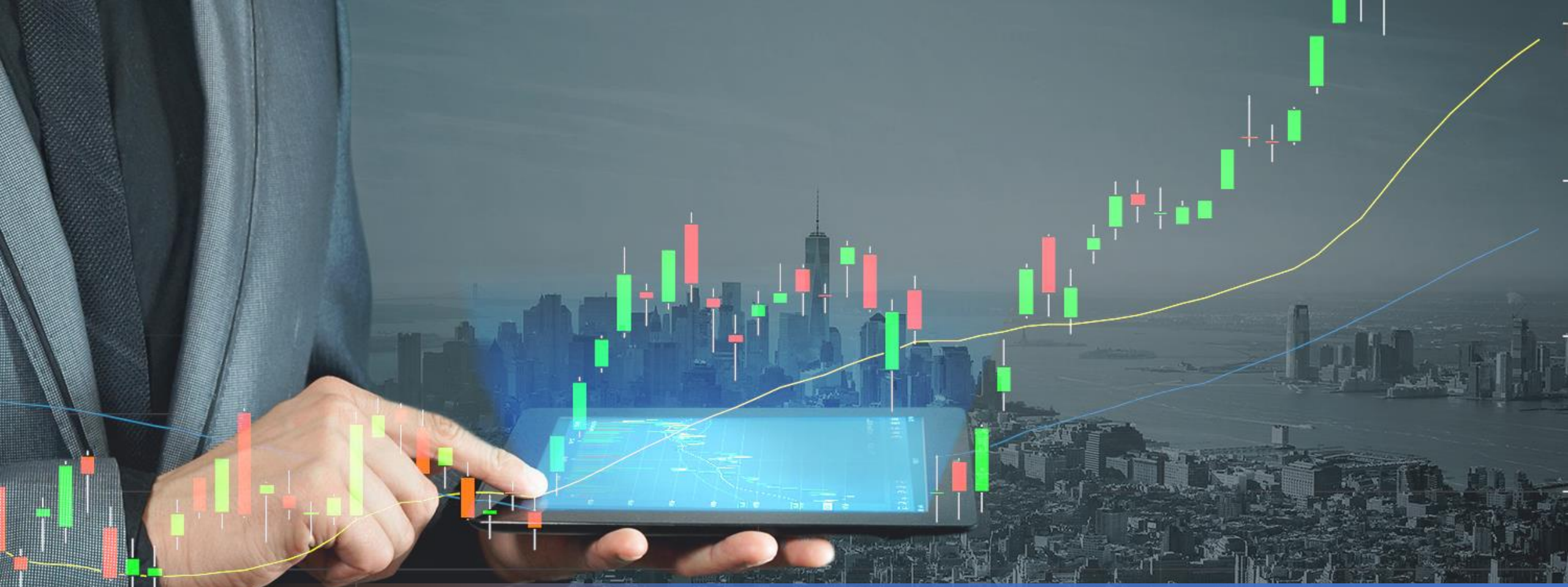


Stock Trend Prediction with News Data using Deep Learning

Li Kam Po & Lau Tsz Yui
LYU2004

Motivation

Buy today Sell tomorrow (BSTS) trading

- Buy stock and sell it within several days
- Profit from frequent transactions

Advantage: Easier to manage risk

Disadvantage: High transaction cost

- Stock patterns have high accuracy in predicting stock price
- Research on the relation of Twitter and the stock market

Project Context & Objectives

Simplify the situation

1. Focus on one large cap stock, Apple Inc (AAPL)
2. Focus on double bottom pattern
3. Not considering the transaction cost

Objectives

- Train a machine learning model to identify the pattern
- Study social media sentiment values and stock price

Development platform

- Python - Jupyter notebook
- Tensorflow, Vader, Tkinter

Overview

Recap of the last term

Quick recap of last term work

Pattern recognition

Explain how we get our dataset and experiment result

Social Media Analysis

Discuss the relationship of social media and stock

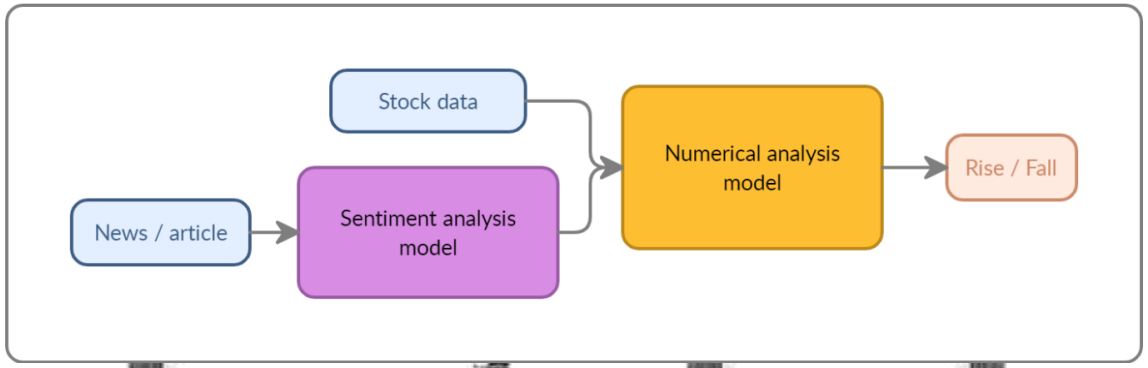
Tool demonstration

A tool that integrates what we do in this project

Q&A

Recap of the Last Term

- Stock prediction using stock data and news data
- Applied different models to conduct experiments
  - Numerical analysis: LSTM, GRU, KNN, Prophet
  - Sentiment analysis: Vader, Bert
- Merged the two components, input the resent stock data and news to predict whether AAPL will rise or fall

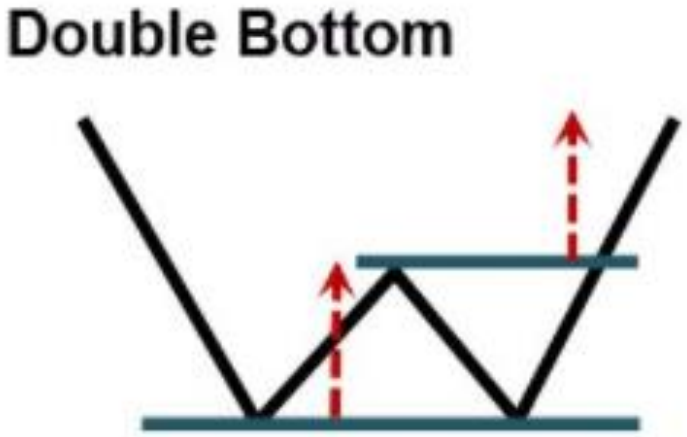


Pattern recognition

Pattern Recognition

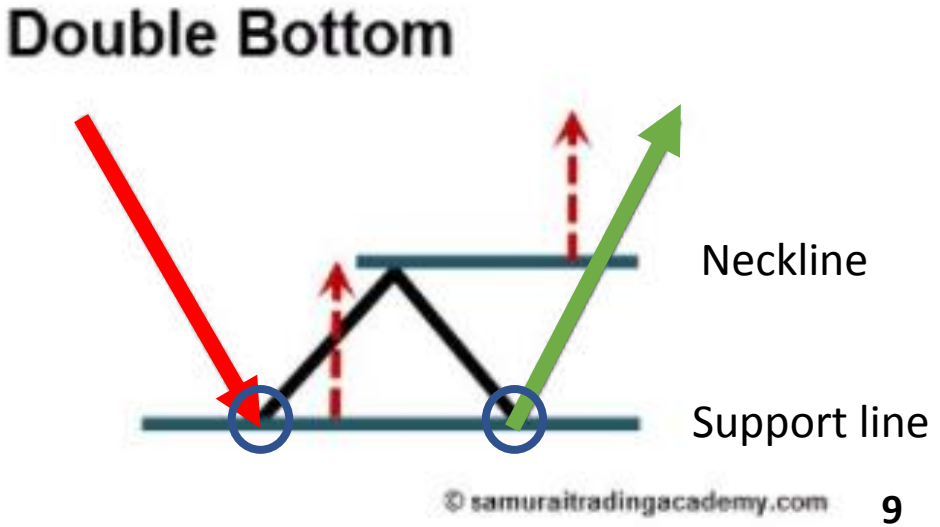
- Focus on double bottom pattern
- Stock patterns show future trend
  - Double bottom pattern: 78.55%

- Hard-coded (rules-based) Detector
  - No available dataset on the internet
  - From components of S&P 500
  - Building a dataset for machine learning



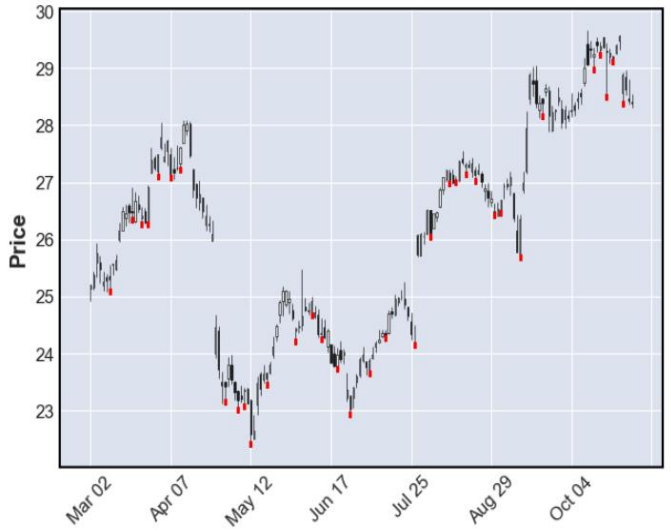
Double Bottom

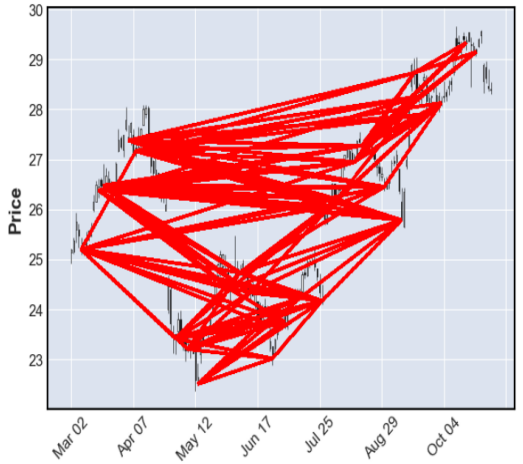
- First drop (red solid arrow)
  - 10-20%
- Two bottoms (blue circles)
  - The difference is 3-4%
- Last rise (green arrow)
  - Twice of the distance between the support and the neckline



Hard-coded Detector

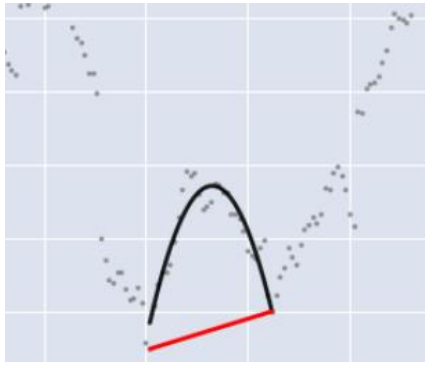
- Step 1: Locate the local minima
- Step 2: Find the support lines
  - Draw a line if the difference of two local minima is 3-4%

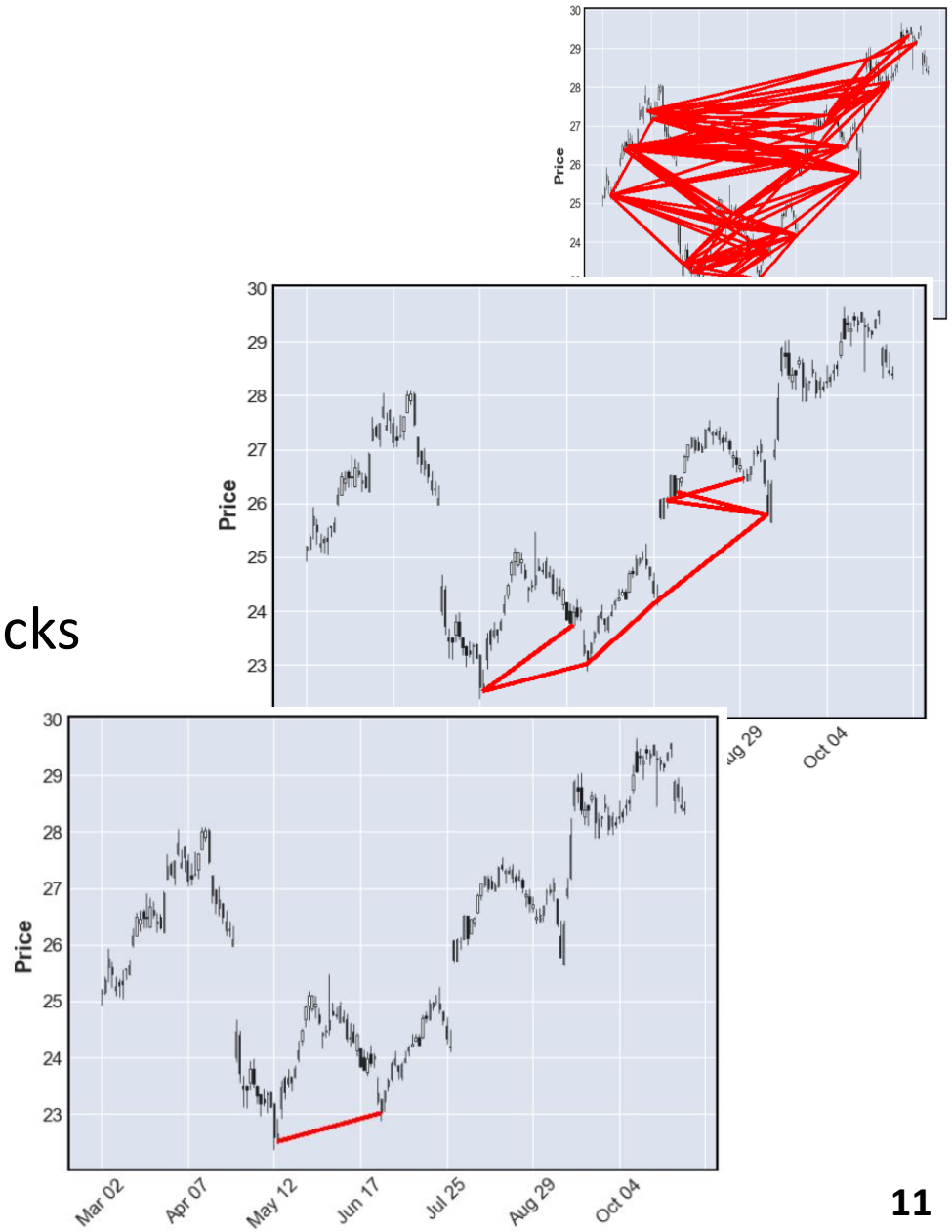




Hard-coded Detector

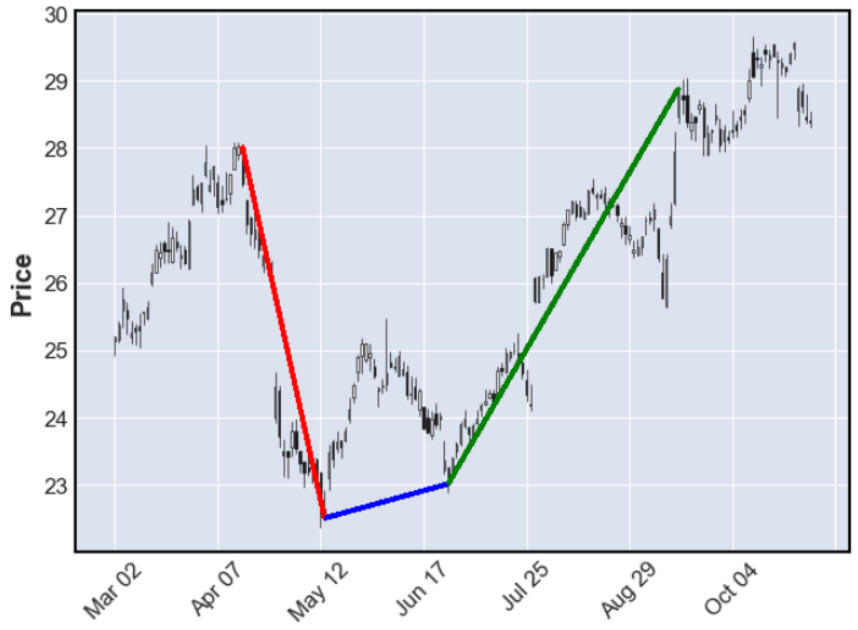
- Step 3: Filter the support lines
  - Check the existence of middle inverted “V”
    - Fit a parabola between the two bottoms
  - Support lines should not cut through candlesticks
  - No candlestick can under support lines





Hard-coded Detector

- Step 4: Check the existence of a 10-20% drop end at the first bottom
- Step 5: Check the existence and intensity of the last rise



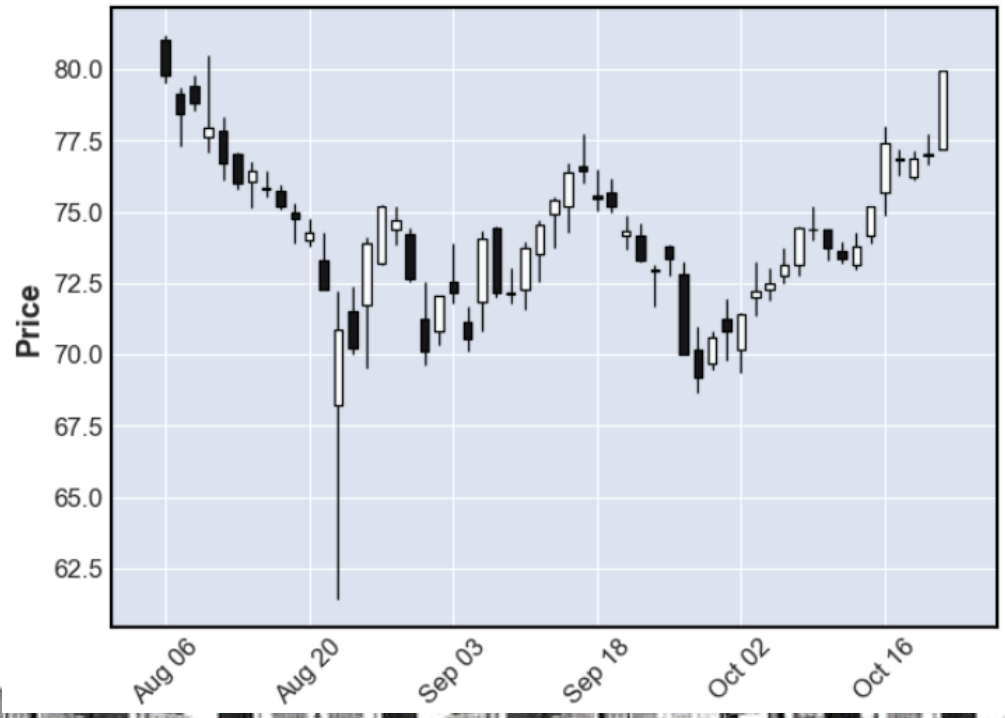
Dataset & preprocessing

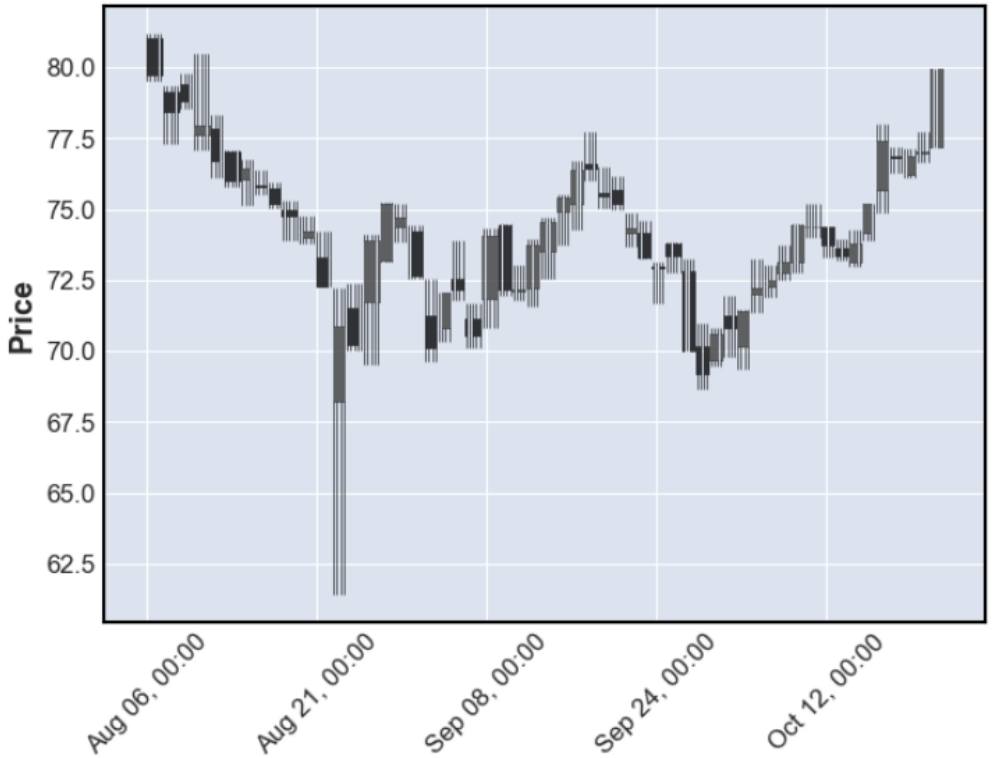
- 500 stocks from the S&P index, after filtering
  - 182 stock show double bottom pattern
  - We have 234 patterns in total
  - Timespan from 55 days to 235 days

- Prepare another 235 negative samples
  - Randomly picked from the rest 318 stock without replacement
  - For each sample, Timespan is randomly chosen from range 55 to 235 days

Dataset & preprocessing

- Unify the time span to 235
  - Repeating themselves
  - e.g. 64 days to 235 days





Machine learning detecter

- LSTM
- Suitable for time series data
- Input features: OHLC, adjusted close and volume
- Sequence length: 235 days
- Predict whether the input has double bottom patterns

LSTM – Experiment setting

- Cross validation to choose the hyperparameter
- 90% for training, 10% for testing
- Find the average of loss and accuracy
- Each model is composed of LSTM layers and dense layers
- Hyperparameter we want to test:
  - LSTM: [4], [8], [16], [32], [8, 4], [16, 4], [16, 8], [32, 16], [16, 8, 4]
  - Dense: [1], [4, 1], [8, 1], [16, 1], [16, 8, 1]

LSTM – Experiment result

LSTM / Dense	[4]	[8]	[16]	[32]	[8, 4]	[16, 4]	[16, 8]	[32, 8]	[32, 16]	[16, 8, 4]
[1]	0.726831	0.696726	0.699982	0.707643	0.766637	0.702040	0.698780	0.711815	0.723233	0.704372
[4, 1]	0.707077	0.693055	0.694595	0.703030	0.697822	0.806689	0.730751	0.751265	0.696239	0.693188
[8, 1]	0.694618	0.700404	0.701528	0.704191	0.765635	0.694593	0.693295	0.694566	0.707367	0.693120
[16, 1]	0.695024	0.709587	0.700311	0.703101	0.694556	0.693155	0.704758	0.694818	0.704564	0.694842
[16, 8, 1]	0.690570	0.698109	0.699208	0.708586	0.719594	0.695222	0.711063	0.690496	0.701868	0.693134

LSTM / Dense	[4]	[8]	[16]	[32]	[8, 4]	[16, 4]	[16, 8]	[32, 8]	[32, 16]	[16, 8, 4]
[1]	0.536232	0.543478	0.538043	0.538043	0.543478	0.532609	0.539130	/	0.539855	0.521739
[4, 1]	0.543478	0.586957	0.532609	0.534783	0.532609	/	0.528986	/	0.538043	0.543478
[8, 1]	0.543478	0.543478	0.530435	0.573913	0.532609	0.543478	0.521739	0.532609	0.554348	0.521739
[16, 1]	0.556522	0.543478	0.543478	0.539855	0.536232	0.543478	0.543478	0.536232	0.543478	0.543478
[16, 8, 1]	0.565217	0.526087	0.532609	0.536232	0.521739	0.543478	0.536232	0.565217	0.543478	/

- The best combination
  - LSTM: [4]
  - Dense: [16, 8, 1]
- Average accuracy: 56.5%

Calculation of the average


```
{'lstm': [32, 8],
 'dense': [1],
 'train_loss': [0.6783584356307983,
  0.6860380172729492,
  0.6598420143127441,
  7.712474822998047,
  7.712474346160889,
  0.6632499694824219,
  0.6781659126281738,
  0.6660026907920837,
  7.712474346160889,
  0.6869624257087708],
 'val_loss': [0.6989296078681946,
  0.6965937614440918,
  0.7346721887588501,
  7.7124738693237305,
  7.7124738693237305,
  0.7233051657676697,
  0.7057515978813171,
  0.7307489514350891,
  7.7124738693237305,
  0.6927011609077454],
 'train_accu': [0.5614973306655884,
  0.5,
  0.6203208565711975,
  0.5,
  0.5,
  0.6096256971359253,
  0.5614973306655884,
  0.6203208565711975,
  0.5,
  0.5614973306655884],
 'val_accu': [0.5,
  0.5,
  0.43478259444236755,
  0.5,
  0.5,
  0.47826087474823,
  0.5,
  0.41304346919059753,
  0.5,
  0.47826087474823]}
```

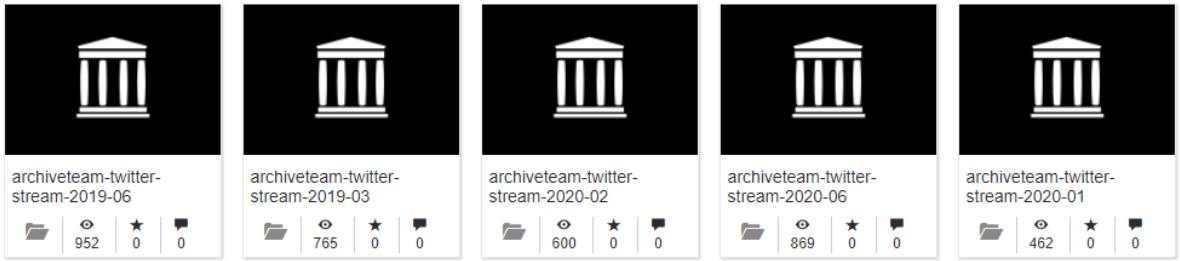

- Unsuccessful training
  - Ignore the result
  - Loss ≈ 7.71
  - Accuracy: 0.5
- Calculate the average without them
  - The reason why we have “/” in previous page.

Social Media Analysis

Twitter
Reddit
YouTube

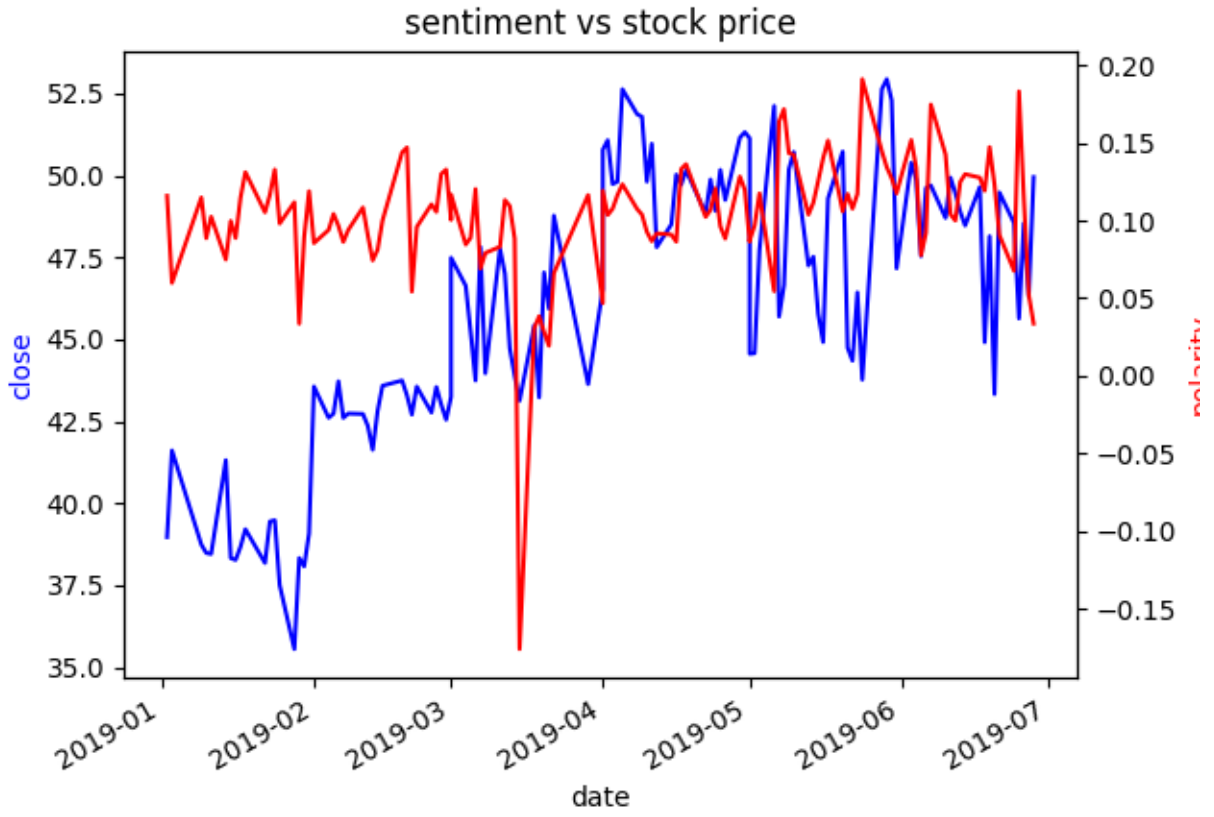
Twitter (IA)

- Internet Archive
  - non-profit, public digital library provides free public access to collections of digitized materials.
  - Including a collection of Twitter tweets
- Collected Features:
  - Date, user, tweets, hashtags, links, etc
- In this experiment
  - Date: 2019-01 to 2019-06
  - Total tweets: ~128GB
  - Sentiment Analyzer: VADAR Sentiment



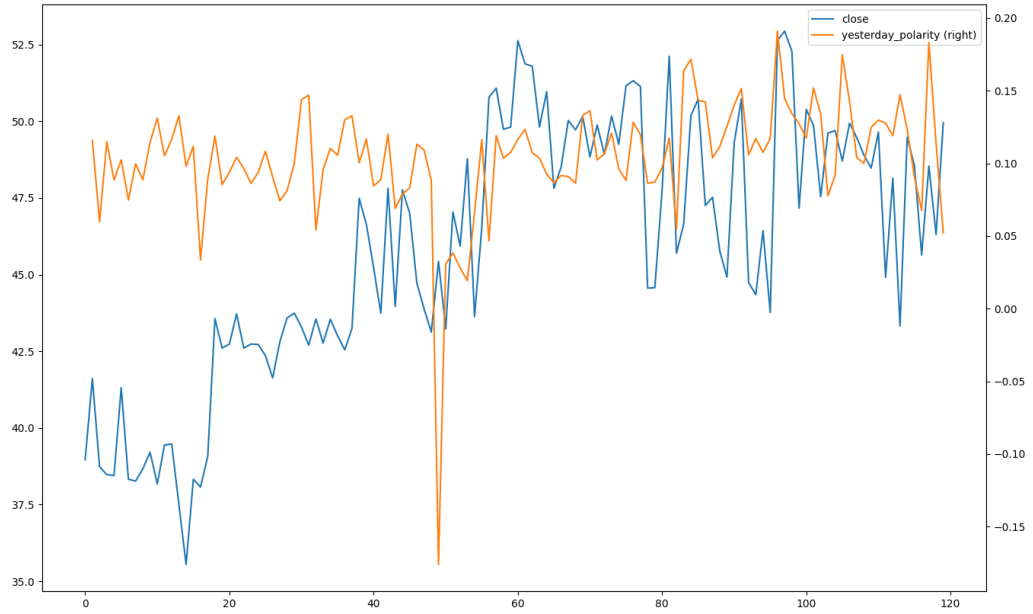
Twitter (IA)

- Raw sentiment vs stock price
- Covariance: 0.023761
- Correlation: 0.139869
- Aim: see the relationship
- Outcome: can use to predict stock price



Twitter (IA)

- Yesterday polarity vs stock price
- Covariance: 0.032677
- Correlation: 0.19541
- Aim: see if yesterday polarity can use to predict today stock price
- Outcome: suitable to predict stock price



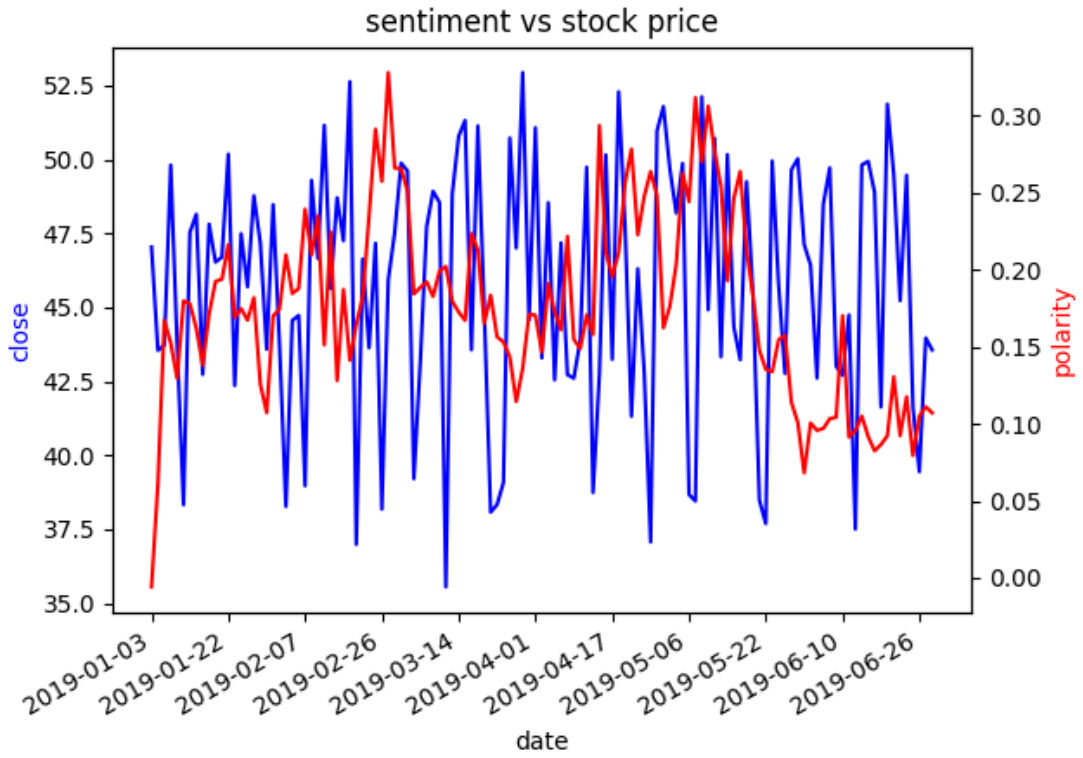
Twitter (Twint)

- Twint – python library to scrape tweets without worry about the limitation of official API
- Collected Features:
  - Date, user, tweets, hashtags, links, etc
- In this experiment:
  - Date: 2019-01 to 2019-06
  - Total tweets: 233217 records
  - Sentiment analyzer: VADAR Sentiment
  - Others: Fourier transform

	Fourier polarity	Fourier price	covariance	correlation
1	N/A	N/A	-0.015243	-0.059368
2	5	5	0.004014	0.228205
3	5	10	0.004586	0.081181
4	5	15	0.004586	0.048984
5	5	20	0.004586	0.041089
6	10	5	0.004586	0.240461
7	10	10	0.000304	0.004969
8	10	15	0.000696	0.006855
9	10	20	0.000696	0.00575
10	15	5	0.004586	0.231462
11	15	10	0.000696	0.010936
12	15	15	-0.002506	-0.023763
13	15	20	-0.003914	-0.031133
14	20	5	0.004586	0.226844
15	20	10	0.000696	0.010718
16	20	15	-0.003914	-0.036375
17	20	20	-0.008449	-0.06586

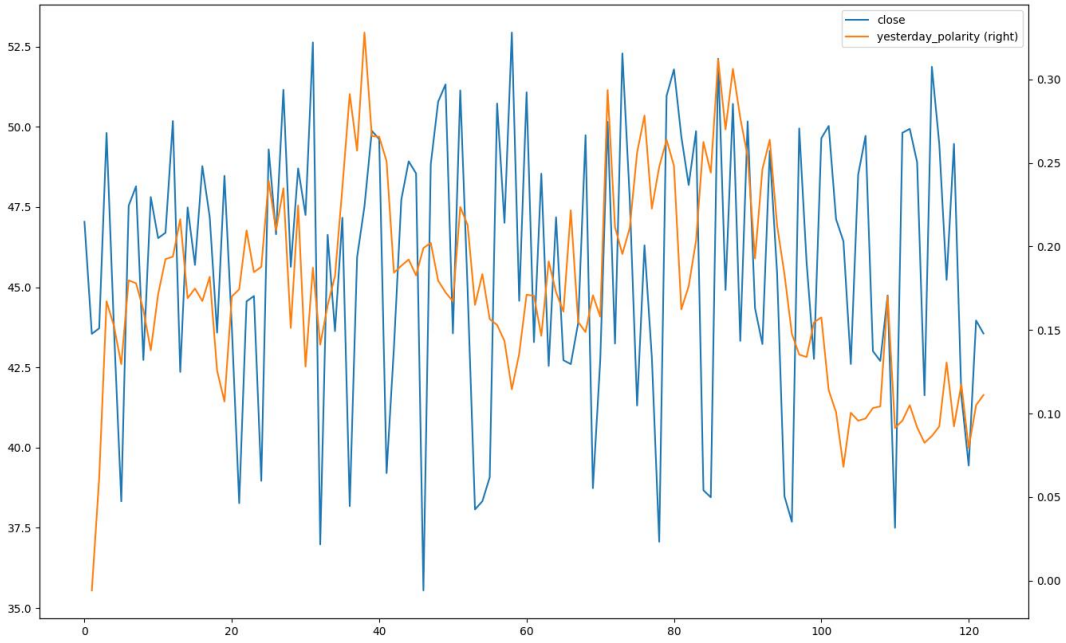
Twitter (Twint)

- Raw comparison
- Covariance: -0.015243
- Correlation: -0.059368
- Aim: see the relationship



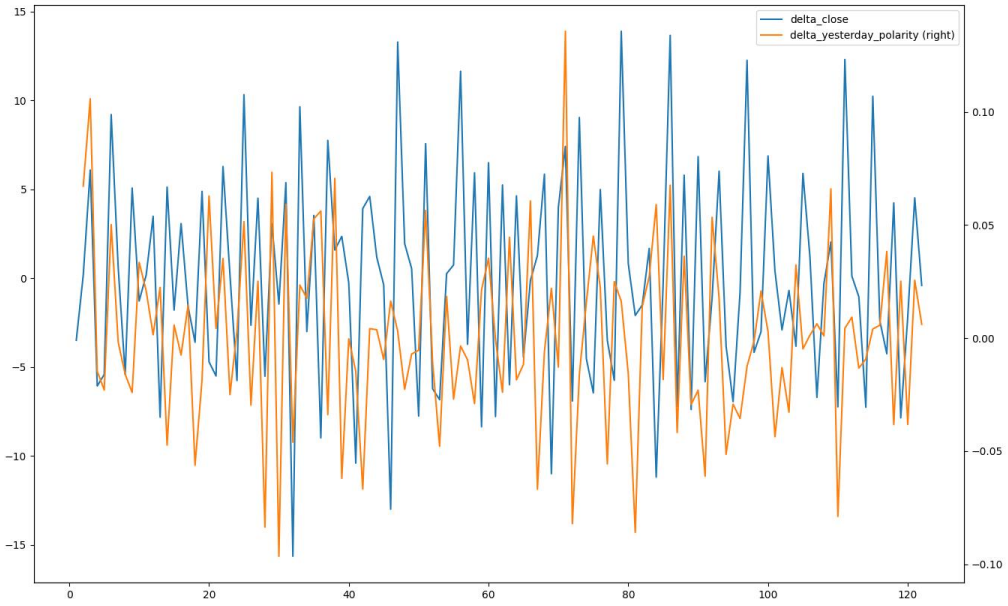
Twitter (Twint)

- Yesterday polarity vs stock price
- Covariance: 0.015503
- Correlation: 0.060237
- Aim: see if yesterday polarity can use to predict the next stock price



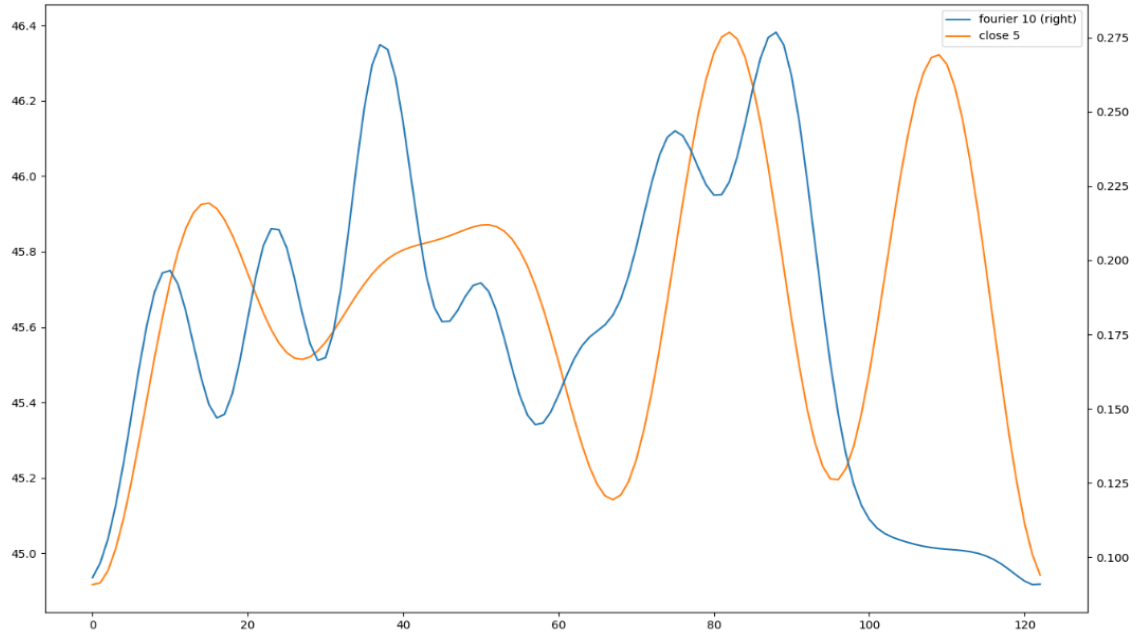
Twitter (Twint)

- Delta yesterday polarity vs delta stock price
- Covariance: 0.050172
- Correlation: 0.201895
- Aim: see if this comparison better than previous one
- Outcome: we can use this to predict the rise/fall of tomorrow stock price



Twitter (Twint)

- Fourier polarity 10 vs Fourier price 5
- Covariance: 0.004586
- Correlation: 0.240461



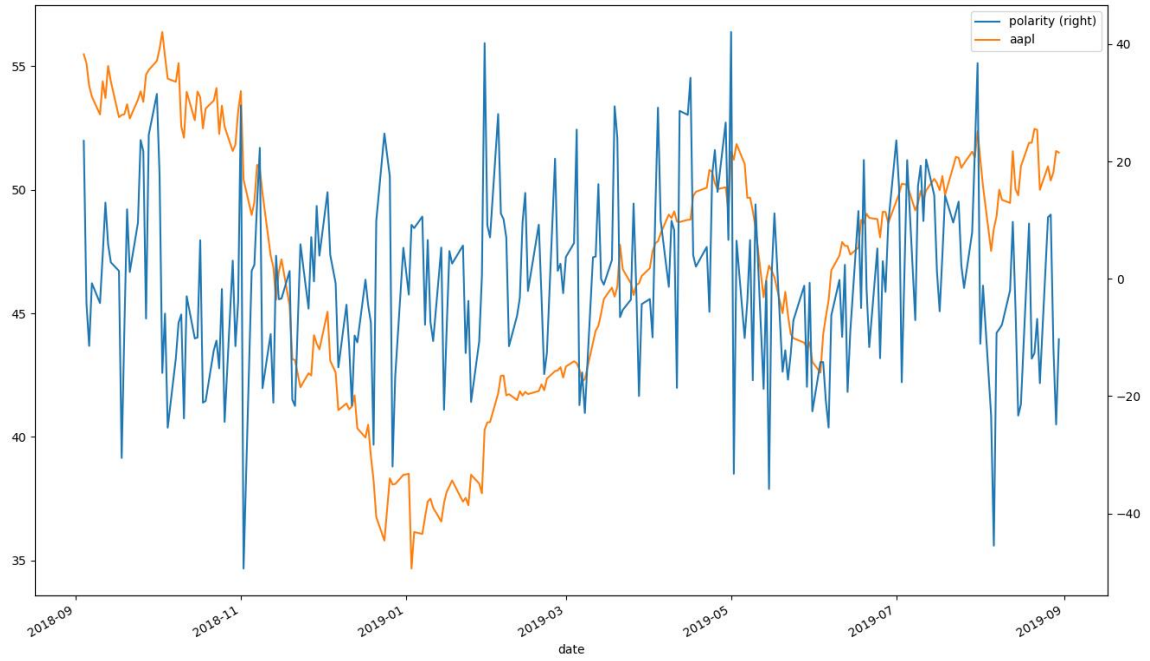
Reddit

- PRAW – python reddit API wrapper
- Collected Features:
  - Date, user, comments, etc
- In this experiment:
  - Date: 2018-09-01 to 2019-09-01
  - Total comments: 123857 records
  - Sentiment analyzer: VADAR Sentiment
  - Others: Fourier transform

	Fourier polarity	Fourier price	covariance	correlation
1	N/A	N/A	5.413715	0.068209
2	5	N/A	0.791039	0.030308
3	10	N/A	2.285104	0.065746
4	15	N/A	2.628963	0.069038
5	20	N/A	3.745961	0.087944

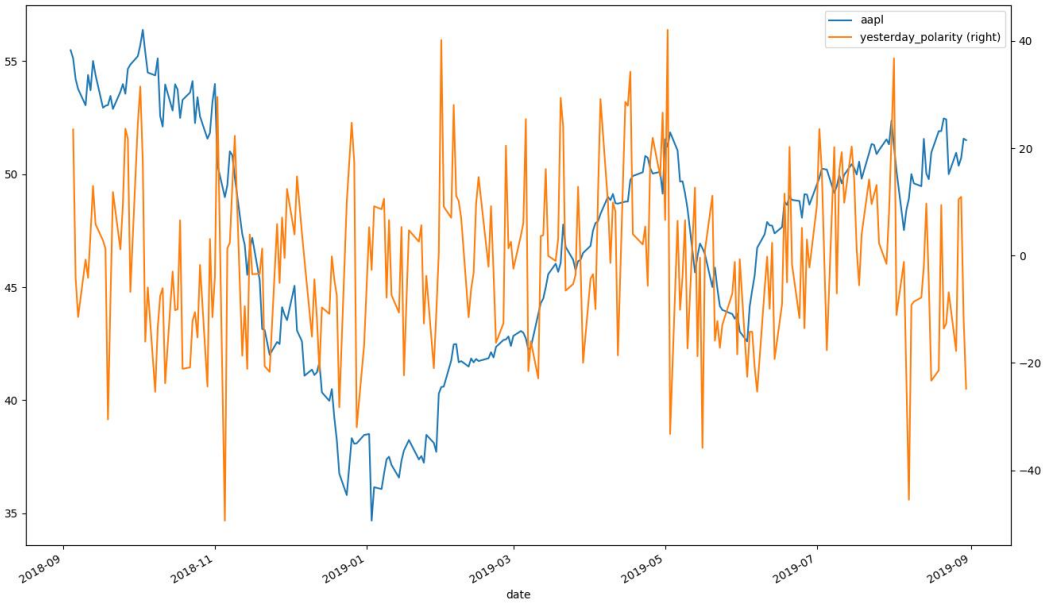
Reddit

- Raw comparison
- Covariance: 5.413715
- Correlation: 0.068209
- Aim: see their relationship



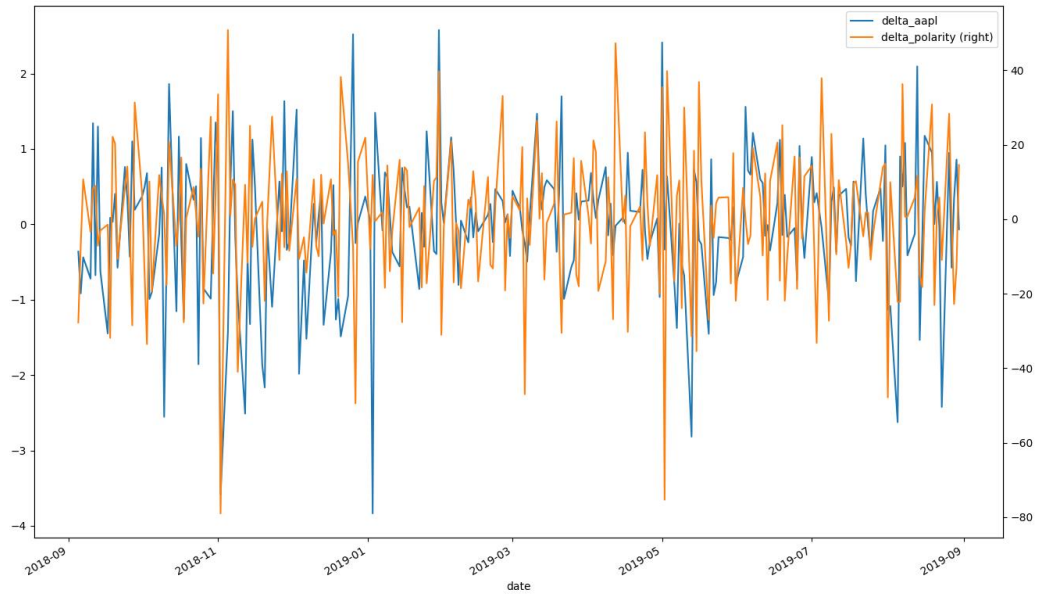
Reddit

- Yesterday polarity vs stock price
- Covariance: 5.726788
- Correlation: 0.072307
- Aim: see if yesterday's polarity can use to predict tomorrow's stock price
- Outcome: yesterday's polarity can use in prediction



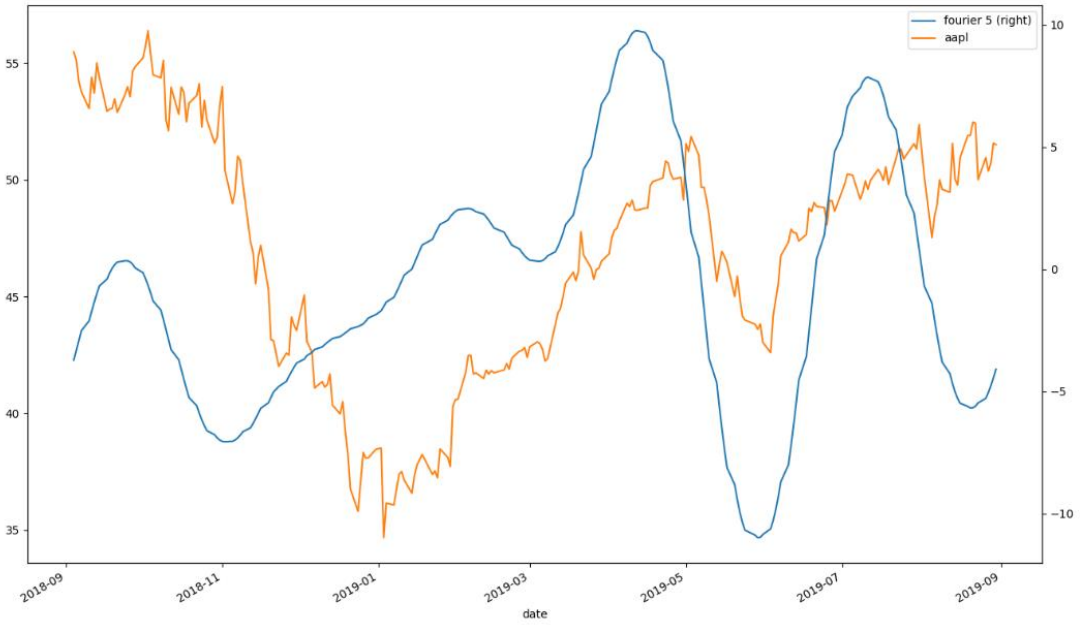
Reddit

- Delta polarity vs delta stock price
- Covariance: 3.453137
- Correlation: 0.19473
- Aim: see ↑/↓ polarity result in ↑/↓ stock price
- Outcome: ↑/↓ polarity = ↑/↓ stock price



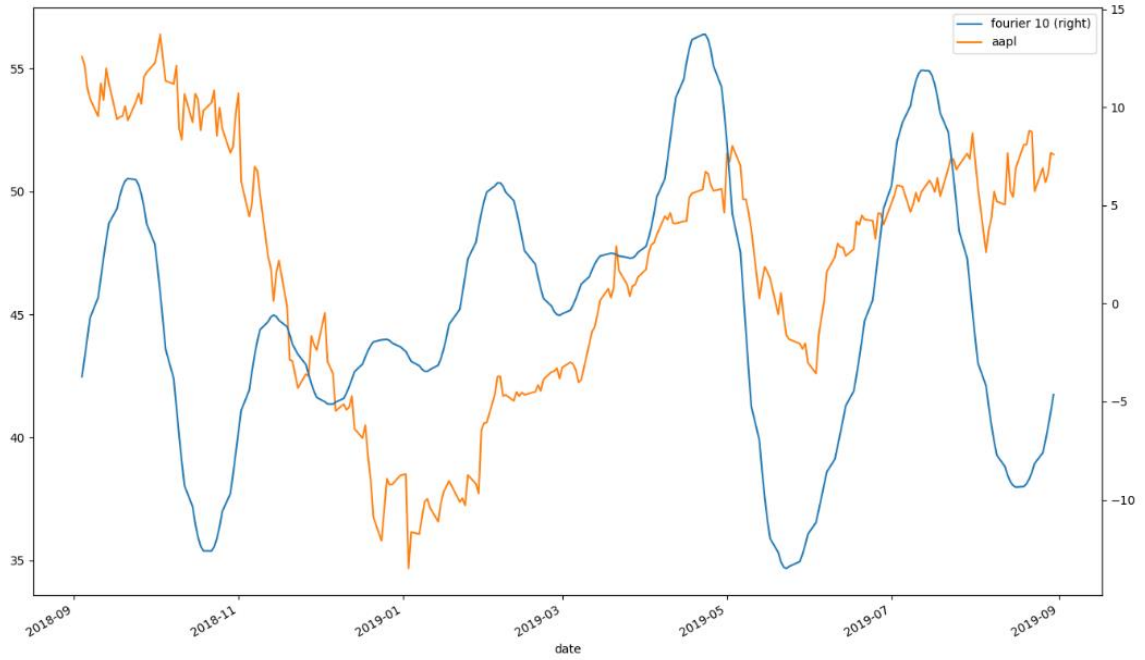
Reddit

- Fourier 5 polarity vs stock price
- Covariance: 0.791039
- Correlation: 0.030308



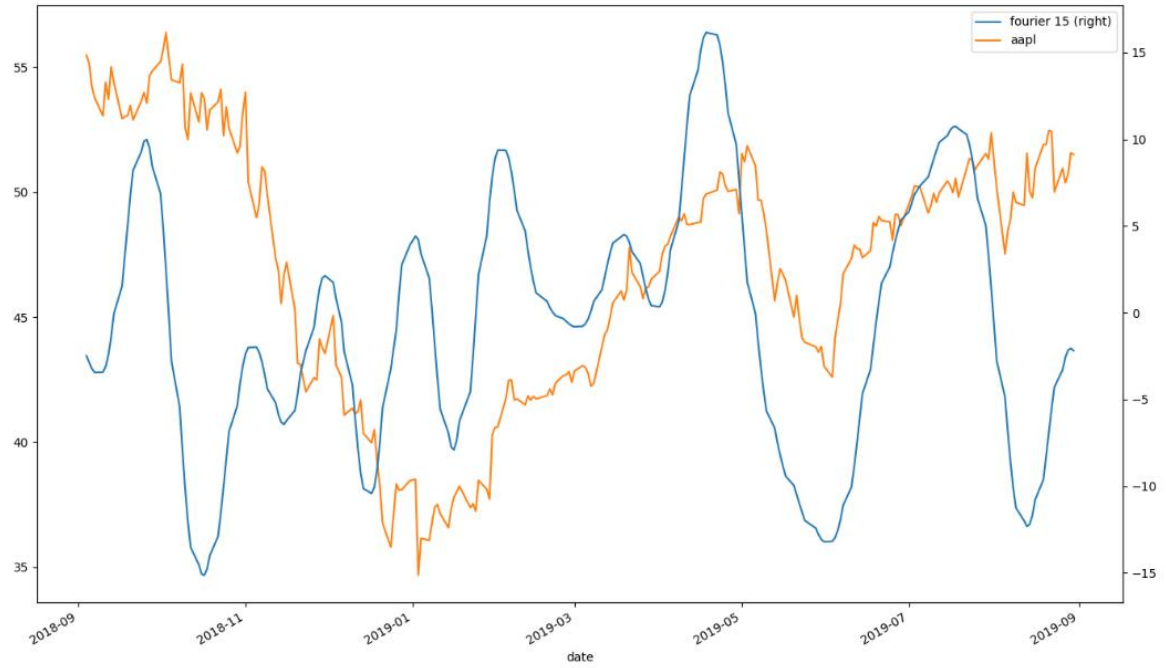
Reddit

- Fourier 10 polarity vs stock price
- Covariance: 2.285104
- Correlation: 0.065746



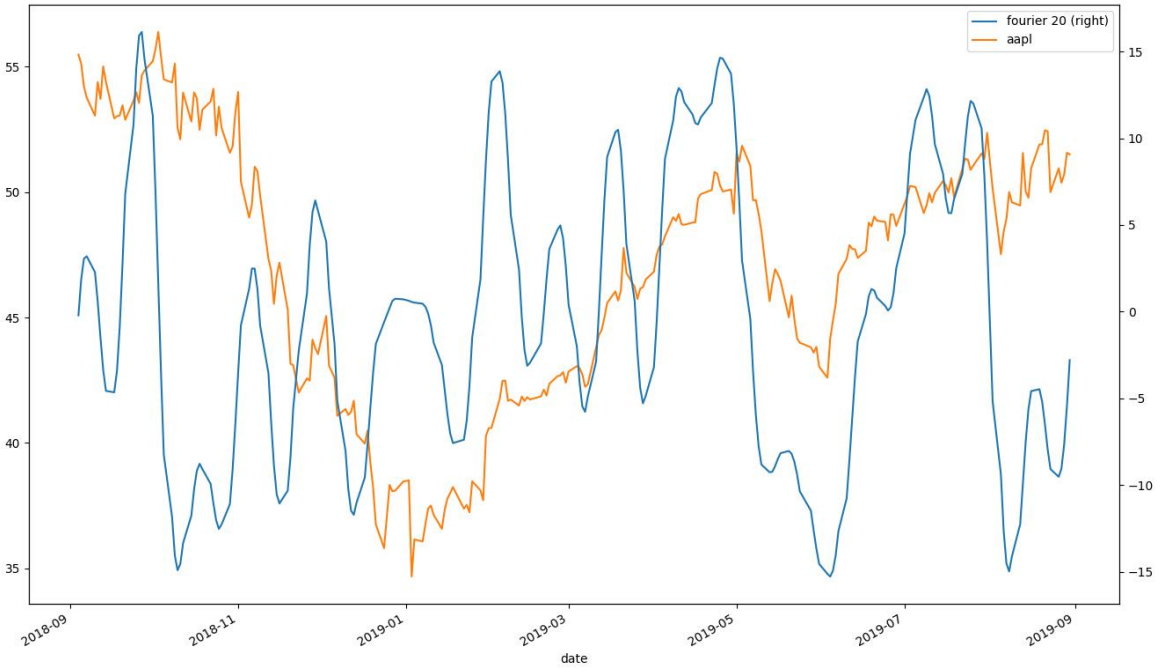
Reddit

- Fourier 15 polarity vs stock price
- Covariance: 2.628963
- Correlation: 0.069038



Reddit

- Fourier 20 polarity vs stock price
- Covariance: 3.745961
- Correlation: 0.087944



YouTube Sentiment Analysis

- Data source: YouTube Data API
- Search for a list of videos related to Apple
  - Limit to the science and technology category
- Request for comments for each video

- Limitation of the API
  - Hard to get old comments. API limits us to request comment started from the latest
  - Quota limits the volume of use

YouTube Comment Dataset

- Timespan: 2021-01 to 2021-03
- 80K+ comments
- Preprocessing
  - Sterilizing
    - Remove reply tags, HTML tag, URLs, emoji
  - Information extraction
    - Remove punctuation, stop words
  - Spelling correction
- Sentiment analysis
  - VADAR Sentiment analysis

YouTube Comments and Stock Data

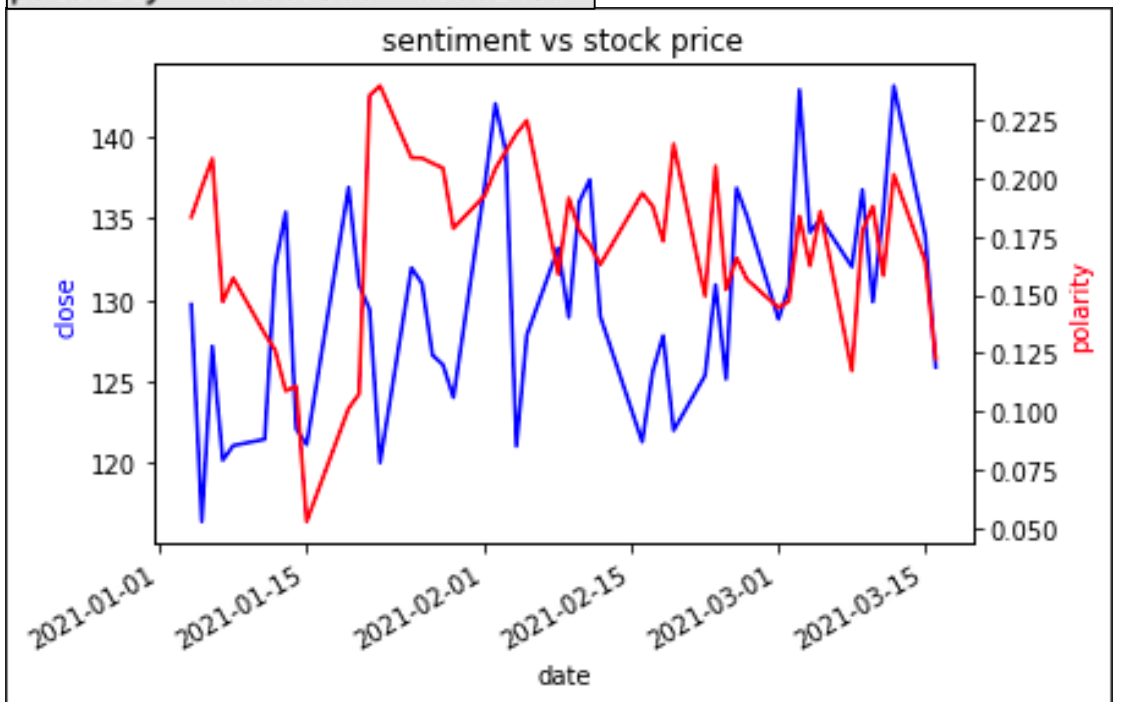
- Study the relation of them step by step
- Summary of testing

	Graph	Covariance	Correlation
1 (Base case)	Sentiment vs stock price	0.00796	0.125
2	Change in sentiment vs change in stock price	0.0439	0.541
3	Yesterday sentiment vs change in stock price	0.0508	0.661

YouTube Comments and Stock Data

- Sentiment vs stock price
- To check the covariance of
  - The sentiment of YouTube comments
  - Stock close price
- As a base case for later comparison
  - Red line: sentiment value (polarity)
  - Blue line: close price
- Covariance: 0.00796
- Correlation: 0.125

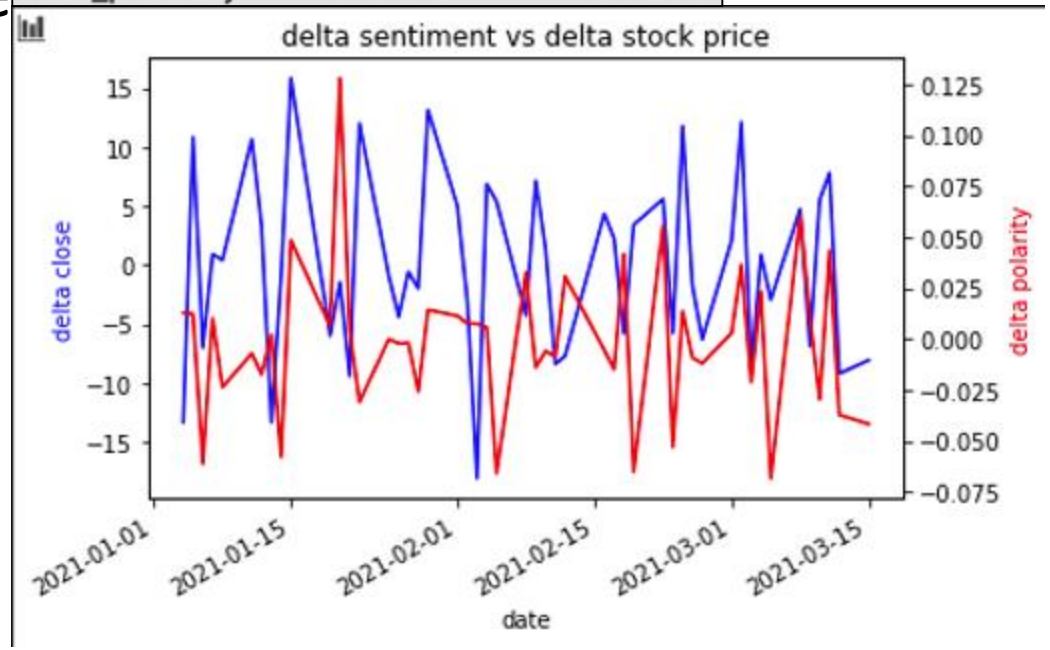
	close	polarity
close	42.579533	0.007960
polarity	0.007960	0.001496



YouTube Comments and Stock Data

- Delta sentiment vs delta stock price
- Aim to study if a positive change in the comment will lead to the stock rise
- Compared to the base case:
  - Covariance: 0.00796 -> 0.0439
  - Correlation: 0.125 -> 0.541
- Spikes quite matched
- Positive change in comments likely to see the stock rise

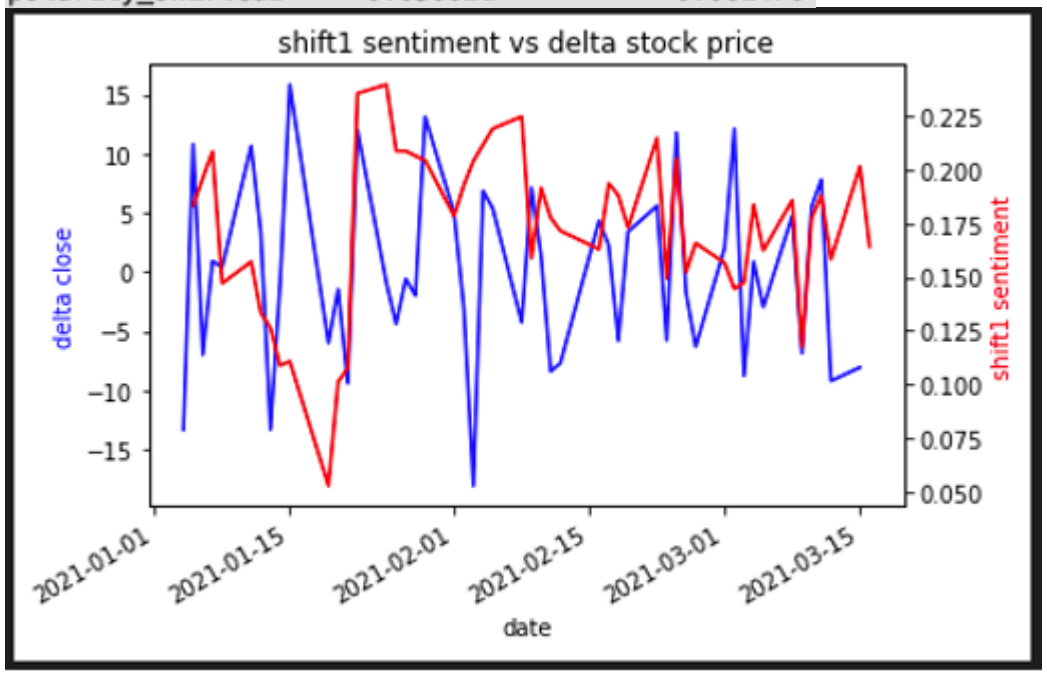
	delta_close	delta_polarity
delta_close	59.668095	0.043852
delta_polarity	0.043852	0.001357



YouTube Comments and Stock Data

- Yesterday sentiment vs delta stock price
- Aim to study if it is possible to use sentiment value to predict the stock price
- Compared to the previous case:
  - Covariance: 0.0439 -> 0.50821
  - Correlation: 0.541 -> 0.661
- Using yesterday sentiment value can give us a better prediction

	delta_close	polarity_shifted1
delta_close	59.668095	0.050821
polarity_shifted1	0.050821	0.001476



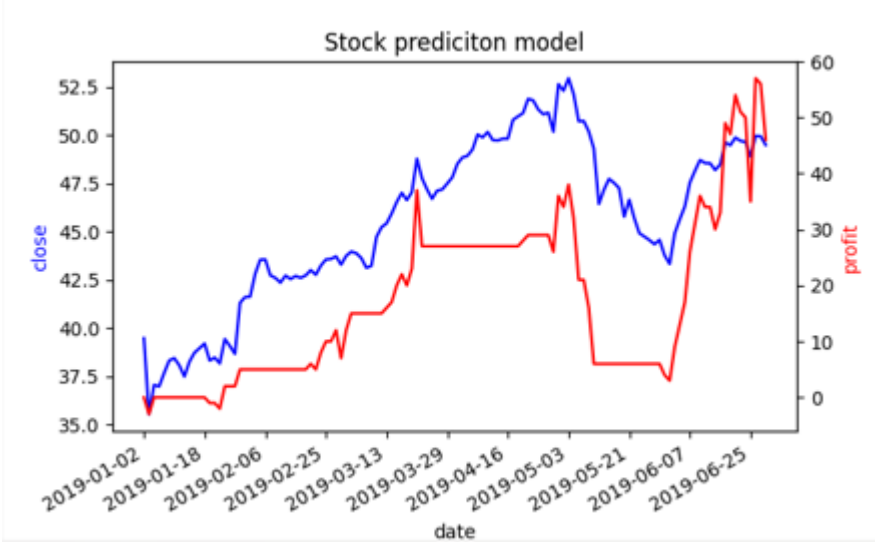
Tool Demonstration

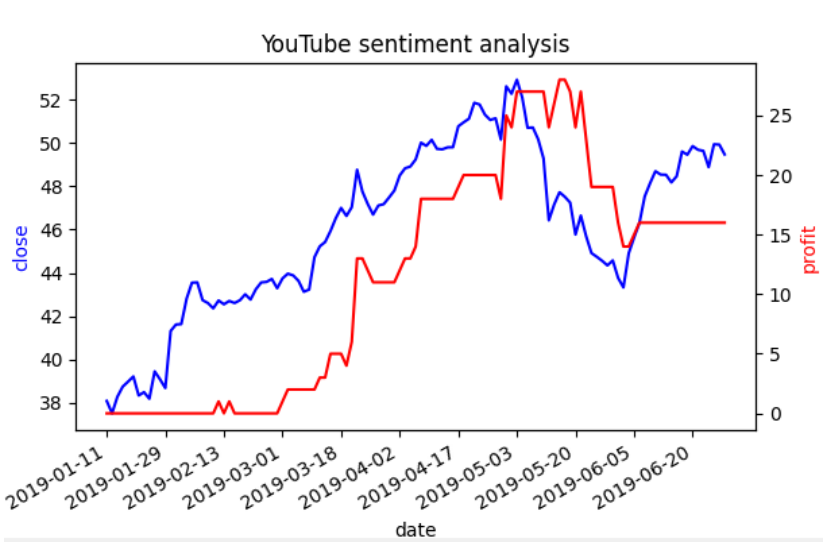
Investment tool

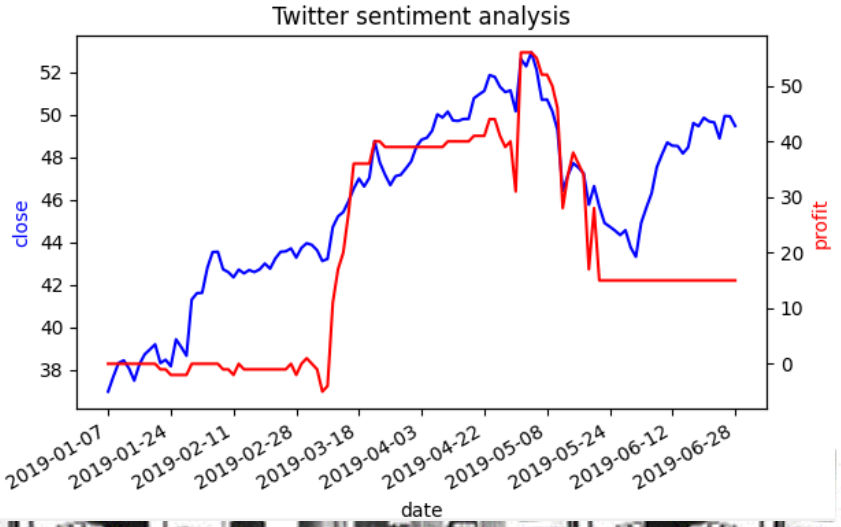
- Wrap up our project by integrating them into a tool
- A tool that provides our founding in these two semesters
  - Stock prediction using our term 1 model
  - Pattern recognition
  - Social media sentiment analysis
  - Backtest
- Tool demonstration

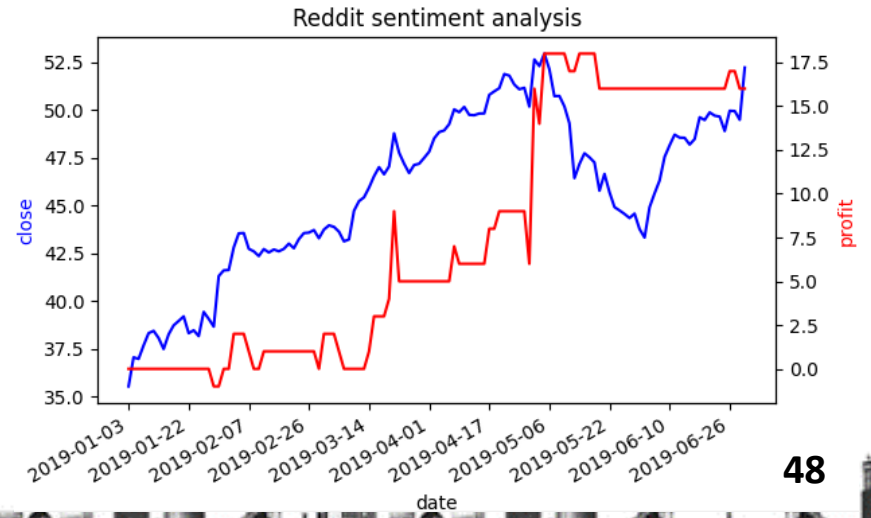
Investment tool - backtest

Strategy	Profit
Prediction model	46
YouTube sentiment	16
Twitter sentiment	15
Reddit sentiment	16





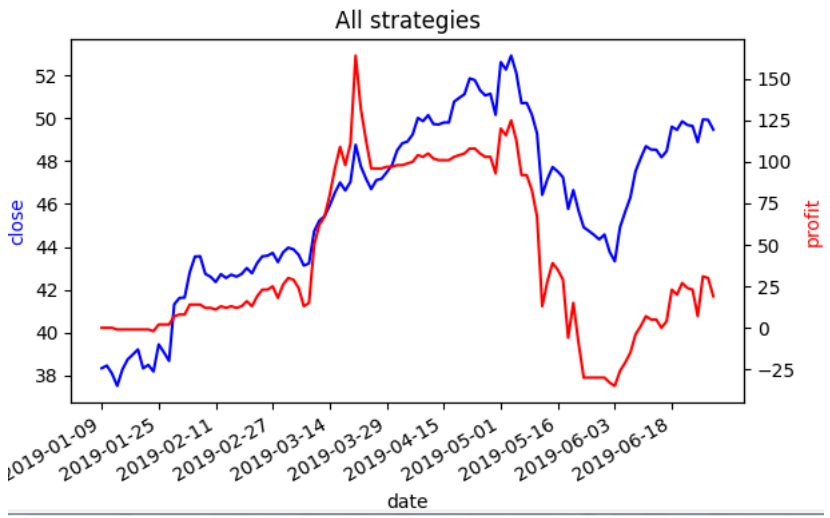




Investment tool - backtest

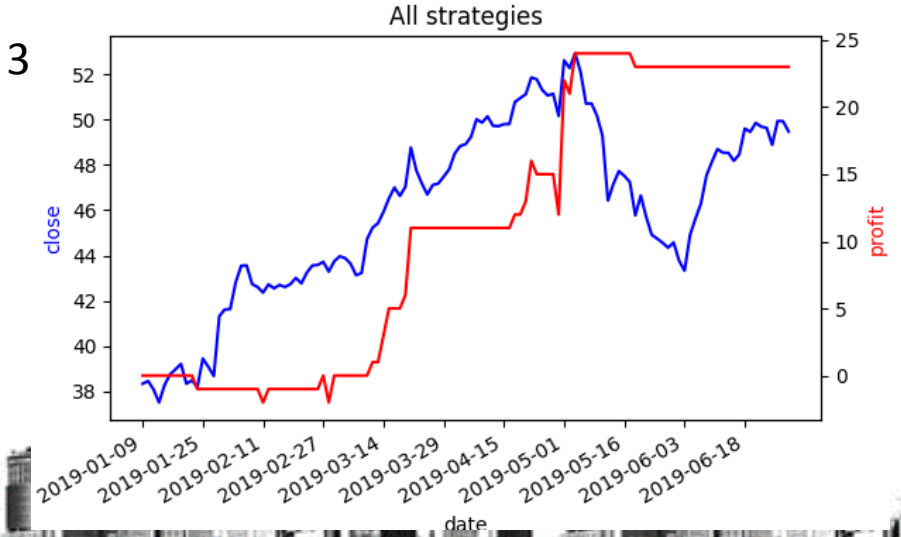
Minimum	Profit
1	19
2	-21
3	23
4	11

1



2

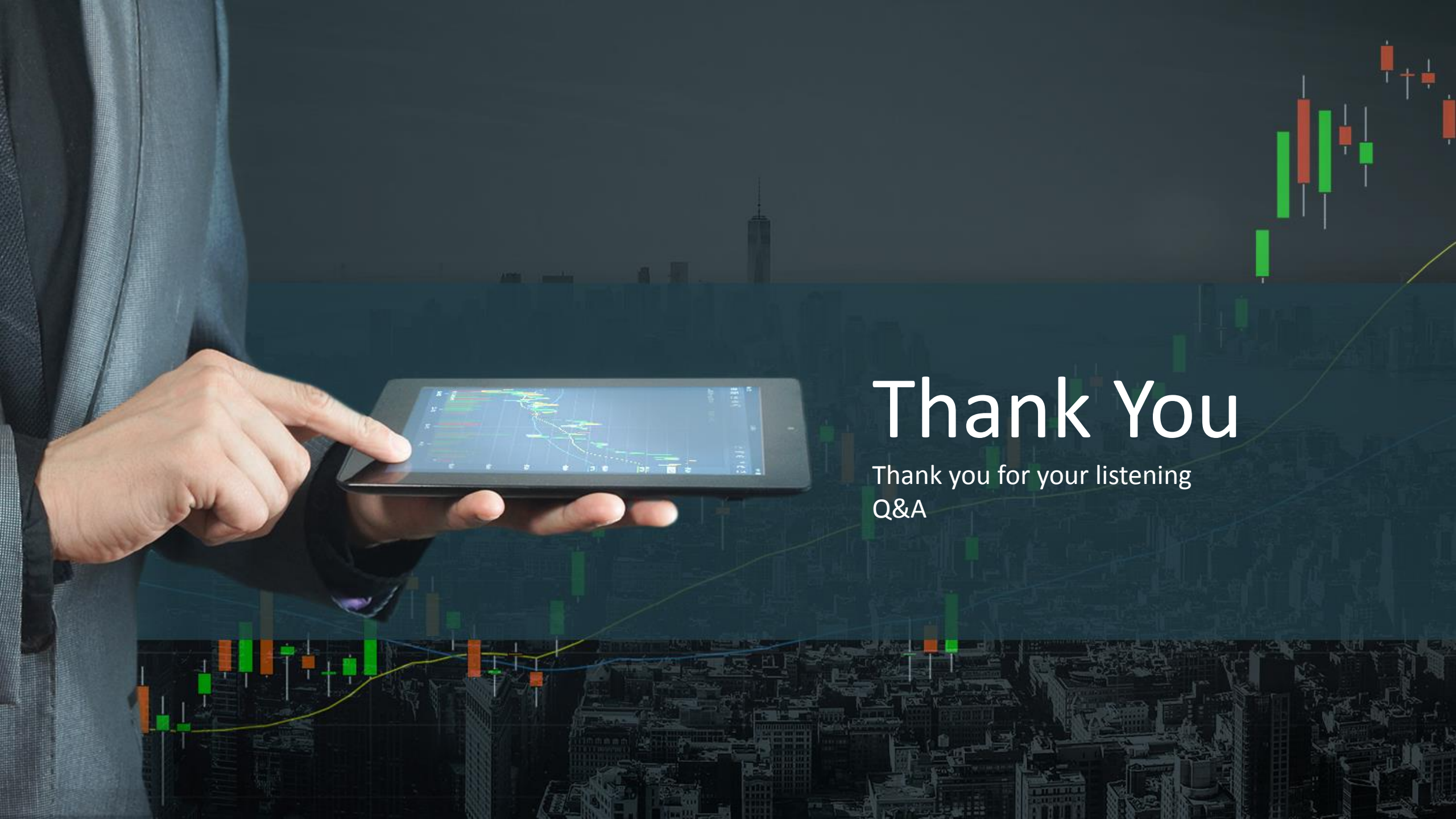
3



4

Conclusion

- Pattern recognition
  - The outcome of the training is not satisfying, many unsuccessful training
  - No existing dataset and quality of the hardcode detector is not good enough
  - Dataset size is too small
- Twitter (Internet Archive)
  - Inconsistent in data
  - Takes too much time to process the data
- Twitter (Twint)
  - Using time lag & delta gives a better correlation
- Reddit
  - Raw data is good compared to Twitter, adding delta gives a better result
- YouTube sentiment shows us an interesting finding
  - Adding time lag gives us a better correlation
  - Possibility of using YouTube comment as a media to predict the stock



Thank You

Thank you for your listening

Q&A