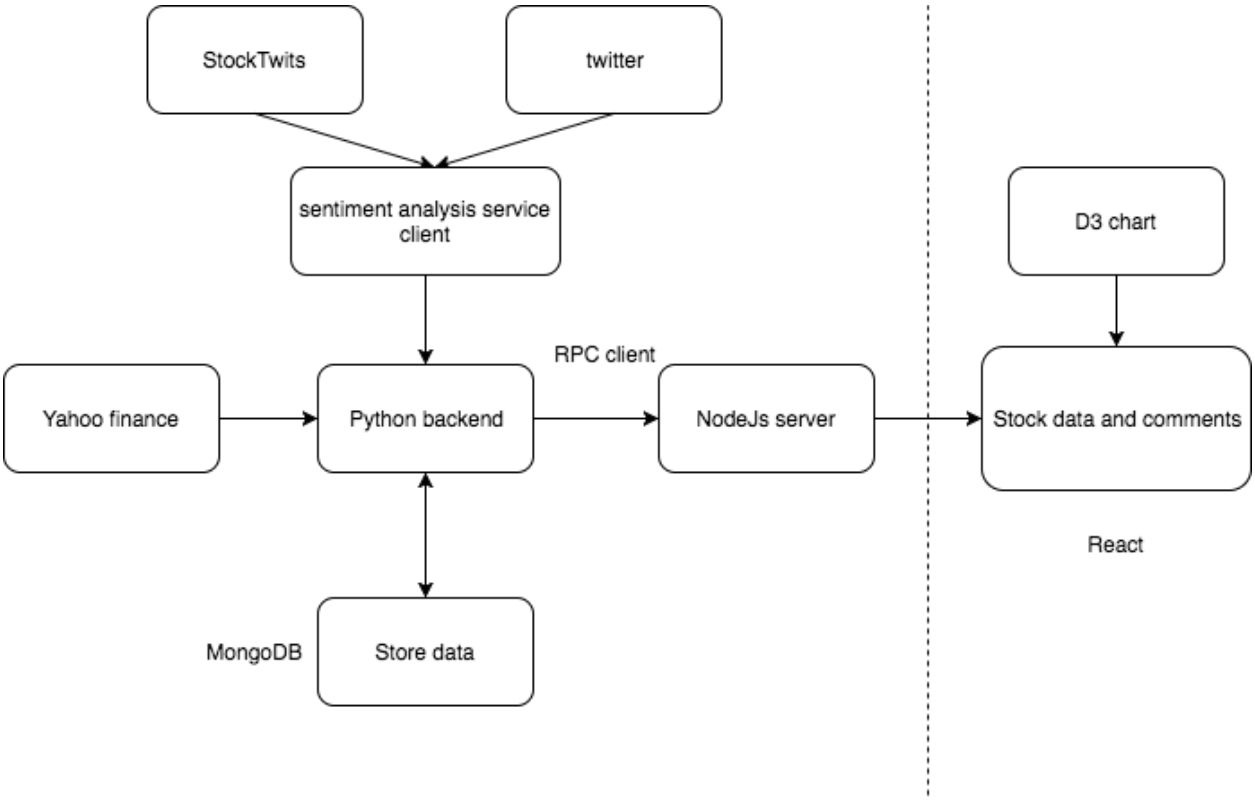
StockList App

Yifan Tian

Overview:

- Show list of stocks with data and chart.
- Show relevant comments from Stocktwits and Twitter.
- Performing sentiment analysis on tweets using both NLTK and Azure cognitive service

Structure:



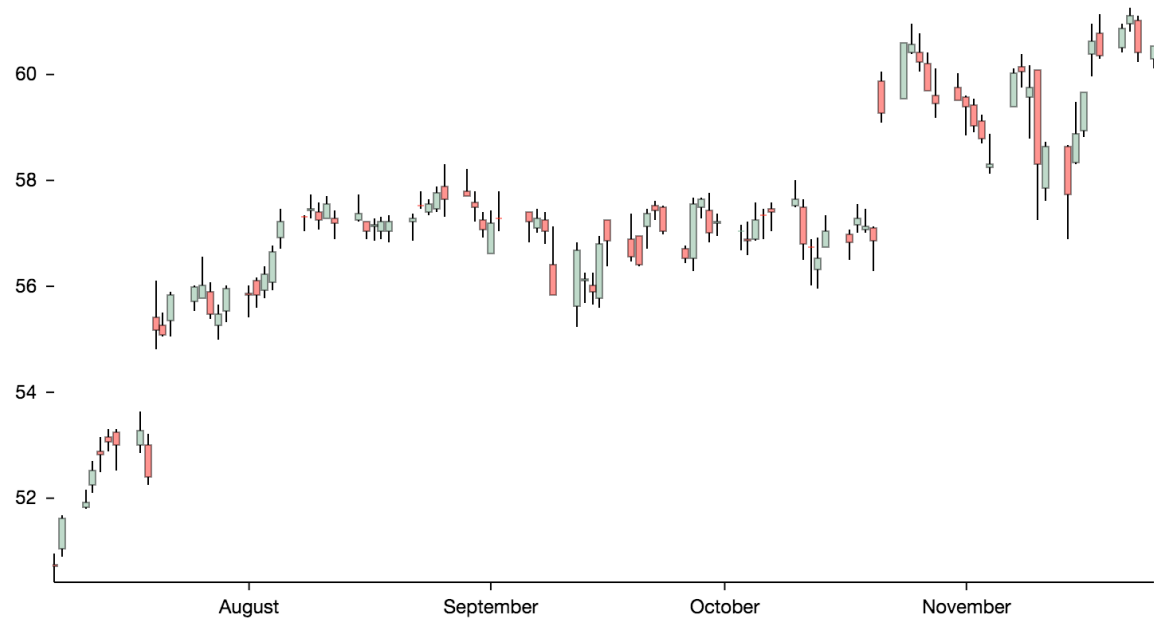
Front End:

Main components:

- Navbar
- Stocks Cards (stocks info)

Chart:

D3stock chart:



BackEnd:

Stock source:

1. Yahoo finance
   https://github.com/lukaszbanasiak/yahoo-finance

2. Google finance
   https://github.com/hongtaocai/googlefinance

3. Historical data using yahoo finance
   https://github.com/c0redumb/yahoo_quote_download

Tweets source

1. **StockTwits**

- Get stock of one id:
  https://api.stocktwits.com/api/2/streams/symbol/AAPL.json

Get body and analyze the body and show the provided sentiment.

{"id":95022156,"body":"Estimize revenue expectations are 0.60% higher than that of Wall Street for $AAPL Q4 [Reporting 10/24 AMC]
utm_content=AAPL&utm_medium=eps_update&utm_source=stocktwits#chart=historical","created_at":"2017-09-16T01:09:06Z
{"id":727510,"username":"EstimizeAlerts","name":"Estimize","avatar_url":"https://avatars.stocktwits.com/production/727510/thumb-
1461073766.png","avatar_url_ssl":"https://avatars.stocktwits.com/production/727510/thumb-1461073766.png","join_date":"2016-04-

2. **Twitter**
   Twitter API.

Analysis of the sentiment:

1. NLTK

- Using “negative” or “positive” or “neutral”

  PG Option Order Flow Sentiment is 78 3 Bullish

  NLTK: neutral

  NKE has hit a perfect moonshot with the new NBA collection

  NLTK: positive

- Not sensitive to some domain-specific words.

2. Azure cognitive sentiment analysis.
   Using a score to evaluate the sentiment.
   Examples:
   'text': 'I really enjoy the new XBox One S. It has a clean look, it has 4K/HDR resolution and it is affordable.’
   ‘text’: ‘Este ha sido un dia terrible, llegué tarde al trabajo debido a un accidente automobilistico.’
   "score": 0.99984133243560791
   "score": 0.024017512798309326

DB: MongoDB
Two tables:

1. Stocks with stocktwits data
2. Tweets data

Tweets data streaming:

Kafka producer and consumer.

- **Producer:**
  producer = KafkaProducer(bootstrap_servers='localhost:9092')
  producer.send('AAPL', {'tweet': text, 'digist':digist})

- **Consumer:**
  consumer = KafkaConsumer('AAPL')
  for msg in consumer:
  tweet = ast.literal_eval(msg.value)

Further Application:
Allow user to login and save Favorites.
Allow user to search symbols.
Save more data and provide prediction using sentiment analysis.