Text analysis 3

Junyan Yao

10/16/2017

Research question: Whether chat engagement is associated with test outcomes(see how students' collaborative performance can be associated with students' performance in these math problems).

Data: Chat data and test outcome data

```
library(corpus)
library(Matrix)
```

Load data

```
data<- read.csv("~/Documents/NYU/Fall 2017/Text Analysis Project/cpsv_text_project/chat_
time_series.csv")
data<- data[,c(2,5,8)] #extract needed column
head(data)
```

```
##   group_id type                                                   content
## 1        1 chat                                 So how should we do this?
## 2        1 chat So I guess one of us should pick c and one should pick a?
## 3        1 chat                                                       Yes
## 4        1 chat                                                Ill pick a
## 5        1 chat                                               I'll take a
## 6        1 chat                                                    c then
```

```
#subset the data
chatdata<- data[which(data$type=="chat"),] #this is what we want to look at for now
problemdata<- data[which(data$type=="problem"),]
head(chatdata)
```

```
##   group_id type                                                   content
## 1        1 chat                                 So how should we do this?
## 2        1 chat So I guess one of us should pick c and one should pick a?
## 3        1 chat                                                       Yes
## 4        1 chat                                                Ill pick a
## 5        1 chat                                               I'll take a
## 6        1 chat                                                    c then
```

```
#load the outcome data
outcomedata<-read.csv("~/Documents/NYU/Fall 2017/Text Analysis Project/cpsv_text_projec
t/group_outcomes.csv")

head(outcomedata)
```

```
##   X group_id         w       delta
## 1 1      -53 0.9255376  0.09307671
## 2 2      -52 0.4795482  0.40842891
## 3 3      -51 0.9904785  1.01937085
## 4 4      -50 0.9254073  0.66388004
## 5 5      -49 0.9865247  0.65585781
## 6 6      -48 0.9176420 -0.28448291
```

```
subset1<- outcomedata[outcomedata$group_id>0,] #this will get rid of all negative group_
id

summary(subset1$delta)
```

```
##      Min.   1st Qu.    Median      Mean   3rd Qu.      Max.
## -2.698000 -0.481000 -0.001394 -0.043190  0.405800  3.310000
```

```
performance<-ifelse(subset1$delta>0.4058,"high",ifelse(subset1$delta< -0.481,"low","in-b
etween"))
temp22<-cbind(subset1,performance)
#try to get rid of the missing rows(some group id are missing in the outcome data)
merged_data<- merge(x=chatdata,y=temp22,by="group_id")

#now back to these two dataset, treat the merged_data as the chat data.
#chatdata2<- merged_data[,1:3]
#head(chatdata2)
```

Tokenlize data

```
#split to two groups- High performance group and low performance group;
high_group<- merged_data[which(merged_data$performance=="high"),]
low_group<- merged_data[which(merged_data$performance=="low"),]

#get the most common non-punctuation, non-stop word terms in the chat
Y<- term_stats(merged_data$content, drop=stopwords_en, drop_punct=TRUE) #the support is
 the number of texts containing the term.
# by using drop= stopwords_en, we can exclude these "functional" words
head(Y, 10)
```

```
##    term   count support
## 1  ok      1043    1040
## 2  one      850     830
## 3  =        718     631
## 4  next     599     596
## 5  think    557     553
## 6  answer   504     497
## 7  yes      490     489
## 8  2        485     442
## 9  x        467     438
## 10 1        450     417
```

```
Y_high<- term_stats(high_group$content,drop=stopwords_en, drop_punct=TRUE)
Y_low<- term_stats(low_group$content,drop=stopwords_en, drop_punct=TRUE)

S<- subset(Y, Y$support>5)
S_high<-subset(Y_high,Y_high$support>5)
S_low<-subset(Y_low,Y_low$support>5)

#probably not drop the "functional" words
YY<- term_stats(merged_data$content)
head(YY,10)
```

```
##    term count support
## 1  i     2802    2558
## 2  ?     2445    2339
## 3  the   2595    2050
## 4  .     3155    1782
## 5  is    1798    1661
## 6  ,     1905    1537
## 7  to    1260    1135
## 8  you   1131    1097
## 9  ok    1043    1040
## 10 it    1020     957
```

```
YY_high<- term_stats(high_group$content)
head(YY_high,10)
```

```
##    term count support
## 1  i      864     797
## 2  ?      777     752
## 3  the    816     638
## 4  .     1016     558
## 5  ,      694     555
## 6  is     534     489
## 7  you    368     356
## 8  to     401     352
## 9  a      334     303
## 10 and    327     303
```

```
YY_low<-term_stats(low_group$content)
head(YY_low, 10)
```

```
##    term count support
## 1  i      586     517
## 2  ?      529     498
## 3  the    562     454
## 4  .      694     404
## 5  is     417     384
## 6  ,      474     372
## 7  ok     255     254
## 8  you    251     240
## 9  to     257     232
## 10 it     238     223
```

```
#higher-order n-grams
term_stats(merged_data$content,ngrams = 3)
```

```
##    term               count support
## 1  . . .                416     356
## 2  the next one         135     135
## 3  i have no             89      89
## 4  the value of          86      81
## 5  on this one           76      76
## 6  have no idea          74      74
## 7  for the next          71      71
## 8  i don't know          67      67
## 9  total number of       77      63
## 10 the answer is         63      63
## 11 what is the           59      59
## 12 go with that          53      53
## 13 value of x            51      50
## 14 number of boxes       48      46
## 15 . . i                 46      46
## 16 what do you           44      44
## 17 number of students    44      43
## 18 in class a            46      42
## 19 i think the           42      42
## 20 next one is           42      42
## ⋮  (39129 rows total)
```

```
term_stats(merged_data$content,ngrams = 4)
```

```
##    term                     count support
## 1  i have no idea              64      64
## 2  for the next one            37      37
## 3  . . . .                     39      35
## 4  total number of students    32      31
## 5  . . . i                     29      29
## 6  total number of boxes       28      27
## 7  number of students in       27      26
## 8  sounds good to me           26      26
## 9  the total number of         31      24
## 10 i have no clue              24      24
## 11 i think the answer          24      24
## 12 what do you think           24      24
## 13 the next one ?              23      23
## 14 think the answer is         23      23
## 15 are you on the              22      22
## 16 sold in class a             21      21
## 17 i think it is               20      20
## 18 on the next one             20      20
## 19 number of boxes of          21      19
## 20 . what is the               19      19
## ⋮  (40400 rows total)
```

```
term_stats(merged_data$content,ngrams = 5)
```

```
##    term                        count support
## 1  i think the answer is          21      21
## 2  total number of students in    21      20
## 3  number of boxes of cookies     20      18
## 4  i don't know how to            17      17
## 5  of boxes of cookies sold       18      16
## 6  number of students in class    17      16
## 7  what do you think ?            16      16
## 8  boxes of cookies sold in       17      15
## 9  of cookies sold in class       17      15
## 10 of students in class a         15      15
## 11 for the next one ?             14      14
## 12 the total number of boxes      14      13
## 13 total number of boxes of       14      13
## 14 no idea on this one            13      13
## 15 have no idea how to            12      12
## 16 no clue on this one            12      12
## 17 what is the value of           12      12
## 18 - 3 , - 1                      11      11
## 19 cookies sold in class a        11      11
## 20 i have no idea how             11      11
## ⋮  (36130 rows total)
```

```
term_stats(high_group$content,ngrams = 4)
```

```
##    term                      count support
## 1  i think the answer           15      15
## 2  think the answer is          14      14
## 3  . . . i                      13      13
## 4  on this one .                13      13
## 5  for the next one             12      12
## 6  go with that .               11      11
## 7  let's go with that           11      11
## 8  what do you think            11      11
## 9  total number of boxes        11      10
## 10 number of students in        10      10
## 11 the value of a               10      10
## 12 total number of students     10      10
## 13 i have no clue                9       9
## 14 one . . .                     9       9
## 15 the next one ?                9       9
## 16 nice working with you         8       8
## 17 sounds good to me             8       8
## 18 that's what i got             8       8
## 19 the total number of          10       7
## 20 + 18 = 0                      7       7
## ⋮  (13701 rows total)
```

```
term_stats(low_group$content,ngrams = 4)
```

```
##    term                    count support
## 1  i have no idea             12      12
## 2  . . . .                    11       9
## 3  ok . . .                    8       8
## 4  the next one is             8       8
## 5  lets go with that           7       7
## 6  boxes of cookies sold       6       6
## 7  cookies sold in class       6       6
## 8  don't know how to           6       6
## 9  how to do this              6       6
## 10 i am on the                 6       6
## 11 i don't know how            6       6
## 12 nice working with you       6       6
## 13 on the next one             6       6
## 14 on the next question        6       6
## 15 on to the next              6       6
## 16 . . . i                     5       5
## 17 . what is the               5       5
## 18 average number of boxes     5       5
## 19 have no idea how            5       5
## 20 i think it is               5       5
## ⋮  (9932 rows total)
```

Emotion-lexicon

```
#Emotion-Lexicon
affect<- subset(affect_wordnet,emotion != "Neutral")
affect$emotion<- droplevels(affect$emotion) #drop the unused neutral level
affect$category<- droplevels(affect$category) #drop unused categories

term_stats(merged_data$content, subset = term %in% affect$term)
```

```
##    term      count support
## 1  good        284     281
## 2  sorry       160     160
## 3  like        139     139
## 4  submit       90      89
## 5  still        82      80
## 6  cool         78      77
## 7  move         76      76
## 8  great        64      63
## 9  easy         50      45
## 10 bad          37      37
## 11 hope         36      36
## 12 down         21      20
## 13 hopefully    18      18
## 14 close        16      16
## 15 hate         16      16
## 16 positive     16      16
## 17 trust        15      15
## 18 confused     14      14
## 19 confusing    14      14
## 20 care         12      12
## ⋮  (74 rows total)
```

```
term_stats(high_group$content, subset = term %in% affect$term)
```

```
##    term      count support
## 1  good        100      97
## 2  like         47      47
## 3  sorry        43      43
## 4  submit       31      31
## 5  cool         27      27
## 6  move         26      26
## 7  great        26      25
## 8  easy         25      21
## 9  bad          13      13
## 10 still        12      12
## 11 down         10       9
## 12 hate          9       9
## 13 hope          9       9
## 14 hopefully     6       6
## 15 care          5       5
## 16 close         5       5
## 17 dear          5       5
## 18 positive      5       5
## 19 trust         5       5
## 20 weight        5       5
## ⋮  (44 rows total)
```

```
term_stats(low_group$content, subset = term %in% affect$term)
```

```
##    term      count support
## 1  good         52      52
## 2  still        41      39
## 3  sorry        36      36
## 4  submit       30      30
## 5  like         23      23
## 6  move         19      19
## 7  cool         12      12
## 8  hope         10      10
## 9  hopefully    10      10
## 10 bad           9       9
## 11 great         7       7
## 12 easy          6       5
## 13 close         3       3
## 14 horrible      3       3
## 15 positive      3       3
## 16 terrible      3       3
## 17 confused      2       2
## 18 down          2       2
## 19 score         2       2
## 20 stupid        2       2
## ⋮  (45 rows total)
```

```
text_sample(high_group$content,"hard")
```

```
##   text            before            instance            after
## 1 3718  individual part wasn't this    hard    for me
## 2 600                                  hard    question
## 3 686                                  hard    questions
## 4 372                next one looks    hard
## 5 381                next one looks    hard
```

```
text_sample(low_group$content,"hard")
```

```
##   text            before            instance            after
## 1 234                          also    hard    one
## 2 396           Thanks for all your    hard    work.
## 3 475          Some of these are so    hard    bc I can't remember how to …
## 4 198                      this one    hard
## 5 2948                    this is a    hard    one now
## 6 94   …guess the other because its    hard
```

```
#term emotion matrix
#segment the text into smaller chunks and then compute the emotion occurence rates in ea
ch chunk, broken down by category ("positive","negative","ambiguous")

term_score<- with(affect, unclass(table(term,emotion)))
head(term_score) #while not very informative
```

```
##             emotion
## term         Positive Negative Ambiguous
##   abase             0        2         0
##   abash             0        1         0
##   abashed           0        1         0
##   abashment         0        1         0
##   abhor             0        1         0
##   abhorrence        0        1         0
```

create 2 by 2 tables for each term in the chat

```
YY_high<- YY_high[,-3] #drop the support column
YY_low<- YY_low[,-3] #drop the support column
names(YY_high)[2]<- paste("high")
names(YY_low)[2]<- paste("low")
dat<- merge(YY_high,YY_low, by="term",all = TRUE)

dat[is.na(dat)]<- 0


#create 2 * 2 tables for each term
aux<- 1:length(dat$term)
x<- rep(list(diag(2)), 2677)
for (i in 1:length(aux)){
  x[[i]][1,1]<-dat$high[[i]]
  x[[i]][2,1]<-dat$low[[i]]
  x[[i]][1,2]<-colSums(dat[,c(2,3)])[1]-dat$high[[i]]
  x[[i]][2,2]<-colSums(dat[,c(2,3)])[2]-dat$low[[i]]
  colnames(x[[i]])<- c(dat$term[i], paste0("\u00ac",dat$term[i]))
  rownames(x[[i]])<- c("high", "low")
}

#one example
x[[2010]]
```

```
##      right ¬right
## high   146  26350
## low     76  17911
```

This table shows the frequency of “right” term is 146 in the high performance group, and another type is 26350. In the low performance group, the frequency is 76. The ratio below this term for these two groups are 146/76=1.92

Now we would like to explore all terms ratio between high preformance groups and low preformance groups

```
ratio<- matrix(NA,nrow=2677,ncol=2)
for (i in 1:length(x)){
  ratio[i,1]<- colnames(x[[i]])[1]
  ratio[i,2]<- x[[i]][1,1]/(x[[i]][2,1]+0.001)#add 0.01 here to avoid infinite value
}
```

Here are the rates between the term and the rest of terms

Rates=High/low

```
rates<- matrix(NA, nrow = 2677, ncol = 2)
for (i in 1:length(x)){
  rates[i,1]<-colnames(x[[i]])[1]
  rates[i,2]<- x[[i]][1,1]/(x[[i]][2,1]+0.001)
}
```

Some terms only apprear once in the dataset. This could be unreliable and not very informative. So we discard them