Sentiment Analysis on Tweets of Stocks

Dr. Khalaj

Department of Electrical Engineering

Arad Mahdinejad Kashani 400102028

Project Report
Foundations of Data Science

February 6, 2024

Sentiment Analysis on Tweets of Stocks

Project Report

Arad Mahdinejad Kashani 400102028

Introduction

I did use Git for this project. I have messaged you on Telegram so as to add you to my private repository. If needed, you can contact my Telegram at *@aradmnk*. Not much is happening, though. There are only 4 python notebooks and the .csv file for the web-crawler ('divar.csv', because I crawled 'divar.ir').

The process is also documented in the notebook files themselves. There is text, and the code is segmented by the tasks in each phase of the project, so hopefully it is readable. There are 4 notebook files, for the 3 phases and 1 for the bonus points.

Phase 1

After reading the files, I made a realization that there exists a certain tag (referred to a **CashTag $** in the paper mentioned in the project documentation) in the df['text'] column. I used regex to separate them in a different column and used that from then on.

Task 1

Task

Find the most and least tweeted stocks.

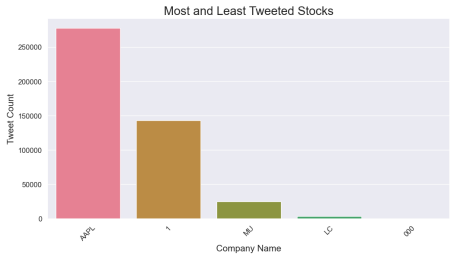
```
Most tweeted about:ticker          AAPL
tweet_count    277799
Name: 11288, dtype: object
Least tweeted about:ticker         000
tweet_count      1
Name: 3, dtype: object
```

Figure 1: Most and least stocks

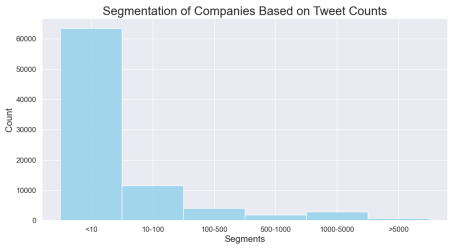
As we can see, the most tweeted about stocks is **Apple Inc.** ***(AAPL)***. The least tweeted about is irrelevant because there existed many tickers with only 1 tweet (I checked it manually, it is not included in the notebook).

Task

Segment the companies based on tweet counts.



(a) Some companies and their respective tweet counts



(b) Segmentation of the companies

Figure 2

The middle companies of *3.(a)* were hand picked from the middle, by index.

Task 2

Task

Statistics on distributions of 5 individual stocks over time. Choose the individual stocks to perform reflect different sectors of the economy.

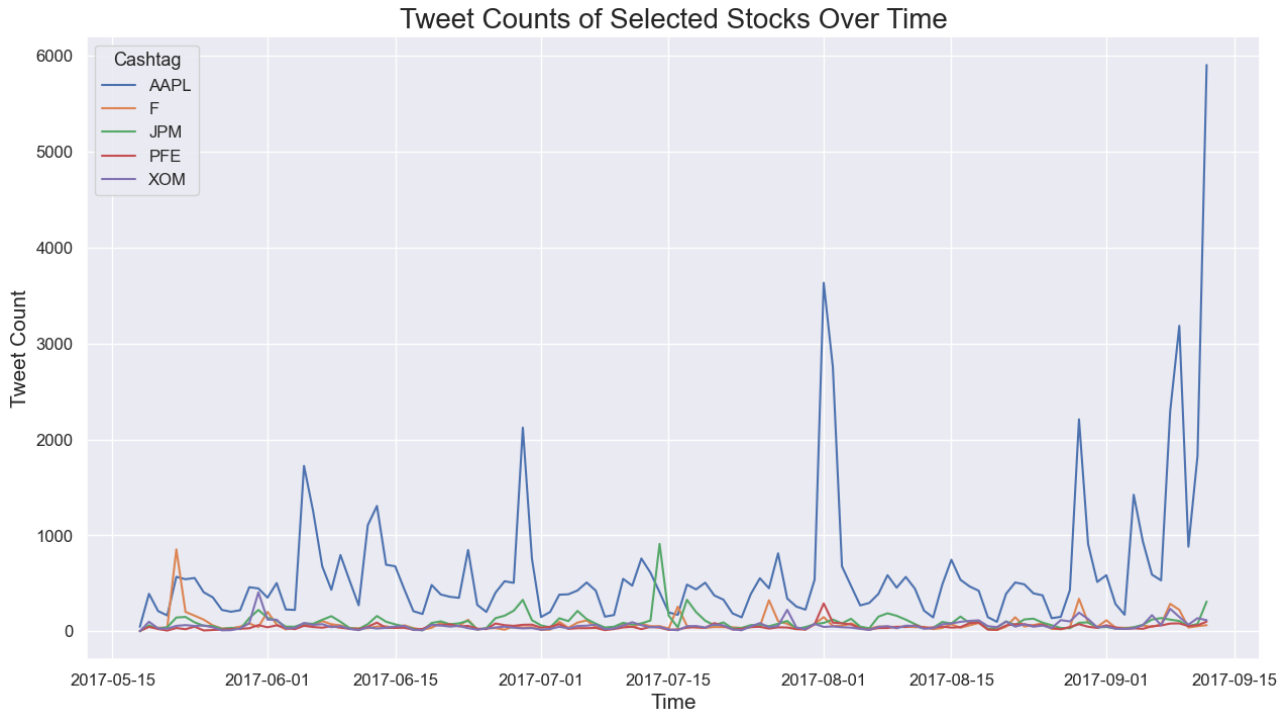


Figure 3: Individual stock distributions

I manually chose **Apple** (Technology), **ExxonMobil** (Energy), **Pfizer** (Healthcare), **JP-Morgan Chase** (Financial), and **Ford** (Automotive).

Task 3

Task

Statitistics on distributions of all financial tweets over time.

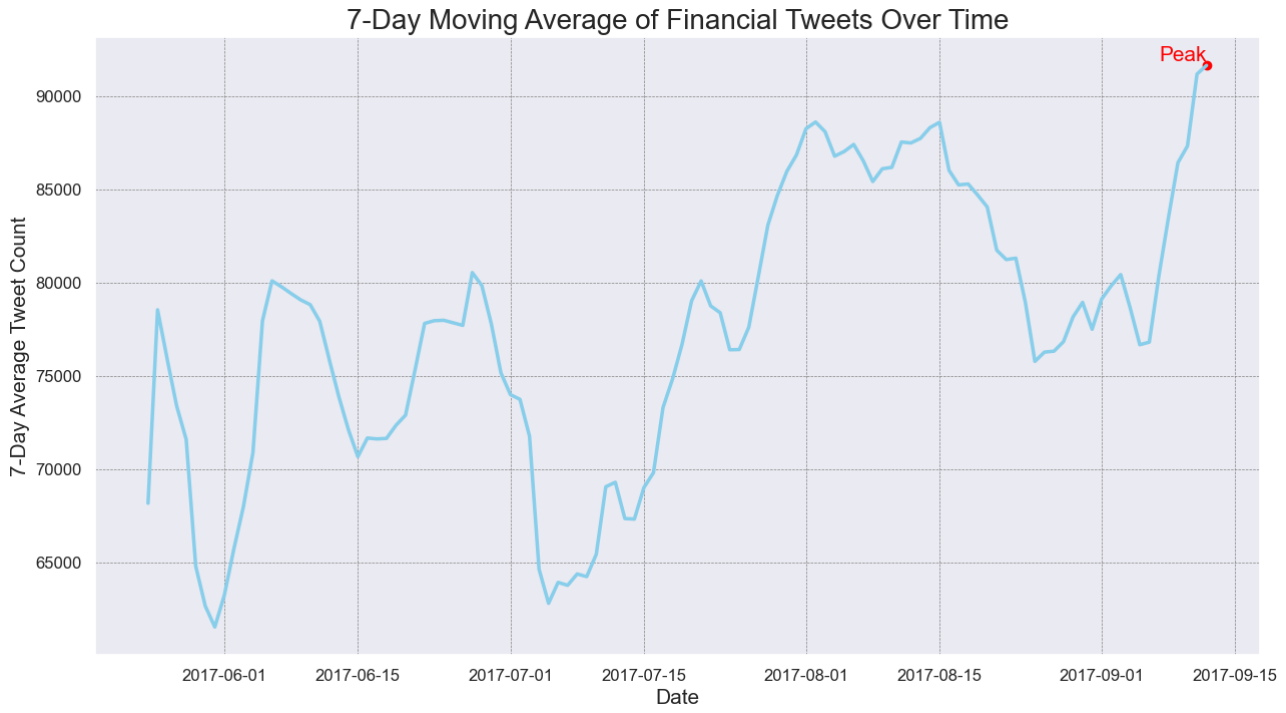


Figure 4: 7-day average of all financial tweets over time

For this one I took on a 7-day moving average approach, just for the sake of doing it.

Task 4

Task

Statistics on distributions of retweets per tweets including individual stocks (at least 2 chosen stocks) over time.

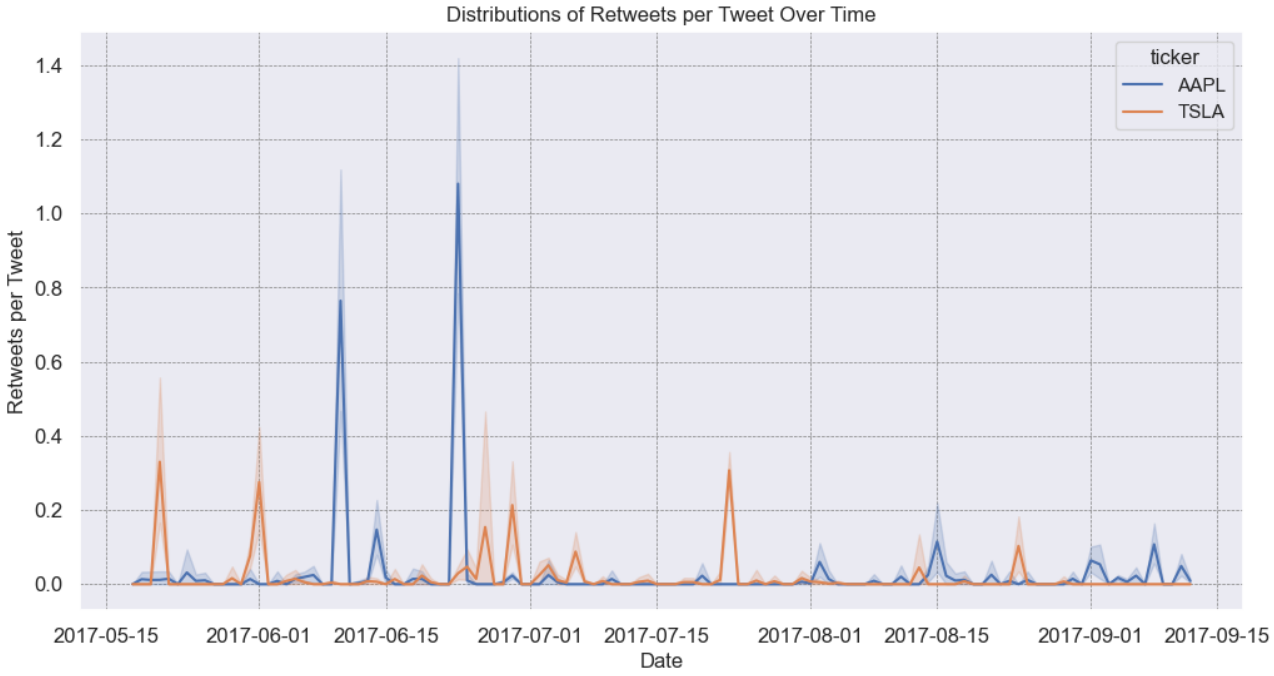


Figure 5: Distribution of retweets-per-tweet of **AAPL** and **TSLA** over time

I used Apple Inc. **(AAPL)** and Tesla Inc. **(TSLA)**, because I saw those in the demos and decided to use them for some reason.

Task 5

Task

Statistics on most important financial information on individual stocks (at least 2 chosen stocks) computed solely from the financial information (not the tweets).

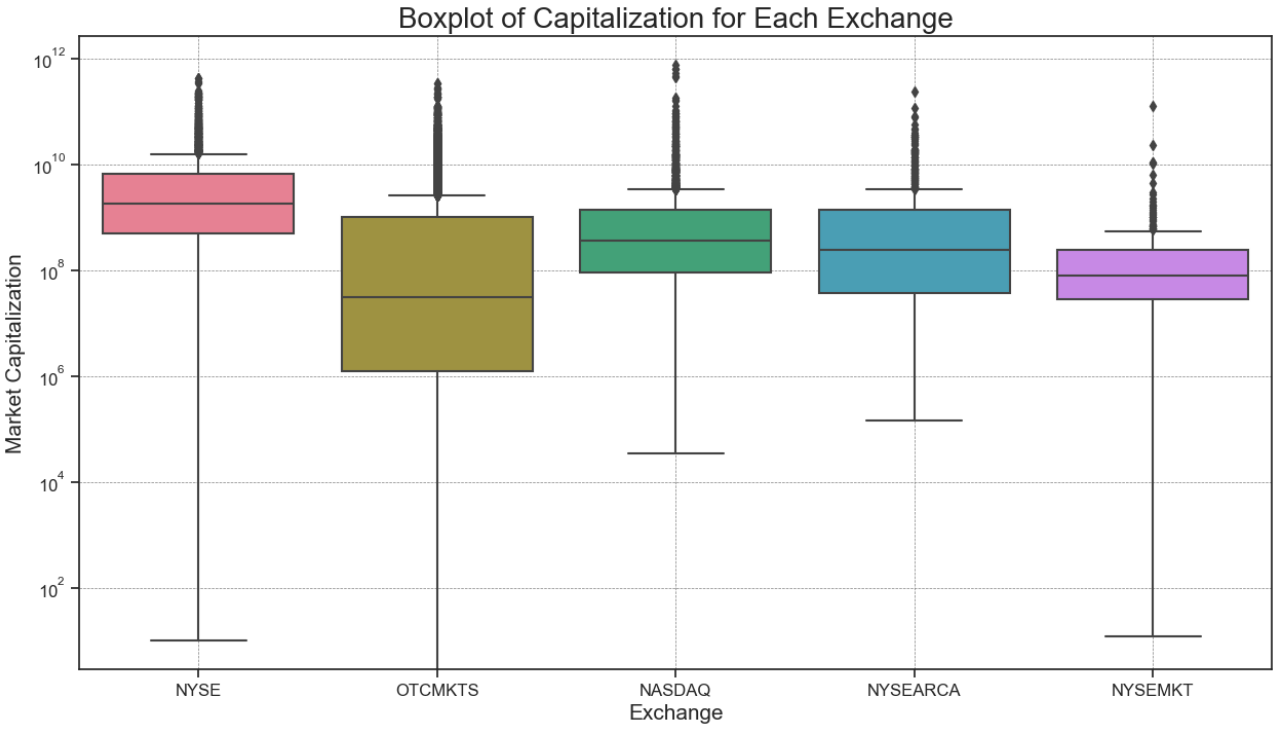
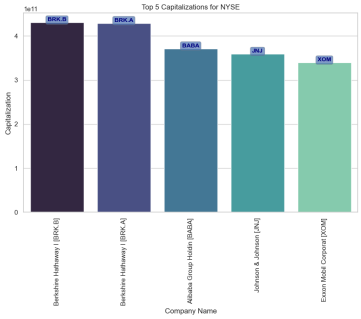
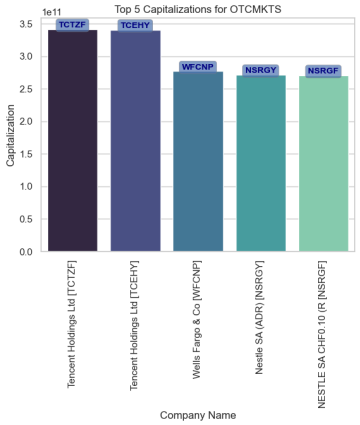


Figure 6: Box-plot of capitalizations per exchange (logarithmic)

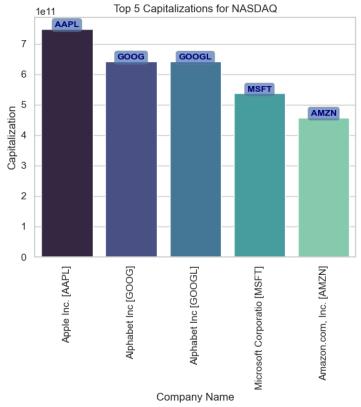
The data for the box-plots was obtained from the 'companies.csv' file.



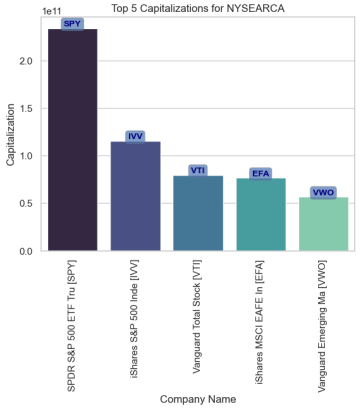
(a) NYSE



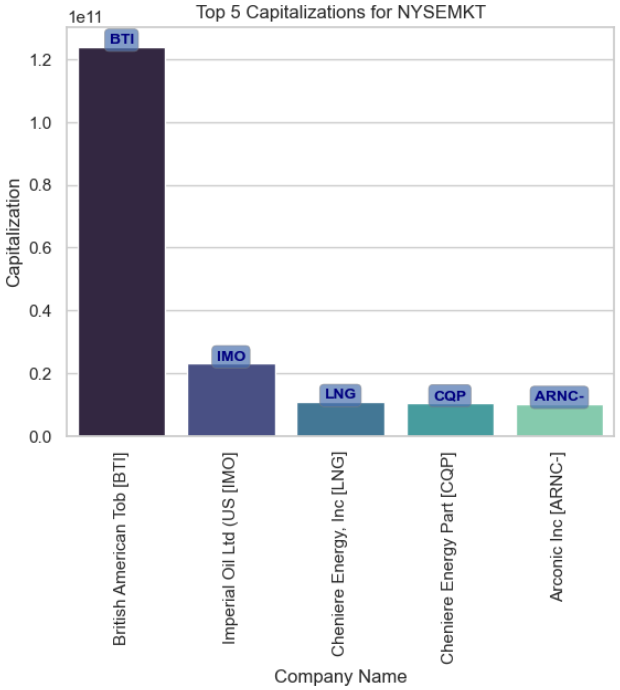
(b) OTCMKTS



(c) NASDAQ



(d) NYSEARCA



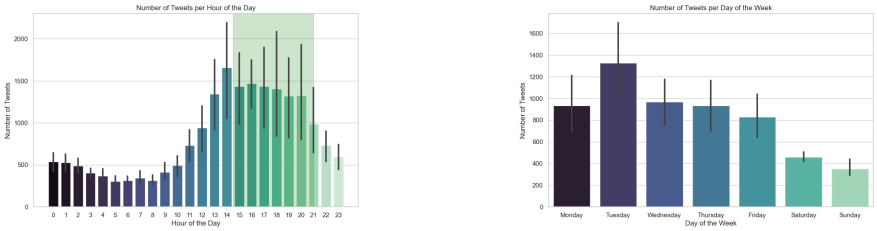
(e) NYSEMKT

Figure 7: Top 5 capitalizations for each exchange

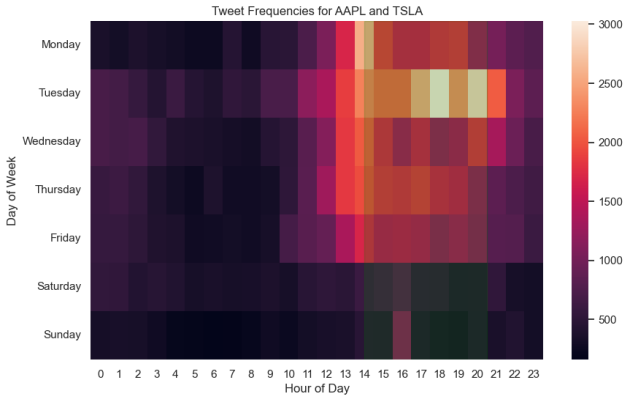
Task 6

Task

Time series movement directions through time for individual stocks (at least 2). Choose companies you are familiar with. Try to explain the reason behind these directions from real world news.



(a) Number of tweet per hour of the day(b) Number of tweet per day of the week



(c) Tweet frequencies per day of the week and hour of the day

Figure 8: Time series movement for number of tweets

- In figure *8.(a)*, we can see a peak at the trading hours (highlighted green) and even an hour before trading hours. A psychological analysis may be that the traders will be checking tweeter an hour before trading to see how to market may be going, or maybe those interested in selling their shares may try to persuade buyers to buy their share of stock.
- In figure *8.(b)*, we can see a drop on the days off (Saturday and Sunday). Tuesday seems to have a peak as well, but I lack knowledge of the market to be able to analyize why!
- In figure *8.(c)*, we can see the joint distribution per day and week. The most frequent trading time is the trading hours in the working days. Tuesdays seem to be very heated (and I do not know why)!

Task 7

Task

Co-occurence of various stocks in the same tweets.

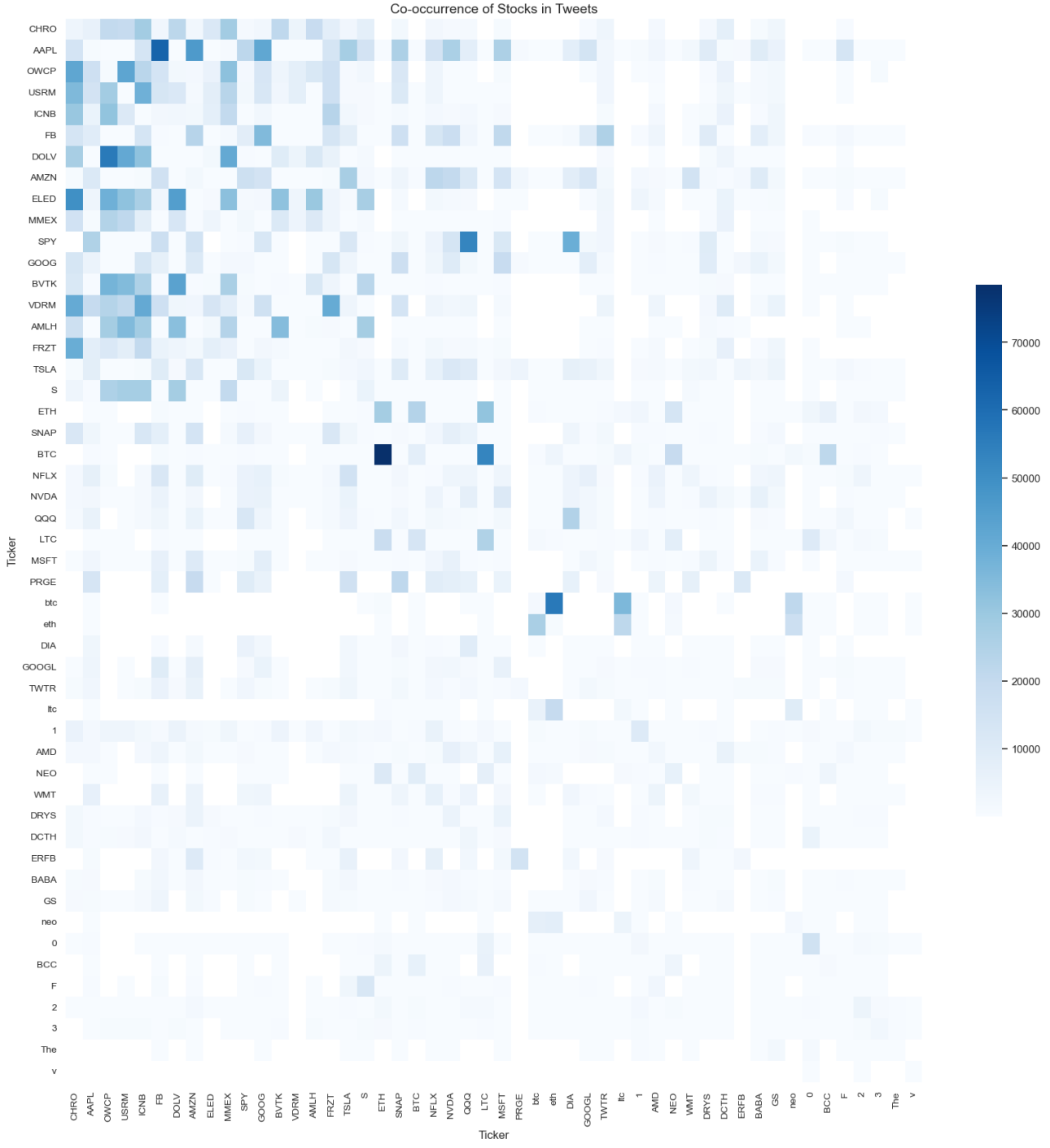


Figure 9: Co-occurence matrix of various stocks in the same tweet

The process is very time-consuming. I had to get the top 50 most common tickers to be able to computationally handle the task. It seems **ETH** and **BTC** occur the most together. Another contestant to that is **AAPL** and **FB**.

Phase 2

After reading the data and processing the time-series data, I was ready to begin the training process. The problem was that the data size was too huge for me to work with properly (as I tend to do the work with trial and error), so I sampled the data and worked with a $N = 100,000$ sample dataframe. All of the models were trained and evaluated this way.

I also had to use **Kaggle** for this Phase (and the next), because my device did not have a usable GPU for training the models; I only had a GPU there. Hopefully the file directories will make more sense when you read the notebook now.

Cleaning the data

Task

Clean the data.

- Removing duplicate values and useless data (both columns and rows).
- Handling upper/lower case, etc.

First the links were removed. Then, the hashtags were separated from the text data, and punctuation was removed from the tweet. All non-english characters were removed, and tokenized with the stop-words removed. The stop-words were obtained from nltk.stop_words.

Bag Of Words

Support Vector Machine (SVM)

Train accuracy	86.71%
Validation accuracy	52.25%
Test accuracy	51.77%

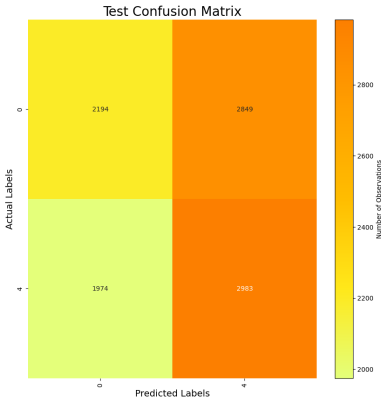


Figure 10: SVM + BOW Confusion Matrix

Random Forest

Train accuracy	96.43%
Validation accuracy	52.12%
Test accuracy	52.32%

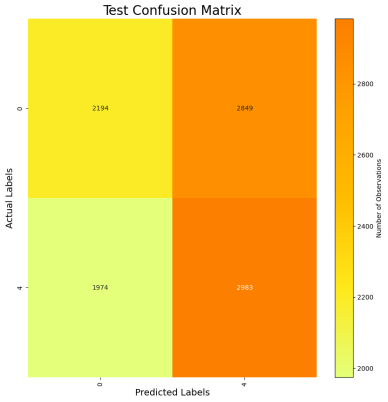


Figure 11: Random Forest + BOW Confusion Matrix

The Random Forest already looks more promising, judging by the confusion matrices. The main diagonal of the confusion matrix seems brighter. It seems sentiment analysis is not linearly separable (what a surprise).

TF-IDF

Support Vector Machine (SVM)

Train accuracy	94.92%
Validation accuracy	77.62%
Test accuracy	77.27%

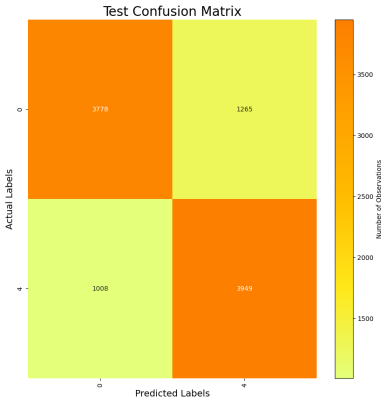


Figure 12: SVM + TF-IDF Confusion Matrix

Random Forest

Train accuracy	99.46%
Validation accuracy	75.63%
Test accuracy	75.54%

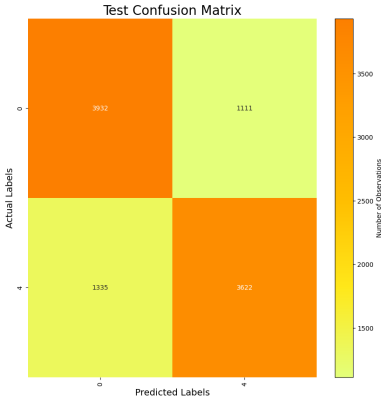


Figure 13: Random Forest + TF-IDF Confusion Matrix

The Random Forest already looks more promising, judging by the confusion matrices, as the main diagonal of the confusion matrix seems brighter. TF-IDF is already performing better than the BOW method.

SpaCy

There is not much to say here really, except that I used the 'textcat' pipeline, already pre-prepared for our purpose (binary text-classification).

Validation Set Performance:

	precision	recall	f1-score	support
NEGATIVE	0.79	0.79	0.79	4980
POSITIVE	0.79	0.79	0.79	5020
accuracy			0.79	10000
macro avg	0.79	0.79	0.79	10000
weighted avg	0.79	0.79	0.79	10000

Test Set Performance:

	precision	recall	f1-score	support
NEGATIVE	0.78	0.79	0.78	4965
POSITIVE	0.79	0.78	0.78	5035
accuracy			0.78	10000
macro avg	0.78	0.78	0.78	10000
weighted avg	0.78	0.78	0.78	10000

Figure 14: SpaCy classifier metrics

Fine-tuning HuggingFace: Distil-BERT

For the next section, I chose not to use the GPT API (because I did not have enough time to calculate the token and chose to accept risk) and instead used a fine-tuned version of the Distil-BERT model from HuggingFace. The reason I used SpaCy, was the fact that I did not like having two BERT models for this project.

Train accuracy	99.06%
Validation accuracy	78.95%
Test accuracy	77.85%

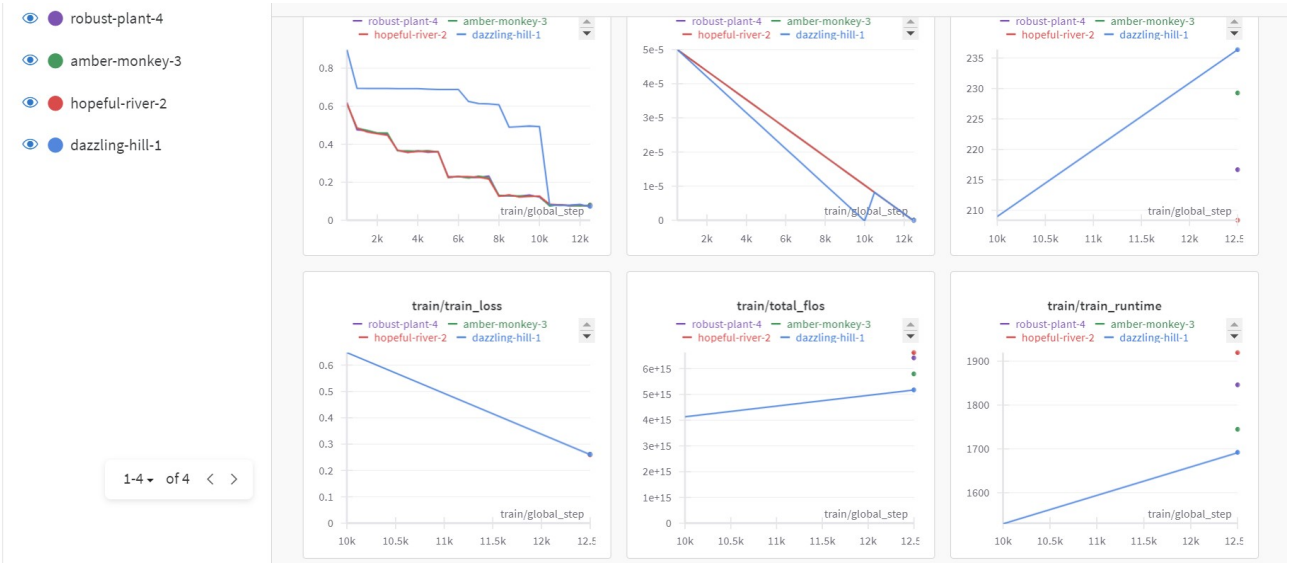


Figure 15: HuggingFace workspace dashboard (robust-plant-4 is the final model, purple)

Comparison of Models

Accuracy-wise, it can be easily shown that the last two models are better. That was not too far-fetched, as BOW is weak and RF and SVM and other traditional methods are not very practical in replicating the attention mechanisms required for NLP and Sentiment Analysis. They *can* work, but deep methods with attention mechanisms (or other adjacency methods) are better at doing this special task; as the task is not neccessarily simple.

The SpaCy model and the Distil-BERT model were saved and used for the next phase, as they yielded the best results.

Phase 3

Applying Classifiers

The main database is **very large**. I had to sample it again to achieve my results in reasonable time. I again used $N = 100,000$ samples, applied the CashTag($) extraction and used both of the models on the preprocessed tweet texts.

Pearson and Spearman Correlation

Task

Using different correlation measures (at least 2 and you can use libraries), find the correlation between the sentiments of each CashTag and its value (from the financial data in the companies dataset) in a time interval.

As the task required a 'time interval', I used the middle month as a time to calculate the correlation. I used the 'scipy.stats' library for these correlations.

```
Pearson correlation between capitalization and DistilBERT sentiment for the middle month: -0.00522908478452948
Spearman correlation between capitalization and DistilBERT sentiment for the middle month: -0.0125210023149065
Pearson correlation between capitalization and spaCy sentiment for the middle month: 0.004165189124265656
Spearman correlation between capitalization and spaCy sentiment for the middle month: -0.029781320887004303
```

Figure 16: Spearman and Pearson correlations for each model between sentiments and capitalizations

Task

Using different correlation measures, find the correlation between change of sentiments of CashTag and the number of tweets related to that CashTag in the time series (By time series of tweets, we mean the number of tweets per given time that was calculated in the first part).

```
Correlation between tweet count and DistilBERT sentiment: 0.003474215296178894
Correlation between tweet count and spaCy sentiment: -0.004355228140295381
```

Figure 17: Correlation for each model between sentiments and retweets-per-tweet

I may or may not have forgotten to use two correlation measures…

The correlation amounts are very small, in general. Some of them are negative as well. This *may* have a meaning, but the correlations I have arrived at are not meaningful/significant enough for conclusions.

Classifier Conflicts

Task

Find examples where the classifiers do not agree on the sentiment of the tweet and analyze the results and discuss where does each classifier make mistakes.

Text	**Distil-BERT**	**SpaCy**
At Home Group Inc $HOME Forecasted to Earn Q4 2018 Earnings of $0.30 Per Share https://t.co/VDycVabjjq	NEGATIVE	POSITIVE
$OCRX Company has begun dosing of the first patients in Part Two of a Phase1/Phase 2a clinical trial of oral OCR-0… https://t.co/csMCRZ2oBq	NEGATIVE	POSITIVE
SG Americas Securities LLC Cuts Position in Vector Group Ltd. $VGR https://t.co/7Rp2CfVfOo	NEGATIVE	POSITIVE
Dimensional Fund Advisors LP Has $15.57 Million Position in Capital City Bank Group $CCBG https://t.co/x5hl85YeKQ	POSITIVE	NEGATIVE
Startin the week off with a bang. $RGNX https://t.co/AX5RTRcGNh TV_TradingIdeas	POSITIVE	NEGATIVE

Table 1: Classifier disagreement examples

For the first, second, and the fourth examples, I personally cannot decide on the sentiment as a human. However, for the third and fifth examples, I think the SpaCy model is slightly weaker than the Distil-BERT model. For instance in the fifth example, I think the model thinks that "bang" is a bad word because of guns and weapons; while the phrase "starting off with a bang" is a totally positive sentiment. Distil-BERT has managed to capture that.

Discussion

Soloution

Discuss transfer learning and fine-tuning and how they can be used to improve the overall effect of the project. Use your limited results from the first part.

Transfer Learning and **Fine-Tuning** are powerful techniques in machine learning that can significantly improve the performance of a model, especially when dealing with tasks where the available data is limited.

Transfer Learning is a technique where a pre-trained model, which has been trained on a large-scale dataset, is used as a starting point for a related task. The idea is to leverage the knowledge that the model has already learned from the large dataset to perform the new task. This is particularly useful when the new task has limited data, as it allows the model to generalize better.

For example, in the context of this project, a pre-trained language model like BERT or Distil-BERT, which has been trained on a large corpus of text, can be used as a starting point for sentiment analysis. The pre-trained model already understands the semantics of the language to a great extent, which can be leveraged to understand the sentiment of the tweets.

Fine-Tuning is a process that adjusts the pre-trained model to the new task. After initializing our model with the pre-trained weights, we continue training the model on our specific task (like sentiment analysis in this project). During this process, the model learns to adjust its weights and biases to specialize on the new task.

In the context of this project, fine-tuning can be used to adapt the pre-trained language model to the specific language and style used in the tweets. This can lead to a more accurate sentiment prediction.

In summary, transfer learning and fine-tuning can greatly improve the performance of the model in this project by leveraging the knowledge from large-scale pre-training and adapting it to the specific task of tweet sentiment analysis. This can lead to more accurate sentiment predictions, which in turn can improve the correlation analysis between tweet sentiments and company values.

We can check how the sentiment and amount of tweets change drastically in a given time to loosely predict how the stock and capitalization of companies will in turn, change. This is especially useful in trading and forecasting the market. For example in the first part the time series plot of tweets increase when the market reports arrive, and we can see peaks. The sentiments may be mixed, however. Sentiments do not perform that well when used for predicting the market (as our analysis showed). This is especially important as piggybacking may occur and bots may be involved to change the public views on some stock so that people will buy them.

Bonus Time!

I only did the web-crawling part. I chose “divar.ir” to crawl because it seemed easier.

Web-Crawler

The results are available as ‘divar.csv’. Because of the variance between loading times and the internet connection and etc…, I used many try/catch statements and that is why some of the fields inside the .csv file is empty. While doing the web-crawling, the website gave me a 408 time-out because there were too many requests in such little time. A part of the dataset may be affected by that.

Scrolling to the bottom was implemented by selecting the body tag, which usually is located at the bottom of documents. The program will click on links and open them on new tabs, extract the data, and then close it. Waits were scattered here and there to account for the loading times. The whole process took 5 minutes, and only yielded 150 rows. By putting more time (which I do not have for this project), the dataset can be scaled up. The amount of empty cells can be reduced by waiting more (which means putting more time, and I do not have that luxury).

	title	descriptions	row_dict	info	url
0	اموزش تخصصی فارکس	[]	{}	گارانتی🖌\nآموزش تضمینی ...لایو ت✌\n🖌بازگشت وجه	https://divar.ir/v/%D8%A7%D9%85%D9%88%D8%B2%D8...
1	لباس عروس عربی برند مزون هستی	[در حد نو, ۱۹,۶۰۰,۰۰۰ تومان]	{'زنانه/مردانه': 'زنانه','وضعیت': 'در حد نو',...	یکبار-n\مناسب سایز ۴۲ تا ۴۸- ... پوشیده شده، درحد	https://divar.ir/v/%D9%84%D8%A8%D8%A7%D8%B3-%D...
2	تشریفات در عمارت مجلل اختصاصی بی واسطه	[]	{}	مژده مژده افر بسیارسیار ...ضمن عرض سلام و n\ویژه	https://divar.ir/v/%D8%AA%D8%B4%D8%B1%DB%8C%D9...
3	عسلی مبل	[نو, ۲,۷۰۰,۰۰۰ تومان]	{}		https://divar.ir/v/%D8%B9%D8%B3%D9%84%DB%8C-%D...
4	خدمات چاه بازکنی تهرانپارس پیروزی ... نارمک رسالت	[...لوله بازکنی، حفر]	{}		https://divar.ir/services/craftsmen/ncuUA
...	...	...	...	...	...
153	در حدنو بدون هیچ خط وخشی	[در حد نو, ۳,۰۰۰,۰۰۰ تومان]	{'وضعیت': 'در حد نو','مایل به معاوضه': 'نیستم'...	سالم سالم در حد نو	https://divar.ir/v/%D8%AF%D8%B1-%D8%AD%D8%AF%D...
154	سفارش عروسک دستدوز	[نو, ۲۱۷,۰۰۰ تومان]	{'وضعیت': 'نو', 'مایل به معاوضه': 'نیستم',... 'قی	سفارش انواع عروسک دستدوز ...پذیرفته م\گلدوزی شده	https://divar.ir/v/%D8%B3%D9%81%D8%A7%D8%B1%D8...
155	کت وشلوار جلیقه همراه کفش بچه گانه	[نو, ۸۰۰,۰۰۰ تومان]	{}		https://divar.ir/v/%DA%A9%D8%AA-%D9%88%D8%B4%D...

Figure 18: The extracted dataframe

End of Project Report