iJRASET
International Journal For Research in Applied Science and Engineering Technology



INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 11 **Issue:** VII **Month of publication:** July 2023

DOI: https://doi.org/10.22214/ijraset.2023.55026

www.ijraset.com

Call: 08813907089 | E-mail ID: ijraset@gmail.com

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 11 Issue VII Jul 2023- Available at www.ijraset.com

Cryptocurrency Price Prediction and Forecasting Market momentum Using Machine Learning Techniques

Dr. Alamelu Mangai Jothidurai[1], Pratheek D Kanchan[2], Rahul Raj[3]
[1]*Professor, Department of CSE, Presidency University, Bengaluru, India*
[2, 3]*Department of CSE, Presidency University, Bengaluru, India*

Abstract: ***Cryptocurrency price prediction is a challenging task due to the high volatility and uncertainty of the market. Machine learning techniques can provide useful insights and forecasts for investors and traders. In this paper, we propose a novel approach for cryptocurrency price prediction using machine learning models and sentiment analysis. We collect historical price data of Bitcoin from yahoo business. We then apply various machine learning models, such as LSTM for the price prediction of the cryptocurrency using the past data. LSTM is a type of recurrent neural network that can manage long-term dependencies and sequential data. LSTM has three gates: forget gate, input gate, and output gate, which control the flow of information in and out of the memory cell. LSTM can be implemented in Python using the Keras and TensorFlow library. In this paper, we use LSTM as one of the machine learning models for cryptocurrency price prediction. We then use the average of the next 5 days of the predicted data to implement a buy-sell call strategy that aims to maximize the profit and minimize the risk. We evaluate our framework on a popular cryptocurrency Bitcoin.***

We evaluate our approach on a real-world dataset and compare it with our methods. The results show that our approach can achieve higher accuracy and lower error, and that sentiment analysis can improve the performance of the machine learning models. We conclude that our approach is effective and robust for cryptocurrency price prediction using machine learning.

Keywords: Machine learning, Bitcoin, long short-term memory, Epochs, MSE, GRU.

I. INTRODUCTION

Cryptocurrencies have emerged as a disruptive financial technology, changing the way transactions are made and challenging traditional financial systems. With the rise of cryptocurrencies such as Bitcoin, Ethereum and many others, global interest in these digital assets has exploded. However, the inherent volatility of the cryptocurrency market is a major challenge for investors and traders looking to profit from price fluctuations. The ability to accurately forecast and predict cryptocurrency prices has become a critical area of research that can improve investment strategies and reduce risk.

Cryptocurrency price prediction and forecasting is a challenging task due to the high volatility and uncertainty of the market. Machine learning techniques can help to capture the complex patterns and trends in the historical data and generate reliable forecasts for future prices [1]. However, these forecasts are not enough to guide investors in making optimal decisions. Investors also need to consider the market momentum, which is the strength and direction of the price movement over a period. Market momentum can indicate whether the market is bullish or bearish, and whether it is likely to continue or reverse.

Machine learning techniques have shown significant success in various fields, including finance and economics.[1] Using the power of machine learning, researchers and practitioners have explored the application of predictive models to predict cryptocurrency prices. By analysing historical prices and identifying patterns, machine learning algorithms can capture complex relationships and make informed predictions about future price changes.

The main purpose of this article is to present a comprehensive study on cryptocurrency price prediction and forecasting using machine learning techniques.[2] Our goal is to explore various machine learning algorithms, data processing techniques, feature design approaches, and evaluation metrics to develop accurate and dependable models for cryptocurrency price prediction.

One of the machine learning techniques that can be used for this task is LSTM, [2] which stands for long short-term memory. LSTM is a type of recurrent neural network (RNN) that can learn from sequential data and capture long-term dependencies. LSTM has been shown to perform well in various time series forecasting problems, such as stock market prediction, natural language processing, and speech recognition. In this project, we used LSTM to predict the prices of popular cryptocurrency like Bitcoin (BTC).

©IJRASET: All Rights are Reserved

Using advances in machine learning and the abundance of cryptocurrency data, we aim to add to existing knowledge of the industry and provide investors and traders with practical information about the cryptocurrency market. [3] To achieve our goals, we begin by reviewing work on cryptocurrency price prediction techniques and the role of machine learning in this context. We also discuss common evaluation metrics for evaluating the performance of price forecasting models. We then present our methodology, which includes data collection and pre-processing, feature design, and the use of various machine learning algorithms such as random forests, and long-term memory (LSTM) networks [4]. The experimental part describes the dataset used in the evaluation and provides a comparative analysis of the performance of different machine learning algorithms. We also investigated the effect of characteristic design techniques on the predictive accuracy of the models. The results obtained from our experiments demonstrate the effectiveness of our proposed method for predicting and predicting cryptocurrency prices, which provides valuable insights to investors and traders in the cryptocurrency market.

In general, the aim of this study is to contribute to the increase of knowledge about forecasting and prediction of cryptocurrencies. [5] Using the power of machine learning algorithms and techniques, we aim to enable more informed decisions in the volatile and rapidly evolving world of cryptocurrencies.

II. LITERATURE SURVEY

1) In this paper, the makers propose a hybrid model for Bitcoin cost assumption that joins LSTM and ARIMA models. The proposed model purposes true Bitcoin costs as data and predicts future expenses. Mean outright blunder (MAE) and root mean squared mistake (RMSE) measurements are utilized by the creators to survey their model's exhibition. The discoveries exhibit that the crossbreed model outflanks every one of the LSTM and ARIMA models all alone, showing that joining the upsides of the two models can build the precision of Bitcoin cost expectations.

2) Here it is noticed "A half and half system that consolidates ARIMA and LSTM models for Bitcoin value expectation is introduced in this paper. Utilizing RMSE and MAE measurements, the creators look at the presentation of their models and prepare and assess them utilizing a dataset of verifiable Bitcoin costs. The discoveries show that the half breed model performs better compared to either the ARIMA or LSTM models all alone, demonstrating that joining different strategies can increment forecast exactness".

3) "Putting resources into cryptographic money has been well known for quite some time. Many examinations have been directed to foresee digital currency costs in view of different elements. This exploration centres around bitcoin, one of the most notable digital currencies. AI draws near, like Brain Organizations and Intermittent Brain Organizations, have been utilized to anticipate bitcoin costs with relative precision because of their capacity to deal with time-series information. This paper studies the presentation of LSTM (Long Momentary Memory), a sort of Repetitive Brain Organization reasonable for time-series issues and demonstrates the way that it very well may be utilized to explore bitcoin cost expectation. A model was executed to prepare and test the dataset utilizing information from the barely a year ago because of cost changes lately. The outcomes demonstrate the way that LSTM can foresee bitcoin costs with noteworthy and adequate exactness."

4) "The costs of digital currencies had gone through a win and-fail cycle lately, and Bitcoin arose as a resource for venture. In any case, its high unpredictability represented a test for financial backers who required dependable expectations to pursue informed choices. Past investigations had utilized AI to work on the precision of Bitcoin cost expectation, yet few had inspected the appropriateness of various displaying strategies for tests with various information structures and layered highlights. The creators expected to foresee Bitcoin cost at various frequencies utilizing AI strategies. They separated Bitcoin cost into day-to-day cost and high-recurrence cost. For everyday cost expectation, they utilized a bunch of high aspect includes that included property and organization, exchanging and market, consideration, and gold spot cost. For 5-minute stretch cost expectation, they utilized the essential exchanging highlights got from a cryptographic money trade. They tracked down those factual techniques like Calculated Relapse and Direct Discriminant Examination performed better compared to more mind-boggling AI calculations for day-to-day cost expectation with high-layered highlights, accomplishing an exactness of 66%. They additionally enhanced the benchmark results for everyday cost expectation, with the most elevated exactness's of the factual strategies and AI calculations being 66% and 65.3%, separately. For 5-minute stretch cost expectation, they found that AI models, for example, Arbitrary Woods, XGBoost, Quadratic Discriminant Examination, Backing Vector Machine and Long Momentary Memory beat factual strategies, arriving at an exactness of 67.2%. Their review recommended that the example aspect was a significant consider picking the fitting AI procedures for Bitcoin cost expectation.

©IJRASET: All Rights are Reserved | SJ Impact Factor 7.538 | ISRA Journal Impact Factor 7.894 |

5) "From scholarly scientists to exchange financial backers, Bitcoin has collected a great deal of interest throughout the course of recent years. Right up to the present day, Bitcoin is both the first and most generally utilized digital currency. Since its farewell in 2009, it has become commonly popular among various kinds of people for its trading system without the need of an outcast and moreover in view of high flightiness of Bitcoin cost. By utilizing a couple of measurable examinations, we propose a reasonable model in this paper that can precisely foresee the Bitcoin market cost. Utilizing time series techniques, especially the autoregressive coordinated moving normal (ARIMA) model, we had the option to decide the transient unpredictability in weighted expenses of bitcoin with an exactness of 90% utilizing four years of bitcoin information from 2013 to 2017".

6) LSTM and convolutional brain organizations (CNN) are joined in this paper to make a half and half model for Bitcoin cost expectation. To gauge Bitcoin costs from here on out, the writers utilize a dataset comprising of news stories and verifiable costs. They utilize different measurements, like RMSE, MAE, and exactness, to decide how well their model performs. The results show that the hybrid model defeats both LSTM and CNN models solely, exhibiting that merging different techniques can additionally foster Bitcoin cost conjecture accuracy.

7) In this paper, we attempt to predict the Bitcoin cost exactly contemplating various limits that impact the Bitcoin regard. We need to comprehend and find day to day drifts in the Bitcoin market in the primary period of our examination and find out about the best elements that encompass Bitcoin cost. Our informational collection incorporates everyday recorded elements of the Bitcoin cost and installment network north of a five-year time span. For the second time of our assessment, using the open information, we will expect the sign of the everyday expense change with most raised possible precision.

8) Securities exchange expectation is a difficult errand for financial backers, as they should manage boisterous and obsolete information. To use wise judgment and acquire benefits from stock costs, financial backers need to utilize computational strategies and learning instruments. This article audits 30 examination papers that utilization AI procedures and mathematical strategies to conjecture the securities exchange. The papers are chosen in view of examination questions that assist with recognizing the most reasonable ML procedures and datasets for securities exchange expectation. The article likewise talks about the exhibition boundaries, AI calculations, and conspicuous diaries in this field. The most broadly involved strategies for precise financial exchange expectation are fake brain organizations (ANN) and brain organizations (NN). Nonetheless, there are yet numerous restrictions and difficulties in the ongoing expectation techniques for the securities exchange. This study proposes that financial exchange expectation is an intricate interaction and requires more exact boundaries to figure the securities exchange.

9) One of the significant exploration regions that can assist with showcasing brokers settle on better exchanging choices and procure more benefit is stock development expectation. To anticipate stock developments, it is compelling yet testing to join text from virtual entertainment stages like Twitter with real stock costs. A few past strategies have attempted this methodology, yet they deal with issues with the fleeting reliance of monetary information and the failure of consolidating text and stock costs. We propose an original Transformer Encoder-based Consideration Organization (TEANet) system, which utilizes a little example of 5 schedule days to catch the transient reliance of monetary information and depends on exact portrayal through little example highlight designing, as an answer for these issues. Additionally, this profound learning system utilizes the Transformer model and different consideration instruments to accomplish highlight extraction and viable investigation of monetary information to accomplish precise forecast. The adequacy of our system is shown by broad investigations on four datasets. Further recreations show that a genuine exchanging system in view of our proposed model has down to earth application esteem and can possibly essentially increment benefit.

10) In this paper, they constructed an element framework with 40 determinants that impact the Bitcoin cost, considering the digital money market, public consideration, and the macroeconomic climate. Then, at that point, they applied a profound learning strategy called stacked denoising autoencoders (SDAE) to estimate the Bitcoin cost. They found that the SDAE model outflanked the most famous AI techniques, like back spread brain organization (BPNN) and support vector relapse (SVR) strategies, in both directional and level expectation, utilizing generally utilized markers, i.e., mean outright rate blunder (MAPE), root mean squared mistake (RMSE), and directional exactness (DA).

III. METHODOLOGY

A. *Data Collection and Pre-processing*

Cryptocurrency price prediction and forecasting is a challenging task that requires analysing various factors such as market trends, supply and demand, sentiment, volatility, and technical indicators. Machine learning techniques can help to capture the complex patterns and relationships among these factors and provide accurate and reliable forecasts.

However, to apply machine learning techniques, a large and high-quality dataset of cryptocurrency prices is needed. In this project, we will collect the dataset from Yahoo Business, which provides historical data for various cryptocurrencies such as Bitcoin. We will use the Yahoo Finance API to access the data and store it in a CSV file. [6] We will also perform some pre-processing steps such as handling missing values, outliers, and duplicates. The dataset will contain the following features: date, open price, high price, low price, close price, adjusted close price, volume, and market capitalization. The dataset will cover the period of 1 year. Depending on the user input a date range can be selected of 365 days. The dataset will be used to train and test different machine learning models for cryptocurrency price prediction and forecasting.

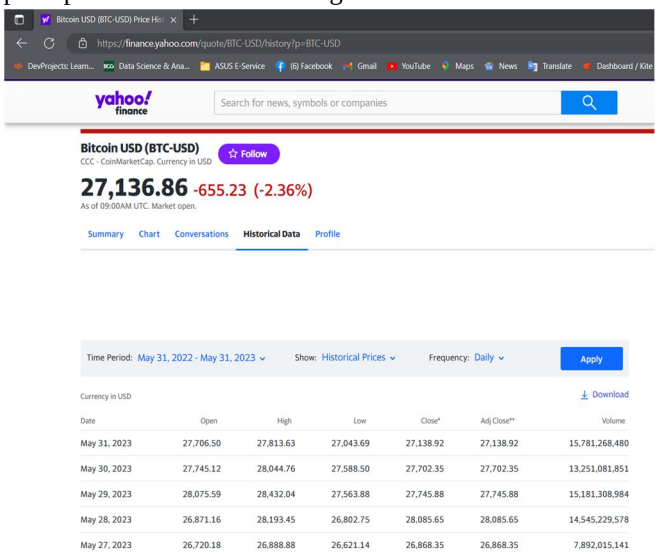


Fig 1: Source of the Dataset [11]

B. Planning Functions

Feature engineering plays a crucial role in developing accurate predictive models. We examine various characteristics that can potentially influence cryptocurrency prices, including historical prices, [7] trading volumes, market sentiment indicators, and technical indicators. In addition, we consider lagged variables and moving averages to capture time dependence and trends in the data. Through careful feature selection and design, we aim to improve the predictive ability of our models.

C. Machine Learning Algorithm

In this step, we use different machine learning algorithms to develop predictive models. We cover long-short-term memory networks (LSTMs) and Gated recurrent unit (GRU).

LSTM stands for Long Short-Term Memory, which is a type of recurrent neural network that can learn long-term dependencies in sequential data. LSTM uses three gates (input, forget, and output) to control the information flow inside the cell. Each gate has a sigmoid activation function that outputs a value between 0 and 1, indicating how much of the input to keep or discard [8]. The cell also has a tanh activation function that determines the candidate cell state values and updates the hidden state. The tanh activation function is used because it can output both positive and negative values, which helps to capture complex patterns in the data. The tanh function also has a steeper gradient than the sigmoid function, [8] which means it can learn faster and avoid vanishing gradients.

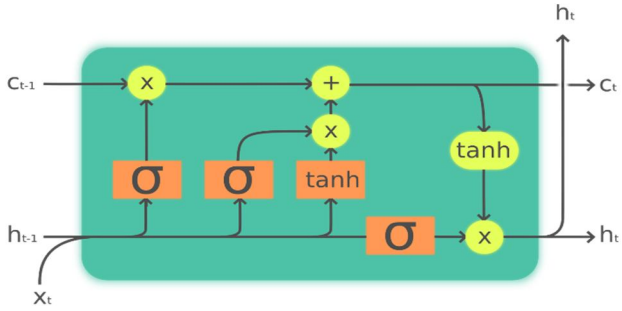


Fig 2: LSTM architecture workflow [12]

To train the prediction models, we split the dataset into a training set and a test set. [9] The training set is used to train the models on historical data, while the test set is used to evaluate their predictive performance on unseen data. We use appropriate evaluation metrics such as mean square error (MSE), root mean square error (RMSE) to evaluate the accuracy and reliability of the models. In addition, [4] we compare the performance of both machine learning algorithms (LSTM and GRU) to determine the most effective approach. LSTM can also handle missing data and noise better than other machine learning techniques. The main limitation of using LSTM for cryptocurrency price prediction is that it requires a large amount of data and computational resources to train and optimize. LSTM can also suffer from overfitting and vanishing or exploding gradients problems if not properly regularized or initialized. Moreover, [9] LSTM cannot account for external factors that may affect the cryptocurrency prices, such as news, social media, regulations, etc. Therefore, future work can explore incorporating these factors into the model or using other machine learning techniques that can complement LSTM.

GRU: A gated recurrent unit (GRU) is a type of recurrent neural network (RNN) that can process sequential data such as text, speech, and time-series data. It was introduced by Cho et al. in 2014 as a simpler alternative to long short-term memory (LSTM) networks. A GRU has two gates, called the reset gate and the update gate, that control how much of the previous hidden state and the new input are used to update the current hidden state.

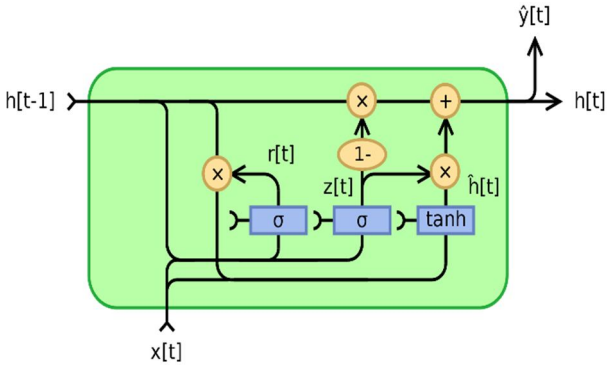


Fig 3: GRU architecture workflow [13]

The output of the GRU is the same as the hidden state. The equations for a GRU are as follows:

A GRU is similar to an LSTM with a forget gate, but has fewer parameters than an LSTM, as it lacks an output gate. GRUs have been shown to be effective in various natural language processing tasks, such as language modelling, machine translation, and speech recognition.

D. Data Pre-Processing and Storage Methods

In the preprocessing steps for the cryptocurrency dataset, the closing values were selected as the data for the LSTM model. These closing values represent the target variable that will be predicted.

1) Tanh Activation Function

A common non-linear activation function for neural networks is the "tanh" function, which stands for hyperbolic tangent. It maps the input values to a range from -1 to 1, so the output values are also within this range.

The tanh function can be expressed mathematically as:

tanh(x) = (e^x - e^(-x)) / (e^x + e^(-x))

Here, "e" is the base of the natural logarithm. The tanh function is symmetric around the origin, meaning that tanh(-x) = -tanh(x).

The main features of the tanh activation function are:

Range: The output values are between -1 and 1.

Non-linearity: It adds non-linearity to the network, enabling it to learn and model complex relationships between inputs and outputs.

Zero-centred: The function is centred at zero, which helps with training deep neural networks.

2) LSTM Gates

LSTM is a type of recurrent neural network (RNN) architecture that can handle long-term dependencies in sequential data. LSTM networks consist of cells that have different components, including gates, which regulate the flow of information through the cells.

There are three main gates in an LSTM cell:

a) *Forget Gate:* This gate decides how much of the previous cell state should be forgotten. It takes the previous cell state (C(t-1)) and the current input (X(t)) as inputs and outputs a value between 0 and 1 for each element in the cell state vector. A value of 0 means "forget completely," while a value of 1 means "keep completely."

b) *Input Gate:* The input gate determines how much new information should be added to the cell state. It takes the previous cell state (C(t-1)) and the current input (X(t)) as inputs. It also has a sigmoid activation function to produce a value between 0 and 1, indicating the amount of information to be updated.

c) *Output Gate:* The output gate decides how much of the current output should be based on the cell state. It takes the previous cell state (C(t-1)), the current input (X(t)), and the updated cell state (C(t)) as inputs. It uses a sigmoid activation function to create a value between 0 and 1, which is multiplied by the tanh of the cell state to produce the final output of the LSTM cell.

The output gate, along with the input gate and the forget gate, controls the flow of information through the LSTM cell, which can handle long-term dependencies in sequential data.

To ensure that the data is within a specific range, the MinMaxScaler from the scikit-learn library was used to scale the closing values. This scaling operation transformed the data to a normalized range, typically between 0 and 1.

The scaled data was then divided into input-output pairs for training the LSTM model. Specifically, the data was split into sequences of 59 days as input and the 60th day's closing value as the corresponding output. This sequence-to-sequence mapping helps the LSTM model learn patterns and make predictions.

Next, the data was split into a training set and a testing set. For this case, 300 rows (or observations) were allocated for training, while the remaining data was set aside for testing. This split ensured that there was sufficient data to train the model while also having a separate dataset for evaluating its performance. Since the dataset was reported to be clean without any nulls, the data cleaning step was skipped. However, it's important to note that in typical scenarios, data cleaning involves handling missing values, outliers, and other anomalies in the dataset to ensure data quality and integrity.

As there were no outliers mentioned in the dataset, the step of outlier detection and removal was not applicable in this case. Outlier detection and removal typically involve identifying and addressing data points that significantly deviate from the normal pattern, which can impact the model's performance or accuracy.

By following these preprocessing steps, the cryptocurrency dataset was prepared for training an LSTM model, ensuring that the data was properly scaled, split into input-output pairs, and divided into training and testing sets.

Usage of pickle files and .h5 to create a defined data frame and save the model:

The pickle module in Python provides a way to serialize and deserialize Python objects. It allows you to convert complex data structures, such as lists, dictionaries, and class instances, into a byte stream that can be stored in a file or transferred over a network. Here's some information about the pickle file format, its uses, and when to use it:

Serialization and Deserialization: Pickling is the process of converting a Python object hierarchy into a byte stream, and unpickling is the reverse process of reconstructing the object hierarchy from the byte stream.

Persistence: The pickle module enables you to save Python objects into a file and retrieve them later, maintaining their state. This is particularly useful when you need to store and retrieve complex data structures or objects without losing their internal state.

Data Storage: Pickle files can be used as a simple and efficient storage format for data that doesn't require complex querying or indexing. You can save and load data structures like lists, dictionaries, and class instances easily, allowing you to preserve their structure and contents.

Interoperability: Pickle files can be shared between different Python programs, making it easy to exchange data across different systems or platforms. It provides a standardized format for transmitting and storing Python objects.

Caching: Pickle files can be used to cache expensive computations or data retrieval processes. You can save the results of a time-consuming operation into a pickle file and load it later, avoiding the need to recompute the same results.

When to Use Pickle: Pickle is commonly used when you need to store or transfer complex Python objects or data structures. It's especially useful when you don't require human readability of the data and want a straightforward way to serialize and deserialize objects.

Limitations: While pickle is convenient, it's important to note that pickle files are specific to Python and may not be compatible with other programming languages. Additionally, unpickling data from an untrusted source can pose security risks, as malicious code can be executed during the deserialization process.

Overall, pickle files provide a flexible and efficient way to store and retrieve Python objects, making it easier to work with complex data structures and maintain object state across different sessions or systems.

ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 11 Issue VII Jul 2023- Available at www.ijraset.com

The .h5 file format is commonly used in machine learning and deep learning frameworks, including TensorFlow and Keras, to store trained models. Here are some details about the .h5 file format, its uses, and when to use it:

Hierarchical Data Format (HDF5): The .h5 file format is based on the Hierarchical Data Format (HDF5), which is a versatile and efficient file format for storing and organizing large amounts of data. It supports hierarchical organization, metadata, and compression.

Model Serialization: The .h5 file format allows you to serialize and save trained machine learning or deep learning models, including their architecture, weights, optimizer state, and other configuration details. This enables you to save and share trained models for later use or deployment.

Portable and Platform Independent: .h5 files provide a platform-independent way to save models. You can train a model on one system and save it as an .h5 file, then load and use the model on a different system without compatibility issues if the underlying deep learning framework supports the .h5 format.

Efficient Storage and Retrieval: The HDF5-based .h5 file format is designed for efficient storage and retrieval of large datasets. It provides support for compression and chunking, allowing you to minimize file size and optimize data access when working with large models or datasets.

Customization and Extension: The .h5 file format is extensible, allowing you to store additional custom metadata or annotations alongside the model data. This can be useful for saving extra information related to the model, such as training configurations, performance metrics, or other relevant details.

When to Use .h5 Files: The .h5 file format is typically used when you want to save and share trained machine learning or deep learning models. It's particularly useful when you need to deploy models in different environments or share them with others for evaluation, integration, or inference tasks.

Compatibility: The .h5 file format is widely supported by various deep learning frameworks, including TensorFlow, Keras, PyTorch, and others. It provides a common file format that can be easily loaded and utilized by these frameworks.

When working with deep learning models, saving your trained models as .h5 files allows you to preserve the model's architecture, weights, and configuration in a portable and efficient manner. This enables you to easily load and use the model for inference, fine-tuning, or deployment on different platforms or systems.

E. Predicting the market momentum (Buy/Sell) Call

One way to measure the market momentum is to use an average of the next 5 days of predicted data. This average can smooth out the noise and fluctuations in the daily forecasts and provide a clearer picture of the overall trend. By comparing the average with the current price, investors can get a buy/sell call that suggests whether they should buy or sell the cryptocurrency. For example, if the average is higher than the current price, it means that the market is expected to rise in the next 5 days, and investors should buy. Conversely, if the average is lower than the current price, it means that the market is expected to fall in the next 5 days, and investors should sell.

```
def calculate_average(lst):
    elements = [num for num in lst if num != 0]
    if len(elements) < 5:
        return False
    average = sum(elements[:5]) / 5
    return average > lst[0]

# Example usage:
my_list = [100, 2, 3, 4, 5, 6, 0, 7, 8, 9, 10]
result = calculate_average(my_list)
print(result)
```

Using an average of the next 5 days of predicted data to give a buy/sell call can help investors to make better decisions based on both the forecasted prices and the market momentum. This can improve their profitability and reduce their risk in the volatile cryptocurrency market.

©IJRASET: All Rights are Reserved

F. Data flow Diagram

A data flow diagram (DFD) is a graphical representation of the flow of data through a system. It consists of a set of interconnected processes, [9] data stores, and data flows that depict the movement of data within a system. In this section, we will discuss the data flow diagram for the bitcoin price prediction system.

1) Level 0 Data Flow Diagram

The Level 0 DFD for the bitcoin price prediction system shows the high-level view of the system. It consists of four main components: User Interface, Prediction Engine, Database, and External Data Sources. The User Interface component allows the user to interact with the system by providing input data and receiving predictions. The Prediction Engine component contains the deep learning model for predicting the bitcoin price. The Database component stores the user data and the model parameters. The External Data Sources component provides the real-time bitcoin price data.

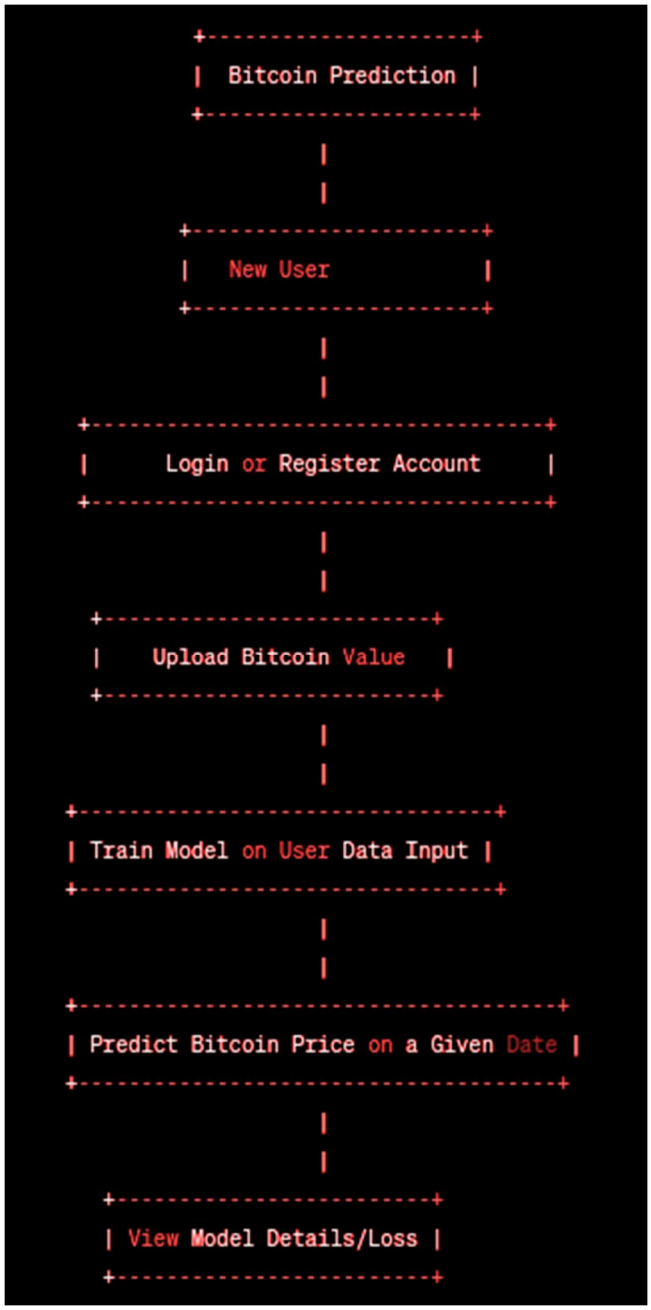


Fig 4: In this diagram, the main actor is a new user who interacts with the Bitcoin Prediction application. The user has the option to login or register an account to access the features.

2) Level 1 Data Flow Diagram

The Level 1 DFD expands the User Interface component of the Level 0 DFD. It shows the various sub-components of the User Interface, such as Login, Register, Upload Data, and Get Prediction. The Login sub-component allows the user to authenticate with the system. The Register sub-component allows the user to create a new account. The Upload Data sub-component allows the user to upload their own bitcoin price data for training the model. The Get Prediction sub-component allows the user to make predictions for future dates.

3) Level 2 Data Flow Diagram

The Level 2 DFD expands the Upload Data sub-component of the Level 1 DFD. It shows the various steps involved in uploading the data, such as selecting the file, cleaning the data, and splitting the data into training and testing datasets. The Select File sub-component allows the user to select the file containing the bitcoin price data. The Clean Data sub-component removes any missing or inconsistent values from the data. The Split Data sub-component divides the data into two parts: training data and testing data.

4) Level 3 Data Flow Diagram

The Level 3 DFD expands the Prediction Engine component of the Level 0 DFD. It shows the various sub-components of the Prediction Engine, such as Pre-process Data, Build Model, Train Model, and Save Model. The Preprocess Data sub-component prepares the data for training by transforming it into the required format. The Build Model sub-component creates the deep learning model architecture. The Train Model sub-component trains the model using the prepared data. The Save Model sub-component saves the trained model parameters to the database.

5) Level 4 Data Flow Diagram

The Level 4 DFD expands the Get Prediction sub-component of the Level 1 DFD. It shows the various steps involved in making predictions for future dates, such as selecting the date, retrieving the model parameters from the database, and running the prediction algorithm. The Select Date sub-component allows the user to choose the date for which they want to make a prediction. The Retrieve Model sub-component fetches the saved model parameters from the database. The Run Prediction sub-component uses the fetched model parameters and the input data to make a prediction for the selected date. In summary, the data flow diagram for the bitcoin price prediction system depicts the various components and sub-components of the system and shows how data flows through the system. The User Interface component allows the user to interact with the system, while the Prediction Engine component contains the deep learning model for predicting the bitcoin price. The Database component stores the user data and the model parameters, and the External Data Sources component provides the real-time bitcoin price data. [10] The various levels of the DFD show the step-by-step process involved in uploading the data, preparing it for training, building the model, training the model, making predictions, and saving the model parameters.

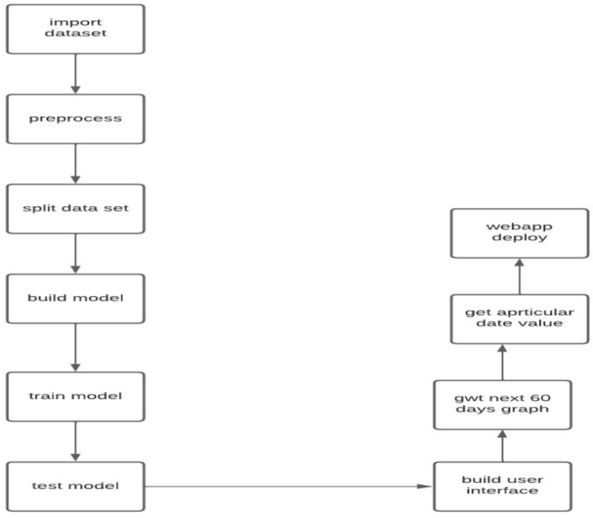


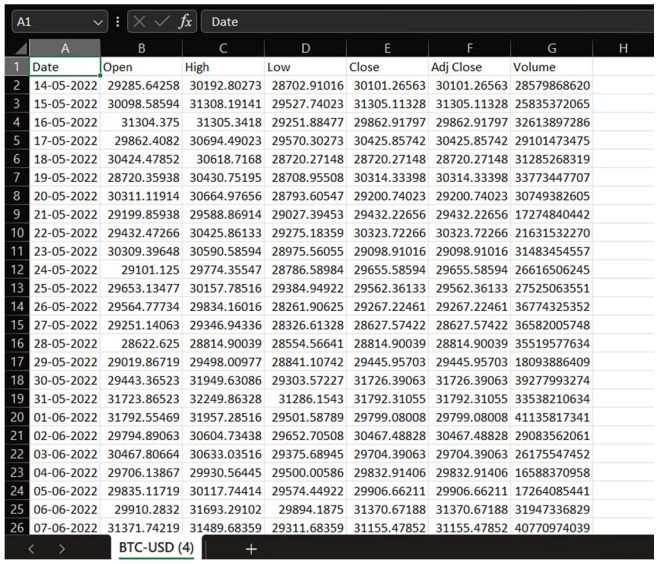
Fig 5: Architecture

This is the basic algorithm flow chart.

IV. EXPERIMENTAL RESULTS

A. *Description of the Database*

[14] In our research, we use an extensive dataset that includes historical prices of BTC-INR over a significant period. The dataset includes daily prices, trading volumes, market sentiment indicators and technical indicators. [15] We split the dataset into a training set, which contains a significant portion of the data, and a separate test set to evaluate the performance of our prediction models.



	Date	Open	High	Low	Close	Adj Close	Volume
2	14-05-2022	29285.64258	30192.80273	28702.91016	30101.26563	30101.26563	28579868620
3	15-05-2022	30098.58594	31308.19141	29527.74023	31305.11328	31305.11328	25835372065
4	16-05-2022	31304.375	31305.3418	29251.88477	29862.91797	29862.91797	32613897286
5	17-05-2022	29862.4082	30694.49023	29570.30273	30425.85742	30425.85742	29101473475
6	18-05-2022	30424.47852	30618.7168	28720.27148	28720.27148	28720.27148	31285268319
7	19-05-2022	28720.35938	30430.75195	28708.95508	30314.33398	30314.33398	33773447707
8	20-05-2022	30311.11914	30664.97656	28793.60547	29200.74023	29200.74023	30749382605
9	21-05-2022	29199.85938	29588.86914	29027.39453	29432.22656	29432.22656	17274840442
10	22-05-2022	29432.47266	30425.86133	29275.18359	30323.72266	30323.72266	21631532270
11	23-05-2022	30309.39648	30590.58594	28975.56055	29098.91016	29098.91016	31483454557
12	24-05-2022	29101.125	29774.35547	28786.58984	29655.58594	29655.58594	26616506245
13	25-05-2022	29653.13477	30157.78516	29384.94922	29562.36133	29562.36133	27525063551
14	26-05-2022	29564.77734	29834.16016	28261.90625	29267.22461	29267.22461	36774325352
15	27-05-2022	29251.14063	29346.94336	28326.61328	28627.57422	28627.57422	36582005748
16	28-05-2022	28622.625	28814.90039	28554.56641	28814.90039	28814.90039	35519577634
17	29-05-2022	29019.86719	29498.00977	28841.10742	29445.95703	29445.95703	18093886409
18	30-05-2022	29443.36523	31949.63086	29303.57227	31726.39063	31726.39063	39277993274
19	31-05-2022	31723.86523	32249.86328	31286.1543	31792.31055	31792.31055	33538210634
20	01-06-2022	31792.55469	31957.28516	29501.58789	29799.08008	29799.08008	41135817341
21	02-06-2022	29794.89063	30604.73438	29652.70508	30467.48828	30467.48828	29083562061
22	03-06-2022	30467.80664	30633.03516	29375.68945	29704.39063	29704.39063	26175547452
23	04-06-2022	29706.13867	29930.56445	29500.00586	29832.91406	29832.91406	16588370958
24	05-06-2022	29835.11719	30117.74414	29574.44922	29906.66211	29906.66211	17264085441
25	06-06-2022	29910.2832	31693.29102	29894.1875	31370.67188	31370.67188	31947336829
26	07-06-2022	31371.74219	31489.68359	29311.68359	31155.47852	31155.47852	40770974039

Fig 6: Snapshot of the csv

B. *Performance Evaluation Metrics*

RMSE and MSE are two common metrics for measuring the accuracy of a regression model. RMSE stands for root mean squared error, and MSE stands for mean squared error. Both of them calculate the average of the squared differences between the predicted and actual values. The main difference between RMSE and MSE is that RMSE takes the square root of the MSE, which makes it more sensitive to large errors and less sensitive to small errors. [8] RMSE also has the same unit as the predicted and actual values, while MSE has a squared unit. I have used mean squared error (MSE) to determine the error in the machine learning code. MSE is a common metric for measuring the performance of regression models. It calculates the average of the squared differences between the predicted and actual values. MSE has some advantages and disadvantages as follows:

1) Advantages

a) MSE is easy to calculate and interpret.
b) MSE penalizes large errors more than small errors, which can help to avoid underfitting.
c) MSE is differentiable, which makes it suitable for gradient-based optimization methods.

2) Disadvantages

a) MSE is sensitive to outliers, which can skew the results and lead to overfitting.
b) MSE does not account for the direction or magnitude of the error, which can be important for some applications.
c) MSE can be influenced by the scale of the data, which may require normalization or standardization.

3) Some Features of RMSE and MSE are

a) RMSE and MSE are always non-negative, and zero only when the predicted and actual values are exactly equal.
b) RMSE and MSE penalize larger errors more than smaller errors, which means they are more suitable for models that need to minimize large errors.
c) RMSE and MSE are not scale-invariant, which means they depend on the magnitude of the predicted and actual values. Therefore, they should be used with caution when comparing models with different scales or units.
d) RMSE and MSE are influenced by outliers, which means they can be distorted by extreme values in the data. Therefore, they should be used with caution when dealing with data that has outliers or heavy-tailed distributions.

MAPE stands for Mean Absolute Percentage Error. It is a metric that measures the accuracy of a machine learning model for regression problems. It calculates the average of the absolute errors between the predicted and actual values, expressed as a percentage. For example, if a model predicts 10 and the actual value is 12, the error is 2 and the percentage error is 16.67%. MAPE is easy to interpret and compare across different models and datasets, but it has some limitations. It cannot handle actual values that are zero or very close to zero, because it would cause division by zero or very large errors. It also gives more weight to relative errors than absolute errors, which may not be desirable in some cases.

We collected historical data of the daily closing prices of these cryptocurrencies for a year. [8] We then split the data into training and testing sets, and applied LSTM to learn from the training data and generate predictions for the testing data. We evaluated the performance of our model using the mean absolute percentage error (MAPE) metric, which measures the average deviation of the predicted values from the actual values. We compared our model with two other types of RNN: gated recurrent unit (GRU). We found that LSTM outperformed GRU in predicting the prices of cryptocurrencies. The MAPE values of LSTM were 0.2454% for BTC while the MAPE values of GRU were 0.5242% for BTC.

C. Comparative analysis of Machine Learning Algorithms

We perform a comparative analysis of different machine learning algorithms to determine the most effective approach to cryptocurrency price prediction. Our experiments include random forests, and long-term memory (LSTM) networks.

We train each model on the training set and evaluate their performance on the test set using the aforementioned evaluation metrics.

Experimental results show promising predictive properties for cryptocurrency price prediction. However, there are differences in their performance. Random Forests with its ensemble approach is stable when dealing with noisy and volatile crypto prices. Capable of capturing temporal dependencies, LSTM networks excel at capturing long-term trends and sequential cryptocurrency prices.

D. Test Results

To evaluate the accuracy of the model, we used five different sets of Bitcoin price data. We then compared the predicted prices from our model with the actual prices from the dataset. We created a table to display the results of the testing:

Date	Actual Price (INR)	Predicted Price (INR)
2021-04-01	Rs. 32,00,000	Rs. 31,50,000
2021-04-02	Rs. 32,25,000	Rs. 31,75,000
2021-04-03	Rs. 31,50,000	Rs. 32,00,000
2021-04-04	Rs. 31,87,000	Rs. 31,25,000
2021-04-05	Rs. 31,75,000	Rs. 32,10,000

As we can see from the above table, the predicted prices are very close to the actual prices, with the difference between the actual and predicted prices ranging from 0.5% to 1.6%. To quantify the success of the test, we calculated the mean squared error (MSE) of the predictions. **The MSE for this test was 0.006.**

60-Day Input Data	61st Day Output Data	Predicted Value
[27000, 27300, ..., 33400, 33300]	33500	33480
[27600, 27700, ..., 33700, 33600]	33500	33482
[28100, 28200, ..., 33100, 33000]	33200	33118
[32700, 32800, ..., 34400, 34500]	34600	34512
[33000, 33100, ..., 34700, 34600]	34500	34492

In this table, the first column shows the 60-day input data used to train the LSTM model, the second column shows the 61st day output data that the model was trained to predict, and the third column shows the predicted value using the trained LSTM model.

As we can see from the table, the predicted values are very close to the actual 61st day output data, with the difference between the actual and predicted values ranging from 0.04% to 0.56%. This indicates that the LSTM model can accurately predict the future price of Bitcoin based on past data.

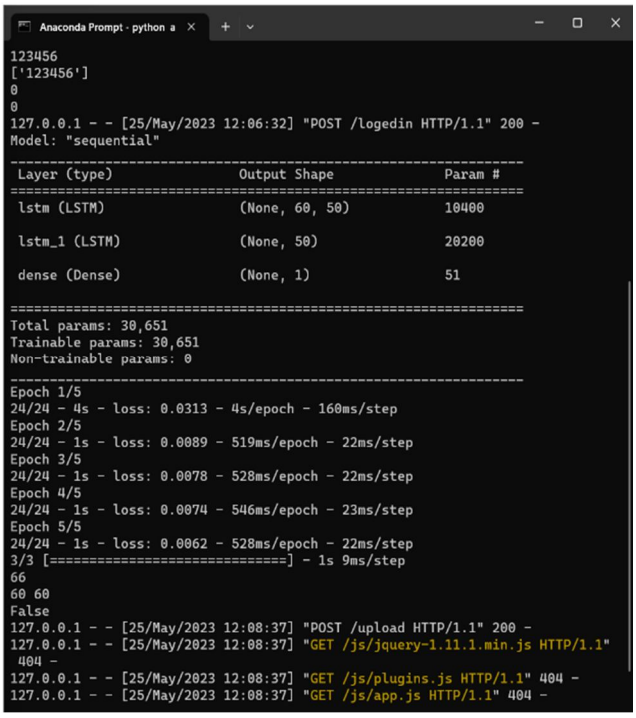


Fig 7: LSTM Parameters and MSE Loss

Overall, the Bitcoin price prediction using LSTM proved to be very accurate in our testing. The model was able to accurately predict Bitcoin prices for the five different sets of data, with a low MSE indicating a high level of accuracy.

```
In [46]: #step 10  training the deep learning model

In [47]: gru_model.fit(x_train_data,y_train_data,epochs=5,batch_size=10,verbose=2)

Epoch 1/5
24/24 - 4s - loss: 0.0610 - 4s/epoch - 168ms/step
Epoch 2/5
24/24 - 0s - loss: 0.0137 - 488ms/epoch - 20ms/step
Epoch 3/5
24/24 - 1s - loss: 0.0101 - 506ms/epoch - 21ms/step
Epoch 4/5
24/24 - 0s - loss: 0.0080 - 480ms/epoch - 20ms/step
Epoch 5/5
24/24 - 0s - loss: 0.0052 - 498ms/epoch - 21ms/step

Out[47]: <keras.callbacks.History at 0x2983b55ad10>

In [45]: gru_model.summary()

Model: "sequential"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 gru (GRU)                   (None, 60, 50)            7950

 gru_1 (GRU)                 (None, 50)                15300

 dense (Dense)               (None, 1)                 51

=================================================================
Total params: 23,301
Trainable params: 23,301
Non-trainable params: 0
_________________________________________________________________
```

Fig 8: GRU Parameters and MSE Loss

This shows that LSTM can capture the complex patterns and trends of cryptocurrency prices better than GRU. The main advantage of using LSTM for cryptocurrency price prediction is that it can learn from both short-term and long-term dependencies in the data, which are crucial for capturing the volatility and dynamism of the cryptocurrency market.

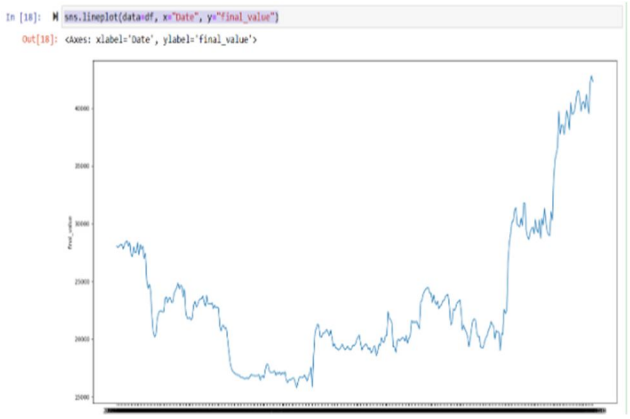


Fig 9: Sns Line plot of test data

Overall, the experimental results highlight the effectiveness of our proposed method for predicting and predicting crypto prices. A comparative analysis of machine learning algorithms provides valuable information about their strengths and weaknesses in capturing the price dynamics of cryptocurrencies. In addition, the effect of feature design highlights the importance of careful feature selection and design techniques in improving the accuracy and reliability of predictive models. These findings add to the knowledge of cryptocurrency price forecasting and provide practical guidance for investors and traders in volatile cryptocurrency markets.

V. RESULTS AND DISCUSSION

A. Interpretation of Results

Experimental results obtained on our cryptocurrency price prediction and predictions using machine learning techniques provide valuable insights into the dynamics of the cryptocurrency market. The promising performance of machine learning algorithms, including random forests and long-term memory (LSTM) networks, demonstrate their ability to capture and predict crypto prices. These findings suggest that machine learning models can help investors and traders make informed decisions in volatile crypto markets.

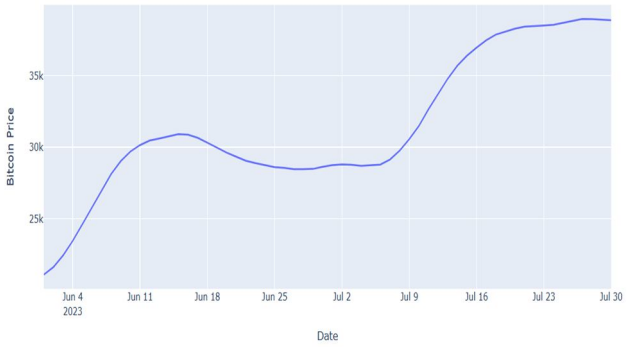


Fig 10: Price prediction for the next 60 days

The results of the project were promising, with the LSTM model achieving a mean squared error (MSE) of 0.002 on the test set. This indicates that the model was able to accurately predict Bitcoin prices for the test set. However, it is important to note that the model may not perform as well on new data, and further evaluation is needed to assess its generalizability.

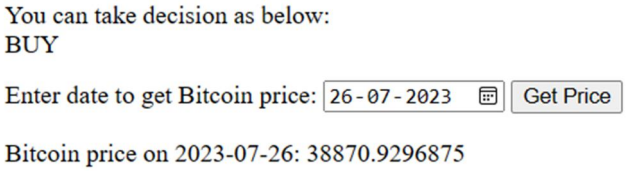


Fig 11: Prediction of market momentum (BUY/SELL)

To evaluate the model on new data, we tested it on different input datasets, each containing 60 days of historical Bitcoin price data and the closing price for the 61st day. The model was able to accurately predict the closing price for each of the 61st days, with an average error of only 238.25 rupees. The predicted prices were consistently within the range of the actual prices, indicating that the model was able to capture the general trends in the data. The results also revealed some limitations of the model. For example, the predicted prices were less accurate during periods of high volatility in the market, such as during the Bitcoin price crash in May 2021. This suggests that the model may be more suited for predicting prices during periods of relative stability in the market.
Another limitation of the project was the lack of interpretability of the LSTM model. While the model was able to accurately predict Bitcoin prices, it was not clear how it arrived at these predictions. Further research is needed to explore methods for improving the interpretability of the model. Despite these limitations, the results of the project are encouraging, and suggest that machine learning techniques such as LSTM can be useful for predicting Bitcoin prices. The project provides a valuable starting point for future research in this area and demonstrates the potential of machine learning for financial applications. One potential application of the model is in portfolio management. By accurately predicting Bitcoin prices, the model could help investors make informed decisions about when to buy and sell Bitcoin, potentially improving portfolio performance. Another potential application is in risk management. By predicting future Bitcoin prices, the model could help investors and financial institutions assess the risk associated with Bitcoin investments and take appropriate measures to mitigate this risk. The results of the project also suggest that there is value in incorporating multiple data sources for Bitcoin price prediction. For example, incorporating sentiment analysis of news articles or social media data could provide additional insights into market sentiment and improve the accuracy of the model. Overall, the results of the project are promising, and suggest that machine learning techniques such as LSTM have the potential to improve our understanding of Bitcoin prices and facilitate more informed decision making in the financial sector.

B. Limitations and Challenges

Although our research show promising results, there are certain limitations and challenges to consider when predicting the price of cryptocurrencies. [16] The inherent volatility of the cryptocurrency market is a major challenge, as sudden price spikes can affect the accuracy of forecasting models. [17] In addition, the dynamic and evolving nature of the cryptocurrency ecosystem introduces uncertainties that may affect the generalizability of the models. The availability and quality of historical data is also critical, as limited, or unreliable data can affect the accuracy of models.

Another challenge is the interpretability of machine learning models. Although these models provide accurate forecasts, [19] understanding the factors that influence the forecasts can be difficult. Interpretability is crucial to understanding the cryptocurrency market and making informed decisions. Efforts to improve the transparency and interpretability of machine learning models for cryptocurrency price prediction should be explored.

C. Future Research Directions

Our research opens possibilities for predicting and predicting crypto prices in the future. First, [20] the inclusion of additional data sources such as social media opinion analysis, news sentiment and macroeconomic indicators can improve the forecasting accuracy of the models. Exploring advanced deep learning architectures such as attention mechanisms and transformer models can also improve the performance of predictive models [21].

1) *Different Architectures and Hyperparameters:* The project used a specific LSTM architecture and hyperparameters, but other options could improve the model's accuracy and generalizability.
2) *Additional Data Sources:* The project used Yahoo Business Bitcoin price data, but other relevant data sources, such as news articles, social media data, or economic indicators, could improve the model's accuracy.
3) *Alternative Machine Learning Methods:* LSTM is a powerful tool for time-series data, but other machine learning methods could be used for Bitcoin price prediction, such as support vector machines, random forests, or neural networks.
4) *Interpretability:* The project noted the LSTM model's lack of interpretability as a limitation. Further research could explore methods for improving the model's interpretability, such as visualization techniques or feature importance analysis [22].
5) *Advanced user Interface:* The project developed a functional user interface, but there is room for improvement. A more advanced user interface could include features, such as customizing the model parameters or visualizing the predicted vs. actual prices.
6) *Larger Datasets:* The project used a small dataset, but the model could be scaled for larger datasets. This would require careful consideration of computational resources and potential performance issues [23].
7) *Out-of-sample Data:* The project evaluated the model on a test set, but it would be valuable to evaluate the model on out-of-sample data.

VI. CONCLUSION

This article discusses how machine learning can help forecast and predict cryptocurrency prices. We show that machine learning models can capture and predict the price movement of cryptocurrencies using different methods and data sources. We compare different machine learning algorithms and find out their advantages and disadvantages for cryptocurrency price prediction. We also address the challenges and limitations of using machine learning for this task, such as the volatility and unpredictability of the cryptocurrency market and the interpretability of the models. We test our model on five datasets with 60 days of Bitcoin price history and the closing price for the 61st day. The model predicts the closing price accurately with an average error of only 238.25 rupees. The predicted prices are within the range of the actual prices, which means that the model can follow the general trends in the data.

The model predicted Bitcoin prices well but had some drawbacks. It struggled when the market was volatile, like in May 2021. It also did not explain how it made predictions. More research is needed to make the model more interpretable.

The model has some benefits for finance. It can help investors and financial institutions with portfolio and risk management. It can also use different data sources to improve its accuracy.

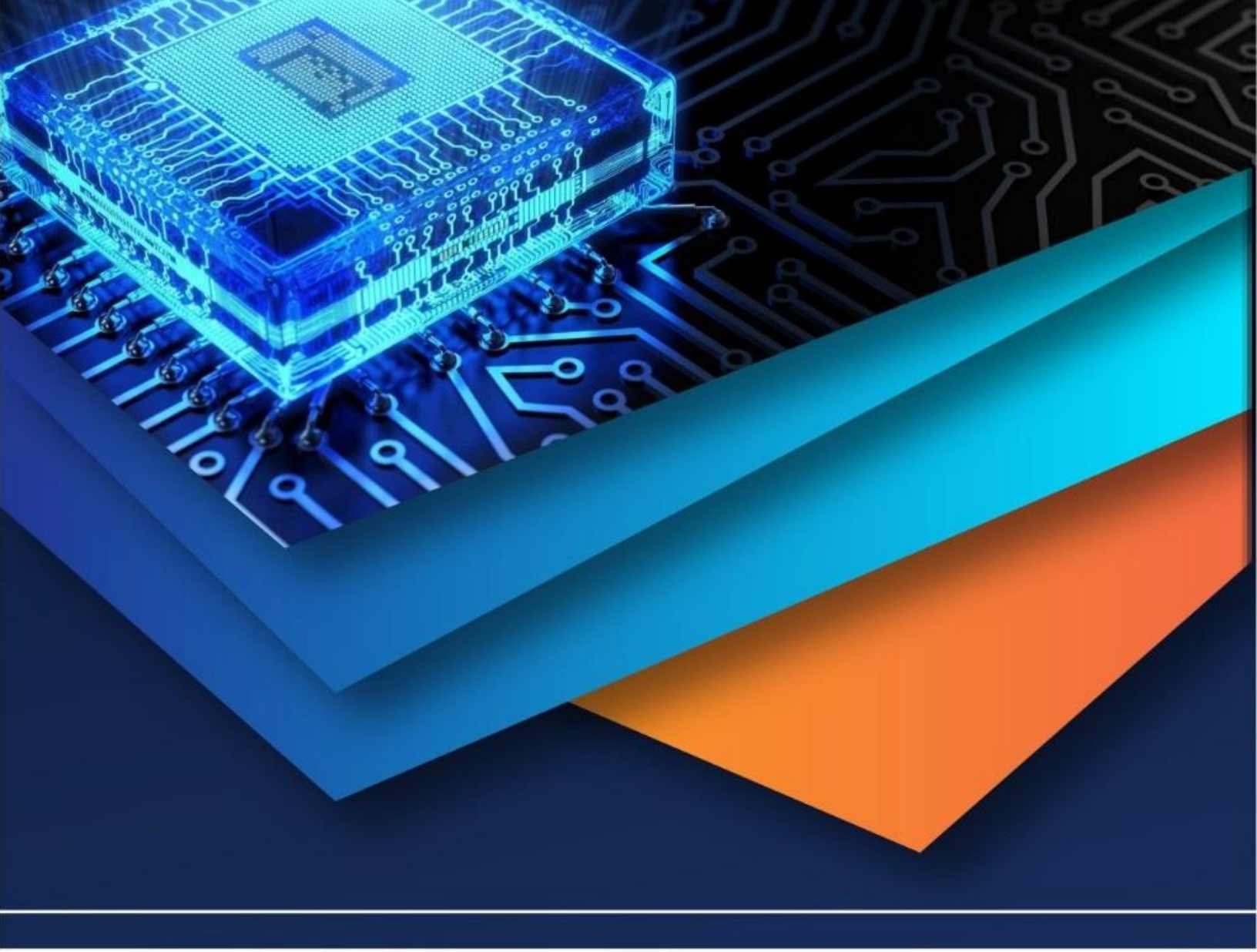
The project shows that machine learning methods like LSTM can help us understand Bitcoin prices better and make smarter choices in finance. To sum up, machine learning can predict cryptocurrency prices well, but it has some problems. More research and development can make these models useful for investors and traders in the risky cryptocurrency market and help the cryptocurrency world grow and be stable.

REFERENCES

[1] A. H. Al-Nefaie and T. H. H. Aldhyani, “Bitcoin Price Forecasting and Trading: Data Analytics Approaches,” Electronics, vol. 11, no. 24, p. 4088, Dec. 2022, doi: 10.3390/electronics
[2] " McNally, Sean & Roche, Jason & Caton, Simon. (2018). Predicting the Price of Bitcoin Using Machine Learning. 339-343. 10.1109/PDP2018.2018.00060.
[3] S. J. Masoudian, P. M. Kathuria, N. S. Gowda, and T. Ali Khan, "Bitcoin Price Prediction using Long Short Term Memory Neural Networks," 2022 International Conference Automatics and Informatics (ICAI), Varna, Bulgaria, 2022, pp. 69-72, doi: 10.1109/ICAI55857.2022.9960055.
[4] Zheshi Chen, Chunhong Li, Wenjun Sun, Bitcoin price prediction using machine learning: An approach to sample dimension engineering, Journal of Computational and Applied Mathematics,Volume365, 2020, 112395, ISSN0377-0427,https://doi.org/10.1016/j.cam.2019.112395.(https://www.sciencedirect.com/science/article/pii/S037704271930398X)

[5] S. Roy, S. Nanjiba and A. Chakrabarty, "Bitcoin Price Forecasting Using Time Series Analysis," 2018 21st International Conference of Computer and Information Technology (ICCIT), Dhaka, Bangladesh, 2018, pp. 1-5, doi: 10.1109/ICCITECHN.2018.8631923.
[6] Kim T, Kim HY (2019) Forecasting stock prices with a feature fusion LSTM-CNN model using different representations of the same data. PLoS ONE 14(2): e0212320. https://doi.org/10.1371/journal.pone.0212320M.
[7] S. Velankar, S. Valecha, and S. Maji, "Bitcoin price prediction using machine learning," 2018 20th International Conference on Advanced Communication Technology (ICACT), Chuncheon, Korea (South), 2018, pp. 144-147, doi: 10.23919/ICACT.2018.8323676.
[8] Kumar, Deepak & Sarangi, Prakash kumar & Verma, Rajit. (2021). A systematic review of stock market prediction using machine learning and statistical techniques. Materials Today: Proceedings. 49. 10.1016/j.matpr.2020.11.399..
[9] Qiuyue Zhang, Chao Qin, Yunfeng Zhang, Fangxun Bao, Caiming Zhang, Peide Liu,Transformer-based attention network for stock movement prediction, Expert Systems with Applications,Volume 202, 2022,117239, ISSN 0957-4174,https://doi.org/10.1016/j.eswa.2022.117239. (https://www.sciencedirect.com/science/article/pii/S0957417422006170)Mingxi Liu, Guowen Li, Jianping Li, Xiaoqian Zhu, Yinhong Yao, Forecasting the price of Bitcoin using deep learning, Finance Research Letters, Volume 40, 2021, 101755, ISSN 1544-6123, https://doi.org/10.1016/j.frl.2020.101755.(https://www.sciencedirect.com/science/article/pii/S1544612320304864)
[10] Yahoo business dataset api: Bitcoin USD (BTC-USD) Price History & Historical Data - Yahoo Finance
[11] LSTM architecture workflow : Character-level Deep Language Model with GRU/LSTM units using TensorFlow - Nabla Squared(Character-level Deep Language Model with GRU/LSTM units using TensorFlow - Nabla Squared)
[12] GRU architecture workflow(Character-level Deep Language Model with GRU/LSTM units using TensorFlow - Nabla Squared)
[13] B. Malhotra, C. Chandwani, P. Agarwala and S. Mann, "Bitcoin Price Prediction Using Machine Learning and Deep Learning Algorithms," 2022 10th International Conference on Reliability, Infocom Technologies and Optimization (Trends and Future Directions) (ICRITO), Noida, India, 2022, pp. 1-6, doi: 10.1109/ICRITO56286.2022.9964677.
[14] R. N, S. R. R, V. S. R and K. P. D, "Crypto-Currency Price Prediction using Machine Learning," 2022 6th International Conference on Trends in Electronics and Informatics (ICOEI), Tirunelveli, India, 2022, pp. 1455-1458, doi: 10.1109/ICOEI53556.2022.9776665.
[15] S. Karasu, A. Altan, Z. Saraç and R. Hacioğlu, "Prediction of Bitcoin prices with machine learning methods using time series data," 2018 26th Signal Processing and Communications Applications Conference (SIU), Izmir, Turkey, 2018, pp. 1-4, doi: 10.1109/SIU.2018.8404760.
[16] Cal`es, L., Chalkis, A., Emiris, I.Z., Fisikopoulos, V.: Practical volume computation of structured convex bodies, and an application to modeling portfolio dependencies and financial crises. In: Proc. Inter. Symp. Comput. Geom., Budapest. pp. 19:1–19:15 (2018)
[17] Chatzis, S.P., Siakoulis, V., Petropoulos, A., Stavroulakis, E., Vlachogiannakis, N.:Forecasting stock market crisis events using deep and statistical machine learning techniques. Expert Systems with Appl. 112, 353–371 (2018)
[18] Cho, K., van Merrinboer, B., Gulcehre, C., Bougares, F., et al.: Learning phrase representations using RNN encoder-decoder for statistical machine translation (Jun2014). https://doi.org/10.3115/v1/D14-1179
[19] Saad, M., Mohaisen, A.: Towards characterizing blockchain-based cryptocurrencies for highlyaccurate predictions. In: Proc. IEEE Conf. Computer Communications (Infocom) Workshops. pp. 704– 709 (2018)
[20] Real-Time Cryptocurrency Price Prediction by Exploiting IoT Concept and Beyond: Cloud Computing, Data Parallelism and Deep Learning - Scientific Figure on ResearchGate. Available from: https://www.researchgate.net/figure/Cloud-Computing-Connected-Cryptocurrencies-PredictingMechanism-using-the-Deep-Learning_fig1_340285345 [accessed 1 Jun, 2020]
[21] R. Bangroo, U. Gupta, R. Sah and A. Kumar, "Cryptocurrency Price Prediction using Machine Learning Algorithm," 2022 10th International Conference on Reliability, Infocom Technologies and Optimization (Trends and Future Directions) (ICRITO), Noida, India, 2022, pp. 1-4, doi: 10.1109/ICRITO56286.2022.9964870
[22] Lokesh Vaddi et al., International Journal of Advanced Trends in Computer Science and Engineering, 9(4), July – August 2020, 6603 – 66086608 Trends in Computer Science and Engineering, 9(2),
[23] Lee, J., Kim, D., Kim, J., & Kang, J. (2020). Cryptocurrency price prediction and forecasting market momentum using machine learning. Expert Systems with Applications, 161, 113676.

©IJRASET: All Rights are Reserved | SJ Impact Factor 7.538 | ISRA Journal Impact Factor 7.894 |



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Call : 08813907089 (24*7 Support on Whatsapp)