Stock price movement based on off-trading hour Twitter data

Andrew Lee, Phillip Chen, Hannah Do

CSCI 49362/76000 | Fall 2020

Problem Description

Stock prices tend to shift based on news releases, announcements, introduction of a new product or a recall, change of management, and many other factors.

And these popular trends or news tend to increase volume on tweets within a few hours after the event, with a range of various sentiments.

Our goal is to predict this volatile stock market via history of Twitter collection open to the public, which makes it an inexpensive and efficient tool for prediction.

Research Paper Influences

- Garcia-Lopez et al. (2018) provided instructions on tweet normalization
- Valencia et al. (2019) mentioned using VADER for their sentiment analysis
- Ranco et al. (2015) and Smailovic et al. proved there is be a correlation between tweets stock data on event-based timeline - suggested causality test on given events or time period - for our project, differences in stock prices between closing hours and reopening hour the next day.

Approach - Stock Data Collection

Symbols	AAPL	MSFT	FB	GOOG	AMZN	^GSPC
Date						
2020-09-01	3.720001	-0.020004	1.509979	2.449951	38.620117	7.129883
2020-09-02	3.410004	0.699997	3.440002	13.065063	47.879883	17.110107
2020-09-03	-4.489990	-2.379990	-6.510010	-18.566040	-46.449951	-16.100098
2020-09-04	-0.809998	-2.199997	-3.869995	-17.579956	-50.000000	-1.459961
2020-09-08	-7.010002	-7.750000	-11.450012	-57.530029	-150.620117	-55.080078
...	...	...	...	...	...	...
2020-11-24	0.060005	-0.520004	0.059998	-4.359985	2.110107	16.929932
2020-11-25	0.380005	1.250000	1.220001	4.010010	23.810059	0.090088
2020-11-27	0.540001	0.980011	1.800018	1.659912	26.189941	8.900146
2020-11-30	0.380005	-1.129990	-1.779999	-12.005981	13.139893	-4.170166
2020-12-01	1.959999	0.439987	2.190002	13.630005	20.459961	24.240234

64 rows × 6 columns

Google Colab - Datareader

Using ticker symbol and Panda's datareader, we retrieved stock values from queried companies and timeframe.

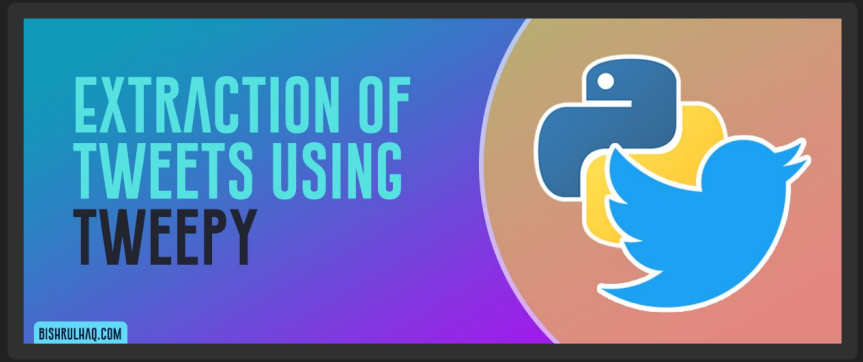
Closing price - Opening price (the following day)

- AAPL : Apple
- MSFT : Microsoft
- FB : Facebook
- GOOG : Google
- AMZN : Amazon

Approach - Twitter Collection

We used **Snscrape**, which is a library that can search for keywords or profiles in social media.
i.e. Facebook, Instagram, Reddit, Twitter, Weibo.

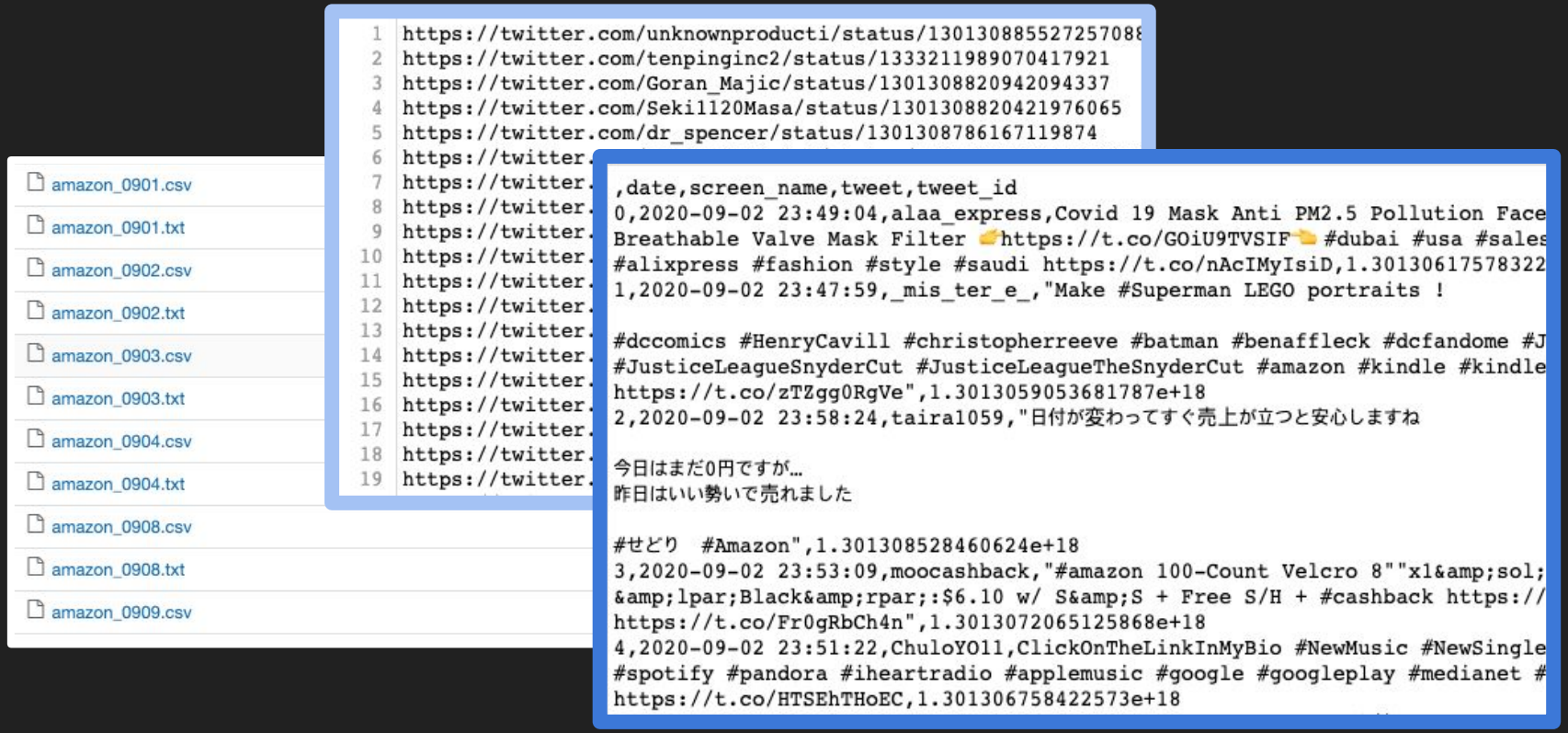
1. *Retrieved URL of the tweets by querying company names.*



And **Tweepy**, a Python Library that can access various attributes of a twitter account.

2. *With the URLs of the tweets obtained for Snscrape, we extracted the tweet content through Tweepy.*
   (Issue : There was an extraction rate limit - we applied for 3 developer accounts to extract tweets)

Approach - Twitter Collection



Approach - Sentiment Analysis

Vader's Sentiment Intensity Analyzer :

Vader stands for Valence Aware Dictionary for Sentiment Reasoning.
It is adequate for tweets since it calculates sentiment scores for emojis and various word types. It returns positive, negative, and neutral values - along with the average compound score.

tweet	score_dict	score_compound
Hey @amazon, your driver is dumping packages i...	{'neg': 0.058, 'neu': 0.861, 'pos': 0.081, 'co...	-0.0258
"Rii RK100+ Rainbow LED Backlit Wired Mechanic...	{'neg': 0.0, 'neu': 0.93, 'pos': 0.07, 'compou...	0.1280
#book #childrensbook #childrensmysterynovel #m...	{'neg': 0.179, 'neu': 0.821, 'pos': 0.0, 'comp...	-0.5423
You gotta peep this one:\nPetlinks Pure Bliss ...	{'neg': 0.0, 'neu': 0.862, 'pos': 0.138, 'comp...	0.6114
I don't think I'd ever get tired of checking t...	{'neg': 0.101, 'neu': 0.839, 'pos': 0.059, 'co...	-0.2960
...	...	...

How does VADER work? (part 1)

Punctuation

```
[41] #Base Sentence
     sentiment_analyzer_scores('The food here is good!')

     {'neg': 0.0, 'neu': 0.556, 'pos': 0.444, 'compound': 0.4926}

     #Capitalized Sentence
     sentiment_analyzer_scores('The food here is good!!!')

     {'neg': 0.0, 'neu': 0.514, 'pos': 0.486, 'compound': 0.5826}
```

Capitalization

```
[43] #Base Sentence
     sentiment_analyzer_scores('The food here is great!')

     {'neg': 0.0, 'neu': 0.477, 'pos': 0.523, 'compound': 0.6588}

     #Capitalized Sentence
     sentiment_analyzer_scores('The food here is GREAT!')

     {'neg': 0.0, 'neu': 0.438, 'pos': 0.562, 'compound': 0.729}
```

How does VADER work? (part 2)

Conjunctions

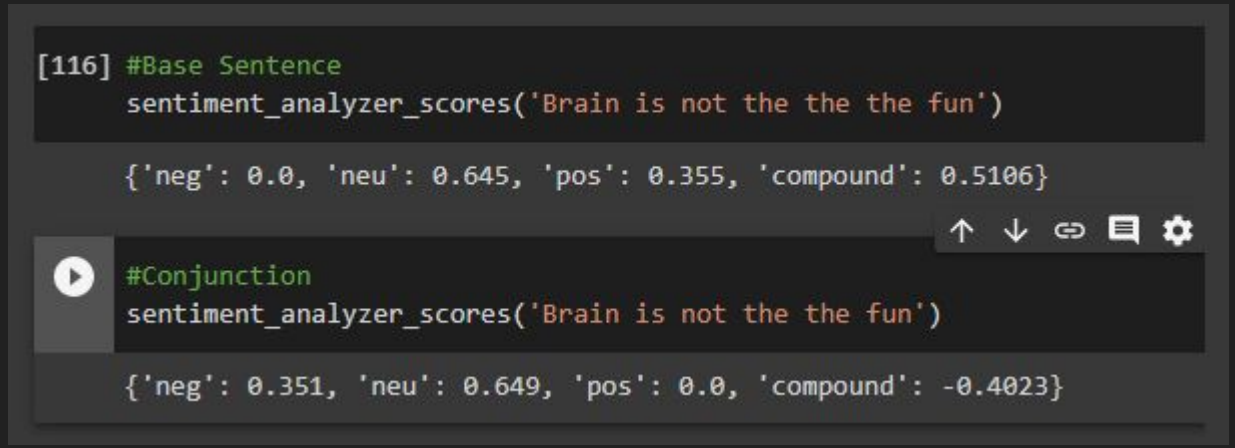
```
[100] #Base Sentence
      sentiment_analyzer_scores('Brain is smart')

      {'neg': 0.0, 'neu': 0.426, 'pos': 0.574, 'compound': 0.4019}

      #Conjunction
      sentiment_analyzer_scores('Brain is smart, but lazy')

      {'neg': 0.401, 'neu': 0.37, 'pos': 0.228, 'compound': -0.34}
```

Degree Modifiers

```
[128] #Base Sentence
      sentiment_analyzer_scores('The service is good')

      {'neg': 0.0, 'neu': 0.508, 'pos': 0.492, 'compound': 0.4404}

      #Degree modifier
      sentiment_analyzer_scores('The service is extremely good')

      {'neg': 0.0, 'neu': 0.556, 'pos': 0.444, 'compound': 0.4927}
```

How does VADER work? (part 3)

Preceding Tri-gram To Detect Negation :



```
[116] #Base Sentence
      sentiment_analyzer_scores('Brain is not the the the fun')

      {'neg': 0.0, 'neu': 0.645, 'pos': 0.355, 'compound': 0.5106}

      #Conjunction
      sentiment_analyzer_scores('Brain is not the the fun')

      {'neg': 0.351, 'neu': 0.649, 'pos': 0.0, 'compound': -0.4023}
```

How does VADER work? (part 4)

Slangs

```
[75] #Base Sentence
     sentiment_analyzer_scores('The food here sucks')

     {'neg': 0.455, 'neu': 0.545, 'pos': 0.0, 'compound': -0.3612}

     #Capitalized Sentence
     sentiment_analyzer_scores('The food here sux')

     {'neg': 0.455, 'neu': 0.545, 'pos': 0.0, 'compound': -0.3612}
```

Emoji and Emoticon

```
[96] #emoji 1
     sentiment_analyzer_scores('I am 😆 today')

     {'neg': 0.0, 'neu': 0.472, 'pos': 0.524, 'compound': 0.6705}

     #emoji 2
     sentiment_analyzer_scores('I am 😢 today')

     {'neg': 0.706, 'neu': 0.294, 'pos': 0.0, 'compound': -0.34}
```

Approach - Feature selection

	date	stock price change	twitter sentiment	twitter volume	up or down
1	20201103	5.750000	0.3182	10089	1
2	20201104	60.070068	-0.5707	4184	1
3	20201105	31.869995	0.8918	4192	1
4	20201106	-9.420044	0.1739	4339	0
5	20201109	29.150024	0.4019	4110	1
6	20201110	-31.910034	-0.3802	8737	0
7	20201111	9.609985	0.5707	3956	1
8	20201112	-5.079956	0.1697	4535	0
9	20201113	7.790039	-0.2960	1019	1
11	20201117	-4.440063	0.3182	5361	0
13	20201119	-8.400024	0.3612	3330	0
14	20201120	1.289917	0.7579	3712	1
15	20201123	7.410034	0.2247	3704	1
16	20201124	-4.359985	0.7615	7958	0

Current features :

- Stock price movement
- Mean value of the multiple tweet sentiment
- Total volume of tweets
- Day of the week
- S&P 500 / Nasdaq values
- Up or down : **target vector,** binary value of stock price movement

Possible additional features :

- # of neutral tweets
- sentiment score before & after the neutral tweet removal
- measure of strong/weakness of the polarity

Approach - ML models

Logistic Regression :

Logistic regression is an algorithm used commonly for solving classification problems. Given an input variable (X) where the output variable (y) is a discrete value which ranges between 1 (yes) and 0 (no). Uses logistic (Sigmoid) function for classification.

KNN :

The KNN algorithm assumes that similar things exist in close proximity. Transforming the data points into feature vectors and calculates the distance between the points to classify them.

SVM-SVR :

The support vector machine is a model used for both classification and regression problems though it is mostly used to solve classification problems. The algorithm creates a hyperplane or line (decision boundary) which separates data into classes.

RF :

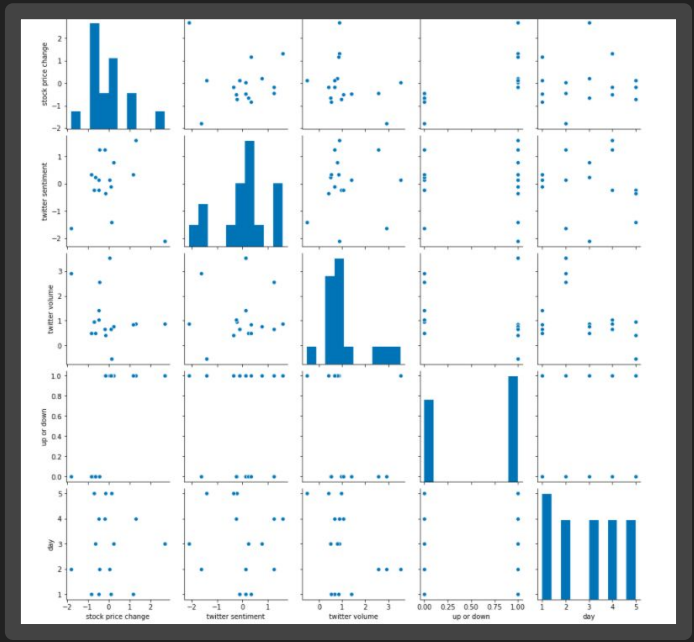
Random Forest consists of a large number of individual decision trees. Each tree predicts the class and class with the most votes becomes the eventual prediction.

Result - 1 month time period

(Google, Nov. 2020)

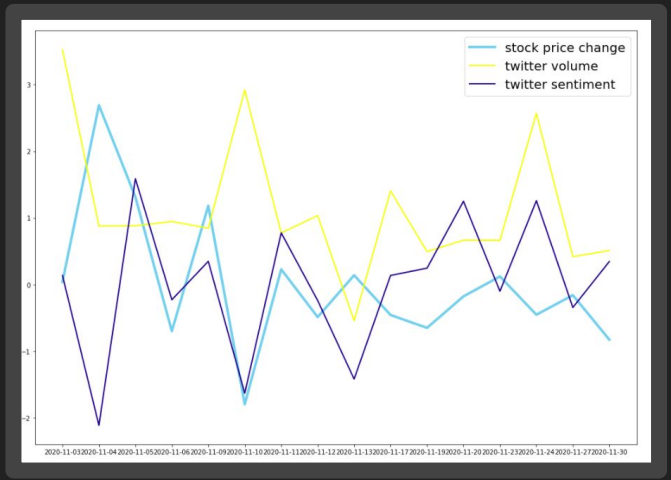
	date	stock price change	twitter sentiment	twitter volume	up or down
1	2020-11-03	0.039720	0.136099	3.526099	1
2	2020-11-04	2.691114	-2.114922	0.877014	1
3	2020-11-05	1.314652	1.588664	0.880603	1
4	2020-11-06	-0.700739	-0.229322	0.946550	0
5	2020-11-09	1.181889	0.348058	0.843817	1
6	2020-11-10	-1.798489	-1.632506	2.919568	0
7	2020-11-11	0.228128	0.775521	0.774729	1
8	2020-11-12	-0.488897	-0.239958	1.034479	0
9	2020-11-13	0.139295	-1.419281	-0.542859	1

Normalization of scores by tuning the values to z-scores



Result - 1 month time period

(Google, Nov. 2020)



Linear line plot on each column - ‘stock price change’, ‘twitter volume’ and ‘twitter sentiment’

```
[K-NN algorithm] accuracy_score: 1.000.
[Logistic regression algorithm] accuracy_score: 0.250.
[SVM.SVC] accuracy_score: 0.750.
[Random forest algorithm] accuracy_score: 1.000.
```

Accuracy tested after various ML methods after CV :
unstable - not enough data

Result - 3 month time period

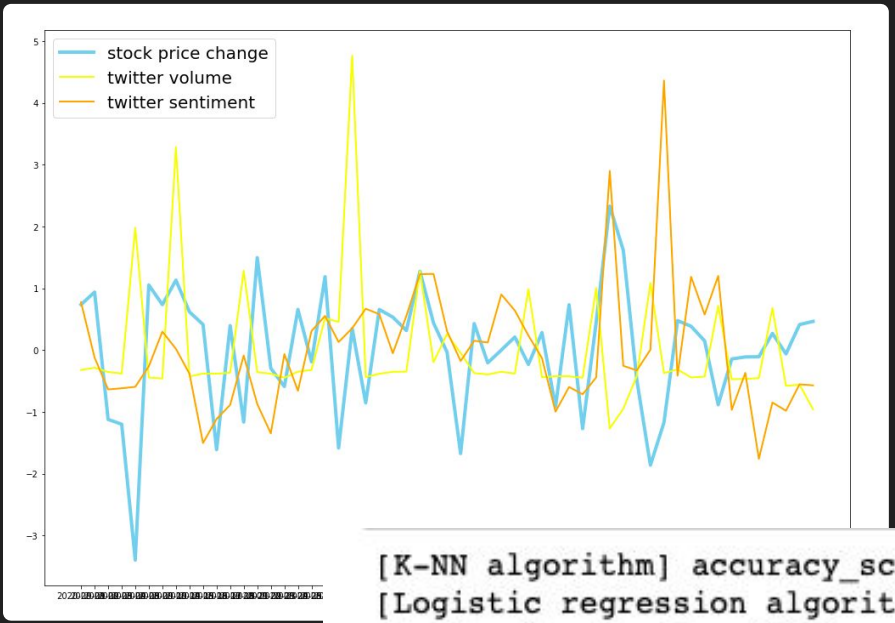
(Amazon, Sep-Nov. 2020)

	date	stock price change	S&P 500	twitter sentiment	twitter volume	day of w	up or down
1	2020-09-01	38.620117	7.1298828125	0.338225	11182	2	1
2	2020-09-02	47.879883	17.110107421875	0.305760	11608	3	1
3	2020-09-03	-46.449951	-16.10009765625	0.287029	10801	4	0
4	2020-09-04	-50.000000	-1.4599609375	0.287713	10511	5	0
5	2020-09-08	-150.620117	-55.080078125	0.288550	38280	2	0
6	2020-09-09	53.149902	37.97998046875	0.300570	9745	3	1
7	2020-09-10	38.609863	13.60009765625	0.321017	9569	4	1
9	2020-09-14	56.719971	22.590087890625	0.310872	53610	1	1
10	2020-09-15	33.189941	24.18994140625	0.296226	9935	2	1
11	2020-09-16	23.860107	10.030029296875	0.255528	10515	3	1

S&P 500 values, day of the week added as new feature sets

Result - 3 month time period

(Amazon, Sep-Nov. 2020)



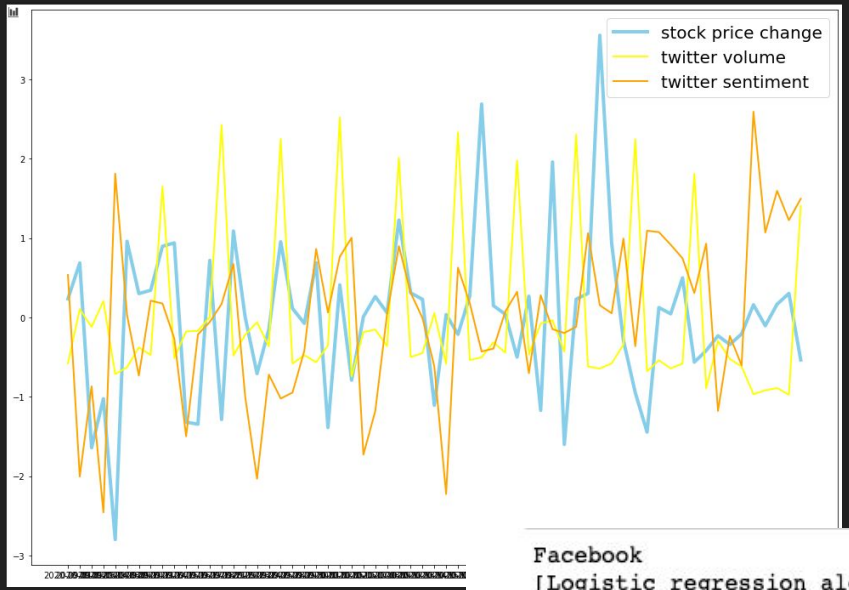
Predicted and evaluated with various ML methods :

Random Forest performs the best with the accuracy of 0.909

```
[K-NN algorithm] accuracy_score: 0.818.
[Logistic regression algorithm] accuracy_score: 0.727.
[Random forest algorithm] accuracy_score: 0.909.
```

Result - 3 month time period

(FB, Sep-Nov. 2020)



Observation:

The string ‘fb’ had too much noise in the tweets, collecting bulk of unrelated data into the feature set. This may be due to Facebook being a social account.

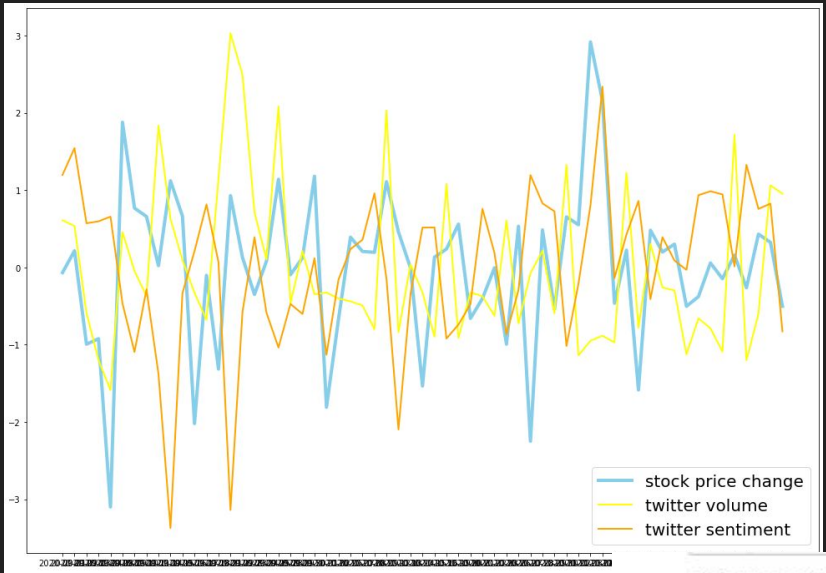
Alternative option we can try is querying by ‘$fb’ to find only stock-related values.

Logistic Regression performs the best with the accuracy of 0.846

```
Facebook
[Logistic regression algorithm] accuracy_score: 0.846.
[Random forest algorithm] accuracy_score: 0.769.
[SVM.SVC] accuracy_score: 0.615.
```

Result: 3 month time period

(Microsoft, Sep-Nov. 2020)



Observation:

Microsoft is known to be very steady, hard to predict any big change in the stock price.

Also the volume of the tweets were much lower than that of Facebook, which may be the reason that the predictions are not so accurate.

Random Forest performed the best with the accuracy of 0.769

```
Microsoft
[Random forest algorithm] accuracy_score: 0.769.
[K-NN algorithm] accuracy_score: 0.615.
[Logistic regression algorithm] accuracy_score: 0.615.
```

Discussion

- Data malforming while conversion - lead to some missing data
- Would be preferable to have unlimited access to tweets to run in ML models with a wider time frame - at least a year to collect enough datasets
- More features related to the sentiment or characteristics of the tweets
- Sourcing from another platform - i.e. *News Headlines*
- Knowing the magnitude of the stock movements would be helpful
- Adding ‘first hour’ of the stock market instead of getting (closing price - opening price)
- Tweets may sometimes have much less to do with how user views a company. (e.g. “please like my post on facebook”)

Team Roles

Hannah :

Collected tweets and stock data for Google (1 month) and Amazon (3 months) and implemented feature extraction and ML models

Phillip :

Collected tweets and stock data for Facebook (3 months) evaluated performance based on the correlation between feature sets and prediction of the ML model

Andrew :

Collected tweets and stock data for Microsoft (3 months) and implemented efficient collection of Twitter URL and text for data gathering, along with evaluation on the prediction to suggest new feature sets

Thank you.