# BITCOIN PRICE PREDICTION

Submitted in partial fulfillment of the requirements
of the degree of

**T. E. Computer Engineering**

By

**Preeti Suvarna 21 182113**
**Jash Tailor 23 182115**
**Abraham Thothiyil 25 182115**

Guide (s):

**Varsha Shrivastava**
Assistant Professor

Department of Computer Engineering
St. Francis Institute of Technology
(Engineering College)

University of Mumbai
2020-2021

# CERTIFICATE

This is to certify that the project entitled **"BITCOIN PRICE PREDICTION"** is a bonafide work of **Preeti Suvarna (21), Jash Tailor (23) and Abraham Thothiyil (25)** submitted to the University of Mumbai in partial fulfillment of the requirement for the award of the degree of T.E. in Computer Engineering

**(Name and sign)**
**Guide**

**(Name and sign)**
**Head of Department**

# Project Report Approval for T.E.

This project report entitled (***Title***) by (***Author Name***) is approved for the degree of ***T.E. in Computer Engineering.***

Examiners

1.----------

2.----------

Date:

Place:

# Declaration

I declare that this written submission represents my ideas in my own words and where others' ideas or words have been included, I have adequately cited and referenced the original sources. I also declare that I have adhered to all principles of academic honesty and integrity and have not misrepresented or fabricated or falsified any idea/data/fact/source in my submission. I understand that any violation of the above will be cause for disciplinary action by the Institute and can also evoke penal action from the sources which have thus not been properly cited or from whom proper permission has not been taken when needed.

---

(Signature)

---

(Name of students and Roll No.)

Date:

# Abstract

Machine learning plays a major role from past years in object detection, self-driving vehicles, speech recognition, movie and music recommendation and medical diagnosis. Present machine learning algorithm helps us in enhancing security alerts, ensuring public safety and improve medical enhancements. The objective of this project was to design a system that analyzes the public sentiment regarding Bitcoin by classifying posts from Twitter and Reddit. Furthermore, apart from getting to know the public sentiment, time series forecasting was also deployed to see the trends and patterns in the price of Bitcoin. Our project fetches data from Twitter and Reddit using their respective APIs. The collected data is then passed through a labeler that labels the various posts as Positive, Negative, or Neutral. This helps us in figuring out the general sentiment of the public on Bitcoin. The time-series data is taken from Yahoo finance.

We used the Time-series approach in order to predict the future values of bitcoin prices to know whether there will be a rise or fall in the prices. Time series is basically a statistical technique that deals with the analysis of trends. We use the past and current trends of bitcoin prices to make predictions about the future. In addition to this, users may view the current sentiment of people on Bitcoin based on their Tweets and Reddit posts and make a well-informed decision on whether to invest or not.

# Contents

# List of Figures

# List of Abbreviations

| Sr. No. | Abbreviation | Expanded form |
|---|---|---|
| I | FR | Functional Requirement |
| ii | QR | Non-Functional Requirement |
| iii | UI | User Interface |

# Chapter 1

# Introduction

On 9 January 2009, a person with the pseudonym Satoshi Nakamoto released the first Bitcoin client version. Shortly before that, on 31 October 2008, the paper that first introduced Bitcoin was published.1 “Bitcoin: A Peer-to-Peer Electronic Cash System” introduced on only nine pages a revolutionary electronic alternative to cash. Until today, the inventor Satoshi has not revealed his true identity. Currently, the core developer team of Bitcoin maintains the open-source code on GitHub.

Bitcoin and other cryptocurrencies do not differ much compared to fiat currencies on the first view. A bitcoin can be split into smaller subunits, called Satoshis, and similar to regular money, it can be used to buy services or goods from retailers who accept Bitcoin. However, several essential features distinguish it from regular currencies. The three essential components of Bitcoin are Wallets, the Peer-to-Peer (P2P) network, and most essentially the blockchain.

A process known as mining broadcasts all transactions to the P2P network. Miners attempt to create hashes that match specific criteria with an enormous computational effort. The loan they receive is a block reward of currently a little more than 12.5 BTC. Furthermore, they can charge transaction fees for the transactions which they will broadcast to the network as described previously. In addition to creating bitcoins through mining, it is also possible to purchase bitcoins at currency exchange markets such as Bitstamp, Coinbase, or Kraken. These platforms offer the opportunity to buy bitcoins in exchange for other currencies such as Dollars or euros. Depending on the service fees, the prices slightly differ from marketplace to marketplace. Similar to any market, the currency exchange markets bring together buyers who want to make a deal with sellers and vice versa.

## 1.1 Description

The objective of this project was to design a system that analyzes the public sentiment regarding Bitcoin by classifying posts from Twitter and Reddit. Furthermore, apart from getting to know the public sentiment, time series forecasting was also deployed to see the trends and patterns in the price of Bitcoin. Our project fetches data from Twitter and Reddit using their respective APIs.

The collected data is then passed through a labeler that labels the various posts as Positive, Negative, or Neutral. This helps us in figuring out the general sentiment of the public on Bitcoin. The time-series data is taken from Yahoo finance.

## 1.2 Problem Formulation

Cryptocurrency is a new type of digital currency that utilizes blockchain technology and cryptography functions. Bitcoin is one of the first and the most well-known cryptocurrency. Bitcoin has hit an all-time high in the past few years.

Off lately, cryptocurrency has sparked the interest of many. Our website will guide them on current trends wherein they can enter the dates for which they would like to know the price of bitcoin, and our website will present them with the relevant data. Furthermore, people would also like to be informed about the view of the public before investing. Therefore, we used tweets and Reddit posts to analyze public sentiment to help make a well-educated decision.

## 1.3 Motivation

Interest in cryptocurrencies has seen a meteoric rise in the last couple of years because of their decentralized nature. People of all ages and all walks of life have been keenly interested in this new technology and it has been touted as Fourth Industrial Revolution. One of the primary reasons for the high level of interest in cryptocurrencies is that there are no regional limitations.

By saving the processing time and fees, cryptocurrencies create a faster and cheaper method for international money transfers.

Looking at the immense demand and the underserved market needs for good and reliable forecasting of cryptocurrencies we decided to go with this topic for our project.

## 1.4 Proposed Solution

We used the Time-series approach in order to predict the future values of bitcoin prices to know whether there will be a rise or fall in the prices. Time series is basically a statistical technique that deals with the analysis of trends. We use the past and current trends of bitcoin prices to make predictions about the future. In addition to this, users may view the current sentiment of people on Bitcoin based on their Tweets and Reddit posts and make a well-informed decision on whether to invest or not.

## 1.5 Scope of the Project

This project has the potential to succinctly explain the movement of the price of Bitcoin by showing them the sentiment of the public and the time-series forecasting done using various statistical techniques. This can be used from a 15-year-old kid to a trader at a leading hedge fund.

# Chapter 2

## Review of Literature

The Investments & Markets has had a significant impact on all aspects of our society. As today society relies more and more on the Technology, the dependability of accurate prediction and recommending applications by using the technology has become increasingly important. To make these applications more dependable, for the past decade researchers have proposed various techniques to implement Machine Learning. Our literature search for related studies retrieved 3 papers in the area of Machine Learning and AI, which have appeared between 2000 and 2013.

Papers like 'KryptoOracle: A Real-Time Cryptocurrency Price Prediction Platform Using Twitter Sentiments' written by Mohapatra, Ahmed and Alencar [1] were extremely useful in helping us along the way. A paper is written by Karasu, Altan et al [2] titled 'Prediction of Bitcoin prices with machine learning methods using time series data' helped us in figuring out which algorithms to use for time-series prediction. 'Bitcoin Price Prediction Using Deep Learning and Real-Time Deployment' written by Mahendra, Madan et al [3] was helpful in guiding on how to use these datasets correctly and to their full potential.

Other than these papers, Bitcoin's whitepaper was also crucial in helping us to understand it as a concept and technology.

# Chapter 3

# System Analysis

## 3.1 Functional Requirements:

**FR1: USER INTERFACE:** The user interface will be a website. The user has to enter all the attributes correctly and in the required format.

**FR2: PROPER FORECASTING:** The system has to properly predict the price of the house according to the input given by the user.

**FR3: SENTIMENT ANALYSIS:** According to tweets and Reddit posts, the system will display the sentiment of people on Bitcoin

**FR3: DATABASE:** Dataset should contain real-time data consisting of a large number of entities so that it will increase the accuracy of the predicted price and suggest a better property.

## 3.2 Non-Functional Requirements:

**QR1: Platform Independent:**

The application would be platform independent if all the requirements are installed in the device.

**QR2: Performance**:

The application should have better accuracy and should provide the information in less time.

**QR3: Capacity:**

The capacity of the storage should be high so that large amount of data can be stored in order to train the model.

## Performance Requirements

1. The system must process the number of transactions based on the following calculation method.
2. Once an error has occurred, the System should detect and display an error message in no more than 5 seconds.

## Safety Requirements

If there is extensive damage to a wide portion of the database due to catastrophic failure, such as a disk crash, the recovery method restores a past copy of the database that was backed up to archival storage (typically tape) and reconstructs a more current state by reapplying or redoing the operations of committed transactions from the backed up log, up to the time of failure

## Security Requirements

- Updates shall only be made by authorized developers.
- The Administrator of the system is the only one responsible for the change of all the system data.

## 3.3 Software Quality Attributes

**Availability-1**: Once the website is launched it is available to the World Wide Web.

**Availability -2**: Provided that the web server and web hosting is up, the required features can also be accessed at ease.

**Installability-1**: The system does not require installation.

**Maintainability-1**: For the Updates and Maintenance of the system, the source codes for the system are well documented.

**Accurateness-1**: The system provides accurate information regarding movies shown in a theater, availability of tickets and seats.

**Testability-1**: The system will not be available if some errors occur for easier debugging.

**Usability-1**: The system is well designed to allow easy usage and navigation.

**Usability-2**: The Graphical User Interface of the System provides buttons and menus that can be easily understood by the user.

**Flexibility-1**: The system can be access in internet browsers such as Internet Explorer, Google Chrome, Mozilla Firefox and Safari.

## 3.4 Specific. Requirements

**Hardware Requirements:**

CPU --Intel Core i5

Hard Disk Space -- 1TB

Display – Any Generic 15.6 inch Display

Memory – 4gb

Other Devices – Laptop

**Software Requirements:**

Front End – Streamlit

Back End – Github

Languages -- Python

Operating System -- WINDOWS 10

Web Server – Streamlit

# Chapter 4

# Analysis Modeling

## 4.1 Activity Diagram

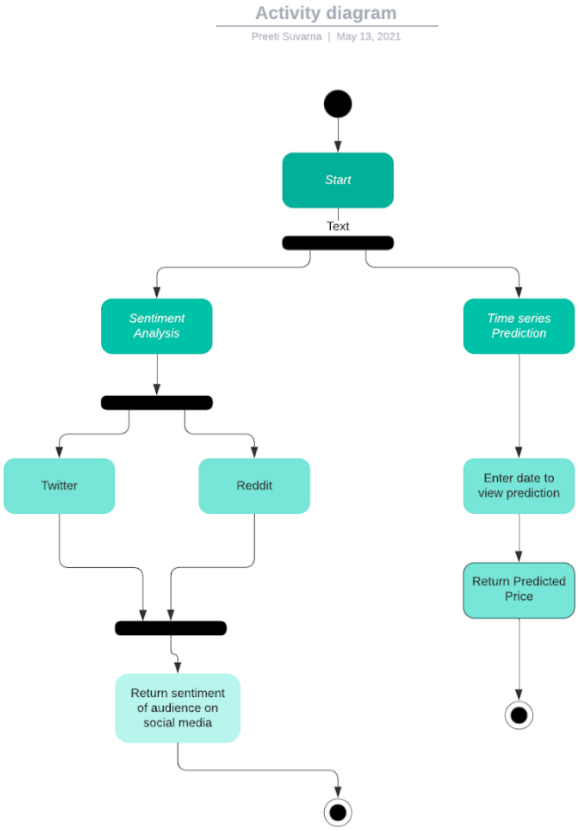



Figure 1: Activity Diagram

## 4.2 Functional Modeling

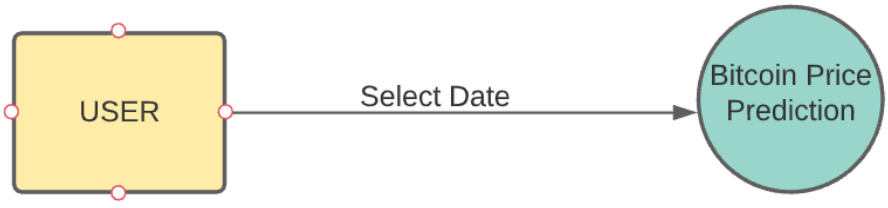



Figure 2:Level 0 User

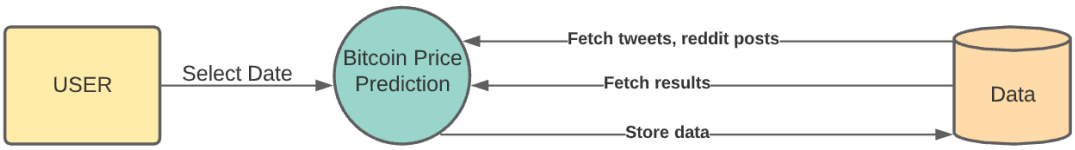



Figure 3:Level 1 User

# Chapter 5

# Design

## 5.1 Architectural Design

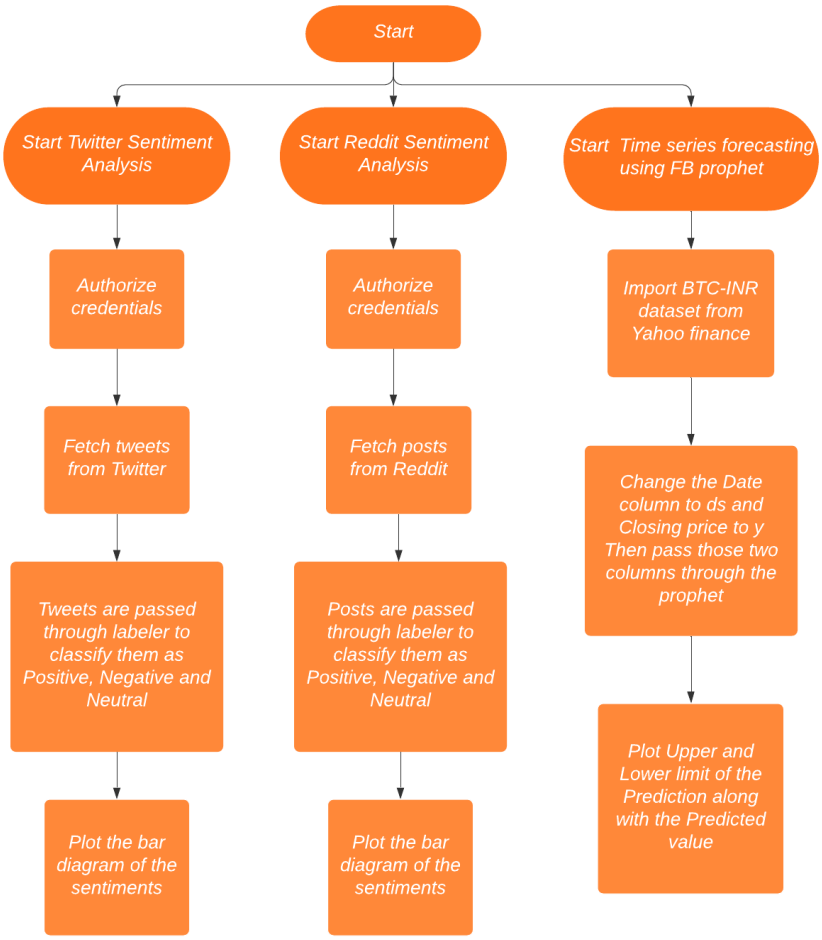



Figure 4:Architectural Design

**Phase 1: Collection of data**

Raw data is collected from Reddit and Twitter using their respective APIs.

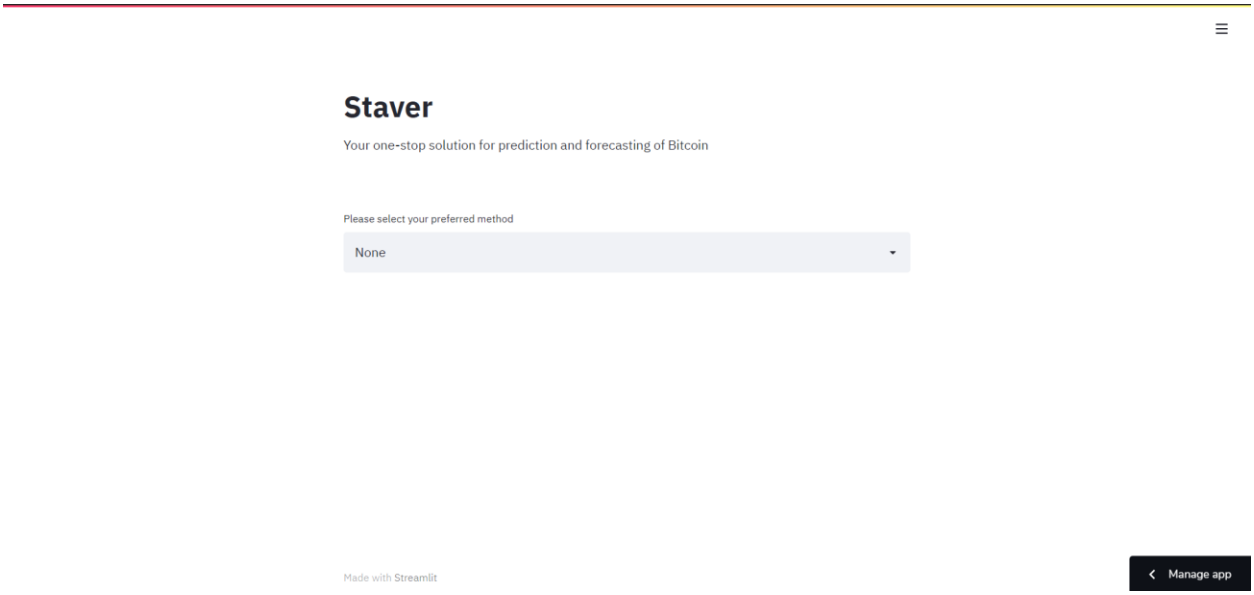
**Phase 2: Classifying based on sentiment**

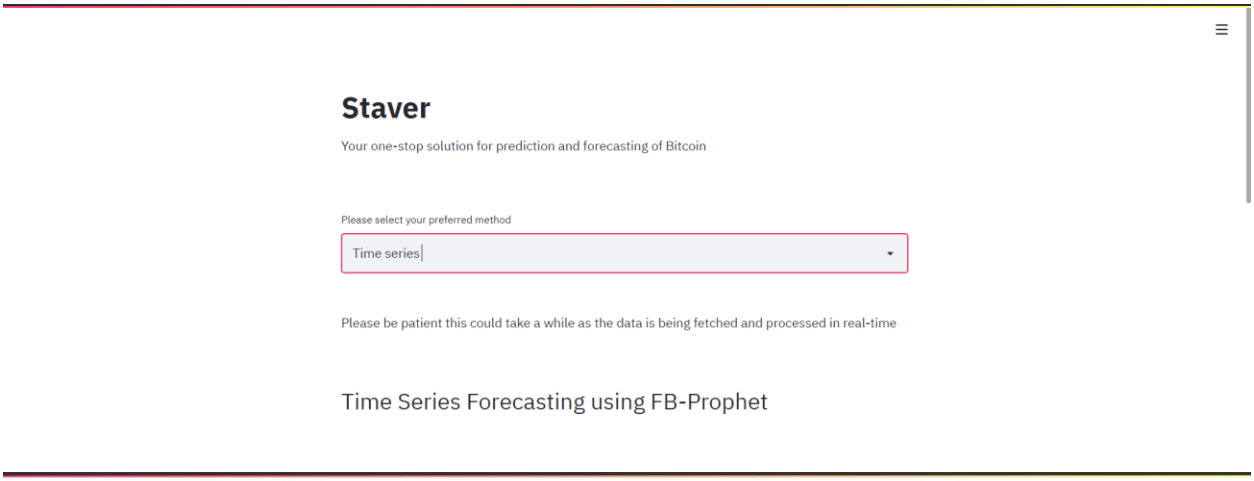
The raw pieces of texts are classified based on their sentiment.

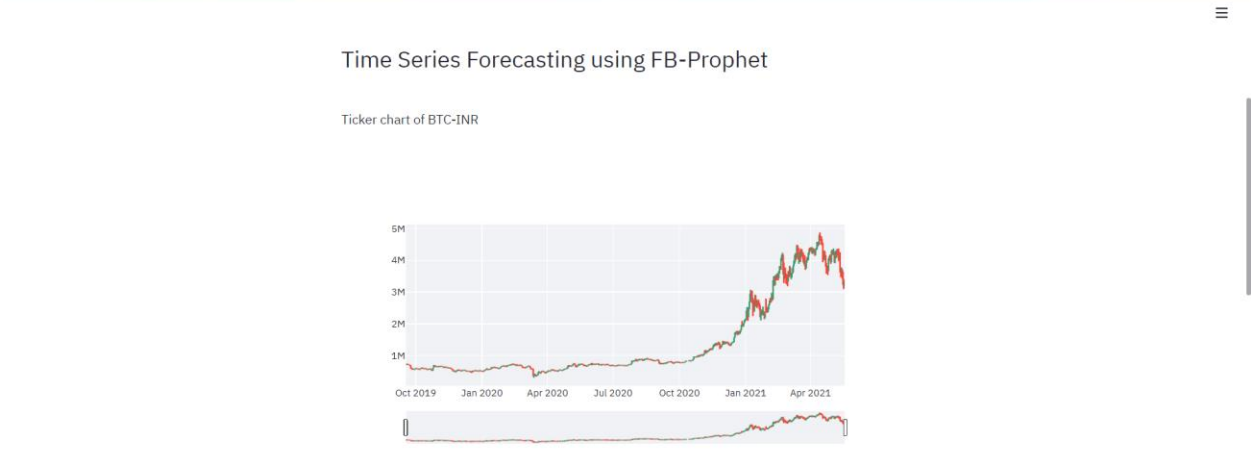
**Phase 3: Time-series forecasting**

Dynamic dataset of BTC-INY is imported from Yahoo Finance and then changes are made as mentioned in the diagram.

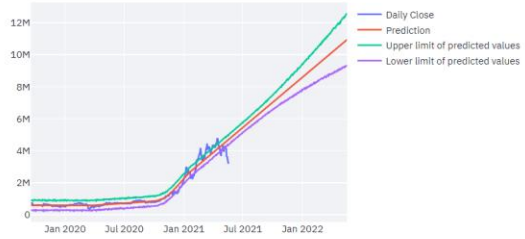
## 5.2 User Interface Design



The forecasted price of Bitcoin




Enter Date

Date
*empty*

Lower limit of Prediction
*empty*

Upper limit of Prediction
*empty*

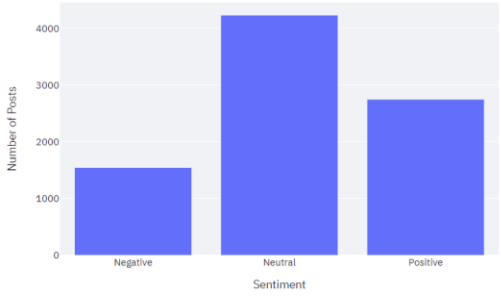
Prediction
*empty*

Please select your preferred method

Sentiment Analysis

Please be patient this could take a while as the data is being fetched and processed in real-time

## Reddit Sentiment Analysis




Manage app

## Sentiment is Positive

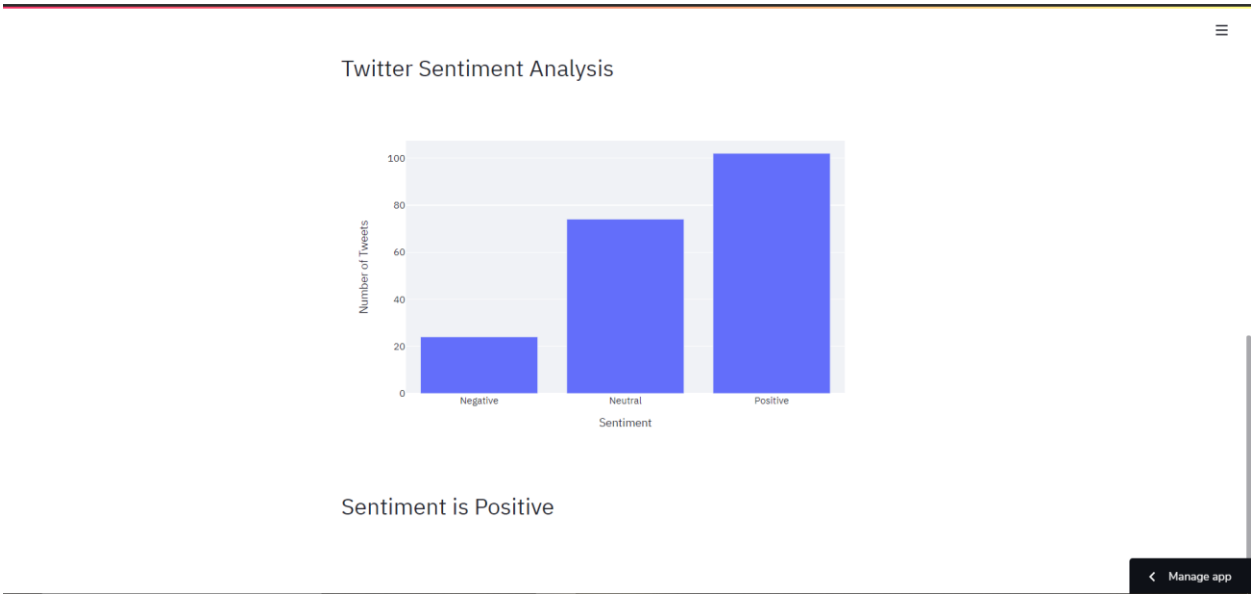



Figure 5:User Interface Design

# Chapter 6

# Implementation

## 6.1 Algorithms/Methods Used

The algorithm we used for this project was SentimentIntensityAnalyzer and FB-Prophet was used for creating the predict function for the front-end. Sentiment Analysis is the process of figuring out if a piece of text is positive, negative, or neutral. In our project, we have set out to find the sentiment of the public on cryptocurrencies and more specifically Bitcoin. Now there are two ways in which we can classify pieces of text based on their sentiment:

- **polarity-based:** here pieces of texts are classified as either positive or negative
- **valence-based:** here the intensity of the sentiment is taken into account

We have used the Natural Language Toolkit (NLTK) to process the raw data obtained from Twitter and Reddit. The NLTK library has the VADER (Valence Aware Dictionary and sEntiment Reasoner) package. VADER has a polarity-based approach to classifying pieces of text. VADER's lexicon consists of sentiment-related words where each word has a rating.

| Word | Sentiment Rating |
|---|---|
| tragedy | -3.4 |
| rejoiced | 2.0 |
| insane | -1.7 |
| disaster | -3.1 |
| great | 3.1 |

For example, the sentence "The food is good and the atmosphere is nice" has two words in the lexicon (good and nice) with ratings of 1.9 and 1.8 respectively. VADER produces four

sentiment metrics from these word ratings, which you can see below. The first three, positive, neutral, and negative, represent the proportion of the text that falls into those categories. As you can see, our example sentence was rated as 45% positive, 55% neutral, and 0% negative. The final metric, the compound score, is the sum of all of the lexicon ratings (1.9 and 1.8 in this case) which have been standardized to range between -1 and 1. In this case, our example sentence has a rating of 0.69, which is pretty strongly positive.

| Sentiment Metric | Value |
|---|---|
| Positive | 0.45 |
| Neutral | 0.55 |
| Negative | 0.00 |
| Compound | 0.69 |

The other method we have used in this project is Time-series forecasting where we have used live BTC-INR ticker data to predict the future values of Bitcoin.

Time-series forecasting involves taking models to fit historical data and using them to predict future observations. The FB-Prophet, the python library we have used in our project, analyses the trends and patterns seen previously in the data and uses them to predict future values.

## 6.2 Working of the Project

```
import yfinance as yf
import praw as pw
import tweepy as tw
import pandas as pd
import numpy as np
import nltk
nltk.download('wordnet')
nltk.download('vader_lexicon')
```

```
from nltk.sentiment.vader import SentimentIntensityAnalyzer as SIA
from nltk.stem import WordNetLemmatizer, SnowballStemmer
from nltk.stem.porter import *
import plotly.express as px
import plotly.graph_objects as go
import plotly.figure_factory as ff
import fbprophet
from fbprophet import Prophet
from fbprophet.diagnostics import cross_validation, performance_metrics
from fbprophet.plot import add_changepoints_to_plot, plot_cross_validation_metric
from datetime import date
import streamlit as st

st.write("""
# Staver
Your one-stop solution for prediction and forecasting of Bitcoin
""")

st.text('')
st.text('')
st.text('')

def time_series():

    st.text('')
    st.text('')

    st.write("""
    Please be patient this could take a while as the data is being fetched and processed in real-time
    """)

    st.text('')
```

```
st.text('')

# Time Series Forecasting using FB-Prophet

st.write("""
## Time Series Forecasting using FB-Prophet
""")

st.text('')
st.text('')

# importing the time series dataset of bitcoin prices
today = date.today()
tickerSymbol = 'BTC-INR'
tickerData = yf.Ticker(tickerSymbol)
df = tickerData.history(period='1d', start='2010-10-08', end=today)
df.reset_index(inplace=True)

st.write("""
Ticker chart of BTC-INR
""")
fig = go.Figure(data=go.Ohlc(x=df['Date'],
            open=df['Open'],
            high=df['High'],
            low=df['Low'],
            close=df['Close']))
st.plotly_chart(fig)

st.write("""
The forecasted price of Bitcoin
""")
```

```
model = Prophet()
Date = df['Date']
Close = df['Close']
df_prophet = pd.DataFrame()
df_prophet['ds'] = Date
df_prophet['y'] = Close
df_prophet.head(10)

model.fit(df_prophet)
future_dates = model.make_future_dataframe(periods=365);
prediction = model.predict(future_dates)
fig = go.Figure()
fig.add_trace(go.Scatter(x=df['Date'], y=df['Close'],
                mode='lines',
                name='Daily Close'))
fig.add_trace(go.Scatter(x=prediction['ds'], y=prediction['yhat'],
                mode='lines',
                name='Prediction'))
fig.add_trace(go.Scatter(x=prediction['ds'], y=prediction['yhat_upper'],
                mode='lines',
                name='Upper limit of predicted values'))
fig.add_trace(go.Scatter(x=prediction['ds'], y=prediction['yhat_lower'],
                mode='lines',
                name='Lower limit of predicted values'))
st.plotly_chart(fig)

df_final = pd.DataFrame()
df_final['Date'] = prediction['ds']
df_final['Lower limit of Prediction'] = prediction['yhat_lower']
df_final['Upper limit of Prediction'] = prediction['yhat_upper']
df_final['Prediction'] = prediction['yhat']
```

```
    print(df_final.head(10))

    user_input = st.text_input("Enter Date")
    a = df_final.loc[df_final['Date'] == user_input]

    st.write(a['Date'], '\n', a['Lower limit of Prediction'], '\n', a['Upper limit of Prediction'], '\n', a['Prediction'])



def sentiment_analysis():

  st.text('')
  st.text('')

  st.write("""
  Please be patient this could take a while as the data is being fetched and processed in real-time
  """)

  st.text('')
  st.text('')

  # Reddit Sentiment Analysis

  st.write("""
  ## Reddit Sentiment Analysis
  """)

  # reddit credentials
  reddit = pw.Reddit(client_id = 'UYBiraXAwH8bcw',
                     client_secret = 'RMg2VFM9ncuAwLl61YB301SBfTZkUQ',
                     user_agent = 'MyAPI/0.0.1',
                     check_for_async=False
```

```
                )

# getting posts from the subreddits
lst_reddit = []
sentiment_lst = ['Negative', 'Neutral', 'Positive']

# bitcoin subreddit
subreddit = reddit.subreddit('bitcoin')
# hot posts
for post in subreddit.hot(limit=1000):
  lst_reddit.append(post.title)
# new posts
for post in subreddit.new(limit=1000):
  lst_reddit.append(post.title)

# CryptoCurrency subreddit
subreddit = reddit.subreddit('CryptoCurrency')
# hot posts
for post in subreddit.hot(limit=1000):
  lst_reddit.append(post.title)
# new posts
for post in subreddit.new(limit=1000):
  lst_reddit.append(post.title)

# btc subreddit
subreddit = reddit.subreddit('btc')
# hot posts
for post in subreddit.hot(limit=1000):
  lst_reddit.append(post.title)
# new posts
for post in subreddit.new(limit=1000):
  lst_reddit.append(post.title)
```

```
# Crypto_General subreddit
subreddit = reddit.subreddit('Crypto_General')
# hot posts
for post in subreddit.hot(limit=1000):
  lst_reddit.append(post.title)
# new posts
for post in subreddit.new(limit=1000):
  lst_reddit.append(post.title)

# Coinbase subreddit
subreddit = reddit.subreddit('Coinbase')
# hot posts
for post in subreddit.hot(limit=1000):
  lst_reddit.append(post.title)
# new posts
for post in subreddit.new(limit=1000):
  lst_reddit.append(post.title)

# Binance subreddit
subreddit = reddit.subreddit('Binance')
# hot posts
for post in subreddit.hot(limit=1000):
  lst_reddit.append(post.title)
# new posts
for post in subreddit.new(limit=1000):
  lst_reddit.append(post.title)


# converting the list into a dataframe and displaying it
df_reddit = pd.DataFrame(lst_reddit, columns=['Post Titles'])
```

```
# classifying the post as positive, negative or neutral and displaying the results
sia = SIA()
results = []
for line in lst_reddit:
  pol_score = sia.polarity_scores(line)
  pol_score['Post Titles'] = line
  results.append(pol_score)

df_reddit_nlp = pd.DataFrame(results)

# compound is taken as the deciding factor is classifying the sentiment

# positive
df_reddit_nlp.loc[df_reddit_nlp['compound'] > 0, 'Sentiment'] = '1'

# negative
df_reddit_nlp.loc[df_reddit_nlp['compound'] < 0, 'Sentiment'] = '-1'

# neutral
df_reddit_nlp.loc[df_reddit_nlp['compound'] == 0.0, 'Sentiment'] = '0'

# grouping post by sentiment
df_reddit_groupby = df_reddit_nlp.groupby('Sentiment').count()
lst1 = df_reddit_groupby['Post Titles']
dict1 = {'Sentiment': sentiment_lst, 'Number of Posts': lst1}
reddit_sent = pd.DataFrame(dict1)
fig = px.bar(reddit_sent, x='Sentiment', y='Number of Posts')
st.plotly_chart(fig)
if lst1[0] > lst1[2]:
   st.write('## Sentiment is Negative')
elif lst1[0] < lst1[2]:
   st.write('## Sentiment is Positive')
```

```
st.text('')
st.text('')

# Twitter Sentiment Analysis

st.write("""
## Twitter Sentiment Analysis
""")

# twitter credentials
consumer_key= 'uLPC3KfMtGFcEeq4CxEOohZeg'
consumer_secret= 'tywsJRvcr2zz5ICg7bkadbSIIjhGFmAlOLjJECjPqMfaRuwc1T'
access_token= '1300465599823314944-VkC6tWnEUrbxTZ1wYpWIxbc8LQCPNL'
access_token_secret= 'DDiF0cmidxoQlT2rgEUCGkP4E2DI8PBwz6WMS5QL51zOG'
auth = tw.OAuthHandler(consumer_key, consumer_secret)
auth.set_access_token(access_token, access_token_secret)
api = tw.API(auth, wait_on_rate_limit=True)

search_words = ['crypto', 'bitcoin']
date_since = '2021-04-20'
tweet_text = []
date_time = []
location = []

# extracting tweet text, datetime and location
for words in search_words:
  #st.write(words)
  tweets = tw.Cursor(api.search,
             q=search_words,
             lang="en",
             since=date_since).items(100)
```

```
  for tweet in tweets:
    str1 = tweet.text
    str2 = tweet.created_at
    str3 = tweet.user.location
    tweet_text.append(str1)
    date_time.append(str2)
    location.append(str3)

df_twitter = pd.DataFrame()
df_twitter['Tweets'] = tweet_text
df_twitter['Created at'] = date_time
df_twitter['Location'] = location

sia = SIA()
results = []
for line in tweet_text:
  pol_score = sia.polarity_scores(line)
  pol_score['Tweets'] = line
  results.append(pol_score)
#st.write(results)
df_twitter_nlp = pd.DataFrame(results)

# compound is taken as the deciding factor is classifying the sentiment

# positive
df_twitter_nlp.loc[df_twitter_nlp['compound'] > 0, 'Sentiment'] = '1'

# negative
df_twitter_nlp.loc[df_twitter_nlp['compound'] < 0, 'Sentiment'] = '-1'

# neutral
df_twitter_nlp.loc[df_twitter_nlp['compound'] == 0.0, 'Sentiment'] = '0'
```

```
    df_twitter_nlp['Created at'] = date_time
    df_twitter_nlp['Location'] = location

    df_twitter_groupby = df_twitter_nlp.groupby('Sentiment').count()
    lst2 = df_twitter_groupby['Tweets']
    sentiment_lst = ['Negative', 'Neutral', 'Positive']
    dict2 = {'Sentiment': sentiment_lst, 'Number of Tweets': lst2}
    twitter_sent = pd.DataFrame(dict2)
    fig = px.bar(twitter_sent, x='Sentiment', y='Number of Tweets')
    st.plotly_chart(fig)
    if lst2[0] > lst2[2]:
      st.write('## Sentiment is Negative')
    elif lst2[0] < lst2[2]:
      st.write('## Sentiment is Positive')




# dropdown menu
option = st.selectbox(
     'Please select your preferred method',
     ('None', 'Sentiment Analysis', 'Time series'))

if option == 'Time series':
    time_series()
elif option == 'Sentiment Analysis':
    sentiment_analysis()
```

# Chapter 7

## Conclusion

While developing this project we learned a lot, from extracting data using APIs to processing natural language and labeling social media posts according to their sentiments. This project also taught us how to deal with time-series data. As described in the slide above we were able to show accurately the sentiment of the public on Bitcoin in real-time while at the same time predicting the future 1-year price based on past trends and seasonalities. To find out the current public sentiment and our latest predictions you can visit our website: https://share.streamlit.io/jashtailor/bitcoin-price-predictor-mk4/main/mk4.py

## References

[1] S. Mohapatra, N. Ahmed and P. Alencar, "KryptoOracle: A Real-Time Cryptocurrency Price Prediction Platform Using Twitter Sentiments," 2019 IEEE International Conference on Big Data (Big Data), 2019, pp. 5544-5551, doi: 10.1109/BigData47090.2019.9006554.

[2] 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), 2018, pp. 1-4, doi: 10.1109/SIU.2018.8404760.

[3] E. Mahendra, H. Madan, S. Gupta and S. V. Singh, "Bitcoin Price Prediction Using Deep Learning and Real Time Deployment," 2020 2nd International Conference on Advances in Computing, Communication Control and Networking (ICACCCN), 2020, pp. 264-268, doi: 10.1109/ICACCCN51052.2020.9362735.

[4] Nakamoto, Satoshi. (2009). Bitcoin: A Peer-to-Peer Electronic Cash System. Cryptography Mailing list at https://metzdowd.com.

[5] S. Velankar, S. Valecha and S. Maji, "Bitcoin price prediction using machine learning," 2018 20th International Conference on Advanced Communication Technology (ICACT), 2018, pp. 144-147, doi: 10.23919/ICACT.2018.8323676

# Acknowledgements

The ability to create and implement in real life can come only through motivation and guidance. We express our sincere thanks to Prof Mrs. Varsha Shrivastava our course instructor for her guidance and periodic assessment of our project without whom this project wouldn't have possible.