DEPARTMENT OF ELECTRICAL AND COMPUTER ENGINEERING
UNIVERSITY OF VICTORIA, BC, CANADA

Analysing Twitter Feeds to Predict Stock Movements

by
Anoop Venkataramana

A Report Submitted in Partial Fulfilment of the
Requirements for the Degree of
MASTER OF ENGINEERING

Supervisory Committee

Dr. T. Aaron Gulliver, Supervisor

(Department of Electrical and Computer Engineering)

Dr. Kin Fun Li, Departmental Member

(Department of Electrical and Computer Engineering)

ABSTRACT

On average, every second, approximately 6,000 tweets are tweeted on Twitter, which accounts for approximately 500 million tweets a day, and hence, 200 billion tweets per year. In 2010, tweets per day were around 50 million, so in just five years the amount of data has increased by ten times. This exponential increase in data creation and user activity makes Twitter an ideal tool for analysing financial trends. Sentiment analysis is the process of identifying and categorizing opinions expressed in text and determining writer attitudes towards a particular topic. There are few existing systems for analysing tweets to predict sentiments and results may not be accurate due to the random and short nature of tweets. Existing information retrieval techniques rely heavily on linguistic features like part of the speech or trigger words and perform poorly because they cannot understand sentiments. In this project, a segmentation algorithm is used to improve the accuracy and hence provide better sentiment prediction. In the proposed model, a tweet is split into meaningful segments (a word or group of words), while context is preserved and extracted from the segments.

Table of Contents

List of Figures

List of Tables

Acknowledgements

I would like to thank my supervisor Dr. T. Aaron Gulliver for guiding me on this project and for supervising my Masters degree at UVIC. I am very honoured to be a student of Dr. Gulliver who always helped me follow my passion and dreams. He supported me in every step from choosing courses to getting coop and has been a great motivation for me to succeed in Canada. I am very glad to work with him and I would like to give him my sincere gratitude for making this a remarkable journey of my life.

I would also like to thank Dr. Kin Fun Li for serving on my supervisory committee and also Dr. Wu-Sheng Lu for serving as chair of the oral examination committee.

I also would like to thank Mr. Gord Owens and Mr. Ming-Pang for mentoring me at IBM during my internship and Mr. Omar Amin for giving me the opportunity to work on this project at IBM.

I wish to express my deep appreciation to my parents, family and friends for having faith in me and supporting me all the time. Finally, I would like to thank everyone who directly or indirectly helped me in this journey.

Dedication

I dedicate this work to all retail traders who would like to use sentiment indicators in their trading decisions, and to everyone who is working towards making the world a better place to live.

Disclaimer

This report is for educational purposes only. Trading financial instruments is very risky and may not be suitable for all investors. I am not recommending any buying or selling of financial instruments in this report. Investors or traders are fully responsible for any losses incurred due to usage of this system. Please consult your financial advisor before making any trading decisions.

Abbreviations and Acronyms

AI	Artificial Intelligence
API	Application Program Interface
DB	Database
ETL	Extract Transformation and Loading
GPRO	GoPro
HDFS	Hadoop Distributed File System
HQL	Hive Query Language
IR	Information Retrieval
JDBC	Java Database Connection
JSON	JavaScript Object Notation
N/A	Not Applicable
NLP	Natural Language Processing
ODBC	Other Database Connection
RDBMS	Relational Database Management System
SQL	Structured Query Language
UDF	Universal Disk Format

Chapter 1: Introduction

1.1 Related work and motivation

A recent study from Indiana University on making money in the stock market presented a scientific formula that made use of social networking trends [1]. Researchers found that by analyzing the mood of the public on websites such as Twitter, Facebook, Google+, and LinkedIn, they could predict the market three to four days in advance with up to 87% accuracy [2]. The intersection between social media and financial trends is a subject undergoing intense study on Wall Street. The financial district has adopted this method and is using it to predict public sentiment regarding specific topics or products by looking at social networking buzz and chatter [2]. Studies that have looked at popular companies including Starbucks, Nike, and Apple, show a rise in stock prices proportional to the number of followers on social networking sites like Facebook. Usually, stocks that have a higher number of followers have a stronger consumer group and the probability of stock performance correlates with this [3].

Hedge funds are investment funds that invest in stocks and other financial instruments. It is a trillion dollar industry worldwide. Hedge fund managers use sentiment analysis in their trading strategies. For example, Derwent Capital, a hedge fund company based in London, looks at 10% of all available tweets with respect to a company before trading stocks. These tweets are broken into 12 different categories such as happy or sad, depicting the underlying emotions. Derwent Capital has established algorithms which make use of these categories to predict the movement of stocks and if necessary automatically execute trades based on the overall sentiment from Twitter [4].

Retail traders are also using sentiment analysis to plan their trades in the stock market. To cater to

the needs of retail traders, several websites provide information such as tweets and sentiments related to hashtags. A hashtag is a type of label or metadata tag used on social network websites. This way of using sentiment analysis as a technical indicator of stock trading has become popular with retail traders and hedge fund managers [5]. In the financial industry, the term technical indicator is a mathematical calculation used to predict future stock price levels by looking at past patterns [6]. Twitter is a significant source of information related to stocks, hence Twitter is used in this project to predict stock price movements.

1.2 Data mining and Twitter analytics

Data produced on the internet is increasing exponentially with time, leading to an information explosion of unstructured text data. The term unstructured data refers to information that does not have a pre-defined model like a table format. Data mining is a technique for gathering and analyzing data using Artificial Intelligence (AI), machine learning, statistics, and database systems. Data mining is very popular and is implemented in generic programs like those that display advertisements during Google searches and mark spam emails [7]. Mining of Twitter is done to explore trending topics and find what people are talking about on a particular subject or product.

Twitter is a social networking site where users post 140 character long messages called tweets which provide personal opinions on topics or subjects. Users use hashtags when posting these messages. There are 310 million monthly active Twitter users worldwide. Every second, an average of around 6,000 tweets are tweeted, which is approximately 21 million per hour, 500 million per day, and over 200 billion in a year [8]. Figure 1 demonstrates the rapid increase in tweets posted by users. This data is large and unstructured in nature [7-9]. It can be analyzed to understand consumer sentiments on a product or as in this project, the view of Twitter users on stocks.

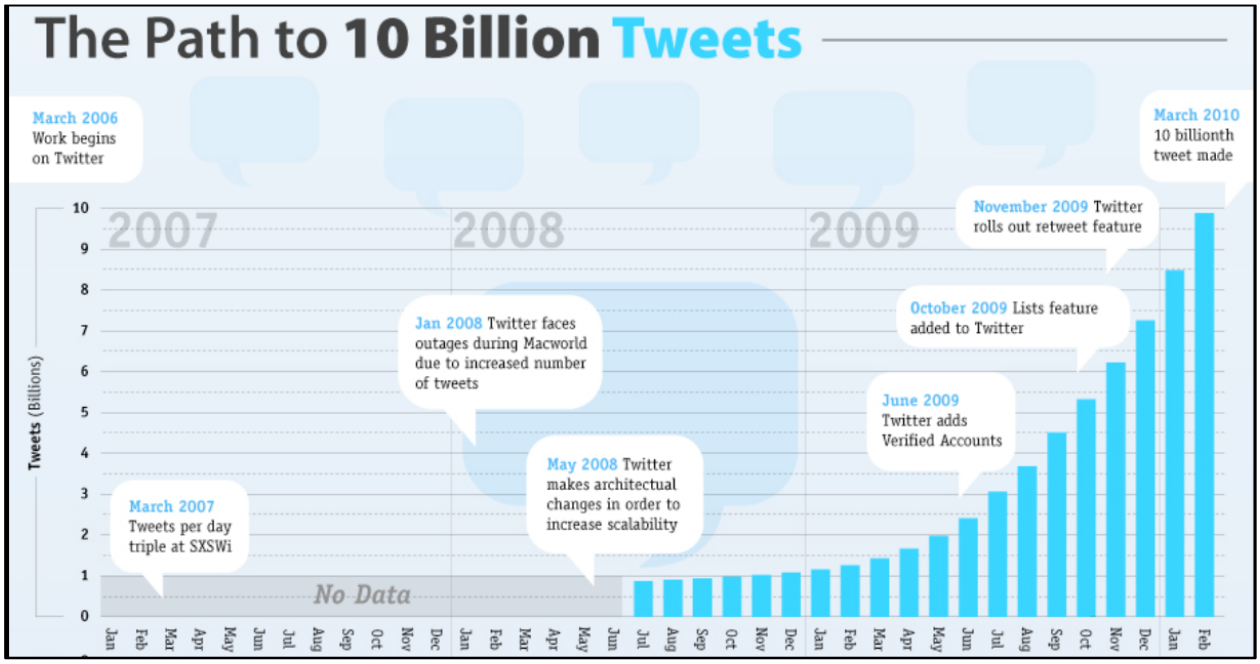

Figure 1: Exponential growth in tweets with time [9].

1.3 Watson and Artificial Intelligence

IBM Watson is one of the leading AI software programs in this new age of cognitive computing. Cognitive computing is the simulation of human thought processes using a mathematical model [10]. Watson is a new and different kind of computing system which can understand sentiments by analyzing human language. In this project, Watson is used for sentiment analysis of data available from Twitter servers [11].

Human beings generally follow four steps in making decisions.

1. **Observation** of phenomena and bodies of evidence.
2. **Interpretation** of existing information and generating hypotheses.
3. **Evaluation** or testing of hypotheses.
4. **Decide** by choosing the best option.

Watson uses these same steps to make decisions. Watson can understand unstructured data such as blogs, poetry, literature, and tweets, which constitutes 80% of internet data today, to draw conclusions similar to those made by human beings. In order to process the complex high level languages used by human beings, Watson employs Natural Language Processing (NLP) techniques to understand grammar context and culture. NLP is focused on developing efficient algorithms to process human language and to make information accessible to computer applications. Watson breaks down each sentence grammatically, relationally, and structurally, and tries to understand the logic behind the sentence to draw appropriate inferences [12, 13].

1.4 Hive and HQL

The term Big Data is used for collections of large data sets that are complex in nature. These large datasets cannot be processed by traditional data processing applications causing the Big Data problem [14]. As the amount of data increases, it becomes expensive to use traditional data management systems like Oracle or DB2. Hadoop is a Java based programming framework that supports the processing of Big Data in a distributed computing environment. It provides a cost effective, reliable, tolerant, and scalable solution to the Big Data problem. The Hadoop architecture is shown in Figure 2. This figure also shows the big analytics architecture (data analytics for Big Data) on top of Hadoop [15]. The left part of Figure 2 shows that Hadoop can process both unstructured and structured data. Hadoop has different tools to operate on data and this processed data is uploaded to databases like NoSQL as shown in Figure 2. In the analytics part, end users can use different visualization tools to display the processed data.

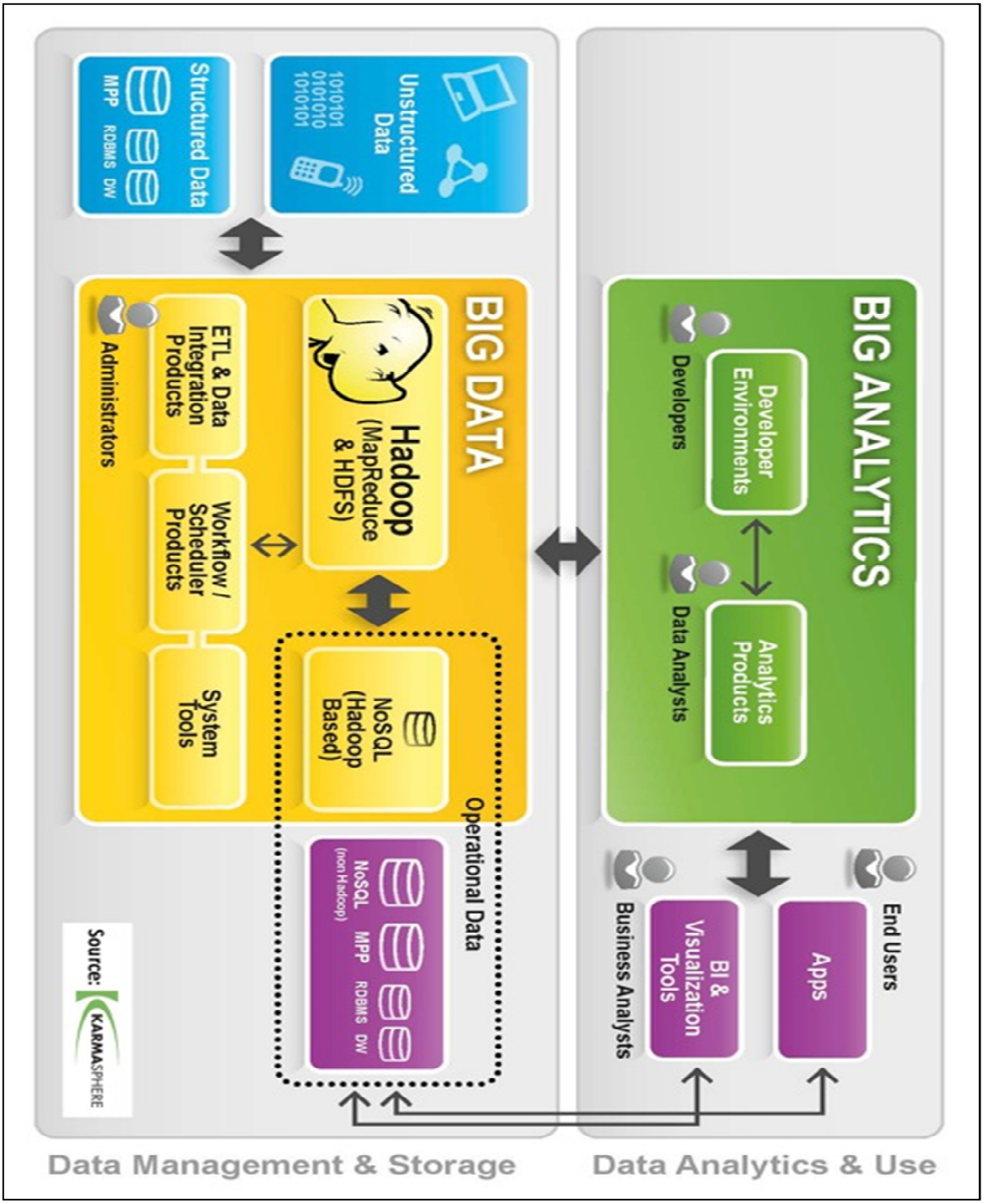

Figure 2: The Hadoop architecture [16].

A data warehouse is a system for the storage of data accumulated from a wide range of sources. Hive is based on the Hadoop framework and is a data warehouse infrastructure tool used to

process structured data [17]. Hive makes querying and analyzing easy. It uses a structured query language called HiveQL for data manipulation, querying, aggregation, and analysis of data, thereby avoiding the need for complex Java access programs. HiveQL is built on top of Hadoop, so it is able to leverage the functionalities of Hadoop. It is a platform used to develop SQL-type scripts on the Hadoop Distributed File System (HDFS) to do MapReduce operations. MapReduce is an operation that divides a task into various subtasks and distributes it across servers to achieve parallel computing. HDFS is a fault-tolerant file system which allows Hadoop to scale data across physical servers [18].

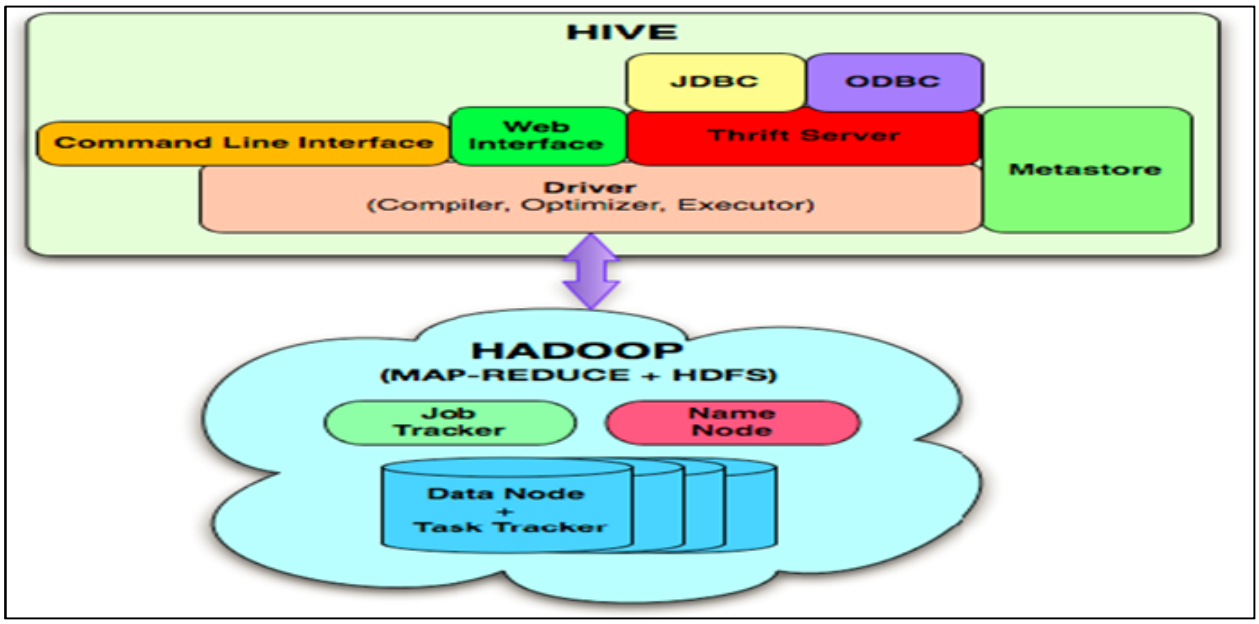

Figure 3: The Hive architecture [19].

Figure 3 shows how Hive interfaces with HDFS. Hive also provides access to application connectivity tools like Java Database Connectivity (JDBC) and Open Database Connectivity (ODBC), which helps in using analytic software. The primary difference between Hive and a Relational Database Management System (RDBMS) is the way the schema is created on data. A schema is created on data to convert unstructured data into structural data. In a traditional

database, schema creation happens at data load time, i.e. schema on write. Hive creates a schema when the query is issued to read data, i.e. schema on read [19].

1.5 Proposed model

In this project, tweets were downloaded from the Twitter data server into a flat file. A flat file is a simple text document. These tweets were then stored in the HDFS and a schema was created on this data by Hive. A single tweet was taken from the database and broken into different parts in order to assign a sentiment. In BigInsights the sentiment of a word is assigned by a lookup from a database table, called a dictionary, where sentiments are stored based on whether a given word is positive, negative, or neutral. Watson breaks a tweet into segments, where a segment is a group of words like a phase or idiom, and uses in-built machine learning techniques to understand the tweet sentiment.

Once the sentiment of a word is known, the sentiment of an entire tweet is calculated. Then, a bullish-bearish sentiment is assigned based on the aggregate sentiment score for a tweet. An example is given in Figure 4 for a stock called GPRO. This Figure shows tweets in the first column and the sentiments assigned in the second column.

Tweet	Sentiment assigned
sold #gpro calls for .80 cost was .40 20 contracts	Bearish
Worst performance industries for july 19, 2016 worst performing stock #CX #GPRO	Very bearish (strong sell call)
loving #GPRO over $60!!	Very bullish (strong buy call)
Introduce New #GPRO Commercial #Grapics Series	Neutral (no trade)
at a day high, lets gooo #GPRO	Bullish
#GPRO has superior tech and room to grow	Very bullish (strong buy call)
#GPRO outlook looks good for this month	Bullish

Figure 4: Example of sentiments assigned for the GPRO stock.

1.6 Report structure

In this project, a new algorithm is presented to do sentiment analysis for stocks. This algorithm can predict the market direction along with the magnitude of the rise or fall in price. This algorithm is first implemented in BigInsights (an IBM Big Data tool) with Hql and Java and then later compared with results using Watson to implement the same algorithm. In BigInsights, Hql implements machine learning techniques, whereas Watson has in-built machine learning capabilities which can be used for more accurate results [20].

The remainder of this report is organized as follows. In Chapter 2, the algorithm design is discussed in five steps. Each part of the main flowchart is explained separately. In Chapter 3, the steps of the algorithm are implemented using HQL and simulation results using BigInsights and Watson are presented. In Chapter 4, a summary is given and future work is discussed along with applications for using Big Data analytics and sentiment analysis.

Chapter 2: The Proposed Algorithm

In the financial industry, different algorithms are implemented to predict market direction using sentiment analysis but there are few algorithms to predict the direction and magnitude of the movement of a stock price. In this project, a new algorithm is designed to predict the magnitude along with the direction of stock price movement. Every stock analysed will have a trading call like buy or sell based on the predicted direction of the stock price movement. Bearish or bullish strength are calculated to predict the magnitude of this movement. A higher strength indicates that sentiment towards the stock is stronger and can give higher returns. This algorithm is divided into five steps.

(1) Downloading and cleaning data.

(2) Breaking a tweet into words or segments.

(3) Lookup action for assigning sentiments to words.

(4) Calculating the aggregate score.

(5) Assigning sentiment indicators to tweets and evaluating the sentiment strength for stocks.

This algorithm is implemented using BigInsights and Watson separately. In step 2, a tweet is broken into words in BigInsights whereas it is broken into words or group of words called segments in Watson. In step 3, a dictionary is used to understand word sentiment in BigInsights whereas machine learning is used in Watson for understanding sentiments. These steps are shown in Figure 5. Each of these steps is explained in detail in this chapter.

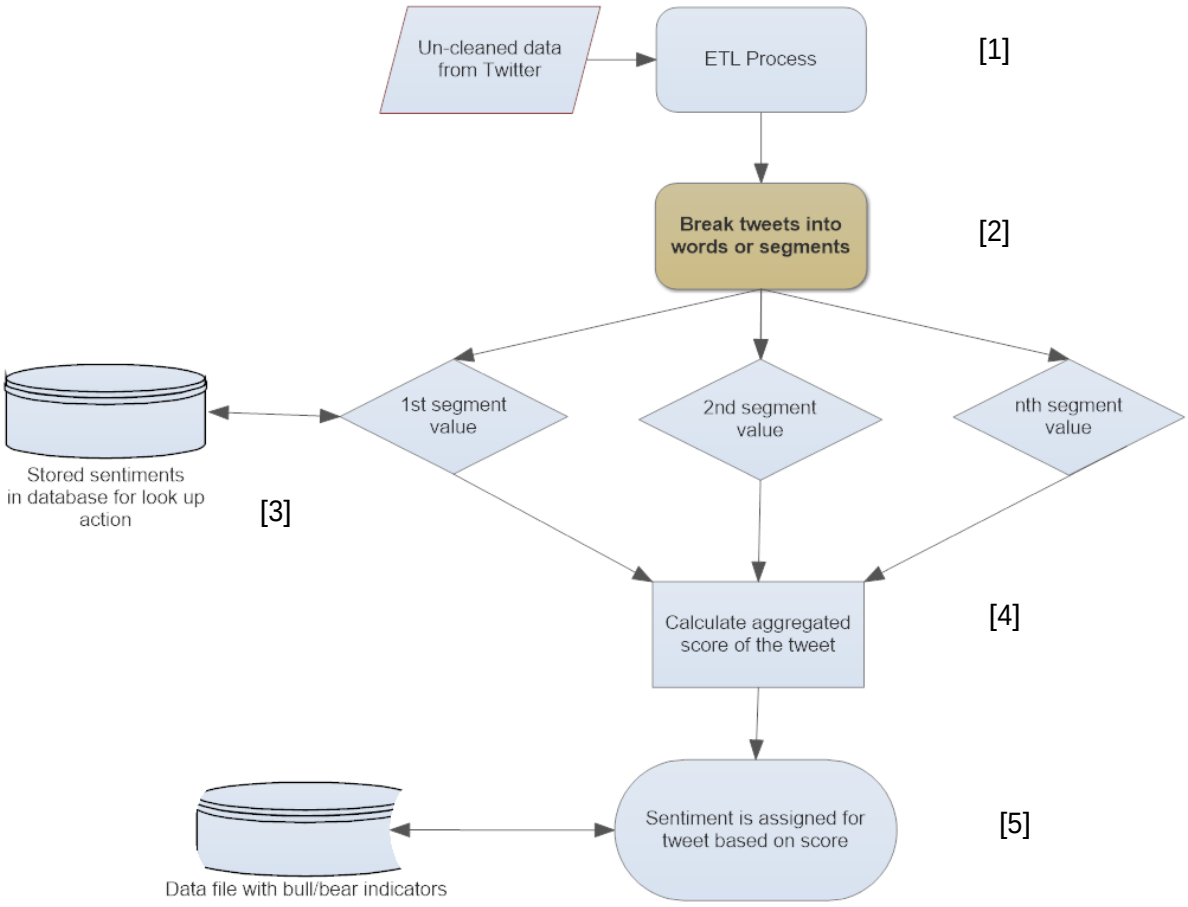

Figure 5: The proposed algorithm flowchart.

STEP 1: Downloading and cleaning data: The Twitter Application Program Interface (API) is used to download tweets in JSON format [21, 22]. This data is written into a flat file as unstructured data. The data are cleaned by the Extract Transformation and Loading (ETL) process and loaded into the database as structured data. Figure 6 shows the flowchart for cleaning data [22].

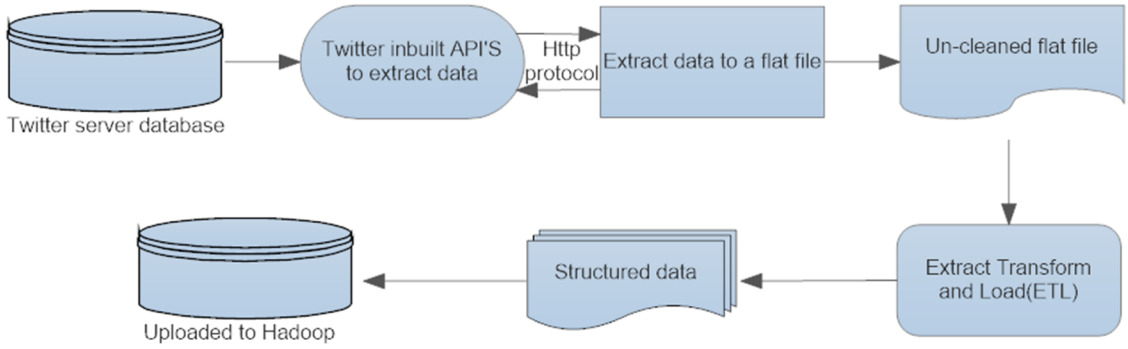

Figure 6: Cleaning data flowchart.

Data are collected as a flat file from the Twitter server using built-in APIs. All data are downloaded with the tweets related to a particular hashtag. This data may include attributes such as re-tweeted status, tweet creator name, tweet created time, and tweet created country. Schema, a skeleton structure that represents the logical view of the entire database, is created on this data, which separates the attributes by dividing them into different columns.

STEP 2: Breaking a tweet into words or segments: A tweet is broken into words in BigInsights whereas in Watson a tweet is broken into segments. A flowchart of breaking a tweet into words or segments is shown in Figure 7. A schema is created for unstructured data by Hive to give a table structure for the uploaded data in HDFS. The unnecessary data are then removed and a unique tweet identification (ID) is created for each tweet. Unnecessary data are attributes in downloaded data like re-tweeted status, tweet creator name, and tweet created time which is not needed in the process. Finally the tweets are broken into words or segments.

Figure 7: Flowchart for breaking tweets.

Consider a random tweet to demonstrate this step. The tweet below is assigned ID 101 and all other column attributes like re-tweeted status, tweet creator name, and tweet created time are removed. This is followed by breaking the tweet into words.

Tweet ID	**Tweet**
101	"This stock is going down … exit this now, it's a falling knife."

STEP 3: Lookup action for assigning sentiments to words: A separate database table named the dictionary was created at the beginning of the process by downloading English words from various websites [23-25]. These words were then inserted into a column of the table and assigned sentiments +1 for positive words, 0 for neutral words and -1 for negative words. This table is used for making decisions regarding tweet sentiment. In this step, a tweet is broken into its constituent words, and each word is assigned -1 for negative, +1 for positive and 0 for neutral by looking it up in the dictionary. For example, in tweet 101, "down", "exit", "falling" and "knife" are assigned -1 in the dictionary. Hence, -1 is assigned to these negative words in the tweet.

STEP 4: Calculating the aggregate score: All tweets are assigned an aggregated score by summing up the individual word scores assigned in the previous step. In tweet 101, the four negative words result in an aggregate score of -4 for this tweet.

STEP 5: Assigning sentiment indicators to tweets and evaluating the sentiment strength for stocks: The sentiment strength is a numerical value of bearish strength or bullish strength, and is

used to determine the magnitude of the stock price movement whereas trading calls are generated as buy or sell based on the predicted direction of stock price movement. If a stock has a higher number of positive sentiment tweets, then the bullish strength is calculated to predict stock return and direction as prices are predicted to go higher hence a buy trading call. If a stock has a higher number of negative sentiment tweets, then the bearish strength is calculated to predict stock return and direction as prices are predicted to go lower, hence a sell trading call.

In this step, the aggregate score of a tweet is considered and the sentiment of the tweet is assigned. The assigned sentiment indicators are very bullish, bullish, neutral, bearish, and very bearish and are assigned to each tweet based on the values given in Table 1.

Table 1: The aggregate score and sentiment

Aggregate score	**Sentiment**
Equal to or more than 2	Very bullish
Equal to 1	Bullish
Equal to 0	Neutral
Equal to -1	Bearish
Equal to or less than -2	Very bearish

In the example of tweet 101, it will be assigned a very bearish sentiment because the score is less than -2. In a similar way all tweets are classified in one of the sentiment categories based on their aggregate scores. The number of tweets in each sentiment category is counted, i.e the total number of very bullish tweets, bullish tweets, neutral tweets, bearish tweets and very bearish tweets. These numbers are used in the following equations to calculate bullish and bearish strength.

If the number of bullish sentiment tweets is greater than the number of bearish sentiment tweets then the total bullishness strength is calculated as

$$Bullish\ strength = \frac{(2 \times number\ of\ very\ bullish\ tweets + number\ of\ bullish\ tweets)}{\begin{pmatrix} 2 \times number\ of\ very\ bullish\ tweets + number\ of\ bullish\ tweets + number\ of\ neutral\ tweets \\ +number\ of\ bearish\ tweets + 2 \times number\ of\ very\ bearish\ tweets \end{pmatrix}} \quad (1)$$

If the number bearish sentiment tweets is greater than the number of bullish sentiment tweets then the total bearishness strength is calculated as

$$Bearish\ strength = \frac{(2 \times number\ of\ very\ bearish\ tweets + number\ of\ bearish\ tweets)}{\begin{pmatrix} 2 \times number\ of\ very\ bullish\ tweets + number\ of\ bullish\ tweets + number\ of\ neutral\ tweets \\ +number\ of\ bearish\ tweets + 2 \times number\ of\ very\ bearish\ tweets \end{pmatrix}} \quad (2)$$

These equations were constructed through a trial-and-error approach and refined over several iterations. If the sentiment strength value is less than 50%, then it indicates no trading calls. If the bullish strength is close to 100%, then it is a strong buy call. If the bearish strength is close to 100%, then it is a strong sell call. Sentiment strength helps in determining the magnitude of sentiment which helps in determining the rise or fall in stock price. For instance an 85% bullish strength shows a higher return probability than a 65% bullish strength.

Chapter 3: Implementation and Simulation Results

3.1 Algorithm implementation on BigInsights

BigInsights is a Big Data management tool used at the enterprise level by IBM on their cloud service called Bluemix. The architecture of BigInsights is shown in Figure 8. Tweets or feeds from the Twitter website are processed using Watson APIs or Java code (given in Appendix A.1) to create reports or analytics. BigInsights has Hive where the HQL code given in Appendix B is implemented to create a module on Bluemix [14].

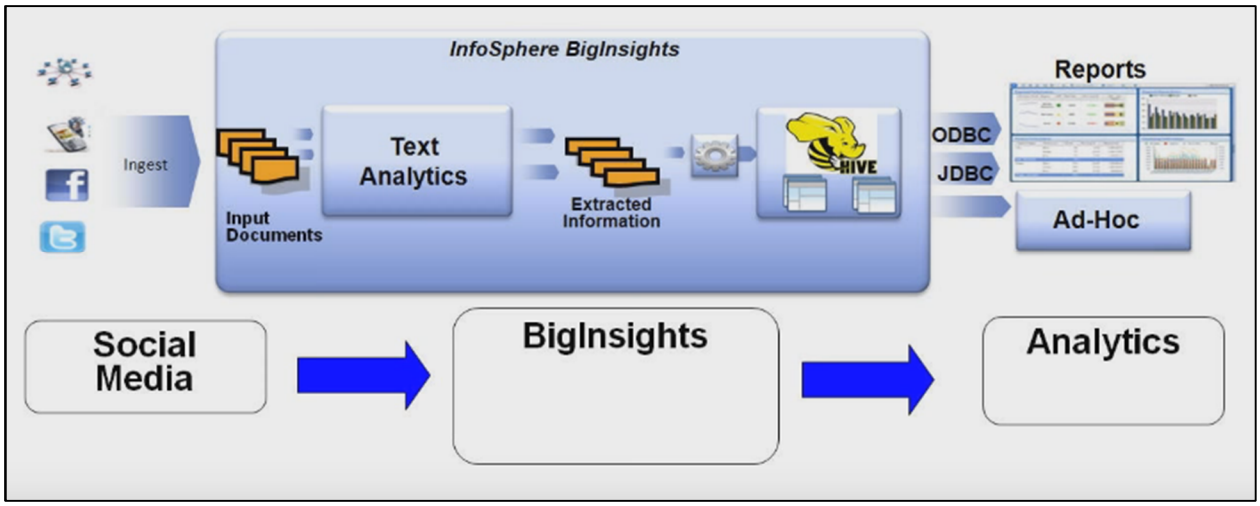

Figure 8: The BigInsights architecture [14].

3.1.1 Implementation on Hive

STEP 1: Downloading and cleaning data: In this project, tweets were downloaded from Twitter using the in-built API called Search API. The unstructured data was created in a flat file. The code implementation converts the unstructured data into structured data. The code used to create the structured data for the flat file in the Hadoop cluster is provided in Appendix B.1.

STEP 2: Breaking a tweet into words or segments: Hive has an in-built function called explode which can be used to break a tweet into its constituent words. A sentiment score is calculated for

each word using dictionary table lookup [26]. The code to implement this step is given in Appendix B.2.

STEP 3: Lookup action for assigning sentiments to words: To understand the sentiments of words, lookup is done from the dictionary. All English words are present in this dictionary with word type, word size, and word sentiment as -1, +1 or 0, respectively. The code to implement this step is given in Appendix B.3.

STEP 4: Calculating the aggregate score: All words in each tweet have -1, 0 or +1 assigned, and these are added to calculate the aggregate score for a tweet. Based on the aggregate score of a tweet, sentiment indicators are assigned according to Table 1. The code to implement this step is given in Appendix B.4.

STEP 5: Assigning sentiment indicators to tweets and evaluating the sentiment strength for stock: All tweets and their IDs and aggregate scores, are written in a Record Columnar File (RCFile). From this file the number of tweets corresponding to very bullish, bullish, neutral, bearish, and very bearish sentiments are counted and a graph is drawn similar to Figure 9 using the Java code in Appendix A1. Equation (1) is used if the bullish sentiment is higher than the bearish sentiment, otherwise (2) is used. The code to implement this step is given in Appendix B.5.

3.1.2 BigInsights simulation results

Tweets were collected for Microsoft stock from July 28, 2015 to August 8, 2015 with #Microsoft, #Windows, and #Windows10 hashtags. This data was processed according to the algorithm described in the previous sections and the resulting sentiments were stored in a table. A graph of this data is shown in Figure 9 which shows the number of very bullish, bullish, neutral, bearish and very bearish tweets which are used in (2) to find the bearish strength.

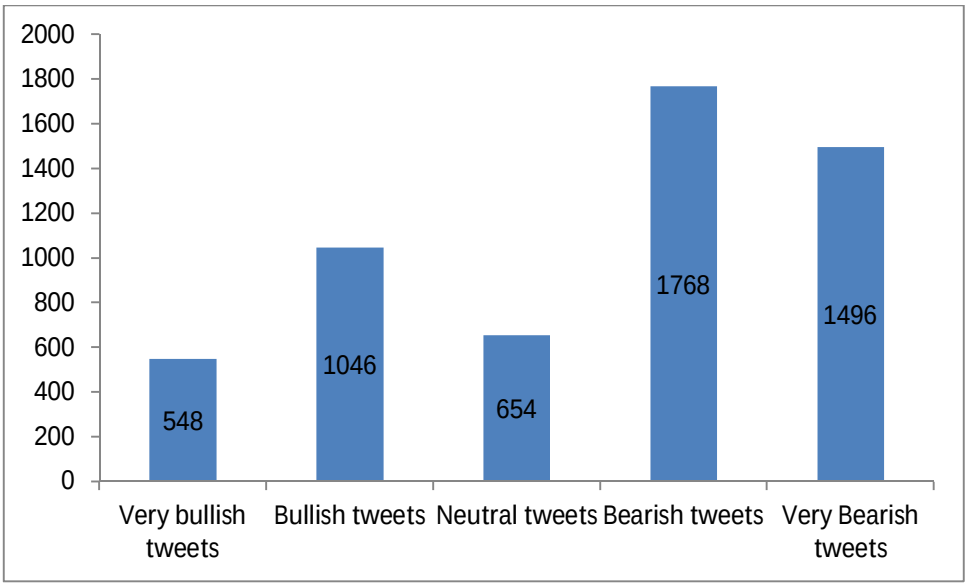

Figure 9: Tweet sentiment count for Microsoft stock.

Substituting in (2) gives

Bearish strength = (2×1496+1768) ÷ 7556 =4760 ÷ 7556= 63%

This indicates a sell call with 63% bearish sentiment strength. Figure 14 shows that the price of Microsoft stock fell from $46.93 to $39.68 after August 8, 2015 corresponding to a fall of 15% in the stock price.

In a similar manner, five more stocks were analyzed using this method and the results are shown in Figure 10. Apple, Bank of America (BofA), Microsoft, Tesla, and Walmart stocks were analyzed from April 1, 2015 to April 14, 2015, IBM was analyzed from February 1, 2015 to February 14, 2015, and stock returns were noted until the next major correction. The term correction in the stock market means a change in direction of the stock price movement and the term stock return is the percentage gain of invested money. Here results match all the buy calls generated by the proposed algorithm except for Walmart where the sentiment indicator gave a bullish indication but

stock returns were negative. Five out of six stocks give positive returns for an 83.3% accuracy. Further, the magnitude of the sentiment is proportional to the stock returns as shown in Figure 10. The stronger the sentiment strength, the higher the stock returns demonstrating a positive correlation.

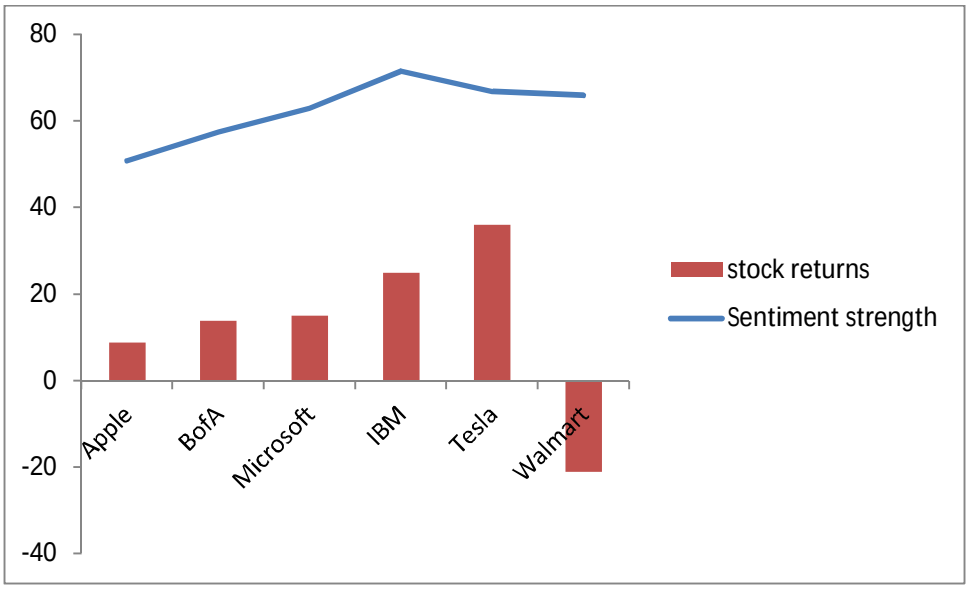

Figure 10: Stock returns and sentiment strength for six stocks using BigInsights.

3.2 Algorithm implementation on Watson

Watson was used for analysing the sentiments behind each word or segments such as phrases or idioms [23]. Consider the tweet “if you are born with a silver spoon in your mouth then buy this stock” which has an idiom in it. Using BigInsights, the aggregate score is calculated as +2 because of the two positive words silver and buy. As Watson is trained to understand and interpret human language, it breaks it into a segment identifying the idiom and hence it understands that the tweet is bearish, as it means that if someone is rich enough to lose money then this stock can be brought, which shows that the risk is high in buying this stock. This leads to a more accurate sentiment strength and hence better stock returns can be predicted. In this step, sentiments were calculated for the Microsoft stock during the Windows 10 release on July 29, 2015. Hashtags used for this were #Microsoft, #Windows, and #Windows10 as shown in Figure 11.

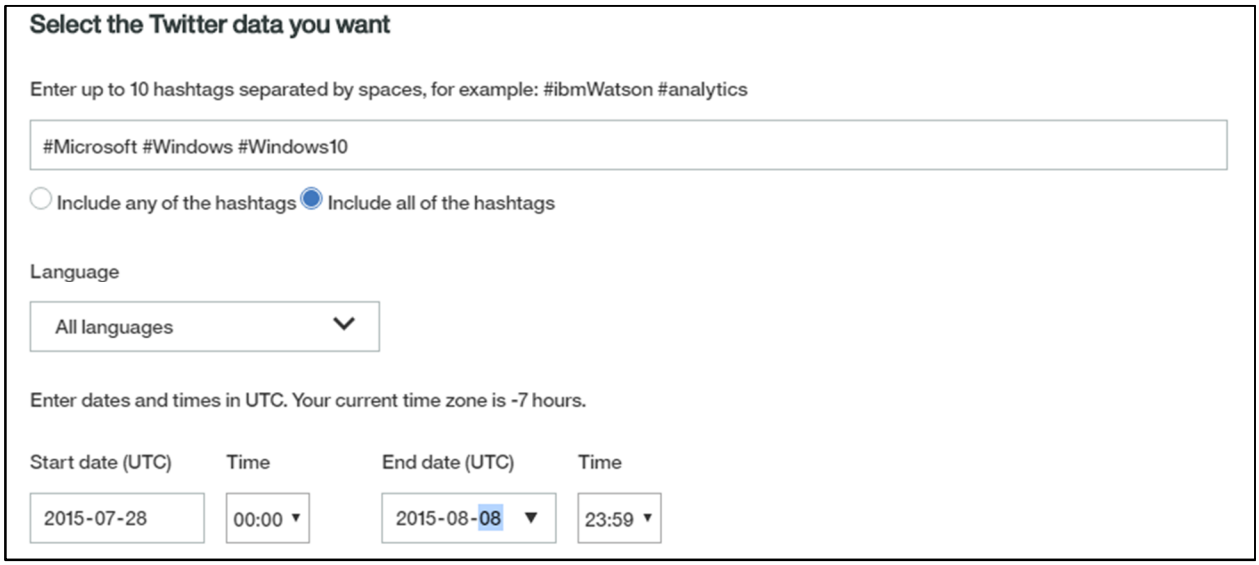

Figure 11: Microsoft stock hashtags used for Windows10 release.

3.2.1 Watson simulation results

Negative and positive sentiments for Microsoft stock were analyzed separately. The negative sentiment for Microsoft stock was calculated first. Figure 12 shows the results of the analysis of Microsoft stock during the Windows 10 release. The number of very bearish tweets with negative aggregate score of -2 is 674,024, and the number of bearish tweets with aggregate score of -1 is 1,326,201. Figure 12 is from the Watson analytics dashboard. The left side shows the spiral graph with various Twitter attributes like the re-tweet count in green dots. Sentiments are strong if these dots are close to the center. The right side shows the top negative sentiment words by count and negative words found such as nerd, stupid, and bogus.

Figure 13 is from the Watson analytics dashboard. The left side of this figure shows the spiral graph with various Twitter attributes such as the re-tweet count in green dots. Sentiments are strong if these dots are close to the center. On the right side, it shows the positive sentiment words by count and positive words found such as best, excited and thanks. Positive sentiment is now calculated for Microsoft stock. Figure 13 shows the results for Microsoft stock during the Windows 10 release.

The number of very bullish tweets with aggregate score of +2 is 195,057, number of neutral tweets is 45,712, and the number of bullish tweets with aggregate score of +1 is 417,181 tweets.

Substituting in (2) gives

Bearish strength = (2×674024+1326201) ÷ 3527256 =2.674M ÷ 3.527M= 76%

Figure 14 shows Microsoft stock performance after the Windows 10 release. The horizontal axis shows the dates and the vertical axis shows the stock price. The tweets analyzed until August 8,

2015 indicate a strong sell call. Figure 14 shows that the price of Microsoft stock fell from $46.93 to $39.68 after August 8, 2015 which is a fall of 15.51%.

Figure 12: Negative sentiment towards Microsoft stock after Windows 10 release.

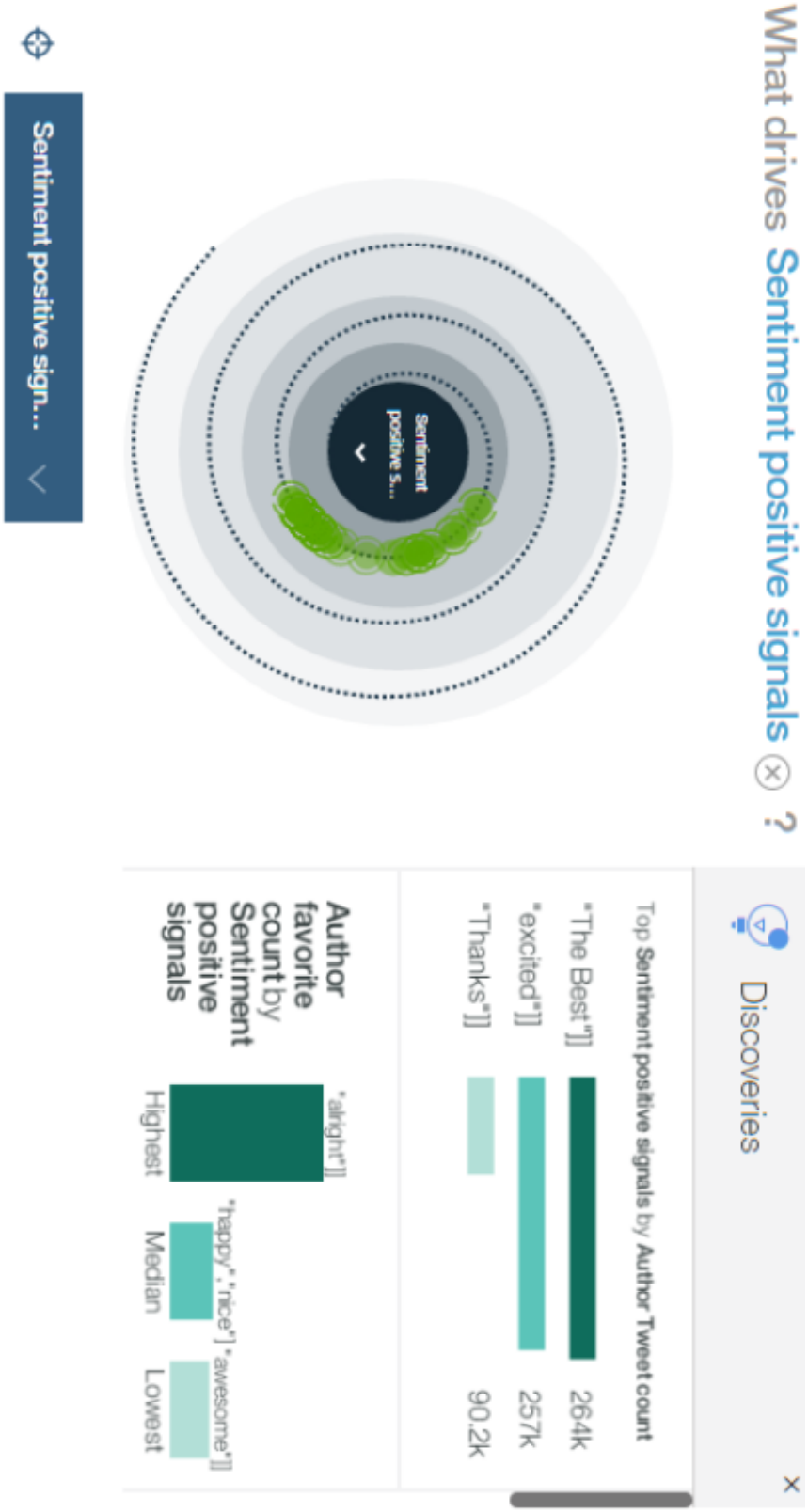

Figure 13: Positive sentiment towards Microsoft stock after Windows 10 release.

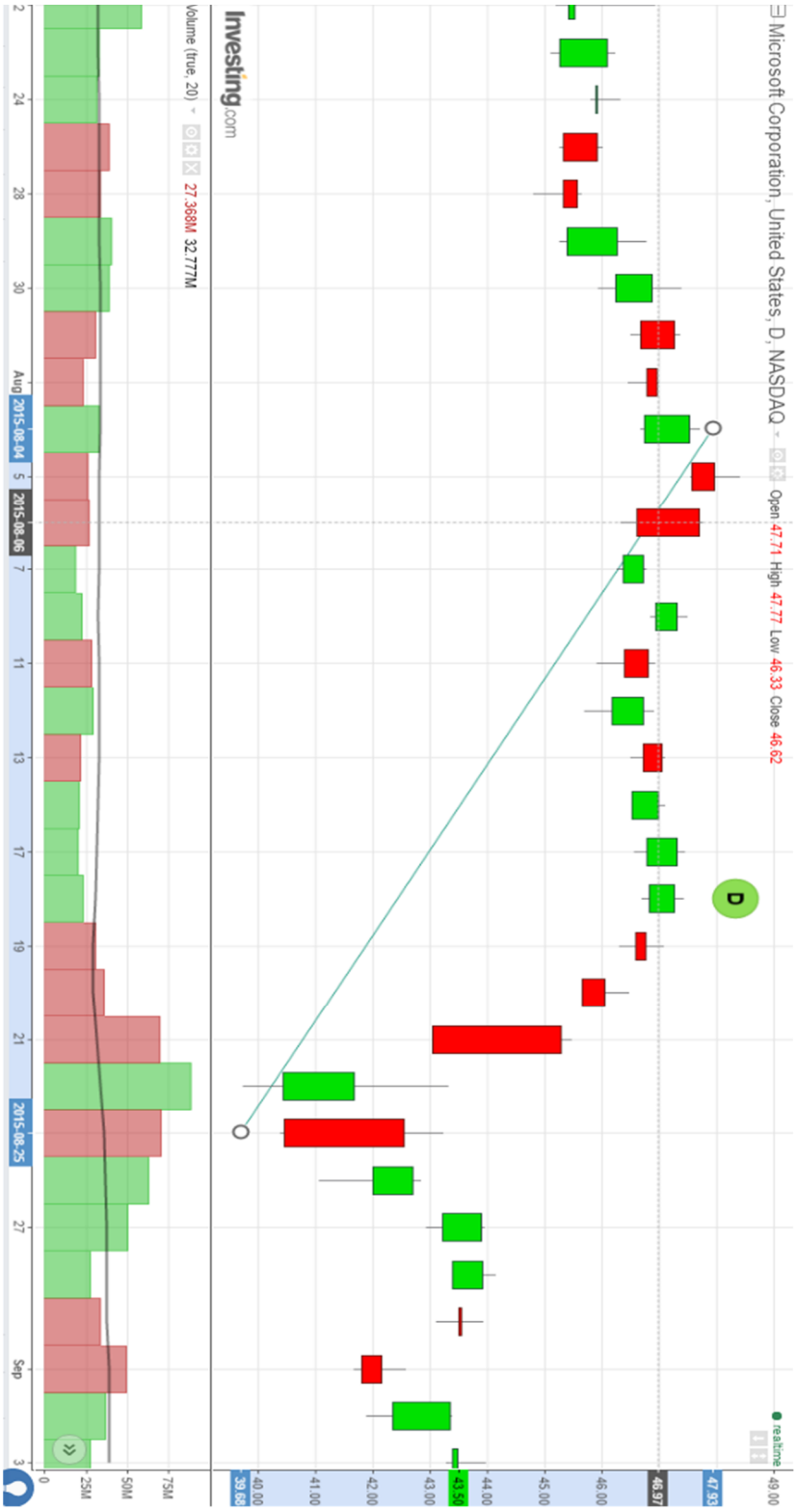

Figure 14: Microsoft stock falling after Windows 10 release [27].

Similarly, five more stocks were analyzed using Watson and the results are shown in Figure 15. Apple, Bank of America, Microsoft, Tesla, and Walmart stock were analyzed from April 1, 2015 to April 14, 2015, IBM was analyzed from February 1, 2015 to February 14, 2015, and stock returns were noted until the next major correction. These results match all the buy calls generated by the sentiment strength except the Walmart indicator which gave a bullish indication but the stock returns were negative. Five out of six stocks give positive returns for an 83.3% accuracy. Figure 15 also shows that the sentiment strength is proportional to the stock returns.

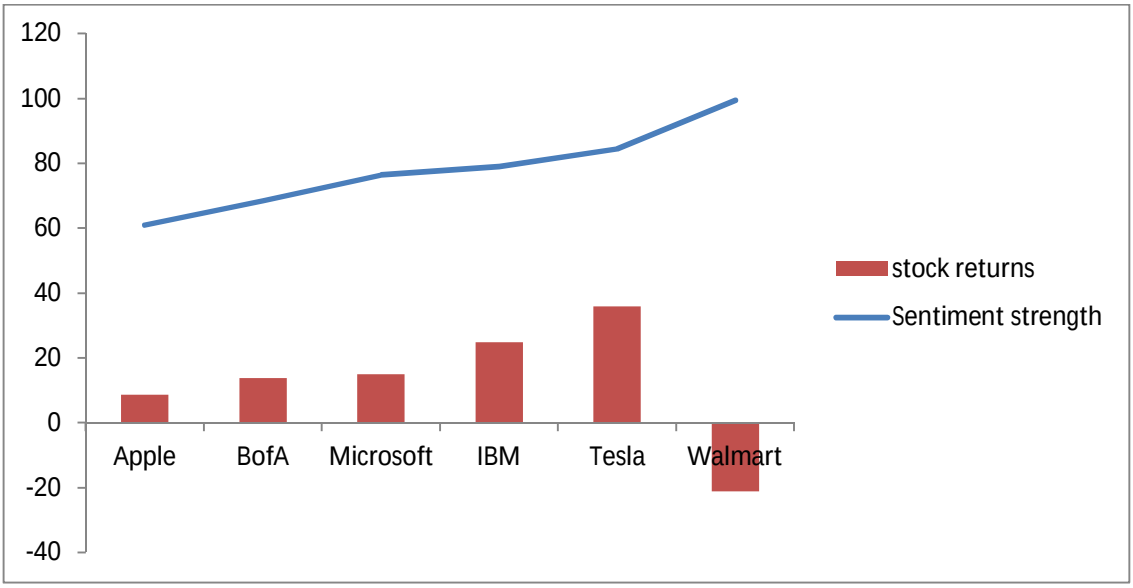

Figure 15: Stock returns and sentiment strength for six stocks using Watson.

Figures 10 and 15 show the stock returns and sentiment strength using BigInsights and Watson, respectively. Both figures show positive results for stock returns and a rise in stock price for all stocks except Walmart. However, the Watson sentiment strength rise demonstrates better sentiment learning due to the better machine learning algorithm in Watson. From these results, it can be concluded that sentiment analysis can be used as a technical indicator for making trading

calls. Five out of the six stocks gave correct responses, and the magnitude or rise in the stock price can be predicted using sentiment strength.

Chapter 4: Summary and Applications

4.1 Summary and future work

Sentiment analysis is a technique which can be used to predict the attitude of the public towards a topic or product. Through AI algorithms, sentiments associated with tweets can be obtained automatically. This is a new and innovative method for predicting stock price movements. Both retail traders and trading institutions are currently using sentiment analysis with profitable results.

Techniques for sentiment analysis for predicting stock price movement and prices were presented in this report. This involved the use of IBM products like Watson and BigInsights to analyse public sentiment of stocks on Twitter. The accuracy of the results was shown by simulation results.

Future work lies in developing better algorithms to accommodate fundamental aspects such as economic events, and events related to the stock industry to predict market price trends more accurately. Social networking sites such as Facebook and Google+ can also be used. Similarly, the proposed approach can be extended to other applications, some of which are given below.

4.2 Applications

Application 1: Forecast market demand for a product

Most product-based companies use sentiment analysis to gauge the popularity of their product. This analysis helps to control the scale of production for companies like Tesla, a car manufacturing company. Using sentiment analysis allows for accurate demand prediction, which will help companies plan for the expansion of factories and investment in resources to meet the requirements of consumers. It can be used to analyse demand towards a demo product. Sentiments

can also be differentiated and categorized by region, sex, age, and time to predict the demand for a product [28].

Application 2: In pharmaceutical industry

Sentiment analysis or opinion mining aims to identify and extract the opinions, moods, and attitudes of individuals and communities regarding a particular product. With the role of social media expanding rapidly, more data and information are shared among ordinary citizens having access to the internet. Over a third of consumers are using social media to make important healthcare decisions. Patients are increasingly turning to popular social media sites like Twitter, Facebook, YouTube, blogs, and forums to obtain and share information related to their health and the drugs they use by leaving feedback about the products. Sentiment analysis on social media now plays a key role in sharing information like drug comparisons using user comments and opinions as shown in Figure 15. Similar kinds of drug side effects are categorised and the sentiment are shown.

The broader conversation about diseases, medications, and treatments can contain valuable information about products that have a strong influence on patient lives. Identifying the opinions of healthcare professionals and patients is a complicated process, particularly because of the spam surrounding pharmaceuticals. Determining the safety of drugs and cosmetics with the help of the experience of consumers and retailers is important [29].

Sentiment mining research is important not only for commercial establishments, but also for people to confirm that they are using the correct drugs. Opinion mining holds great potential for the healthcare industry to identify inefficiencies and best practices that improve care. As shown in Figure 16, social media analytics can be applied at various stages of a drug life cycle, from the drug discovery stage to the release of the finished product [30].

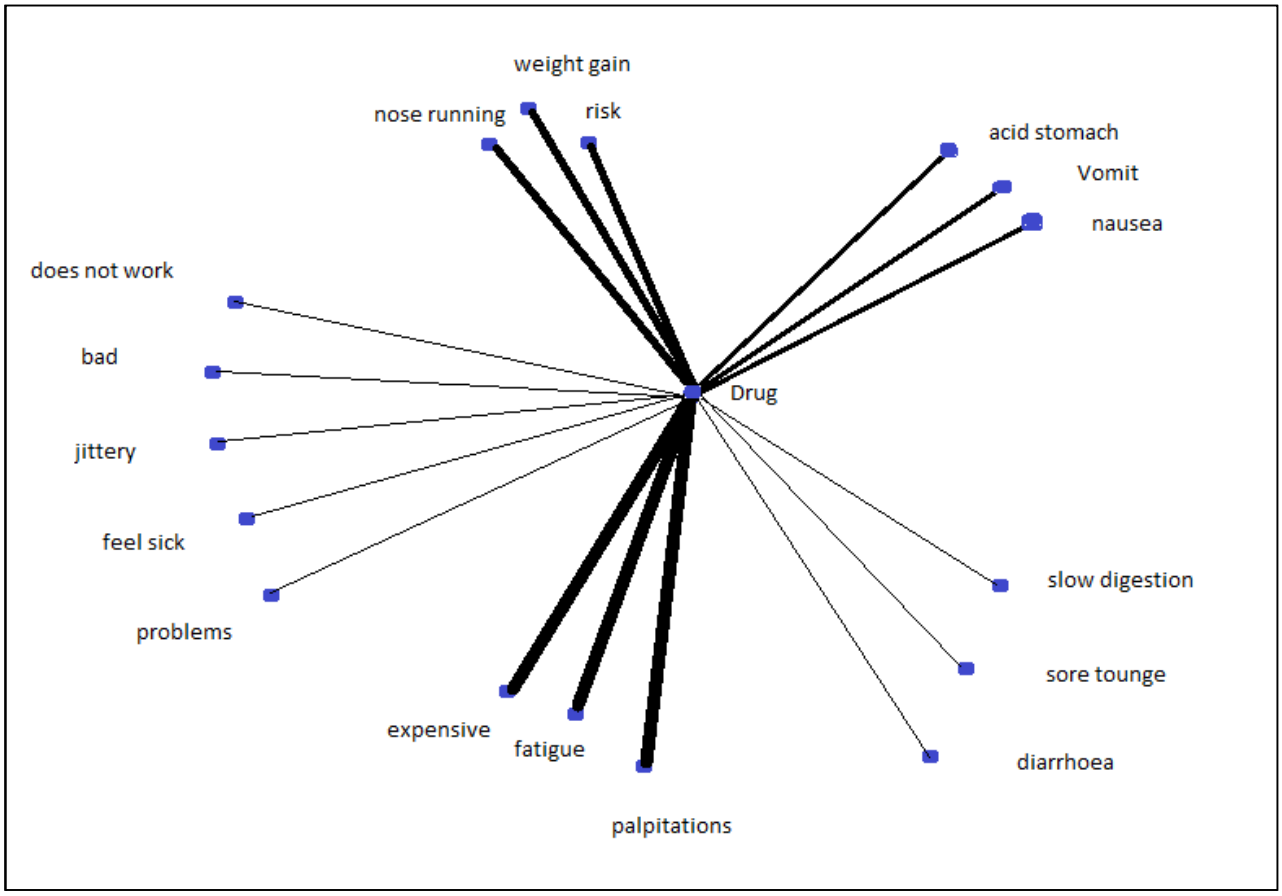

Figure 16: Tweets showing feedback about a drug.

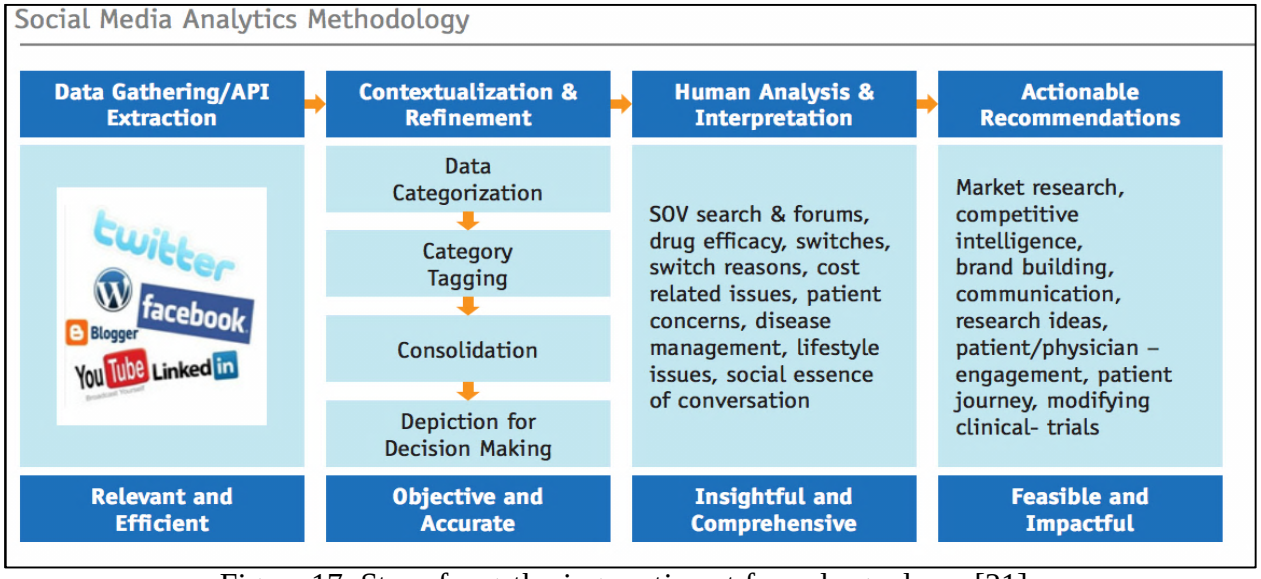

Figure 17: Steps for gathering sentiment for a drug release [21].

Application 3: Predicting public sentiment

Sentiment towards economic or public policy like bills or federal rate hike decisions can be predicted. Data analytics from Twitter can provide the sentiments of people from country to country. For example, the central bank of UK could have saved the pound from too much devaluation after Brexit by predicting the mood of citizens using Twitter analytics [3] [31].

Appendix

A.1 Java code for retrieval of data from a database and plotting the sentiment of a stock

```
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.ResultSet;
import java.sql.Statement;
import org.jfree.chart.ChartFactory;
import org.jfree.chart.ChartFrame;
import org.jfree.chart.ChartUtilities;
import org.jfree.chart.JFreeChart;
import org.jfree.chart.plot.PiePlot;
import org.jfree.chart.plot.PlotOrientation;
import org.jfree.data.category.DefaultCategoryDataset;
import org.jfree.data.general.DefaultPieDataset;

public class Results

 {
public static void main(String[] args)

{


    String topic="";
    String topic1="";
    String topic2="";
    String topic3="";
    String topic4="";
    String topic5="";


     int seg=0;
     int seg1=0;
     int seg2=0;
     int seg3=0;
     int seg4=0;
```

```
int seg5=0;

int count=0;


        try
        {
                Class.forName("com.mysql.jdbc.Driver");
                Connection con =
DriverManager.getConnection("jdbc:mysql://localhost:3306/twitter_schema","root","root");
                Statement st=con.createStatement();
                ResultSet rs=st.executeQuery("select * from finalcount");

                while(rs.next()==true)
                {
                count++;


                 if(count==1)
                 {
                        topic=rs.getString(1);
                        seg=rs.getInt(2);

                 }

                 if(count==2)
                 {
                        topic1=rs.getString(1);
                        seg1=rs.getInt(2);

                 }
                 if(count==3)
                 {
                        topic2=rs.getString(1);
                        seg2=rs.getInt(2);

                 }
                 if(count==4)
                 {
                        topic3=rs.getString(1);
```

```
                seg3=rs.getInt(2);

            }
            if(count==5)
            {
                topic4=rs.getString(1);
                seg4=rs.getInt(2);

            }
            if(count==6)
            {
                topic5=rs.getString(1);
                seg5=rs.getInt(2);

            }


           }


             DefaultCategoryDataset dataSet = new DefaultCategoryDataset();
          dataSet.setValue(seg, "No.of segmentation",topic+"");
          dataSet.setValue(seg1, "No.of segmentation1",topic1+"");
          dataSet.setValue(seg2, "No.of segmentation2",topic2+"");
          dataSet.setValue(seg3, "No.of segmentation3",topic3+"");
          dataSet.setValue(seg4, "No.of segmentation4",topic4+"");
          dataSet.setValue(seg5, "No.of segmentation5",topic5+"");


          JFreeChart chart = ChartFactory.createBarChart3D(
                  "Analysing twitter feeds to predict a trend in financial markets by
segmentation of feeds", "Total Tweet Segmentation  ", "No.Of Segmentation",
                  dataSet, PlotOrientation.VERTICAL, true, true, true);
          ChartFrame chartFrame=new ChartFrame("Tweet Segmentation
Details",chart);
          chartFrame.setVisible(true);
          chartFrame.setSize(800,500);


      }
```

```
            catch(Exception ex)
            {
            System.out.println(ex);

            }
    }
    }
```

B.1 Code for cleaning data

```
CREATE EXTERNAL TABLE newtweets_table_raw (
    tweet_id BIGINT,
    tweet_created_timestamp STRING,
    favourite BOOLEAN,
    retweet_count INT,
    retweeted_status STRUCT<
      text:STRING,
      user:STRUCT<screen_name:STRING,name:STRING>>,
    entities STRUCT<
      urls:ARRAY<STRUCT<expanded_url:STRING>>,
      user_mentions:ARRAY<STRUCT<screen_name:STRING,name:STRING>>,
      hashtags:ARRAY<STRUCT<text:STRING>>>,
    text STRING,
    user STRUCT<
      screen_name:STRING,
      name:STRING,
      friends_count:INT,
      followers_count:INT,
```

```
        statuses_count:INT,
        verified:BOOLEAN,
        utc_offset:STRING,
        timezone:STRING>,
    in_reply_to_screen_name STRING,
    yearoftweet int,
    monthoftweet int,
    dayoftweet int,
    houroftweet int
)
ROW FORMAT SERDE 'org.openx.data.jsonserde.JsonSerDe'
STORED AS TEXTFILE
LOCATION '/user/anoopv/newtweets'
```

B.2 Code for breaking tweets into words

```
CREATE VIEW newtweets_simple AS
    SELECT
        tweet_id,
        cast ( from_unixtime( unix_timestamp(concat( '2016',
substring(tweet_created_timestamp,6,01)),
        'yyyy MMM dd hh:mm:ss')) as timestamp) tmsp,
        text,
        user.timezone
    FROM newtweets_table_raw;
```

```
CREATE VIEW newtweets_clean AS
SELECT
    tweet_id,
    tmsp,
    text,
    FROM newtweets_simple n
LEFT OUTER JOIN timezone m ON n.timezone = m.timezone;
```

```
//Creating some views that can be used in the sentiment calculations. Three levels of views are
//created in this approach.
```

```
CREATE VIEW level1_breaking_of_words AS
SELECT
    tweet_id,
    words
FROM newtweets_table_raw LATERAL VIEW explode(sentences(lower(text))) dummy AS
words;
```

```
CREATE VIEW level2_breaking_of_words AS
SELECT
    tweet_id,
    word
FROM level1_breaking_of_words LATERAL VIEW explode( words ) dummy AS word ;
```

B.3 Code for creating a dictionary table and assigning sentiments to words

```
CREATE EXTERNAL TABLE dictionary_table (
        typeofword string,
        lengthofword int,
        word string,
        pos string,
        stemmed string,
        polarity string
    )
    ROW FORMAT DELIMITED FIELDS TERMINATED BY '\t'
    STORED AS TEXTFILE
    LOCATION '/user/anoopv/data/dictionary_table';

//Creating time zone table because tweets are from different countries so different time zone may
//be required for detail analysis.


    CREATE EXTERNAL TABLE timezone (
        timezone string,
                tweet_country string,
        notes string
    )
    ROW FORMAT DELIMITED FIELDS TERMINATED BY '\t'
    STORED AS TEXTFILE
```

```
LOCATION '/user/anoopv/data/timezone';
```

//To assign a sentiment value after lookup action of +1 for positive and -1 for negative:

```
CREATE VIEW level3_assign_polarity AS
SELECT
   tweet_id,
   level2_breaking_of_words.word,
   CASE d.polarity
      WHEN ‘negative’ THEN -1
      WHEN 'positive' THEN 1
      ELSE 0
   END AS polarity
FROM level2_breaking_of_words LEFT OUTER JOIN dictionary_table d ON
level2_breaking_of_words.word = d.word;
```

B.4 Code for calculating the aggregate scores

```
CREATE VIEW tweets_sentiment AS
SELECT
   tweet_id,
   CASE
                  WHEN sum( polarity ) > 2 THEN 'Very_bullish'
                  WHEN sum( polarity ) > 0 AND < 1 THEN ‘Bullish'
                  WHEN sum( polarity ) = 0 THEN 'Neutral'
                  WHEN sum( polarity ) > 0 AND < -1 THEN 'Bearish'
```

```
        WHEN sum( polarity ) < -2 THEN 'Very_bearish'
    ELSE 'Neutral'
  END AS sentiment
FROM level3_assign_polarity GROUP BY tweet_id;
```

B.5 Code for assigning sentiment indicators to tweets

```
// In Table 1, the aggregate score is used to assign sentiment. As these values are negative this may
//reduce performance of lookup action. To eliminate negative scores from the aggregate score
//calculation, positive numbers are assigned as 4 for Very_bullish, 3 for Bullish, 2 for neutral, 1 for
//Bearish and 0 for Very_bearish.

CREATE TABLE finaltweetsanalysis
    STORED AS RCFile
    AS
    SELECT
      n.*,
      case s.sentiment
        when 'Very_bullish' then 4
        when Bullish' then 3
        when 'Neutral' then 2
        when 'Bearish' then 1
        when 'Very_bearish' then 0
      end as sentiment
    FROM newtweets_clean n
    LEFT OUTER JOIN tweets_sentiment s on n.tweet_id = s.tweet_id;


//Below codes for checking total number of assigned indicators and sentiment strength
```

```
CREATE TABLE finalcount (
                    tc STRING,
                   count INT )

INSERT INTO finalcount (tc,count) VALUES
(Very_bullish,(select count(*) from finaltweetsanalysis where sentiment=4) as b1)
(Bullish,(select count(*) from finaltweetsanalysis where sentiment=3) as b2)
(Neutral,(select count(*) from finaltweetsanalysis where sentiment=2) as n)
(Bearish,(select count(*) from finaltweetsanalysis where sentiment=1) as s1)
(Very_bearish,(select count(*) from finaltweetsanalysis where sentiment=0) as s2)
(Sentiment_strength, (2*b1+b2)/(2*b1+b2+n+s1+2*s2))
```

Bibliography

[1] "Social media trends to predict the stock market". Available: http://www.unit.org.au/social-media-trends-to-predict-the-stock-market/

[2] P. Sam, "Using Twitter to gauge news effect on stock market moves". Available: http://cs229.stanford.edu/proj2013/Paglia-UsingTwitterNewsandReactionstoPredictFinancialMarketMoves.pdf

[3] N. Oliveira, P. Cortez, and N. Areal, "On the predictability of stock market behavior using stocktwits sentiment and posting volume," in *Portuguese Conference on Artificial Intelligence*, 2013, pp. 355-365.

[4] "Derwent capital markets". Available: https://en.wikipedia.org/wiki/Derwent_Capital_Markets

[5] C. Llewellyn and L. Cram, "Brexit? Analyzing opinion on the uk-eu referendum within twitter," in *Proceedings of the International AAAI Conference on Web and Social Media*, 2016, pp. 760-761.

[6] "Technical Indicators". Available: http://www.investopedia.com/terms/t/technicalindicator.asp

[7] "Data mining". Available: https://en.wikipedia.org/wiki/Data_mining

[8] "Twitter usages statistics". Available: http://www.statista.com/statistics/282087/number-of-monthly-active-twitter-users/

[9] D. Hicks, "Twitter as delayed gratification". Available: https://hibasme.wordpress.com/category/statistics/

[10] "Cognitive computing". Available: http://whatis.techtarget.com/definition/cognitive-computing

[11] P. Rich, ""How does IBM Watson work?". Available: http://adnarchist.com/how-does-ibm-watson-work-video-transcript/

[12] S. Gandel, "How to use Twitter to make millions and beat the market, maybe". Available: http://business.time.com/2011/07/13/how-to-use-twitter-to-make-millions-and-beat-the-market-maybe/

[13] X. Liu, S. Zhang, F. Wei, and M. Zhou, "Recognizing named entities in tweets," in *Proceedings of the Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, 2011, pp. 359-367.

[14] "Definitions of Big Data". Available: http://www.opentracker.net/article/definitions-big-data

[15] "Big Data". Available: https://en.wikipedia.org/wiki/Big_data

[16] "Hadoop application software". Available: https://gigaom.com/2012/02/06/what-it-really-means-when-someone-says-hadoop/

[17] "Hive introduction". Available: http://www.tutorialspoint.com/hive/hive_introduction.htm

[18] "Hadoop introduction". Available: http://www.hadooptpoint.com/category/hive/

[19] "Hive architecture". Available: http://www.cubrid.org/blog/dev-platform/platforms-for-big-data/

[20] D. Ferrucci, E. Brown, J. Chu-Carroll, J. Fan, D. Gondek, A. Kalyanpur, A. Lally, J. William Murdock, E. Nyberg, J. Prager, N. Schlaefer, and C. Welty "The AI behind Watson". Available: http://www.aaai.org/Magazine/Watson/watson.php

[21] M. Migurski, "A beginners guide to streamed data from Twitter", 2012. Available: http://mike.teczno.com/notes/streaming-data-from-twitter.html

[22] A. Moujahid, "An introduction to text mining using Twitter streaming API and Python", 2014. Available: http://adilmoujahid.com/posts/2014/07/twitter-analytics/

[23] "Postive words and adjectives list for sentiment analysis". Available: http://positivewordsresearch.com/2015/05/08/list-of-negative-words/

[24] "Negative words and adjectives list for sentiment analysis". Available: http://dreference.blogspot.com/2010/05/negative-ve-words-adjectives-list-for.html

[25] "Opinion-lexicon-English". Available: https://github.com/jeffreybreen/twitter-sentiment-analysis-tutorial-201107/tree/master/data/opinion-lexicon-English

[26] C. Li, A. Sun, J. Weng, and Q. He, "Tweet segmentation and its application to named entity recognition," *IEEE Transactions on Knowledge and Data Engineering*, vol. 27, pp. 558-570, 2015.

[27] " Stock price movement reference". Available: http://in.investing.com/equities/microsoft-corp-chart

[28] K. Nishida, T. Hoshide, and K. Fujimura, "Improving tweet stream classification by detecting changes in word probability," in *Proceedings of the International ACM SIGIR Conference on Research and Development in Information Retrieval*, 2012, pp. 971-980.

[29] "Pharma & healthcare social media marketing strategies", 2010. Available: http://www.marketsandmarkets.com/Market-Reports/Pharma-Social-Media-245.html

[30] I. Vermeren, "Noting the rising role of social analytics in healthcare", 2015. Available: https://www.brandwatch.com/2015/03/research-noting-rising-role-social-analytics-healthcare/

[31] J. Bollen, H. Mao, and X. Zeng, "Twitter mood predicts the stock market," *Journal of Computational Science*, vol. 2, pp. 1-8, 2011.