Spark Near Real-Time Sentiment Analysis

*Mohan K. Patnam, CN Chen, Bruno Janota (CSCI E-63)*

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Motivation:** *This project studies whether social media feeds focused towards stock market and concerning companies, are a suitable data source for forecasting the direction and volatility in the underlying stock prices. In particular, we focus on StockTwits.com which is a popular social media platform where real investors, traders and general public alike express their opinion on companies and their performance in the form of tweets. We begin this project by providing a generic approach to download historical tweets for the chosen stocks over a chosen period. We then introduce a couple of models to calculate social sentiment scores using two variations of Bag-of-Words (dictionary) and a Naive-Bayes classifier. We apply basic text processing techniques using NLTK to preprocess tweets using the NLTK lemmatizer and stopwords. We further perform regression analysis to see if social sentiment can serve as a leading indicator to predict changes in underlying stock prices. We demonstrate an ability to sense the market pulse or sentiment using a near real-time stream of tweets. Our aim here is illustrate the use of modern machine-learning and real-time parallel processing technologies while answering a fundamental question of whether social sentiment matters for stock market performance.*

# Introduction

This project is divided into four phases. The first phase of the project focuses on data download and wrangling into a suitable format for further processing. We collected historical tweets for chosen stocks for a period of two months from [www.StockTwits.com](http://www.stocktwits.com) using their restful API. As per our research on Investopedia.com, there are few stocks that are very popular on stocktwits platform ([Most followed stocks in stocktwits.com](http://www.investopedia.com/articles/markets/021316/most-followed-stocks-stocktwits-nflx-fb.asp)). We chose three such stocks for our sentiment analysis in this project - AAPL, FB, and TSLA. We also collected historical stock prices (close price, traded volume) for the same stocks for the same period from Yahoo! Finance (<http://finance.yahoo.com>). The second phase of the project aims to build a couple of models to predict the social sentiment in the form of a weighted score. We use this score to categorize the social sentiment of a tweet into one of the three categories - Bullish (positive), Bearish (negative), and Neutral (none). The first two models use a deterministic bag-of-words approach applying different financial dictionaries to guide the lookup of keywords in the tweet and further derive the sentiment score. The second model is based on a probabilistic approach using a Naive-Bayes Classifier. We train the Naive-Bayes model using training-set data and use the test-set data to predict the score. We provide confusion matrices for both approaches to highlight the accuracy of the respective models. The third phase of the project aims to build a regression model by using an aggregate daily sentiment score as a predictor of stock close price. We use GradientDescentOptimizer to train the regression model first and then use the test-data to predict stock close prices. We document model results and accuracy in the form of Tensor board summaries (loss) and computing graph. The final phase simulates the StockTwits continuous data feed and provides a sentiment score in near real-time using Spark.

**Definitions**

1. **Tweet** - a short message posted by an user on stocktwits platform with a max of 140 characters. A tweet on StockTwit will always include a “cashtag” prefix denoting the ticker of the company relevant to the message. See example below.  
   <https://www.stocktwits.com/symbol/AAPL?q=AAPL>  
   
2. **Social Sentiment** **Category** - The tweet (a.k.a message) can be optionally tagged with a category (Bullish or Bearish) indicating the sentiment towards a particular stock. For example: the above tweet is tagged with a bullish sentiment. A bullish sentiment indicates user is optimistic about a particular stock’s performance in the near future. A bearish sentiment indicates user is pessimistic about stock’s performance in the near future.
3. **Social Sentiment Score** - Besides a category, one can assign a score for each tweet using several approaches. A score is a numeric value and we normalize it to be within [-1 to +1] for this project purpose. Values in the range [-0.5 to 0.5] indicates a neutral sentiment. A value of 0.5 and above indicates a positive sentiment and likewise, a value of -0.5 and below indicates negative sentiment.
4. **Daily Sentiment Score** - An aggregate sentiment score assigned to a stock on a daily basis by processing the sentiment score of individual tweets for the same day. This is the raw score.
5. **Weighted Daily Sentiment Score** - An aggregate sentiment score for a day calculated using raw sentiment score weighted by the ratio of tweet-volume for the day over the average-tweet-volume for the chosen time period (2 months in our case). A bullish or bearish overall sentiment on a given day should be weighted higher or lower depending on the number of tweets posted on that day in comparison to the rest of the days.  
     
   (n is the day index)  
   *TweetVolume(n) = # tweets for ‘n’the day.  
   AvgTweetVolume = Average # tweets per day considering two month period.*

***WeightedDailySentimentScore(n)*** *= (RawDailySentimentScore(n)) \* (TweetVolume(n) / AvgTweetVolume)*In our case, we collected historical tweets for a period of 2 months from March 1st, 2017 to April 30th, 2017. In that period, there were 42 trading days excluding holidays and weekends.

1. **Stock Close Price** - Stock price of a chosen company as indicated by the close price on a given trading day. All stock prices mentioned in this document refers to close price.
2. **LoughranMcDonald or Financial Dictionary** - This “dictionary” is a set of multiple lists of words, where each list pertains to a certain quality (e.g. positive, uncertain, strong, weak) specifically in the financial domain, as defined by Tim Loughran and Bill McDonald in their 2011 paper “When is a Liability not a Liability?” The two word lists we use are FinPos (financially positive sentiment) and FinNeg (financially negative sentiment), with 354 and 2,355 words respectively.
3. **Harvard IV-4 Dictionary** - This relates to a generalist “dictionary” which applies to non-financial contexts as well. This dictionary is the updated, expanded version of the popular Harvard IV-4 TagNeg dictionary. The two lists used are called Positiv (1,637 words) and Negativ (2,006 words).
4. << **definitions for naive-bayes model**>>

**Project Scope and Assumptions**

1. Based on our research, we chose three popular stocks on StockTwits platform - AAPL, FB, TSLA. Typically technology and social-media stocks tend to be much popular on social media platforms.
2. Historical tweets - Collected for three stocks (AAPL, FB, TSLA) for a time period of two months starting from March 1st to April 30th. This period had 42 trading days excluding holidays and weekends. In other words, we ignored tweets during holidays and weekends.
3. Each tweet collected from StockTwits.com can have an optional sentiment attached by user (either bearish or bullish). A majority of tweets are classified as none with the assumption that underlying tweets doesn't have any keywords that strongly indicate any sentiment. In other words, these tweets are classified under 'neutral' sentiment for the project demonstration purpose.

**Sources**

1. Investopedia - [Most followed stocks in stocktwits.com](http://www.investopedia.com/articles/markets/021316/most-followed-stocks-stocktwits-nflx-fb.asp)
2. StockTwits API - <https://stocktwits.com/developers/docs>
3. Yahoo! Finance for historical stock prices - [AAPL Historical Prices](https://finance.yahoo.com/quote/AAPL/history?period1=1488344400&period2=1493524800&interval=1d&filter=history&frequency=1d)
4. Bloomberg for Intraday prices - Bloomberg Terminal (subscription required)
5. LoughranMcdonald Financial Dictionary - <http://www3.nd.edu/~mcdonald/Word_Lists.html>
6. Harvard IV-4 Dictionary - <http://www.wjh.harvard.edu/~inquirer/spreadsheet_guide.htm>
7. Naive Bayes Classifier - ??

**API and Tools**

1. Python 3.5 on Windows-10 64-bit machine and Linux (Cloudera VM) platform
2. IDE - PyCharm and IPython Notebook
3. Packages - numpy, tensorflow, matplotlib, urllib, json, nltk, pandas, blpapi…
4. StockTwits RESTful API
5. Tensorboard API
6. Spark
7. Bloomberg API
8. ...

# Data Collection and Wrangling

Data Source for tweets. [www.StockTwits.com](http://www.stocktwits.com)

URLs to download and process tweets using RESTful API -

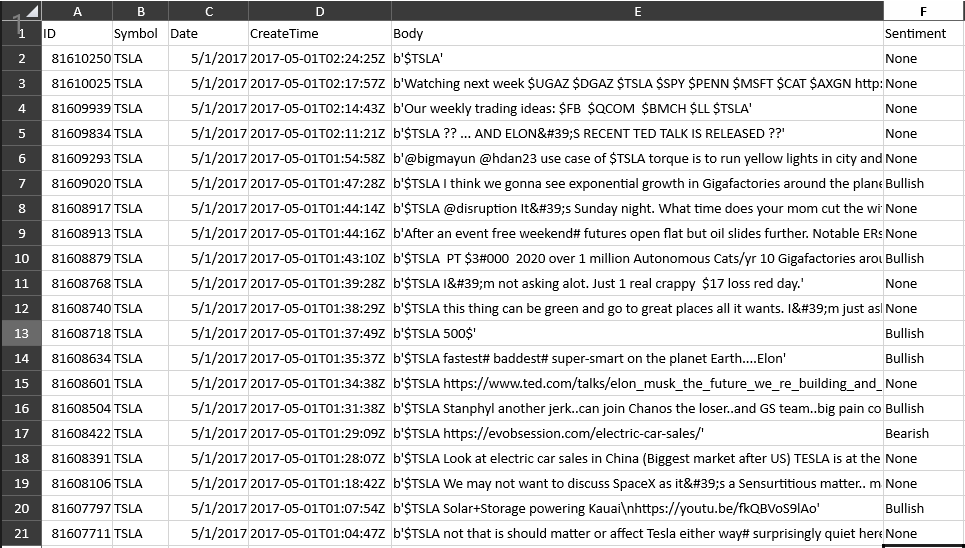
AAPL:<https://api.stocktwits.com/api/2/streams/symbol/AAPL.json>

TSLA:<https://api.stocktwits.com/api/2/streams/symbol/TSLA.json>  
FB: <https://api.stocktwits.com/api/2/streams/symbol/FB.json>

Sample JSON response for AAPL tweets:



Show processed csv file



**Tweets basic stats:**

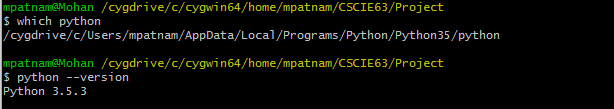
#days: 42

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Stock** | **Total Tweets** | **Total Bullish** | **Total Bearish** | **Average Tweets/Day** |
| AAPL | 24,550 | 7154 | 1988 | 584 |
| FB | 15,679 | 3883 | 1772 | 373 |
| TSLA | 34,009 | 9175 | 5537 | 809 |

Implementation details:

Development environment:

Python 3.5 on Windows-10 64-bit./n



Download application logs:

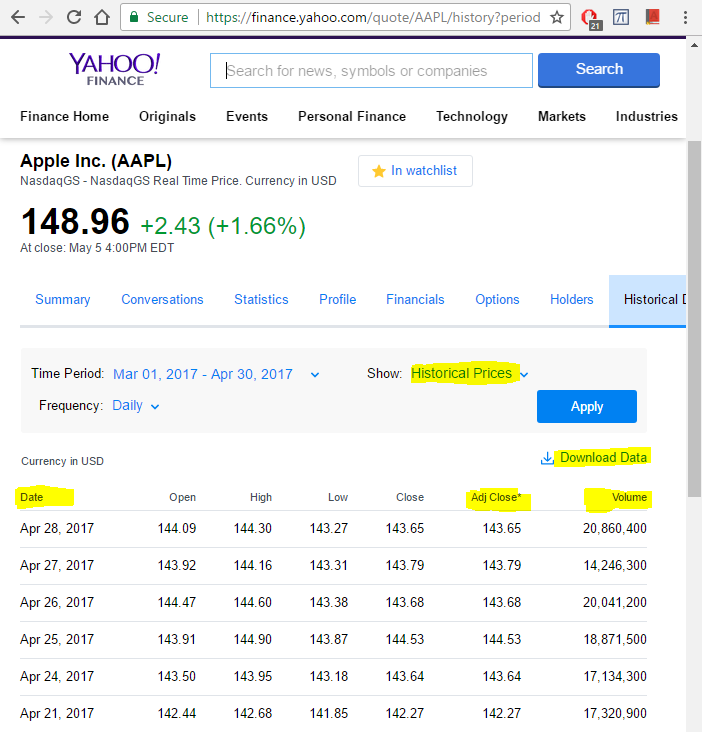
<https://github.com/mpatnam/CSCIE63-Project/blob/master/Logs/Data%20Collection/aapl.20170430.log>

Historical prices (source Yahoo Finance)

Show url for chosen stocks

Show sample table

AAPL:<https://finance.yahoo.com/quote/AAPL/history?period1=1488344400&period2=1493524800&interval=1d&filter=history&frequency=1d>  
TSLA:<https://finance.yahoo.com/quote/TSLA/history?period1=1488344400&period2=1493524800&interval=1d&filter=history&frequency=1d>  
FB: <https://finance.yahoo.com/quote/FB/history?period1=1488344400&period2=1493524800&interval=1d&filter=history&frequency=1d>



Intraday prices (source Bloomberg)

Show sample table

Tweets processing - json/restful api

Python code details

Results: table showing #tweets collected for 3 stocks.

# Sentiment Analysis

Basic analysis of tweets.

Describe input data format.

Basic stats for #tweets each day - SD/mean/median.

Show various plots.

#tweets vs close price,

#tweets vs traded volume

# Bag-Of-Words Model

Describe Approach

Describe input data format

Any assumptions?

Show code snippets and logs

Results: Confusion Matrix

Daily Sentiment Score vs Close Price changes

# Naive-Bayes Classifier

Describe approach

Describe input data format

Any assumptions?

Show code snippets and logs

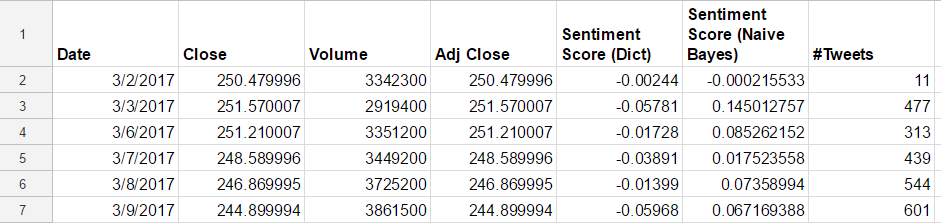
Results: Confusion Matrix

Daily Sentiment Score vs Close Price

# Regression Model

Describe approach

Describe input data format



Any assumptions?

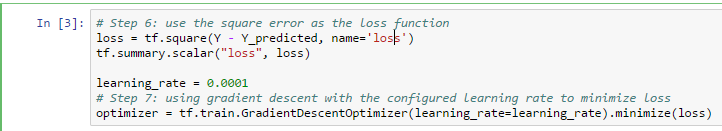
Show code snippets and logs

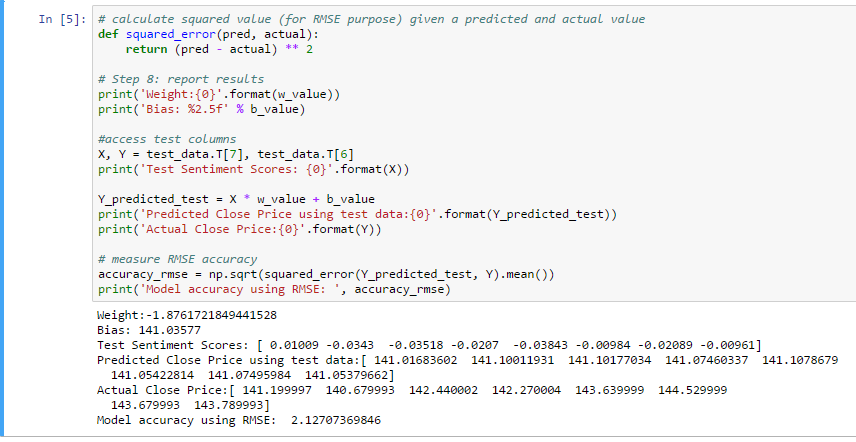
Results: Linear regression plot

Tensorboard summaries and graph

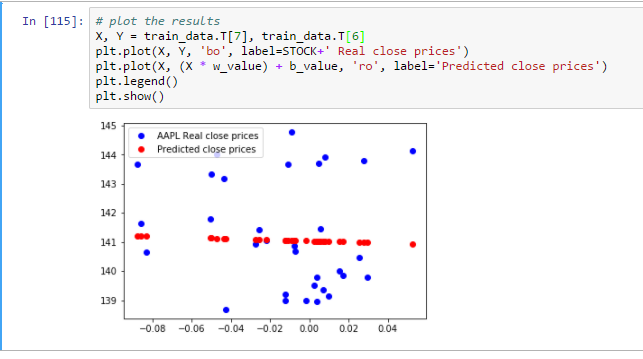
# Real time streaming

# Future work

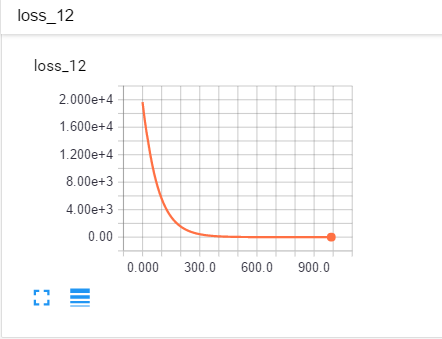




Regression plots for AAPL (Daily sentiment score vs close price)



Loss function



Tensorboard graph for GradientDescentOptimizer training.

TSLA:

