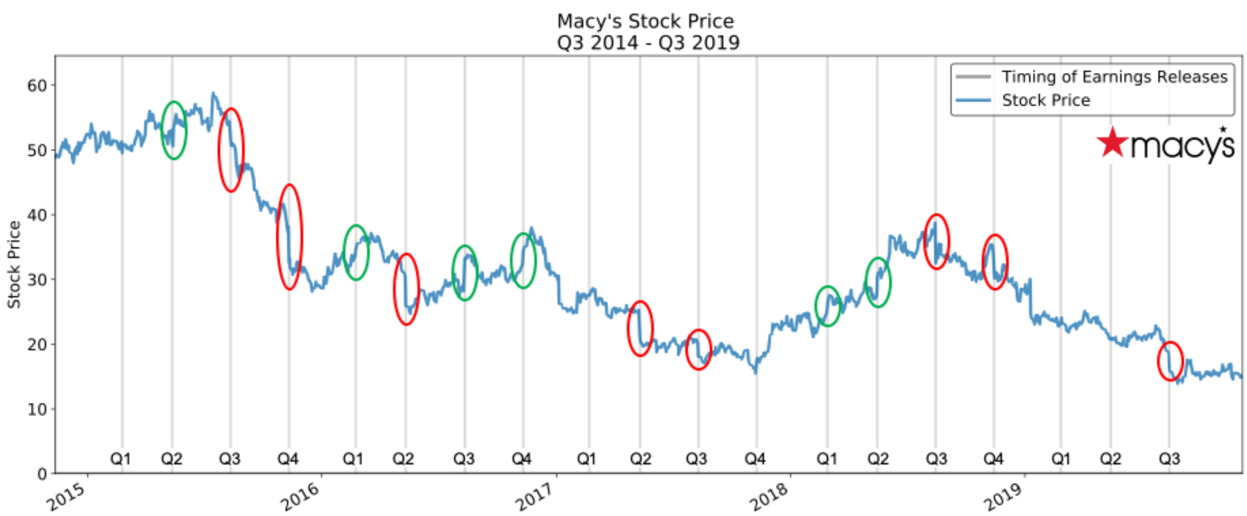
Project 5 Summary:
Predicting Stock Performance from Quarterly Earnings Conference Calls

Linda Ju

Domain

Motivation
Every quarter, public companies must report their earnings. Many companies also host a conference call in order to provide additional commentary and answer questions from Wall Street analysts.

These reports and calls have a strong influence on the stock market. Below is an example of Macy's stock price for the past five years. The gray bars indicate the timing of Macy's earning releases. As you can see, on many release days, the stock price could surge (in green) or plummet (in red).



For my project, I extracted features from the transcripts of department stores earnings calls with Natural Language Processing ("NLP") and built a classifier to recommend buy, sell, and hold investment decisions.

Methodology

Natural Language Processing

Pre-Processing
The dataset I used consists of 281 earnings call transcripts for JCPenney, Kohl's, Macy's, and Nordstrom as detailed below.

Department Store	Stock Ticker	Time Period	Number of Transcripts
JCPenney	JCP	2001 - Present	69
KOHL'S	KSS	2001 - Present	71
macy's	M	2003 - Present	70

Department Store	Stock Ticker	Time Period	Number of Transcripts
NORDSTROM	JWN	2002 - Present	71
			Total = 281

I web-scraped these transcripts from BamSEC as transcribed by Thomson Reuters. The transcripts include a prepared remarks section, a question and answer section, and instructions from the call operator. For the purpose of my analysis, I analyzed only the prepared remarks from the company's representatives.

As part of pre-processing and data cleaning, I took the following steps:

- Replaced instances of "n't" with " not"
- Removed numbers (did not remove if part of hashtag or mention) and punctuation
- Removed capitalization
- Removed stop words, stop phrases ("good morning")
- Lemmatize words
- Removed pre-processed one-letter words
- Removed stop paragraphs such as introductory or disclosure statements which appear frequently

I vectorized the data with TfidfVectorizer()and ngram_range = (1,2).

Topic Modeling
For dimensionality reduction, I used Non-Negative Matrix Factorization (NMF) with number of components equal to 30.

Sentiment Analysis
I used the Loughran and McDonald Master Financial Dictionary to measure sentiment of the calls. Their dictionary is often cited as the de facto financial dictionary for NLP analysis.

I calculated the percentage of pre-processed words that were:

- Negative
- Positive
- Uncertain
- Litigious
- Constraining
- Interesting
- Modal

More information about the master dictionary can be found here.

Classification

Calculate Returns
Buy, sell, and hold signals were assigned in the following manner:

- Buy [1]: one-month return is at least 3%

- Sell or short-sell [-1]: one-month return is less than or equal to -3%
- Hold [0] otherwise

To calculate the return, I assumed the stock would be bought at close of business on the day of the earnings release and sold at close of business in 30 days.

Recommend Investment Decisions
Given topic and sentiment features, I trained and validated a random forest model with n_estimators = 400 to classify buy, sell, or hold. The training data set were transcripts of calls which occurred before those in the validation data set.

Results

Had I taken the recommendations from the random forest model on 55 transcripts starting from Q1 2016, I would have earned a 7.9% return, which is approximately 3x the return I would have earned had I simply bought on every earnings release day, 2.6%. The classification model provides value!

Final Comments

Future Enhancements
In the future, I'd like to try out more classification models and hyperparameters, analyze other industries, process the Q&A section of the call, and improve the train-test-split pipeline. Currently, I am training on the train data set and predicting on all of the validation/test data set. Instead, I could improve my results by training on all transcripts before the transcript I'm predicting.