Social Media based Stock Market Analysis using Big-Data Infrastructure

Shashanka Venkatesh 18 5001 145
Venkataraman Nagarajan 18 5001 192
Vishakan Subramanian 18 5001 196

BE CSE, Semester 8

Dr. N. Sujaudeen
Supervisor

Project Review: 2 (22 April 2022)
Department of Computer Science and Engineering
SSN College of Engineering

---

1 Abstract

There are several factors that influence the value of a stock apart from the typical quantitative and qualitative parameters seen in the fundamental analysis of stocks like balance sheets, income statements, cash flow statements etc. One of the most influential factors in recent years are social media trends.

The aim of this research is to analyze and understand the effect of such articles over the value of a stock at any given time. In our proposed methodology, we consider Twitter, a popularly used micro-blogging site as our social media source. We perform sentiment analysis on collected Tweets, and measure their correlation to stocks of companies within the sector the Tweet appeals to. To analyze such voluminous amounts of data in an efficient & effective manner, we propose to build an architecture that makes use of popular tools for analysing large-scale data at real time, like Apache Spark & Apache Kafka.

2 Proposed Work

Predicting how the stock market moves is a challenging issue due to many factors involved in it, like interest rates, economic growth, current trends, politics, etc.

Investors use various sources of information to determine whether or not to buy/sell stocks of any company. Two such sources that are widely used (especially in recent years) are news events & social media trends. Social media trends particularly seem to be influential in decision-making for the novice stock market investor, and thus its influence on present day stock market cannot be ignored.

Henceforth, we wish to understand and analyze the statistical correlation between social media trends and its effect on the relevant industry's stock price valuations.

To evaluate the correlation, we require a large sample space with periodically collected data points from various trustworthy sources of information. Stock data, current news & social media trends tend to be voluminous with a high degree of velocity. To appropriately wrangle such copious amounts of raw data to extract valuable information out of it requires the usage of a big-data architecture.

Our research works with the data collected from the time period from 04 April 2020 to 20 March 2022 - a reasonably wide sample space to help us obtain a fair perspective of the inherent correlation between social media trends and the stock market. We chose to work with sectors that are also considerably discussed about in online forums viz. Tech, Oil, Electric Vehicles (EVs), Gaming and and Cryptocurrency.

Using Twitter as the social media entity to capture public mood is justified by the research conducted by Mittal et al.[14]

To appropriately relate the Tweet trends of different market sectors to their respective industries, we consider stocks of a few companies that have a high market share in their sectors. For example, to correlate the tweets made regarding the EV sector, we consider the stock prices of Tesla Motors (NASDAQ: TSLA) and Lucid Group (NASDAQ: LCID) for analysis. This leads to a more concrete understanding of the inherent correlation than compared to the work done by Lee et al.[1] which uses the Dow Jones Industrial Average (INDEXDJX) for their correlation analysis.

3 Architectural Design

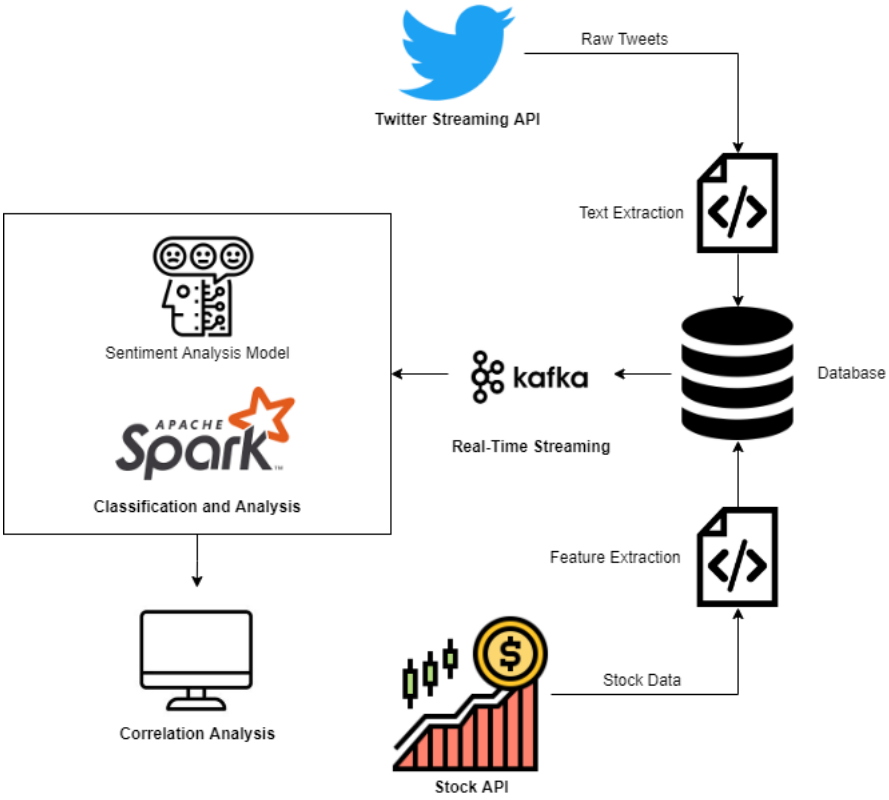

Figure 1: Proposed Architecture

3.1 Twitter Data Pre-Processing

The Twitter Streaming API allows to query tweets with certain keywords or hashtags in them. The collected tweets are processed to get the raw text in them, and are then stored in a database. But this method is prone to having false positives, i.e., tweets containing the keyword we're looking for, but are not actually relevant to the sector itself.

To collect relevant Tweets, we use the official Twitter API v2 with the Academic Research access level[15], which provides us with access to historical Tweets and an extensive filtering mechanism, along with increased API limits. Using appropriate filters based on related keywords like `"$TSLA"`, `"$ATVI"`, `"#GOOGL"`, `"#LCID"` we are able to fetch Tweets made by users worldwide during our chosen period of study.

The next step is to analyze the sentiment of the tweet. This can then be used to

see if there is any correlation between tweets of about a certain sector, and the stock market changes of the companies that come under that sector.

An NLP model BERTweet (Nguyen et al. [12]), can be used to analyze the sentiment of the tweets. BERTweet, having the same architecture as BERT-base (Devlin et al. [10]), is trained using the RoBERTa pre-training procedure.[11] Experiments show that BERTweet outperforms strong baselines RoBERTa-base and XLM-R-base (Conneau et al., 2020), producing better performance results than the previous state-of-the-art models on three Tweet NLP tasks: Part-of-speech tagging, Named-entity recognition and text classification.

The output of BERTweet is a 3-tuple of confidence values in the format of [negative_sentiment, neutral_sentiment, positive_sentiment]. This output is then converted to a single integer based on whatever sentiment the model has maximum confidence in (-1: negative; 0: neutral; 1: positive).

The category classifier and the sentiment analyzer can be set-up on the Spark Lambda Architecture. The Tweets can then be fed to Spark using Kafka, from the database, in order to simulate real-time analysis.

3.2 The BERT NLP Model

BERT is a multi-layered encoder. In Devlin et al.'s [10] paper, two models were introduced, BERT base and BERT large. The BERT large has double the layers compared to the base model. By layers, we indicate transformer blocks. The BERT encoder expects a sequence of tokens. The below image shows how tokens are processed and converted. [CLS] is a special token inserted at the beginning of the first sentence. [SEP] is inserted at the end of each sentence. We created segment embeddings by adding a segment 'A' or 'B' to distinguish between the sentences. We also add the position of each token in the sequence to get position embeddings.
Our classification model and sentiment analysis model makes use of the BERT base architecture, considering significant overheads associated with employing BERT large, which utilizes more computing resources.

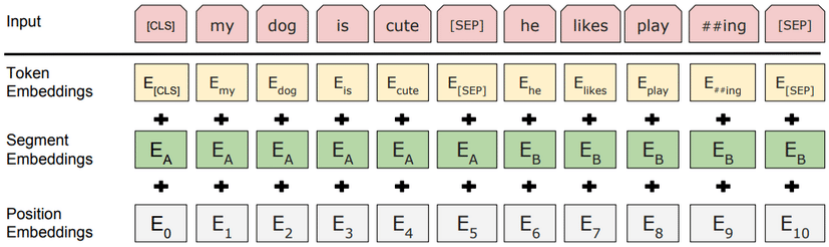

Figure 2: BERT Embeddings

The sum of the above three embeddings is the final input to the BERT Encoder. BERT takes an input sequence, and it keeps traveling up the stack. At each block, it is first passed through a Self Attention layer and then to a feed-forward neural network.

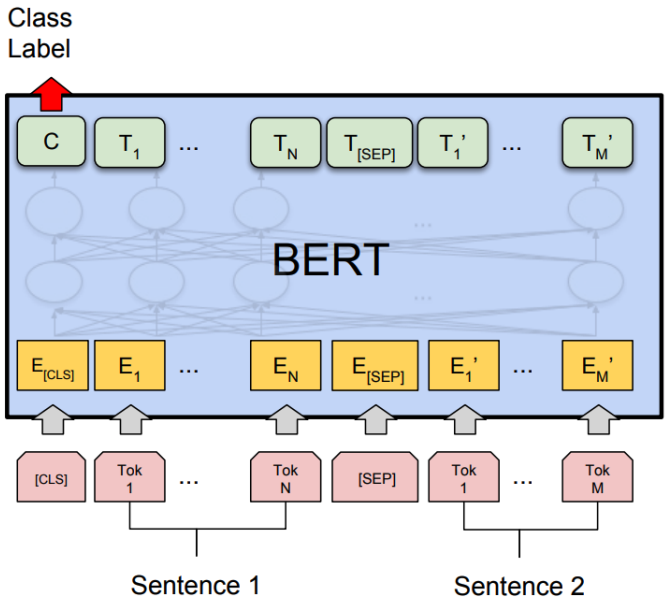

Figure 3: The BERT Sequence Classifier

3.3 Sentiment Analysis

According to Nyugen et al.[12], the BERTweet model presented by them performs much better than existing models like RoBERTa and XLM-R which were tested on the SemEval2017-Task4A dataset. It has an accuracy of 72% and an F1-score of 72.5%.

3.4 Stock Data Analysis

Raw stock data is collected from public stock APIs, and is then pre-processed in a pre-defined format, to be stored in the database. Using Apache Kafka, the stock data can then be streamed to the Spark Architecture, where the data trend-analyzed current mood towards that sector's stock.

This trend analysis is then compared with the trends & public moods seen from the Twitter sentiment analysis, for further statistical studies.

3.5 Statistical Correlation Analysis

The main objective of our research is to identify whether there exists any correlation between Twitter data & stock market data in real-time, and if so, quantify the correlation between them in different sectors.

Correlation analysis needs to be done sector-wise, since social media might have a varying degree of influence over each sector - a logical example would be the cryptocurrency market, which is highly volatile with respect to social media sentiments.

The Pearson correlation coefficient is a measure of linear correlation between two sets of data. It is the ratio between the covariance of two variables and the product of their standard deviations. Thus it is essentially a normalized measurement of the covariance, such that the result always has a value between -1 and 1.

$$r_{xy} = \frac{\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^{n}(x_i - \bar{x})^2}\sqrt{\sum_{i=1}^{n}(y_i - \bar{y})^2}}$$

where:

- n is sample size
- x_i, y_i are the individual sample points indexed with i
- $\bar{x} = \frac{1}{n}\sum_{i=1}^{n} x_i$ (the sample mean); and analogously for $\bar{y}$

Figure 4: The Pearson Correlation Coefficient for a sample

4 Techniques Used

4.1 Tweet Data collection and Pre-Processing

The Twitter API (`GET https://api.twitter.com/2/tweets/search/recent`) along with keywords for each category of analysis was used to collect tweets.

A Python script was written to collect the data, and generate a dataset of tweets labelled according to their category.

4.2 Stock Data collection and Pre-Processing

The Polygon API (`GET https://api.polygon.io/v1/open-close` and `GET https://api.polygon.io/v1/open-close/crypto`) was used to collect stock ticker data. The API accepted the type of ticker data required, and the date to get the data from. A Python script was used to collect historic Stock data for the past two years, and was stored in a dataset categorized according to the sector of the companies queried.

4.3 Storage of Tweet and Stock Data

In order to facilitate real-time inflow of data, Apache Kafka needs to be used to stream the collected data to Apache Spark. To provide a data source for Kafka to stream data continuously, we use an SQLite Database within which we store our collected Tweet and stock information. The SQLite database is populated using a Python script.

4.4 Data Ingestion Using Apache Kafka

The raw collected data is read from the SQLite database and produced into separate topics using a Kafka Producer. The data is split and sent periodically as micro-batches, which simulates a real-time environment and also provides us with a level of uniformity to measure various processing times.

4.5 Data Processing Using Apache Spark

A Spark application is written to process the incoming stream of data from Apache Kafka and perform real-time processing and conduct correlation analysis.

We utilise the PySpark interface for implementing our Spark application. The usage of PySpark allows us to easily integrate our application with the sentiment analysis model, which uses the PyTorch library.

Raw Tweet data is processed with the map-reduce method to aggregate the overall Tweet sentiment of a market category of a Tweet for a given date.

Similarly, raw stock data is ingested from another Kafka topic and processed by Spark in real-time to calculate the percentage change in the stock value for each trading day in our selected time frame. A single aggregate percentage change value is calculated based upon the individual stock price changes, which acts as the representative change for that industrial sector's performance in the stock market for the given day.

The processed intermediate results are then stored back in a SQLite Table to prevent redundant processing, and to use it as the aggregated batch data for subsequent real-time processing events.

4.6 Correlation Analysis Using Apache Spark

Over the span of our time period of study (4 April 2020 to 20 March 2022), the stock market is active for around 500 days. During those days, we compare the Tweet trends and the subsequent change in stock valuations, to understand whether there is any significant correlation between the overall mood reflected by people in social media to stock market volatility.

The aggregated Tweet sentiment score of a sector and its respective percentage change (calculated earlier) is used to perform a correlation study.

We use normalization techniques to scale the data into similar ranges, and use different statistical correlation techniques like the Pearson correlation coefficient and Spearman correlation coefficient.

The data is also plotted onto a time-series graph using Matplotlib to better visualise the observed trends.

5 Current Outcomes

5.1 Data Collection

Tweet data and stock data were collected from API sources and stored in an SQLite Database.

	category	tweetDate	count	tweet
	Search column...	Search column...	Search colur	Search column...
1	Tech	2020-04-07	10	BLOCK TRADE detected in #GOOGL
2	Tech	2020-04-07	10	@PlayAdoptMe I like it #goog
3	Tech	2020-04-07	1	Apple Inc price at close, 2020-04-07, is 271.1. #apple #AAPL
4	Tech	2020-04-07	1	Apple Inc stock rose by 3.288%! Currently priced at 271.1. #apple #AAPL
5	Tech	2020-04-07	1	Looking into the chart of #AAPL makes you wonder if it was just a correction and not a crash. That strong, respected key level perfectly

Figure 5: Sample Tweets stored in the SQLite Database

	category	ticker	stockD...	open	close	pre_market	afterHours	high	low
	Search column...	Search colum	Search column...	Search column	Search column.	Search column...	Search column...	Search colum	Search colum
1	Tech	GOOGL	2020-03-26	1114.72	1162.92	1090.32	1163	1171.48	1092.03
2	Tech	AAPL	2020-03-26	61.63	64.61	60.8	64.7325	64.67	61.59
3	Gaming	EA	2020-03-26	90.47	99.2	88.38	99.23	99.57	90.08
4	Tech	GOOGL	2020-03-27	1127.47	1110.26	1123.91	1109.79	1151.05	1104.0027
5	Tech	AAPL	2020-03-27	63.1875	61.935	63.61	61.75	63.9675	61.7625

Figure 6: Sample Stocks stored in the SQLite Database

5.2 Data Ingestion Using Kafka

Data stored in the SQLite database is streamed live to the Spark context.

Out[20]:

	category	date	count	tweet	score
0	Tech	2020-04-07	10	BLOCK TRADE detected in #GOOGL	0
1	Tech	2020-04-07	10	@PlayAdoptMe I like it #goog	1
2	Tech	2020-04-07	1	Apple Inc price at close, 2020-04-07, is 271.1. #apple #AAPL	0
3	Tech	2020-04-07	1	Apple Inc stock rose by 3.288%! Currently priced at 271.1. #apple #AAPL	0
4	Tech	2020-04-07	1	Looking into the chart of #AAPL makes you wonder if it was just a correction and not a crash. That strong, respected key level perfectly	0

Figure 7: Sample tweet Data streamed to the Spark Context

Out[18]:

	category	ticker	stockDate	open	close	percentage
0	Tech	GOOGL	2020-03-26	1114.7200	1162.9200	4.323
1	Tech	GOOGL	2020-03-26	1114.7200	1162.9200	4.323
2	Tech	AAPL	2020-03-26	61.6300	64.6100	4.835
3	Tech	GOOGL	2020-03-27	1127.4700	1110.2600	-1.526
4	Tech	AAPL	2020-03-27	63.1875	61.9350	-1.982
5	Tech	GOOGL	2020-03-30	1132.6400	1146.3100	1.206

Figure 8: Sample stock Data streamed to the Spark Context

5.3 Sentiment Analysis Model

The BERTweet Sentiment Analysis model is used for Tweet sentiment analysis. It gives an array of confidence values (positive, negative & neutral) of which we take the maximum value and map it to a -1 to 1 scale.

```
[17] sentiment_score("Looks like the EVs are the next big thing")

     1

[18] sentiment_score("Something is horribly wrong with the crypto market!!! Do not invest!")

     -1

[19] sentiment_score("Extensions: someone can add a filter in Twitter to help users get relevant tweets")

     0

[21] sentiment_score("Plugin ideas: you should make a web browser plugin to convert any #fiat to #bitcoin \
     For example: Instead of this costs $50 a night, it should change it into 0.000192 BTC a night")

     0
```

Figure 9: Sentiment Analysis done on sample Tweet data

5.4 Aggregated Intermediate Results

Raw tweets and stock data are processed and aggregated based on the methods discussed above and stored in a relational table in SQLite.

	category	date	score	count
	Search column...	Search column...	Search colum	Search column.
1	Tech	2020-04-13	32	453
2	Tech	2020-04-14	192	585
3	Tech	2020-04-08	-51	416
4	Tech	2020-04-12	6	210
5	Tech	2020-04-07	149	217

Figure 10: Sample aggregated tweet results for a given period

	category	date	agg_percent
	Search column...	Search column...	Search column...
1	Tech	2020-04-16	0.0045
2	Tech	2020-04-17	-0.88175
3	Tech	2020-05-01	-0.37625
4	Tech	2020-05-12	-2.2245
5	Tech	2020-06-22	1.205

Figure 11: Sample aggregated stock results for a given period

5.5 Correlation Analysis

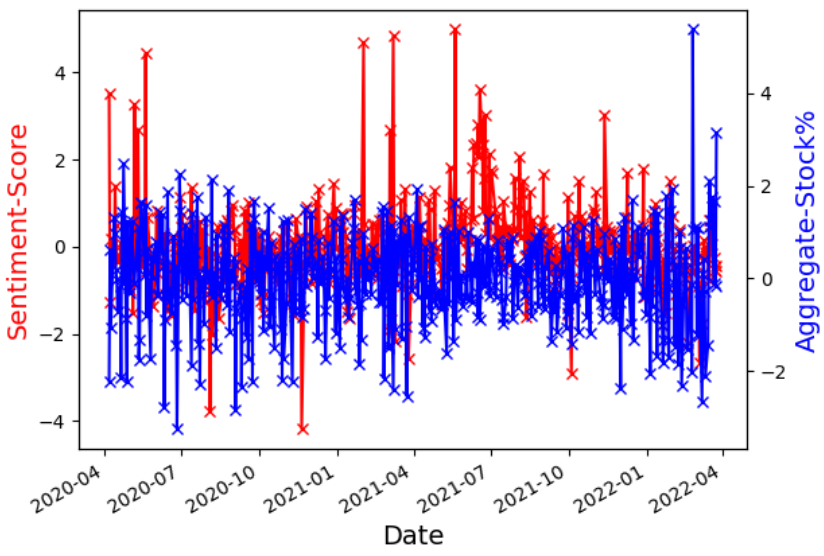

Figure 12: Tech - Time Series plot between Sentiment Score and Stock Price Change

The aggregated intermediate results of the Tweets and stock data is fetched and joined on the basis of date and category for correlation analysis.

The observed correlation between the aggregated percentage value is tabulated and a time series graph is plotted to visualise the aggregated results clearly.

6 Expected Outcomes

The main purpose of the entire analysis, is to unearth a correlation between Tweets and Stock market changes, if any. The expected outcome, therefore, is an accurate correlation analysis between tweet sentiments, and the stock market variations in the related sectors.

7 Further work

1. **Conducting the analysis:** The correlation analysis performed by Spark should be conducted on various other market sectors as well, along with further study on using other Tweet metrics like retweet count of a particular Tweet (weighing the sentiment of a Tweet based on number of retweets)

2. **Visualization of the analysis:** The results should be augmented with the help of visual aids like time-series graphs and charts to understand the aggregated information in a clear manner.

3. **Extrapolation of the results:** If a significant correlation is found between a market sector and its respective Tweets, it is worthwhile to predict the market's performance using the observed social media trends. Techniques like regression could be used for prediction.

References

[1] Chungho Lee, Incheon Paik, *Stock Market Analysis from Twitter and News Based on Streaming Big Data Infrastructure*

[2] Vaanchitha Kalyanaraman, Sarah Kazi, Rohan Tondulkar, Sangeeta Oswal, *Sentiment Analysis on News Articles for Stocks*

[3] Zhihao Peng, *Stocks Analysis and Prediction Using Big Data Analytics*, 2019 International Conference on Intelligent Transportation, Big Data & Smart City (ICITBS)

[4] Gurcan, Fatih & Berigel, Muhammet, *Real-Time Processing of Big Data Streams: Lifecycle, Tools, Tasks, and Challenges*, 2018 2^{nd} International Symposium on Multidisciplinary Studies and Innovative Technologies (ISMSIT)

[5] Aparna Nayak, M. M. Manohara Pai and Radhika M. Pa, *Prediction Models for Indian Stock Market*

[6] Dharmaraja Selvamuthu, Vineet Kumar and Abhishek Mishra, *Indian stock market prediction using artificial neural networks on tick data*

[7] Kanavos, Andreas & Vonitsanos, Gerasimos & Mohasseb, Alaa & Mylonas, Phivos. (2020). *An Entropy-based Evaluation for Sentiment Analysis of Stock Market Prices using Twitter Data*

[8] Tomas Mikolov, Kai Chen, Greg Corrado, Jeffrey Dean, *Efficient Estimation of Word Representations in Vector Space*

[9] Marsland, Stephen, *Machine Learning: An Algorithmic Perspective*, 2014

[10] Jacob Devlin, Ming-Wei Chang, Kenton Lee, Kristina Toutanova, *BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding*

[11] Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, Veselin Stoyanov, *RoBERTa: A Robustly Optimized BERT Pretraining Approach*

[12] Dat Quoc Nguyen, Thanh Vu, Anh Tuan Nguyen, *BERTweet: A pre-trained language model for English Tweets*

[13] Pagolu, Sasank & Reddy, Kamal & Panda, Ganapati & Majhi, Babita. (2016). Sentiment analysis of Twitter data for predicting stock market movements.

[14] Mittal, A. (2011). Stock Prediction Using Twitter Sentiment Analysis.

[15] https://developer.twitter.com/en/products/twitter-api/academic-research