Bryan Melvison

Sentiment Analysis on Stock Market Prediction

$NFLX Edition

PROFITS

Why is it Relevant?

- Provides insights into public perception of a company, which affects investor behavior.

- Identify the potential patterns that can affect stock prices (positively or negatively)

PROFITS

What exactly are we hoping to achieve

- Finding out whether the sentiments revolving around Netflix affects the profitability of an investor’s entry at the opening price and exit at the closing price.
- Exploring the relation between the sentiment of the company throughout the day and its impact on the stock price

LETS GET STARTED

Methodology

Data Preprocessing

Twitter Data Scraping
Natural Language Processing
Sentiment Analysis
Stock Data Scraping

Modelling

Data Partitioning
Defining Model:
KNN
Random Forest

Evaluation & Conclusion

Test on Unseen Data
Compare Accuracy Results
Conclusion

Twitter Data Scraping

- Due to time constraints, we’re gonna walk through an already cleaned dataset from the github repository and add additional changes necessary for our exploration.

- Scraped twitter data from 1 January 2018 - 11 July 2022

	date	renderedContent
0	2019-12-31 23:54:41+00:00	Original content rather than licensed titles doninating $NFLX's 2020 viewership (Netflix data taken with a pinch of salt as ever).
1	2019-12-31 23:43:26+00:00	FBAMZN $GOOGL$NFLX #FANG 2019 https://t.co/RdVlnrPiR3
2	2019-12-31 23:13:37+00:00	i think many retail folks early next week will look back on this end of year tape paint in markets today as a missed opportunity to sell the longs they were waiting to sell due to 2021 tax deferrance. \n\nWe'll see.\n\n$SPX$SPY $TSLA$NFLX $AAPL$AMD $NVDA$VIX VXXVXXB QQQIBB
3	2019-12-31 23:01:00+00:00	#Netflix continues to grow its global subscriber base but it's about to face some stiff competition from #Apple and #Disney. #stocks $NFLX$DIS $AAPL https://t.co/sIJu4fzAl0
4	2019-12-31 22:56:49+00:00	If only... $NFLXandchill?No,$NFLX & $DPZ. Both started the decade at 7.93 and 8.53 respectively. nearly 4000% gain.

Text Cleaning

	date	renderedContent
0	2019-12-31 23:54:41+00:00	Original content rather than licensed titles doninating $NFLX's 2020 viewership (Netflix data taken with a pinch of salt as ever).
1	2019-12-31 23:43:26+00:00	FBAMZN $GOOGL$NFLX #FANG 2019https://t.co/RdVlnrPiR3
2	2019-12-31 23:13:37+00:00	i think many retail folks early next week will look back on this end of year tape paint in markets today as a missed opportunity to sell the longs they were waiting to sell due to 2021 tax deferrance. \n\nWe'll see.\n\n$SPX$SPY $TSLA$NFLX $AAPL$AMD $NVDA$VIX VXXVXXB QQQIBB
3	2019-12-31 23:01:00+00:00	#Netflix continues to grow its global subscriber base but it's about to face some stiff competition from #Apple and #Disney. #stocks $NFLX$DIS $AAPL https://t.co/sIJu4fzAl0
4	2019-12-31 22:56:49+00:00	If only... $NFLXandchill?No,$NFLX & $DPZ. Both started the decade at 7.93 and 8.53 respectively. nearly 4000% gain.

What to do...?

EMOJIS: 😀😍🥰😘

Text Cleaning

- Uncapitalize the text
```
tweet <- tolower(tweet)
```
- Removing hashtag and cashtag symbol
```
tweet <- gsub("[$#]", "", tweet)
```
- Translate emoji into their description
```
library(emojifont)
tweet <- emoji_replace(tweet)
```
- Remove links from text
```
tweet <- gsub("https://t.co/[A-Za-z\\d]+", " ", tweet)
```
- Remove numerical values
```
tweet <- gsub("\\d+", " ", tweet)
```
- Tokenize the tweets using unnest_tokens
- Drop the stop_words, whilst also preserving all the negative auxiliary verbs (shouldn't, couldn't, aren't, etc)

```
> stop_words
# A tibble: 1,149 × 2
   word        lexicon
   <chr>       <chr>
 1 a           SMART
 2 a's         SMART
 3 able        SMART
 4 about       SMART
 5 above       SMART
 6 according   SMART
 7 accordingly SMART
```

Result

	date	renderedContent	cleaned
0	2019-12-31 23:54:41+00:00	Original content rather than licensed titles doninating $NFLX's 2020 viewership (Netflix data taken with a pinch of salt as ever).	original content rather license title doninating nflx's viewership netflix data take pinch salt ever
1	2019-12-31 23:43:26+00:00	FBAMZN $GOOGL$NFLX #FANG 2019 https://t.co/RdVlnrPiR3	amzn googl nflx fang
2	2019-12-31 23:13:37+00:00	i think many retail folks early next week will look back on this end of year tape paint in markets today as a missed opportunity to sell the longs they were waiting to sell due to 2021 tax deferrance. \n\nWe'll see.\n\n$SPX$SPY $TSLA$NFLX $AAPL$AMD $NVDA$VIX VXXVXXB QQQIBB	think many retail folk early next week look back end year tape paint market today miss opportunity sell longs wait sell due tax deferrance we'll see spx spy tsla nflx aapl amd nvda vix vxx vxxb qqq ibb
3	2019-12-31 23:01:00+00:00	#Netflix continues to grow its global subscriber base but it's about to face some stiff competition from #Apple and #Disney. #stocks $NFLX$DIS $AAPL https://t.co/sIJu4fzAl0	netflix continue grow global subscriber base face stiff competition apple disney stock nflx dis aapl
4	2019-12-31 22:56:49+00:00	If only... $NFLXandchill?No,$NFLX & $DPZ. Both started the decade at 7.93 and 8.53 respectively. nearly 4000% gain.	nflx chill nflx dpz start decade respectively nearly gain

Sentiment Analysis

- Twitter-Roberta-Base-Sentiment
  - Trained model on 198 million tweets to get polarity of tweet (0 = neg, 1 = neu, 2 = pos)
    - Then converted to 3 values (normalize): 1, 0, -1 denoting positive, neutral, negative respectively

cleaned	label	Polarity
original content rather license title doninating nflx's viewership netflix data take pinch salt ever	Negative	-1
amzn googl nflx fang	Neutral	0
think many retail folk early next week look back end year tape paint market today miss opportunity sell longs wait sell due tax deferrance we'll see spx spy tsla nflx aapl amd nvda vix vxx vxxb qqq ibb	Neutral	0
netflix continue grow global subscriber base face stiff competition apple disney stock nflx dis aapl	Neutral	0
nflx chill nflx dpz start decade respectively nearly gain	Neutral	0

date	P_mean	P_sum	twt_count
2018-01-01	0.007519	1	133
2018-01-02	0.020833	10	480
2018-01-03	0.071217	24	337
2018-01-04	-0.018519	-4	216
2018-01-05	-0.019737	-6	304

Processed Data

Stock Data Scraping

- Data accessed from **yahoo! finance**

Date	Open	High	Low	Close*	Adj Close**	Volume
Jul 10, 2018	417.24	419.44	413.08	415.63	415.63	9,382,900
Jul 09, 2018	415.95	419.12	411.10	418.97	418.97	11,127,500
Jul 06, 2018	397.45	408.65	395.52	408.25	408.25	8,629,600
Jul 05, 2018	393.80	399.24	390.86	398.39	398.39	8,448,900
Jul 03, 2018	399.49	399.98	389.50	390.52	390.52	5,280,300
Jul 02, 2018	385.45	398.38	380.00	398.18	398.18	8,142,500
Jun 29, 2018	399.19	401.33	390.55	391.43	391.43	9,252,500
Jun 28, 2018	395.00	396.90	387.10	395.42	395.42	12,219,900
Jun 27, 2018	407.56	411.59	390.00	390.39	390.39	16,541,400
Jun 26, 2018	393.28	404.78	389.05	399.39	399.39	15,191,200
Jun 25, 2018	404.69	405.99	378.75	384.48	384.48	22,490,900
Jun 22, 2018	419.98	420.50	409.65	411.09	411.09	10,428,600

Stock Data Scraping

- Merge the 2 tables by the date:

	date	Open	High	Low	Close	Adj Close	Volume	P_mean	P_sum	twt_count
0	2018-01-02	196.100006	201.649994	195.419998	201.070007	201.070007	10966900	0.020833	10	480
1	2018-01-03	202.050003	206.210007	201.500000	205.050003	205.050003	8591400	0.071217	24	337
2	2018-01-04	206.199997	207.050003	204.000000	205.630005	205.630005	6029600	-0.018519	-4	216
3	2018-01-05	207.250000	210.020004	205.589996	209.990005	209.990005	7033200	-0.019737	-6	304
4	2018-01-08	210.020004	212.500000	208.440002	212.050003	212.050003	5580200	-0.007663	-2	261
...	...	...	...	...	...	...	...	...	...	...
1132	2022-07-01	176.490005	180.100006	174.270004	179.949997	179.949997	5194700	-0.062315	-21	337
1133	2022-07-05	176.279999	185.919998	172.679993	185.880005	185.880005	7334300	-0.058824	-25	425
1134	2022-07-06	185.199997	186.220001	180.820007	184.059998	184.059998	5753400	-0.014870	-8	538
1135	2022-07-07	184.270004	190.210007	183.500000	189.270004	189.270004	6334500	-0.055427	-24	433
1136	2022-07-08	186.020004	189.910004	182.750000	186.979996	186.979996	5831300	-0.043011	-12	279

Stock Data Scraping

- Add 2 more columns, Finalize the data for the model training:

	date	Open	High	Low	Close	Adj.Close	Volume	P_mean	P_sum	twt_count	net_increase	res
1	2018-01-02	196.10	201.65	195.42	201.07	201.07	10966900	0.020833333	10	480	4.970001	P
2	2018-01-03	202.05	206.21	201.50	205.05	205.05	8591400	0.071216617	24	337	3.000000	P
3	2018-01-04	206.20	207.05	204.00	205.63	205.63	6029600	-0.018518519	-4	216	-0.569992	N
4	2018-01-05	207.25	210.02	205.59	209.99	209.99	7033200	-0.019736842	-6	304	2.740005	P
5	2018-01-08	210.02	212.50	208.44	212.05	212.05	5580200	-0.007662835	-2	261	2.029999	P
6	2018-01-09	212.11	212.98	208.59	209.31	209.31	6125900	-0.025210084	-6	238	-2.800003	N

Methodology

Data Preprocessing

Twitter Data Scraping
Natural Language Processing
Sentiment Analysis
Stock Data Scraping

Modelling

Data Partitioning
Defining Model:
KNN
Random Forest

Evaluation & Conclusion

Test on Unseen Data
Compare Accuracy Results
Conclusion

Data Partitioning

- Partition our data into training set, and validation set
  - 80% of our samples are for training
  - 20% of our samples are for testing

```
#Partition the data into testing and training set
set.seed(42)
tweet$res <- factor(tweet$res, levels = c("N", "P"))
tweet
training <- tweet |> select(P_mean, res)
train_index <- createDataPartition(training$res, p = 0.8, list = FALSE)
train_set <- training[train_index, ]
test_set <- training[-train_index, ]
```

Classification Models Used:

Random Forest

A form of decision tree where multiple decision trees are trained on different subsets of the data and perform majority voting to form a final prediction.

K Nearest Neighbor

Look for the K closest training examples,then using the neighbor to predict the class of the new observation.

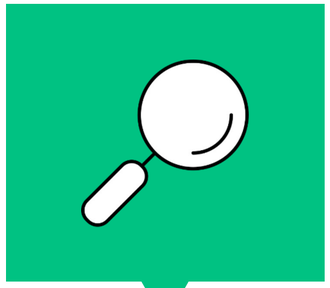
Model Training

```
#Train Model
train_rf <- train(res ~ . , method = "rf", data = train_set)
train_knn <- train(res ~ . , method = "knn", data = train_set)
```

Methodology

Data Preprocessing

Twitter Data Scraping
Natural Language Processing
Sentiment Analysis
Stock Data Scraping



Modelling

Data Partitioning
Defining Model:
KNN
Random Forest

Evaluation & Conclusion

Test on Unseen Data
Compare Accuracy Results
Conclusion

Test on Unseen Data

- We use accuracy as our key indicator of our model's performance:

```
y_hat_rf <- predict(train_rf, test_set, type = "raw")
confusionMatrix(y_hat_rf, test_set$res)$overall[["Accuracy"]]
[1] 0.5558824
```

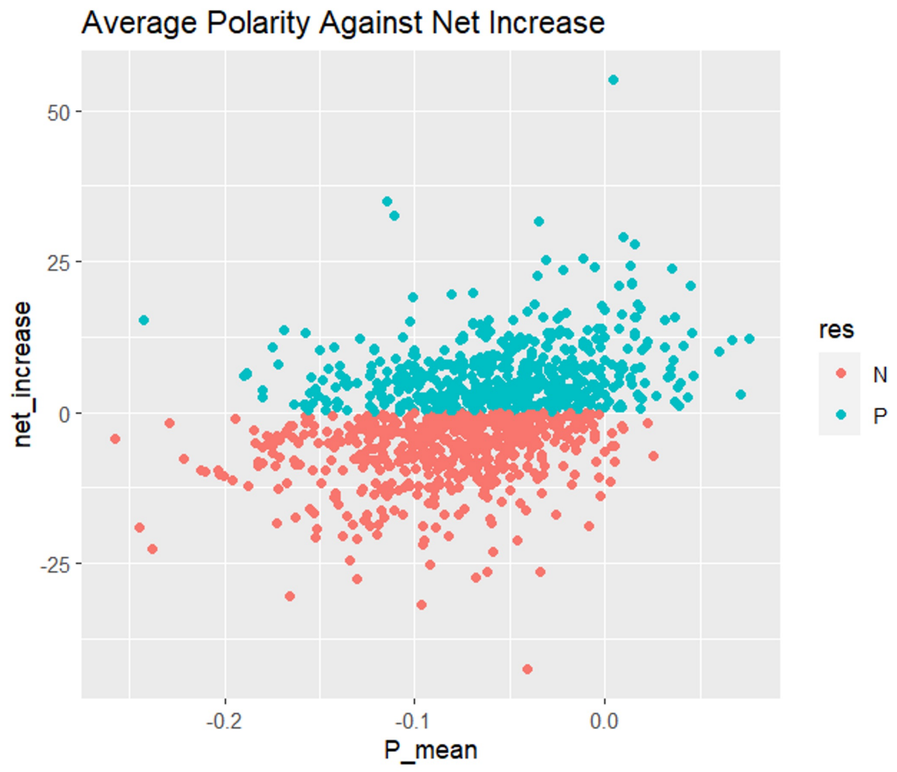
```
y_hat_knn <- predict(train_knn, test_set, type = "raw")
confusionMatrix(y_hat_knn, test_set$res)$overall[["Accuracy"]]
[1] 0.5941176
```

Although KNN model performs better by 4%, it is not significant enough as the accuracy of the 2 models were still really low.

Observations

- The model is trained with 1 feature, P_mean, against the result, potentially due to the lack of relationship between the 2?
- Not complex model, potentially underfitting

Graph out Relationship




```
Call:
lm(formula = net_increase ~ P_mean, data = tweet)

Residuals:
    Min      1Q  Median      3Q     Max 
-44.241  -4.462  -0.177   4.247  50.365 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept)   4.3712     0.4077   10.72   <2e-16 ***
P_mean       63.3777     4.7224   13.42   <2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 7.981 on 1135 degrees of freedom
Multiple R-squared:  0.137,	Adjusted R-squared:  0.1362 
F-statistic: 180.1 on 1 and 1135 DF,  p-value: < 2.2e-16
```

Very small R-squared value

- Weakly Correlated

Conclusion

- Based on our observation, it hasn’t really shown/ proven to be directly correlated, however it could also be due to our own limitations as mentioned previously. However, I believe there’s a lot of room for improvements and a lot more factors to take into consideration

- **Possible Improvements / Extensions:**
  - Do a Time Series Linear Regression Model as it helps with predicting future values
  - Add more features(through feature engineering) to make it more complex, and reduce underfitting (high bias)

References

https://github.com/KerolosAtef/Stock-market-prediction-using-sentiment-analysis-of-twitter