Final Project Report

FE-595 Group 3

NLP-Based Trading Strategy and Prediction

Qiuchen Ma, Kairu Zhang, Pengfei Liu, Yuyang Fei

Dec 18, 2020

1. Introduction

1.1 Background

Human preferences are practically unpredictable and we have invented sciences like psychology and sociology to help us study these things. With data freely available, Data scientists can do the same thing with the best psychology tool – Twitter. Twitter is one of the largest microblogging and social networking service companies in the United states or even in the world. It has hundreds of millions monthly active users, and 340 million tweets are posted every day. These tweets include financial news from companies like Bloomberg, Wall street Journal, and financial reports or researches from banks, funds, fintech companies, for example, Morgan Stanley, Chase, and Two Sigma. These official information from financial industries are collected by users as soon as it has been released and they can quickly create a reaction responding to the news. Due to this characteristic, a lot of companies are now using sentiment analysis in the stock market to predict the market trend or movement of a particular stock. Sentiment analysis is the process of detecting positive or negative sentiment in text. It's often used by businesses to detect sentiment in social data, gauge brand reputation, and understand customers.

Speaking of the stock trading price, it is directly affected by supply and demand, however supply and demand are affected by many factors inside or outside the stock market. For example, a company's operating condition, reputation, development prospects, dividend distribution or even a company's external economic system, government's political or economic policy. These factors could have positive or negative influence on the stock market. And trading volume, trading methods, and trader composition could cause short-term stock price volatility. In addition, artificial manipulation from investors, hedge funds, investment institutions can also cause stock prices to fluctuate. The latest information from twitter provides great chances for short-term or long-term investment. For example, in mid 2020, the stock market gave immediate reactions to news about the funds rate cut by the Federal Reserve System. By collecting Tweets posts, we are able to perform a NLP-based Strategy, using these short-term price fluctuations to parse tweets into tokens, perform sentiment analysis and then hold positions before or during the expected price fluctuations.

1.2 Purpose

For our project, we are going to build a model that extracts recent 5 years twitter data of a selected stock, then performing sentiment analysis. Doing sentiment analysis of the tweets enables us to calculate numerical values of subjectivity and polarity. This could help us to better understand this Twitter account in terms of the language that is being used. Combining this with additional information about likes and comments can be very useful from a marketing point of view and can enable us to find some correlation between subjectivity, polarity and the engagement of the users for a specific Twitter account. From this, we can study the correlation between twitter and stock's price.

2. Deployment

Choosing a stock, finding related twitter data, performing sentiment analysis and comparing the sentiment analysis to stock price prediction.

Basic Guideline:

Step 1: Download related tweets and equity/ETF/index prices

Step 2: Data cleaning and visualization, Twitter word frequency statistics

Step 3: LDA modeling & test score

Step 4: Perform sentiment analysis, analyze the relationship between sentiment score and stock price & log return

Step 5: Prediction by sentiment score(rolling average as one month period)

For the project, our team choose Dow Jones Industrial Average Index as tag and date from 2015-December to 2020-December, following as detailed steps:

2.1 Collecting Tweets & Dow Jones Index data

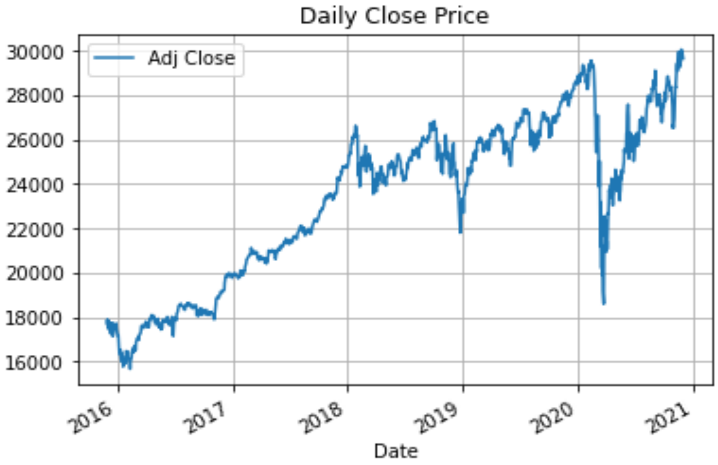
First, we chose to use python library 'Snscrape' to scrape tweets, it provides us a good solution to collect data history without official twitter API limitation. And by giving Some related keywords, we can define the search parameters we want to get data. Then we chose the Dow

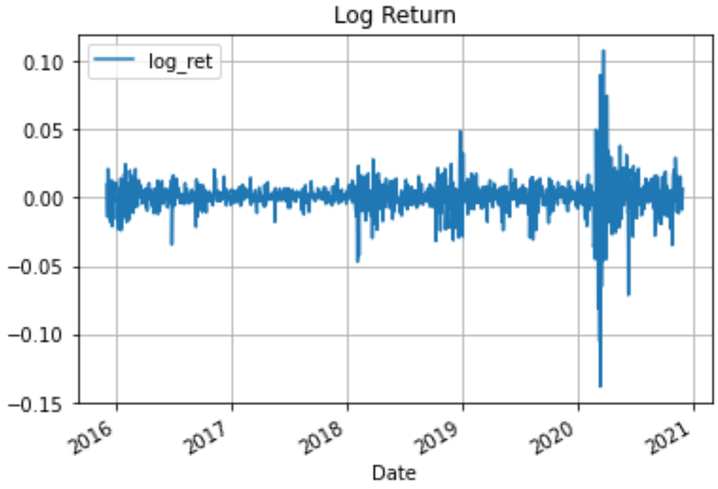
Jones index as the tag we are interested in, as well as the time period we need, which is from 2015-12-01 to 2020-06-01, and output data as external files.

We use Yahoo Finance as our data source and download information directly through Python. And then we visualized historical daily close prices and log returns.

2.2 Data cleaning and Visualization

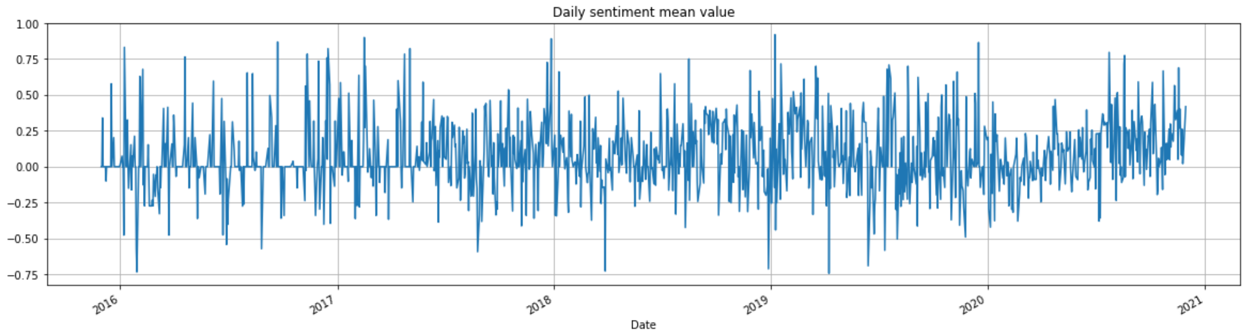
For Index data, here are graphs of daily close prices and log returns for Dow Jones Index from the year of 2015 to 2020.





And for tweets data, we need to clean these information in order to provide interpretability for our models and sentiment analysis.

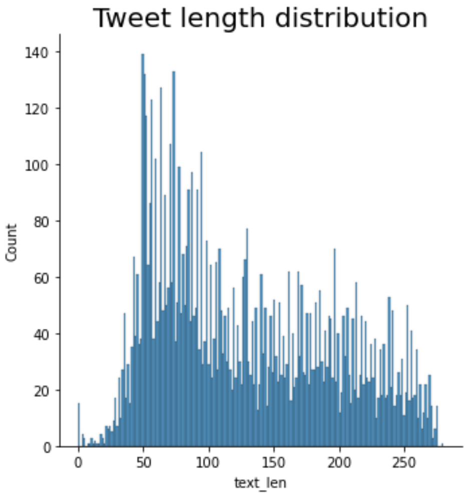
First, we constructed average sentiment on each day, then calculated the sentiment mean value and visualized it.



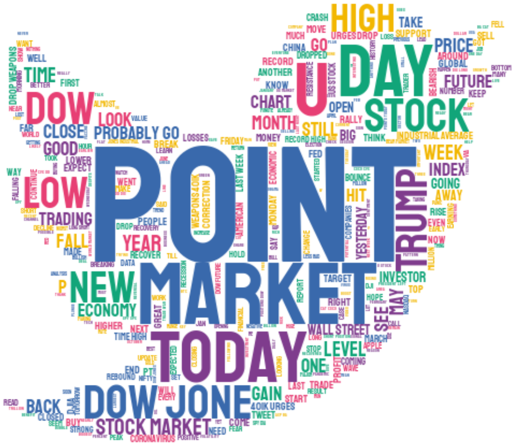
Then performing sentiment analysis requires us to clean up tweets from an unnecessary data, we created clean up tweets function that will:

- remove mentions
- remove hashtags
- remove retweets
- remove urls

We define some lambda functions to delete the letters and characters that match the expression we want to delete, then we perform some basic analysis of Twitter data by checking the composition of tweets, for example calculated the average length of tweets and plotted the distribution of tweet text length.



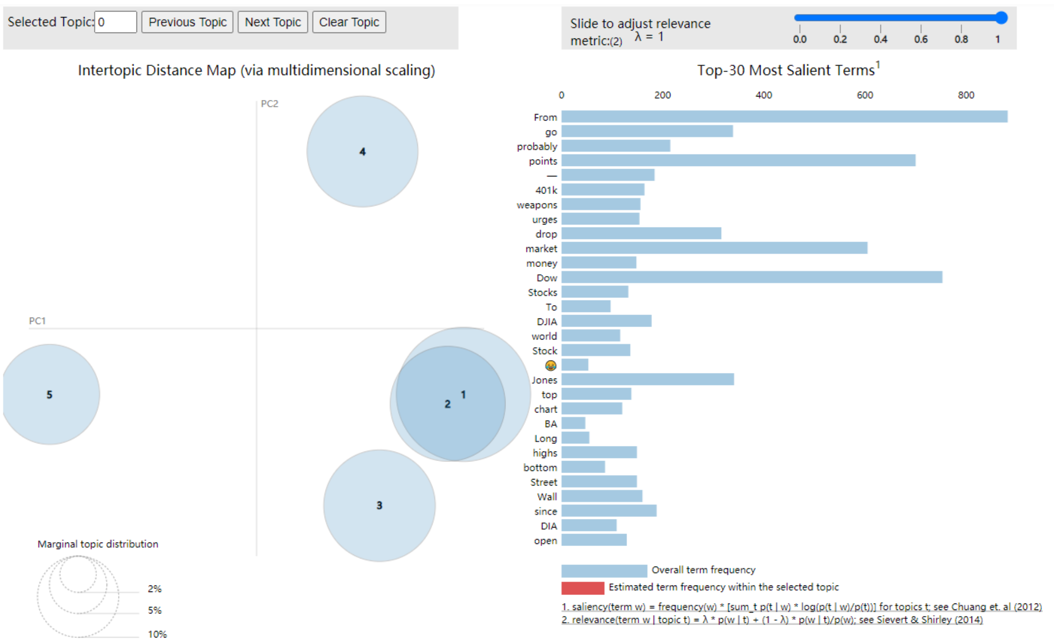
Next step is cleaning the stop word, we use NLTK library function stopwords, easily delete all stop words from the data.Then we need to consider the word order, we use N-grams to test the importance in tweets dataframe. And from the Tweet length distribution, we can find that most tweets' lengths are over 50 words, thus knowing that the word order is definifty important in getting correct sentiment analysis scores. The graph listed below is the visualization of our hot words.



2.3 LDA modeling & Test score

LDA, as known as Latent Dirichlet allocation, is a statistical model that allows sets of observations to be explained by unobserved groups, which explains why some parts of the data are similar.

In addition, in order to understand the distribution of tags and keywords, we chose to use 'pyLDAvis' to launch an interactive widget, making it ideal for use in python. After observing the result graphs, we found that the LDA model does a good job of capturing the significant keywords and their constituent words in Twitter data.



1. saliency(term w) = frequency(w) * [sum_t p(t | w) * log(p(t | w)/p(t))] for topics t; see Chuang et. al (2012)
2. relevance(term w | topic t) = λ * p(w | t) + (1 - λ) * p(w | t)/p(w); see Sievert & Shirley (2014)

To test the LDA model , we chose to compute the Coherence Score, the result is 0.2926706. The results vary from different tags we choose.

2.4 Sentiment Analysis

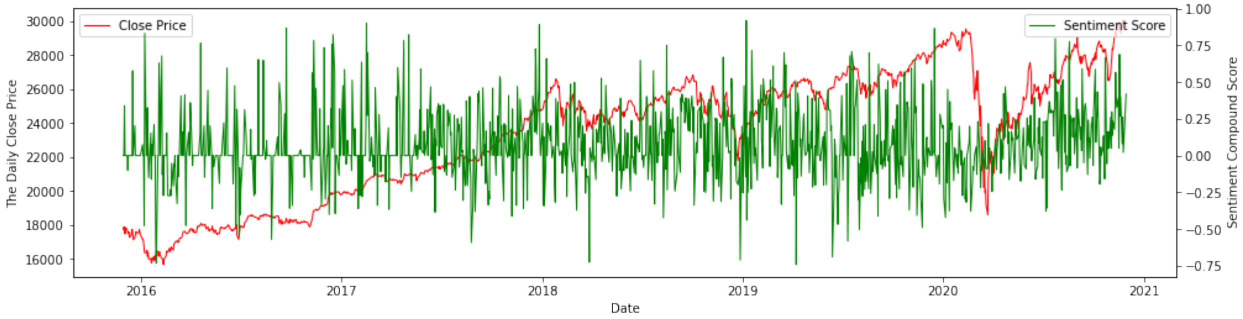
We performed sentiment analysis to the clean tweets and got the sentiment scores. The compound score is a metric that calculates the sum of all the lexicon ratings which have been normalized between -1(most extreme negative) and +1 (most extreme positive).

- positive sentiment : (compound score >= 0.05)
- neutral sentiment : (compound score > -0.05) and (compound score < 0.05)
- negative sentiment : (compound score <= -0.05)

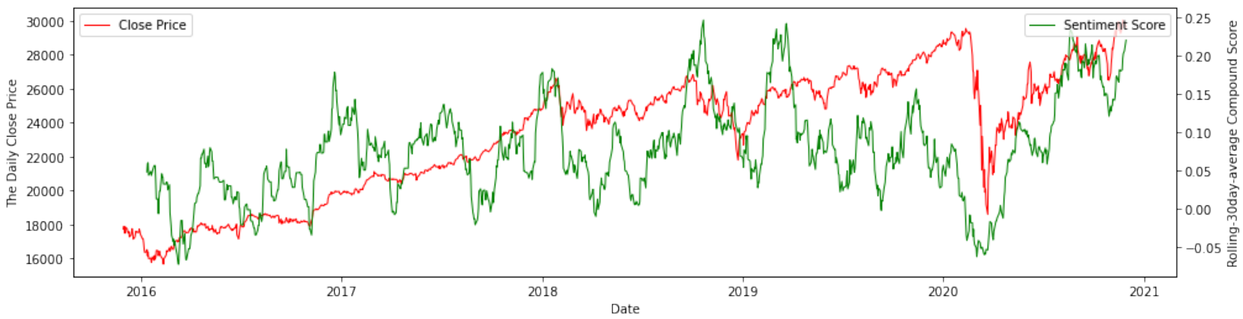
With the information above, we plotted the sentiment score and stock's close prices to study the correlation of these two and this can help us to perform the stock price prediction for the next step.

2.5 Prediction based on Sentiment score

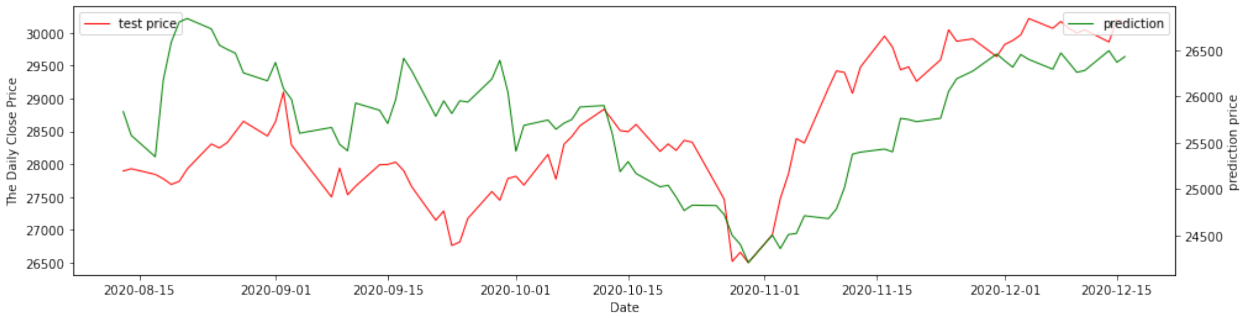
Moving on to the final step, with the collection of the relationship between sentiment scores and stock's prices, we can perform a prediction on stock prices and compare that with the actual price to see how well our model performs. First, we use the compound to predict the stocks's close price. The result did not seem so well.



We can not graph a useful relationship from this comparison. We decided to switch the object to a 30-day rolling average compound sentiment score to improve the prediction. This time, the relationship between the compound score and close price seems much obvious.



Before 2020, the compound sentiment score has higher volatilities. It has the same direction (up & down) as the price. After 2020, they seem to have a common trend and similar volatility. Based on this, we build a linear model and performed a prediction.



This time, the prediction value and actual market price are very close. Especially after October, the model fits very well. In this way, you can have a simple way to have a research on whether there exists a relationship between the tag and the equity/ETF/index you select.

Reference

https://en.wikipedia.org/wiki/Latent_Dirichlet_allocation

https://en.wikipedia.org/wiki/Twitter

https://towardsdatascience.com/evaluate-topic-model-in-python-latent-dirichlet-allocation-lda-7d57484bb5d0#:~:text=But%20before%20that%E2%80%A6-,What%20is%20topic%20coherence%3F,are%20artifacts%20of%20statistical%20inference.

https://www.reddit.com/r/learnpython/comments/jlgmnm/scraping_tweets_using_snscrape/