Cryptocurrency Prediction and Alert Using Email

Submitted in partial fulfilment of the requirements for the award of degree of

BACHELOR OF ENGINEERING IN COMPUTER SCIENCE & ENGINEERING

(Established under Galgotias University Uttar Pradesh Act No. 14 of 2011)

Under The Supervision of : Dr. B.Balamurgan.

Assistant Professor

Submitted By:

Abhinav Arora – 18SCSE1180009

Tushar Sisodia– 18SCSE1180020

SCHOOL OF COMPUTING SCIENCE AND ENGINEERING DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERINGGALGOTIAS UNIVERSITY, GREATER NOIDA

SCHOOL OF COMPUTING SCIENCE AND ENGINEERING
GALGOTIAS UNIVERSITY, GREATER NOIDA

CANDIDATE'S DECLARATION

I/We hereby certify that the work which is being presented in the project, entitled **" cryptocurrency price predictionand alert using mail"** in partial fulfillment of the requirements for the award of the B.Tech submitted in the School of Computing Science and Engineering of Galgotias University, Greater Noida, is an original work carried out during the period 2021-22, under the supervision of Dr. Michal Raj TF Designation, Department of Computer Science and Engineering, of School of Computing Science and Engineering , Galgotias University, Greater Noida

The matter presented in the thesis/project/dissertation has not been submitted by me/us for the award of any other degree of this or any other places.

Abhinav Arora,18SCSE1180009
Tushar Sisodia,18SCSE1180020

This is to certify that the above statement made by the candidates is correct to the best of my knowledge.

Supervisor Name - Dr. B. Balamurgan
Professor

CERTIFICATE

The Final Project Viva-Voce examination of Abhinav Arora-18SCSE1180009, Tushar Sisodia -18SCSE1180020 has been held on __________________ and his/her work is recommended for the award of Bachelors of Technology.

Signature of Examiner(s) **Signature of Supervisor(s)**

Signature of Project Coordinator

Signature of Dean

Date : December, 2021
Place : Greater Noida

Table of Contents

Abstract:-

Stock investors all agree that one of the toughest things they have to do is keep up with the speedy changes within the markets. The internet contains a huge amount of up-to-the minute records for buyers Such As stock price ,exchange rate, sales yolumes and much more. Although such stock-related facts is publicly and freely available over the internet, investors face problems monitoring the modifications and updates fast enough to make s funding decisions.

Our research objective is to find a time-consuming and accurate model of Bitcoin price prediction from different machine learning models (Multivariate Linear Regression) and advanced learning algorithms such as (LSTM, R-NN)

This paper also proposes a model to monitor stock available from internet offerings or dynamic HTML documents at the internet. This device can cope with public data available at the internet on a single or more than one web sites and from a group or bunch a couple of web services because the information assets. The model permits investors to apply these sources to create personalised stock tracking criteria after which acquire updated notifications from the device primarily based on the required standards., E mail notification messages can be generated and sent to the investor notifying him/her whenever the standards are met. some of applications associated with inventory investments can benefit form this machine to provide actual-time useful information to the buyers.

Introduction:-

Nowadays, cryptocurrency is a trendsetter. Within the financial sector, over time Bitcoin has emerged as one of the most popular digital currencies distributed. Presented bySatoshi Nakamoto in January 2009.

Unlike stock market trading, cryptocurrency trading platforms open up daily volatility and integration of market volatility, and cryptocurrency traders and managers face the challenge of trying to keep track of all the prices of their assets. For example, in Bitmaxexchanges, there was an unexpected 80% drop in its BTMX token that required 31 minutes [1]. It would not have been possible for them to monitor all price actions on hand and on a regular basis. Therefore, our research objective is to find a small time-consuming and accurate model for predicting

The price of Bitcoin comes from “machine learning and deep learning”. There are various factors like the price of gold, economic hardship, social media tweets continue to affect bitcoin price. Features open, close, high and low bitcoin daily price used for forecasting function. In this paper, we predict daily price changes of many cryptocurrencies. A few of them are bitcoin, ripple,

NMC and more. We suggest a pricing method

using one of the most popular methods of machine learning, i.e.

multivariate line reversal. Our program begins with pre-processing data, where we clean up the database by removing lines

at missing value. Next, we examine the independent

features in the database, which help us to predict the highest quality

the price of cryptocurrency. Next we find a link

between dependent and independent variables, and finally, we can cost forecasting.

We are also making bitcoin pricing alert system using IoT, so that when the bitcoin price reaches

The limit price (the amount at which the user wants to sell bitcoin) will send a warning to the user in various ways such as sending a text message, and sending an email .

1. What is cryptocurrency?

Cryptocurrency is a form of payment that can be exchanged online for goods and services.Many companies have issued their own currencies, often called tokens, and these can be traded specifically for the good or service that the company provides. Think of them as you would arcade tokens or casino chips. You'll need to exchange real currency for the cryptocurrency to access the good or service. Cryptocurrencies work using a technology called blockchain. Blockchain is a decentralized technology spread across many computers that manages and records transactions. Part of the appeal of this technology is its security.

How does cryptocurrency work?

Cryptocurrencies run on a distributed public ledger called blockchain, a record of all transactions updated and held by currency holders. Units of cryptocurrency are created through a process called mining, which involves using computer power to solve complicated mathematical problems that generate coins. Users can also buy the currencies from brokers, then store and spend them using cryptographic wallets.

If you own cryptocurrency, you don't own anything tangible. What you own is a key that allows you to move a record or a unit of measure from one person to another without a trusted third party.

Although Bitcoin has been around since 2009, cryptocurrencies and applications of blockchain technology are still emerging in financial terms, and more uses are expected in the future. Transactions including bonds, stocks, and other financial assets could eventually be traded using the technology.

Understanding Bitcoin

Bitcoin is a cryptocurrency founded in January 2009. It is the world's most valuable crypto currency and is traded in more than 40 exchanges worldwide, accepting more than 30 currencies. As a currency, Bitcoin offers a new opportunity to predict prices as it has a high, very high volatility compared to traditional currencies.

The bitcoin system is a set of separate nodes that use the bitcoin code and maintain its blockchain. Figuratively, a blockchain can be considered as a set of blocks. For each block, there is a set of tasks. Because all computers using the blockchain have the same list of blocks and transactions, and can clearly see these new blocks being filled with new bitcoin transactions, no one can cheat the system.

Bitcoin uses peer-to-peer technology to make quick payments. Miners are responsible for processing blockchain transactions and are driven by repo funds.

The way bitcoin works is the key to understanding why it is so popular. Unlike other investments, cryptocurrency is not tied to tangible assets or to the American dollar. Its main purpose is to allow two people anywhere to exchange direct value. What this means is that there is no medium that controls this network. No government, no big bank can close down or raise or reduce the price without a reason.

It will be interesting to see how central banks start making their money digitally. As financial systems become more digital, it leads to more common bitcoin, but digital currency recovery is also closely related to the global financial situation.

Each time someone performs a task, a separate encrypted signature is added to the manual for confirmation

2. Related work

Financial market forecasting is a major field of financial research and has been extensively analyzed.5 There is mixed evidence about the predictions of financial markets.7,8 The standard method for analyzing forecasting indicators is to perform a retrospective analysis of potential signals. to define return on assets.9,10 However, a regression line can not dynamically integrate a large number of factors and put strong ideas on how performance signals indicate market movements. In contrast, machine learning methods, which often do not limit those, have been widely used in financial market predictions.11,3 Among those, neural network-based methods may be expected to be particularly relevant, as already described. to be an outstanding way to predict the volatility of financial markets.12

2.1. Market efficiency and financial market predictions

2.1.1. Theory of market efficiency

In successful financial markets, prices reflect all available information and cannot be predicted to earn unusual returns. In order to determine the level of market efficiency, Fama7 defines a formal framework for three levels — the weak, the weakest, and the effective market performance of the strong form. In well-functioning markets, prices reflect all information about past prices, while in less efficient markets, prices reflect all available information publicly. In well-functioning markets, prices also reflect all confidential information. Although regulators aim to prevent investors from accessing confidential information, it is generally agreed that large financial markets operate more efficiently. time. If information is expensive and prices always show all available information, experienced traders will stop to get information, leading to market prices deviating from the value of the underlying commodity. Besides, there is evidence in the financial books of science about a large number of potential problems in the markets. Green et al, 15 for example, point to more than 330 forecasting signals officially available in the US stock market published between 1970 and 2010. Similarly, Lo8 forms the hypothesis of dynamic markets, depending on whether the markets may exist temporarily. ineffective. Thus, the length of the short-term malfunction is influenced by the level of competition between the market and the limits of arbitrage.16 Experienced traders use these inefficiencies so that prices reflect all available information as well. In summary, the question remains open, whether the forecasts signal market volatility or represent risk factors with fair value. Also, some of the most prominent symbols have disappeared after publication, 17 indicating that part of the predicted symbols may only exist in the sample period or have been deleted due to traders using exploitation strategies. The green and al15 determine whether the combined model of market efficiency or inefficiency should address the return indicators that are continuously identified.

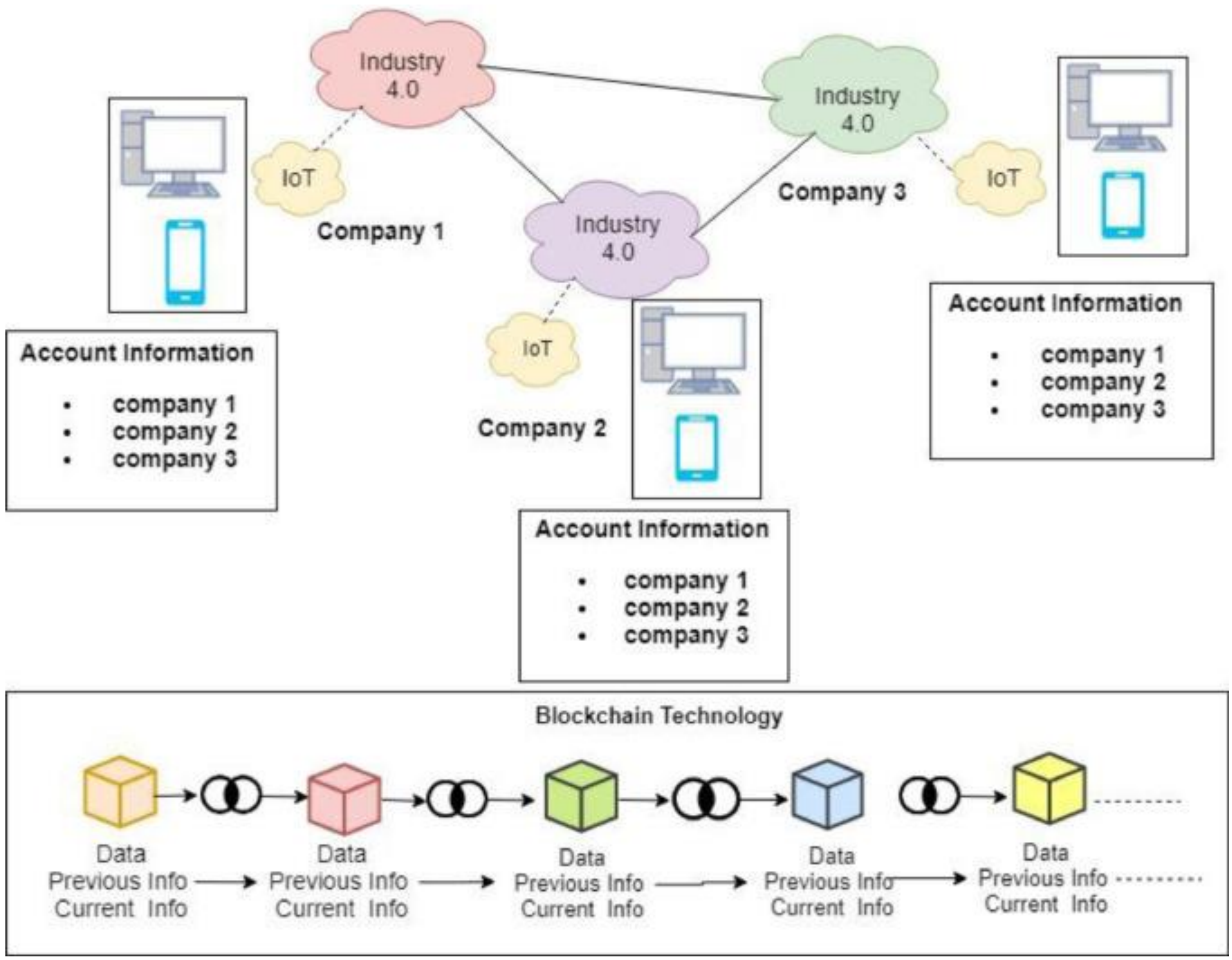
2.1.2. Bitcoin market performance

Several findings in the financial books18, 19, 20, 21 suggest that bitcoin may cover a new class of assets. Therefore, the findings regarding the effectiveness of the weaker form in some financial markets may not hold up to the bitcoin market. Several researchers are evaluating the efficiency of the bitcoin market using different time horizons. First, Urquhart22 is investigating a series of daily bitcoin prices (August 2010 to July 2016). You find that the bitcoin market does not work even in a weaker form. However, dividing the study period reveals that the bitcoin market is becoming more and more efficient over time. Reviewing this data, Nadarajah and Chu23 found that the reversal of the return bitcoin used used to satisfy the fragile well-functioning form of the hypothesis market. Similarly, Bariviera24 examines bitcoin prices daily (August 2011 to February 2017), using Hurst exponent25 and shows that the bitcoin market is not a weaker form before 2014, but a weaker effective form after 2014. Vidal-Tomás and Ibañez approach the question of semi-strong form bitcoin market success from an event research perspective.26 With data on issues related to monetary policy changes and bitcoin (September 2011 to December 2017), they show that the bitcoin market does not respond to changes monetary policy but is becoming increasingly effective. about bitcoin-related events. Examining the adaptive market hypothesis, Khuntia and Pattanayak27 analyze bitcoin prices daily (July 2010 to December 2017), finds evidence of a growing level of market performance in a weaker form. They conclude that these findings are evidence that the adaptive market hypothesis has dominated the bitcoin market.

In summary, there is mixed evidence among experts about the effectiveness of the bitcoin market. However, many researchers find that the bitcoin market has been very successful over the years. The growing level of market efficiency seems justified, as the bitcoin market has grown since its inception and, therefore, increased competition.

2.2. Bitcoin market prediction by machine learning

Conducting book reviews, Jaquart et al5 analyzes bitcoin market prediction books by machine reading published until April 2019. They examined the body of literature in terms of machine learning methods, features of return predictions, prediction horizons, and forecast types. The revised textbook uses both editing and retrospective models almost equally often, while retrospective models are used gradually. Due to the use of different horizons, target and flexibility feature, parameter specification



Literature Review:-

We have all speculated when the cost of bitcoin will be one year, two years, five years or even 10 years from now. It's really hard to expect but each of us loves to do it. Big profits can be made by buying and selling bitcoins, whenever they are done correctly. But this does not come without your evil again. If you are not careful and well calculated, you can lose a lot of money too. You have to have an amazing understanding of why and why bitcoin costs change (organic market, guidelines, news, etc.), which means you have to be careful how people make their bitcoin predictions. Considering these factors (offers and demands, regulations, news, etc.), one should also consider bitcoin technology and its progress. This aside, we now have to deal with technical aspects using various algorithms and technologies that can predict accurate bitcoin prices. Although we have encountered various models that currently exist as Biological neural networks. (BNN), Frequent Neural Network (RNN), Short-term Memory (LSTM), Integrated Auto Regressive Rate (ARIMA), etc. by machine learning and in-depth concepts of the neural network. A time series is usually a sequence of numbers over time. This is because this is a timetable data set, the total data sets should be divided into two parts: input and output.Moreover, LSTM is much better compared to older models' models, as it can easily handle. many input prediction problems.

The following paper [3] uses the machine learning algorithms to predict the preice of bitcoin and all. But as I describe it earlier that these algorithms can produce the results over short time which is not as useful for investing purpose as they used the data gathered from twitter and reddit but sometimes it is not useful as it should be sometimes the negativity of that hollow information could lead to the fault or imbalance in the prediction .

The use of RNN model by [2] they execute the famous Auto backward incorporated moving normal (ARIMA) model for time arrangement gauging as a correlation with the profound learning models. The ARIMA forecast is out performed by the nonlinear deep learning methods which performed much better. Finally both the profound learning models are benchmarked on both a GPU and CPU.

The training time on the CPU is outflanked by the GPU execution by 67.7%. In the base papers selected by us, the author collected a data set of over 25 features relating to the bitcoin price and payment network over a period of five years, recorded on a daily basis were able to predict the sign of the daily bitcoin price change with an incredible accuracy of 98.7%. It is kindly notice that RNN could be used in that field too bu the acuuracy we get from the LSTM will be more than the any other useful algorithms.

In the second phase of our project what we are going to do is alerting the people using the mail notification method as you know many research papers just predict the price of the coin and other currencies but what is the use of prediction if it cannot be effectively used by the user or encrypted for their use so what we are going to use is using the SMTP library to make a function which will send the mail notification of the change in price of their selected coin as you know it is not htat much easy or handiful task to look over their price of the particular coin continuously so it would be effective to get a notification

Project Design:-

1.System Architecture

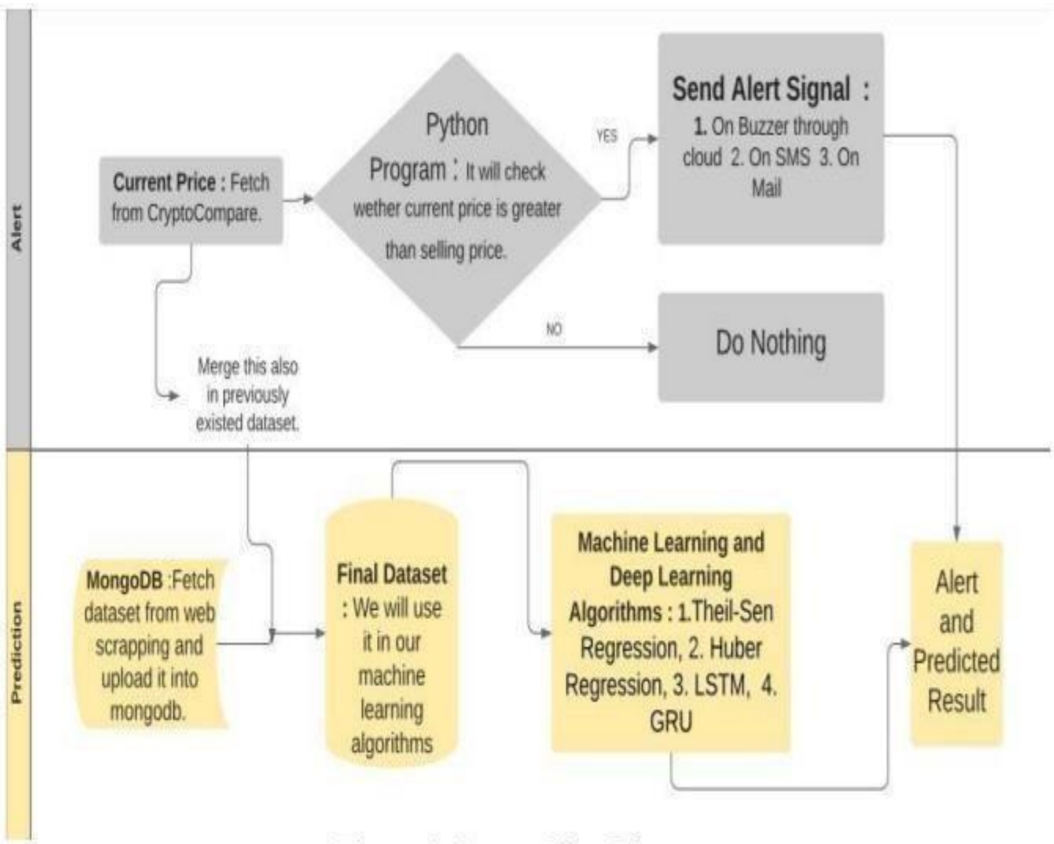
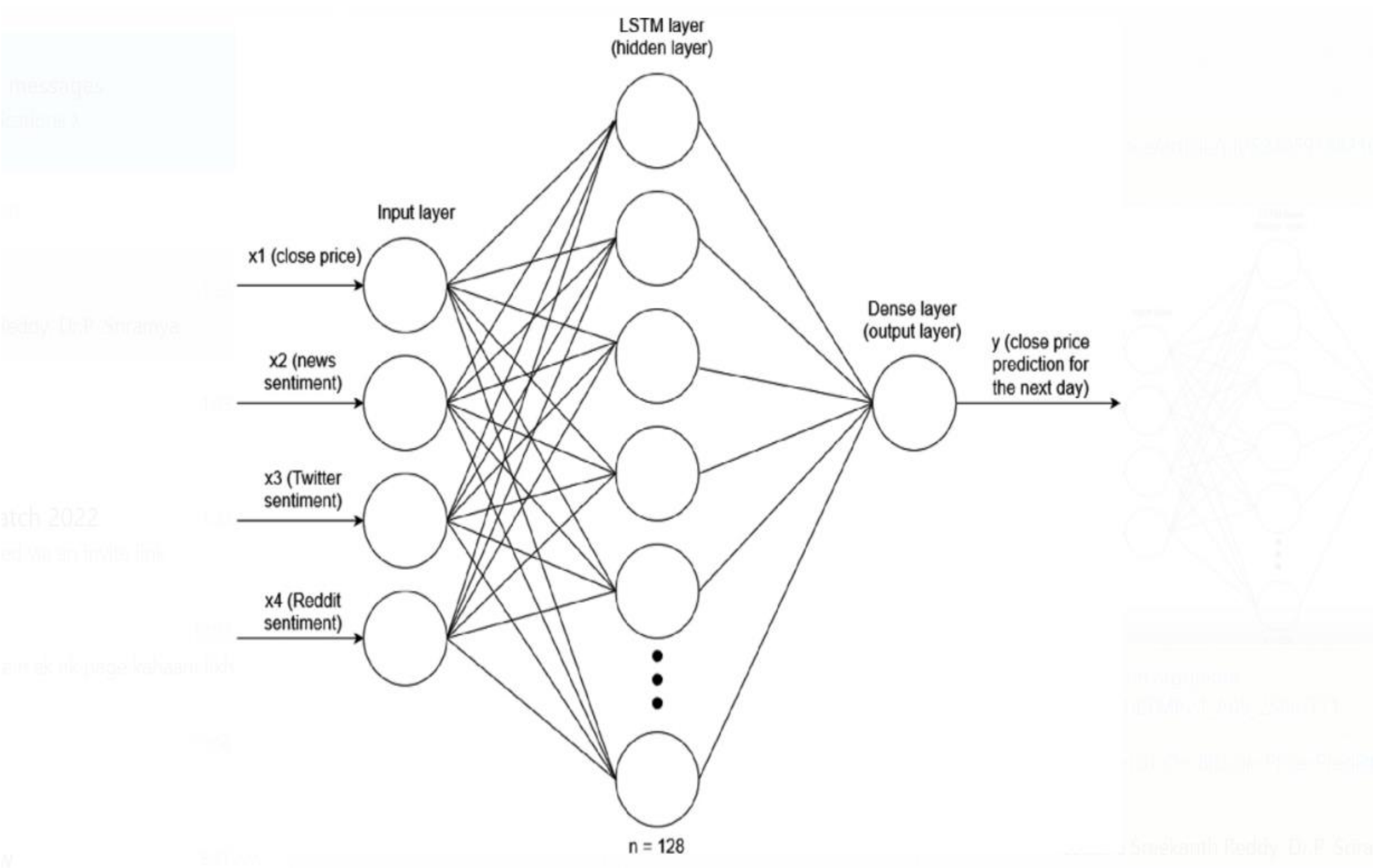


Figure 1. System Flow Diagram

UML diagram:-


LSTM layer
(hidden layer)
Input layer
x1 (close price)
x2 (news sentiment)
x3 (Twitter sentiment)
x4 (Reddit sentiment)
Dense layer
(output layer)
y (close price prediction for the next day)
n = 128

The Case Diagram of the Proposed System

The case diagram of the proposed system is shown in fig 11. The proposed system allows the user to provide the training data and the threshold constant. Training Neural network requires initialization of input weights and the first input weight will be adjusted until the optimal prediction is achieved

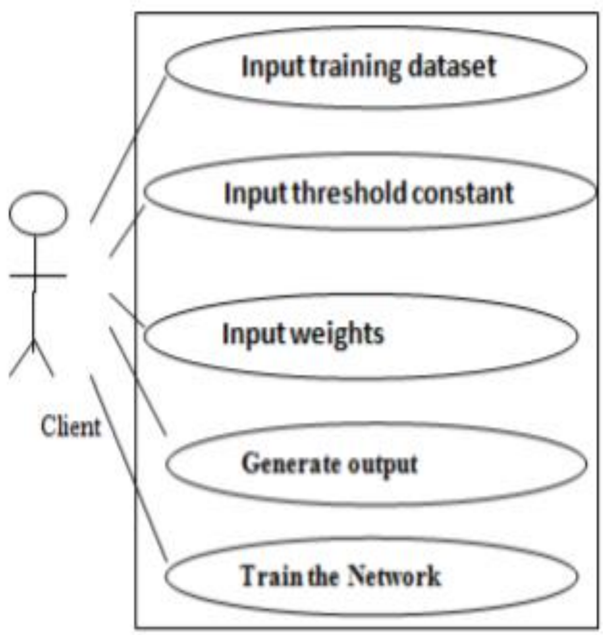
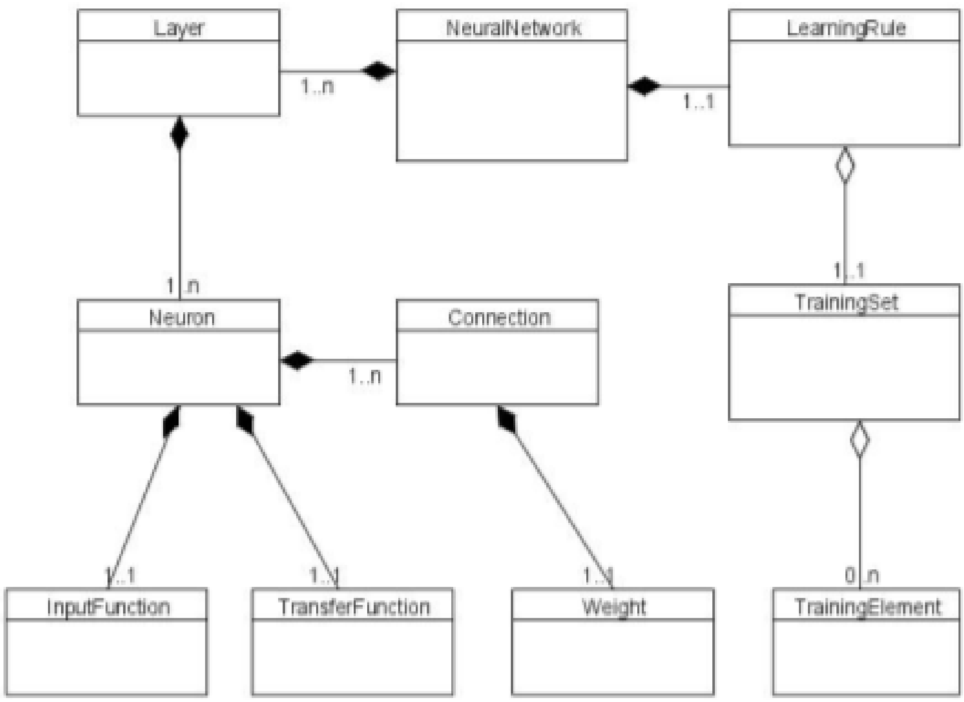


Fig 11.The Case Diagram of the Proposed System.

Activity Diagram and Sequence Diagram of the Proposed System

. An activity diagram is essentially a flowchart, showing flow of controls from one activity to another. Unlike a traditional flowchart, it can model the dynamic functional view of a system. An activity diagram represents an operation on some classes in the system that results to changes in the state of the system. From the diagram shown in fig , the client is expected to provide the input dataset and the required output. The required output is used in back propagation, for the system uses it to compare its predicted value from time to time in other to get the optimal prediction. The client select a threshold constant and looking at the input value the neural network initializes the weight for the input layer and the hidden layer. Then the calculation for the activation function of both the input and the output layer is done and the system calculates the error. If the is large then the system adjust the weight and backpropagate it to the input layer for further calculation to be done. The system outputs its prediction whenever the error is small. shows the sequence diagram of the proposed system. The sequence diagram number the actions starting from the data input to the optimal prediction. There is an arrow direction to show the sequence of flow for the action taking to arrive at the optimal prediction.

Fig below shows the class diagram for the proposed system. The layer is connected to both the parent Network and the Neurons also the neuron connects to the parent Layer while the Neural Network connects to the Layer. The Neural Network is connected to the Learning Rule, the Layer, output Neurons and the input Neuron while the Layer and the Learning Rule are connected to the Neural Network. The Connection connects the Neuron and the Weight while the Neurons connect the input Connection and the output Connection. The Connection connects the Weight. The Learning Rule connects the Neural Network and the Learning Rule connects the Neural. The Input Function connects Summing Function. The input Function is connected to the Summing Function and Weight Function. The Neuron Connects the Transfer Function. The Neuron connects the Input Connection, the Output Connection, the Layer, the Input Function and the transfer Function. The Connection and the layer are connected to the neuron while the output Neuron and the Input Neuron are connected to the Neural Netork. The Connection connects the Weight.



the architectural design of the system

SYSTEM IMPLEMNTETION:

3. Data Collection:-

For the collection of price of cryptocurrency to test and train the model we use a api key to store or access the data.

API（Application Programming Interface）Service platform is committed to building a cooperative, prosperous and open API economic ecosystem for the simplified, modularized and standardized application of technology such as artificial intelligence, Internet of Things and blockchain. In this way, global enterprises, individuals can simply use emerging technology and build up a perfect ecology of economic application.

API returns the current price of any cryptocurrency and all the trading info (price, vol, open, high, low etc) for the requested pairs. We provide pricing data for 5,300+ coins and 240,000+ currency pairs.

Get daily, hourly and minute historical data, daily data at any given timestamp, daily average price based on hourly vwap and total daily and hourly exchange volume.

CryptoCompare's APIs provide highly reliable and scalable endpoints, reaching 180 million requests per hour at peak times (25 billion per month). Our infrastructure is running on multiple servers across several data centres to ensure the fastest data delivery and the lowest latency possible. We have redundant hosting and load-balanced environments for maximum reliability.

API endpoints use HTTPS (TLS encryption) so user traffic is secured when requesting and receiving data. To ensure maximum security, you can also request that we sign our API responses. Some endpoints require registration and the generation of an API Key.

With the largest variety of markets and the biggest value - having reached a peak of 318 billion USD - Bitcoin is here to stay. As with any new invention, there can be improvements or flaws in the initial model however the community and a team of dedicated developers are pushing to overcome any obstacle they come across. It is also the most traded cryptocurrency and one of the main entry points for all the other cryptocurrencies. The price is as unstable as always and it can go up or down by 10%-20% in a single day.

Bitcoin is an SHA-256 POW coin with almost 21,000,000 total minable coins. The block time is 10 minutes. See below for a full range of Bitcoin markets where you can trade US Dollars for Bitcoin, crypto to Bitcoin and many other fiat currencies too.

Subscription

Once connected, subscribe to your desired channels using the subscription ID. For example to subscribe to the CryptoCompare Aggregate Index for Bitcoin in USD, your subscription ID will be 5~CCCAGG~BTC~USD. You can send a list of subscriptions in an array to subscribe to all channels of interest at once.

Your subscription message should be:

```
{
    "action": "SubAdd",
    "subs": ["5~CCCAGG~BTC~USD", "0~Coinbase~ETH~USD", "2~Binance~BTC~USDT"]
}
```

Your unsubscribe message should be:

```
{
    "action": "SubRemove",
    "subs": ["5~CCCAGG~BTC~USD", "0~Coinbase~ETH~USD", "2~Binance~BTC~USDT"]
}
```

You can subscribe to the following channels:

Type	Channel	Subscription	Example
0	Trade	0~{exchange}~{base}~{quote}	0~Coinbase~BTC~USD
2	Ticker	2~{exchange}~{base}~{quote}	2~Coinbase~BTC~USD
5	Aggregate Index (CCCAGG)	5~CCCAGG~{base}~{quote}	5~CCCAGG~BTC~USD
8	Order Book L2	8~{exchange}~{base}~{quote}	8~Coinbase~BTC~USD
11	Full Volume	11~{base}	11~BTC
21	Full Top Tier Volume	21~{base}	21~BTC
24	OHLC Candles	24~{exchange or CCCAGG}~{base}~{quote}	24~CCCAGG~BTC~USD
30	Top of Order Book	30~{exchange}~{base}~{quote}	30~Coinbase~BTC~USD

3. Technologies used

1. Recurrent Neural Networks

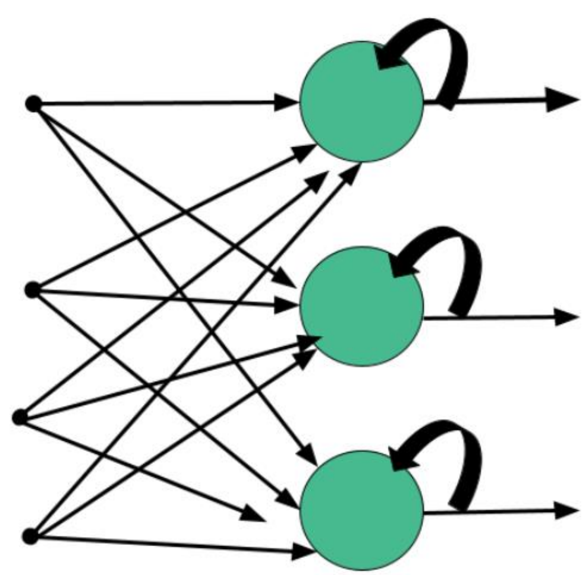
RNNs are a robust and powerful type of neural network and are considered one of the

most professional algorithms because they are the only ones with internal memory.

Recurrent neural networks were first created in the 1980s, but only in recent years has their true potential been realized. The increase in its computational power, along with the gigantic amounts of data we now have to work with, and the invention of short-term memory (LSTM) in the 1990s, has really brought RNNs to the fore.

The algorithm performs very well for sequential data such as time series, speech, text, financial data, audio, video, weather, and more. RNNs are able to form a much deeper understanding of a sequence and its context compared to other algorithms.

In an RNN, the information goes through a cycle. When making a decision, it considers the current input and also what it has learned from the inputs it has received previously.The image below illustrates how the flow of information works in the RNN algorithm.



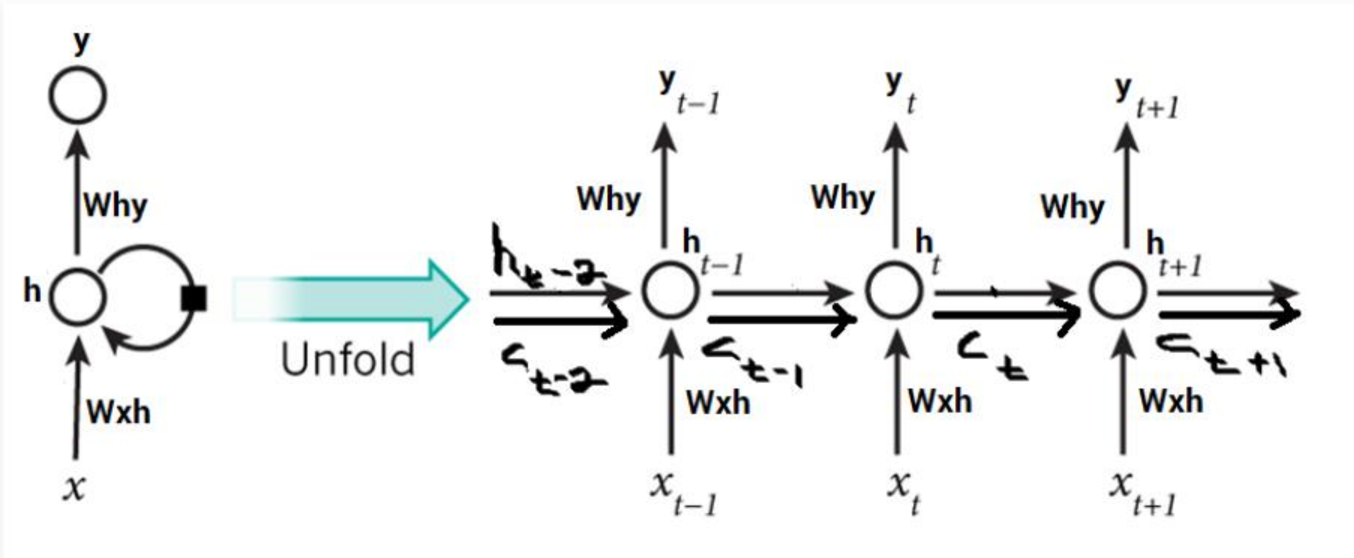
LSTM implementation:-

Long Short Term Memory is a kind of recurrent neural network. In RNN output from the last step is fed as input in the current step. LSTM was designed by Hochreiter & Schmidhuber. It tackled the problem of longterm dependencies of RNN in which the RNN cannot predict the word stored in the long-term memory but can give more accurate predictions from the recent information. As the gap length increases RNN does not give an efficient performance. LSTM can by default retain the information for a long period of time. It is used for processing, predicting, and classifying on the basis of time-series data.

Architecture:

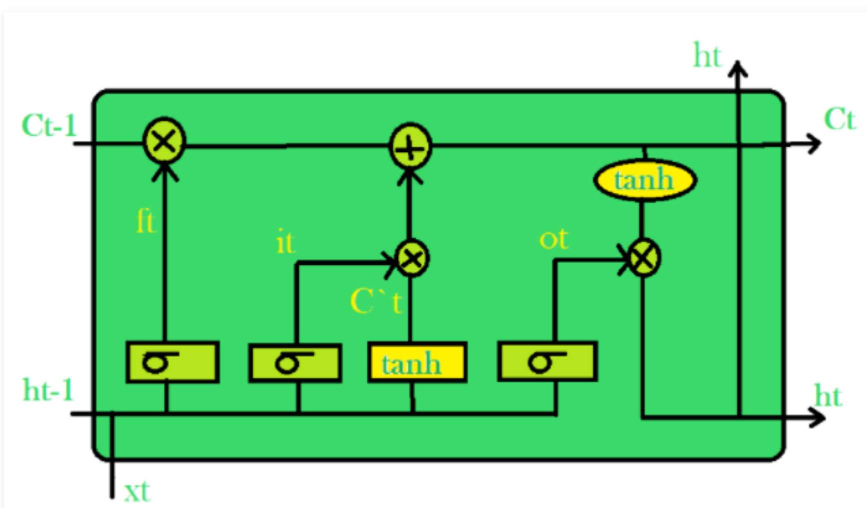
The basic difference between the architectures of RNNs and LSTMs is that the hidden layer of LSTM is a gated unit or gated cell. It consists of four layers that interact with one another in a way to produce the output of that cell along with the cell state. These two things are then passed onto the next hidden layer. Unlike RNNs which have got the only single neural net layer of tanh, LSTMs comprises of three logistic sigmoid gates and one tanh layer. Gates have been introduced in order to limit the information that is passed through the cell. They determine which part of the information will be needed by the next cell and which part is to be discarded. The output is usually in the range of 0-1 where ‘0’ means ‘reject all’ and ‘1’ means ‘include all’.

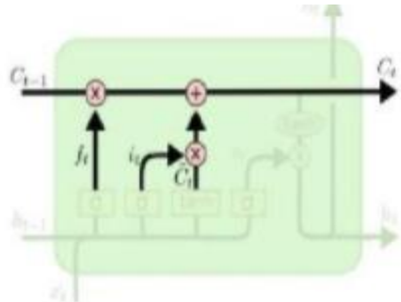
Hidden layers of LSTM :



Each LSTM cell has three inputs h_{t-1}, C_{t-1} and x_t and two outputs h_t and C_t. For a given time t, h_t is the hidden state, C_t is the cell state or memory, x_t is the current data point or input. The first sigmoid layer has two inputs– h_{t-1} and x_t where h_{t-1} is the hidden state of the previous cell. It is known as the forget gate as its output selects the amount of information of the previous cell to be included. The output is a number in [0,1] which is multiplied (point-wise) with the previous cell state C_{t-1}.

Conventional LSTM:

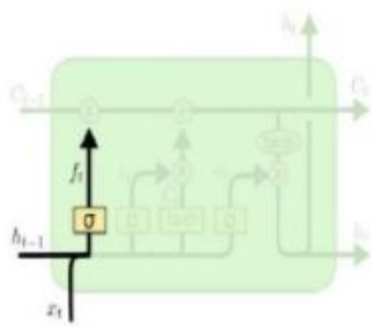




$$C_t = f_t * C_{t-1} + i_t * \tilde{C}_t$$

Figure 5:

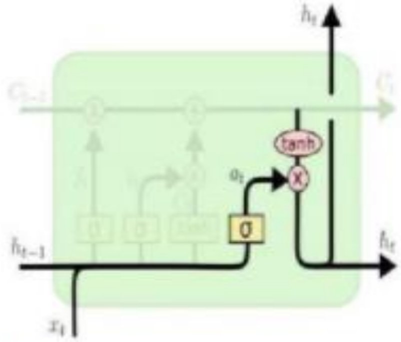
Forget gate layer



$$f_t = \sigma\left(W_f \cdot [h_{t-1}, x_t] \ + \ b_f\right)$$

Figure 6:

Output layer



$$o_t = \sigma\left(W_o \ [h_{t-1}, x_t] \ + \ b_o\right)$$
$$h_t = o_t * \tanh\left(C_t\right)$$

Figure 7:

Input gate layer

STEPS:-

1.Getting Real Time Cryptocurrency Data:-

The data is collected the current data for Bitcoin from cryptocompare

```
endpoint = 'https://min-api.cryptocompare.com/data/histoday'
res = requests.get(endpoint + '?fsym=BTC&tsym=CAD&limit=500')
hist = pd.DataFrame(json.loads(res.content)['Data'])
hist = hist.set_index('time')
hist.index = pd.to_datetime(hist.index, unit='s')
target_col = 'close'
```

```
hist.drop(["conversionType", "conversionSymbol"], axis = 'columns', inplace = True)
```

```
hist.head(5)
```

	high	low	open	volumefrom	volumeto	close
time						
2020-07-19	12580.29	12414.84	12497.20	87.39	1089684.48	12536.02
2020-07-20	12581.67	12415.65	12536.02	187.85	2347450.10	12458.63
2020-07-21	12816.63	12457.68	12458.63	292.54	3707728.51	12732.47
2020-07-22	12969.45	12596.97	12732.47	242.62	3084529.36	12969.45
2020-07-23	12984.24	12785.37	12969.45	348.66	4494307.45	12952.75

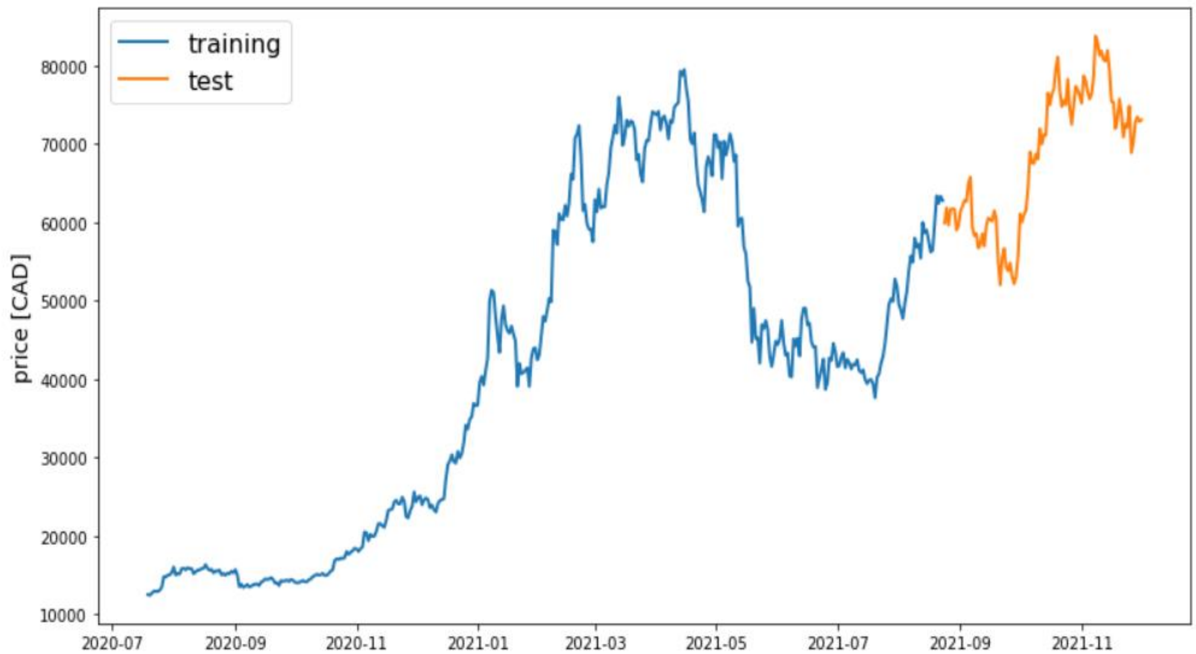
2.Training and Testing the Dataset:-

```
def train_test_split(df, test_size=0.2):
    split_row = len(df) - int(test_size * len(df))
    train_data = df.iloc[:split_row]
    test_data = df.iloc[split_row:]
    return train_data, test_data
```

```
train, test = train_test_split(hist, test_size=0.2)
```

```
def line_plot(line1, line2, label1=None, label2=None, title='', lw=2):
    fig, ax = plt.subplots(1, figsize=(13, 7))
    ax.plot(line1, label=label1, linewidth=lw)
    ax.plot(line2, label=label2, linewidth=lw)
    ax.set_ylabel('price [CAD]', fontsize=14)
    ax.set_title(title, fontsize=16)
    ax.legend(loc='best', fontsize=16);
```

```
line_plot(train[target_col], test[target_col], 'training', 'test', title='')
```



3. Normalization

The first step we will take to our data is to normalize its values. The goal of normalization is to change the values of numeric columns in the data set to a common scale, without distorting differences in the ranges of values.

Among the best practices for training a Neural Network is to normalize your data to obtain a mean close to 0. Normalizing the data generally speeds up learning and leads to faster convergence. Also, the (logistic) sigmoid function is hardly ever used anymore as an activation function in hidden layers of Neural Networks, because the tanh function (among others) seems to be strictly superior.

While this might not be immediately evident, there are very similar reasons for why this is the case. The tanh function is quite similar to the logistic sigmoid. The main difference, however, is that the tanh function outputs results between -1 and 1, while the sigmoid function outputs values that are between 0 and 1 — therefore they are always positive.

In [35]:

```
def normalise_zero_base(df):
    return df / df.iloc[0] - 1

def normalise_min_max(df):
    return (df - df.min()) / (data.max() - df.min())
```

In [36]:

```
def extract_window_data(df, window_len=5, zero_base=True):
    window_data = []
    for idx in range(len(df) - window_len):
        tmp = df[idx: (idx + window_len)].copy()
        if zero_base:
            tmp = normalise_zero_base(tmp)
        window_data.append(tmp.values)
    return np.array(window_data)
```

```
array([[1.49732345e-02, 1.29013200e-02, 1.49400698e-02, 1.44534769e-02,
        3.35749244e-04],
       [1.45066780e-02, 1.23321258e-02, 1.28489753e-02, 1.27508263e-02,
        6.33453324e-04],
       [1.28093283e-02, 1.08719155e-02, 1.13293978e-02, 1.12164013e-02,
        7.09650970e-04],
       ...,
       [3.70008086e-01, 3.67365217e-01, 3.78051927e-01, 3.74990337e-01,
        4.97548412e-01],
       [3.75325771e-01, 3.64390763e-01, 3.77862744e-01, 3.68279031e-01,
        5.07057851e-01],
       [3.68805505e-01, 3.58377151e-01, 3.72197021e-01, 3.63134123e-01,
        4.69226533e-01]])
```

4.Predict the price of cryptocurrency using LSTM neural network (deep learning)

This is the model-building stage. Finding the right model is an art, and it will take several tweaks and attempts to find the right layers and hyperparameters for each one.

The model building is quite simple and standard for this type of problem.

Training this model is something you can do even without a GPU, the amount of data is very low and the network architecture is very simple. When it comes to more advanced models with more granular information, it can take hours or days to train.

In [39]:

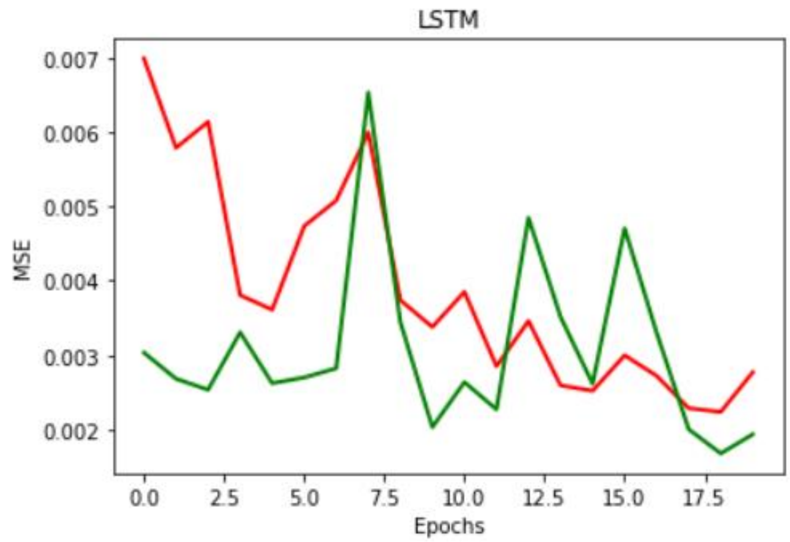
```
np.random.seed(42)
window_len = 5
test_size = 0.2
zero_base = True
lstm_neurons = 100
epochs = 20
batch_size = 32
loss = 'mse'
dropout = 0.2
optimizer = 'adam'
```

In [41]:

```
model = build_lstm_model(
    X_train, output_size=1, neurons=lstm_neurons, dropout=dropout, loss=loss,
    optimizer=optimizer)
history = model.fit(
    X_train, y_train, validation_data=(X_test, y_test), epochs=epochs, batch_size=batch_size, verbose=1, shuffle=True)
```

```
Epoch 1/20
13/13 [==============================] - 2s 39ms/step - loss: 0.0070 - val_loss: 0.0030
Epoch 2/20
13/13 [==============================] - 0s 10ms/step - loss: 0.0058 - val_loss: 0.0027
Epoch 3/20
13/13 [==============================] - 0s 9ms/step - loss: 0.0061 - val_loss: 0.0025
Epoch 4/20
13/13 [==============================] - 0s 8ms/step - loss: 0.0038 - val_loss: 0.0033
Epoch 5/20
13/13 [==============================] - 0s 8ms/step - loss: 0.0036 - val_loss: 0.0026
Epoch 6/20
13/13 [==============================] - 0s 9ms/step - loss: 0.0047 - val_loss: 0.0027
Epoch 7/20
13/13 [==============================] - 0s 8ms/step - loss: 0.0051 - val_loss: 0.0028
Epoch 8/20
13/13 [==============================] - 0s 9ms/step - loss: 0.0060 - val_loss: 0.0065
Epoch 9/20
13/13 [==============================] - 0s 8ms/step - loss: 0.0037 - val_loss: 0.0034
Epoch 10/20
13/13 [==============================] - 0s 8ms/step - loss: 0.0034 - val_loss: 0.0020
Epoch 11/20
13/13 [==============================] - 0s 8ms/step - loss: 0.0039 - val_loss: 0.0026
Epoch 12/20
13/13 [==============================] - 0s 8ms/step - loss: 0.0029 - val_loss: 0.0023
Epoch 13/20
13/13 [==============================] - 0s 8ms/step - loss: 0.0035 - val_loss: 0.0048
Epoch 14/20
13/13 [==============================] - 0s 8ms/step - loss: 0.0026 - val_loss: 0.0035
Epoch 15/20
13/13 [==============================] - 0s 8ms/step - loss: 0.0025 - val_loss: 0.0026
Epoch 16/20
13/13 [==============================] - 0s 8ms/step - loss: 0.0030 - val_loss: 0.0047
Epoch 17/20
13/13 [==============================] - 0s 10ms/step - loss: 0.0027 - val_loss: 0.0033
Epoch 18/20
13/13 [==============================] - 0s 10ms/step - loss: 0.0023 - val_loss: 0.0020
Epoch 19/20
13/13 [==============================] - 0s 8ms/step - loss: 0.0022 - val_loss: 0.0017
Epoch 20/20
13/13 [==============================] - 0s 8ms/step - loss: 0.0028 - val_loss: 0.0019
```

```python
import matplotlib.pyplot as plt
plt.plot(history.history['loss'],'r',linewidth=2, label='Train loss')
plt.plot(history.history['val_loss'], 'g',linewidth=2, label='Validation loss')
plt.title('LSTM')
plt.xlabel('Epochs')
plt.ylabel('MSE')
plt.show()
```



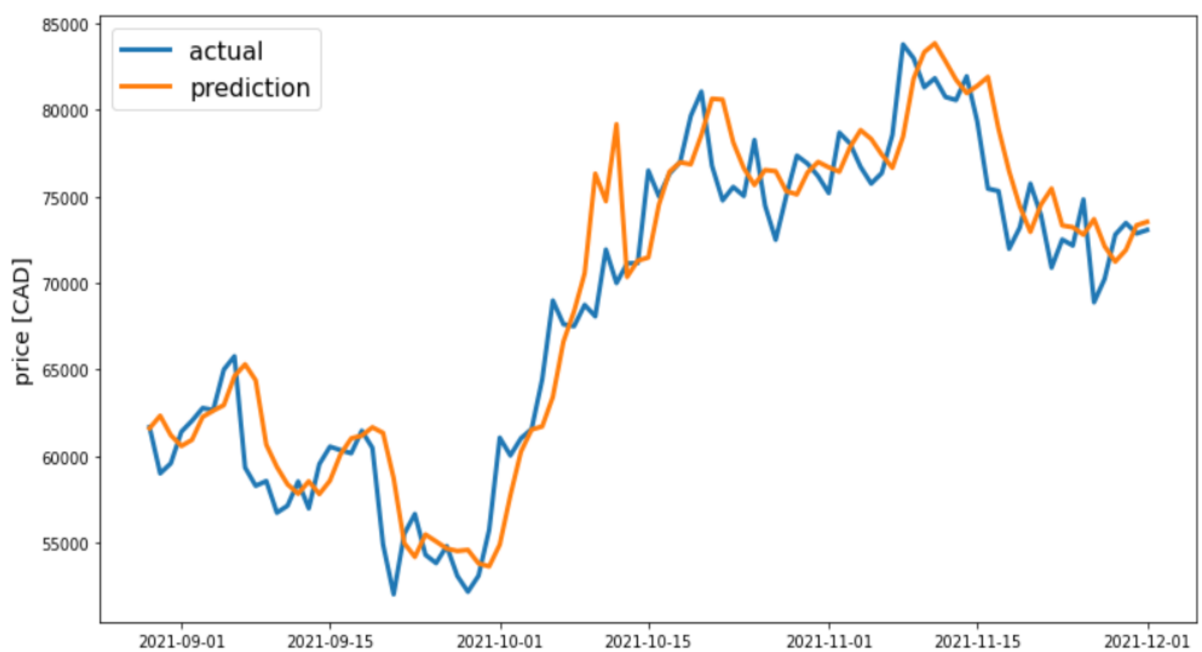
5.Test Data

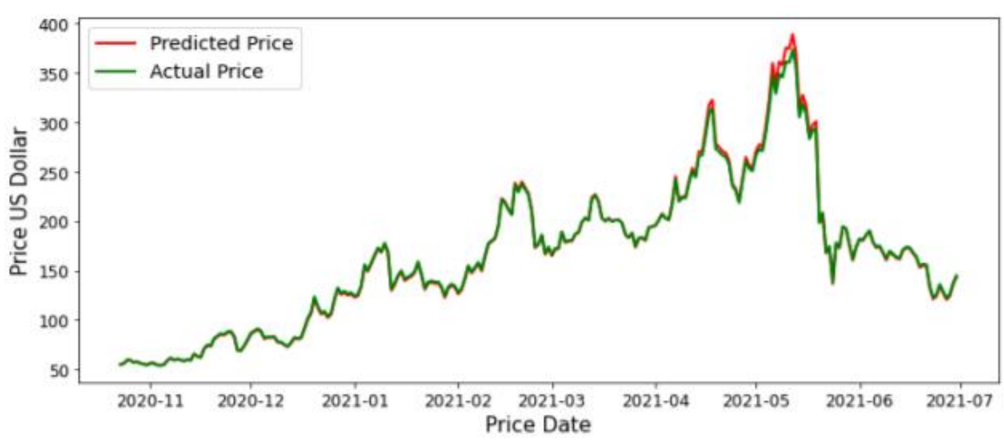
In [43]:
```
targets = test[target_col][window_len:]
preds = model.predict(X_test).squeeze()
mean_absolute_error(preds, y_test)
```

Out[43]:
```
0.03254128527601039
```

	Open	High	Low	Close	Volume
0	9259.783203	9377.486328	9249.587891	9324.717773	2.124268e+10
1	9324.787109	9379.806641	9141.251953	9235.354492	2.113222e+10
2	9235.607422	9505.051758	9191.485352	9412.612305	2.617026e+10
3	9413.004883	9457.417969	9256.931641	9342.527344	2.619861e+10
4	9340.864258	9423.237305	9305.909180	9360.879883	2.313390e+10

```
In [46]:
preds = test[target_col].values[:-window_len] * (preds + 1)
preds = pd.Series(index=targets.index, data=preds)
line_plot(targets, preds, 'actual', 'prediction', lw=3)
```





Actual and predicted price of LTC using the LSTM model.

6.Mean Absolute Error

It measures the average magnitude of the errors in a set of predictions, without considering their direction. It's the average over the test sample of the absolute differences between actual and predicted observations where all individual differences have equal weight.

$$\mathrm{MAE} = \frac{1}{n}\sum_{j=1}^{n}|y_j - \hat{y}_j|$$

n [43]:

```
targets = test[target_col][window_len:]
preds = model.predict(X_test).squeeze()
mean_absolute_error(preds, y_test)
```

ut[43]: 0.03254128527601039

n [44]:

```
from sklearn.metrics import mean_squared_error
MAE=mean_squared_error(preds, y_test)
MAE
```

ut[44]: 0.0019352016628199065

7. Sending the Mail:-

Connect to an SMTP server

The next step after gathering the price is to connect to an email server to be able to send messages to and from some email address. Luckily for us Google provides a free SMTP server we can use.

Send the cryptocurrency price as text to your email address

Last but not least is to take the price of the cryptocurrency and mail it to your email address using the SMTP server after the price passes some arbitrary threshold.

```
#Import the libraries
from bs4 import BeautifulSoup
import requests
import time
import smtplib
import ssl
from email.mime.text import MIMEText as MT
from email.mime.multipart import MIMEMultipart as MM
```

Store the email addresses and email password into variables to be able to use them later.

```
In [ ]: #Store the email addresses for the receiver, and the sender and store the senders password
        receiver = '<RECEIVER_EMAIL_ADDRESS>'
        sender = '<SENDER_EMAIL_ADDRESS>'
        sender_password = '<SENDER_PASSWORD>'
```

Next, create a function to send emails, by connecting to Googles SMTP server. If this function doesn't work for you, then you may need to configure your email client

In [ ]:

```python
#Create a function to send emails
def send_email(sender, receiver, sender_password, text_price):
  #Create a MIMEMutltipart Object
  msg = MM()
  msg['Subject'] = "New Crypto Price Alert !"
  msg['From']= sender
  msg['To']= receiver
#Create the HTML for the message
  HTML = """
    <html>
      <body>
        <h1>New Crypto Price Alert !</h1>
        <h2>"""+text_price+"""
        </h2>
      </body>
    </html>
    """
  #Create a html MIMEText Object
  MTObj = MT(HTML, "html")
  #Attach the MIMEText Object
  msg.attach(MTObj)
```

In [ ]:

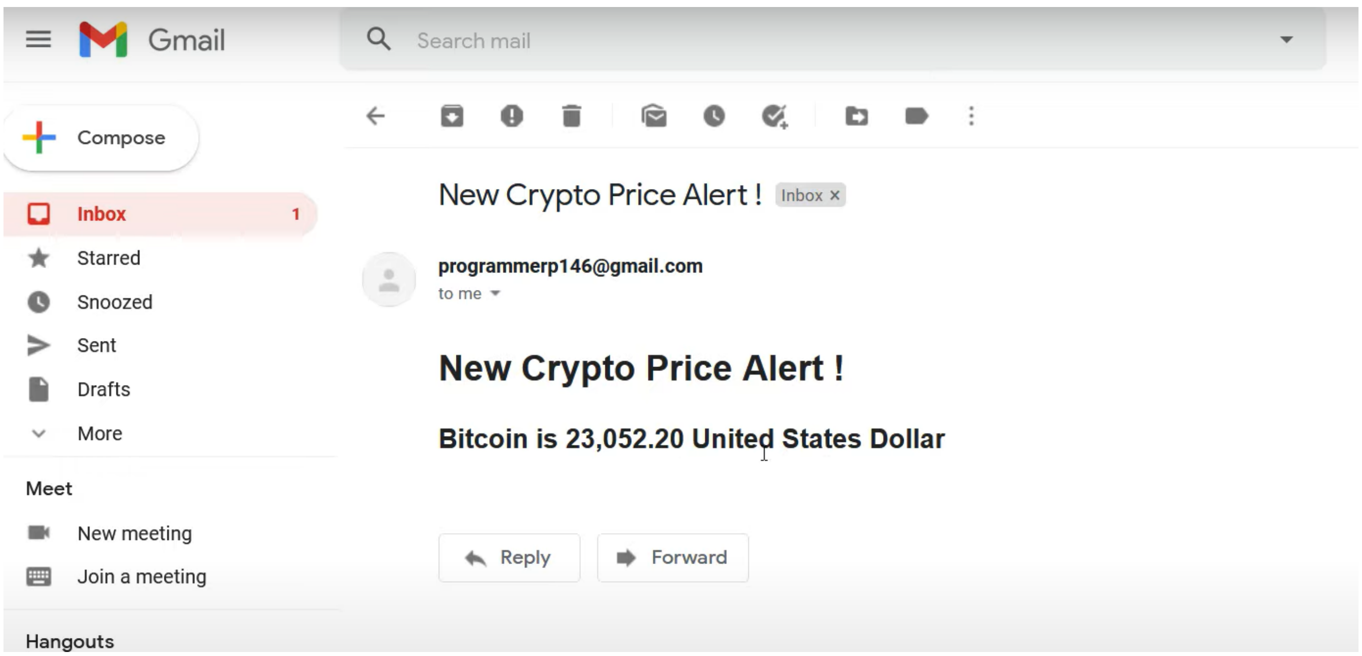
```python
#Create the secure socket layer (SSL) context object
  SSL_context = ssl.create_default_context()
  #Create the secure Simple Mail Transfer Protocol (SMTP) connection
  server = smtplib.SMTP_SSL(host="smtp.gmail.com", port=465, context=SSL_context)
  #Login to the email
  server.login(sender, sender_password)
  #Send the email
  server.sendmail(sender, receiver, msg.as_string())
```

In [ ]:

```python
#Create a function to send the alert
def send_alert():
  last_price = -1
  #Create an infinite loop to continuously send/show the price
  while True:
    #Choose the cryptocurrency/coin
    coin = 'bitcoin'
    #Get the price of the cryptocurrency
    price = get_crypto_price(coin)
    #Check if the price has changed
    if price != last_price:
      print(coin.capitalize()+' price: ', price)
      price_text = coin.capitalize()+' is '+price
      send_email(sender, receiver, sender_password, price_text)
      last_price = price #Update the last price
      time.sleep(3)
```

```
In [ ]:
#Send the alert
send_alert()
```

And at the last you will receive a email regarding the alert of the price



Module Description:-

This model use some of the python libraries:-

1.Numpy:-

NumPy is a general-purpose array-processing package. It provides a high-performance multidimensional array object, and tools for working with these arrays.
It is the fundamental package for scientific computing with Python. It contains various features including these important ones:

- A powerful N-dimensional array object
- Sophisticated (broadcasting) functions
- Tools for integrating C/C++ and Fortran code
- Useful linear algebra, Fourier transform, and random number capabilities

Besides its obvious scientific uses, NumPy can also be used as an efficient multi-dimensional container of generic data.
Arbitrary data-types can be defined using Numpy which allows NumPy to seamlessly and speedily integrate with a wide variety of databases.

2.Pandas:-

Pandas is an open-source library that is made mainly for working with relational or labeled data both easily and intuitively. It provides various data structures and operations for manipulating numerical data and time series. This library is built on top of the NumPy library. Pandas is fast and it has high performance & productivity for users.

Pandas were initially developed by Wes McKinney in 2008 while he was working at AQR Capital Management. He convinced the AQR to allow him to open source the Pandas. Another AQR employee, Chang She, joined as the second major contributor to the library in 2012. Over time many versions of pandas have been released. The latest version of the pandas is 1.3.4

Advantages

- Fast and efficient for manipulating and analyzing data.
- Data from different file objects can be loaded.
- Easy handling of missing data (represented as NaN) in floating point as well as non-floating point data
- Size mutability: columns can be inserted and deleted from DataFrame and higher dimensional objects
- Data set merging and joining.
- Flexible reshaping and pivoting of data sets
- Provides time-series functionality.
- Powerful group by functionality for performing split-apply-combine operations on data sets.

3.Matplotlib:-

Matplotlib is an amazing visualization library in Python for 2D plots of arrays. Matplotlib is a multi-platform data visualization library built on NumPy arrays and designed to work with the broader SciPy stack. It was introduced by John Hunter in the year 2002.

One of the greatest benefits of visualization is that it allows us visual access to huge amounts of data in easily digestible visuals. Matplotlib consists of several plots like line, bar, scatter, histogram etc.

4.Seaborn:-

Seaborn is an amazing visualization library for statistical graphics plotting in Python. It provides beautiful default styles and color palettes to make statistical plots more attractive. It is built on the top of matplotlib library and also closely integrated to the data structures from pandas.

Seaborn aims to make visualization the central part of exploring and understanding data. It provides dataset-oriented APIs, so that we can switch between different visual representations for same variables for better understanding of dataset.

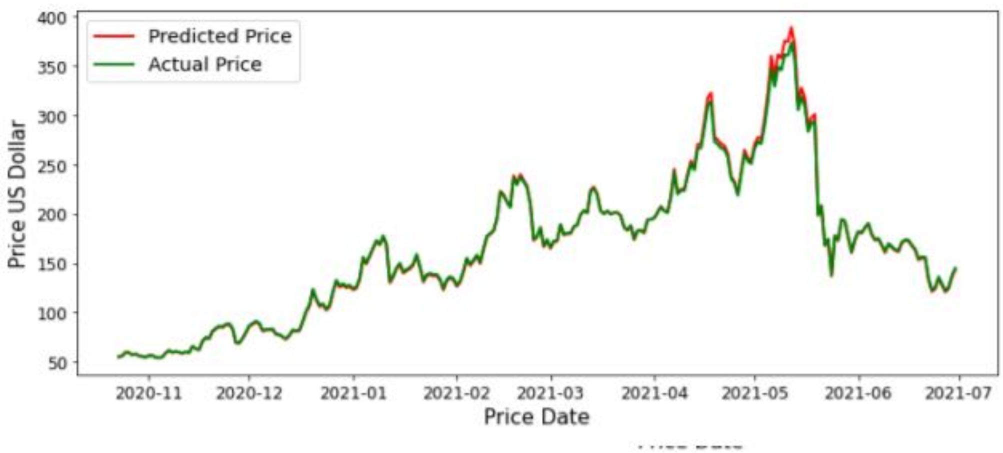
5.SMTP

An SMTP instance encapsulates an SMTP connection. It has methods that support a full repertoire of SMTP and ESMTP operations. If the optional host and port parameters are given, the SMTP connect() method is called with those parameters during initialization. If specified, local_hostname is used as the FQDN of the local host in the HELO/EHLO command. Otherwise, the local hostname is found using socket.getfqdn(). If the connect() call returns anything other than a success code, an SMTPConnectError is raised. The optional timeout parameter specifies a timeout in seconds for blocking operations like the connection attempt (if not specified, the global default timeout setting will be used). If the timeout expires, TimeoutError is raised. The optional source_address parameter allows binding to some specific source address in a machine with multiple network interfaces, and/or to some specific source TCP port. It takes a 2-tuple (host, port), for the socket to bind to as its source address before connecting. If omitted (or if host or port are '' and/or 0 respectively) the OS default behavior will be used.

Results:-

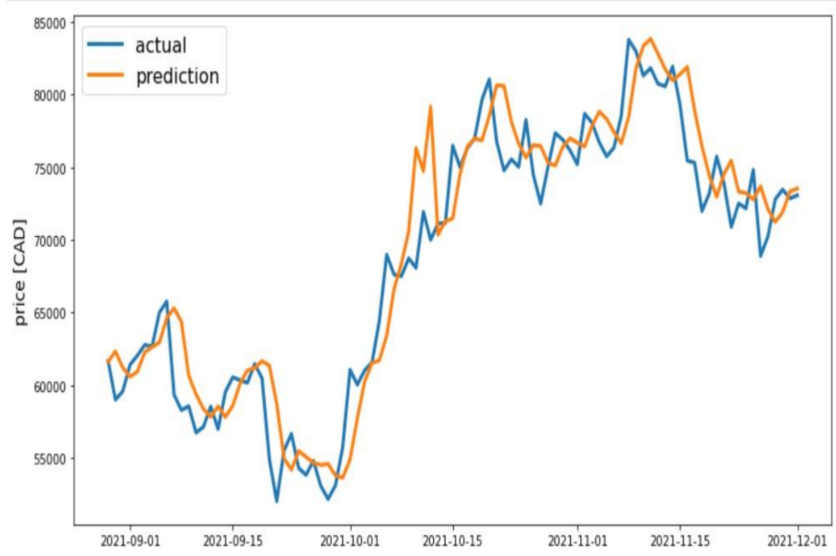
This paper also proposes a model to monitor stock available from internet offerings or dynamic HTML documents at the internet. This can cope with public data available at the internet on a single or more than one web sites and from a group or bunch a couple of web services because the information assets. The model permits investors to apply these sources to create personalised stock tracking criteria after which acquire updated notifications from the device primarily based on the required standards., E mail notification messages can be generated and sent to the investor notifying him/her whenever the standards are met. some of applications associated with inventory investments can benefit form this machine to provide actual-time useful information to the buyers.

So basically we predict the value of bitcoin as depicted in the graph below and as you can see it predicted the downfall in the price of bitcoin in the starting of the month of ‘october’ which do happen and increase in the mid of November too sometimes it do get fail but it was really helpful on predicting the price for long periods so that you can decide to hold it for long period of time or not.



Actual and predicted price of LTC using the LSTM model.

```
In [46]:
preds = test[target_col].values[:-window_len] * (preds + 1)
preds = pd.Series(index=targets.index, data=preds)
line_plot(targets, preds, 'actual', 'prediction', lw=3)
```



And you can also get the mail regarding the price of coin which is beneficial for you to track the time in your buy day to day routine

CONCLUSION

Overall, predicting price-related fluctuations is difficult given the relative strength of the market. Add to that the fact that prices are more dependent on future prospects than historical data. However, using deep neural networks has given us a better understanding of Bitcoin, as well as LSTM architecture. Ongoing work, including the use of hyperparameter tuning, to obtain a more accurate network configuration. Also, other factors can be considered (although from our experiment with Bitcoin, many factors did not always lead to better results). Microeconomic features can be incorporated into the model for a better guessing effect. However, perhaps 6 Conclusions Overall, predicting price-related fluctuations is difficult given the magnitude of the potential market forces. In addition, the fact that prices are largely based on future prospects rather than historical data. However, using deep neural networks has given us a better understanding of Bitcoin, as well as LSTM architecture. Ongoing work, including the use of hyperparameter tuning, to obtain a more accurate network configuration. Also, other factors can be considered (although from our experiment with Bitcoin, many factors did not always lead to better results). Microeconomic features can be incorporated into the model for a better guessing effect. However, perhaps the data we have collected for Bitcoin, although collected over the years, may be interesting, producing only historical translations in the last few years. In addition, a successful transition from peer to peer performance continues and transforms the status of payment services. Although all doubts seem to have been resolved, it may be time to do something about it. We think it’s hard to give a mature idea about Bitcoin for the future.

In future work, we will investigate other factors that might affect the prices of the cryptocurrency market, and we will focus on the effect that social media in general and tweets in particular can have on the price and trading volume of cryptocurrencies by analyzing tweets using natural language processing techniques and sentiment analysis.

References

- Jang, H. & Lee, J. An Empirical Study on Modelling and Prediction of Bitcoin Prices with Bayesian Neural Networks based on Blockchain Information. *An Empirical Study on Modelling and Prediction of Bitcoin Prices with Bayesian Neural Networks based on Blockchain Information* .

- Kathyayini, R. S., Jyothi, D. G. & Crypto, ". Currency Price Prediction using Machine Learning". *International Journal of Advanced Research in Computer and Communication Engineering* .

- Struga, K. & Qirici, O. Bitcoin Price Prediction with Neural Networks. *Bitcoin Price Prediction with Neural Networks* . Retrieved from http://ceur-ws.org/Vol-2280/paper-06.pdf

- R. Beck, M. Avital, M. Rossi, J.B. Thatcher
Blockchain technology in business and information systems research
Inf Syst Eng, 59 (2017), pp. 381-384, 10.1007/s12599-017-0505-1

- ◦