Labatt Playbook Analysis

Social Media Analysis to support Quench new business proposal

WildFig

2016-10-18

Table of Contents

## Executive Summary

* Insight 1
* Insight 2
* Insight 3

## Abstract

## Introduction

## Methods

### Data Collection

Data was imported using the \data\_gathering.RMD script. See that script for details of collection.

pander(twitter\_summary\_stats)

Table continues below

|  |  |  |  |
| --- | --- | --- | --- |
| Company | Twitter\_Followers | Twitter\_Statuses | Twitter\_Likes |
| Labatt USA | 18535 | 2816 | 13417 |
| Molson Canadian | 17258 | 4541 | 8784 |
| Michelob ULTRA | 2931 | 2931 | 41684 |
| Bud Light | 17758 | 17758 | 13085 |

|  |  |
| --- | --- |
| Twitter\_Retweets | Twitter\_EngagementPerUser |
| 8645 | 1.19 |
| 4287 | 0.7574 |
| 15411 | 1.046 |
| 5235 | 0.1146 |

pander(summary\_stats)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Company | Comments | Likes | Shares | Total.Posts |
| Labatt USA | 6717 | 127377 | 27884 | 1315 |
| Molson Canadian | 7170 | 60077 | 10678 | 517 |
| Michelob ULTRA | 116516 | 4614127 | 254690 | 3484 |
| Bud Light | 531451 | 20137767 | 1878365 | 6927 |

### Data Shaping

Taking in raw data and adding a parseable timestamp while filtering on the date and client\_ids.

### Function Definition

Define functions to create posts per day of week graphs, and timeseries of engagement line graphs.

### Additional Data Shaping for Engagement

Shape data into vertical data formats.

##   
## Attaching package: 'chron'

## The following objects are masked from 'package:lubridate':  
##   
## days, hours, minutes, seconds, years

## Results

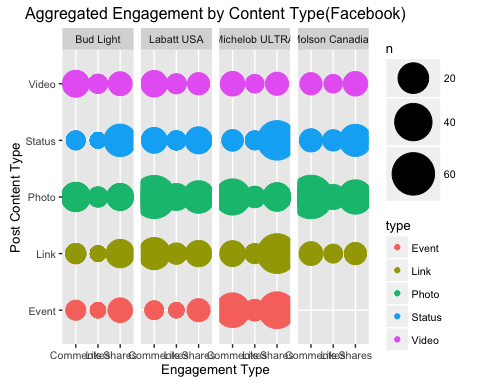
### Summary Statistics

* Lets start here with a table of summary statistics

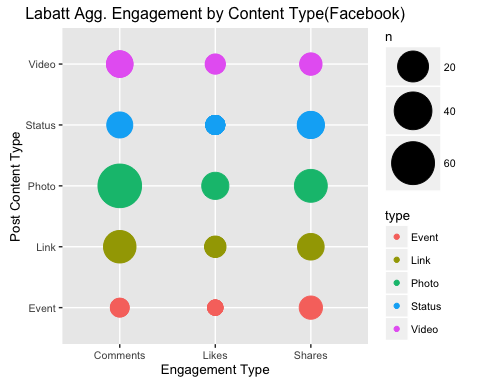
## [1] "tbl\_df" "tbl" "data.frame"

### Matrices plots of Engagement

First plot is aggregated engagement by content type. Second plot, it engagement by type for client(Labatt).



* As *Bud Light* and *Michelob ULTRA* are the to companies with the highest engagement, comparison of

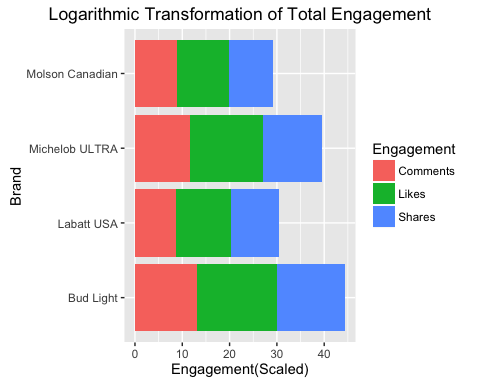


* Looking at the engagement by content type we see that **Labatt** is garnering its most significant engagment on Photos, Video, and Links.
* [ ] TODO: we need to compare posting activity with engagement activity (scatter plot)

### Summary Plots

Horizontal stacked bar chart for total engagement comparison of all companies

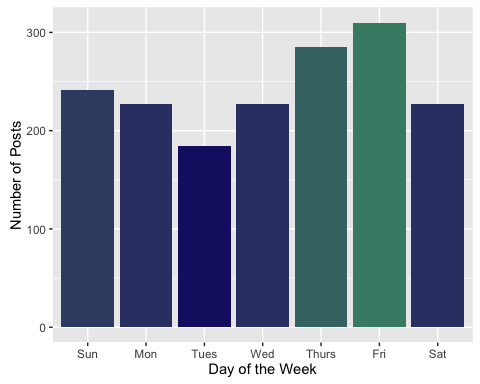
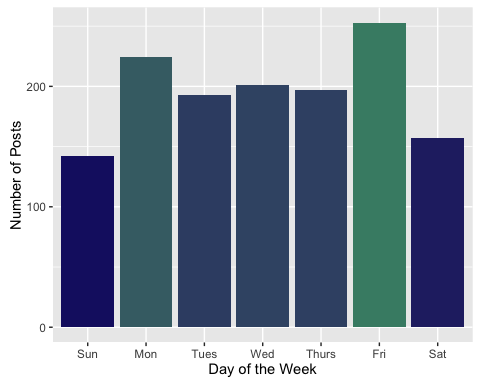
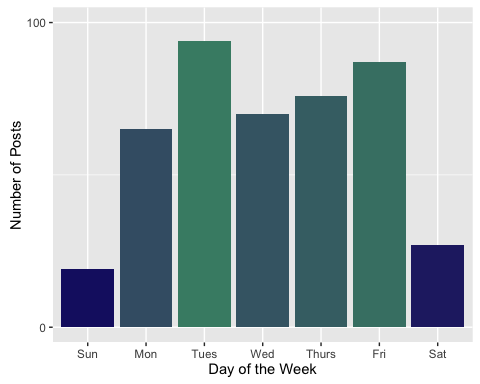
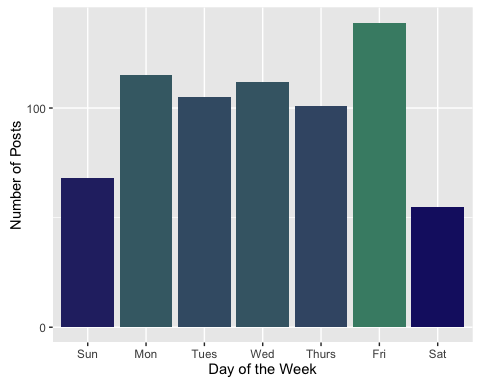
reorder\_size <- function(x) {  
 factor(x, levels = names(sort(table(x))))  
}  
p <- summary\_stats %>%  
 filter(Engagement != "Total.Posts") %>%  
 ggplot(., aes(x = Company, y = log(Number), fill = Engagement)) +  
 geom\_bar(stat = "identity") +  
 xlab('Brand') + ylab('Engagement(Scaled)') +  
 ggtitle('Logarithmic Transformation of Total Engagement') +  
 coord\_flip()  
  
plot(p)



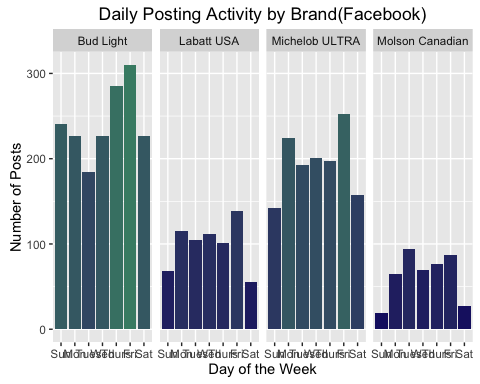
### Day of Week

Total posts per day of the week.

# without brand ID these are uninformative  
for(i in seq\_along(df\_names)) {  
 p <- day\_of\_week(df\_names[i], client\_names[i])  
 plot(p)  
}

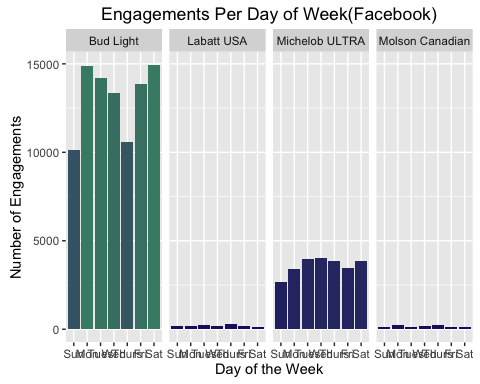


p <- ggplot(data = all\_companies\_ts, aes(x = wday(timestamp, label = TRUE))) +  
 geom\_bar(aes(fill = ..count..)) +  
 theme(legend.position = "none") +  
 xlab("Day of the Week") + ylab("Number of Posts") +  
 scale\_fill\_gradient(low = "midnightblue", high = "aquamarine4") +   
 facet\_wrap(~from\_name, ncol = 4) +  
 ggtitle("Daily Posting Activity by Brand(Facebook)")  
plot(p)



* What is the total number of posts?

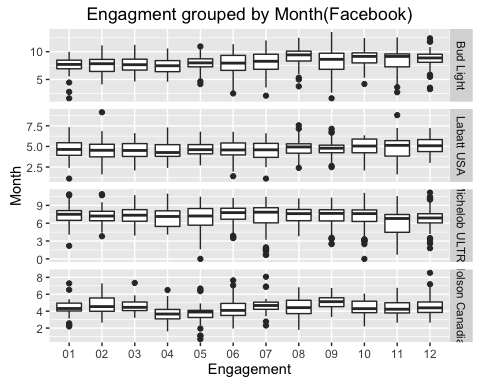
dowDat <- select(all\_companies\_ts, total\_engagement,from\_name, timestamp)  
dowDat$dow <- wday(dowDat$timestamp, label=TRUE)  
dowDat <- aggregate(total\_engagement~dow+from\_name, data=dowDat, FUN=mean)  
  
p <- ggplot(dowDat, aes(x = dow, y = total\_engagement)) +  
 geom\_bar(stat="identity", aes(fill = total\_engagement)) +   
 facet\_grid(~from\_name) +   
 ggtitle('Engagements Per Day of Week(Facebook)') +  
 theme(legend.position = "none") +  
 xlab("Day of the Week") + ylab("Number of Engagements") +  
 scale\_fill\_gradient(low = "midnightblue", high = "aquamarine4")  
plot(p)



-[ ] TODO: Create a plot for Post by engagement graphics (scatter plot). To answer the question on days with lots of posts do we get lots of engagment.

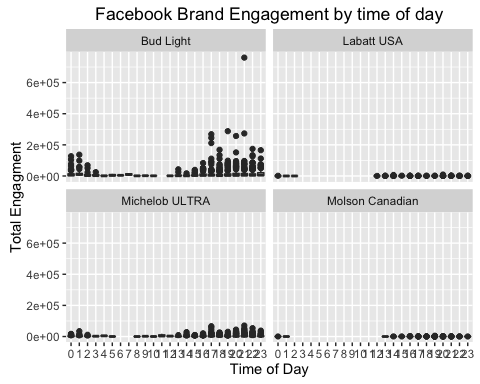
mdat <- all\_companies\_ts  
mdat$month <- format(as.POSIXct(mdat$timestamp), '%m')  
mdat %>%  
 ggplot(aes(month, log(total\_engagement))) +  
 geom\_boxplot() +  
 ggtitle('Engagment grouped by Month(Facebook)') + xlab('Engagement') + ylab('Month') +  
 facet\_grid(from\_name ~ ., scales = "free")

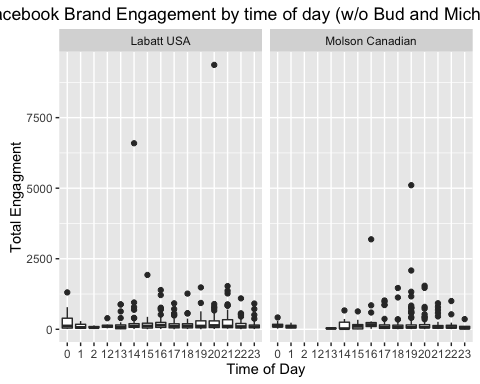
## Warning: Removed 2 rows containing non-finite values (stat\_boxplot).



* [] TODO: With that data we can ask what posts get the most engagment, we can look at top engagment and bottom engagements posts and what qualities they share or differ by.

### Engagement by Time of Day (TOD)

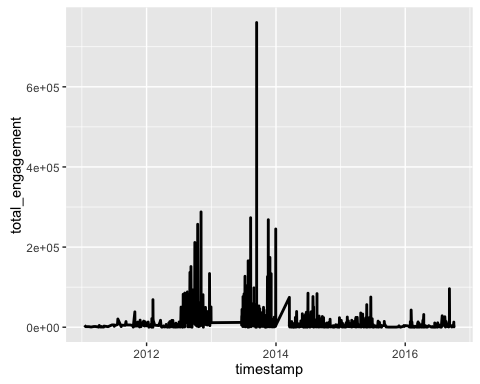
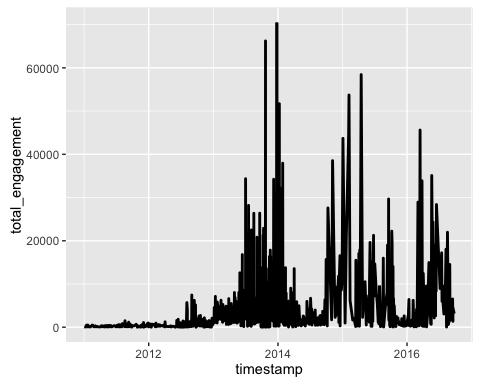
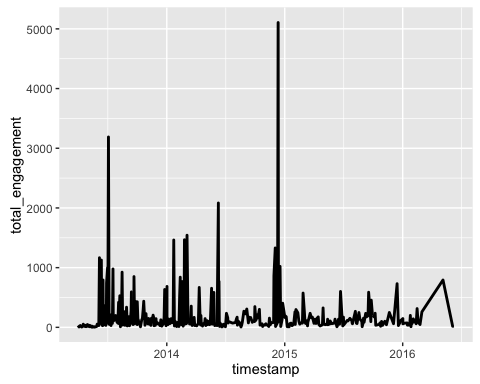
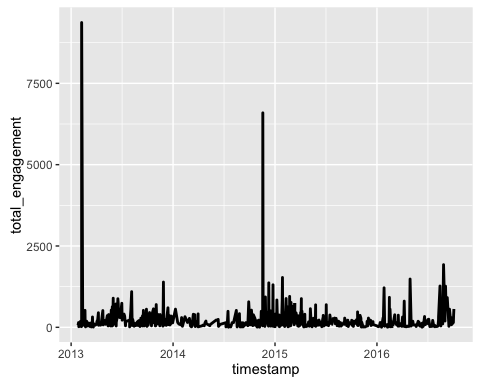




### Timeseries Engagement

Plots for the timeseries engagement line.

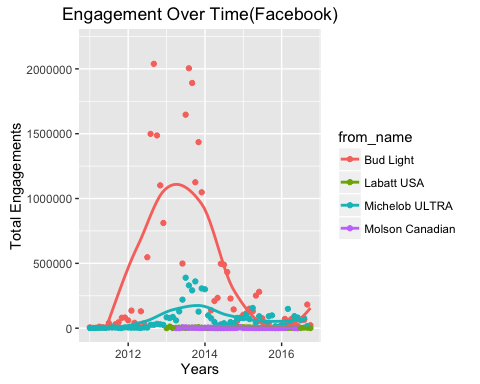
for(i in seq\_along(df\_names)) {  
 p <- timeseries\_engagement(client\_names\_proper[i])  
 plot(p)  
}



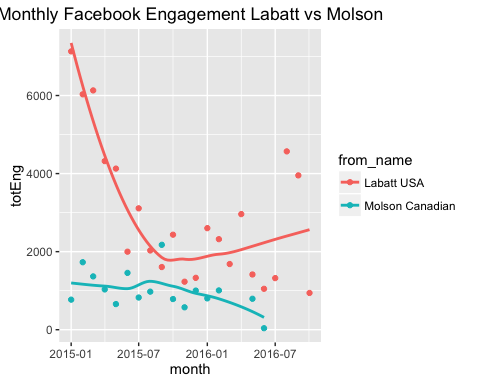
### Initial Visualization of engagement over time on a line

all\_companies\_ts <- all\_companies\_ts %>%  
 filter(from\_id %in% client\_ids) %>%  
 mutate(month = as.Date(cut(all\_companies\_ts$timestamp, breaks = "month")))  
  
all\_companies\_ts %>%  
 select(from\_name, month, total\_engagement) %>%  
 group\_by(from\_name,month) %>%  
 summarise(totEng = sum(total\_engagement)) %>%  
 ggplot(., aes(x = month, y = totEng)) +  
 ylab('Total Engagements') + xlab('Years') +  
 geom\_point(aes(color = from\_name)) + ylim(0, 2200000) +  
 ggtitle('Engagement Over Time(Facebook)') +  
 geom\_smooth(aes(color = from\_name), se = FALSE)

## Warning: Removed 18 rows containing missing values (geom\_smooth).



all\_companies\_ts %>%  
 select(from\_name, month, total\_engagement, timestamp) %>%  
 filter(from\_name != "Bud Light" ) %>%  
 filter(from\_name != "Michelob ULTRA") %>%  
 filter(year(timestamp) %in% c('2015', '2016')) %>%  
 group\_by(from\_name,month) %>%  
 summarise(totEng = sum(total\_engagement)) %>%  
 ggplot(., aes(x = month, y = totEng)) +  
 geom\_point(aes(color = from\_name)) +  
 geom\_smooth(aes(color = from\_name), se = FALSE) +  
 ggtitle("Monthly Facebook Engagement Labatt vs Molson")

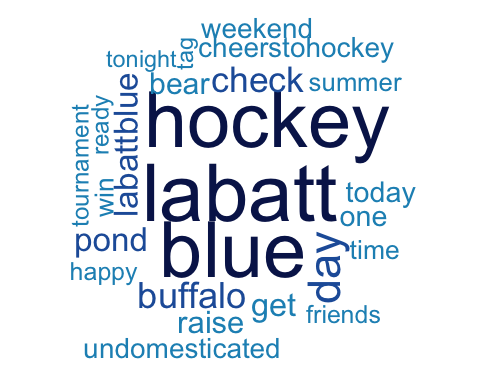


* This is an interesting drop of ~30% over the first 6 months of 2015. The brand has still not recovered from that reduction.
* What is different about the content during this period?
* Might be valuable to look back at the entire timeseries for periods of distinct dynamism.

### Labatt Wordclouds

Removed filter because labatt does not have significant inflection point whereas previous analysis

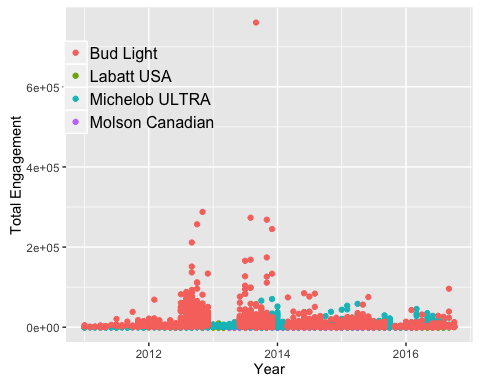
labatt$timestamp <- date(labatt$timestamp)  
  
labatt\_clean\_pre <- str\_replace\_all(labatt$message, "@\\w+", "")  
labatt\_clean\_pre <- gsub("&amp", "", labatt\_clean\_pre)  
labatt\_clean\_pre <- gsub("(RT|via)((?:\\b\\W\*@\\w+)+)", "", labatt\_clean\_pre)  
labatt\_clean\_pre <- gsub("@\\w+", "", labatt\_clean\_pre)  
labatt\_clean\_pre <- gsub("[[:punct:]]", "", labatt\_clean\_pre)  
labatt\_clean\_pre <- gsub("[[:digit:]]", "", labatt\_clean\_pre)  
labatt\_clean\_pre <- gsub("http\\w+", "", labatt\_clean\_pre)  
labatt\_clean\_pre <- gsub("[ \t]{2,}", "", labatt\_clean\_pre)  
labatt\_clean\_pre <- gsub("^\\s+|\\s+$", "", labatt\_clean\_pre)  
  
labatt\_corpus\_pre <- Corpus(VectorSource(labatt\_clean\_pre))  
labatt\_corpus\_pre <- tm\_map(labatt\_corpus\_pre, removePunctuation)  
labatt\_corpus\_pre <- tm\_map(labatt\_corpus\_pre, content\_transformer(tolower))  
labatt\_corpus\_pre <- tm\_map(labatt\_corpus\_pre, removeWords, stopwords("english"))  
labatt\_corpus\_pre <- tm\_map(labatt\_corpus\_pre, removeWords, c("amp", "2yo", "3yo", "4yo"))  
labatt\_corpus\_pre <- tm\_map(labatt\_corpus\_pre, stripWhitespace)  
  
pal <- brewer.pal(9,"YlGnBu")  
pal <- pal[-(1:4)]  
set.seed(123)  
  
wordcloud(words = labatt\_corpus\_pre, scale=c(5,0.1), max.words=25, random.order=FALSE,   
 rot.per=0.35, use.r.layout=FALSE, colors=pal)



### Point Graphs for Posts

Displays engagement per post to find outliers.

p <- ggplot(all\_companies\_ts, aes(x = month, y = total\_engagement)) +  
 geom\_point(aes(color = from\_name)) +  
 xlab("Year") + ylab("Total Engagement") +   
 theme(legend.title=element\_blank(),   
 legend.text=element\_text(size=12),   
 legend.position=c(0.18, 0.77),   
 legend.background=element\_rect(fill=alpha('gray', 0)))  
plot(p)

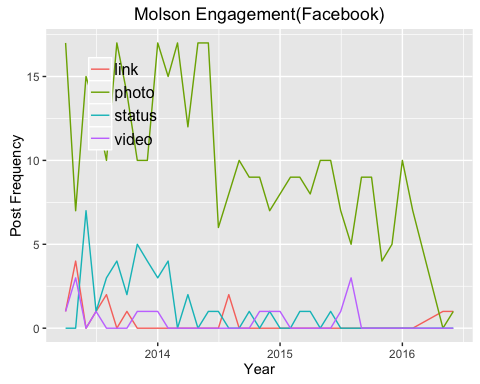


### Total Engagement Line

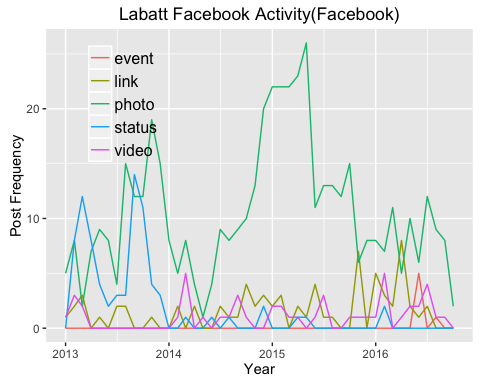
# q <- aggregate(all\_companies\_ts$total\_engagement~all\_companies\_ts$month+  
# all\_companies\_ts$from\_name,  
# FUN=sum)  
#   
# ggplot(q, aes(x = q$`all\_companies\_ts$month`, y = q$`all\_companies\_ts$total\_engagement`)) +  
# geom\_line(aes(color=q$`all\_companies\_ts$from\_name`)) +  
# ylab("Total Engagement") + xlab("Year") +  
# theme(legend.title=element\_blank(),   
# legend.text=element\_text(size=12),   
# legend.position=c(0.18, 0.77),   
# legend.background=element\_rect(fill=alpha('gray', 0)))

### Engagement by Company

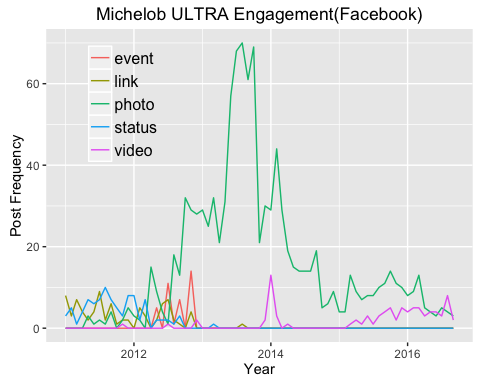
### molson Content Over Time ###  
t <- all\_companies\_ts %>%  
 filter(., from\_name == "Molson Canadian")  
t <- data.frame(table(t$month, t$type))  
  
t$Var1 <- date(t$Var1)  
ggplot(t, aes(x = Var1, y = Freq, group = Var2)) +  
 geom\_line(aes(color=Var2)) +  
 ggtitle('Molson Engagement(Facebook)') +  
 xlab("Year") + ylab("Post Frequency") +  
 theme(legend.title=element\_blank(),   
 legend.text=element\_text(size=12),   
 legend.position=c(0.18, 0.77),   
 legend.background=element\_rect(fill=alpha('gray', 0)))



#TRISTEN'S GRAPHS!!  
#Labatt Content Over Time  
  
### Labatt Content Over Time ###  
t <- all\_companies\_ts %>%  
 filter(., from\_name == "Labatt USA")  
t <- data.frame(table(t$month, t$type))  
  
t$Var1 <- date(t$Var1)  
ggplot(t, aes(x = Var1, y = Freq, group = Var2)) +  
 geom\_line(aes(color=Var2)) +  
 ggtitle('Labatt Facebook Activity(Facebook)') +  
 xlab("Year") + ylab("Post Frequency") +  
 theme(legend.title=element\_blank(),   
 legend.text=element\_text(size=12),   
 legend.position=c(0.18, 0.77),   
 legend.background=element\_rect(fill=alpha('gray', 0)))

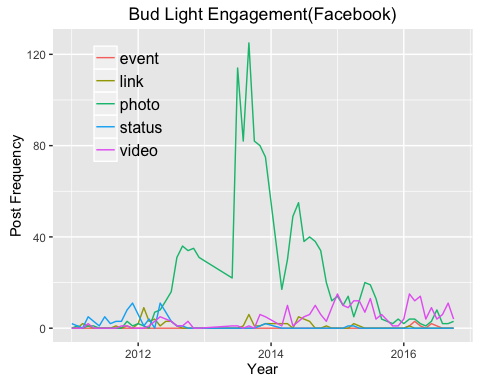


#Labatt Content Over Time  
  
#MichelobULTRA Content Over Time ###  
t <- all\_companies\_ts %>%  
 filter(., from\_name == "Michelob ULTRA")  
t <- data.frame(table(t$month, t$type))  
  
t$Var1 <- date(t$Var1)  
ggplot(t, aes(x = Var1, y = Freq, group = Var2)) +  
 geom\_line(aes(color=Var2)) +  
 ggtitle('Michelob ULTRA Engagement(Facebook)') +  
 xlab("Year") + ylab("Post Frequency") +  
 theme(legend.title=element\_blank(),   
 legend.text=element\_text(size=12),   
 legend.position=c(0.18, 0.77),   
 legend.background=element\_rect(fill=alpha('gray', 0)))



* Is this true? TODO: Verify that these are the only content types for Molson.

#Labatt Content Over Time  
  
#Bud Light Content Over Time ###  
t <- all\_companies\_ts %>%  
 filter(., from\_name == "Bud Light")  
t <- data.frame(table(t$month, t$type))  
  
t$Var1 <- date(t$Var1)  
ggplot(t, aes(x = Var1, y = Freq, group = Var2)) +  
 geom\_line(aes(color=Var2)) +  
 ggtitle('Bud Light Engagement(Facebook)') +  
 xlab("Year") + ylab("Post Frequency") +  
 theme(legend.title=element\_blank(),   
 legend.text=element\_text(size=12),   
 legend.position=c(0.18, 0.77),   
 legend.background=element\_rect(fill=alpha('gray', 0)))



### Pulling #hastags

I found an example on [Stackoverflow](http://stackoverflow.com/questions/27168226/extracting-hashtags-from-tweets)

### Experiment with Hashtag extraction

# LabattUSA\_timeline %>%   
# filter()  
#   
#   
# tweets <- LabattUSA\_timeline$text  
# match <- regmatches(tweets,gregexpr("#[[:alnum:]]+",tweets))  
#   
# # Convert the list to a corpus  
# # new\_corpus <- as.VCorpus(new\_list) from Stackoverflow (http://stackoverflow.com/questions/34061912/how-transform-a-list-into-a-corpus-in-r)  
#   
# new\_corpus <- as.VCorpus(match)  
# class(new\_corpus)  
# inspect(new\_corpus)  
#   
# EnsurePackage <- function(x) {  
# # EnsurePackage(x) - Installs and loads a package if necessary  
# # Args:  
# # x: name of package  
#   
# x <- as.character(x)  
# if (!require(x, character.only=TRUE)) {  
# install.packages(pkgs=x, repos="http://cran.r-project.org")  
# require(x, character.only=TRUE)  
# }  
# }  
#   
# MakeWordCloud <- function(corpus) {  
# # Make a word cloud  
# #  
# # Args:  
# # textVec: a text vector  
# #  
# # Returns:  
# # A word cloud created from the text vector  
#   
# EnsurePackage("tm")  
# EnsurePackage("wordcloud")  
# EnsurePackage("RColorBrewer")  
#   
# corpus <- tm\_map(corpus, function(x) {  
# removeWords(x, c("via", "rt", "mt"))  
# })  
#   
# ap.tdm <- TermDocumentMatrix(corpus)  
# ap.m <- as.matrix(ap.tdm)  
# ap.v <- sort(rowSums(ap.m), decreasing=TRUE)  
# ap.d <- data.frame(word = names(ap.v), freq=ap.v)  
# table(ap.d$freq)  
# pal2 <- brewer.pal(8, "Dark2")  
#   
# wordcloud(ap.d$word, ap.d$freq,   
# scale=c(8, .2), min.freq = 3,   
# max.words = Inf, random.order = FALSE,   
# rot.per = .15, colors = pal2)  
# }  
#   
# MakeWordCloud(new\_corpus)

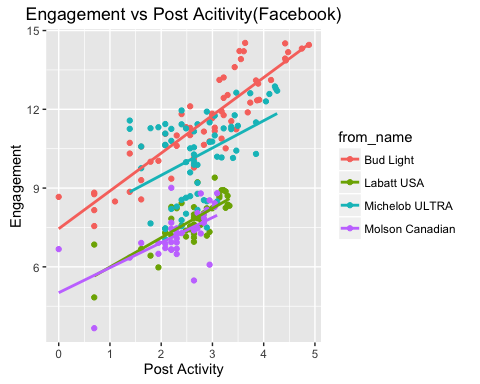
### Mosaic Plot Experiment

* [ ] TODO: Full timeseries of total eng by brand. (To look for seasonality) - if sports are a driver than seasonality might be important

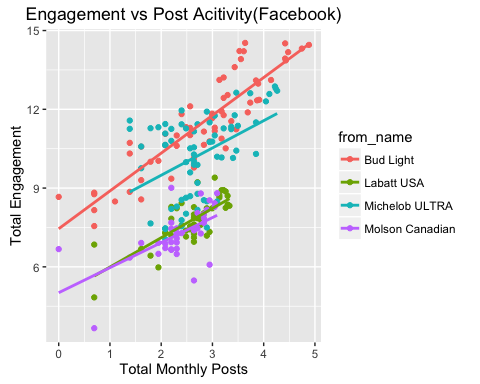
# p <- unfiltered\_ts %>%  
# summarise(jd = doy(timestamp)) %>%  
# group\_by(jd) %>%  
# ggplot(aes(factor(jd),total\_engagement)) +  
# geom\_boxplot() +   
# facet\_grid(~ from\_name)  
# plot(p)

* [ ] Populate a table of top performing posts and low performing posts - Tristen can pull shot of tweets for discussion
* [ ] Create a data.frame with these columns brand, data, tweet, engagement (I think this is a subset of all\_companies)
* [ ] summary table of brand, month, totEng, see examples:<http://leonawicz.github.io/HtmlWidgetExamples/ex_dt_sparkline.html>

all\_companies\_ts %>%  
 select(from\_name, timestamp, total\_engagement) %>%  
 group\_by(from\_name, month(timestamp), year(timestamp)) %>%  
 summarise(count = n(),   
 engagement = sum(total\_engagement)) %>%  
 ggplot(., aes(y = log(engagement), x = log(count), colour = from\_name)) +  
 geom\_point() +  
 xlab('Post Activity') + ylab('Engagement') +  
 geom\_smooth(se = FALSE, method = "lm") +  
 #geom\_smooth(se = FALSE)  
 ggtitle("Engagement vs Post Acitivity(Facebook)")

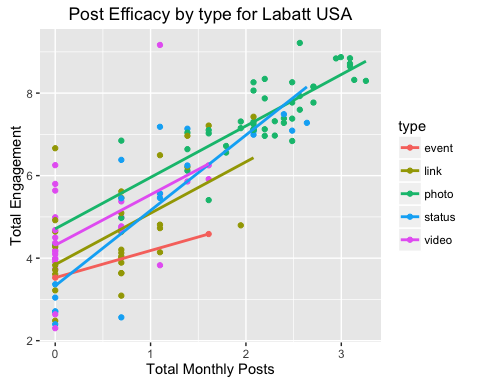


all\_companies\_ts %>%  
 #filter(from\_name != "Bud Light" ) %>%  
 #filter(from\_name != "Michelob ULTRA") %>%  
 select(from\_name, timestamp, total\_engagement) %>%  
 group\_by(from\_name, month(timestamp), year(timestamp)) %>%  
 summarise(count = n(),  
 engagement = sum(total\_engagement)) %>%  
 ggplot(., aes(y = log(engagement), x = log(count), colour = from\_name)) +  
 geom\_point() +  
 geom\_smooth(se = FALSE, method = "lm") +  
 ggtitle("Engagement vs Post Acitivity(Facebook)") +  
 ylab("Total Engagement") + xlab("Total Monthly Posts")



* There is a positive relationship between post activity (ie counts) and total engagement.

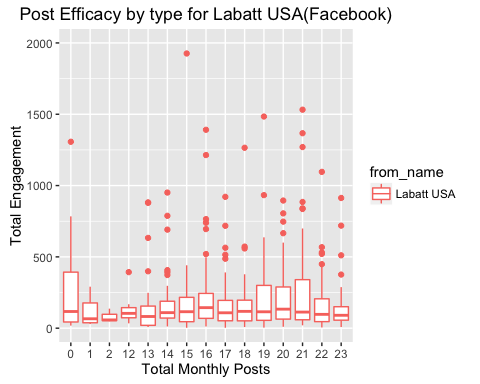
all\_companies\_ts %>%  
 filter(from\_name == "Labatt USA" ) %>%  
 select(from\_name, timestamp, type, total\_engagement) %>%  
 group\_by(from\_name, month(timestamp), year(timestamp), type) %>%  
 summarise(count = n(),  
 engagement = sum(total\_engagement)) %>%  
 ggplot(., aes(y = log(engagement), x = log(count), colour = type)) +  
 geom\_point() +  
 geom\_smooth(se = FALSE, method = "lm") +  
 ggtitle("Post Efficacy by type for Labatt USA") +  
 ylab("Total Engagement") + xlab("Total Monthly Posts")



* [X] TOD vs engagement similar to post activity vs Engagement

all\_companies\_ts %>%  
 filter(from\_name == "Labatt USA" ) %>%  
 select(from\_name, tod, total\_engagement) %>%  
 ggplot(., aes(y = total\_engagement, x = factor(tod), colour = from\_name)) +  
 geom\_boxplot() +  
 ylim(c(0,2000)) +  
 ggtitle("Post Efficacy by type for Labatt USA(Facebook)") +  
 ylab("Total Engagement") + xlab("Total Monthly Posts")

## Warning: Removed 2 rows containing non-finite values (stat\_boxplot).

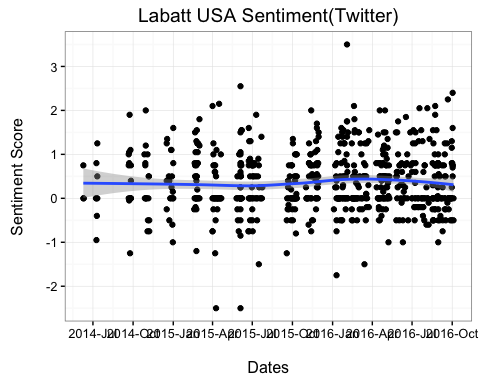


### Kevins Questions

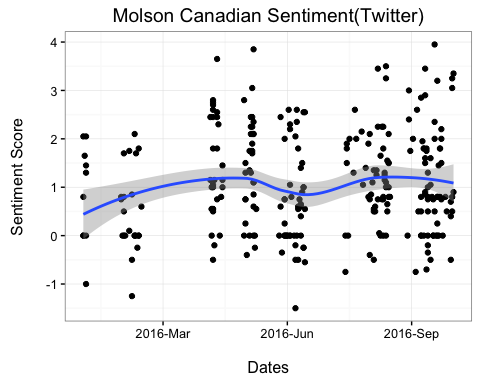
# load('processed\_data/bud\_fb.RData')  
# bud$total\_engagement <- rowSums(bud[,9:11])  
# z <- bud %>%  
# arrange(desc(total\_engagement))  
# head(z)  
# Updated upstream

## Twitter

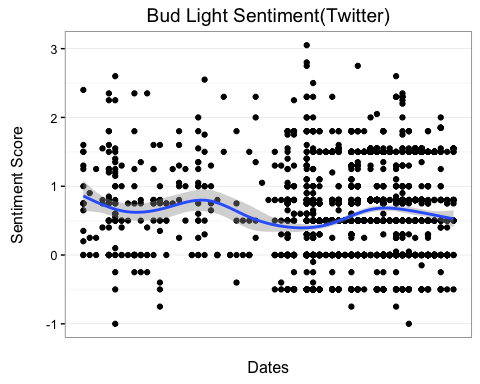
text\_clean <- function(cleanliness) {  
 cleanliness <- str\_replace\_all(cleanliness, "@\\w+", "")  
 cleanliness <- gsub("&amp", "", cleanliness)  
 cleanliness <- gsub("(RT|via)((?:\\b\\W\*@\\w+)+)", "", cleanliness)  
 cleanliness <- gsub("@\\w+", "", cleanliness)  
 cleanliness <- gsub("[[:punct:]]", "", cleanliness)  
 cleanliness <- gsub("[[:digit:]]", "", cleanliness)  
 cleanliness <- gsub("http\\w+", "", cleanliness)  
 cleanliness <- gsub("[ \t]{2,}", "", cleanliness)  
 cleanliness <- gsub("^\\s+|\\s+$", "", cleanliness)  
 return(cleanliness)  
}  
  
LabattUSA\_timeline$sentiment <- lapply(text\_clean(LabattUSA\_timeline$text), get\_nrc\_sentiment)  
labatt\_sentiment <- data.frame('created' = LabattUSA\_timeline$created,  
 'text' = LabattUSA\_timeline$text,  
 'sentiment' = as.character(LabattUSA\_timeline$sentiment))  
labatt\_sentiment$score <- get\_sentiment(as.character(text\_clean(labatt\_sentiment$text))) %>% as.numeric()  
  
labatt\_sentiment %>%  
 ggplot(aes(as\_date(created), score)) +  
 geom\_point() +  
 geom\_smooth() +  
 scale\_color\_manual(values = colourList) +  
 scale\_x\_date(name = '\nDates', breaks = date\_breaks("3 months"), labels = date\_format("%Y-%b")) +  
 scale\_y\_continuous(name = "Sentiment Score\n", breaks = seq(-5, 5, by = 1)) + theme\_bw() +  
 ggtitle('Labatt USA Sentiment(Twitter)')



Molson\_Canadian\_timeline$sentiment <- lapply(text\_clean(Molson\_Canadian\_timeline$text), get\_nrc\_sentiment)  
molson\_sentiment <- data.frame('created' = Molson\_Canadian\_timeline$created,  
 'text' = Molson\_Canadian\_timeline$text,  
 'sentiment' = as.character(Molson\_Canadian\_timeline$sentiment))  
molson\_sentiment$score <- get\_sentiment(as.character(text\_clean(molson\_sentiment$text))) %>% as.numeric()  
  
molson\_sentiment %>%  
 ggplot(aes(as\_date(created), score)) +  
 geom\_point() +  
 geom\_smooth() +  
 scale\_color\_manual(values = colourList) +  
 scale\_x\_date(name = '\nDates', breaks = date\_breaks("3 months"), labels = date\_format("%Y-%b")) +  
 scale\_y\_continuous(name = "Sentiment Score\n", breaks = seq(-5, 5, by = 1)) + theme\_bw() +  
 ggtitle('Molson Canadian Sentiment(Twitter)')



budlight\_timeline$sentiment <- lapply(text\_clean(budlight\_timeline$text), get\_nrc\_sentiment)  
budlight\_sentiment <- data.frame('created' = budlight\_timeline$created,  
 'text' = budlight\_timeline$text,  
 'sentiment' = as.character(budlight\_timeline$sentiment))  
budlight\_sentiment$score <- get\_sentiment(as.character(text\_clean(budlight\_sentiment$text))) %>% as.numeric()  
  
budlight\_sentiment %>%  
 ggplot(aes(as\_date(created), score)) +  
 geom\_point() +  
 geom\_smooth() +  
 scale\_color\_manual(values = colourList) +  
 scale\_x\_date(name = '\nDates', breaks = date\_breaks("3 months"), labels = date\_format("%Y-%b")) +  
 scale\_y\_continuous(name = "Sentiment Score\n", breaks = seq(-5, 5, by = 1)) + theme\_bw() +  
 ggtitle('Bud Light Sentiment(Twitter)')



MichelobULTRA\_timeline$sentiment <- lapply(text\_clean(MichelobULTRA\_timeline$text), get\_nrc\_sentiment)  
michelob\_sentiment <- data.frame('created' = MichelobULTRA\_timeline$created,  
 'text' = MichelobULTRA\_timeline$text,  
 'sentiment' = as.character(MichelobULTRA\_timeline$sentiment))  
michelob\_sentiment$score <- get\_sentiment(as.character(text\_clean(michelob\_sentiment$text))) %>% as.numeric()  
  
michelob\_sentiment %>%  
 ggplot(aes(as\_date(created), score)) +  
 geom\_point() +  
 geom\_smooth() +  
 scale\_color\_manual(values = colourList) +  
 scale\_x\_date(name = '\nDates', breaks = date\_breaks("3 months"), labels = date\_format("%Y-%b")) +  
 scale\_y\_continuous(name = "Sentiment Score\n", breaks = seq(-5, 5, by = 1)) + theme\_bw() +  
 ggtitle('Michelob ULTRA Sentiment(Twitter)\n')

