INTERNATIONAL JOURNAL OF PROGRESSIVE RESEARCH IN ENGINEERING MANAGEMENT AND SCIENCE (IJPREMS)

www.ijprems.com
editor@ijprems.com

Vol. 04, Issue 05, May 2024, pp: 242-249

e-ISSN : 2583-1062

Impact Factor: 5.725

CRYPTOCURRENCY PRICE PREDICTION MODEL USING LONG SHORT-TERM MEMORY

Dr. Khaleel Ur Rahman Khan[1], Nishitha Lakshmi Bandreddy[2], Kaithalapuram Vaishnavi[3], Bantu Soumith[4], Pambala Shiva Kumar[5]

[1]Dean (Academics), CSE Dept, ACE Engineering College, Hyderabad, India.

[2,3,45]Student, CSE Dept, ACE Engineering College, Hyderabad, India.

ABSTRACT

Cryptocurrencies have gained significant attention in recent years as a revolutionary digital asset class, but their price volatility poses challenges for investors and traders. Predicting cryptocurrency price movements is a complex task due to various factors influencing their value, including market sentiment, news, and technological developments. This abstract presents a Cryptocurrency Price Prediction Model utilizing Long Short-Term Memory (LSTM) networks, a type of recurrent neural network (RNN), to forecast cryptocurrency prices. The proposed model leverages historical price and volume data, as well as sentiment analysis of relevant news and social media posts, to make predictions. LSTM networks are particularly well-suited for this task because they can capture long-term dependencies in time series data. In summary, the Cryptocurrency Price Prediction Model presented in this abstract leverages LSTM networks, historical price data, and technical indicators to forecast cryptocurrency prices. By considering past price trends and relevant market data, the model aims to provide valuable insights and predictions to navigate the volatile cryptocurrency market. The efficacy of the model will be demonstrated through empirical results, and potential applications and limitations will be discussed.

1. INTRODUCTION

Essentially, this project will be able to Predict the Cryptocurrency price using Long short-term Memory. Recurrent neural networks (RNN) are the state-of-the-art algorithm for sequential data and are used by Apple's Siri and Google's voice search. It is an algorithm that remembers its input due to its internal memory, which makes the algorithm perfectly suited for solving machine learning problems involving sequential data. It is one of the algorithms that have great results in deep learning. In this article, it is discussed how to predict the price of Cryptocurrency by analyzing the information of the last 6 years. We implemented a simple model that helps us better understand how time series works using Python and RNNs. Digital currencies have become the favourable and most used for commercial money transactions all over the world. The rising usage is because of its innovative characteristics such as transparency thus increasing acceptance throughout the world. El Salvador became the first country to do this. Furthermore, Bitcoin is the leading cryptocurrency in the world with adoption growing consistently over time. First introduced in 2008, and deployed as open source in 2009 by Satoshi Nakamoto [1] whose identity is still unknown. Currently, the virtual currency market value is close to 1.4 trillion INR, but it varies from time to time. Digital currency especially bitcoin has been adopted by the people, and since then the digital currency market has been growing up. Bitcoin is a peer-to-peer cryptocurrency in which all transactions are not regulated or controlled by any third party. It has highly volatile market price working 24/7[2]. It operates on a decentralized, peer-to-peer and trustless system in which all transactions are posted to an open ledger called the Blockchain.

2. OBJECTIVES

In our project there are 4 objectives. They can be listed as:

- Forecasting Accuracy
- Risk Management
- Trading Strategy Development
- Real-Time Prediction

3. METHODOLOGY

Data is collected from various sources, ensuring accuracy and coverage of relevant factors. After preprocessing to handle missing values and outliers, features are selected and engineered to capture essential market dynamics. An LSTM architecture is designed, with careful consideration given to hyperparameters and model complexity. The model is trained using historical data, optimizing performance through techniques like backpropagation and gradient descent. Evaluation metrics are used to assess the model's accuracy and reliability on unseen data, guiding risk management and trading strategy implementation. Finally, the model is deployed for real-time prediction, with ongoing monitoring and refinement to maintain performance and adapt to changing market conditions.

4. LITERATURE SURVEY

TITLE: A Cryptocurrency Price Prediction Model using Deep Learning.

AUTHOR: V.Akila, M.V.S. Nithin, I. Prasanth, M. Sandeep.

YEAR: 2023

DESCRIPTION:

Cryptocurrencies have gained immense popularity in recent years as an emerging asset class, and their prices are known to be highly volatile. Predicting cryptocurrency prices is a difficult task due to their complex nature and the absence of a central authority. In this paper, our proposal is to employ Long Short-Term Memory (LSTM) networks, a type of deep learning technique to forecast the prices of cryptocurrencies.

DISADVANTAGES:

- The main disadvantage is lack of interpretability can hinder users' ability to gain insights into market dynamics and may reduce trust in the model's predictions.
- The model heavily relies on historical price data and technical indicators, which may not fully capture all relevant factors influencing cryptocurrency prices. This dependency could lead to suboptimal predictions, especially during periods of significant market changes or when new influential factors emerge.

TITLE: Bitcoin: A Peer-to-Peer Electronic Cash System.

AUTHOR: Satoshi Nakamoto

YEAR: 2009

DESCRIPTION:

A purely peer-to-peer version of electronic cash would allow online payments to be sent directly from one party to another without going through a financial institution. Digital signatures provide part of the solution, but the main benefits are lost if a trusted third party is still required to prevent double-spending. We propose a solution to the double-spending problem using a peer-to-peer network. The network timestamps transactions by hashing them into an ongoing chain of hash-based proof-of-work, forming a record that cannot be changed without redoing the proof-of-work

DISADVANTAGES:

- The described system relies on a proof-of-work mechanism, which requires significant computational power to validate transactions and secure the network. This results in high energy consumption, as miners compete to solve cryptographic puzzles to add new blocks to the blockchain. The environmental impact of this energy consumption, particularly in terms of carbon emissions, is a significant drawback of the system.
- While the described system offers decentralized transaction validation and security, it may face scalability challenges as the network grows.

TITLE: Stochastic Neural Network For Cryptocurrency Price Prediction

AUTHOR: Jay Patel, Vasu Kalariya, Pushpendra Parmar, Sudeep Tanwar

YEAR: 2020

DESCRIPTION:

Over the past few years, with the advent of blockchain technology, there has been a massive increase in the usage of Cryptocurrencies. However, Cryptocurrencies are not seen as an investment opportunity due to the market's erratic behavior and high price volatility. Most of the solutions reported in the literature for price forecasting of Cryptocurrencies may not be applicable for real-time price prediction due to their deterministic nature.

Motivated by the aforementioned issues, we propose a stochastic neural network model for Cryptocurrency price prediction

DISADVANTAGES:

- Stochastic neural networks can be complex and difficult to interpret. Understanding how the model arrives at its predictions can be challenging.
- The approach relies on the random walk theory, which assumes price movements are random with no predictable trends. This might not capture all factors influencing cryptocurrency prices.
- Introducing randomness can help capture volatility, but it might also increase the model's susceptibility to overfitting the training data. This could lead to poor performance on unseen data.

5. PROPOSED SYSYTEM

This paper reflects the CRISP technique of data mining. The CRISP-DM motivation for the traditional KDD [26] focuses on the company-level of the forecasting task. The data set is used by Bitcoin covers the period 19 August 2013 to 19 July 2016. Figure 1 displays a time series graph of this. Data is omitted from prior to August 2013 as they no longer represent the network correctly. Dataset is used in bitcoin Ethereum Historical Data and Bitcoin Historical Data. CSV files for bitcoin exchanges from Jan 2014 to July 2019, with by-the minute updates of OHLC (Open, High, Low, Close), Volume in BTC and currency, as well as weighted bitcoin price.

6. HARDWARE AND SOFTWARE REQUIREMENTS

6.1 HARDWARE REQUIREMENTS:

- Processor: Min. Core i3 processor
- RAM: 2GB (Min.) or 8GB (Recommended)
- Hard Disk Space: 50GB+

6.2 SOFTWARE REQUIREMENTS:

- Programming Language: Python
- Operating System: Windows 7 or later versions of windows.

7. PACKAGES USED

TensorFlow

TensorFlow is a popular open-source Python machine learning toolkit for creating and training deep neural networks. It has a versatile architecture and supports a variety of platforms, including CPU, GPU, and TPU. TensorFlow simplifies the implementation of complicated algorithms and models, allowing developers to create scalable and efficient machine learning systems.

Keras

Keras is a Python-based high-level neural network API that operates on top of TensorFlow, Microsoft Cognitive Toolkit, Theano, or PlaidML. It offers an easy-to-use interface for building and training deep learning models, letting users to easily experiment with alternative architectures and hyperparameters. Keras also provides pre-trained models as well as a huge collection of building blocks for developing sophisticated models.

Scikit-learn

Scikit-learn (also referred to as sklearn) is a widely used open-source machine learning library for Python. It provides a comprehensive set of tools and algorithms for various machine learning tasks, including classification, regression, clustering, dimensionality reduction, model selection, and pre-processing.

Scipy

Scipy is a Python package for scientific and engineering computing. It includes modules for optimization, integration, linear algebra, signal processing, and other tasks. Scipy is built on top of Numpy, another famous Python package for scientific computing, and the two combined constitute a strong data analysis and numerical calculation tool.

Numpy

NumPy is an important Python package for scientific computation. It supports huge, multidimensional arrays and matrices, as well as a diverse collection of high-level mathematical operations for these arrays. NumPy is a popular choice for numerical operations in scientific research and data analysis due to its efficient and user-friendly interface.

Pandas

Pandas is a popular open-source Python data analysis and manipulation package. It offers sophisticated data structures and tools for working with structured data, including as data frames and series, and it allows for quick data processing, cleaning, merging, and reshaping. Pandas also supports reading and writing a variety of file types, including CSV, Excel, and SQL databases.

Matplotlib

Matplotlib is a popular Python data visualization package. It includes line graphs, scatter plots, bar plots, and histograms among its 2D and 3D displays.

Matplotlib is a useful tool for data exploration and communication since it is extremely customizable and supports extensive labelling, annotations, and text formatting.

INTERNATIONAL JOURNAL OF PROGRESSIVE RESEARCH IN ENGINEERING MANAGEMENT AND SCIENCE (IJPREMS)
www.ijprems.com
editor@ijprems.com
Vol. 04, Issue 05, May 2024, pp: 242-249
e-ISSN : 2583-1062
Impact Factor: 5.725

Tkinter and NLKT

Tkinter is a standard Python library used for creating graphical user interfaces (GUIs). It provides a set of modules and classes that allow you to develop interactive and visually appealing desktop applications. NLTK is a toolkit build for working with NLP in Python. It provides us various text processing libraries with a lot of test datasets. A variety of tasks can be performed using NLTK such as tokenizing, parse tree visualization, etc NLTK (Natural Language Toolkit) is the go-to API for NLP (Natural Language Processing) with Python. It is a really powerful tool to pre-process text data for further analysis like with ML models for instance. It helps convert text into numbers.

8. TECHNOLOGY DESCRIPTION

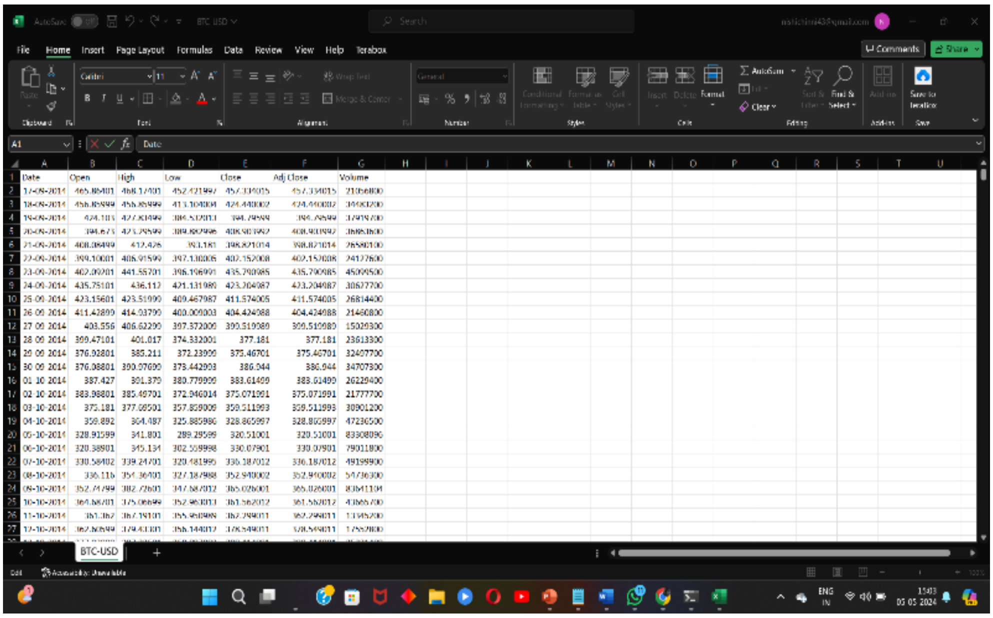
Python is an interpreted high-level programming language that is simple to learn and use. It features a basic and clear syntax that makes it suitable for both beginners and professionals. Python is utilized in many different areas, such as web development, scientific computing, data analysis, and artificial intelligence.

9. SOURCE CODE

```
import numpy as np
import matplotlib.pyplot as plt
import pandas as pd
from sklearn.preprocessing import MinMaxScaler
data = pd.read_csv('BTC-USD.csv', date_parser = True)
data.tail()
data_training = data[data['Date']< '2020-01-01'].copy()
data_training
data_test = data[data['Date']> '2020-01-01'].copy()
data_test
training_data = data_training.drop(['Date', 'Adj Close'], axis = 1)
training_data.head()
scaler = MinMaxScaler()
training_data = scaler.fit_transform(training_data)
training_data
X_train = []
Y_train = []
for i in range(60, training_data.shape[0]):
X_train.append(training_data[i-60:i])
Y_train.append(training_data[i,0])
X_train, Y_train = np.array(X_train), np.array(Y_train)
X_train.shape
from tensorflow.keras import Sequential
from tensorflow.keras.layers import Dense, LSTM, Dropout
regressor = Sequential()
regressor.add(LSTM(units = 50, activation = 'relu', return_sequences = True, input_shape = (X_train.shape[1], 5)))
regressor.add(Dropout(0.2))
regressor.add(LSTM(units = 60, activation = 'relu', return_sequences = True))
regressor.add(Dropout(0.3))
regressor.add(LSTM(units = 80, activation = 'relu', return_sequences = True))
regressor.add(Dropout(0.4))
regressor.add(LSTM(units = 120, activation = 'relu'))
regressor.add(Dropout(0.5))
regressor.add (Dense(units =1))
regressor. compile(optimizer = 'adam', loss = 'mean_squared_error')
regressor.fit(X_train, Y_train, epochs = 20, batch_size =50)
```

```
past_60_days = data_training.tail(60)
df= past_60_days.append(data_test, ignore_index = True)
df = df.drop(['Date', 'Adj Close'], axis = 1)
df.head()
inputs = scaler.transform(df)
inputs
X_test = []
Y_test = []
for i in range (60, inputs.shape[0]):
X_test.append(inputs[i-60:i])
Y_test.append(inputs[i, 0])
X_test, Y_test = np.array(X_test), np.array(Y_test)
X_test.shape, Y_test.shape
Y_pred = regressor.predict(X_test)
Y_pred, Y_test
scaler.scale_
scale = 1/5.18164146e-05
scale
Y_test = Y_test*scale
Y_pred = Y_pred*scale
Y_pred
Y_test
plt.figure(figsize=(14,5))
plt.plot(Y_test, color = 'red', label = 'Real Bitcoin Price')
plt.plot(Y_pred, color = 'green', label = 'Predicted Bitcoin Price')
plt.title('Cryptocurreny Price Prediction using RNN-LSTM')
plt.xlabel('Time')
plt.ylabel('Price')
plt.legend()
plt.show()
```

10. OUTPUT



Date	Open	High	Low	Close	Adj Close	Volume
17-09-2014	465.86401	468.17401	452.421997	457.334015	457.334015	21056800
18-09-2014	456.85999	456.85999	413.104004	424.440002	424.440002	34483200
19-09-2014	424.103	427.83499	384.532013	394.79599	394.79599	37919700
20-09-2014	394.673	423.29599	389.882996	408.903992	408.903992	36863600
21-09-2014	408.08499	412.426	393.181	398.821014	398.821014	26580100
22-09-2014	399.10001	406.91599	397.130005	402.152008	402.152008	24127600
23-09-2014	402.09201	441.55701	396.196991	435.790985	435.790985	45099500
24-09-2014	435.75101	436.112	421.131989	423.204987	423.204987	30627700
25-09-2014	423.15601	423.51999	409.467987	411.574005	411.574005	26814400
26-09-2014	411.42899	414.93799	400.009003	404.424988	404.424988	21460800
27-09-2014	403.556	406.62299	397.372009	399.519989	399.519989	15029300
28-09-2014	399.47101	401.017	374.332001	377.181	377.181	23613300
29-09-2014	376.92801	385.211	372.23999	375.46701	375.46701	32497700
30-09-2014	376.08801	390.97699	373.442993	386.944	386.944	34707300
01-10-2014	387.427	391.379	380.779999	383.61499	383.61499	26229400
02-10-2014	383.98801	385.49701	372.946014	375.071991	375.071991	21777700
03-10-2014	375.181	377.69501	357.859009	359.511993	359.511993	30901200
04-10-2014	359.892	364.487	325.885986	328.865997	328.865997	47236500
05-10-2014	328.91599	341.801	289.29599	320.51001	320.51001	83308096
06-10-2014	320.38901	345.134	302.559998	330.07901	330.07901	79011800
07-10-2014	330.58402	339.24701	320.481995	336.187012	336.187012	49199900
08-10-2014	336.116	354.36401	327.187988	352.940002	352.940002	54736300
09-10-2014	352.74799	382.72601	347.687012	365.026001	365.026001	83641104
10-10-2014	364.68701	375.06699	352.963013	361.562012	361.562012	43665700
11-10-2014	361.362	367.19101	355.950989	362.299011	362.299011	13345200
12-10-2014	362.60599	379.43301	356.144012	378.549011	378.549011	17552800

Fig 10.1: Dataset Values

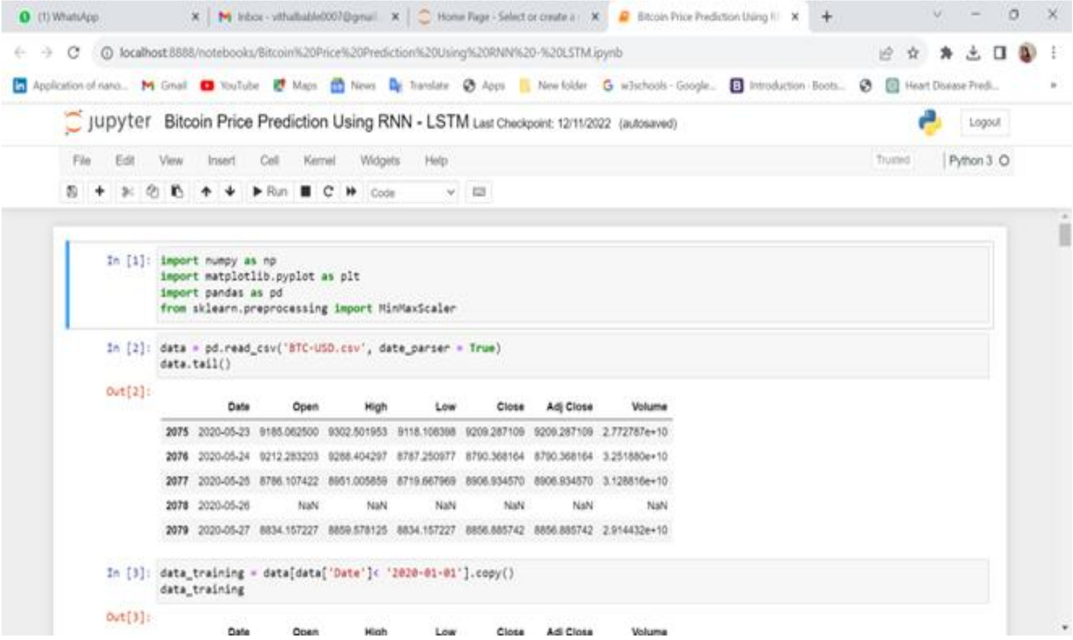


Fig 10.2: Importing Libraries & Checking dataset Values

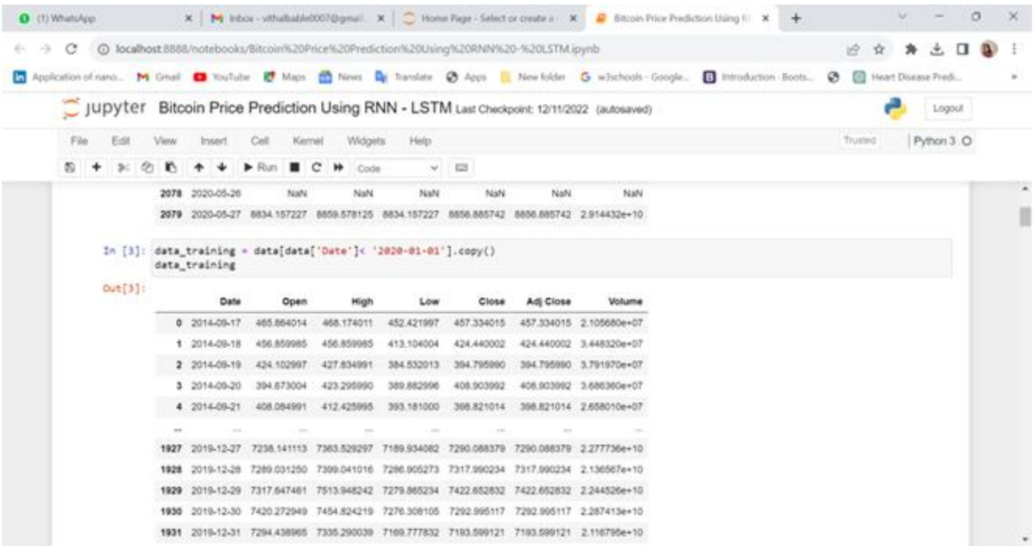


Fig 10.3: Scaling & Fitting the data into Training data

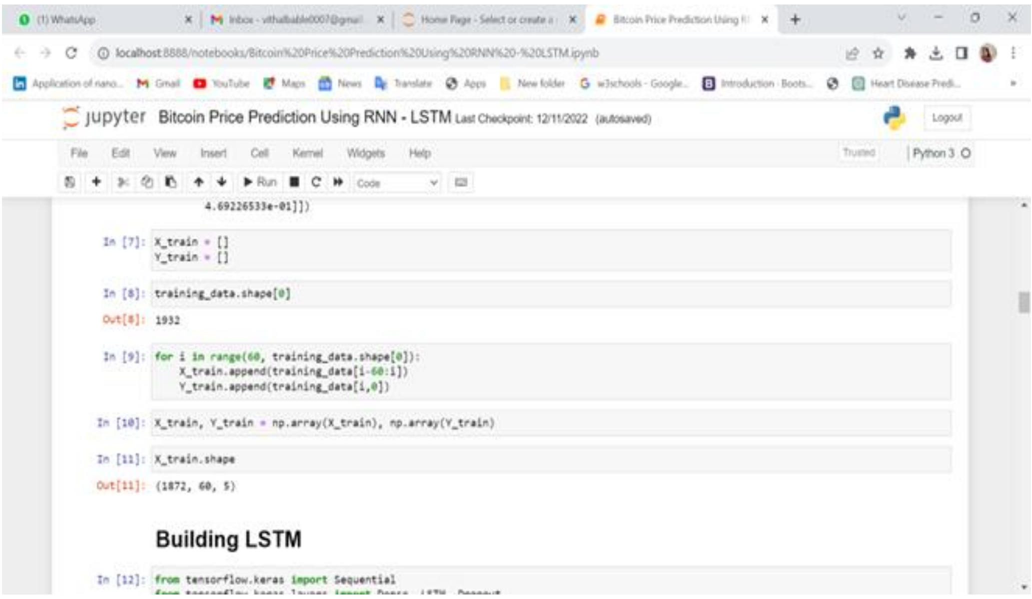


Fig 10.5: Building LSTM Model

INTERNATIONAL JOURNAL OF PROGRESSIVE RESEARCH IN ENGINEERING MANAGEMENT AND SCIENCE (IJPREMS)

www.ijprems.com
editor@ijprems.com

Vol. 04, Issue 05, May 2024, pp: 242-249

e-ISSN : 2583-1062

Impact Factor: 5.725

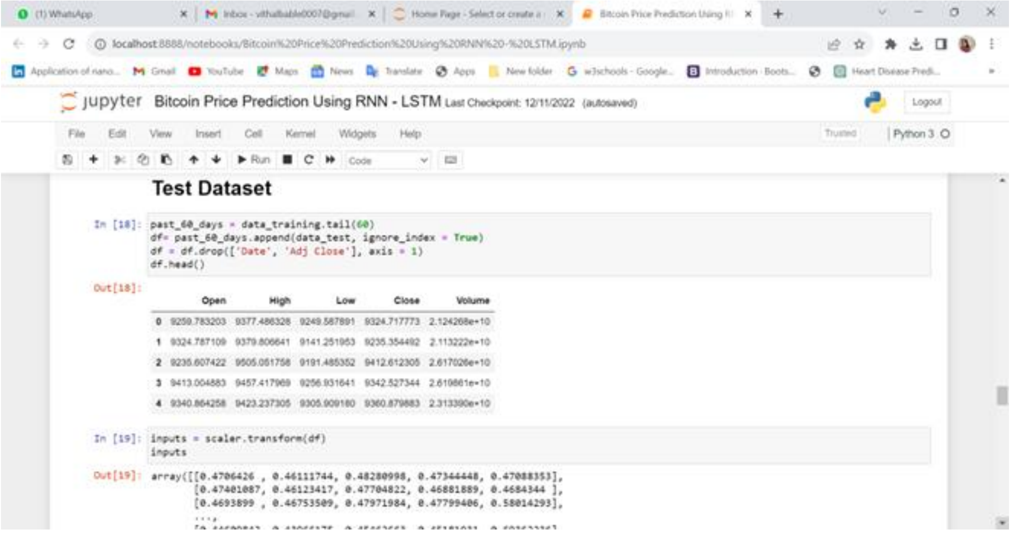


Fig 10.5: Testing the dataset

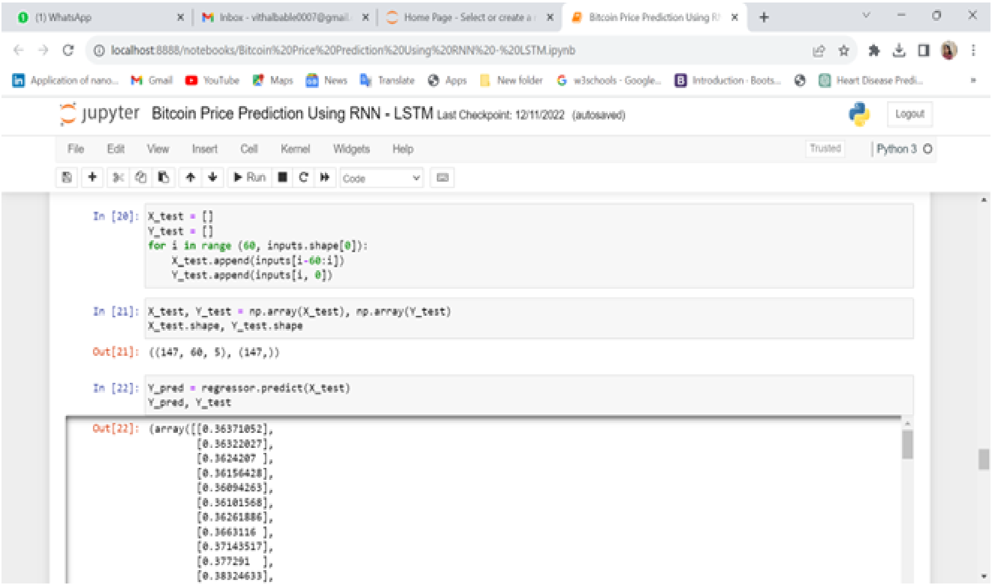


Fig 10.7: Predicting the cryptocurrency price

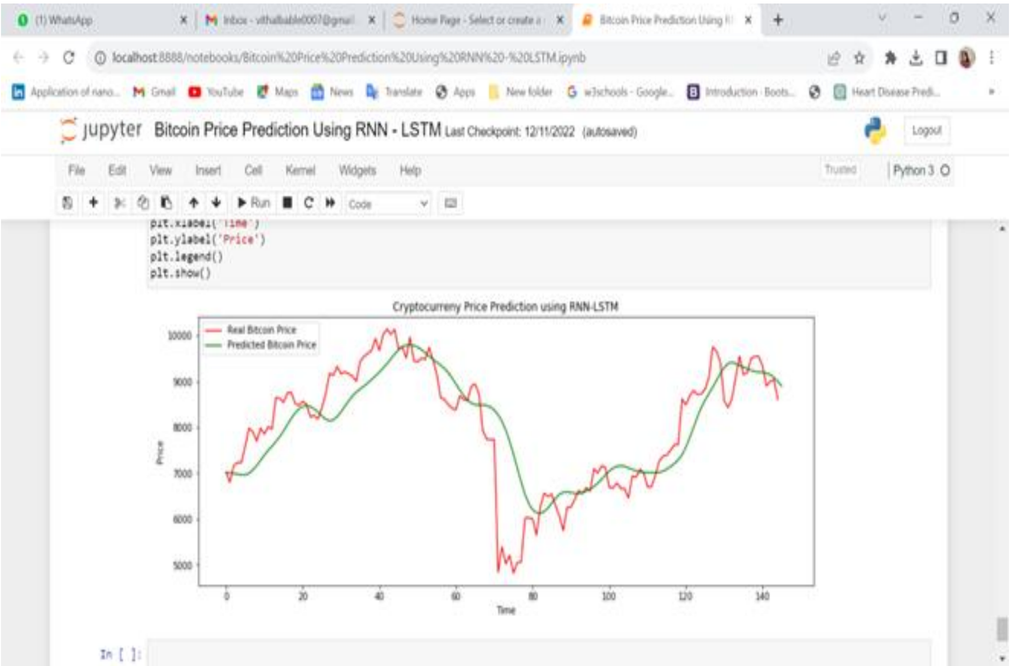


Fig 10.8: Final Output Predicted Cryptocurrency Price

11. CONCLUSION

Our aim of study is to develop model which predict bitcoin price using deep learning. Since deep learning is used to select the parameter to get successive outcomes in developing model. In this latter we implemented for three proposed model RNN, LSTM and GRU we found that total parameter and dataset can influence result. The Previous model developed using RNN and LSTM which had less predicted accuracy that is 52% approximately. Whereas in our comparative analysis GRU model result better as comparatively LSTM model. The Optimal model for GRU result accuracy is 94.70%. The proposed model shows about 42.3% of accuracy improvement. The tests of all our GRUs show the highest detailed outcomes which take time. Furthermore, Selected features: Low, High, Close and Open can't be enough to predict the Bitcoin value, as various factors, including social media responses, legislation and laws each country advertises for handling the digital currency won help to increase and lower the Bitcoin price. Therefore, modified information should always be gathered and applied for the best results of all models.

12. FUTURE SCOPE

- Advanced Model Development

Continuously advancing the LSTM model by integrating cutting-edge techniques from deep learning research, such as transformer architectures, attention mechanisms, or meta-learning approaches.

-Real – Time Application and Integration

Expanding the application of the LSTM model to real-time scenarios and integrating it into cryptocurrency trading platforms, investment apps, or financial services.

-Interdisciplinary Collaboration and Innovation interdisciplinary collaboration and innovation by engaging with experts from fields such as finance, economics, computer science, and data science.

13. REFERENCES

[1] A Cryptocurrency Price Prediction Model using Deep Learning,[Online] https://www.researchgate.net/publication/371309639_A_Cryptocurrency_Price_Prediction_Model_using_Deep_Learning [Accessed on 2023].

[2] Bitcoin: A Peer-to-Peer Electronic Cash System [Online] https://www,researchgate.net/publication/228640975_Bitcoin_A_Peer-to-Peer_Electronic_Cash_System [Accessed on 2009]

[3] Stochastic Neural Networks For Cryptocurrency Price Prediction[Online] https://www.researchgate.net/publication/340884479_Stochastic_Neural_Networks_For_Cryptocurrency_Price_Prediction [Accessed on 2020]