M. (2023). SUNFIT: A machine learning-based sustainable university field training framework for higher education. Sustainability, 15(10): 8057. https://doi.org/10.3390/su15108057
[21] Rahman, A. (2023). GRBF-NN based ambient aware realtime adaptive communication in DVB-S2. Journal of Ambient Intelligence and Humanized Computing, 14: 5929-5939. https://doi.org/10.1007/s12652-020-02174-w
[22] Rahman, A., Dash, S., Luhach, A.K. (2021). Dynamic MODCOD and power allocation in DVB-S2: A hybrid intelligent approach. Telecommunication Systems, 76: 49-61. https://doi.org/10.1007/s11235-020-00700-x
[23] Alqarni, A., Rahman, A. (2023). Arabic Tweets-based sentiment analysis to investigate the impact of COVID-19 in KSA: A deep learning approach. Big Data and Cognitive Computing, 7(1): 16. https://doi.org/10.3390/bdcc7010016
[24] Ahmed, M.I.B., Alotaibi, R.B., Al-Qahtani, R.A., Al-Qahtani, R.S., Al-Hetela, S.S., Al-Matar, K.A., Al-Saqer, N.K., Rahman, A., Saraireh, L., Youldash, M., Krishnasamy, G. (2023). Deep learning approach to recyclable products classification: Towards sustainable waste management. Sustainability, 15(14): 11138. https://doi.org/10.3390/su151411138
[25] Musleh, D.A., Olatunji, S.O., Almajed, A.A., Alghamdi, A.S., Alamoudi, B.K., Almousa, F.S., Aleid, R.A., Alamoudi, S.K., Jan, F., Al-Mofeez, K.A., Rahman, A. (2023). Ensemble learning based sustainable approach to carbonate reservoirs permeability prediction. Sustainability, 15(19): 14403. https://doi.org/10.3390/su151914403
[26] Talha, M., Sarfraz, M., Rahman, A., Ghauri, S.A., Mohammad, R.M., Krishnasamy, G., Alkharraa, M. (2023). Voting-based deep convolutional neural networks (VB-DCNNs) for M-QAM and M-PSK Signals classification. electronics, 12(8): 1913. https://doi.org/10.3390/electronics12081913
[27] Ibrahim, N.M., Gabr, D.G., Rahman, A., Musleh, D., AlKhulaifi, D., AlKharraa, M. (2023). Transfer learning approach to seed taxonomy: A wild plant case study. Big Data and Cognitive Computing, 7(3): 128. https://doi.org/10.3390/bdcc7030128
[28] Rahman, A. (2018). Efficient decision based spectrum mobility scheme for cognitive radio based V2V communication system. Journal of Communications, 13 (9): 498-504. https://doi.org/10.12720/jcm.13.9.498-504
[29] Jan, F., Rahman, A., Busaleh, R., Alwarthan, H., Aljaser, S., Al-Towailib, S., Alshammari, S., Alhindi, K.R., Almogbil, A., Bubshait, D.A., Ahmed, M.I.B. (2023). Assessing acetabular index angle in infants: A deep learning-based novel approach. Journal of Imaging, 9(11): 242. https://doi.org/10.3390/jimaging9110242
[30] Khan, T.A., Fatima, A., Shahzad, T., Rahman, A., Alissa, K., Ghazal, T.M., Al-Sakhnini, M.M., Abbas, S., Khan, M.A., Ahmed, A. (2023). Secure IoMT for disease prediction empowered with transfer learning in healthcare 5.0, the concept and case study. IEEE Access, 11:39418-39430. https://doi.org/10.1109/ACCESS.2023.3266156
[31] Sajid, N.A., Rahman, A., Ahmad, M., Musleh, D., Ahmed, M.I.B., Alassaf, R., Chabani, S., Ahmed, M.S., Salam, A.A., AlKhulaifi, D. (2023). Single vs. multi-label: The issues, challenges and insights of contemporary classification schemes. Applied Sciences, 13(11): 6804. https://doi.org/10.3390/app13116804
[32] Sajid, N.A., Ahmad, M., Rahman, A., Zaman, G., Ahmed, M.S., Ibrahim, N., Ahmed, M.I.B., Krishnasamy, G., Alzaher, R., Alkharraa, M., AlKhulaifi, D., AlQahtani, M., Salam, A.A., Saraireh, L., Gollapalli, M., Ahmed, R. (2023). A novel metadata based multi-label document classification technique. Computer Systems Science and Engineering, 46(2): 2195-2214. https://doi.org/10.32604/csse.2023.033844
[33] Jamal, M., Zafar, N.A., Rahman, A., Musleh, D., Gollapalli, M.A., Chabani, S. (2022). Modeling and verification of aircraft takeoff through novel quantum nets. Computers, Materials & Continua, 72(2): 3331-3348. https://doi.org/10.32604/cmc.2022.025205
[34] Rahman, A., Ahmed, M., Zaman, G., Iqbal, T., Khan, M.A.A., Farooqui, M., Ahmed, M.I.B., Ahmed, M.S., Nabeel, M., Omar, A. (2022). Geo-spatial disease clustering for public health decision making. Informatica, 46(6): 21-32. https://doi.org/10.31449/inf.v46i6.3827
[35] Khan, M.B.S., Rahman, A., Nawaz, M.S., Ahmed, R., Khan, M.A., Mosavi, A. (2022). Intelligent breast cancer diagnostic system empowered by deep extreme gradient descent optimization. Mathematical Biosciences and Engineering, 19(8): 7978-8002. https://doi.org/10.3934/mbe.2022373
[36] Soomro, A.M., Fudzee, M.F.B.M., Hussain, M., Saim, H.M., Zaman, G., Rahman, A., AlUbaidan, H., Nabil, M. (2022). Comparative review of routing protocols in MANET for future research in disaster management. Journal of Communications, 17(9): 734-744. https://doi.org/10.12720/jcm.17.9.734-744
[37] Asif, R.N., Abbas, S., Khan, M.A., Rahman, A., Sultan, K., Mahmud, M., Mosavi, A. (2022). Development and validation of embedded device for electrocardiogram arrhythmia empowered with transfer learning. Computational Intelligence and Neuroscience, 2022: 5054641. https://doi.org/10.1155/2022/5054641
[38] Nasir, M.U., Zubair, M., Ghazal, T.M., Khan, M.F.,

Ahmad, M., Rahman, A., Hamadi, H.A., Khan, M.A., Mansoor, W. (2022). Kidney cancer prediction empowered with blockchain security using transfer learning. Sensors, 22: 7483. https://doi.org/10.3390/s22197483

[39] Nasir, M.U., Khan, S., Mehmood, S., Khan, M.A., Rahman, A., Hwang, S.O. (2022). IoMT-based osteosarcoma cancer detection in histopathology images using transfer learning empowered with blockchain, fog computing, and edge computing. Sensors, 22: 5444. https://doi.org/10.3390/s22145444

[40] Al Najrani, B., Alghamdi, A., Al Otaibi, M., Al Dawod, S., Rahman, A., Nabil, M. (2022). A novel approach to Wikipedia references classification. ICIC Express Letters, 13(12): 1321-1330. http://doi.org/10.24507/icicelb.13.12.1321