FIT 3152 - ASSIGNMENT 1

Prepared By:

MARK DHRUBA SIKDER - 26529548
CLARISSA WONG - 27523217

Table of Contents

Introduction

A forum is an online discussion site where the community have conversations in a form of posted messages. Users of the forum are authors, who freely communicates in different threads where different threads discuss various topics. Our main objective is to investigate and figure out whether members who communicate directly use the same language on an online forum. This report will also be inclusive of analyzing if the language used changes over time.

Preliminary Analysis

Before we proceed further with our investigation, we have to explore the webforum dataset that was provided.

This dataset contains 32 columns. Each of these columns represent a variable. From what we have observed, the PostID, ThreadID, AuthorID and WC are integer data types. Date and time are factors that will be used when it comes to plotting our time series graphs.

By using the built-in function summary for R, we have obtained the summary statistics for all variables. What we have derived is all four variables Analytic, Clout, Authentic and Tone values range from 0 to 100. Additionally, variables starting from ppron to QMark are represented in similarly.

This dataset consists of 20,000 rows. However, we have to perform data cleaning to ensure the results obtained later for investigation will be more accurate. First, we eliminate any AuthorID which has the value -1 as they are anonymous and will not contribute to our investigation. The reasoning for this is such that AuthorID with -1 can be represented by different authors which will be misleading when investigating language changing over time for members. Next, we remove any rows where word count is 0 since no thread can be empty and have no words.

Investigation of whether members who are communicating directly with each other use similar language

To investigate this event, we use the idea of standard deviation and variance instead of using the existing approach like LSM score comparison (Gonzalez et al.).

According to Webopedia, “an online thread can be defined as a series of messages that have been posted as replies to each other. By reading each message in a thread, one after the other, you can see how the discussion has evolved. You can start a new thread by posting a message that is not a reply to an earlier message.” From this definition, it is understood that participants in a thread communicate directly with each other, generally discussing a topic. Therefore, the posts in a particular thread can be seen as direct communication between the participants or authors in that thread.

To determine the direct communication between members where each individual member is represented by a unique Author ID, are not considered. This is because one participant can have

multiple posts and each of the posts from an individual participant is assumed to be related to the other posts.

We used the data fields from LIWC to measure the similarity in the patterns of the language. This data field consists of Analytic, Clout, Authentic, Tone, ppron, I, we, you, shehe, they, affect, posemo and negemo, where *'ppron'* are all the personal pronouns, *'shehe'* are all the third person singular and *'they'* are third person plural. '*posemo*' and '*negemo*' are linguistic features which describes the languages expressing sentiments. We ignored the fields which are not proportions such as word count, anx, anger, family, friend, etc.

Procedure:

Using the cleaned dataset, we chose five different threads (Thread ID's: 223928, 176795, 307701, 127115 and 823462). For this selection, we used the subset command built-in RStudio to pick out all the rows matching the ThreadID we specified. However, we did plan to use a set of samples and then select the threads from that sample, but this had drawbacks. By doing this we could have removed some important information which would have affected our investigation strategy and ultimately prove the whole hypothesis wrong.

Our hypothesis is gathered from the fact that the standard deviations of the key words (linguistic features) in a thread will be low since the people are using similar language. Standard deviation measures the dispersion of values in a distribution. So, similar languages should result in lower standard deviation as the dispersion of the language is low. On the other hand, the forum comprises of numerous threads and, in a discussion forum, the language is thought to be random, since language is possibly different in different threads. Therefore, the standard deviation of a key feature in a thread is expected to be lower than that in the overall forum.

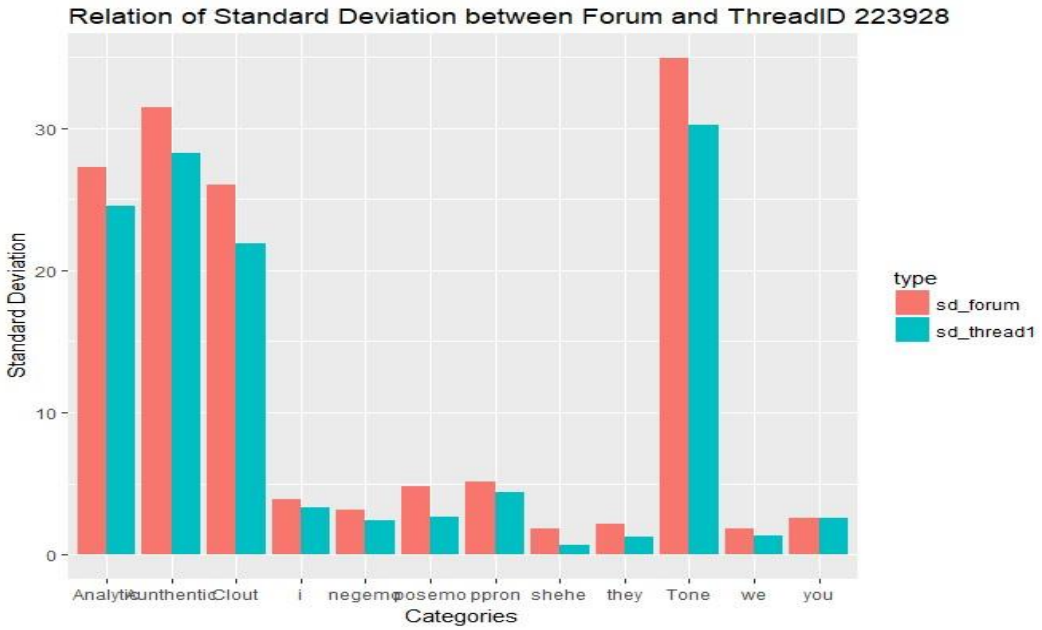
In short, if the members in a certain thread are discussing about a positive issue say Birthday party, then the standard deviation for the key feature, posemo, should be lower compared to the standard deviation of the whole forum. This is because the discussion of the Birthday party is only being discussed in that thread whereas if we consider the whole forum, the discussion about Birthday party is a part. Thus, the dispersion regarding the whole forum should be more than a particular thread. Hence, if the standard deviation of the key features in a thread is observed to be lower than those in a forum, we can say that the members in a similar thread use similar language.

Note: The dispersion mentioned here is basically talking about the variance.

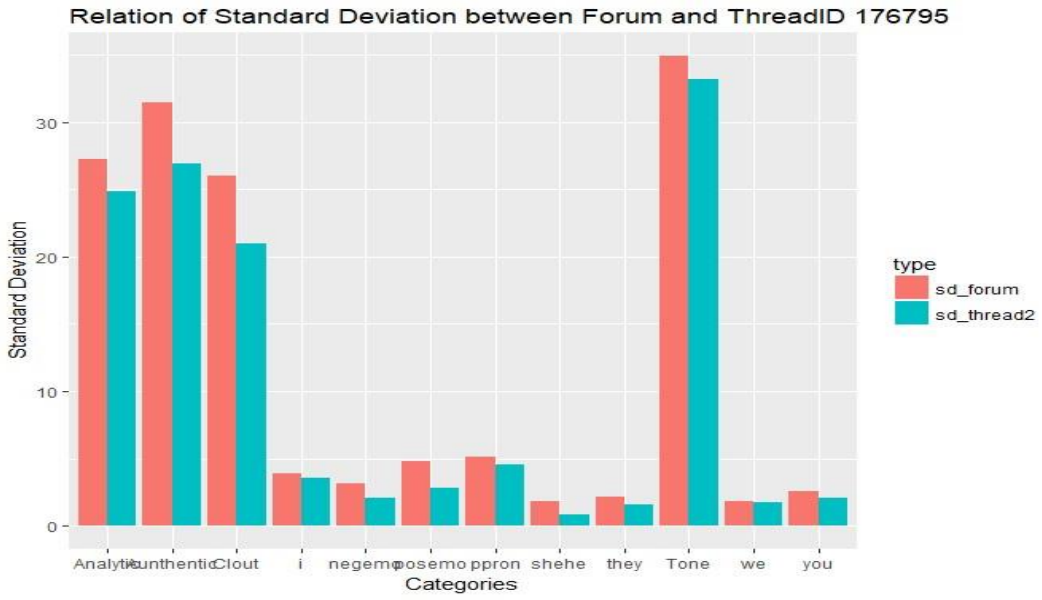
So, now back to our investigation, we set up all the linguistic features in a vector and name it Category. Then, using the Thread ID's chosen previously, we find the standard deviation regarding the categories i.e. Analytic, Clout, Authentic, ppron, I, shehe, etc. for each particular thread and then use the same categories to find the standard deviation for the forum. Then we use ggplot to plot a bar graph.

We excluded word count from investigation because interpreting difference in standard deviation is difficult. It is assumed different threads may have very similar word counts even if different language is used. So, Word Count may not lead to meaningful comparison to measure similar language. Also, since the word count is not standardized like the keywords which range from 0 to 100, a few outlier posts with very high word count may affect the standard deviation, and ultimately, leading to biased comparison of standard deviations. This applies to key words like anx, anger and friends too.

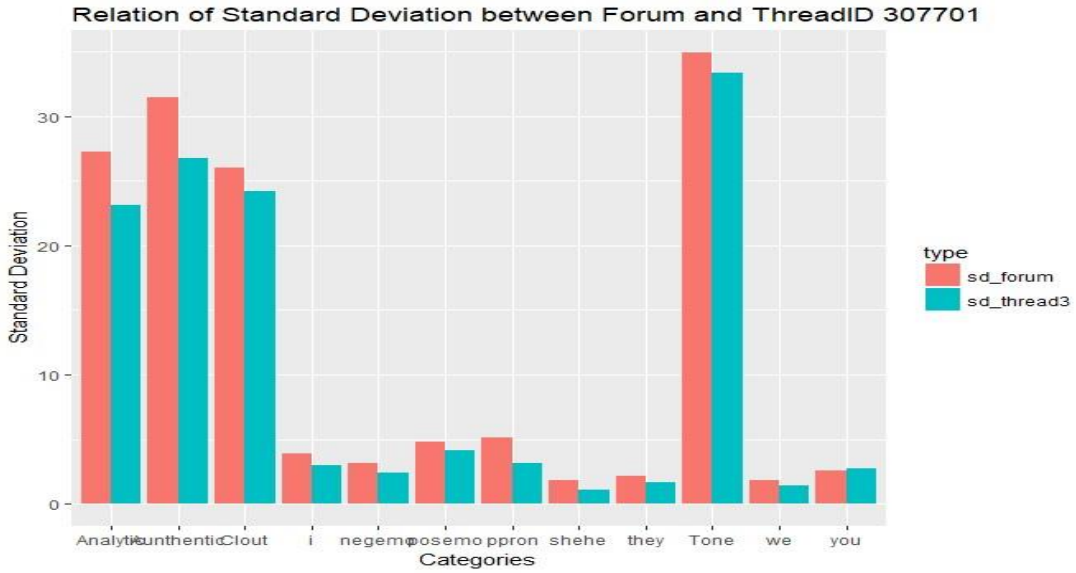
Plot 1


Relation of Standard Deviation between Forum and ThreadID 223928
Standard Deviation
30
20
10
0
Analytic
Authentic
Clout
i
negemo
posemo
ppron
shehe
they
Tone
we
you
Categories
type
sd_forum
sd_thread1

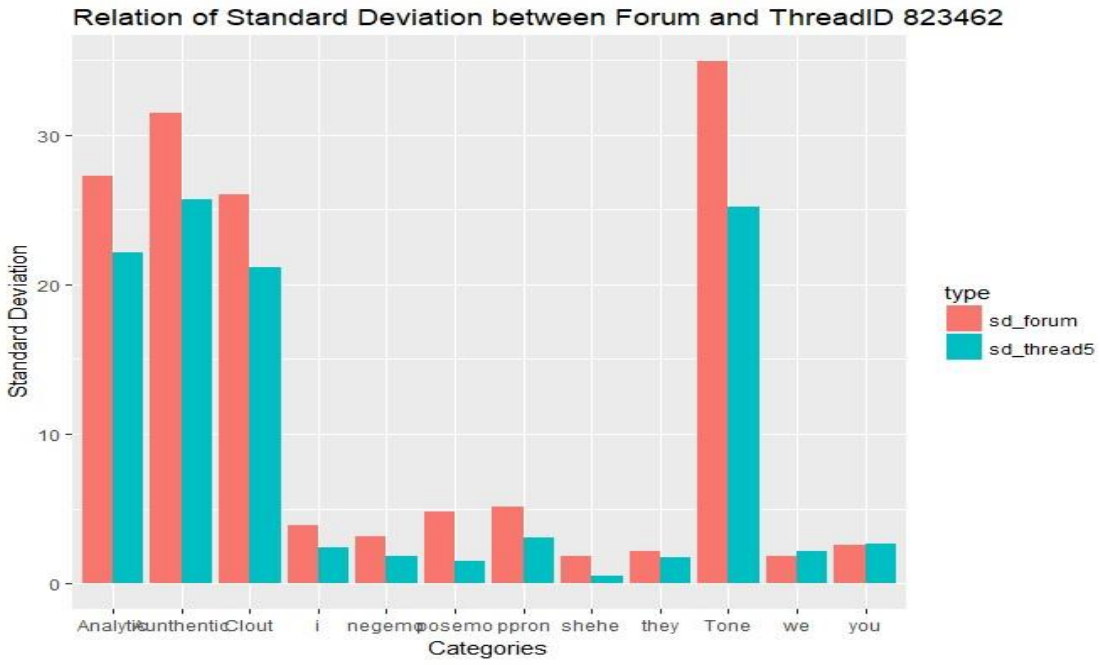
Plot 2


Relation of Standard Deviation between Forum and ThreadID 176795
Standard Deviation
30
20
10
0
Analytic
Authentic
Clout
i
negemo
posemo
ppron
shehe
they
Tone
we
you
Categories
type
sd_forum
sd_thread2

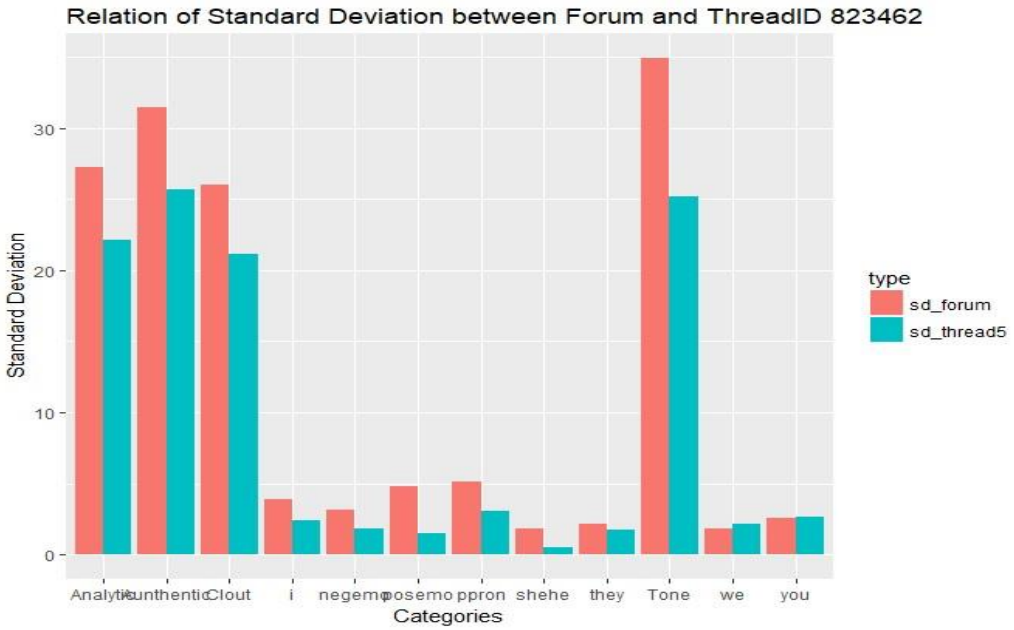
Plot 3


Relation of Standard Deviation between Forum and ThreadID 307701
Standard Deviation
30
20
10
0
Analytic
Authentic
Clout
i
negemo
posemo
ppron
shehe
they
Tone
we
you
Categories
type
sd_forum
sd_thread3

Plot 4


Relation of Standard Deviation between Forum and ThreadID 823462
Standard Deviation
30
20
10
0
Analytic
Authentic
Clout
i
negemo
posemo
ppron
shehe
they
Tone
we
you
Categories
type
sd_forum
sd_thread5

Plot 5



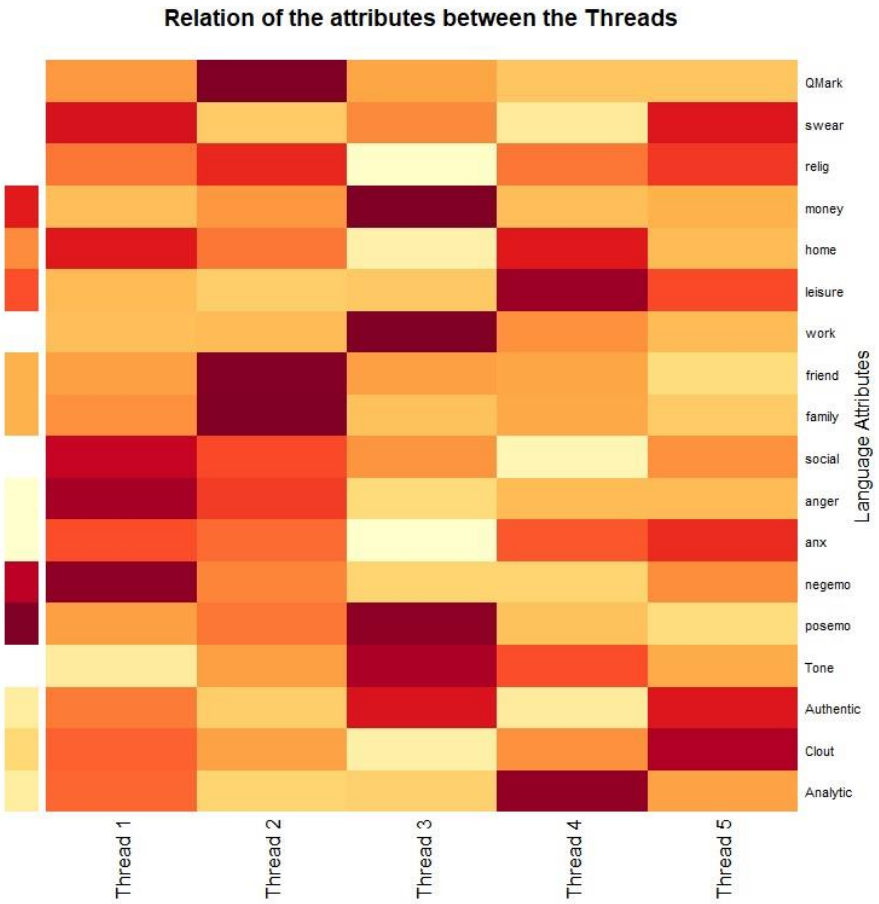
The plots above show the comparison between the forum and the individual threads, identified by its unique Thread ID.

From the charts, it can be observed that the standard deviation of the key features regarding the threads is generally lower than the standard deviation of the key features in the overall forum. However, there may be some cases like in the plot of ThreadID 823462 and 307701, where the standard deviation of some key features in the thread might be equal or greater than the standard deviation in the forum. To be more specific, we observed that the standard deviation of '*you'* are a little greater than the standard deviation of the forum. But in general, there are higher number of key features which has higher standard deviation in the forum compared to the thread's standard deviation.

Thus, after such observations we can conclude that members communicating directly with each other use similar language for communication.

Furthermore, to strengthen the observation, we plotted a heatmap to figure what exactly each group was communicating about and whether different threads talk about different things or the same things in a different thread.

Plot 6



This heatmap basically compares each key feature of different threads. The level of heat is considered using the concept of colour. Intensity of colour is proportional to the heat. For example, if we consider Thread 3, then we can observe that the heat for Authentic, Tone, posemo, work, money is more compared to other threads. This means that the members in Thread 3 are chatting more about work and money which results in more positive emotions. This means that the members might be happy with the earnings and the work they do. This also results in chatting about genuine work-related stuffs which the members are happy about and thus talking in more excitement (higher Tone).

Now, if we consider Thread 1, we observe that the members are communicating via lot of negative emotions and this results in anger and use of more swear words. This might be the case due to socialism. Thus, the tone using is much less compared to other threads, meaning that the people are chatting with less excitement with each other.

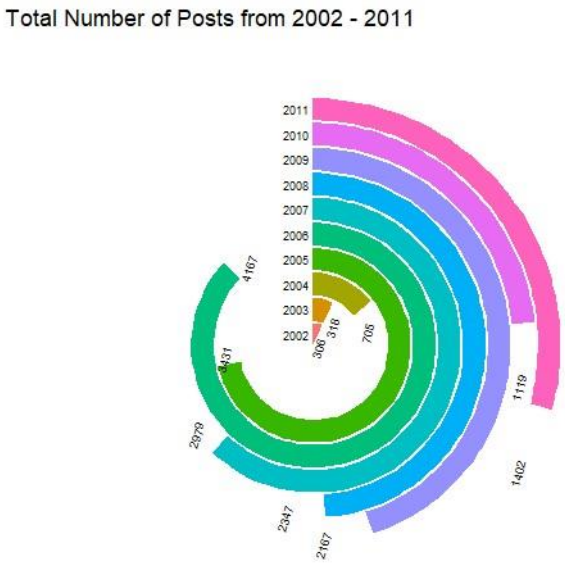
Using the observations from the Heat map we can conclude that members in the same thread are indeed communicating in a similar pattern which means they are communicating in a similar language, and different threads are communicating about different topics but in a similar language relevant to the topic they are discussing on.

Investigation of number of posts in an online forum over time

Procedure

In order to find out during which year was the number of posts the posts we plotted a circular bar plot using the cleaned data. To plot such a graph, firstly we took filtered the data regarding each year assigned each into a variable. After filtering, we had a total of 10 individual datasets with their own specific feature information. Then calculated the number of entries each of the new data set had. This represented the number of posts there were in each year. Finally a graph was plotted using the ggplot library where the years are from 2002-2011 and the number of posts from a range of approximately 300-4200.

Plot 7



Based on the graph, 2006 holds the highest number of posts. Starting 2006 however, we can observe the number of posts gradually decrease. This could be contributed by the fact that there are other platforms where members prefer to communicate or have discussions such as Facebook and Twitter.

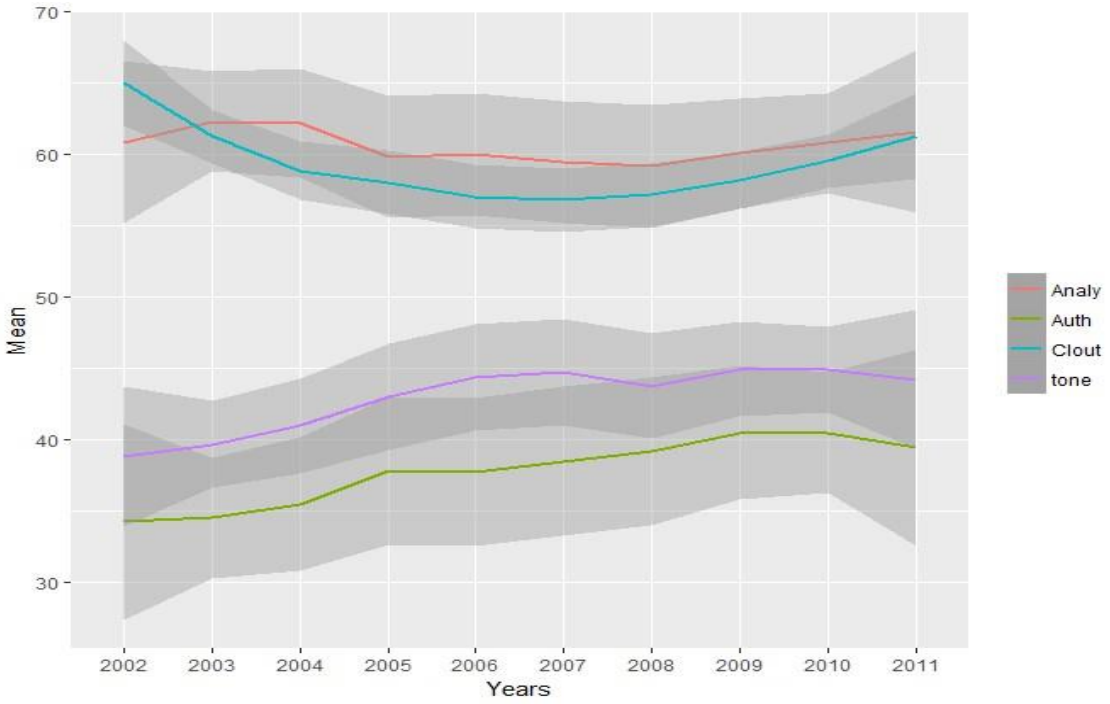
This is supported as Facebook officially launches in 2006 and by 2007 when Microsoft has purchased a share of Facebook and rights were included to advertise the social networking site.

Investigation of whether language used changes over time

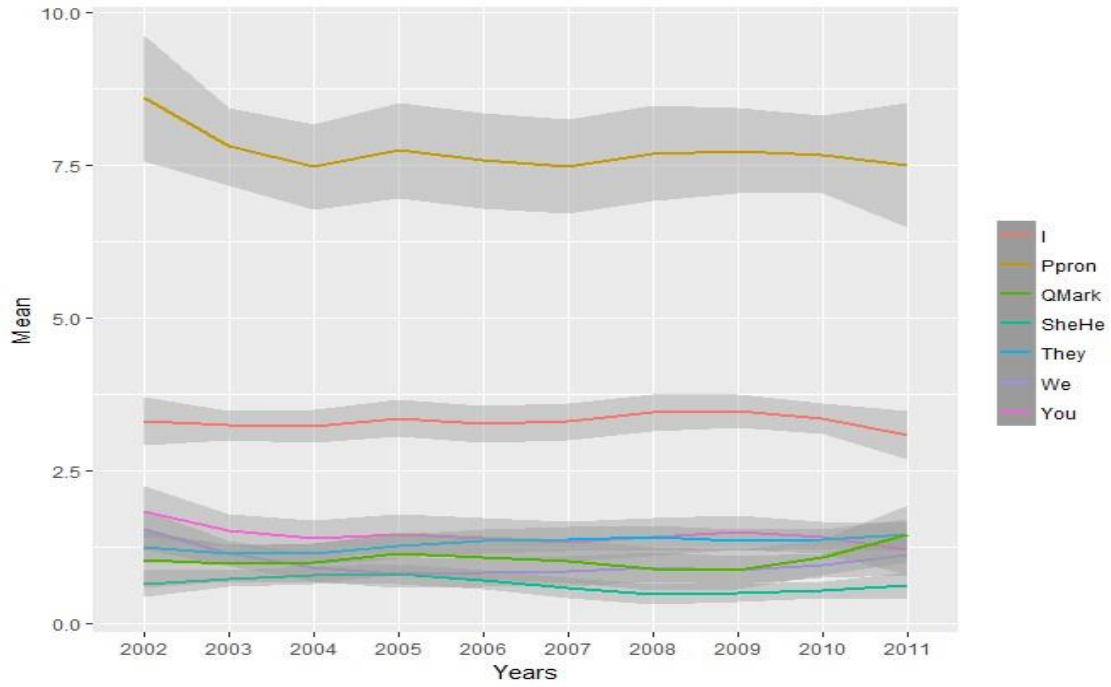
Procedure

Starting with our clean data, we assign one variable corresponding to every year starting from 2002 until 2011. We proceed with creating a function that enables the calculation of mean for each attribute, every year. Set the previous mean results of attributes into corresponding vectors. The vectors are then converted into a data frame for ggplot. Time series graph was plotted using ggplot and geom_smooth was used to see the outliers as well as the quartiles.

Plot 8



Plot 9

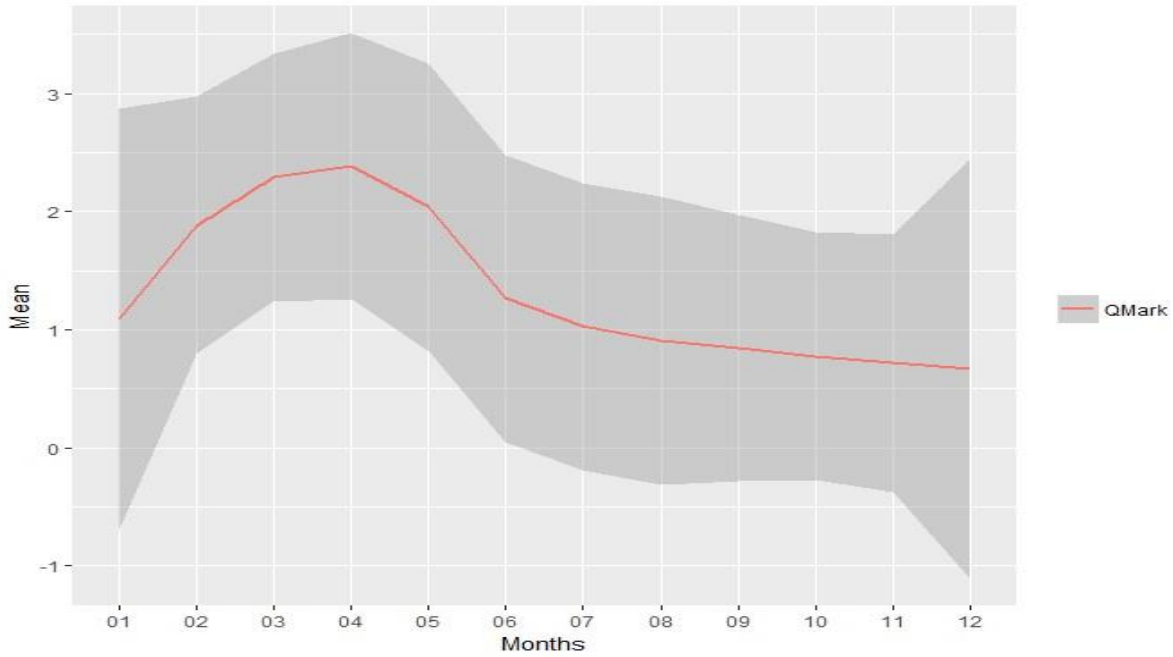


Based on the graphs as shown above, it was derived that in overall, the language used has not really changed over time. There are however a few attributes that have had significant changes over time that we took into consideration. First, authenticity do increase over the years, we believe that this is natural as people communicate more and as time elapsed, the community becomes more honest and genuine in terms of communication. Besides that, we also investigate the rise in tone over the years.

	Tone
ppron	0.014807133
i	0.065289641
we	-0.009105453
you	0.038557693
shehe	-0.024247620
they	-0.099508800
number	-0.001340433
affect	0.236754686
posemo	0.576368997
negemo	-0.464380386
anx	-0.142975314
anger	-0.326231002
social	-0.004638522
family	-0.010009237
friend	0.025347642
work	0.010460332
leisure	0.084792650
home	-0.003161439
money	0.031285314
relig	-0.048093545
swear	-0.087589468
QMark	-0.016940805

With posemo having the highest correlation to Tone, we can conclude that the forum is used to express more positive emotions rather than negative. Lastly, the usage of Question Mark (Punctuation) has increased rapidly in the year 2011. We look further into this case by breaking down the year 2011 into months and observed the graph as below.

Plot 10



The breakdown of the year 2011 has shown that the usage for Question Mark (?) was at its peak in the months of March, April and May. We made assumptions that this was possibly caused by these particular events: Beginning of Syrian Civil War (March), The Royal Wedding (April) and the announcement of Osama bin Laden, the founder and leader of the militant group Al-Qaeda was killed (May).

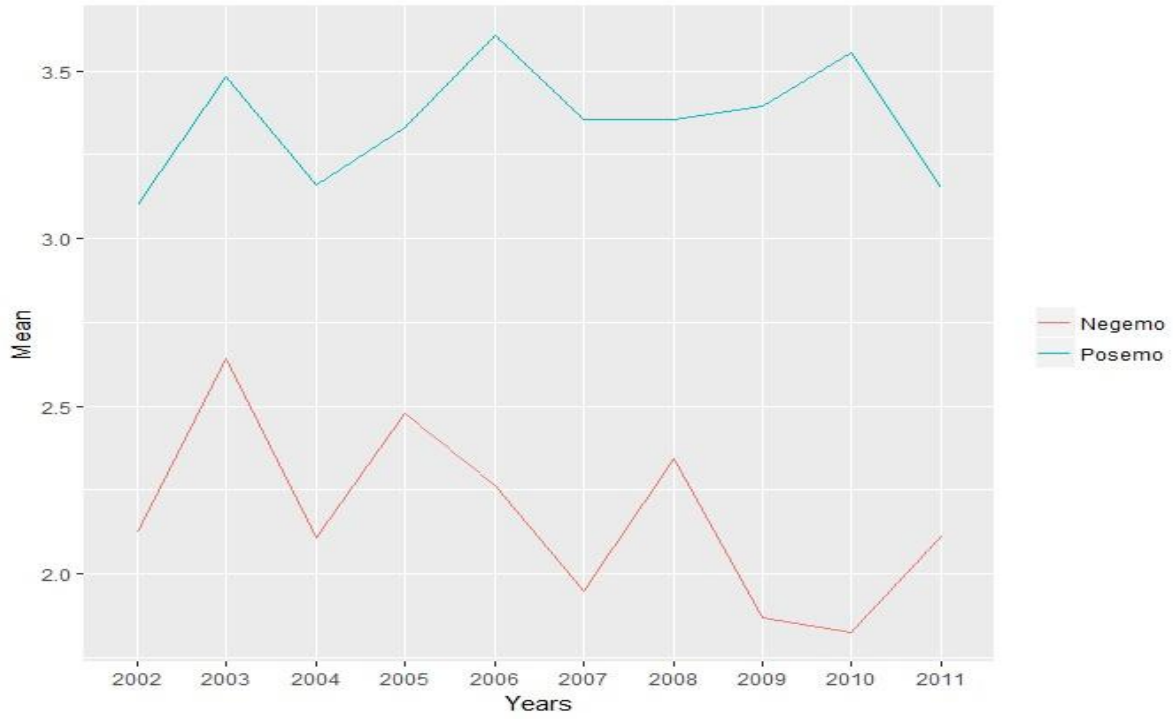
Investigating the proportion of language expressing optimism changes over time

Our approach to investigating the language expressing optimism is by factoring out specific attributes that affect optimism and in this online forum, we chose the two attributes posemo and negemo, two factors that determine the optimism for our time series graph. With this in mind, we can easily observe how the optimism in the forum has changed over time.

Procedure

Filter the rows according to their year with the dataset that has been cleaned. Using the function we have created, the mean was extracted for each attribute by year. Ggplot was used to plot the change in optimism over time.

Plot 11



As we can observe from the above graph, it is evident that the forum was used more to express positive emotions in comparison to negative ones. Although we can see the inconsistency of both attributes, it can be perceived that negative emotions have been slowly declining whereas positive emotions remain somewhat uniform throughout the years. In addition, it is apparent that positive emotion was particularly high in the year 2006 followed by 2010 and the corresponding negative emotion has dropped.

If we look back at Plot 7, the year 2006 is also the year with the highest number of posts at 4167. Hence, it is no surprise that this is the same year with the most noticeable peak for positive emotions as the forum is more inclined to consist of positive emotions compared to negative emotions.

To conclude, the forum was used to express positive emotions more than negative emotions. Over the time however, the optimism has varied by little.

Investigating the variables affecting Posemo and Negemo

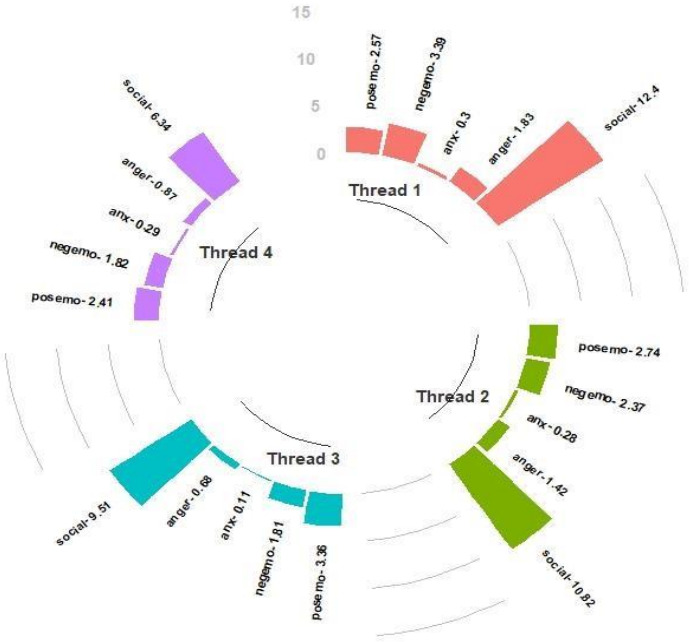
Procedure

Using correlation, we determine the variables that have the closest value to either +1 or -1 against both positive and negative emotions. By selecting a few variables, we then take a few sample threads and construct a plot to evaluate the variables against posemo and negemo.

	posemo	negemo
anx	-0.028693857	0.37973423
anger	-0.064659966	0.69467265
social	0.066986488	0.02748503
family	-0.014580169	-0.03923132
friend	0.080877216	0.01435620
work	0.001794970	-0.04489638
leisure	0.055738069	-0.03454547
home	-0.001367917	-0.03717040
money	-0.016898999	-0.01029692
relig	0.017573516	0.04894938
swear	-0.007162470	0.24206341

Based on this correlation, it can be deducted that the variables closest to posemo are friend, social and anger whereas with negemo are anx, anger and swear. From this alone, anx is noticeably the common variable and affects both emotions differently. However, we have to consider variables that are close to both posemo and negemo and not individually. Hence, we went along with the three variables, anx, anger and social to proceed with our graph.

Plot 12



According to the graph above, social increases as posemo increases. This is because social where social words refer to social processes are most likely linked to celebrations and festivities, this

comprise of positive emotions. Aside from that, anx decreases as posemo increases and is shown to be higher than negemo.

Work Breakdown Structure

Member/Task	Clarissa Wong	Mark Dhruba Sikder	Total
Preliminary Analysis	50%	50%	100%
R Research and Coding	40%	60%	100%
Preparation of Graphics	50%	50%	100%
Analysis of results	50%	50%	100%
Writing up the report	60%	40%	100%

Libraries Used

Library (lattice)

Library (ggplot2)

Library (reshape)

Library (reshape2)

Library ("RColorBrewer")

Library (tidyr)

Library (lubridate)

Library (plotrix)

Library (tidyverse)

Library (dplyr)

Appendix

```
# importing all libraries library(lattice)
#install.packages("ggplot2") library(ggplot2)
#install.packages("yaml")
#install.packages("reshape")
library(reshape) library(reshape2)
#install.packages("RColorBrewer")
library("RColorBrewer")
#install.packages("tidyr")
library(tidyr) library(lubridate)
display.brewer.all()
#install.packages('plotrix')
library(plotrix)
#install.packages("tidyverse")
library(tidyverse)
#install.packages("dplyr") library(dplyr)

# reading the file
#setwd("Z:/Data analytics/Assignment 1 Final")
webforum <- read.csv("C:\\Users\\clarr\\OneDrive\\FIT3152\\webforum.csv")

######################################## Preliminary analysis  summary(webforum)

# Data cleaning
# removing the Word counts which are 0 and anonymous author Id's
data_clean <- subset(webforum, webforum$WC > 0) data_clean <-
subset(data_clean, AuthorID > 0)
View(data_clean)
```

```
###### Investigation of whether members who are communicating directly with each other use similar language

# setting the tidy data as forum for easier understanding  forum

<- data_clean

View(forum)

# choosing 5 different threads

thread1 <- subset(forum, forum$ThreadID == 223928)


#View(thread1)

thread2 <- subset(forum, forum$ThreadID == 176795)

#View(thread2)

thread3 <- subset(forum, forum$ThreadID == 307701)

#View(thread3)

thread4 <- subset(forum, forum$ThreadID == 127115)

#View(thread4)

thread5 <- subset(forum, forum$ThreadID == 823462)

#View(thread5)


# setting categories

category <- c("Analytic", "Clout", "Aunthentic", "Tone", "ppron", "i", "we", "you", "shehe", "they", "posemo", "negemo")



#-------------------------------------------------------------------------------------------------#

# For thread1


# finding the standard deviation for all the variables

sd_thread1 <- c(sd(thread1$Analytic), sd(thread1$Clout), sd(thread1$Authentic), sd(thread1$Tone), sd(thread1$ppron), sd(thread1$i), sd(thread1$we), sd(thread1$you), sd(thread1$shehe), sd(thread1$they),

                sd(thread1$posemo), sd(thread1$negemo))

#View(sd_thread1)
```

```
sd_forum <- c(sd(forum$Analytic),  sd(forum$Clout), sd(forum$Authentic), sd(forum$Tone),
sd(forum$ppron), sd(forum$i), sd(forum$we), sd(forum$you), sd(forum$shehe), sd(forum$they),
sd(forum$posemo), sd(forum$negemo))


# plotting a bar plot graph of the standrad deviations of both the Thread1 and Forum to find a relation.


# setting the sd_thread1 and sd_forum as vectors for plotting
thread1_forum_vec <- c(sd_forum, sd_thread1) type <-
c(rep("sd_forum", 12), rep("sd_thread1", 12))
# converting data to data frame
my_data <- data.frame(category, thread1_forum_vec)


# plot
graph_plot <- ggplot(my_data, aes(category, thread1_forum_vec))
graph_plot <- graph_plot + geom_bar(stat = "identity", aes(fill = type), position = "dodge")
graph_plot <- graph_plot + labs(x = "Categories", y = "Standard Deviation", title = "Relation of Standard
Deviation between Forum and ThreadID 223928") graph_plot


#-------------------------------------------------------------------------------------------#
# For thread2
# finding the standard deviation for all the variables
sd_thread2 <- c(sd(thread2$Analytic), sd(thread2$Clout), sd(thread2$Authentic), sd(thread2$Tone),
        sd(thread2$ppron), sd(thread2$i), sd(thread2$we), sd(thread2$you),
sd(thread2$shehe), sd(thread2$they),
        sd(thread2$posemo), sd(thread2$negemo))


# plotting a bar plot graph of the standrad deviations of both the Thread1 and Forum to find a relation.


# setting the sd_thread1 and sd_forum as vectors for plotting
thread2_forum_vec <- c(sd_forum, sd_thread2) type <-
c(rep("sd_forum", 12), rep("sd_thread2", 12))
# converting data to data frame
```

```
my_data2 <- data.frame(category, thread2_forum_vec)

# plot
graph_plot2 <- ggplot(my_data2, aes(category, thread2_forum_vec))
graph_plot2 <- graph_plot2 + geom_bar(stat = "identity", aes(fill = type), position = "dodge")
graph_plot2 <- graph_plot2 + labs(x = "Categories", y = "Standard Deviation", title = "Relation of Standard Deviation between Forum and ThreadID 176795") graph_plot2

#----------------------------------------------------------------------------------------------------#
# For thread3

# finding the standard deviation for all the variables
sd_thread3 <- c(sd(thread3$Analytic), sd(thread3$Clout), sd(thread3$Authentic), sd(thread3$Tone),
sd(thread3$ppron), sd(thread3$i), sd(thread3$we), sd(thread3$you), sd(thread3$shehe),
sd(thread3$they),
          sd(thread3$posemo), sd(thread3$negemo))

# plotting a bar plot graph of the standrad deviations of both the Thread1 and Forum to find a relation.

# setting the sd_thread1 and sd_forum as vectors for plotting
thread3_forum_vec <- c(sd_forum, sd_thread3) type <-
c(rep("sd_forum", 12), rep("sd_thread3", 12))
# converting data to data frame
my_data3 <- data.frame(category, thread3_forum_vec)

# plot
graph_plot3 <- ggplot(my_data3, aes(category, thread3_forum_vec))
graph_plot3 <- graph_plot3 + geom_bar(stat = "identity", aes(fill = type), position = "dodge")
graph_plot3 <- graph_plot3 + labs(x = "Categories", y = "Standard Deviation", title = "Relation of Standard Deviation between Forum and ThreadID 307701") graph_plot3


#----------------------------------------------------------------------------------------------------#
```

```
# For thread4

# finding the standard deviation for all the variables

sd_thread4 <- c(sd(thread4$Analytic), sd(thread4$Clout), sd(thread4$Authentic), sd(thread4$Tone),
sd(thread4$ppron), sd(thread4$i), sd(thread4$we), sd(thread4$you), sd(thread4$shehe),
sd(thread4$they),
        sd(thread4$posemo), sd(thread4$negemo))

# plotting a bar plot graph of the standrad deviations of both the Thread1 and Forum to find a relation.

# setting the sd_thread1 and sd_forum as vectors for plotting

thread4_forum_vec <- c(sd_forum, sd_thread4) type <-

c(rep("sd_forum", 12), rep("sd_thread4", 12))

# converting data to data frame

my_data4 <- data.frame(category, thread4_forum_vec)

# plot

graph_plot4 <- ggplot(my_data4, aes(category, thread4_forum_vec))

graph_plot4 <- graph_plot4 + geom_bar(stat = "identity", aes(fill = type), position = "dodge")

graph_plot4 <- graph_plot4 + labs(x = "Categories", y = "Standard Deviation", title = "Relation of Standard
Deviation between Forum and ThreadID 127115") graph_plot4

#-------------------------------------------------------------------------------------------------#

# For thread5

# finding the standard deviation for all the variables

sd_thread5 <- c(sd(thread5$Analytic), sd(thread5$Clout), sd(thread5$Authentic), sd(thread5$Tone),
        sd(thread5$ppron), sd(thread5$i), sd(thread5$we), sd(thread5$you),
sd(thread5$shehe), sd(thread5$they),
        sd(thread5$posemo), sd(thread5$negemo))

# plotting a bar plot graph of the standrad deviations of both the Thread1 and Forum to find a relation.
```

```
# setting the sd_thread1 and sd_forum as vectors for plotting

thread5_forum_vec <- c(sd_forum, sd_thread5) type

<- c(rep("sd_forum", 12), rep("sd_thread5", 12))

# converting data to data frame

my_data5 <- data.frame(category, thread5_forum_vec)


# plot

graph_plot5 <- ggplot(my_data5, aes(category, thread5_forum_vec))

graph_plot5 <- graph_plot5 + geom_bar(stat = "identity", aes(fill = type), position = "dodge")

graph_plot5 <- graph_plot5 + labs(x = "Categories", y = "Standard Deviation", title = "Relation of Standard
Deviation between Forum and ThreadID 823462") graph_plot5



#-------------------------------------------------------------------------------------------------------#

# finding the mean of each thread variables

# thread 1

mean_thread1 <- c(mean(thread1$Analytic), mean(thread1$Clout), mean(thread1$Authentic),
mean(thread1$Tone),

        mean(thread1$posemo), mean(thread1$negemo), mean(thread1$anx), mean(thread1$anger),

mean(thread1$social), mean(thread1$family), mean(thread1$friend), mean(thread1$work),

mean(thread1$leisure), mean(thread1$home), mean(thread1$money), mean(thread1$relig),

mean(thread1$swear), mean(thread1$QMark))

#View(mean_thread1)


# thread 2

mean_thread2 <- c(mean(thread2$Analytic), mean(thread2$Clout), mean(thread2$Authentic),
mean(thread2$Tone),

        mean(thread2$posemo), mean(thread2$negemo), mean(thread2$anx), mean(thread2$anger),

mean(thread2$social), mean(thread2$family), mean(thread2$friend), mean(thread2$work),

mean(thread2$leisure), mean(thread2$home), mean(thread2$money), mean(thread2$relig),

mean(thread2$swear), mean(thread2$QMark))

#View(mean_thread2)
```

```
# thread 3
mean_thread3 <- c(mean(thread3$Analytic), mean(thread3$Clout), mean(thread3$Authentic), mean(thread3$Tone),
        mean(thread3$posemo), mean(thread3$negemo), mean(thread3$anx), mean(thread3$anger),
mean(thread3$social), mean(thread3$family), mean(thread3$friend), mean(thread3$work),
mean(thread3$leisure), mean(thread3$home), mean(thread3$money), mean(thread3$relig),
mean(thread3$swear), mean(thread3$QMark))
#View(mean_thread3)

# thread 4
mean_thread4 <- c(mean(thread4$Analytic), mean(thread4$Clout), mean(thread4$Authentic), mean(thread4$Tone),
        mean(thread4$posemo), mean(thread4$negemo), mean(thread4$anx), mean(thread4$anger),
mean(thread4$social), mean(thread4$family), mean(thread4$friend), mean(thread4$work),
mean(thread4$leisure), mean(thread1$home), mean(thread1$money), mean(thread1$relig),
mean(thread4$swear), mean(thread4$QMark))
#View(mean_thread4)

# thread 5
mean_thread5 <- c(mean(thread5$Analytic), mean(thread5$Clout), mean(thread5$Authentic), mean(thread5$Tone),
        mean(thread5$posemo), mean(thread5$negemo), mean(thread5$anx), mean(thread5$anger),
mean(thread5$social), mean(thread5$family), mean(thread5$friend), mean(thread5$work),
mean(thread5$leisure), mean(thread5$home), mean(thread5$money), mean(thread5$relig),
mean(thread5$swear), mean(thread5$QMark))
#View(mean_thread5)

# Making a Heat Map to find out which thread is conversing about what. The mean will help to realize what people (in average) are talking about  # a certain topic.

Mean_Threads <- cbind(mean_thread1,mean_thread2,mean_thread3, mean_thread4, mean_thread5)
data_matrix <- data.matrix(Mean_Threads)
```

```
# changing column names
colnames(data_matrix)[1] <- c("Thread 1") colnames(data_matrix)[2]
<- c("Thread 2") colnames(data_matrix)[3] <- c("Thread 3")
colnames(data_matrix)[4] <- c("Thread 4") colnames(data_matrix)[5]
<- c("Thread 5")
#changing row names
rownames(data_matrix) <- c("Analytic", "Clout", "Authentic", "Tone", "posemo", "negemo", "anx", "anger",
                           "social", "family", "friend", "work", "leisure", "home", "money", "relig", "swear", "QMark")



#View(data_matrix)


my_group=as.numeric(as.factor(substr(rownames(data_matrix), 1 , 1))) my_col=brewer.pal(9,
"YlOrRd")[my_group]
heatmap(data_matrix,Colv=NA,Rowv=NA,col=colorRampPalette(brewer.pal(9,"YlOrRd"))(100),
      ylab = "Language Attributes", main = "Relation of the attributes between the Threads",
RowSideColors=my_col)


#-----------------------------------------------------------------------------------------------------# #
Investigating of number of posts in an online forum over time


# finding the number of posts in each year
no_post1 <- nrow(data_year_two)
no_post2 <- nrow(data_year_three)
no_post3 <- nrow(data_year_four)
no_post4 <- nrow(data_year_five)
no_post5 <- nrow(data_year_six) no_post6
<- nrow(data_year_seven) no_post7 <-
nrow(data_year_eight) no_post8 <-
nrow(data_year_nine) no_post9 <-
nrow(data_year_ten) no_post10 <-
```

```
nrow(data_year_eleven) total_no_of_posts
<-          c(no_post1,
no_post2,no_post3,no_post4,no_post5,no_post6,no_post7,no_post8,no_post9,no_post10)

#plot a bar graph

posts <- data.frame(date,total_no_of_posts) posts
df2 <- data.frame(date, total_no_of_posts = rep(0, 10), Category2 = rep("", 10)) df2
posts$Category2 <- paste0(posts$date," ")
posts$Category3 <- paste0(total_no_of_posts," ") posts
# append number to category name
#posts <- rbind(posts, df2) posts

ggplot(posts, aes(x = date, y = total_no_of_posts,fill = Category2)) +
geom_bar(width = 0.9, stat="identity") +    coord_polar(theta = "y") +
xlab("") + ylab("") +   ylim(c(0,4800)) +
  ggtitle("Total Number of Posts from 2002 - 2011") +
geom_text(data = posts, hjust = 1, size = 2.5,
aes(x = date, y = 0, label = Category2)) +
geom_text(data = posts, hjust = 1, size = 2.5,
          aes(x = date, y = total_no_of_posts + 180, label = Category3), angle = 75) +
theme_minimal() +   theme(legend.position = "none",         panel.grid.major =
element_blank(),         panel.grid.minor = element_blank(),         axis.line =
element_blank(),         axis.text.y = element_blank(),         axis.text.x =
element_blank(),

       axis.ticks = element_blank())

#-------------------------------------------------------------------------------------------------#
# Investigating whether the language used changes over time
```

```
# setting the tidy data as forum for easier understanding  forum
<- data_clean
# converting the date to date format
date <- forum$Date date <-
ymd(date)
View (forum)
#date <- as.Date(date, format="%d.%m.%y") forum
<- cbind(forum, date)

# filtering the rows regarding the specified date.
data_year_two <- with(forum, forum[(forum$date > "2001-12-31") & forum$date < "2003-01-01", ])

#View(data_year_two)
data_year_three <- with(forum, forum[(forum$date > "2002-12-31") & forum$date < "2004-01-01", ])
#View(data_year_three)
data_year_four <- with(forum, forum[(forum$date > "2003-12-31") & forum$date < "2005-1-01", ])
#View(data_year_four)
data_year_five <- with(forum, forum[(forum$date > "2004-12-31") & forum$date < "2006-1-01", ])
#View(data_year_five)
data_year_six <- with(forum, forum[(forum$date > "2005-12-31") & forum$date < "2007-1-01", ])
#View(data_year_six)
data_year_seven <- with(forum, forum[(forum$date > "2006-12-31") & forum$date < "2008-1-01", ])
#View(data_year_seven)
data_year_eight <- with(forum, forum[(forum$date > "2007-12-31") & forum$date < "2009-1-01", ])
#View(data_year_eight)
data_year_nine <- with(forum, forum[(forum$date > "2008-12-31") & forum$date < "2010-1-01", ])
#View(data_year_nine)
data_year_ten <- with(forum, forum[(forum$date > "2009-12-31") & forum$date < "2011-1-01", ])
#View(data_year_ten)
data_year_eleven <- with(forum, forum[(forum$date > "2010-12-31") & forum$date < "2012-1-01", ])
#View(data_year_eleven)
```

```
# counting the mean number of we used over the years 2002 - 2011
# making a function which calculates the mean of all the attributes in each year
attributes <- function(y2,y3,y4,y5,y6,y7,y8,y9,y10,y11){
total_no._used_year_two <- mean(y2)   total_no._used_year_three <-
mean(y3)   total_no._used_year_four <- mean(y4)   total_no._used_year_five
<- mean(y5)   total_no._used_year_six <- mean(y6)
total_no._used_year_seven <- mean(y7)   total_no._used_year_eight <-
mean(y8)   total_no._used_year_nine <- mean(y9)   total_no._used_year_ten
<- mean(y10)   total_no._used_year_eleven <- mean(y11)
attr_vector <-
c(total_no._used_year_two,total_no._used_year_three,total_no._used_year_four,total_no._used_year_five,
total_no._used_year_six,
total_no._used_year_seven,total_no._used_year_eight,total_no._used_year_nine,
total_no._used_year_ten,total_no._used_year_eleven)   attr_vector
}

# arranging data to plot
date <- c("2002", "2003", "2004", "2005", "2006", "2007", "2008", "2009", "2010", "2011")

# setting all the attributes mean results in a vector
We <- attributes(data_year_two$we,
data_year_three$we,data_year_four$we,data_year_five$we,data_year_six$we,data_year_seven$we,data_y
ear_eight$we,data_year_nine$we, data_year_ten$we,data_year_eleven$we)
We
You <- attributes(data_year_two$you,
data_year_three$you,data_year_four$you,data_year_five$you,data_year_six$you,data_year_seven$you,dat
a_year_eight$you,data_year_nine$you, data_year_ten$you,data_year_eleven$you)
You
I <- attributes(data_year_two$i,
data_year_three$i,data_year_four$i,data_year_five$i,data_year_six$i,data_year_seven$i,data_year_eight$i,
data_year_nine$i,
data_year_ten$i,data_year_eleven$i)
```

I

They <- attributes(data_year_two$they, data_year_three$they,data_year_four$they,data_year_five$they,data_year_six$they,data_year_seven$they, data_year_eight$they,data_year_nine$they, data_year_ten$they,data_year_eleven$they)

They

SheHe <- attributes(data_year_two$shehe, data_year_three$shehe,data_year_four$shehe,data_year_five$shehe,data_year_six$shehe,data_year_seven$shehe,data_year_eight$shehe,data_year_nine$shehe, data_year_ten$shehe,data_year_eleven$shehe)

SheHe

Ppron <- attributes(data_year_two$ppron, data_year_three$ppron,data_year_four$ppron,data_year_five$ppron,data_year_six$ppron,data_year_seven$ppron,data_year_eight$ppron,data_year_nine$ppron, data_year_ten$ppron,data_year_eleven$ppron)

Ppron

Auth <- attributes(data_year_two$Authentic, data_year_three$Authentic,data_year_four$Authentic,data_year_five$Authentic,data_year_six$Authentic,data_year_seven$Authentic,data_year_eight$Authentic,

data_year_nine$Authentic,data_year_ten$Authentic,data_year_eleven$Authentic) Auth

Analy <- attributes(data_year_two$Analytic, data_year_three$Analytic,data_year_four$Analytic,data_year_five$Analytic,data_year_six$Analytic,data_year_seven$Analytic,data_year_eight$Analytic,

data_year_nine$Analytic,data_year_ten$Analytic,data_year_eleven$Analytic)

Analy

tone <- attributes(data_year_two$Tone, data_year_three$Tone,data_year_four$Tone,data_year_five$Tone,data_year_six$Tone,data_year_seven$Tone,data_year_eight$Tone,

data_year_nine$Tone,data_year_ten$Tone,data_year_eleven$Tone) tone

QMark <- attributes(data_year_two$QMark, data_year_three$QMark,data_year_four$QMark,data_year_five$QMark,data_year_six$QMark,data_year_seven$QMark,data_year_eight$QMark,

data_year_nine$QMark,data_year_ten$QMark,data_year_eleven$QMark) QMark

Posemo <- attributes(data_year_two$posemo,

```
data_year_three$posemo,data_year_four$posemo,data_year_five$posemo,data_year_six$posemo,data_year_seven$posemo,data_year_eight$posemo,

        data_year_nine$posemo,data_year_ten$posemo,data_year_eleven$posemo) Posemo


Negemo <- attributes(data_year_two$negemo, data_year_three$negemo,data_year_four$negemo,data_year_five$negemo,data_year_six$negemo,data_year_seven$negemo,data_year_eight$negemo,

        data_year_nine$negemo,data_year_ten$negemo,data_year_eleven$negemo) Negemo


swear <- attributes(data_year_two$swear, data_year_three$swear,data_year_four$swear,data_year_five$swear,data_year_six$swear,data_year_seven$swear ,data_year_eight$negemo,

        data_year_nine$swear,data_year_ten$swear,data_year_eleven$swear) swear


Clout <- attributes(data_year_two$Clout, data_year_three$Clout,data_year_four$Clout,data_year_five$Clout,data_year_six$Clout,data_year_seven$Clout ,data_year_eight$Clout,

        data_year_nine$Clout,data_year_ten$Clout,data_year_eleven$Clout) Clout

type <- c(rep("We", 1)) type2

<- c(rep("I", 1)) type3 <-

c(rep("You", 1)) type4 <-

c(rep("They", 1)) type5 <-

c(rep("SheHe", 1)) type6 <-

c(rep("Ppron", 1)) type7 <-

c(rep("Auth", 1)) type8 <-

c(rep("Analy", 1)) type9 <-

c(rep("tone", 1)) type10 <-

c(rep("Posemo", 1)) type11 <-

c(rep("Negemo", 1)) type12 <-

c(rep("QMark", 1)) type13 <-

c(rep("Clout", 1))


# converting the vectors to data frame format for ggplot
```

```
df<-data.frame(date,We,I, You, They, SheHe, Ppron, Auth, Analy, Posemo, Negemo, tone, QMark, Clout)
#View(df)
# graph to find out the change in the use of colums  time_graph1
<- ggplot(df, aes(date)) +
        geom_smooth(mapping = aes(y = df$Auth, group = 1, colour = type7)) +
geom_smooth(mapping = aes(y = df$Analy, group = 1, colour = type8)) +
geom_smooth(mapping = aes(y = df$Clout, group = 1, colour = type13)) +        geom_smooth(aes(y
= df$tone, group = 1, colour = type9)) +
        xlab("Years")  +     ylab("Mean")     +     guides(fill=guide_legend(title=NULL))     +
theme(legend.title=element_blank()) time_graph1



time_graph3 <- ggplot(df, aes(date)) +
  geom_smooth(mapping = aes(y = df$I, group = 1, colour = type2)) +
geom_smooth(aes(y = df$We, group = 1, colour = type)) +   geom_smooth(aes(y
= df$You, group = 1, colour = type3)) +    geom_smooth(aes(y = df$They, group
= 1, colour = type4)) + geom_smooth(aes(y = df$SheHe, group = 1, colour =
type5)) +

  geom_smooth(aes(y = df$Ppron, group = 1, colour = type6)) +
geom_smooth(mapping = aes(y = df$QMark, group = 1, colour = type12)) +
  xlab("Years") + ylab("Mean") + guides(fill=guide_legend(title=NULL)) + theme(legend.title=element_blank())
time_graph3

# Investigating the proportion of language expressing optimism changes over time
time_graph2 <- ggplot(df, aes(date)) + geom_line(aes(y = df$Posemo, group = 1, colour = type10)) +
geom_line(aes(y = df$Negemo, group = 1, colour = type11)) +
  xlab("Years") + ylab("Mean") + guides(fill=guide_legend(title=NULL)) + theme(legend.title=element_blank())
time_graph2

#Correlation of Tone with other variables cor(data_clean[11:32],
data_clean[10])
```

```
# investigating the increase of QMarks in 2011 by breaking it down in months
data_clean[33] = format(as.Date(data_clean$Date, format="%Y-%m-%d"), "%Y") #filter year
data_clean[34] = format(as.Date(data_clean$Date, format="%Y-%m-%d"), "%m") #filter months year11
<- subset(data_clean, data_clean[33] == "2011")
Q <- aggregate(year11[32], year11[34], mean) Q
<- Q[2]
typeQ <- c(rep("QMark", 1))
months <- c("01", "02", "03", "04", "05", "06", "07", "08", "09", "10", "11", "12") df<-data.frame(Q,
months)
time_graph5 <- ggplot(df, aes(months)) +
  geom_smooth(aes(y = df$QMark, group = 1, colour = typeQ)) +
  xlab("Months") + ylab("Mean") + guides(fill=guide_legend(title=NULL)) +
theme(legend.title=element_blank()) time_graph5
```

```
#-------------------------------------------------------------------------------------------------#
#Investigating the variables affecting Posemo and Negemo
#Correlation of posemo and negemo cor(data_clean[21:31],
data_clean[19:20])

# taking four threads for comparison of optimism between groups
mean_thread12 <- c(mean(thread1$posemo), mean(thread1$negemo), mean(thread1$anx),
mean(thread1$anger),
mean(thread1$social))

mean_thread24 <- c(mean(thread2$posemo), mean(thread2$negemo), mean(thread2$anx),
mean(thread2$anger),
mean(thread2$social))
```

```
mean_thread35 <- c(mean(thread3$posemo), mean(thread3$negemo), mean(thread3$anx), mean(thread3$anger),
mean(thread3$social))

mean_thread46 <- c(mean(thread4$posemo), mean(thread4$negemo), mean(thread4$anx), mean(thread4$anger),
mean(thread4$social))



#-------------------------------------------------------------------------------------------------------------#
mean <- data.frame(cbind(mean_thread12,mean_thread24,mean_thread35, mean_thread46)) #
combine the mean thread columns into a single column with separate rows for each thread
data_long <- gather(mean,Threads,value)
proportions <- c("posemo", "negemo", "anx", "anger", "social")
# Create dataset data=data.frame(
  individual=paste(rep(proportions,4), sep=""),
  group=c( rep('Thread 1', 5), rep('Thread 2', 5), rep('Thread 3', 5), rep('Thread 4', 5)) ,
  value=data_long$value
)
# Setting a number of 'empty bar' to add at the end of each group empty_bar=3
to_add = data.frame( matrix(NA, empty_bar*nlevels(data$group), ncol(data)) ) colnames(to_add)
= colnames(data)
to_add$group=rep(levels(data$group), each=empty_bar)
data=rbind(data, to_add) data=data %>%
arrange(group) data$id=seq(1, nrow(data))

# Get the name and the y position of each label label_data=data
number_of_bar=nrow(label_data)
angle= 90 - 360 * (label_data$id-0.5) /number_of_bar     # substract 0.5 because the letter must have the angle
of the center of the bars. Not extreme right(1) or extreme left (0) label_data$hjust<-ifelse( angle < -90, 1, 0)
label_data$angle<-ifelse(angle < -90, angle+180, angle)
```

```
# prepare a data frame for base lines
base_data=data %>%
group_by(group) %>%
  summarize(start=min(id), end=max(id) - empty_bar) %>%
rowwise() %>%
  mutate(title=mean(c(start, end)))

# prepare a data frame for grid (scales) grid_data
= base_data
grid_data$end = grid_data$end[ c( nrow(grid_data), 1:nrow(grid_data)-1)] + 1
grid_data$start = grid_data$start - 1 grid_data=grid_data[-1,]

# Make the plot
p = ggplot(data, aes(x=as.factor(id), y=value, fill=group)) +       # Note: id is a factor. If x is numeric, there is some space between the first bar

  geom_bar(aes(x=as.factor(id), y=value, fill=group), stat="identity", alpha=1) +

  # Add a val=15/10/5/0 lines. I do it at the beginning to make sure barplots are OVER it.
  geom_segment(data=grid_data, aes(x = end, y = 15, xend = start, yend = 15), colour = "grey", alpha=1, size=0.3 , inherit.aes = FALSE ) +
  geom_segment(data=grid_data, aes(x = end, y = 10, xend = start, yend = 10), colour = "grey", alpha=1, size=0.3 , inherit.aes = FALSE ) +
  geom_segment(data=grid_data, aes(x = end, y = 5, xend = start, yend = 5), colour = "grey", alpha=1, size=0.3
, inherit.aes = FALSE ) +
  geom_segment(data=grid_data, aes(x = end, y = 0, xend = start, yend = 0), colour = "grey", alpha=1, size=0.3 , inherit.aes = FALSE ) +
  # Add text showing the value of each 15/10/5/0 lines
  annotate("text", x = rep(max(data$id),4), y = c(15, 10, 5, 0), label = c("15", "10", "5", "0") , color="grey", size=4 , angle=0, fontface="bold", hjust=1) +

  geom_bar(aes(x=as.factor(id), y=value, fill=group), stat="identity", alpha=1) +
ylim(-18,18) +   theme_minimal() +   theme(
```

```
    legend.position = "none",
axis.text = element_blank(),
axis.title = element_blank(),
panel.grid = element_blank(),
plot.margin = unit(rep(-1,10), "cm")
  ) +
  coord_polar() +
  geom_text(data=label_data, aes(x=id, y=value+2, label=paste0(individual, "- ", round(value,2)), hjust=hjust), color="black", fontface="bold",alpha=1, size=3, angle= label_data$angle, inherit.aes = FALSE ) +


  # Add base line information
  geom_segment(data=base_data, aes(x = start, y = -5, xend = end, yend = -5), colour = "black", alpha=0.8, size=0.6 , inherit.aes = FALSE )  +
  geom_text(data=base_data, aes(x = title, y = -2, label=group), hjust=c(1,1,0,0), colour = "black", alpha=0.8, size=4, fontface="bold", inherit.aes = FALSE)


p
```