

# WQD 7005 Data Mining

## Assignment

Student Name: Tan Kai Xuan

Student ID: WQD180036

Team members:

Name	Student ID
Cheng Jiechao	WQD180050
Yew Siew Chen	WQD180004
Choo Jia Yuan	WQD180052
Lim Kaomin	WQD180076

## **1. INTRODUCTION**

Stock market refers to the collection of markets and exchange where regular activities of selling and buying of publicly-held companies take place. Today, it is very easy to do online purchasing for everything. Hence, many platforms are designed for numerous buyers and sellers to interact and transact. Stock market is one of the designated platforms for trading various kind of securities in controlled, secure and managed the environment. Stock market brings thousands of market participants who wish to buy and sell shares, by ensuring the fair pricing practice and transparent transaction.

In this data mining project, Profesor Madya Dr. Teh Ying Wah has given an opportunity for the students to explore the stock market and analyse the market by applying the knowledge from the lecture. The project is separated into 6 milestones which are the guidelines to achieve the end goal – stock market analysis. In this report, I will describe the process and the outcome that I've gone through all the milestones.

## **ANALYSIS GOAL**

Predict the future value of a company stock or other financial instrument traded on a financial exchange by studying and evaluating past and current data. The successful prediction could yield significant profit.

## 2. MILESTONE 1 (GROUP)

We did web scrapping in milestone 1 to obtain the data from the website. We've total scrapped several data from the websites which are stock data, stock news headline, stock details, as well as the financial report of the company. In this process, it is quite a challenging part to those who are not in computer science background like me. My teammates and I were struggling in this milestone as none of us are in computer science background. Hence, we did a lot of web searching and attend some of the online courses such as YouTube and Datacamp to learn how to use python in scrapped the data from the website.

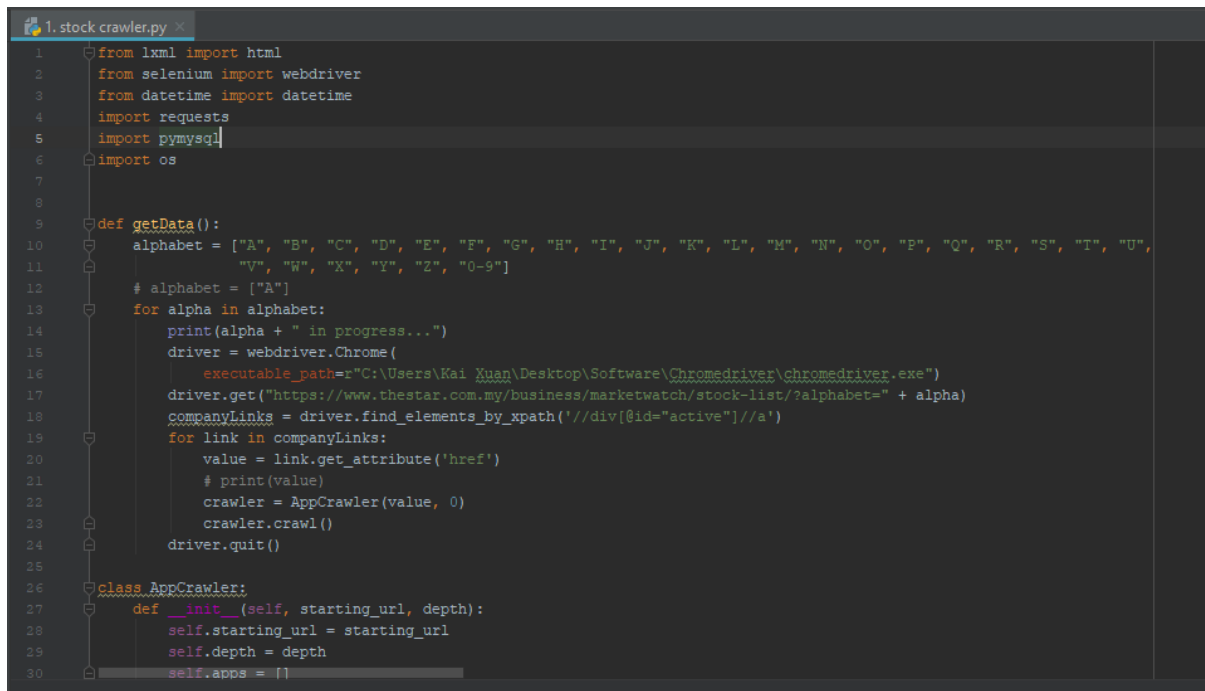
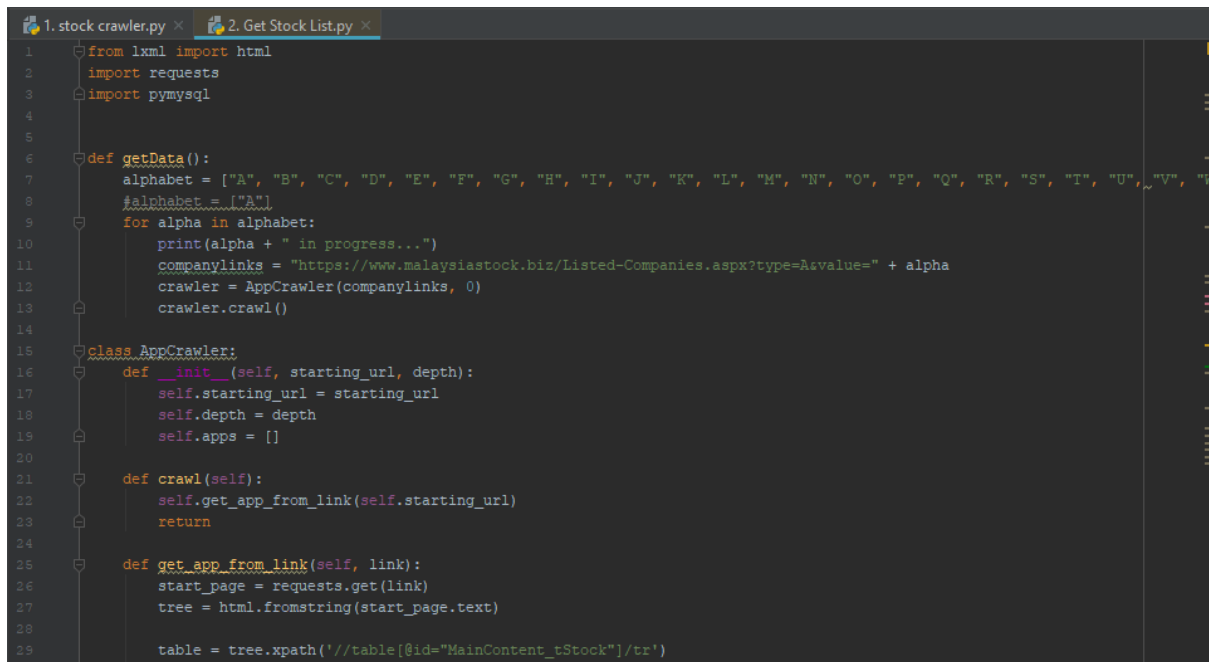
The image is a screenshot of a code editor showing a Python script named '1. stock crawler.py'. The script is designed to scrape stock data from The Star website. It begins with several import statements: 'from lxml import html', 'from selenium import webdriver', 'from datetime import datetime', 'import requests', 'import pymysql', and 'import os'. A function 'def getData()' is defined, which iterates through an alphabet list (including letters A-Z and digits 0-9). For each letter, it prints a progress message, initializes a Selenium WebDriver for Chrome, and navigates to a specific URL on The Star's website. It then finds company links using XPath and iterates through them, extracting the href attribute and creating an instance of an 'AppCrawler' class. The 'AppCrawler' class has an '\_\_init\_\_' method that takes 'starting\_url' and 'depth' as parameters and initializes 'self.apps' as an empty list. The script concludes with 'driver.quit()'.

Figure 2.1: Web scrapping from The Star

Figure 2.1 shows the screenshot of the web scrapping from The Star. We've scrapped around 1800 stocks per day from 1<sup>st</sup> March 2019 to 26<sup>th</sup> April 2019, total 76,341 stock data scrapped from The Star.



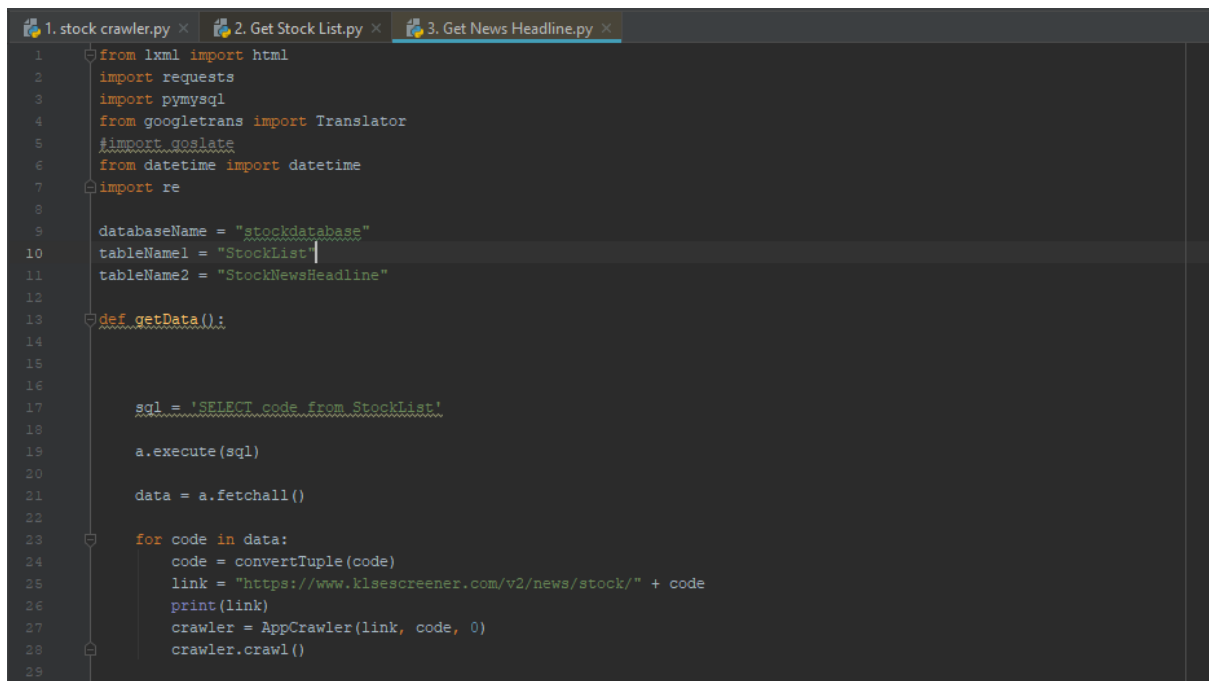
```

1 from lxml import html
2 import requests
3 import pymysql
4
5
6 def getData():
7     alphabet = ["A", "B", "C", "D", "E", "F", "G", "H", "I", "J", "K", "L", "M", "N", "O", "P", "Q", "R", "S", "T", "U", "V", "W", "X", "Y", "Z"]
8     #alphabet = ["A"]
9     for alpha in alphabet:
10         print(alpha + " in progress...")
11         companylinks = "https://www.malaysiastock.biz/Listed-Companies.aspx?type=A&value=" + alpha
12         crawler = AppCrawler(companylinks, 0)
13         crawler.crawl()
14
15 class AppCrawler:
16     def __init__(self, starting_url, depth):
17         self.starting_url = starting_url
18         self.depth = depth
19         self.apps = []
20
21     def crawl(self):
22         self.get_app_from_link(self.starting_url)
23         return
24
25     def get_app_from_link(self, link):
26         start_page = requests.get(link)
27         tree = html.fromstring(start_page.text)
28
29         table = tree.xpath('//table[@id="MainContent_tStock"]/tr')

```

Figure 2.2: Web scrapping from MalaysiaStock.biz

Figure 2.2 shows the screenshot of the web scrapping from MalaysiaStock.biz. We've scrapped total 949 company details including the full name of the company, sector and board.



```

1 from lxml import html
2 import requests
3 import pymysql
4 from googletrans import Translator
5 #import goslate
6 from datetime import datetime
7 import re
8
9 databaseName = "stockdatabase"
10 tableName1 = "StockList"
11 tableName2 = "StockNewsHeadline"
12
13 def getData():
14
15
16
17     sql = 'SELECT code from StockList'
18
19     a.execute(sql)
20
21     data = a.fetchall()
22
23     for code in data:
24         code = convertTuple(code)
25         link = "https://www.klsescreener.com/v2/news/stock/" + code
26         print(link)
27         crawler = AppCrawler(link, code, 0)
28         crawler.crawl()
29

```

Figure 2.3: Web scrapping from KLSE

Figure 2.3 shows the screenshot of the web scrapping from KLSE. We've scrapped the news headline based on the company list that we have from Figure 2.2 for two months.

```
1. stock crawler.py × 2. Get Stock List.py × 3. Get News Headline.py × 4. FinancialReport.py ×
4 from selenium.webdriver.chrome.options import Options
5
6 databaseName = "stockdatabase"
7 tableName1 = "StockList"
8 tableName2 = "StockFinancial"
9
10
11 #WINDOW_SIZE = "1920,1080"
12 chrome_options = Options()
13 chrome_options.add_argument("--headless")
14 #chrome_options.add_argument("--window-size=%s" % WINDOW_SIZE)
15
16 def getData():
17
18
19
20 sql = 'SELECT code from StockList'
21
22 a.execute(sql)
23
24 data = a.fetchall()
25
26 for code in data:
27     code = convertTuple(code)
28     link = "https://klse.i3investor.com/servlets/stk/fin/" + code + ".jsp"
29     print(link)
30     driver = webdriver.Chrome(
31         executable_path=r"C:\Users\Kai Xuan\Desktop\Software\Chromedriver\chromedriver.exe", options=chrome_options)
32     driver.get(link)
```

Figure 2.4: Web scrapping from investor.com

Figure 2.4 shows the screenshot of the web scrapping from investor.com. We've scrapped the financial reports with structure format from this website. The data we've scrapped are announcement date, revenue, PBT, ROE, EPS and so on.

### 3. MILESTONE 2 (GROUP)

Milestone 2 is the data management which we need to put all the data that we scrapped into phpMyAdmin. phpMyAdmin is a free and open source administration tool for MySQL. It is one of the most popular application for MySQL database management which written in PHP. To use phpMyAdmin, XAMPP is needed to be installed on computer. There is a python library available for us to use it with MySQL database which is PyMySQL. Once we have PyMySQL installed, we can start operating MySQL database from Python.

Server: 127.0.0.1 » Database: stockdatabase » Table: stockprice

Current selection does not contain a unique column. Grid edit, checkbox, Edit, Copy and Delete features are not available.

Showing rows 0 - 24 (108979 total, Query took 0.0554 seconds.)

SELECT \* FROM `stockprice`

Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code] [Refresh]

1 > >> Number of rows: 25 Filter rows: Search this table

ID	Name	Code	Board	Date	Time	52WeekHigh	52WeekLow	Open	High	Low	Last	Chg	ChgPercent
201903011911070322	A50CHIN-C22: CW ETF ISHARES FTSE A50 CHINA INDEX E...	070322	Warrants	01 Mar 2019	7:11 PM	0.395	0.015	0.210	0.280	0.210	0.275	0.075	37.50
201903011911070324	A50CHIN-C24: CW ETF ISHARES FTSE A50 CHINA INDEX E...	070324	Warrants	01 Mar 2019	7:11 PM	0.830	0.135	0.800	0.830	0.800	0.830	0.095	12.93
201903011911070326	A50CHIN-C26: CW ISHARES FTSE A50 CHINA INDEX ETF	070326	Warrants	01 Mar 2019	7:11 PM	0.505	0.185	0.465	0.505	0.465	0.505	0.045	9.78

Figure 3.1: phpMyAdmin window

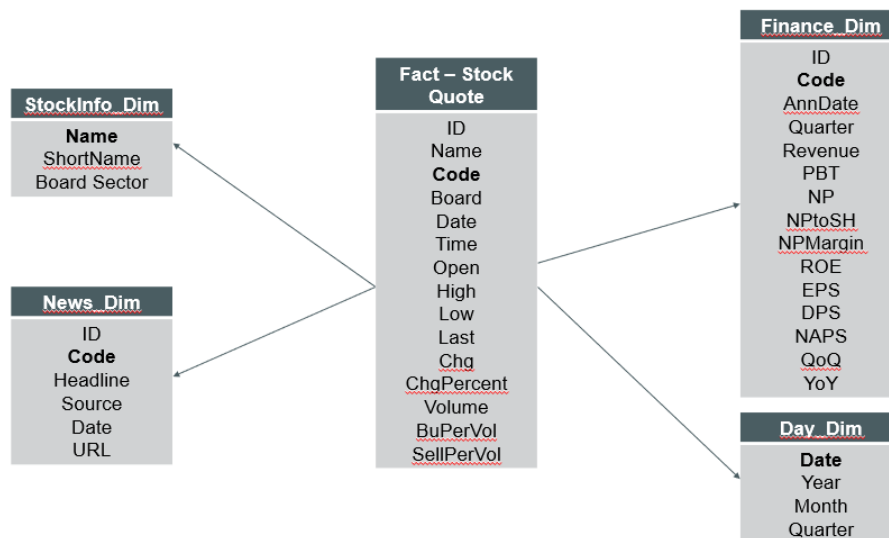


Figure 3.2: Star Schema

The importance of OLAP dimension is to allow users to roll up and drill down on the dataset they have.

Through Day dimension, we can drill down to get the information for the stock price from days up to years. Users are interested the changes of the stock price by days or even by years. With this dimension, users can get the stock price from day to day, month to month, quarter to quarter and year to year in order to get the patterns of the stock.

For finance dimension, we can get the financial information by quarter or by year. Users can compare the financial status of the company (for example, revenue, net profit) and its stock price.

For stock info dimension, users can know the full name and the short name of the stocks as well as the sector of the stocks.

Lastly, users can know the impact of the news either a positive news or negative news to the stock price through news headline dimensions. In this dimension, URL of the news website will be included so users can always refer to the URL to get the detailed of the news.

#### 4. MILESTONE 3 (GROUP)

Covariance provides insights into how two variables are related to each other, which means it is the measurement of how two variables change together in a data set. A positive covariance means that two variables at hand are positively related, and they move in the same direction. A negative covariance means that the variables are inversely related.

```
oneSAX_paa_sax_stock_time_series.py
1
2 # coding: utf-8
3
4 import ...
14
15 # features for stock data
16 # stock_code + stock_name + stock_ref + stock_open + stock_last + stock_change + stock_change_perc + stock_volume
17
18
19 # read data from file list
20
21 path = "./dataset_5d"
22 files = []
23 # r=root, d=directories, f = files
24 for r, d, f in os.walk(path):
25     for file in f:
26         if '.txt' in file:
27             files.append(os.path.join(r, file))
28 print("all files: ")
29 for f in files:
30     print(f)
31
32 # make time series table based on different features
33
34 features = ["stock_code", "stock_name", "stock_ref", "stock_open",
35            "stock_last", "stock_change", "stock_change_percent", "stock_volume"]
36 df_feature= []
37 for fea in features:
38     df_day = []
39     for f in files:
```

Figure 4.1: oneSAX\_paa\_sax\_stock\_time\_series python code

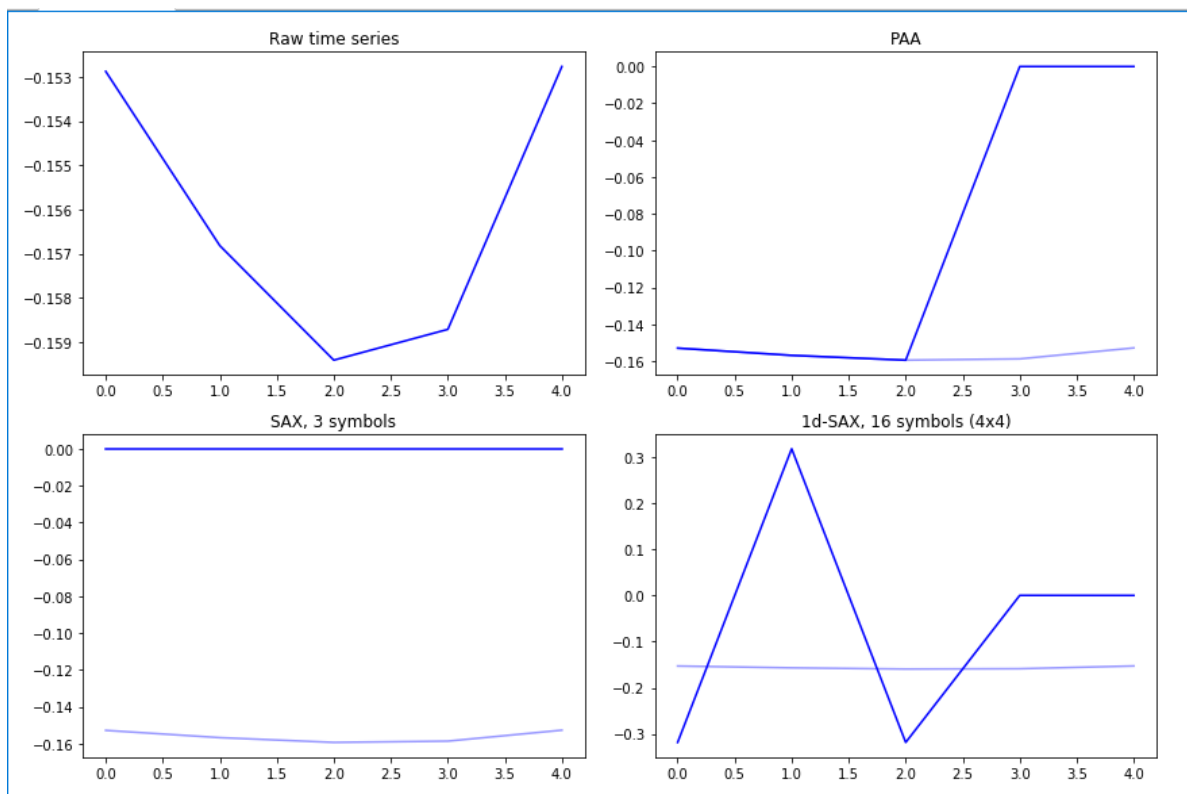


Figure 4.2: Output of paa sax and 1d-sax



Commonly, time series datasets and databases tend to grow to extremely large sizes. Sampling consistently is a requirement in a lot of cases where these databases are involved. During this process, the ability to search through the database becomes unreasonably time consuming. Piecewise Aggregate Approximation of time series (PAA) reduce the dimensionality of the input time series by splitting them into equal-sized segments which are computed by averaging the values in these segments. PAA leads directly to another representation called Symbolic Aggregate approXimation (SAX). This takes the reduced dimensionality and assigns a string representation to the graph. By providing a string representation, the overall data that is required to be stored less than other data mining methods.

## 5. MILESTONE 4 (INDIVIDUAL)

Data interpretation is very important as it assigns the meaning to the information as well as determine its significant and implication. In this step, data interpretation is separated into quantitative and qualitative.

For quantitative analysis, SAS Enterprise Miner is used to do some visualisation of the data set that we've crawled in milestone 1, mainly focus on telecommunication service providers. This sector has total 5 telco in the stock list provided by The Star, they are Axiata Group Berhad, Digi.com Berhad, Maxis Berhad, TIME DOTCOM Berhad as well as Telekom Malaysia Berhad.

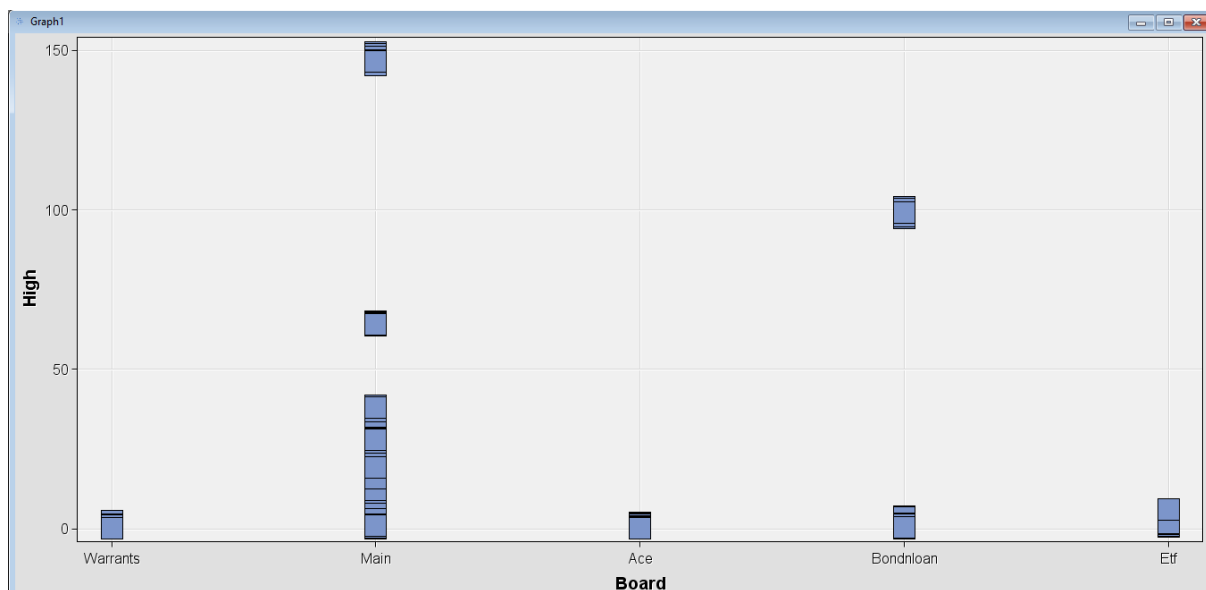


Figure 5.1: High price among the boards

Figure 5.1 shows the comparison of the high price between the boards. From the graph above, we can see than the highest price of the stock is in Main board, whereas the lowest price is in Warrants board.

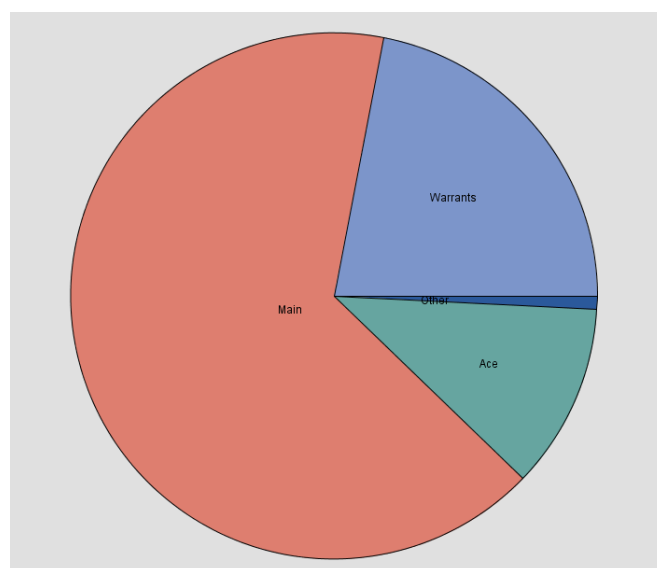


Figure 5.2: Frequency of boards

From Figure 5.2, we know that most of the stock data is from Main board and the highest price in figure 5 is Main board. For the Warrants board, although its frequency is the second highest, their price is the lowest compared to other boards.

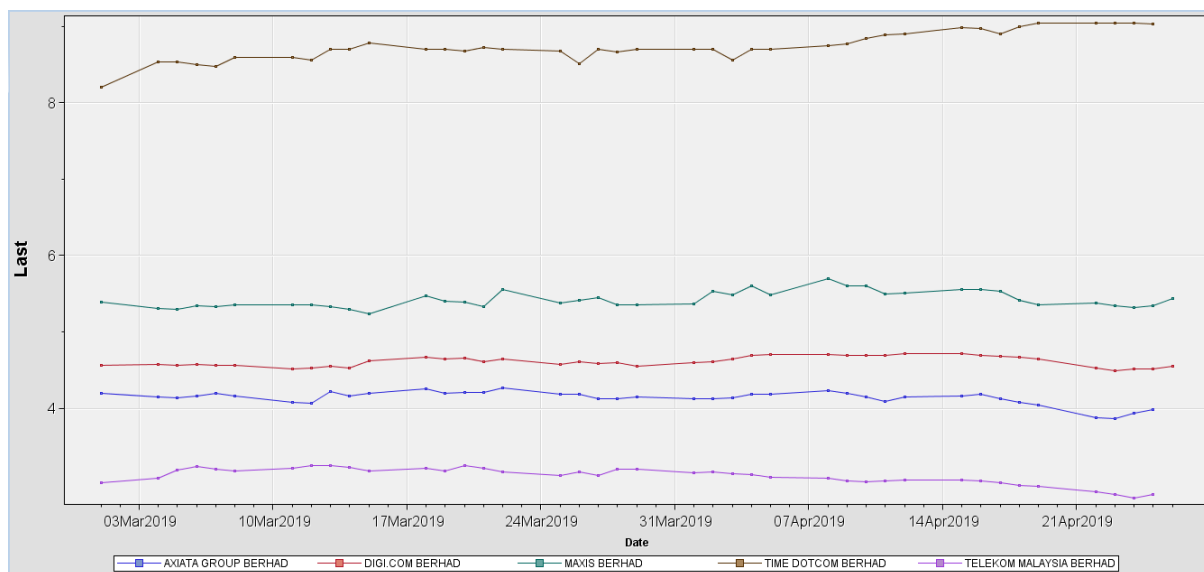


Figure 5.3: Last price by days

From Figure 5.3, we can know that the closing price for each stock for every single day. Each line represents to one telco in Malaysia market. From the graph above, we know that TIME DOTCOM Berhad has the highest stock price among all the telco.

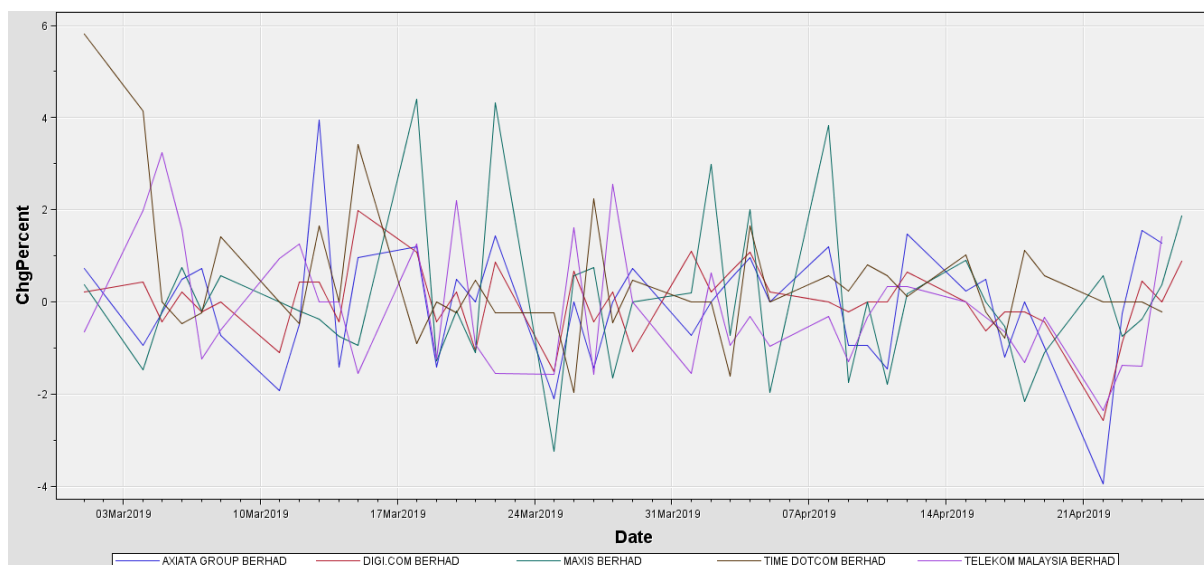


Figure 5.4: Changed percent by days

Figure 5.4 shows the change percent of the stock price for each telco per day. If the change percent is positive, it means that the stock price has increased compared to previous day. If the change percent is negative, the stock price has decreased as well as if the stock price is zero, it means there is no changes on the stock price.

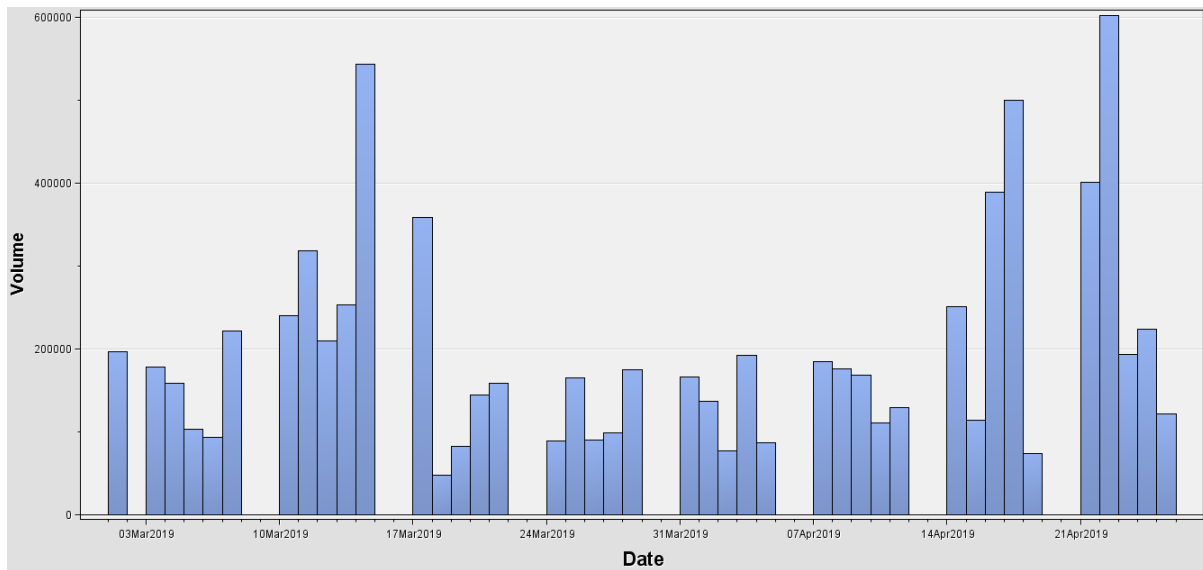


Figure 5.5: Volume by days

From Figure 5.5, we know the total number of shares traded of telco sector from 1st March 2019 until 26th April 2019. 23rd April has the highest volume in telco market during this period.

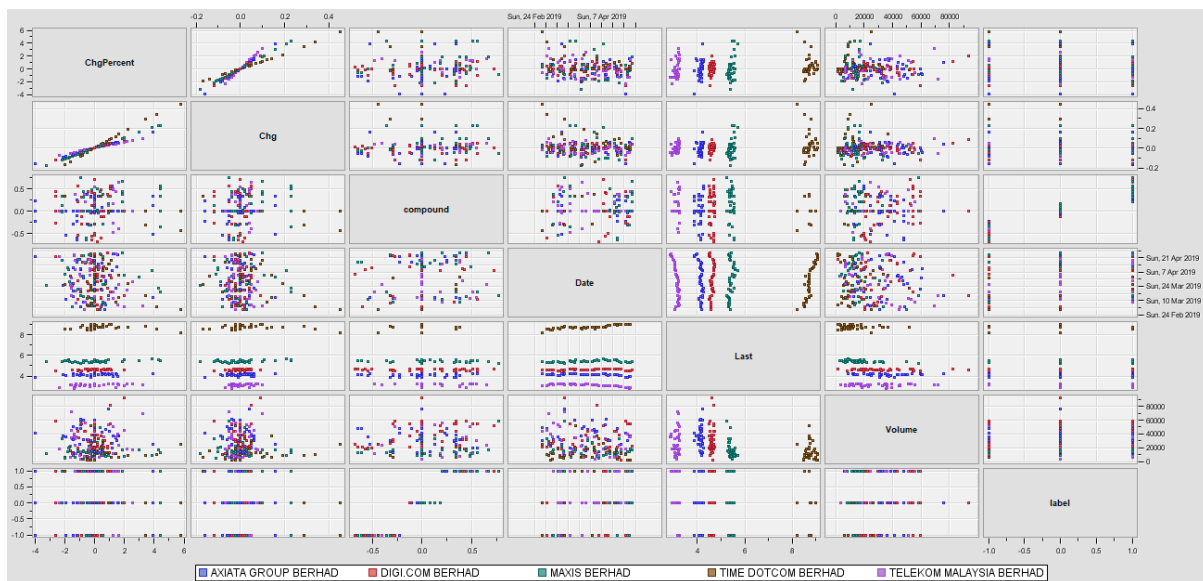


Figure 5.6: Correlation matrix

Figure 5.6 shows the correlation matrix with selected variables: Date, Last, Change Percent, Change, Volume, Compound and Label. Compound and Label are based on the sentiment analysis of the news headline that we've crawled mainly from The Edge and The Star.

In qualitative analysis, I applied sentiment analysis on the news headline on telecommunication service providers. Sentiment analysis combines the power of natural language processing and text analysis to classify response as 'positive', 'negative' or 'neutral'. NLTK's built in Vader Sentiment Analyzer will rank the headline as positive, negative or neutral using a lexicon of positive or negative words. There are total 4 columns from the

sentiment scoring: Positive, Negative, Neutral and compound. Compound is a single number that scores the sentiment which ranges from -1 to 1. I will consider the compound value greater than 0.2 as positive and less than -0.2 as negative.

```
Positive headlines:

['klci erases gains in line with cautious regional markets',
'time dotcom fy2018 net profit soars to rm289 million',
'putrajaya isnt supposed to buy loyalty with pay rise says dr m',
'buying loyalty not the way says mahathir',
'klci gains 0.26 as select blue chips lift']

Negative headlines:

['mixed results for telco sector amid moderating regulatory pressures',
'politics sent malaysia stocks up now u-turns as doubts emerge',
'telekom malaysia cut to neutral at public investment bank',
'klci drifts lower in line with regional pause',
'klci drifts lower in line with regional pause']
```

Figure 5.6: Outputs of positive and negative headlines

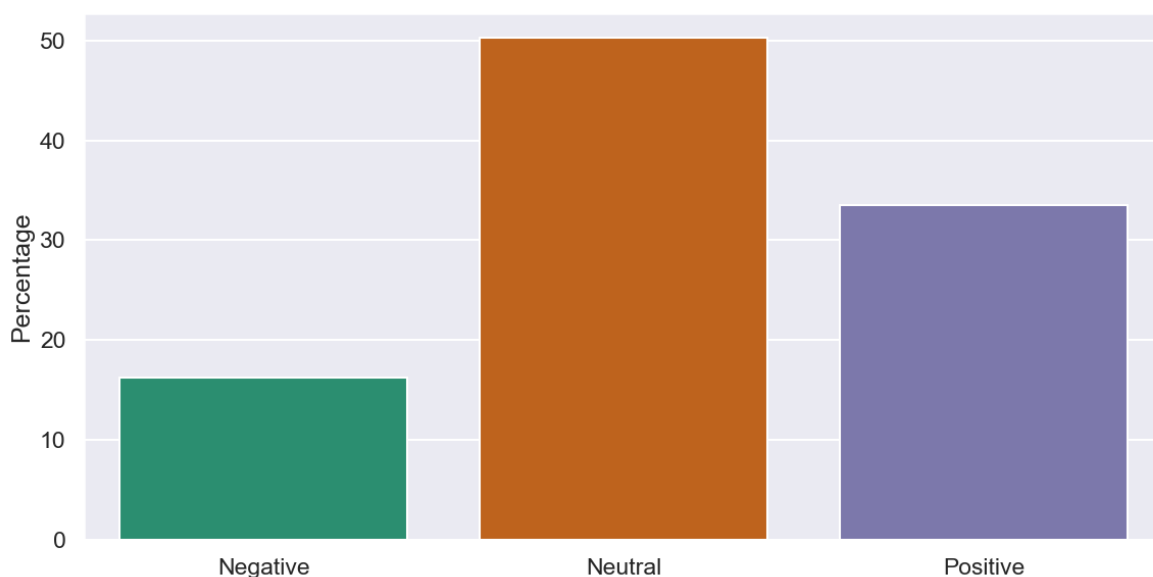


Figure 5.6: Summary of headlines

Next, I am going to show the comparison of qualitative (sum of compound) and quantitative (last price of the day) using SAS Enterprise Miner.

Stock prices move up and down every minute due to fluctuations in supply and demand. If more people want to buy a particular stock, its market price will increase. Conversely, if more people want to sell a stock, its price will drop. This relationship between supply and demand is tied into the type of news reports that are issued at any particular moment.

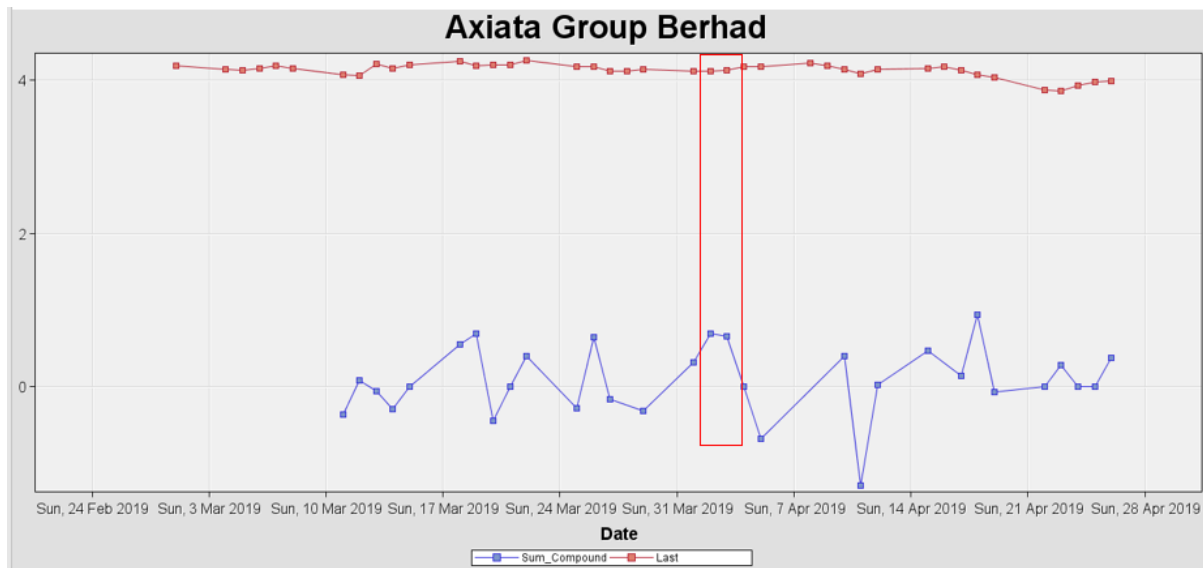


Figure 5.7: Axiata Group Berhad

Figure 5.7 shows different variables in one graph. X-axis represents Date, Y-axis represents two variable that we are going to interpret: blue colour line for Sum\_Compound and red colour line shows the Last price of the day. From Figure 10, we can see that the sum of compound is closely related to the last price of the day. In the red box pointed in Figure 10, we know that when the sum of compound is higher compared to previous day, the last price will tend to increase. However, when the sum of compound dropped, the last price will be dropped eventually.

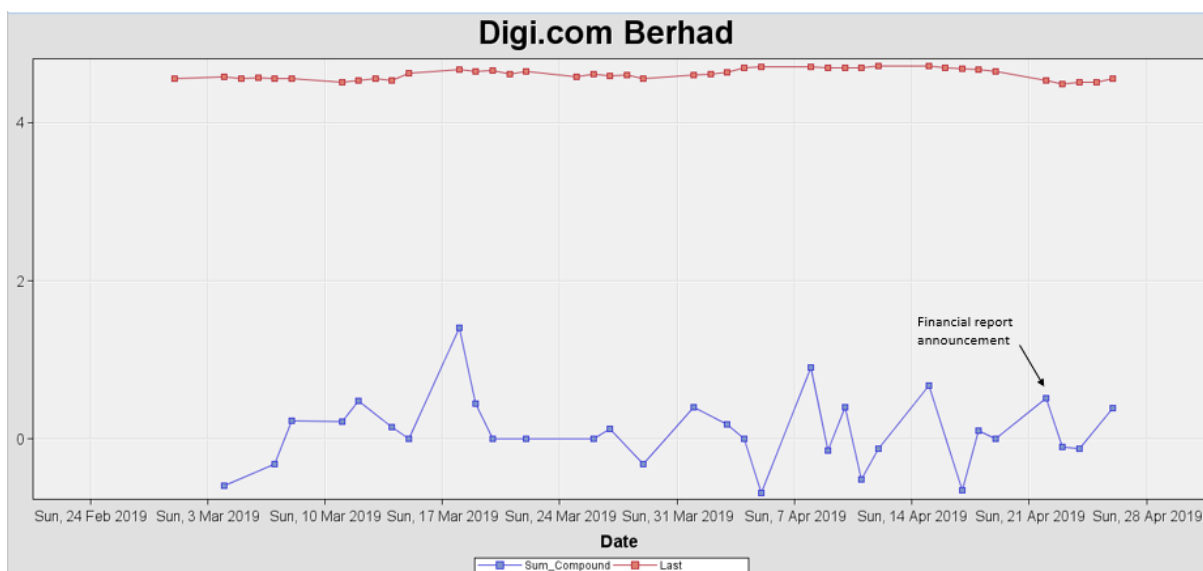


Figure 5.8: Digi.com Berhad

Figure 5.8 included the financial report announcement for Digi.com Berhad on 22nd April 2019. We can see that the sum of compound is increased compared to the previous day. Despite of the last price is the lowest, the last price is increasing for the next two days. Hence, the financial announcement is one of the factors that affect the last price of the stocks.

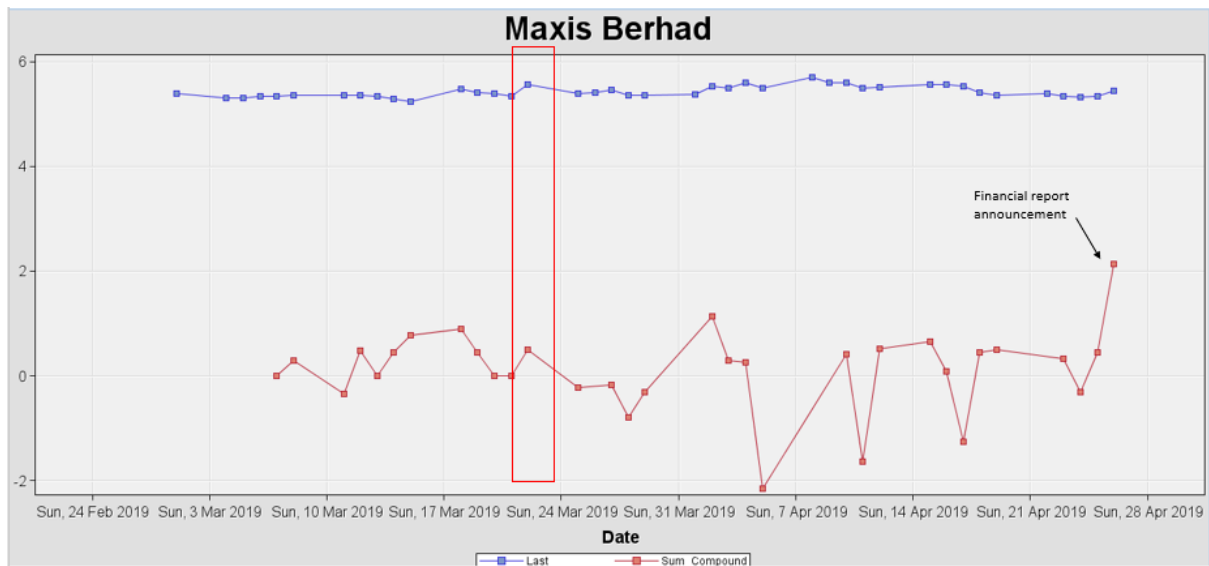


Figure 5.9: Maxis Berhad

In Figure 5.9, we can see a very obvious changes of the sum of compound which affect the last price of the day in Maxis Berhad. The financial report announcement that day, the sum of compound is the highest, and thus the price of the Maxis Berhad increasing.

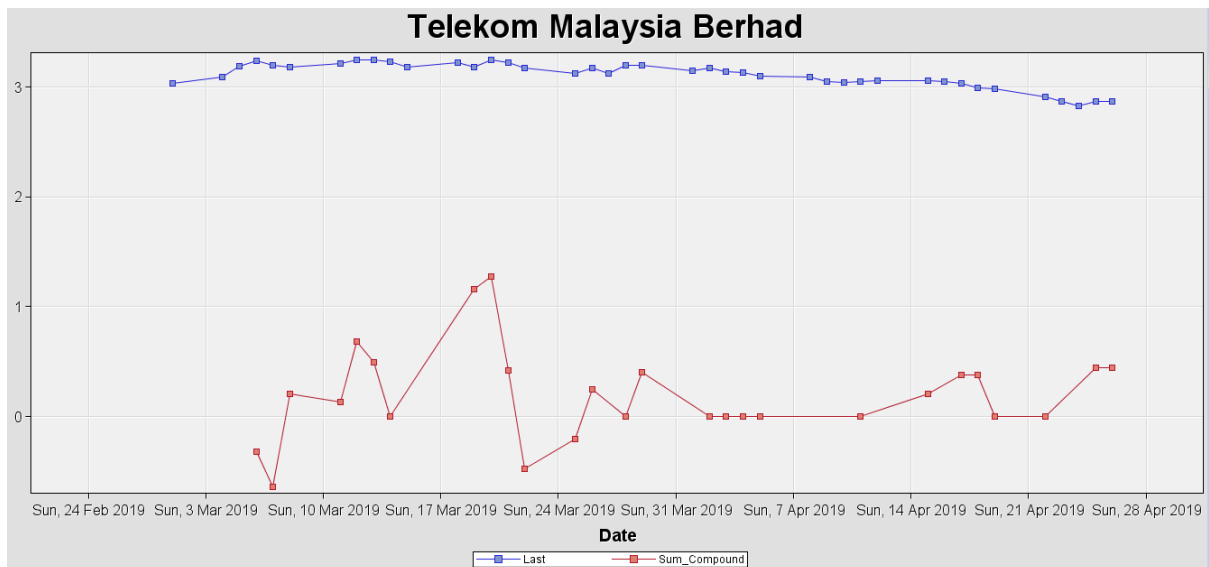


Figure 5.10: Telekom Malaysia Berhad

In Figure 5.10, we know that the sum of compound affects the stock price and cause to increase or fluctuate accordingly.

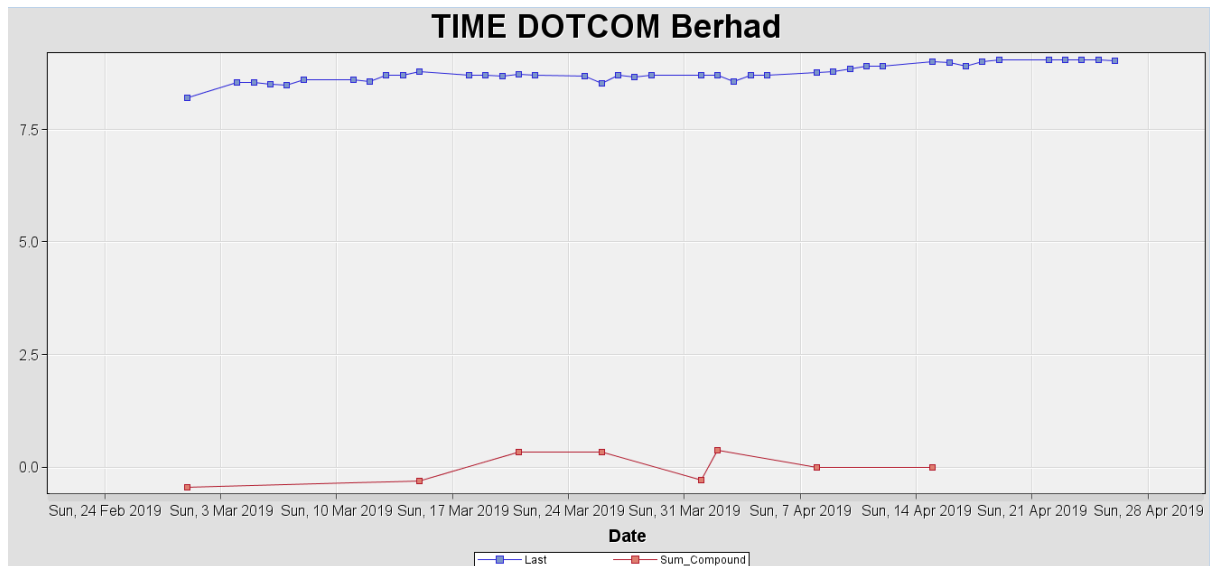


Figure 5.11: TIME DOTCOM Berhad

In Figure 5.11, there is no any news after 14th April, thus, the price of TIME DOTCOM Berhad is kept consistently for the next few days.

In conclusion, from Figure 5.7 to Figure 5.11 for different telco company, we know that the news is closely related to the price of the day. Negative news will cause the individual to sell the stocks, and the price will fall on that particular day or a few days. Positive news will normally cause individuals to buy the stocks and caused the increase in stock price.





## 7. MILESTONE 6 (INDIVIDUAL)

In this milestone, another data mining technique is used for stock market prediction – association rule. Association rule is one of the most interesting research areas for finding the associations, correlations among items in a database. It can discover all useful patterns from stock market dataset. The association rule states that if a customer purchase X, items than he or she is also likely to purchase Y items. Association rules are usually required to satisfy a user-specified minimum support (minsup) and confidence (minconf) at the same time.

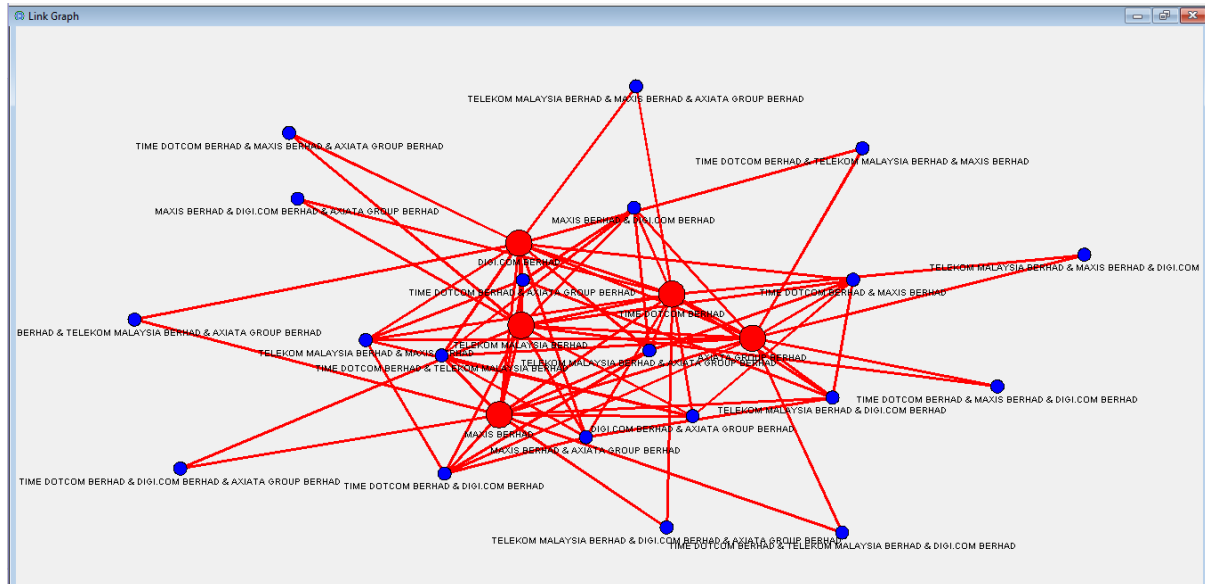


Figure 7.1: Association rules model

Figure 7.1 shows the relationship between the companies within a sector and how they associate with each other. These rules can help to know the market conditions and guide the user when to invest in the market.

## 8. CONCLUSION

Predicting the future is one aspect in designing profitable day trading strategies. Technical analysis analyses price, volume and other market information, whereas fundamental analysis looks at the facts of the company, market, currency or commodity. In this project, both technical analysis and fundamental analysis are taken into consideration by converting the fundamental analysis into sentiment analysis by generating numeric value. Decision Tree and Association Rules models are generated by using SAS Enterprise Miner. These two models are using different techniques to predict the stock market. However, due to the time limitation and insufficient data set to work on both models, I am unable to determine which model perform better. Hence, in future, if I am given enough time and large enough data set, I will be able to generate a better model by using SAS Enterprise Miner.

### Reference:

*DataCamp*. (n.d.). Retrieved from <https://www.datacamp.com/home>

Georges, J. (2010). *Applied Analytic Using SAS Enterprise Miner*. NC, USA.

Sachin Kamley, S. J. (2014). An Association Rule Mining Model for Finding the Interesting Patterns in Stock Market Dataset. 20.

Shubhangi S. Umbarkar, P. S. (2013). Using Association Rule Mining. *Stock Market Events Prediction from Financial News* , 6.

Shweta Tiwari, P. R. (n.d.). Predicting future trends in stock market by decision tree rough-set based hybrid system with HHMM. 10.

### Video Link:

[https://drive.google.com/file/d/1zZ5IzM4f\\_YvcxqQJxzdKd3axWRP0De29/view?usp=sharing](https://drive.google.com/file/d/1zZ5IzM4f_YvcxqQJxzdKd3axWRP0De29/view?usp=sharing)

### GitHub Link:

[https://github.com/kaixuan-tan/DataMining\\_Project](https://github.com/kaixuan-tan/DataMining_Project)