Apple Earnings Call Transcripts Text Analysis: Sentiment Analysis and Stock Price

Huang Lin, Chun 1A233902-6

Political Text Analysis

Research Motivation

When considering project topic, I tried to find a good text source with periodical update and associated with economic environment. After topics searching, I decided to analyze the transcripts of Apple's Earnings Call. Apple held earnings call four times a year and often launched new products with each earnings call. Therefore, I feel it is a good theme to analyze the expression across time, the keyness, the emotion index and its impact on the stock price (AAPL).

```
Sys.setlocale("LC_ALL", 'en_GB.UTF-8')
Sys.setenv(LANG = "en_GB.UTF-8")
getwd()
library(tidyverse)
library(readtext)
library(quanteda)
library(quanteda.textstats)
library(quanteda.textplots)
library(quanteda.textmodels)
library(topicmodels)
if(!require("keyATM")) {install.packages("keyATM"); library(keyATM)}
```

Check the files in our target folder.

```
list.files(path = './transcripts')
```

Import each file into R

```
earn.call <- readtext("./transcripts/*.txt",
                      docvarsfrom = "filenames",
                      dvsep = "_",
                      docvarnames = c("name", "year", "quarter", "date"))
earn.call$text <- gsub("'", " ", earn.call$text)
earn.call$text <- gsub("'", " ", earn.call$text)
```

Data Preprocessing

```
earn.call.corpus <- earn.call %>%

  # Turn earn.call data into corpus
  corpus() %>%

  # Remove punctuation, symbols, numbers, separators, and stopwords
  tokens(remove_punct = TRUE,
         remove_symbols = TRUE,
         remove_numbers = TRUE,
         remove_separators = TRUE) %>%
  tokens_remove(stopwords(language = "en")) %>%

  # lemmatization
  tokens_wordstem(language = "en")

# Turn them to DFM
earn.call.dfm <- earn.call.corpus %>%
  dfm() %>%
  dfm_remove(min_nchar=2)
```

Trimming the DFM

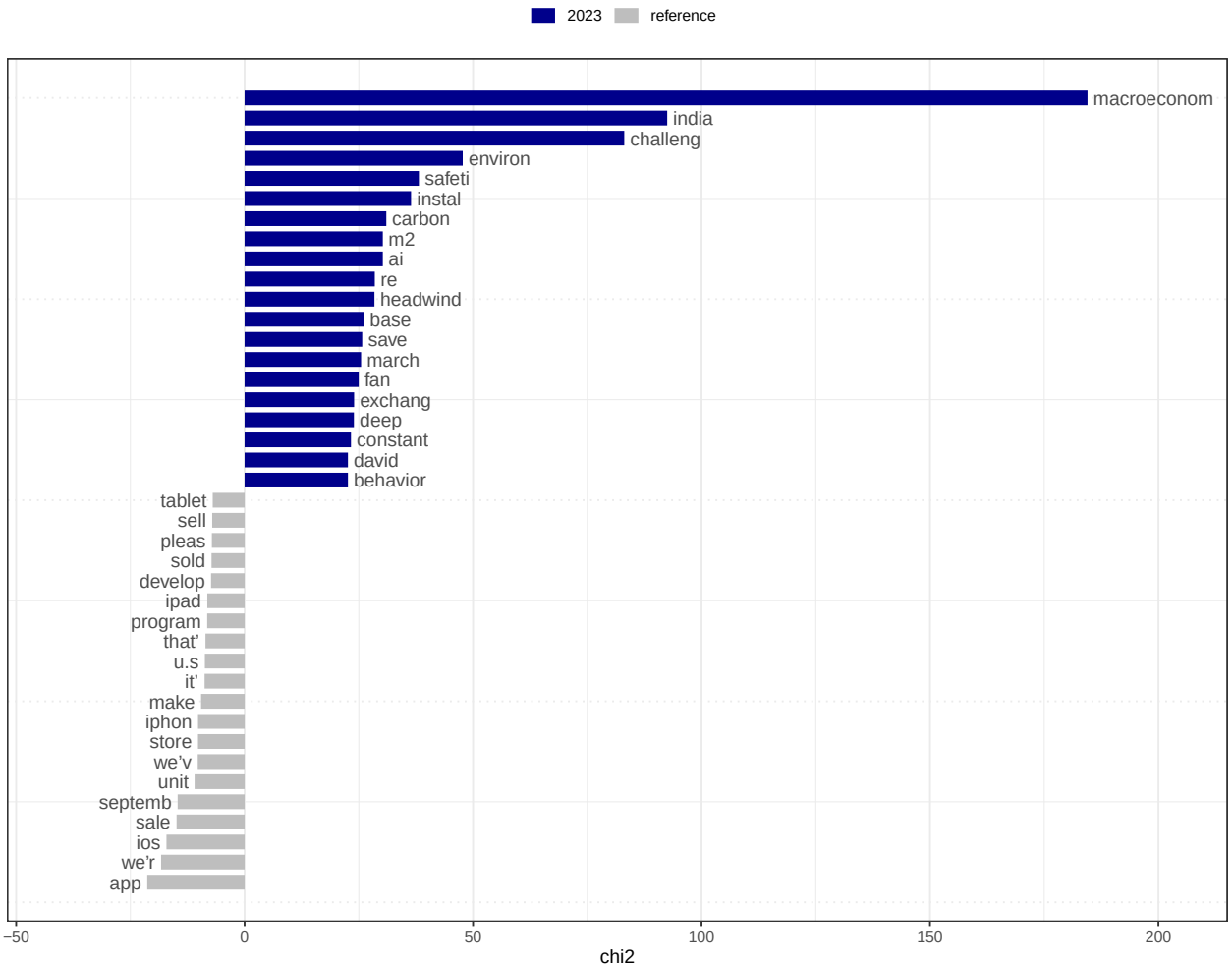
```
earn.call.dfm <- dfm_trim(earn.call.dfm, min_termfreq = 10, min_docfreq = 1)
topfeatures(earn.call.dfm, 20)
```

```
##  quarter    iphon     year    apple      new    think  billion   revenu
##     2907     1915     1699     1621     1431     1417     1380     1303
##    thank  product   servic   custom       re   market      can question
##     1301     1291     1257     1145     1112     1089     1065     1040
##     ipad      see       go     also
##      971      903      878      877
```

Keyness (2023 - 2021)

keyness of 2023

```
year_data <- dfm_group(earn.call.dfm, group = year)
keyness.2023 <- textstat_keyness(year_data, target = '2023')
keyness.2023 <- keyness.2023[ which(keyness.2023$p<=0.05), ]
textplot_keyness(keyness.2023)+ theme(legend.position = "top")
```



In 2023, the significantly mentioned word is "macroeconom", which can be inferred that Apple's profits are affected by the macroecnomics environment magnificently. Associated with the word "challenging" and "environment", we can infer that Apple faced challenge owing to macroeconomics environment. In fact, Tim Cook pointed out that the overall macroeconomic environment reduces the profits in both 2023 Q1 and Q2.

- https://www.cnbc.com/2023/02/02/apple-aapl-earnings-q1-2023.html
- https://tidbits.com/2023/05/05/apples-q2-2023-slightly-down-on-exchange-rates-and-macroeconomic-conditions/

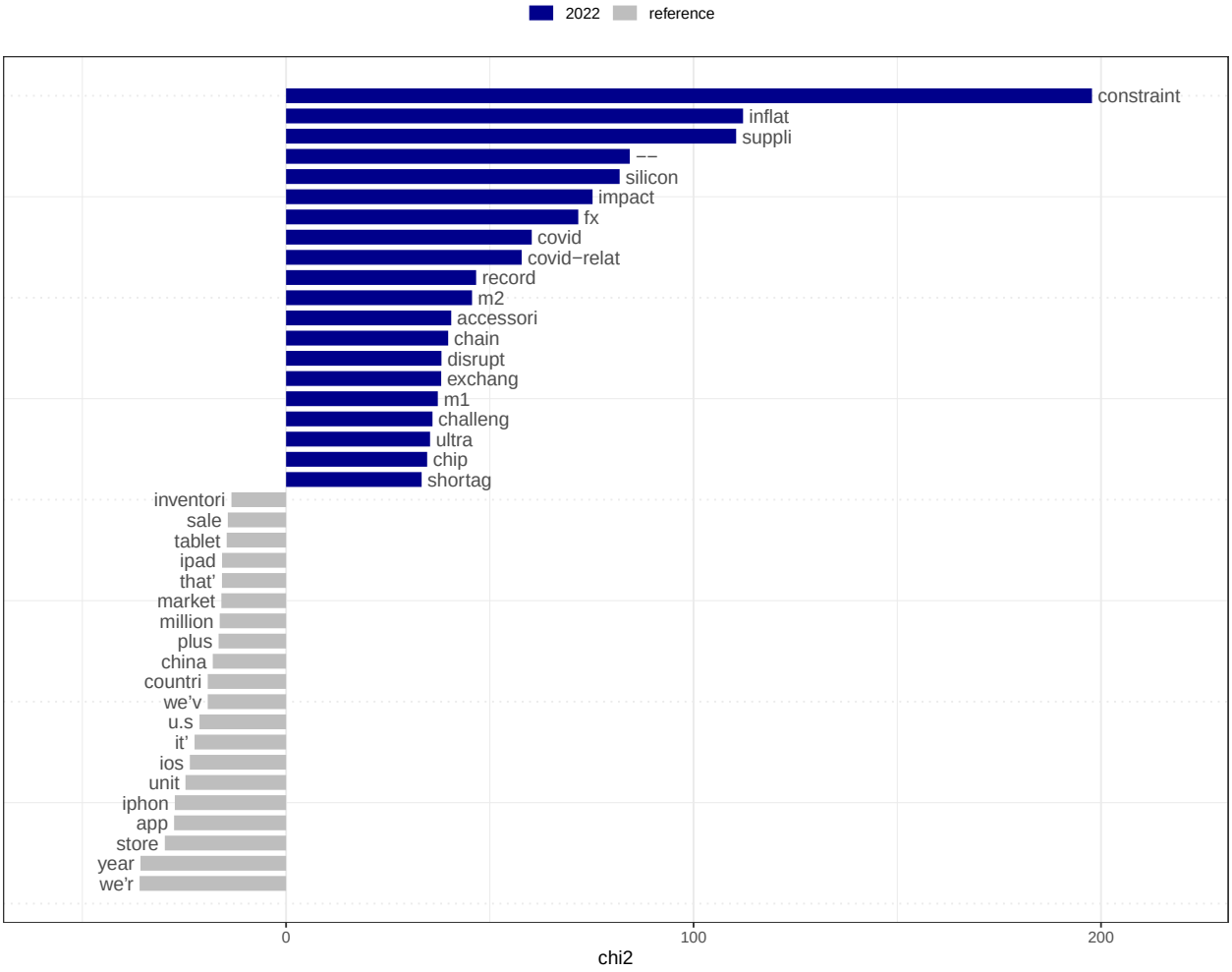
In addition, "india" was also mentioned a lot this year, which can be attributed to the fact that Indian population exceeded that of China in 2023 April. Apple mentioned that probably because they are planning to move their supply chains from China to India and increase the businesses in India.

- https://www.cnbc.com/2023/04/18/why-india-is-so-important-to-apple.html
- https://finance.yahoo.com/news/the-simple-reason-tech-thinks-india-is-its-next-big-market-200003052.html

"m2" is mentioned probably because Apple developed new type of chip (m2) and put emphasis on this new technology.

keyness of 2022

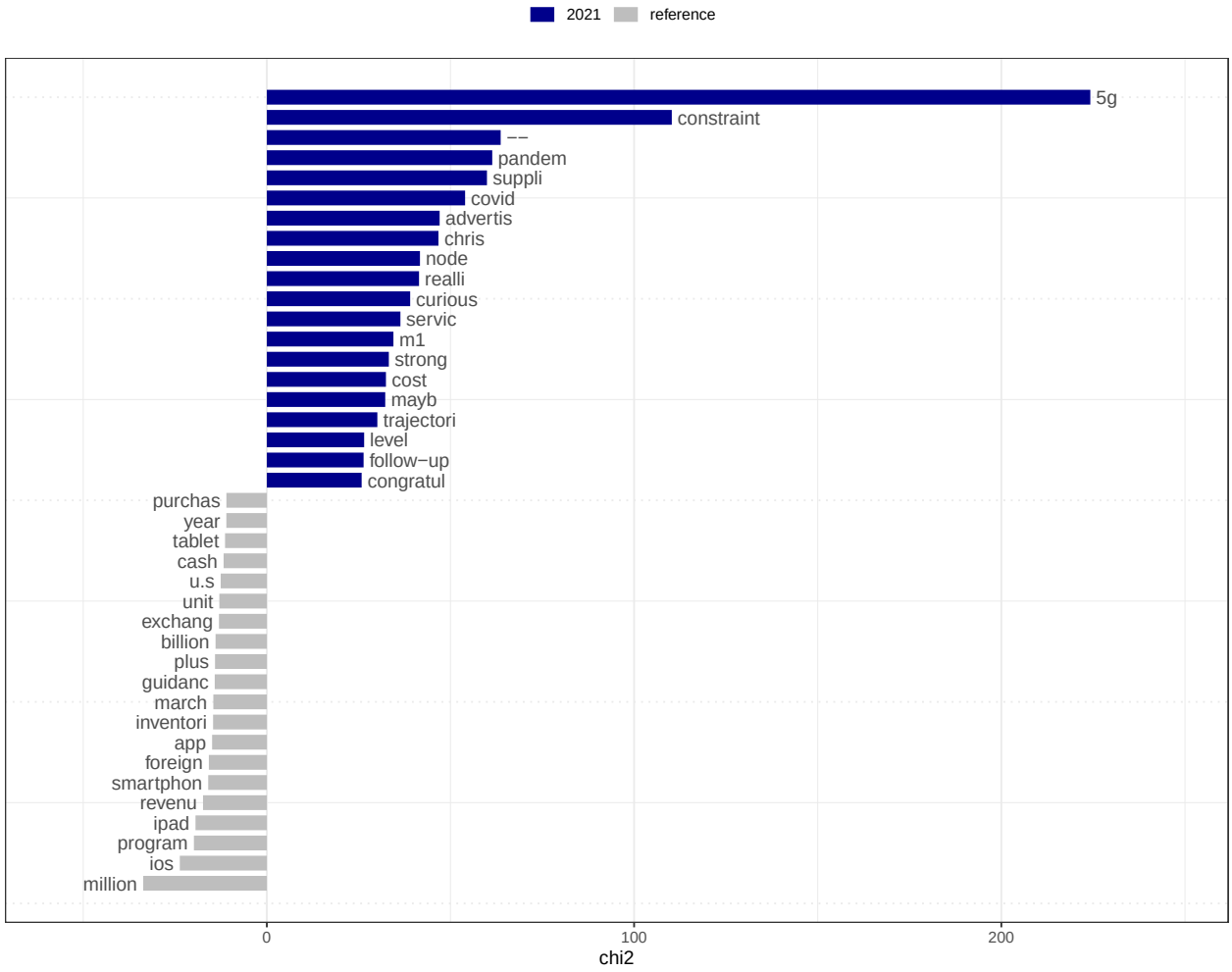
```
year_data <- dfm_group(earn.call.dfm, group = year)
keyness.2022 <- textstat_keyness(year_data, target = '2022')
keyness.2022 <- keyness.2022[ which(keyness.2022$p<=0.05), ]
textplot_keyness(keyness.2022) + theme(legend.position = "top")
```



In 2022, "constraint", "supply", "silicon", and "inflation" were the four most mentioned words. This year, the widespread Covid-19 omicron virus outbreak in Shanghai from February to August caused the strict lockdown in China. This implementation led to the disruption of supply chains of a wide range of businesses, and Apple is not the exception. The pandemic curtailed the productivity of chip and silicon, which caused the shortage on the materials for production.

keyness of 2021

```
year_data <- dfm_group(earn.call.dfm, group = year)
keyness.2021 <- textstat_keyness(year_data, target = '2021')
keyness.2021 <- keyness.2021[ which(keyness.2021$p<=0.05), ]
textplot_keyness(keyness.2021) + theme(legend.position = "top")
```



In 2021, “5g” became the most mentioned word in the earnings call.

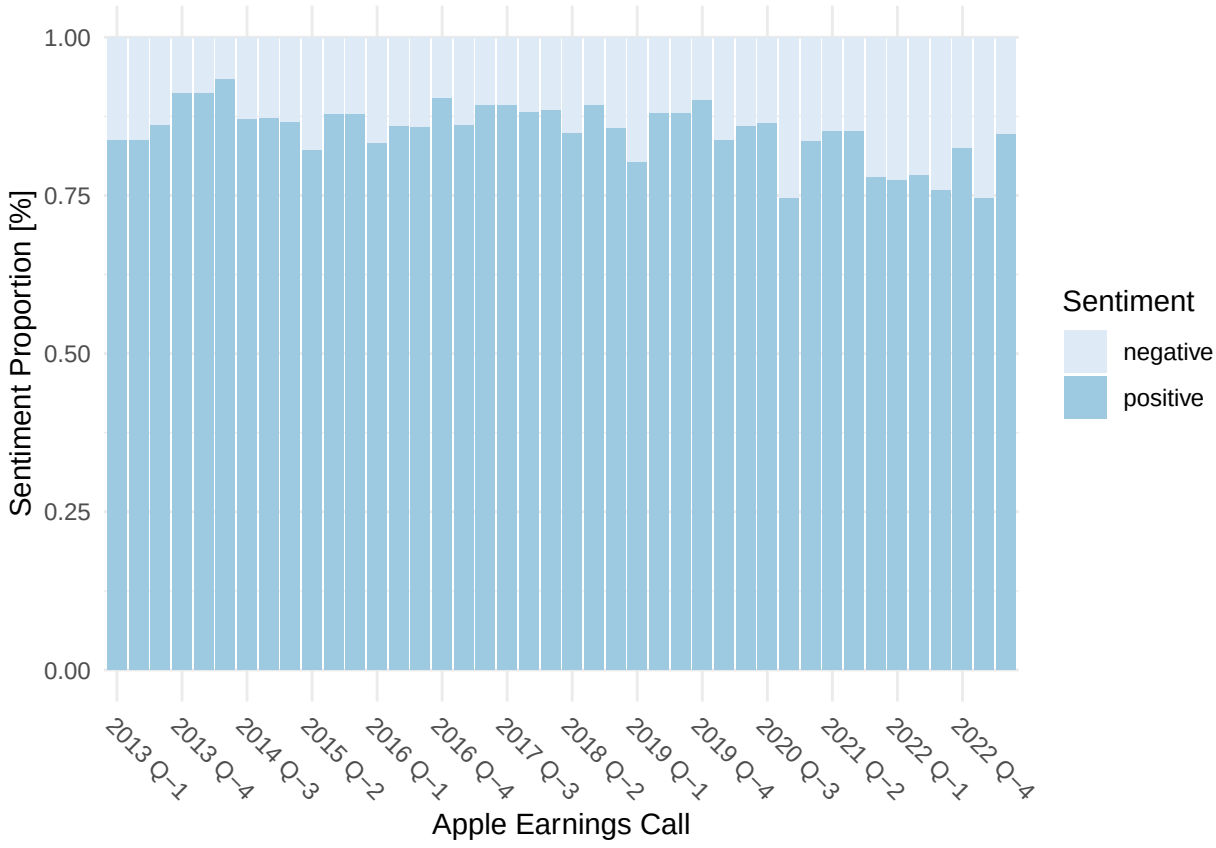
“Constraint”, “pandemic”, “supply”, and “covid” were mentioned a lot, which shows the Covid-19 impact on Apple supply chains.

Sentiment Analysis

To understand the difference across year and quarter, we conducted the sentiment analysis for the transcript each year.

```
earning.sentiment <- earn.call.dfm %>%
  dfm_lookup(dictionary = data_dictionary_LSD2015[1:2]) %>%
  dfm_weight(scheme = "prop") %>%
  convert("data.frame")
```

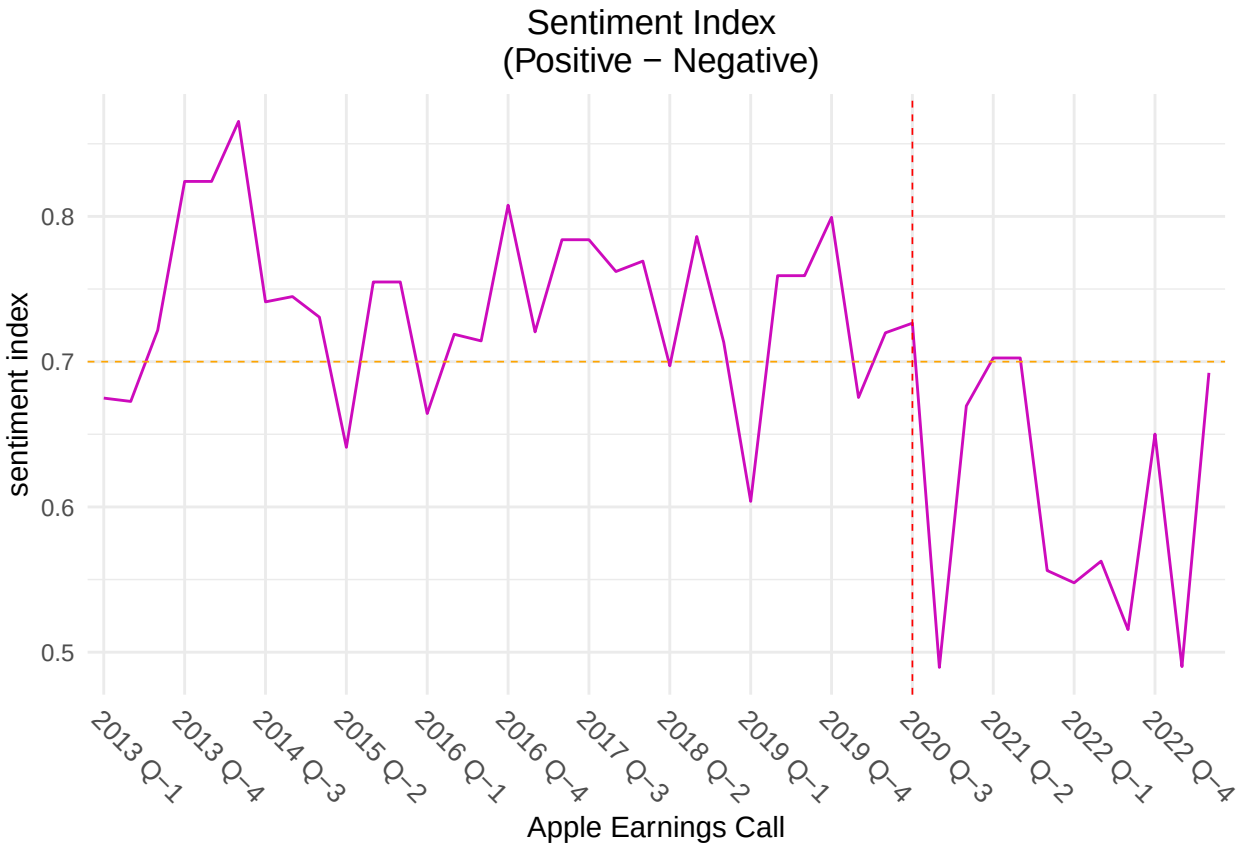
```
earning.sentiment$xlabel = paste(earn.call$year, earn.call$quarter)

earning.sentiment %>%
  pivot_longer(negative:positive, names_to = "Sentiment", values_to = "Proportion") %>%
  ggplot(aes(xlabel, Proportion, group = Sentiment, fill = Sentiment)) +
  geom_bar(stat = 'identity') +
  scale_colour_brewer(palette = "Set1") + scale_fill_brewer(palette = 1) +
  theme_minimal() +
  theme(axis.text.x = element_text(angle = -45, vjust = 1, hjust = 0)) +
  xlab("Apple Earnings Call") +
  ylab("Sentiment Proportion [%]") +
  scale_x_discrete(
    breaks =
      unique(earning.sentiment$
               xlabel[seq(1, length(unique(earning.sentiment$xlabel)), by = 3)]))
```



We can see that there is no significant volatility in the proportion of positive and negative words across time. However, the positive word proportion is relatively low after 2020 quarter 3.

```
# Sentiment Index (positive proportion - negative proportion)
earning.sentiment$emotion <- earning.sentiment$positive -
  earning.sentiment$negative
ggplot(earning.sentiment, mapping = aes(x = xlabel, y = emotion, group = 1)) +
   geom_line(colour = '003366') +
  theme_minimal() +
  theme(axis.text.x = element_text(angle = -45, vjust = 1, hjust = 0, size = 10),
        plot.title = element_text(hjust = 0.5)) +
  labs(x = "Apple Earnings Call", y = 'sentiment index',
       title = "Sentiment Index \n (Positive - Negative)") +
  scale_x_discrete(breaks = unique(earning.sentiment$xlabel[seq(1,length(unique(earning.sentiment$xlabel
  geom_hline(yintercept = 0.7, linetype = "dashed", color = 'orange', size = 0.3) +
  geom_vline(xintercept = 31, linetype = "dashed", color = 'red', size = 0.3)
```

```
## Warning: Using ‘size‘ aesthetic for lines was deprecated in ggplot2 3.4.0.
## i Please use ‘linewidth‘ instead.
## This warning is displayed once every 8 hours.
## Call ‘lifecycle::last_lifecycle_warnings()‘ to see where this warning was
## generated.
```



By defining the sentiment index by subtracting negative proportion from positive proportion, we can see that the sentiment index in the transcripts has a systematic decrease after 2020 Q3. Most points are above 0.7 before 2020 Q3, yet sentiment indexes after 2020 Q3 were lower than 0.7. This could be explained by the higher tendency to speak negative words as the result of tight supply under Covid-19.

Sentiment and AAPL stock price

We then further investigated if the sentiment index influenced Apple stock price. To answer this question, we cannot use Apple's stock price directly, as the stock price is affected by a series of variables, including the market stock price, economic indexes, and media emotion.

We apply Capital Asset Pricing Model (CAPM) to calculate the expected return of Apple, and subtract the expected return from the actual return to obtain the abnormal return. Lastly, we fit the linear regression model of abnormal return on the sentiment index to check whether Tim Cook's word emotion generates abnormal return on stock.

$$ER_{AAPL} = R_f + \beta_{AAPL}(ER_m - R_f)$$

The above equation shows the calculation of expected return for Apple stock.

We download the market return (ER_m) and risk free rate (R_f) from the following link:

- http://mba.tuck.dartmouth.edu/pages/faculty/ken.french/data_library.html#Research

Then, we regress Apple stock excess return ($R_{AAPL} - R_f$) on market risk free rate ($ER_m - R_f$) to get β_{AAPL}.

```
library('zoo')
library('xts')
library('TTR')
library('quantmod')
library('dplyr')
```

```
# Get the stock price of Apple Inc. We only need the close price.
# (The market rate data was only collected until 2023-04-28 so we chose this
# as the end date)
getSymbols("AAPL", from = '2013-01-01', to = '2023-04-28')
AAPL = AAPL$AAPL.Close

# Calculate the daily (actual) return
returns <- (diff(AAPL) / lag(AAPL, 1))*100
colnames(returns) <- c('AAPL.Ret')

# Import the data set of RM-RF data and RF(risk free return)
# Data Source:
# http://mba.tuck.dartmouth.edu/pages/faculty/ken.french/data_library.html#Research
MKT <- read.csv('Market_return_data/F-F_Research_Data_Factors_daily2.csv')
MKT$X <- as.Date(MKT$X, format = "%Y%m%d")

# started by 2013 because we only chose earnings call transcripts from 2013 Q1
MKT <- MKT[MKT$X >= as.Date('2013-01-01'),]
MKT <- MKT[MKT$X <= as.Date('2023-04-28'),]

# Because the last three rows are NA, so delete
last_three = (nrow(MKT) - 2):nrow(MKT)
MKT <- MKT[-last_three,]

# Combine the two dataframe
AAPL_MKT <- bind_cols(MKT,returns)
```

```r
# Calculate the excess return of AAPL (Return - Risk Free)
AAPL_MKT$excess.return <- AAPL_MKT$AAPL.Ret - AAPL_MKT$RF
```

```r
# Run the regression of AAPL excess return on Market excess return
# to get the beta of Apple stock
model <- lm(AAPL_MKT$excess.return ~ AAPL_MKT$Mkt.RF)
beta <- coef(model)[2]
beta
```

```
## AAPL_MKT$Mkt.RF
##        1.134838
```

```r
# According to CAMP, the normal (expected) return is given by RF + beta*(RM - RF)
AAPL_MKT$normal.return = AAPL_MKT$RF + beta * AAPL_MKT$Mkt.RF

# Abnormal return is calculated by Actual Return - Normal Return
AAPL_MKT$abnormal.return = AAPL_MKT$AAPL.Ret - AAPL_MKT$normal.return
```

We finally calculated the abnormal return and we can regress it on the sentiment index.

```r
# variables: to store the date variables of each earnings call
variables = attr(earn.call.dfm, 'docvars')
variables = variables[, -c(1:4)]
colnames(variables)
```

```
## [1] "year"    "quarter" "date"
```

```r
# The market is expected to respond after the date of earnings call
# (Because often, the earnings calls are held after the close time of stock market.)
variables$date = as.Date(variables$date) + 1

# Add the date after the earnings call to sentiment data for stock price analysis
earning.sentiment$date.after <- variables$date

# Define the emotion index as the difference between positive and negative
# proportion of a text
earning.sentiment$emotion <- earning.sentiment$positive -
  earning.sentiment$negative

# Fill in the abnormal return data on each earnings call date.

for (i in 1:42){
  if (earning.sentiment$date.after[i] %in% AAPL_MKT$X){
    earning.sentiment$abnormal.return[i] <-
      AAPL_MKT$abnormal.return[AAPL_MKT$X == earning.sentiment$date.after[i]]
  }
  else{
    earning.sentiment$abnormal.return[i] <- NA
  }
}


head(earning.sentiment, 5)
```

```
##                               doc_id   negative  positive   xlabel   emotion
## 1 transcript_2013_Q-1_2013-01-24.txt 0.16254417 0.8374558 2013 Q-1 0.6749117
## 2 transcript_2013_Q-2_2013-04-23.txt 0.16369048 0.8363095 2013 Q-2 0.6726190
## 3 transcript_2013_Q-3_2013-07-23.txt 0.13919414 0.8608059 2013 Q-3 0.7216117
## 4 transcript_2013_Q-4_2013-10-29.txt 0.08796296 0.9120370 2013 Q-4 0.8240741
## 5 transcript_2014_Q-1_2014-01-28.txt 0.08796296 0.9120370 2014 Q-1 0.8240741
##   date.after abnormal.return
## 1 2013-01-25     -3.02694139
## 2 2013-04-24     -0.23306913
## 3 2013-07-24      5.57874378
## 4 2013-10-30      2.28318843
## 5 2014-01-29      0.07902717
```

```
# Regress abnormal return on sentiment index.
emotional.regression <- lm(earning.sentiment$abnormal.return ~ earning.sentiment$emotion)
summary(emotional.regression)
```

```
##
## Call:
## lm(formula = earning.sentiment$abnormal.return ~ earning.sentiment$emotion)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -5.2046 -1.6001 -0.4824  1.0712  8.9108
##
## Coefficients:
##                            Estimate Std. Error t value Pr(>|t|)
## (Intercept)                  -1.720      3.485  -0.494    0.625
## earning.sentiment$emotion     3.558      4.946   0.719    0.477
##
## Residual standard error: 2.821 on 35 degrees of freedom
##   (5 observations deleted due to missingness)
## Multiple R-squared:  0.01457,    Adjusted R-squared:  -0.01359
## F-statistic: 0.5174 on 1 and 35 DF,  p-value: 0.4767
```

```
library(ggplot2)
library(ggpmisc)
```

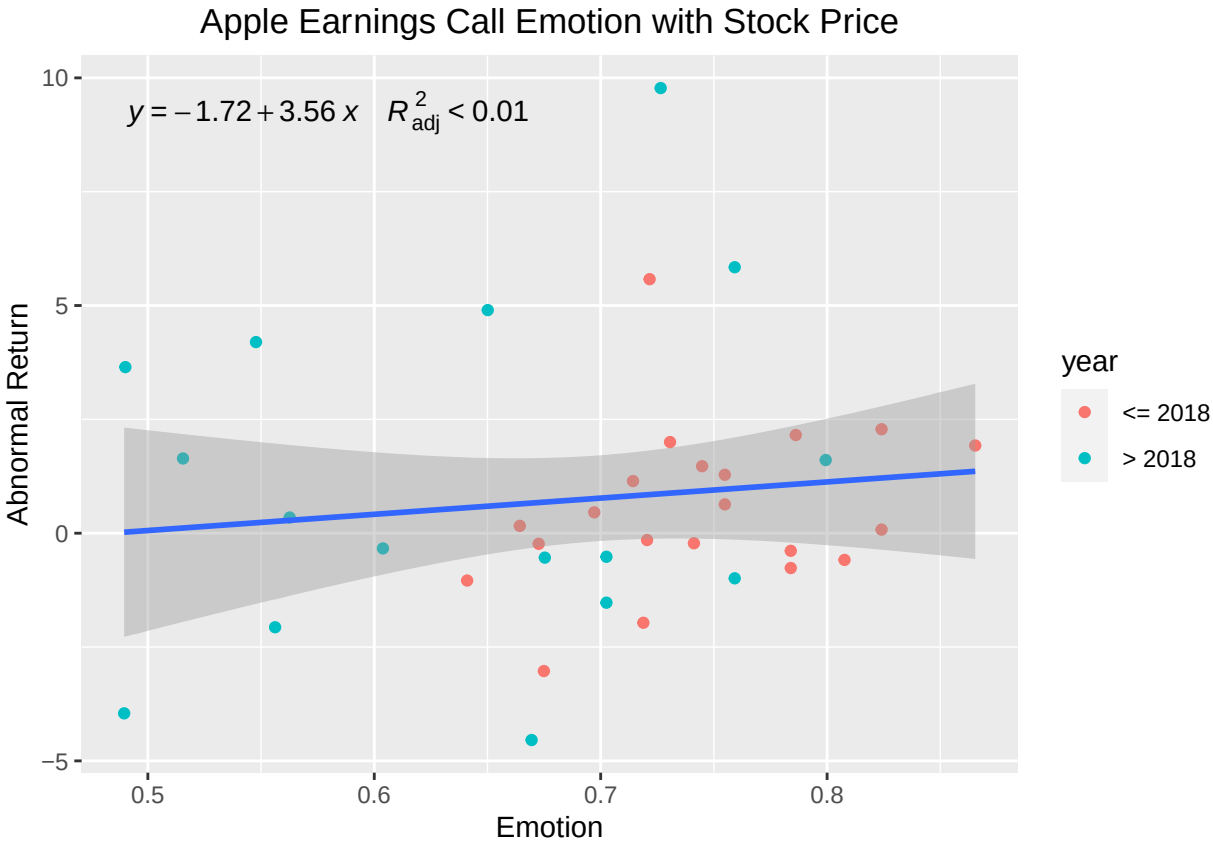
```
## Loading required package: ggpp
```

```
##
## Attaching package: 'ggpp'
```

```
## The following object is masked from 'package:ggplot2':
##
##     annotate
```

```
## Registered S3 method overwritten by 'ggpmisc':
##   method                  from
##   as.character.polynomial polynom
```

```
earning.sentiment$year = earn.call$year
earning.sentiment$year = ifelse(earn.call$year <= 2018, "<= 2018", "> 2018")

ggplot(earning.sentiment, aes(x = emotion, y = abnormal.return)) +
  geom_point(aes(color = year)) + geom_smooth(method = 'lm', formula = y ~ x) +
  stat_poly_eq(aes(label = paste(..eq.label..,
                                 ..adj.rr.label..,
                                 sep = '~~~~')),
               formula = y ~ x) +
  labs(title = "Apple Earnings Call Emotion with Stock Price",
       x = "Emotion",
       y = "Abnormal Return") +
  theme(plot.title = element_text(hjust = 0.5))
```



According to the regression table, Tim Cook's speaking emotion has no significant impact on the Apple abnormal return, as the p value ($Pr(>|t|)$) is greater than 0.05.

In addition, the Multiple R-squared is only 0.01457, which means that the model has extremely restricted ability to explain the relationship between independent variables and dependent variable. This might be caused by omitted variable bias. For instance, media's prospective toward the new launched products might have significant impact on the abnormal return, but we did not put it into our model.

In addition, we can see that the abnormal return sample variance is larger after 2018 than that before 2018.

References

- https://www.youtube.com/watch?v=ucKK528ApCw
- https://www.youtube.com/watch?v=PFMCx7TDBr4
- http://mba.tuck.dartmouth.edu/pages/faculty/ken.french/data_library.html#Research