TWEET SENTIMENTS

YANKUN SONG, ISABELLA CHIAL, STEPHANIE PARROTTA, STEPHANIE GOETZ

Web Analytics (ISGB-7978-V02)

Our Team

1 **Yankun Song**
MS in Business Analytics

2 **Stephanie Goetz**
FT- MBA, concentration in Marketing and Sustainability

3 **Stephanie Parrotta**
FT- MBA, concentration in Marketing and Entrepreneurship

4 **Isabella Chial**
FT- MBA, concentration in Marketing and Information Systems

AGENDA

1. Business Goal Analysis
1. Dataset Description
1. System Design
1. System Implementation
1. Evaluation
1. Conclusion and Future Direction

ABOUT our project

We are stock analysts and our client is a stock trading company

1. Stock price movement
2. The company wants to know if people's sentiments on twitter have an impact on the stock price [monitoring tweets]

We will achieve this by:

1. Crawling and analyzing tweets using: Tweepy, and Vader
2. Create a prediction model: you can make a decision whether to buy or short a stock based on yesterday's sentiment

STOCKS

1. **MICROSOFT CORPORATION**
   Technology Company [MSFT]
2. **APPLE**
   Technology Company [AAPL]
3. **TESLA**
   Technology Company [TSLA]
4. **NIO**
   Technology Company [NIO]

TW EETS

Variable	Description
date	the time the tweet was published
followers	the number of followers of who published the tweet
tweet	the content of the tweet

Since we are analyzing the relationship between tweets sentiments and the movement of stock price, the raw data we obtained has two parts: tweets data and stock price data

To the right is example of the tweets we crawled. We will focus on columns A/B/D

	A	B	C	D	E
52	date	followers	screen_name	tweet	tweet_id
53	10/30/2020 18:05	60	tarangNooooooo	Shorted $AAPL	1.32E+18
54	10/30/2020 19:37	40	bobandjo1	You may want to look at	1.32E+18
55	10/30/2020 21:35	433	AAPL_moves	Apple Inc stock dropped	1.32E+18
56	10/30/2020 21:17	433	AAPL_moves	Apple Inc price at close,	1.32E+18
57	10/30/2020 22:20	27	BuffettPrime	$AAPL showed earnings	1.32E+18
58	10/31/2020 0:22	3355	elliottwaves	Group 1, 2 & 3 Daily	1.32E+18
59	10/30/2020 18:58	195	XbPirlo	Buen momento para cor	1.32E+18
60	10/30/2020 19:30	2730	it_tradingview	#AAPL - Apple, si scende	1.32E+18

STOCKS

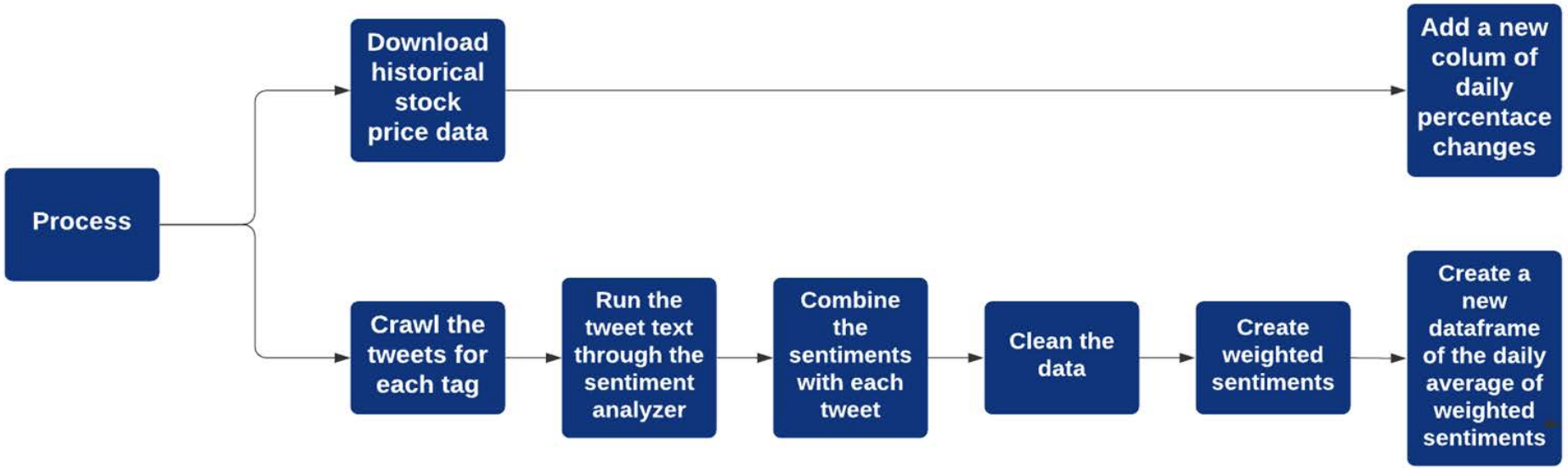
Variable	Description
Date	the date
Open	the opening price on that day
Adj Close	the closing price on that day

The variables we will be focusing on for stocks are: date, open, and adj close

To the right is an example of the stock price movement history we downloaded from Yahoo Finance. We will focus on columns A,B,F

	A	B	C	D	E	F	G
1	Date	Open	High	Low	Close	Adj Close	Volume
2	11/6/2019	64.1925	64.3725	63.8425	64.31	63.67819	75864400
3	11/7/2019	64.685	65.0875	64.5275	64.8575	64.41312	94940400
4	11/8/2019	64.6725	65.11	64.2125	65.035	64.58941	69986400
5	11/11/2019	64.575	65.6175	64.57	65.55	65.10088	81821200
6	11/12/2019	65.3875	65.6975	65.23	65.49	65.04128	87388800
7	11/13/2019	65.2825	66.195	65.2675	66.1175	65.66449	1.03E+08

PT 1DATA preparation



PT 1 1 CRAW L TW EETS

Crawl the tweets for each tag, Time period 2019.11-2020.11 [Snscrape, tweepy]

Company	Sector	Stock Ticker (Tag in Tweets)	Number of Tweets	Avg tweets per day
Microsoft Corporation	Technology company	MSFT	22912	63
Apple	Technology company	AAPL	22909	63
Tesla, Inc.	Electric car company	TSLA	41265	113
NIO	Automobile company	NIO	12995	36

Figure 1 Stock Tweet Numbers from November 2019-November 2020

PT 12 VADER SENTIMENT ANALYSIS

- VADER is a lexicon and rule-based sentiment analysis tool that is specifically attuned to sentiments expressed in social media.
- The VADER sentiment lexicon is sensitive both the polarity and the intensity of sentiments expressed in social media contexts, and is also generally applicable to sentiment analysis in other domains.

Text	Compound	Positive	Neutral	Negative
VADER is smart, handsome and funny	0.8316	0.746	0.254	0.0
VADER is VERY SMART, handsome, and FUNNY!	0.9342	0.767	0.233	0.0
VADER is not smart, handsome, nor funny	-0.7424	0.0	0.354	0.646
Today SUX!	-0.5461	0.0	0.221	0.779
Today only kinda sux! But i'll get by, lol	0.5249	0.317	0.556	0.127

Figure 2 Example of VADER Rating

Examples of crawled tweets and the sentiments analysis

Date	Tweet content	Followers	Compound	Neg	Neu	Pos
2020-06-15	YouTube is built on the back of stolen content: Trent Reznor #GOOGL	5172	-0.4939	0.176	0.824	0.0
2020-06-15	These 3 stocks Will Surprise Investors This Earning Season #AAPL #AMZN #GOOGL	377	0.2732	0.0	0.884	0.116

Figure 3 Crawled Tweets and Sentiment Analysis

PT 13 STOCK PRICE

2019/11/01- 2020/11/01
253 days

	A	B	C	D	E	F	G
1	Date	Open	High	Low	Close	Adj Close	Volume
2	11/6/2019	2.46	2.46	1.96	2.03	2.03	1.14E+08
3	11/7/2019	2.11	2.2	2.05	2.07	2.07	39411800
4	11/8/2019	2.13	2.13	1.94	1.98	1.98	38341600
5	11/11/2019	1.9	1.96	1.78	1.86	1.86	34228400
6	11/12/2019	1.9	2.05	1.83	1.94	1.94	27818000
7	11/13/2019	1.9	1.99	1.85	1.9	1.9	24969400
8	11/14/2019	1.86	1.89	1.66	1.75	1.75	46683200
9	11/15/2019	1.75	1.84	1.66	1.8	1.8	32003800
10	11/18/2019	1.88	1.9	1.73	1.8	1.8	25996000
11	11/19/2019	1.78	1.89	1.78	1.83	1.83	26780800
12	11/20/2019	1.85	1.87	1.77	1.84	1.84	20728500

Figure 4 Screenshot of Data Downloaded

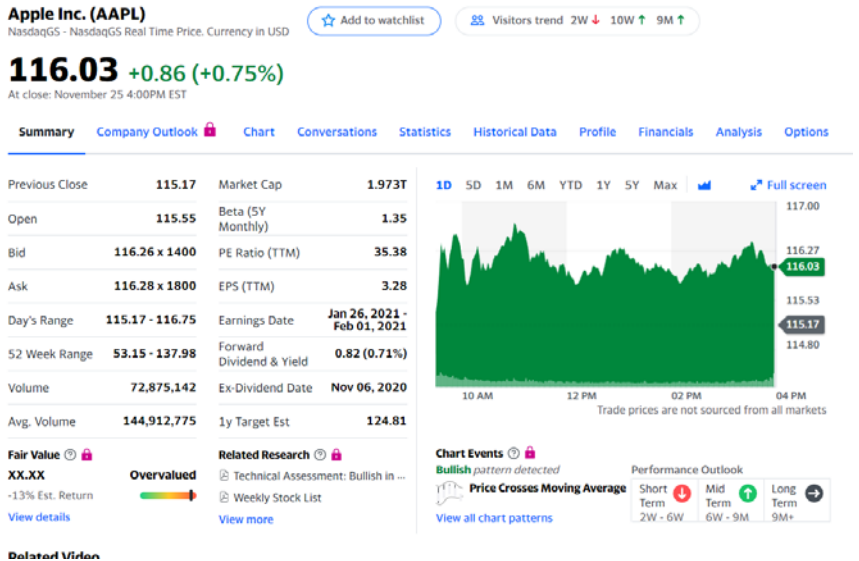


Figure 5 Yahoo Finance

PT 14 JOIN SENTIMENT WITH STOCK PRICE

	A	B	C	D
1		score	day	Movement
2	11/5/2019	-0.2491	1	-0.80119
3	11/6/2019	-0.14155	2	-0.42032
4	11/7/2019	0.222457	3	-0.12848
5	11/10/2019	0.14969	6	0.814369
6	11/11/2019	-0.37167	0	-0.52948
7	11/12/2019	-0.21566	1	0.585132
8	11/13/2019	-0.17791	2	-1.10313
9	11/14/2019	0.016056	3	0.098272
10	11/17/2019	0.023673	6	-0.19941
11	11/18/2019	0.083223	0	-1.28201
12	11/19/2019	0.567047	1	-1.56408
13	11/20/2019	0.048735	2	-1.31791
14	11/21/2019	0.063684	3	-0.99151

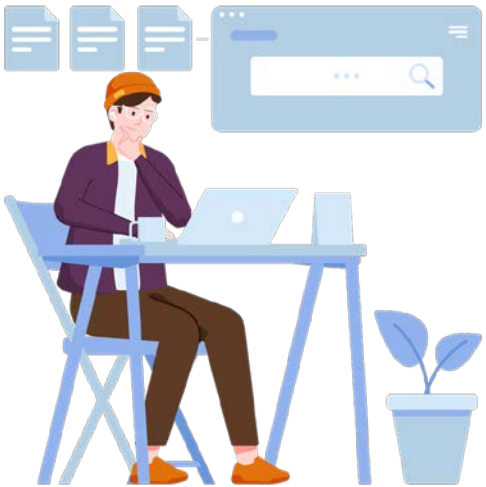
Figure 6 Example of the Data Frame for Future Analysis

Score:
the weighted avg sentiment score on that day

Day: Day of week
Monday 0, Tuesday 1, Wednesday 2, Thursday 3, Friday 4, Saturday 5, Sunday 6

Movement:
Next day's stock price movement(%)

PART 2 DATA ANALYSIS



1

Make a Train/ Test split (80-20) of the time series data

Covering 249 days
Data from 200 days (80%) will be used for **training** and 49 days (20%) will be used for **validation**

2

Train two algorithms with the data for each company

Linear Regression AND Logistic Regression
Check the accuracy of each algorithm for the data with 5-fold cross validation

3

Testing

Trade under the guidance of the prediction model from 2020.10.01 to 2020.10.31, 22 days

PART 2.1
DATA SPLIT

1. For data analysis, **we used a sample of 200 days** from the stock data to train our prediction model to predict the price movements.
    a. 49 days were used for validation of the model for an unbiased evaluation

2. The test dataset provides an unbiased evaluation of our final model based on the training dataset

3. We used **22 days in October 2020** to develop our testing model.

PART 2.2

CROSS VALIDATION

Due to the limited amount of data, using only 20% of the data for validation and 80% of the data for training might not be enough.

To avoid the possibility that the training/test split is not completely random, cross-validation is performed on the data, so that a more representative result of the accuracy of each algorithm is obtained.

The training data is further divided into 5 subsets, and each subset is tested against the other 4 subsets.

PART 2.3 SYSTEM IMPLEMENTATION

```
def sentiment_score(ticker, weighted):
    """calculate each day's weighted sentiments, and save the results into a csv file
    """
    df = pd.read_csv(ticker+"_tweets.csv")
    df = df.dropna()    #remove the rows with na values
    df = df.reset_index()   # reset the index
    df = df.loc[:, ['date', 'followers', 'tweet']]    # only take the key info we'll use
    df['date']=pd.to_datetime(df['date'], format='%Y-%m-%d %H:%M:%S')    # transfer the type of date
    df['followers'] = df['followers'].astype(float)
    df['date'] = df['date'].dt.normalize()    # we only need the date

    from vaderSentiment.vaderSentiment import SentimentIntensityAnalyzer

    def sentiment_scores(sentence):
        """return a sentiment score of the input sentence
        """
        sid_obj = SentimentIntensityAnalyzer()
        sentiment_dict = sid_obj.polarity_scores(sentence)
        return sentiment_dict['compound']

    df['senti_score'] = 0

    for i in range(0, len(df)):
        df.loc[i, 'senti_score'] = sentiment_scores(df.loc[i, 'tweet'])

    def weighted_avg(group, avg_name, weight_name):
        """ calculate weighted avg
        """
        d = group[avg_name]
        w = group[weight_name]
        try:
            return (d * w).sum() / w.sum()
        except ZeroDivisionError:
            return d.mean()

    if weighted == True:
        senti_series = df.groupby("date").apply(weighted_avg, "senti_score", "followers")
        filename = ticker+"_weighted_sentiment.csv"

    else:
        senti_series = df.groupby("date").mean().senti_score
        filename = ticker+"_avg_sentiment.csv"

    senti = pd.DataFrame({'tweets_date':senti_series.index, 'score':senti_series.values})
    senti['day'] = senti['tweets_date'].dt.dayofweek
    senti = senti.set_index("tweets_date")

    senti.to_csv(filename)
```

The python code programmed to analyze daily twitter sentiments. We imported the data into python and used the sentiment intensity analyzer to get scores using a weighted average.

To build our prediction models, we used the *scikit-learn* package in Python. We used **LassoCV** function for the linear regression model, and **LogisticRegressionCV** function for the logistic regression model

PART 2.4
TESTING

We used linear regression and logistic regression to train the prediction model, and compared these two models to see which has a better performance.

THREE STRATEGIES

ALL BUY

1

Long a stock everyday

LINEAR REGRESSION MODEL

2

Long or short a stock based on the result given by model. Predicted to go up, then long; Predicted to go down, then short

LOGISTIC REGRESSION MODEL

3

Long or short a stock based on the result given by model. Predicted to go up, then long; Predicted to go down, then short

PT 2.4 EXAMPLE

Take the first day [10-01-2020] as an example:

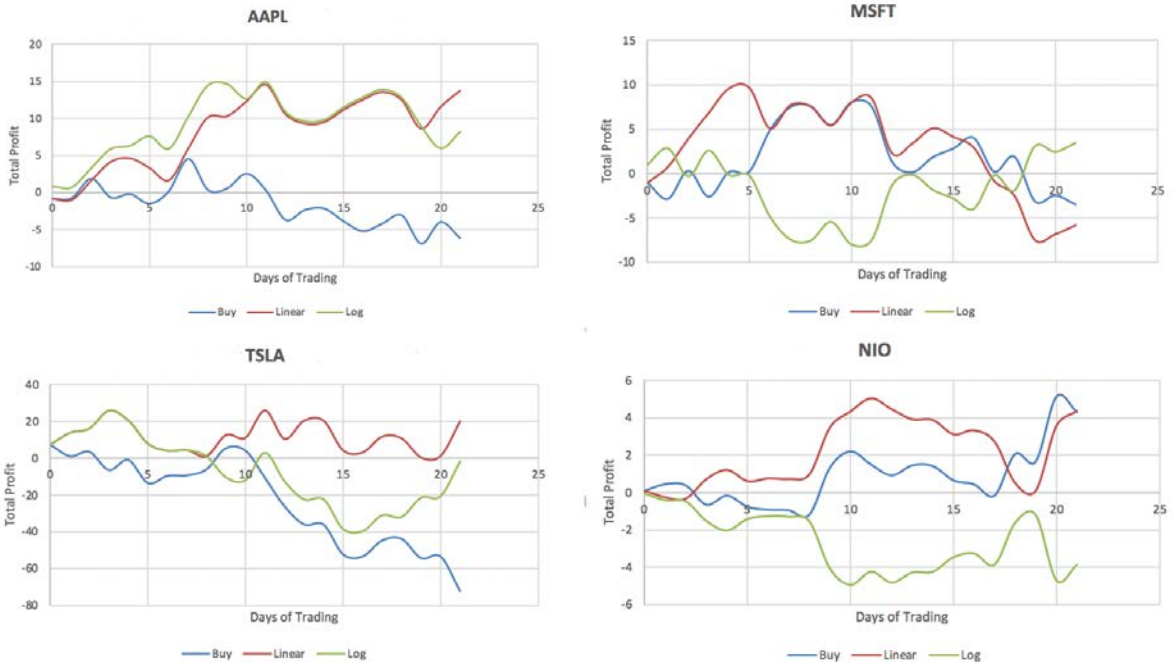
	A	B	C	D	E	F	G	H	I
1	Date	Open	High	Low	Close	Adj Close	Volume	LogReg	LinearReg
2	10/1/2020	117.64	117.72	115.83	116.79	116.79	1.16E+08	0	1
3	10/2/2020	112.89	115.37	112.22	113.02	113.02	1.45E+08	0	0

Based on **Logistic Regression** model:
Price will go down today, so we will short it. Then our cost in the evening will be the Adj Close(116.79), and our gain in the morning will be the Open(117.64). The net profit today is 117.64-116.79=$0.85.
We earned money today, because the model predicted right

Based on the **Linear Regression** model:
Price will go up today, so we will long it. Then our cost in the morning will be the Open(117.64), and our gain in the afternoon will be the Adj Close(116.79). The net profit today is 116.79-117.64= $-0.85.
We lost money today, because the model predicted wrong

PART 2.5 PROFIT CHANGE

The figure below illustrates the change of total profit under different strategies for each stock.
We can see in most cases, the linear regression model outperforms the others



PART 2.6 RETURN ON INVESTMENT

	AAPL	MSFT	TSLA	NIO	Avg
Buy	-5.29%	-1.63%	-16.77%	17.06%	-1.66%
Linear Regression	11.85%	-2.71%	4.67%	17.30%	7.78%
Logistic Regression	7.01%	1.63%	-0.41%	-15.28%	-1.76%

The return on investment (ROI) is also calculated for the three strategies. It also shows that the linear regression model helps us get a considerable amount of returns

PART 3: CONCLUSION

1. By comparing the ROI under each prediction model, **we found the model built under linear regression is the best among the three, generating an average return of 7.78% in a month.**

1. We can also compare this ROI with the SPDR S&P 500 (SPY) during this period. From [10-01-2020] TO [10-31-2020], **the SPY changed from 337.04 to 326.54, dropping 3.12%. It is safe to say that our linear regression model beat the market.**

1. In the future, we would explore the content of tweets in more depth by taking more hashtags into consideration. With this, we would be able to have an even better understanding of how sentiments correlate to stock prices.