Modeling Stock Exposure from Twitter

Nick Tyler (Group 13)

Github link with slides and code: https://github.com/tylernickr/twitterExposure

Introduction

For portfolio managers in the financial industry, picking stocks or choosing which stocks to own relative to their benchmark is often the most critical part of the job. While much attention is given to the returns that portfolio managers are able to deliver from their portfolio, an important consideration when constructing these portfolios is the idea of exposure to different factors. Stock exposure is essentially the amount that a particular stock or a basket of stocks changes value in response to changes to an external factors. A stock with high exposure to oil is likely to move in a way that is highly correlated with the price of oil, where a stock with low exposure to oil would likely be stable as the price of oil fluctuates. These external factors that stocks can have exposure to can be anything and range from China's economy to the lumber industry in the Northwest region of the US.

The aggregate of stock exposure in a portfolio essentially represents the risk profile of the stocks within the portfolio and is a major factor in how the stocks of that portfolio are chosen. A portfolio managers who believes there is high uncertainty in the oil industry may choose to build a portfolio with low exposure to oil to partially protect against this uncertainty. Exposure is often more important than expected returns in stock selection for these reasons.

However, the factors that a stock are exposed to are not all obvious. While major factors such as the price of oil are often clear, more subtle factors such as the impact of labor unions or a company's environmental impact are not always obvious. The primary goal of this project is to create a mechanism to extract out the different factors that contribute to the movement of a stock price. This will be done by extracting the broad topics that are being talked about in the business world, and modeling the amount a stock price moves against the topics being discussed at the time of the price movements. Because of its transparency and increasing prominence in the financial world, the business discussion data will be pulled from Twitter, specifically from tweets made by prominent players in the financial world. The idea is that Twitter discussion from these types of accounts approximates the true topics of discussion happening in the financial world in general, and that we can draw information from the timing and topics of tweets.

Overview

This project will consist of four main phases. The first phase of the project will be data gathering. This will involve going out to two main sources of data, Twitter and AlphaVantage, and retrieving the necessary data for the project. This data will largely consist of tweets, their content, and daily stock data.

The second stage of the project will be preprocessing of the data. For the twitter data, this will involve text processing. This will include tokenization of the content of the tweets, removing stopwords, and stemming the tokens of the related tweets. For the AlphaVantage stock data, this will involve calculating the daily percentage change in the stock price from the delta between two dates of closing prices.

The third stage of the project will involve the development of a model to determine whether or not a tweet is relevant to a particular company. Because the overwhelming majority of tweets will not be about one particular company, it is important to be able to have a filter to remove some of the unrelated tweets from any modeling of a particular company's exposure. In order to accomplish this, a separate set of tweets will be pulled from Twitter using a hashtag created from a company's name or nickname such as "#walmart" for WMT or "#pandg" for Procter & Gamble. This data will act as data labeled as being relevant to the company that can be used to make a model to determine if future tweets without hashtags are relevant to a particular company.

The final stage of the project will be to create an interpretable model of a company's daily stock movement. This model will be created from features extracted from tweets that have been input to the model. Inputs will be filtered and weighted relative to their relevance to the stock. The various features of this final model, such as its coefficients, will be able to tell us information about what contributes to the movement of the stock.

Data Gathering

For this project, the scope of the stocks that will be analyzed are the thirty stocks that make up the Dow Jones stock index.

These stocks include:
GS, AXP, KO, DIS, V, CAT, PG, WBA, UNH, DOW, WMT, UTX, MCD, MSFT, NKE, AAPL, HD, BA, INTC, XOM, MMM, VZ, CVX, IBM, JPM, JNJ, PFE, CSCO, MRK, TRV

Data was gathered entirely from two sources. Stock data was gathered from the free version of the AlphaVantage stock price API. Twitter data was gathered from the free version of the Twitter API.

The stock data gathered was straightforward information about the price of a given stock. The data contained the open, close, high, and low prices of the stock. For the purposes of this project, only the closing price was used. This data stretched back to 1999, but only the most recent two years will be used.

The Twitter data gathered can be placed into three different groups. The first group was tweets from prominent business users and business Twitter accounts. This ranged from breaking news accounts such as CNN, business news channels such as CNBC, individual portfolio managers, and market movers such as the president or Speaker of the House for the United States. These tweets are the tweets that will be acting as the input into the final model that exposure will be derived from.

The next two groups of tweets that were pulled from the API are tweets about the specific companies listed above, as well as completely random tweets pulled from the API. The tweets about specific companies were found by searching for tweets by hashtag. The hashtags used to find tweets for a particular company were largely the company's name or any nicknames that it has. Examples for GS (Goldman Sachs) would be #goldmansachs or #goldman. These tweets were considered as labeled data with the label being that the tweet is a tweet about that particular company. The random tweets were pulled from the "Sample" API endpoint which returns random tweets. These tweets also acted as labeled data with the labels being that the tweets were not about a specific company. These two groups of tweets acted as inputs to the first model, which is used to determine whether or not a particular tweet is related to a specific company.

Data Preprocessing

Several preprocessing steps once the data had been retrieved were necessary for the data to be used in modeling. Preprocessing the stock data was relatively straightforward. The data was initially in the form of date-price. This was modified to be in the form of date-price change. The price change was calculated by subtracting a day's closing price from the previous day's closing price, and then dividing by the previous closing price to get the number in a percentage. The majority of % change values that were calculated and used as labels were less than 2% in absolute value.

Preprocessing of the Twitter data was more involved and required additional steps. First the content of the tweets was extracted, and special characters were removed from the content. Next the remaining content was tokenized, with each word representing one token. A word was considered to be anything between whitespace characters. Next these tokens had stopwords removed from them, and had misspelled words removed from them. This does not include the removal of common proper nouns such as “IPhone”. Lastly, the tokens were put through a stemmer to reduce the dimensionality and collapse similar words into one token.

These tokens were then transformed into word count vectors, with each row of the vector representing a document or tweet, and each column of the vector representing a word. The values were the number of times that a word appeared in a tweet. Most of the values in this vector were either 0 or 1. Once the tweets had been transformed into word count vectors, and LDA model was applied to them to reduce the dimensionality from ~10,000s down to 100 LDA topics. This will make running a model of the data more realistic and should improve performance.

One point of note is the amount of “labeled” tweets that were pulled from the Twitter API. The free Twitter API endpoint makes available less than 1% of all tweets from the past thirty days.. Because many of the companies in this analysis are not frequently hashtagged, this resulted in a number of companies coming back with “labeled” tweet counts under 1,000. Because this is such a small number, the companies where it was not possible to pull more than 1,000 will be excluded from the analysis because models created off this data will likely be unreliable.

Tweet Relevance Model

The twitter relevance model was constructed using the two groups of labeled tweets described above. Data post LDA processing was used, with the tweets retaining their original labels as the response variable. For this task, a model was created for each individual company that predicted if a given tweet was either a random tweet or one relevant to the company. This was done because it is entirely possible that one tweet may be 100% relevant to a number of companies. A model that predicted which company a tweet was most relevant to would miss this information or would dilute the effect of tweets that were relevant to more than one company.

For each model, the training data was made up of half tweets relevant to the company and half tweets from the random tweet data, creating balanced training data. Data was then testing using 5-fold cross-validation on 4 different models. These models included

a Naive Bayes model, a logistic regression model, an ADABoost model, and a Random Forest model.

The average accuracies over all the companies for each model were:

- **Naive Bayes: 85.11%**
- **Logistic: 84.86%**
- **ADABoost: 84.85%**
- **Random Forest: 84.84%**

Because the aggregate results for each model were extremely similar, the logistic regression method for each company was chosen to be the model that would be used during the next phase of the project. This decision was driven largely from the advantages of logistic over the other models, such as simplicity, explainability, and its ability to make predictions of probabilities rather than simply classifications. Because the likelihood that a tweet is relevant to a particular company will be important in the final model, the last of those advantages is particularly important.

Only stocks where the logistic regression model was able to achieve over 80% accuracy were included in the analysis. The reasoning behind this is that without accurate predictions of relevance, subsequent analysis is likely to be unreliable and not worthwhile.

Final Exposure Model

To create the final model, additional preprocessing steps involving the above relevance model were used to once more transform the data. For every day, each tweet was processed by the relevance model to determine how relevant it was to a particular stock. The features of the tweets were then multiplied by this relevance prediction before being aggregated on a daily basis. This resulted in tweets that are 50% relevant to a name having half as much influence on the model input as tweets that were 100% relevant, and excluded irrelevant tweets. Tweets from the same day were added up feature-wise, and then normalized so that each day would have the same sum of values regardless of the amount of tweet data that was processed for that day.

The result of this preprocessing was data that contained one vector per day, with the vector essentially representing the proportions of the 100 LDA topics that were contained within the days aggregate data. Every day of this data was then passed into a standard linear model, with the response variable being the absolute value of the % change of the stock's price on the same day. The resulting coefficients from the models

were then transformed backwards from LDA vectors to individual words, and the coefficients from each word represent the final results of the model. The final results are listed below. (Note: These results contain only stocks where the relevance model was able to achieve 80% and that had a sufficient number of tweets as data to model agains). The top 10 words for each stock are listed alongside their coefficients.

Results:

BA
cockpit: 59.9735
film: 44.5506
seen: 17.8834
airbu: 15.5023
save: 13.209
aviat: 13.1904
air: 13.1527
30: 13.01
hope: 12.1458
giveaway: 11.01

WMT
walk: 59.6552
know: 49.1175
support: 46.0081
anyon: 40.5524
pic: 26.8303
race: 19.8468
site: 17.7868
smile: 16.7154
reveal: 15.9311
believ: 13.5009

DIS
kid: 300.3857
start: 154.3203
thought: 135.9112
hate: 119.5582
fight: 103.6547
need: 96.3888
special: 74.6914
class: 63.6087
take: 45.2403
one: 44.6747

CAT
educ: 4.01
toy: 4.01
rattl: 4.01
start: 2.01
ask: 2.01
question: 2.01
manag: 2.01
march: 2.01
cat: 1.081
money: 1.0737

AXP
decor: 55.5299
art: 52.2292
wall: 31.8453
photographi: 12.01
artistri: 8.01
natur: 6.9395
sale: 4.9813
pack: 4.01
owe: 4.01
50: 4.01

KO
tri: 57.1703
know: 13.1625
send: 12.9005
anyth: 10.9344
say: 9.8258
someon: 8.3861
help: 7.64
work: 7.4239
still: 7.2824
get: 6.6312

AAPL
smart: 77.592
super: 70.9252
basic: 62.01
guid: 58.3927
tutori: 55.955
appl: 55.9448
provid: 55.8717
offici: 52.7613
other: 44.6511
user: 28.8308

INTC
right: 28.3866
hp: 27.01
human: 20.739
ass: 20.3239
million: 14.7995
like: 14.0819
idea: 11.5184

good: 11.2101
hit: 10.9605
time: 10.2188

MSFT
amaz: 146.2394
tonight: 145.01
god: 144.4404
connect: 132.9711
peopl: 91.2609
new: 85.7596
look: 72.8957
amazon: 65.3706
googl: 59.0134
anon: 52.01

CSCO
read: 25.2474
articl: 20.9761
90: 20.01
left: 15.5367
day: 14.9221
let: 13.755
packet: 12.4719
solut: 11.6594
tracer: 11.6247
made: 11.4154

V
act: 12.01
hold: 7.9631
peopl: 6.4858
job: 5.9875
like: 5.3556
messag: 5.01
hurt: 4.0881
relat: 4.01
mascot: 4.01
equal: 4.01

IBM
time: 327.8997
wrong: 35.4301
work: 27.0903
mean: 26.6325
program: 25.9791
place: 23.5472
turn: 21.7843
real: 21.5829
end: 20.8984
right: 17.5052

VZ
phone: 10.9553
worst: 10.01
line: 3.01
trick: 3.01
happi: 2.01
us: 2.01
god: 2.01
thank: 2.01
thanksgiv: 2.01
lie: 2.01

NKE
watch: 282.7475
2020: 160.4521
may: 146.9526
done: 128.4818
past: 94.3627
marvel: 66.01
studio: 54.5888
new: 53.9279
trailer: 46.01
okay: 40.9588

Discussion of Results

The results from the project were mixed at best, with some cases being relatively successful and others being mostly uninsightful or confusing. Some of the companies have items inside their top ten that are actually initially surprising but make intuitive sense. One example is that “watch” is at the top of the list of important terms for NKE (Nike), with the implication here being that Nike is possibly more dependant on the factor of watches than might be initially assumed. Some other interesting results include “smart” being in the list for AAPL (Apple) and “human” and “right” being in the list for INTC (Intel). These results imply that it is possible topics such as smart devices or human rights may be important for these two companies. It is also interesting that a number of the words appearing in the list for AXP (American Express) are artistic based words.

Overall however, the vast majority of words are either somewhat expected or simply reflections of topics that are clear talking points for individual companies. Boeing has results that are essentially all parts of a plane which is common sense for the company. There are also some results like Walmart which are mostly made up of common words. Additionally, the coefficients for words are all over the map, ranging from 1-2 up to a few hundred. It is unlikely that we can draw any conclusions from the coefficients and should focus more on the relative ordering of words.

It seems as if many of the shortcomings are due to lack of data or lack of quality data. The companies with more intuitive results listed above also are largely the ones where it was possible to collect the most data. Given that this project was largely an exercise in language processing and modeling, the presence of large amounts of data is extremely important, and lacking it for many of the companies clearly hurt the results. However, the few companies that did have interesting words appear in them imply that there may be some basis for this type of modeling. The free versions of the Twitter API only allow access to less than 1% of all tweets, but enterprise versions allow access to the complete history of twitter.

It is likely in a professional setting with access to this type of paid data, the models created would be more representative of the mainstream discussion points and would have produced superior results. While the project overall was not a large success, the few bright spots listed above indicate that this modeling may be an area to explore in a less resource constrained environment.