Twitter Content Sentiment Analysis to Predict Crypto Currency (Bitcoin) Movement

...

Members:

- Kaushik, Sarthak
- Liendo, Daniel
- Patel, Tilisha
- Yousuf, Muhammad Mehmood

Motivation & Project aim/objectives

- Influence of Social media on stocks was quite visible for GME prices.
- An announcement, opinion, or remark can be enough launch a news story or directly influence the market.
- Hypothesis is that if a company has positive sentiment, it will lead its stock price to increase in the near future.
- Problem: Use twitter information to derive the sentiments (positive, neutral or negative) and predict the direction of stock movement (increase or decrease) to inform the user about buy/sell recommendations.

Introduction/Background

- Research suggests that news impacts stock market movement and indicates the possibility of market prediction.
- Tweets usually contain hashtags and symbols that facilitate the search for relevant posts and easier to crawl
- Tweets classification is to use them as a time series supposedly correlated to another time series from stock/Bitcoin data.

Tools / Libraries Used

- Python
- Octoparse for Tweets Scraping / Twitter API
- Yahoo-finance
- NLTK NLP library
- SentimentIntensityAnalyzer function for Deriving Sentiments
- Swifter
- Scikitlearn
- Keras

Assumptions

- Stock market movement every 15 minutes frequency
  - Assumption that any news would impact within 15 minutes of the event.
  - With investors, it is now feasible for them to act within minutes of an event
- Stock market is closed, the tweet was analyzed to impact as part of next day opening price
  - When the markets open they react to the information that has happened since the markets were closed
- Tweets sentiments were averaged out during the 15 minutes
- Used pretrained VADER NLTK sentiment analysis algorithm for sentiment analysis

Data Mining

TWEETS

- Web Scraping technique
- 102,683 tweets
- 7 twitter accounts

Features

- Tweet and the derived sentiment
- ~30k tweets after merging & data processing

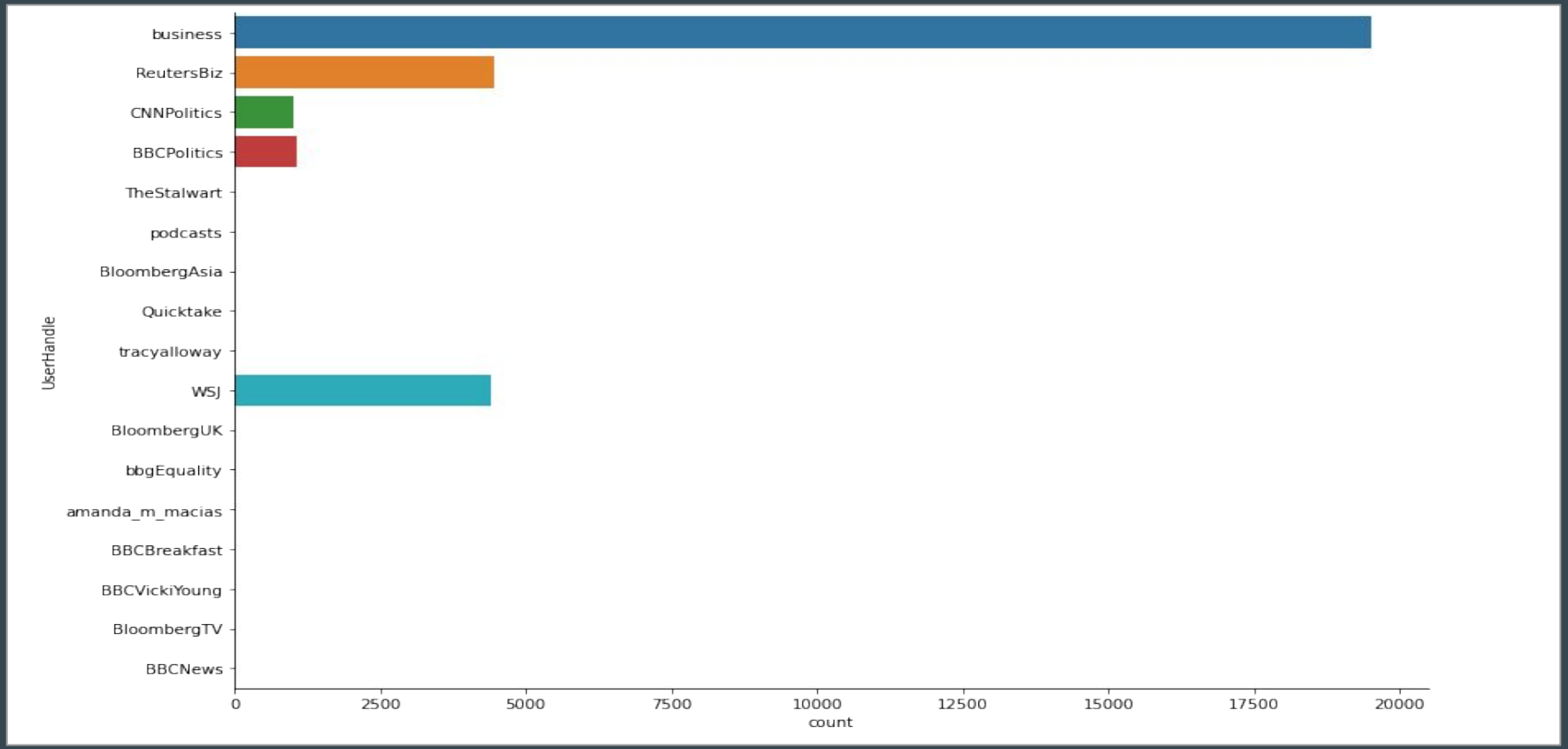
STOCK

- Yahoo finance API
- Stocks / Market Indices / 1 Crypto (BTC)
- 15 minute interval could only be done for the last 60 days (limitation)

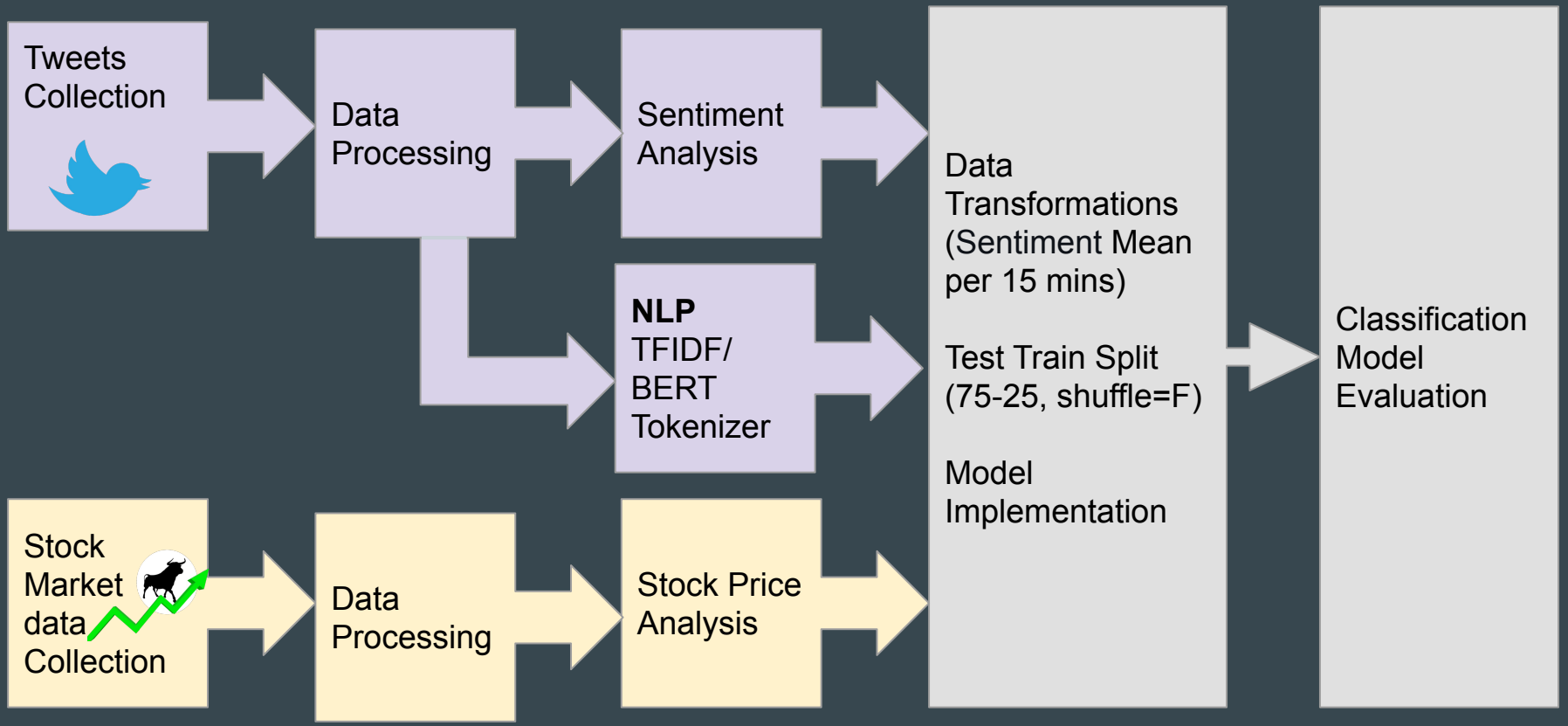
Label

- Stock price
- Derived stock direction (Increase or Decrease) label as Target

Tweet Sources (After Pre-processing)

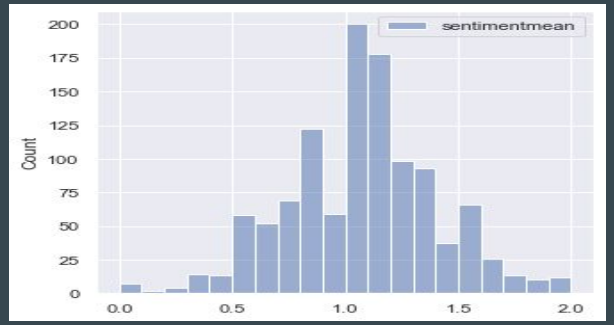


Data Processing Pipeline

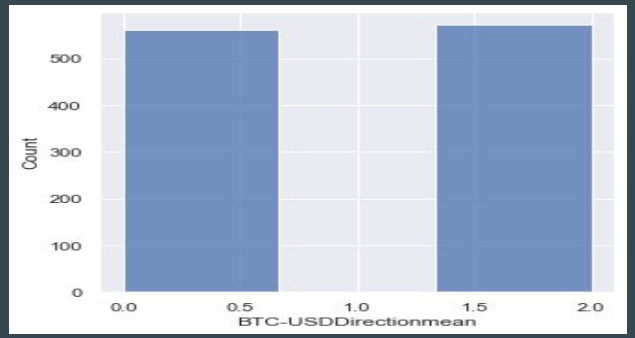


Exploratory Data Analysis (EDA)

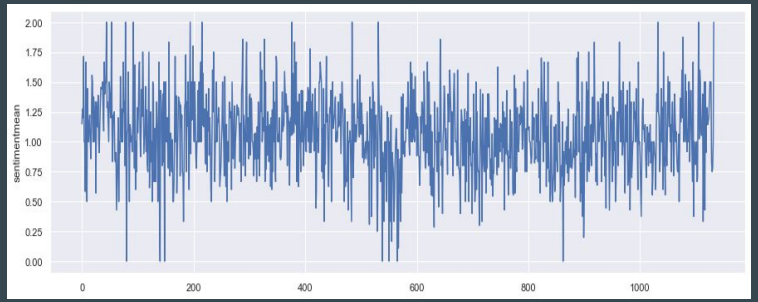
Sentiment Histogram



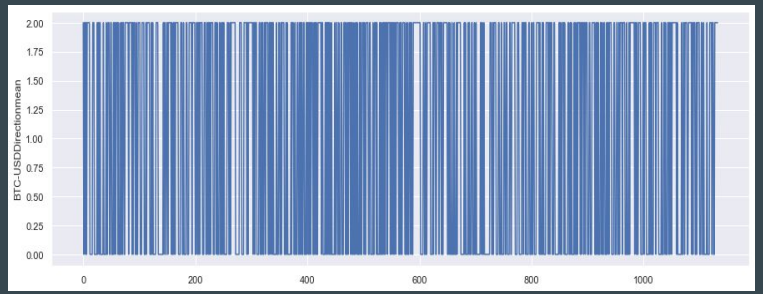
Label Direction Histogram



Time-Series Change for Sentiment



Time-Series Change for Label



Classification Models Tuning Built on Sentiment Analysis

Classification Algorithm	Hyper Parameters
Decision Trees	`'criterion':['gini','entropy'],` `'max_depth':[5,10,20]`

Random Forest Classifier	`'criterion':['gini','entropy'],`
	`'max_depth':[5,10,20],`
	`'n_estimators':[10,20,100]`

SVM Classifier Pipeline	`Pipeline(steps=[("poly_features", PolynomialFeatures()),("svm_clf", SVC())])`
	`param_grid = {'poly_features__degree':[1,2,3],`
	`'svm_clf__C': [0.1, 1, 10, 100, 1000],`
	`'svm_clf__gamma': [1, 0.1, 0.01, 0.001, 0.0001],`
	`'svm_clf__degree':[1,2,3],`
	`'svm_clf__kernel': ['linear','rbf','poly','sigmoid']}`

Naive Bayes & Neural Network (using the tweet content)	No Hyper parameter tuning. (Future Work)

Classification Models Built on Tweet Text

Classification Algorithm	Hyper Parameters
Naive Bayes	`Processed Tweets > CountVectorizer > TFIDF > Naive Bayes Estimator Evaluation`

Neural Network (using BERT Embeddings & BERT Pretrained encoder)	`Processed Tweets >`
	`BERT Preprocessing >`
	`BERT Encoding >`
	`Drop Out >`
	`Output Layer`
	
	`Loss= BinaryCrossentropy`
	`Metrics= BinaryAccuracy`
	`nEpochs=5`

```
classifier_model.summary()
```

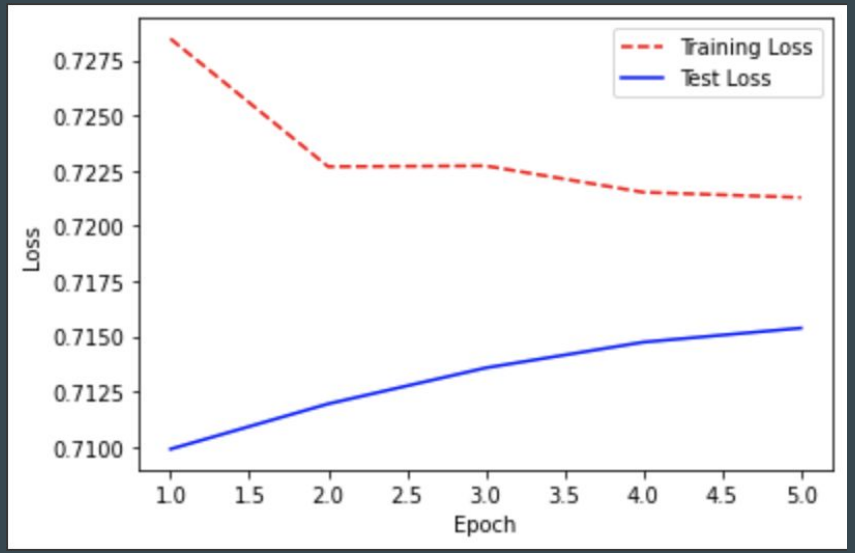
```
Model: "model"
__________________________________________________________________________________________________
 Layer (type)                   Output Shape         Param #     Connected to
==================================================================================================
 text (InputLayer)              [(None,)]            0           []

 preprocessing (KerasLayer)     {'input_type_ids':   0           ['text[0][0]']
                                 (None, 128),
                                 'input_word_ids':
                                 (None, 128),
                                 'input_mask': (Non
                                 e, 128)}

 BERT_encoder (KerasLayer)      {'sequence_output':  4385921     ['preprocessing[0][0]',
                                  (None, 128, 128),               'preprocessing[0][1]',
                                 'encoder_outputs':               'preprocessing[0][2]']
                                 [(None, 128, 128),
                                 (None, 128, 128)],
                                 'pooled_output': (
                                 None, 128),
                                 'default': (None,
                                 128)}

 dropout (Dropout)              (None, 128)          0           ['BERT_encoder[0][3]']

 classifier (Dense)             (None, 1)            129         ['dropout[0][0]']

==================================================================================================
Total params: 4,386,050
Trainable params: 129
Non-trainable params: 4,385,921
__________________________________________________________________________________________________
```

Classification Models Results

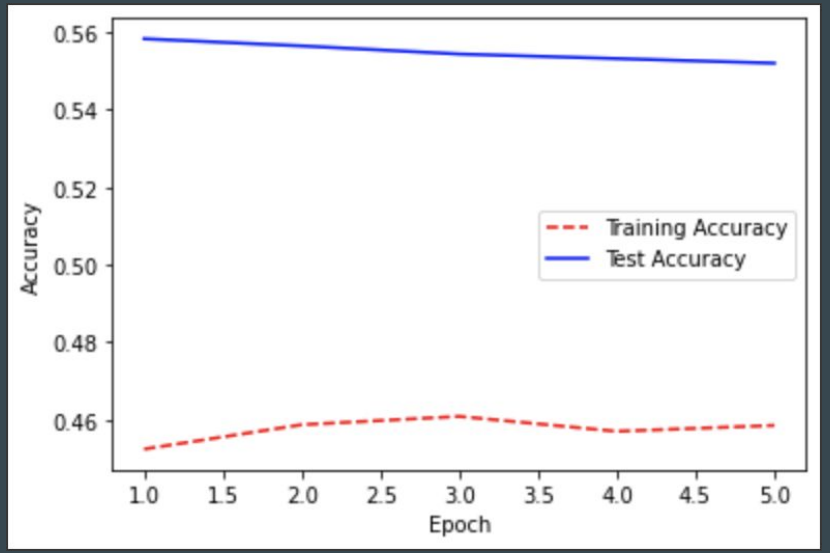
Classification Algorithm	Train Accuracy				Test Accuracy			
	Accuracy	Precision	Recall	F1-Score	Accuracy	Precision	Recall	F1-Score
Decision Trees	64.19	63.64	64.85	64.24	47.89	50.70	48.00	49.32
Random Forest Classifier Pipeline	58.66	59.94	50.12	54.59	49.3	52.59	40.67	45.86
SVM (Best Model)	53.00	52.44	56.06	54.19	52.82	55.56	53.33	54.42
Naive Bayes (using the tweet content)	74.41	69.69	95.91	80.72	44.76	42.83	85.26	57.02
Neural Network BERT Tokenizer (using the tweet content)	45.85				55.20			

Neural Network BERT Tokenizer

Loss Per Epoch



Accuracy per Epoch



Discussion & Conclusion

- Two ways of predicting cryptocurrency movements in 15 minute interval is implemented successfully.
    - Sentiment Analysis is not conclusive in prediction of stock movement. Perhaps inclusion of other features like tweet likes etc. may be evaluated as features
    - Analysis of direct tweet content is also not conclusive in prediction of stock movement.
- The twitter handles must be handpicked to study the impact of a particular index, commodity, cryptocurrency.
- The stock data also limits how much information can be used as initial dataset.
- Perhaps the assumption that the information from social media is able to impact the markets within 15 minutes is not correct.

Challenges During Execution

- Data Mining
  - Struggle with yahoo finance API, limits on getting frequent market quotes beyond 60 days
  - Struggle with Twitter API, had to use Web scraping tool
- Quality of Data
  - Twitter stock data has a lot of noise (stickers, ads, redirection urls, memes etc.)
  - Misclassification of sentiment because of noise
- Data Transformation
  - Combining the data for twitter/stocks

Lesson Learned & Path Forward

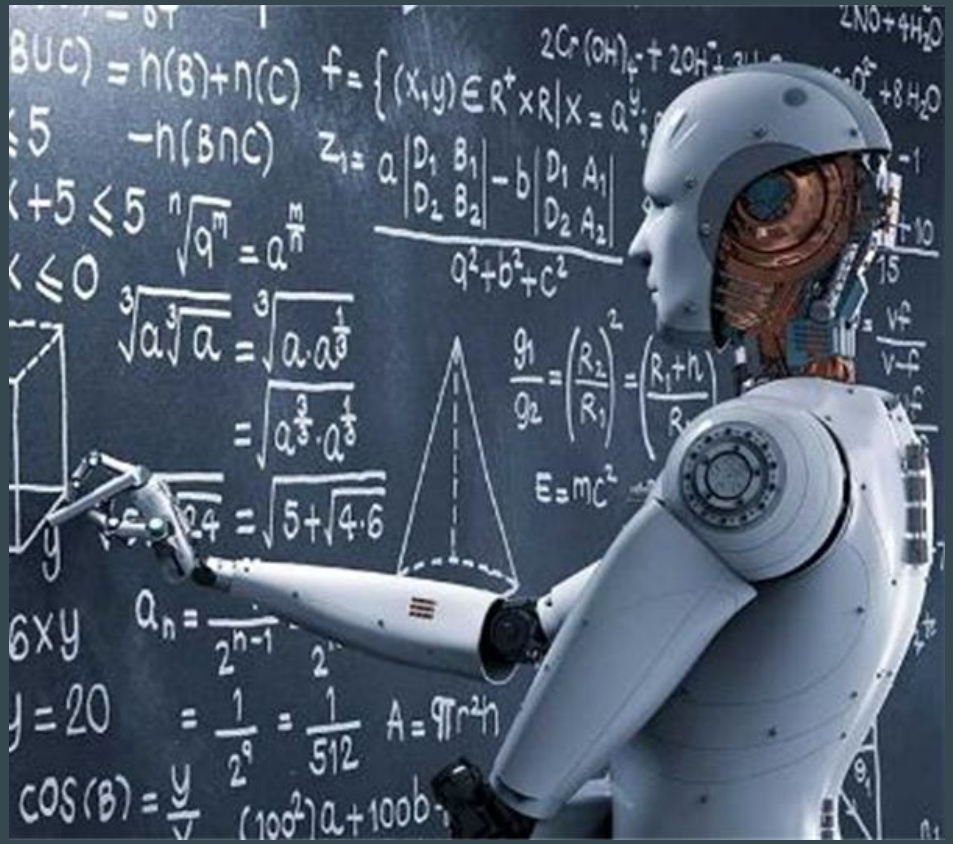
Lessons Learned

- Make sure you have easy access to the data you need and a data strategy in place.
- Proof of Concept Model

Path Forward

- More research on userhandles and the commodity that needs to be predicted to ensure relevant information is gathered.
- Data gathering of 15-minute stock data over a long period of time.
- Expand the data by including other social media platforms like reddit, discord etc.
- Ensemble Model - Voting Classifier

THANK YOU!


$n(B \cup C) = n(B) + n(C)$
$n(B \cap C)$
$\sqrt[n]{q^m} = a^{\frac{m}{n}}$
$\sqrt{a\sqrt[3]{a}} = \sqrt{a \cdot a^{\frac{1}{3}}}$
$= \sqrt{a^{\frac{3}{3}} \cdot a^{\frac{1}{3}}}$
$= \sqrt{5 + \sqrt{4 \cdot 6}}$
$f = \{(x, y) \in R^+ \times R \mid x = a^y\}$
$z_1 = a \begin{vmatrix} D_1 & B_1 \\ D_2 & B_2 \end{vmatrix} - b \begin{vmatrix} D_1 & A_1 \\ D_2 & A_2 \end{vmatrix}$
$a^2 + b^2 + c^2$
$\frac{g_1}{g_2} = \left(\frac{R_2}{R_1}\right)^2 = \left(\frac{R_1 + h}{R}\right)$
$E = mc^2$
$a_n = \frac{}{2^{n-1}}$
$= \frac{1}{2^9} = \frac{1}{512}$
$A = \pi r^2 h$
$\cos(B) = \frac{y}{}$
$(100^2)a + 100b$
$2Cr(OH)_4^- + 2OH^-$
$2NO + 4H_2O$
$+ 8H_2O$