TWITTER ANALYSIS

FINAL PROJECT
DATA ANALYTICS

BY RENATO AND CHRISTIAN

OBJECTIVE

- To analyze Twitter data in order to provide information for marketers. The project consists in to connect to the Twitter API using R and Rstudio. Once connected, the script is set up to pulling out data from @samsungus, @lgus, @moto and @sonyxperia. Data from tweeter timeline and the search engine will be stored in CSV files.

COMPANIES

- Samsung - https://twitter.com/samsungmobileus
- LG - https://twitter.com/LGUS
- Moto - https://twitter.com/moto
- Sony - https://twitter.com/sonyxperia

- It was chosen only mobile accounts from the USA.

PACKAGES USED

- twitteR
- Sentimentr
- Maps
- sentR
- Httpuv
- Mapproj
- data.table
- Ggmap
- Stringr
- Devtools
- Rjson
- RSQLite
- Httr
- ggedit
- RJSONIO
- Tidytext
- Broom
- Scales
- Stringi
- Ggpubr
- Ggplot2

TWITTER APP: HTTPS://APPS.TWITTER.COM

https://apps.twitter.com/app/13782765/show

iCloud | FGTS | Itaú Personnalité | UOL | Despesas | Cacula Adm | Facebook | Outlook.com | Gmail | globo.com | Bradesco Saúde

Application Management

ProjectR

Test OAuth

Details | Settings | Keys and Access Tokens | Permissions

My project in R for UCSC

http://www.roschel.net

Organization

Information about the organization or company associated with your application. This information is optional.

Organization	None
Organization website	None

Application Settings

Your application's Consumer Key and Secret are used to authenticate requests to the Twitter Platform.

Access level	Read and write (modify app permissions)
Consumer Key (API Key)	(manage keys and access tokens)
Callback URL	http://127.0.0.1:1410

DATA SOURCE

- Get information by account:
  getUser("samsungmobileus")
- Get n timeline posts:
  userTimeline('samsungmobileus', n=n_timelines)
- Get n company mentions by users:
  searchTwitter("@samsungmobileus", n=n_timelines)

We will use n=500 for the demonstration. But, in the real world, we could get more data and stored the result.

CLEANING UP DATA FOR MAP PROJECTION – PART 1

Step 1 - Removing users without location:

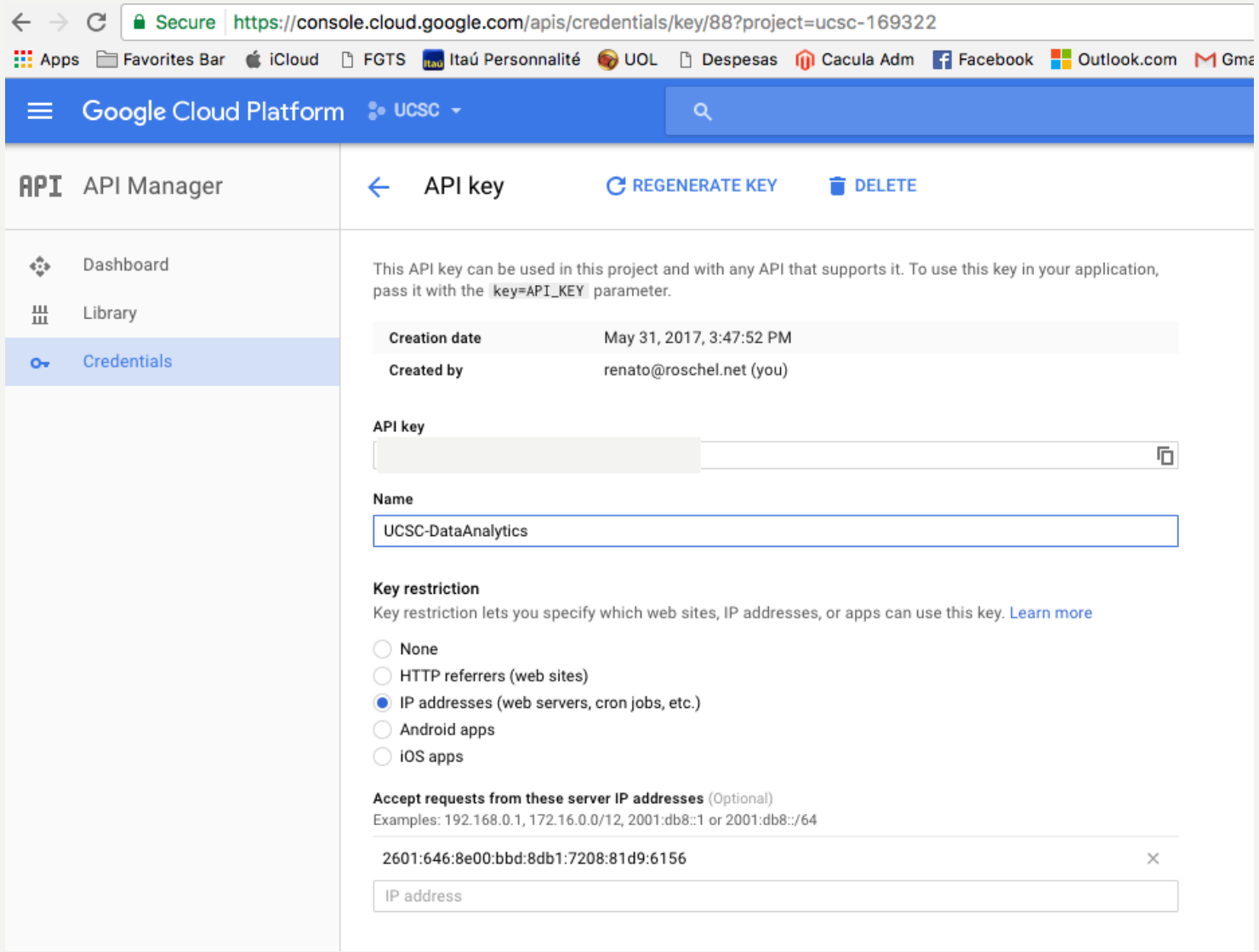
```
total.mentions_c1<-subset(total.mentions_c1, location!="");
```

Step 2 - Removing special characters (Our own function):

```
remove_special_chars("400 ☁ East Baltimore ");
total.mentions_location <- sapply(total.mentions_c2$location,remove_special_chars,simplify = F);
```

GEOCODING USING GOOGLE MAPS API

HTTPS://CONSOLE.CLOUD.GOOGLE.COM/APIS/CREDENTIALS?PROJECT=UCSC-169322



1) Be aware. Google IPI provides only 2,500 search times per day for free.

2) In this project we geocoded 200 address per company, 600 total.

3) In order to use the function geocode(), we needed to modify the code and apply our API key on it. This will allow us to use our quote of 2,500 geocoding times per day. Therefore, the code was downloaded and edited line 174

4) geocode_results <- sapply(mentions$location, geocode_apply, simplify = F)

CLEANING UP DATA FOR MAP PROJECTION – PART 2

- Step 1 - Eliminating results that returns status != OK:

```
condition_c1     <- sapply(samsung.geocode_results, function(x) x["status"]=="OK");
geocode_results <- samsung.geocode_results[condition_c1];
```

- Step 2 - Eliminating results that we don't know exactly where is:

```
condition_c2 <- lapply(samsung.geocode_results, lapply, length);
condition_c2a<- sapply(condition_c2, function(x) x["results"]=="1");
geocode_results<-samsung.geocode_results[condition_c2a];
```

- Step 3 - Script to cleaning up geocoding:

```
geocode_results <- clean_geocoding_results(geocode_results);
```

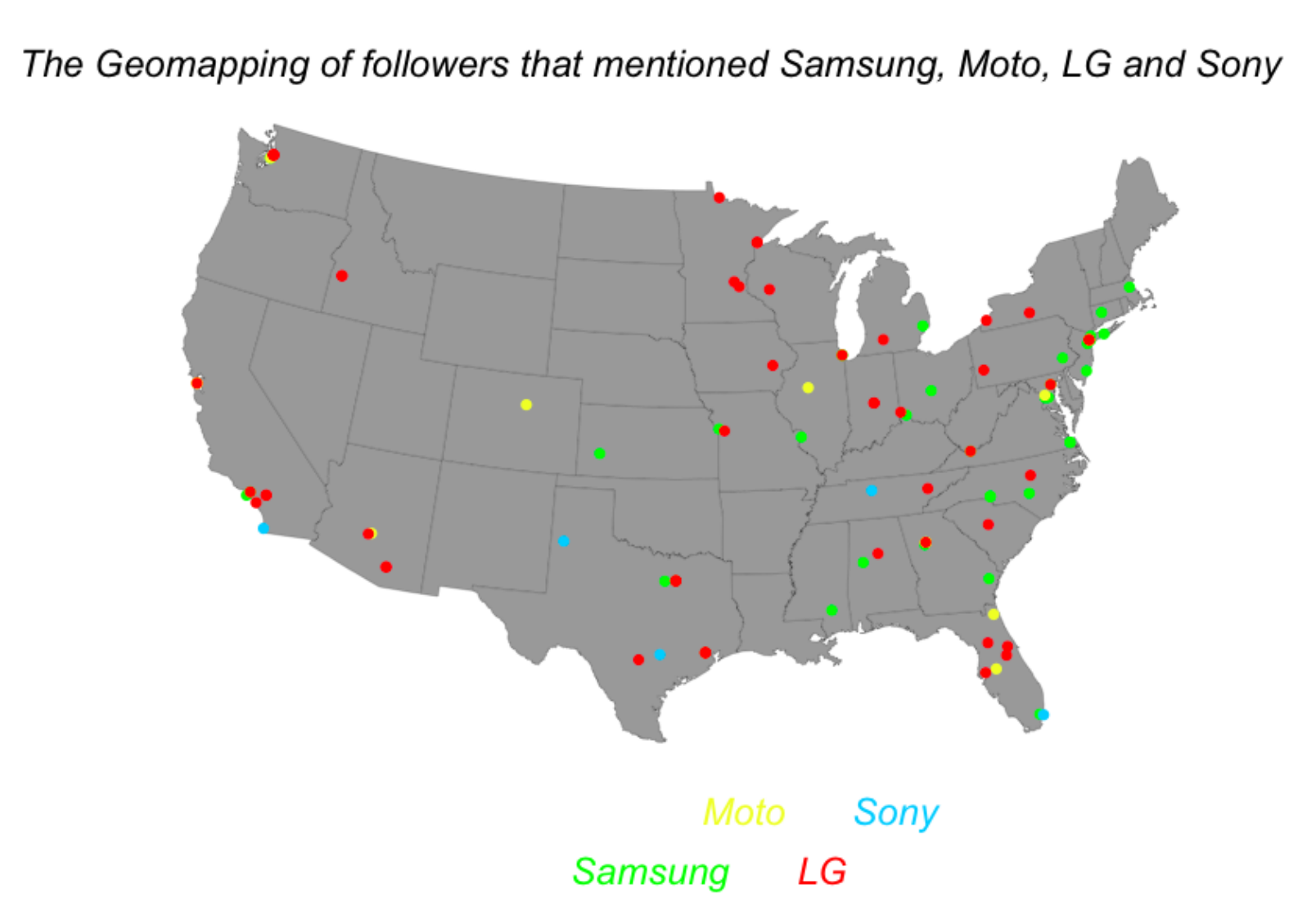
- Step 4 - Removing users from outiside of the USA:

```
american_results<-subset(results_f, grepl(", USA", results_f$Location)==TRUE);
```

- Step 5 – Removing results with one comma or less:

```
american_results$commas<-sapply(american_results$Location, function(x) length(as.numeric(gregexpr(",", as.character(x))[[1]])));
# leave just address with two commas
american_results<-subset(american_results, commas==2);
#Drop the "commas" column:
american_results<-subset(american_results, select=-commas);
```

GEOMAPPING




```
map_projection <-
map("state", proj="albers",
param=c(39, 45),
col="#999999",
fill=TRUE, bg=NA,
lwd=0.2, add=FALSE,
resolution=1)

points(mapproject(samsung
.american_results$lng,
samsung.american_results$
lat), col=NA, bg="#2EFE2E",
pch=21, cex=1.0)

mtext("Samsung", side =
1, line = -2, outer = T,
cex=1.5, font=3,
col="#2EFE2E")
```

SENTIMENTAL ANALYSIS – WORDS LIBRARY

- # Getting words with positive conotation:
  positive <- get_sentiments("bing");
  positive <- subset(positive,sentiment=="positive");
  positive <- positive$word;
- # Getting words with negative conotation:
  negative <- get_sentiments("bing");
  negative <- subset(negative,sentiment=="negative");
  negative <- negative$word;

SENTIMENTAL ANALYSIS – READING DATA FROM MENTIONS

- # read csv file with the storaged mentions:

```
mydata = read.csv("Project/twitter-data-mentions.csv");
```

- # Analyzing data that was not retweet, eliminating RT's:

```
mydata <- subset(mydata,mydata$isRetweet==FALSE);
```

- # Assign only text column, tweets:

```
test <- mydata$text;
```

SENTIMENTAL ANALYSIS – ANALYSING AND STORING

- # 1. Simple Summation:

```
out.aggregate <- classify.aggregate(test, positive, negative);
```

- # 2. Naive Bayes:

```
out.naivebayes <- classify.naivebayes(test);

out.naivebayesmydata$POS      <- out.naivebayes[,1];
mydata$NEG      <- out.naivebayes[,2];
mydata$POS_NEG <- out.naivebayes[,3];
mydata$SENT     <- out.naivebayes[,4];

path.csv <- "twitter-data-sentimental.csv”;
write.csv(mydata , file = path.csv);
```

SENTIMENTAL ANALYSIS – EXAMPLES

text	SENT
@SamsungMobileUS Love this camera / phone. https://t.co/tTBtFKliGG	positive
I'm looking forward to the release of the @SamsungMobileUS #Note8	positive
@SamsungMobileUS It's been a fantastic 2 months í ½í¸ I'm probably keeping this phone for years	positive
@SamsungMobileUS its not just good,its the best phone ive ever had	positive
@SamsungMobileUS your phones are amazing	positive
@SamsungMobileUS thank your much for your great customer service !!!! I love my new gear 2!!!	positive

text	SENT
@SamsungMobileUS Ship as promised? Instead 2-3wks, tell me after I pay. Cancel? No can, already on the mail pony. Jâ€¦ https://t.co/fwDIrYiPvG	negative
@SamsungSupport We ell I responded 2 days ago and still no service #FAIL @SamsungMobileUS @SamsungUS @SamsungMobile	negative
@SamsungMobileUS Your new phones are useless without an SD Card slot and Removable batteries! When the phone freezeâ€¦ https://t.co/dfvsgnsXqr	negative
damn @SamsungMobileUS can we get a variety on family emoji's... https://t.co/1B6ir2XnEq	negative
@bmac0823 @SamsungMobileUS That and they falsely advertised the " full screen " for movies, pics, videos ect... nonâ€¦ https://t.co/gBBvxKFsgR	negative
No, @SamsungMobileUS despite your new commercials, I'm not getting my hand burned again!!	negative

OUTPUTS

- SENTIMENTAL ANALYSIS ON TWEETS
  # twitter-data-sentimental.csv

- ALL DATA ABOUT MENTIONS
  # twitter-data-mentions.csv

- ALL DATA ABOUT TIMELINES
  # twitter-data-timeline.csv

- ALL HEAD DATA SUCH AS likes, followers, following, tweets
  # twitter-data.csv

DATA ANALYSIS

- #1. with the data outputs generated, we proceed to read the csv files and create the dataframes in R.

```
#read csv file from the directory
df.twitter.temp <- read.csv(
file="twitter-data.csv", header=TRUE, sep=",")
```

we repeat this function changing only the name of the file

- SENTIMENTAL ANALYSIS ON TWEETS
  # twitter-data-sentimental.csv 24 features, 2878 observations
- ALL DATA ABOUT MENTIONS
  # twitter-data-mentions.csv 20 features, 8370 observations
- ALL DATA ABOUT TIMELINES
  # twitter-data-timeline.csv 19 features, 8381 observations
- ALL HEAD DATA SUCH AS likes, followers, following, tweets
  # twitter-data.csv

DATA ANALYSIS

- #2. with the data frames loaded we check the raw data, looking for data patrons or data inconsistencies
-

```
> str(df.twitter.data.timeline.temp)
'data.frame':   8381 obs. of  20 variables:
 $ X              : int  1 2 3 4 5 6 7 8 9 10 ...
 $ text           : Factor w/ 8341 levels "\"In-Traffic Replyâ€ allows drivers to stop texting while
 driving. https://t.co/7OlfJ9MWj9",..: 871 7857 1267 4383 2591 7277 1201 1259 3361 38 ...
 $ favorited      : logi  FALSE FALSE FALSE FALSE FALSE FALSE ...
 $ favoriteCount  : int  1 1 0 1 0 0 0 2 2 0 ...
 $ replyToSN      : Factor w/ 5606 levels "___Kees","__DannyMartinez",..: 623 5583 901 3142 1856 515
3 848 893 2407 10 ...
 $ created        : Factor w/ 8378 levels "2016-03-15 16:44:36",..: 8375 8371 8370 8369 8368 8367 83
66 8348 8347 8346 ...
 $ truncated      : logi  FALSE FALSE FALSE FALSE FALSE FALSE ...
 $ replyToSID     : num  8.73e+17 8.73e+17 8.73e+17 8.73e+17 8.73e+17 ...
 $ id             : num  8.73e+17 8.73e+17 8.73e+17 8.73e+17 8.73e+17 ...
 $ replyToUID     : num  1.54e+09 7.69e+17 2.12e+08 2.66e+09 4.95e+07 ...
 $ statusSource   : Factor w/ 8 levels "<a href=\"http://twitter.com\" rel=\"nofollow\">Twitter Web
Client</a>",..: 5 5 5 5 5 5 5 5 5 5 ...
 $ screenName     : Factor w/ 4 levels "LGUS","Moto",..: 3 3 3 3 3 3 3 3 3 3 ...
 $ retweetCount   : int  1 0 0 1 0 0 0 0 0 0 ...
 $ isRetweet      : logi  FALSE FALSE FALSE FALSE FALSE FALSE ...
 $ retweeted      : logi  FALSE FALSE FALSE FALSE FALSE FALSE ...
 $ longitude      : logi  NA NA NA NA NA NA ...
 $ latitude       : logi  NA NA NA NA NA NA ...
 $ location       : Factor w/ 4 levels "","Chicago, IL",..: 4 4 4 4 4 4 4 4 4 4 ...
 $ language       : Factor w/ 1 level "en": 1 1 1 1 1 1 1 1 1 1 ...
 $ profileImageURL: Factor w/ 4 levels "http://pbs.twimg.com/profile_images/440495007212912640/pdP9P
3iK_normal.jpeg",..: 3 3 3 3 3 3 3 3 3 3 ...
> |
```

CLEANING UP DATA

- #3 some Data Types were loaded with default datatype format by R, so we procede to re-format some features.
  - The column "created" was loaded as a factor , and we want date format.

```
$ created : Factor w/ 8378 levels "2016-03-15 16:44:36"
```

- Giving right format to dataframe using as.Date,
-

```
myDate <- as.Date(df.twitter.data.timeline$created)
df.twitter.data.timeline[["created"]] <- myDate
```

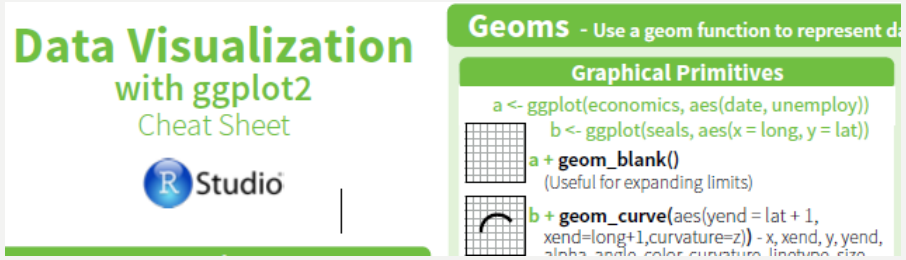
```
$ created : Date, format: "2017-06-07" "2017-06-07" "2017-06-07" ..
```

DATA ANALYSIS - PLOTTING

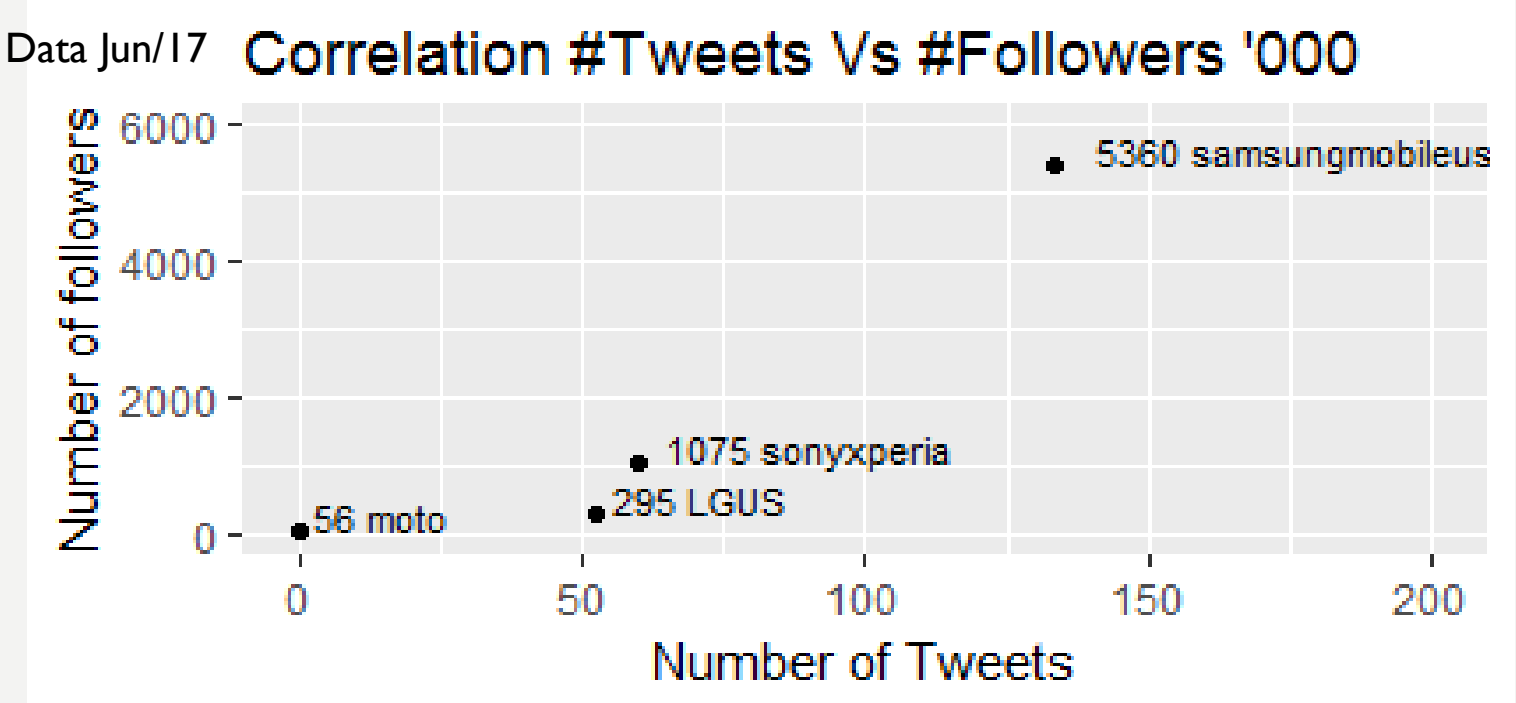
- Check the loaded data and understand the variables.
- Create the blueprint what graphics we want to implement

  - Tweets Vs followers
  - followers per company and people following company
  - Posts , frequency, activity, how was using Twitter the company (brand post or costumer service)
  - Timeline: Posts and Replies
  - Timeline: Engagement favorite tweets (Interactions) (replyes and

- Hint: Using ggplot2 Cheat Sheet as your best friend.

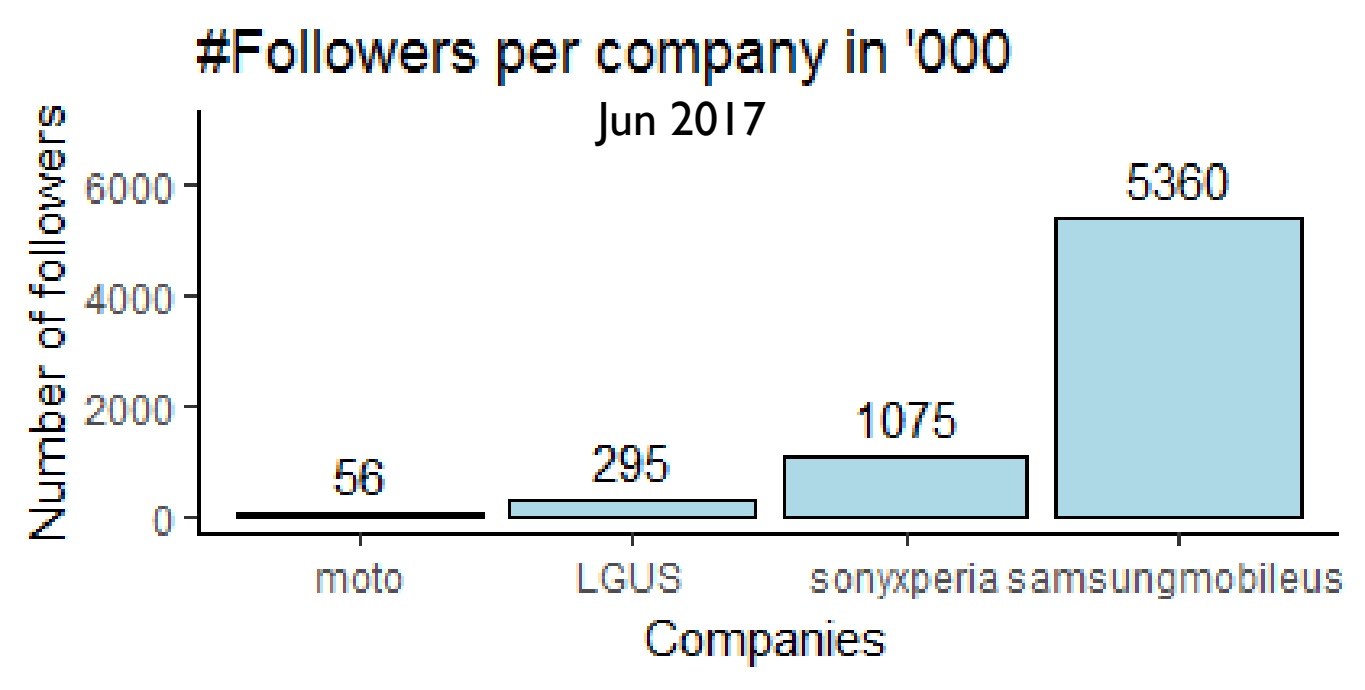


DATA ANALYSIS - PLOTTING

- 

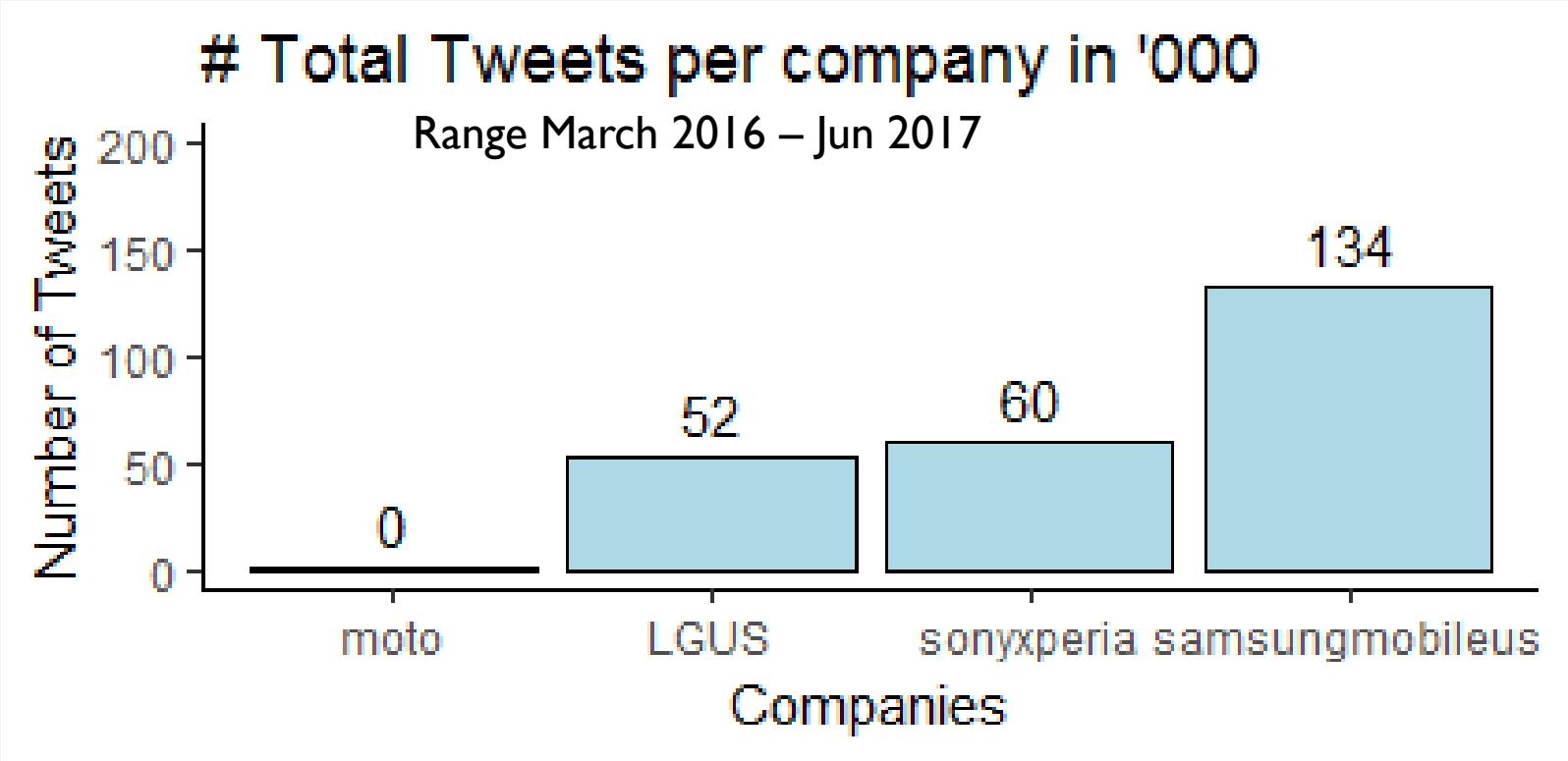

```
dots.chart.tweetsVSfollowers <- ggplot(df.twitter, aes(x = (df.twitter$t.tweets/1000), y
=(df.twitter$t.followers/1000), + label = paste( round(df.twitter$t.followers/1000, digits =
0), df.twitter$t.account))) + + geom_point() + geom_text(size=3, hjust = -0.1, vjust = 0) +
+ labs( x = "Number of Tweets", y = "Number of followers", title ="Correlation #Tweets Vs
#Followers '000") + + xlim(0, 200) + ylim(0,6000)+ theme_gray()
```

DATA ANALYSIS - PLOTTING

- 

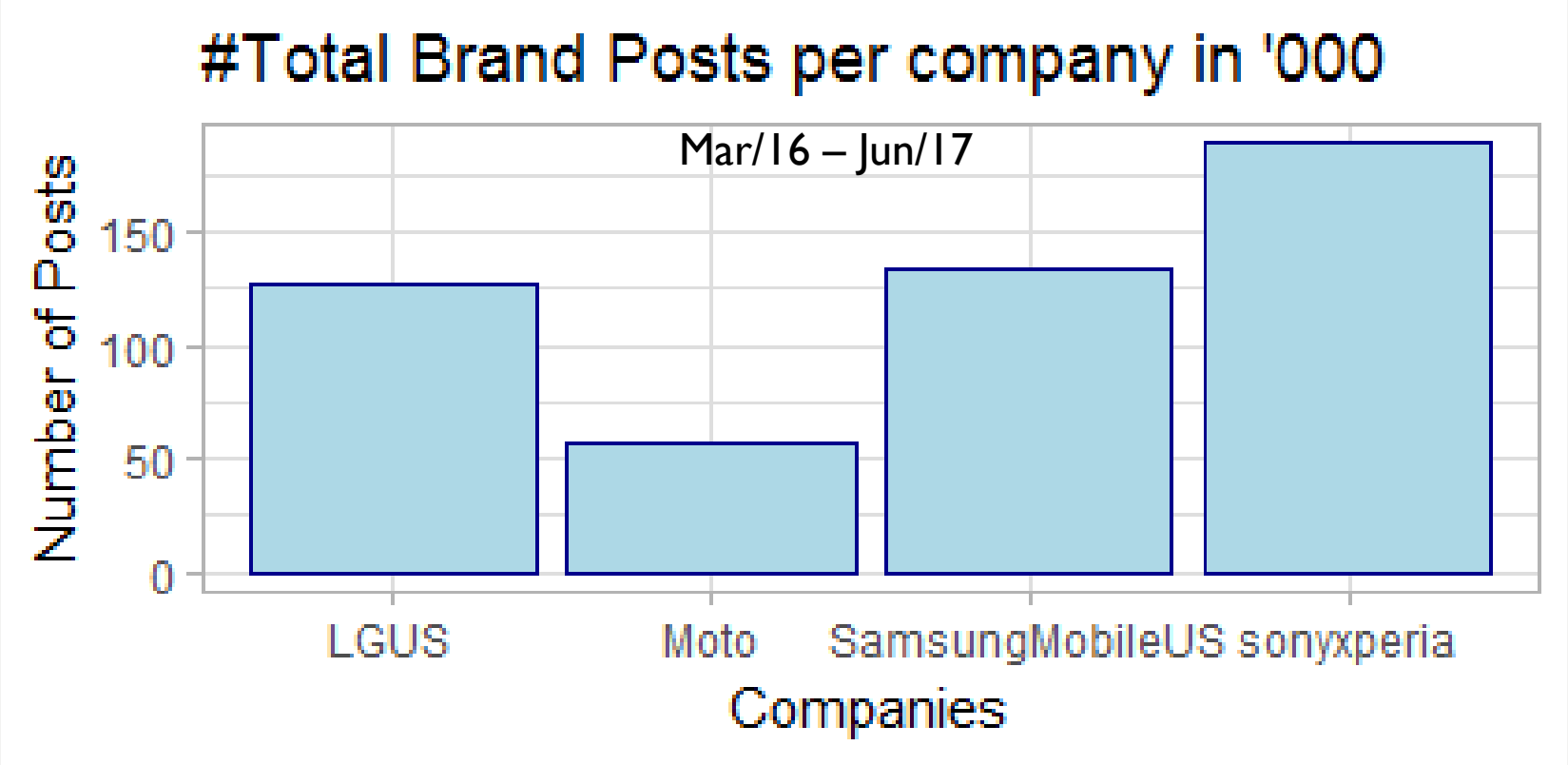

```
> bar.chart.xCompanies.yFollowers <- ggplot(df.twitter,
aes(x=reorder(t.account,t.followers), y=(t.followers/1000)), + df.twitter$t.account) +
geom_bar(stat="identity",fill="lightblue", colour="black")+ + labs( x = "Companies", y =
"Number of followers", title ="#Followers per company in '000")+ + ylim(0,7000)+ +
geom_text(aes(label=paste(round(df.twitter$t.followers/1000, digits = 0)), vjust=-0.5))
```

DATA ANALYSIS - PLOTTING

- 

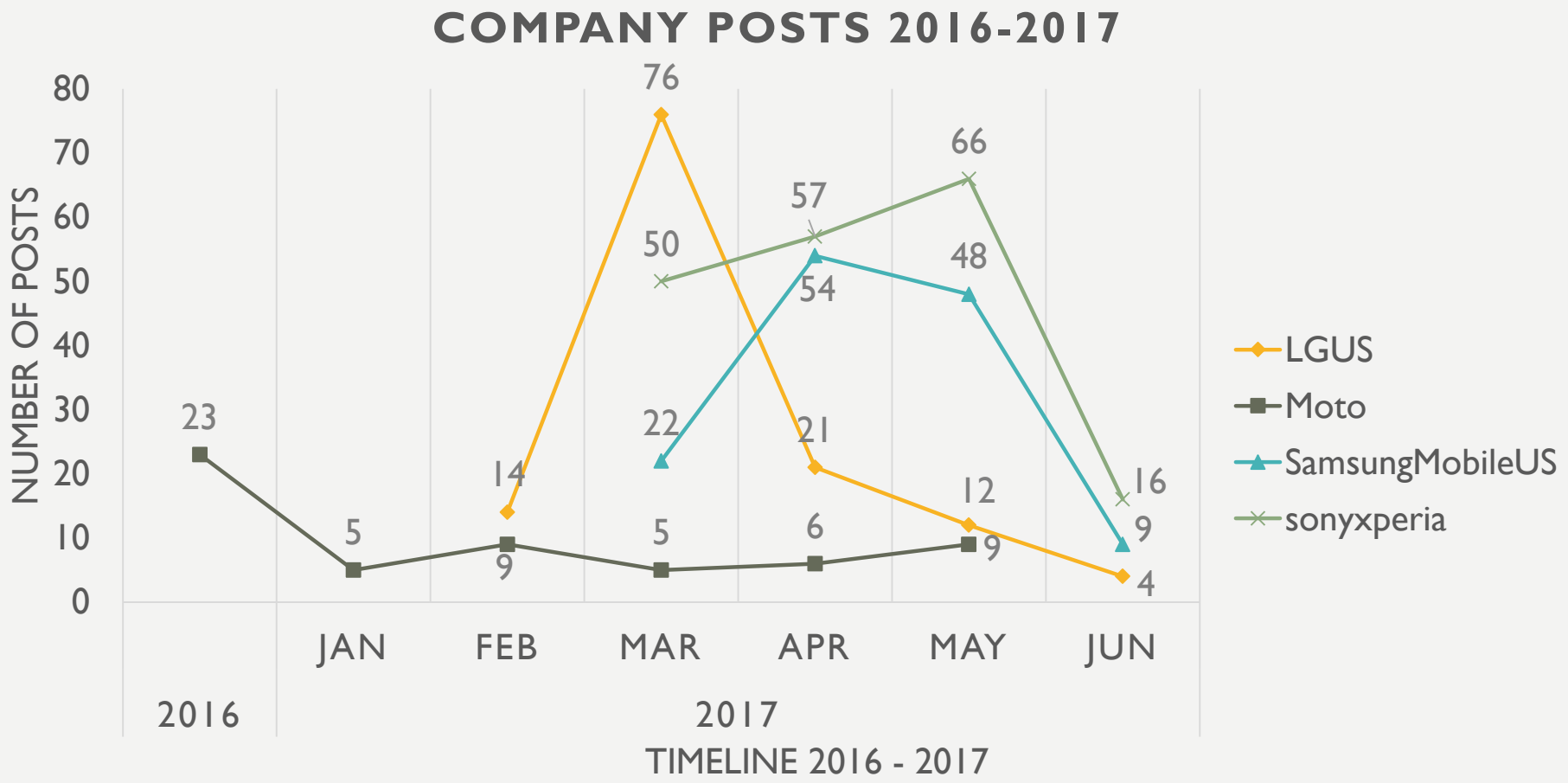
Companies use their accounts to promote products as and advertisement platform

DATA ANALYSIS - PLOTTING

- 

Of the total posts, we classify tweets and separate the own company posts . not including replies to users

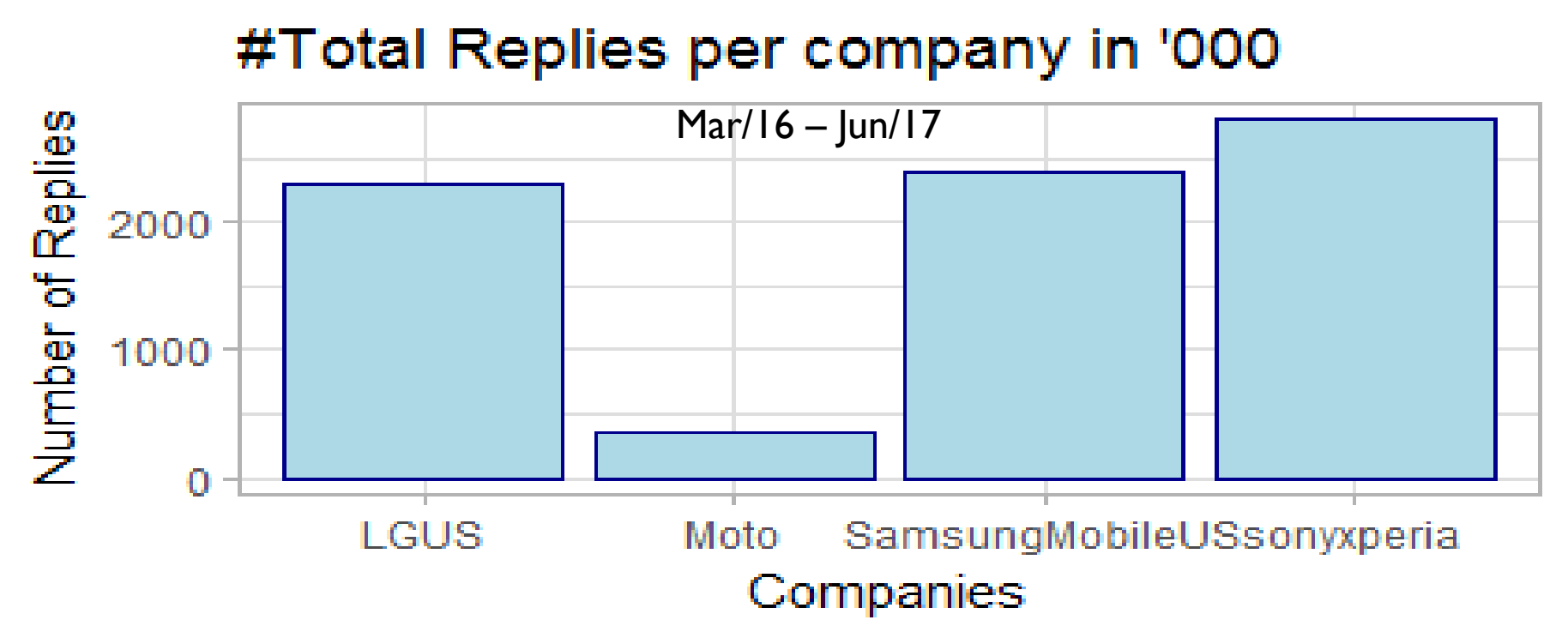
DATA ANALYSIS - PLOTTING



we identified that the peak post days, correspond to big success such Samsung in between April and March, Samsung S8 release

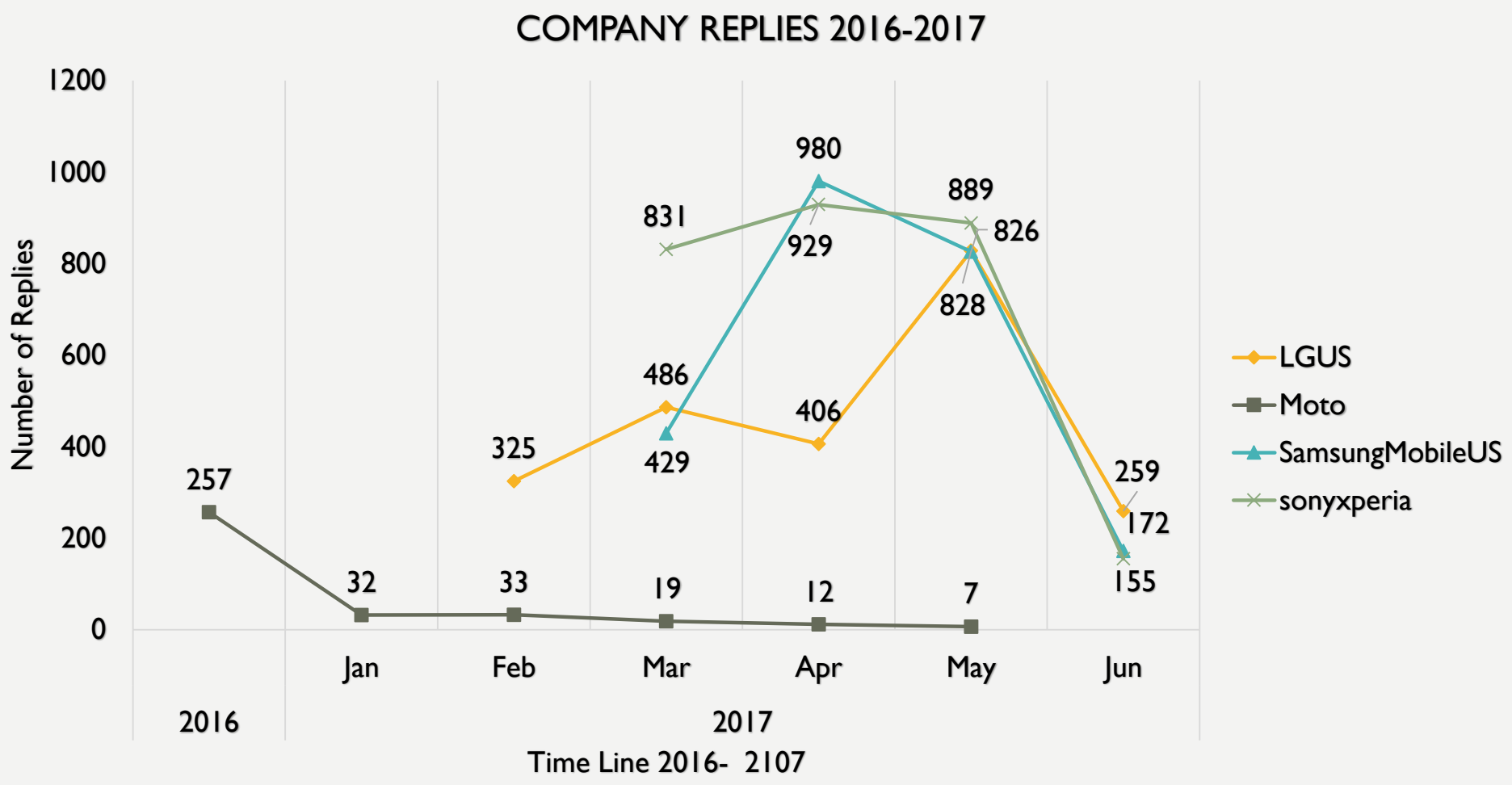
Pre-order the Galaxy S8 or S8+ by 4/20, and get the new Gear VR with Controller for free. https://t.co/toQkBi06VD
See it. Translate it. Galaxy S8. https://t.co/BrAAwYmhhY https://t.co/mBAVFXXflQ

DATA ANALYSIS - PLOTTING

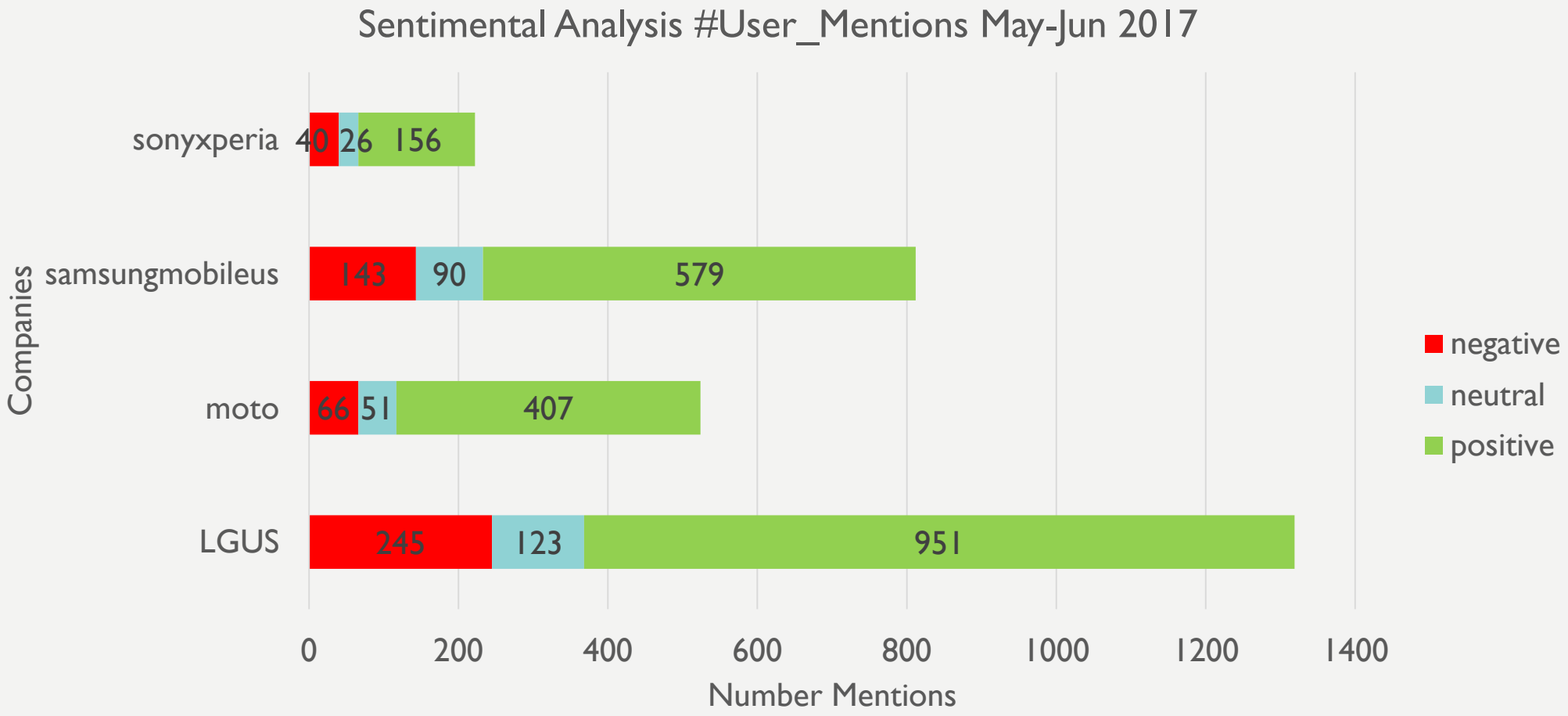


- Companies use Tweeter to interact with users, as customer service platform

DATA ANALYSIS - PLOTTING

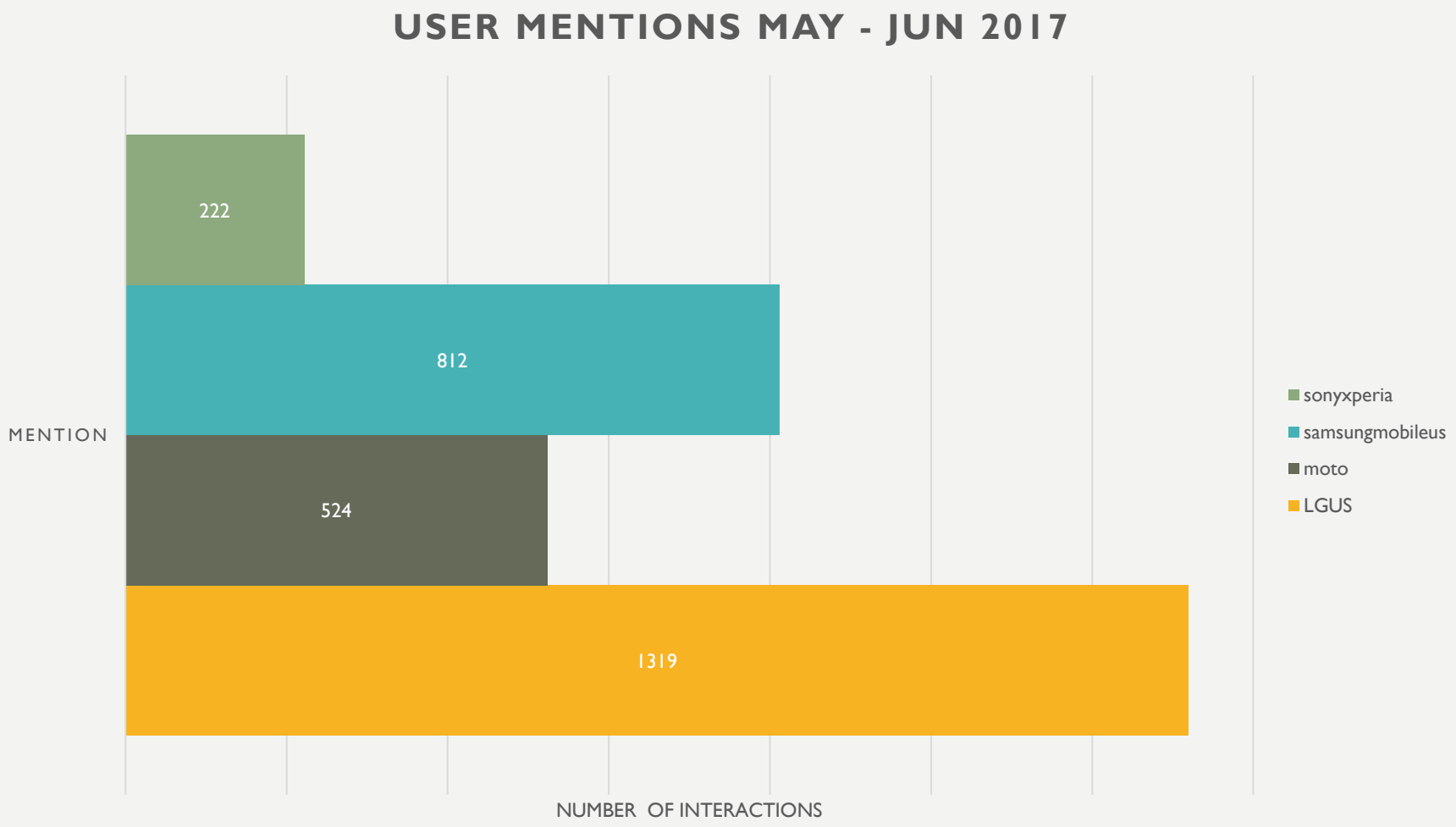
- 

DATA ANALYSIS - PLOTTING



All companies have more positive tweets than negative ones, that means that users are engagement with the brand

DATA ANALYSIS - PLOTTING



THANK YOU


NOT SURE IF CLASS IS APPALUDING
BECAUSE I WAS GOOD
OR BECAUSE
THE PRESENTATION IS FINALLY OVER
quickmeme.com