Week 13 Project

Charlene Zhang

4/20/2020

R Studio API Code

Libraries & Authorization

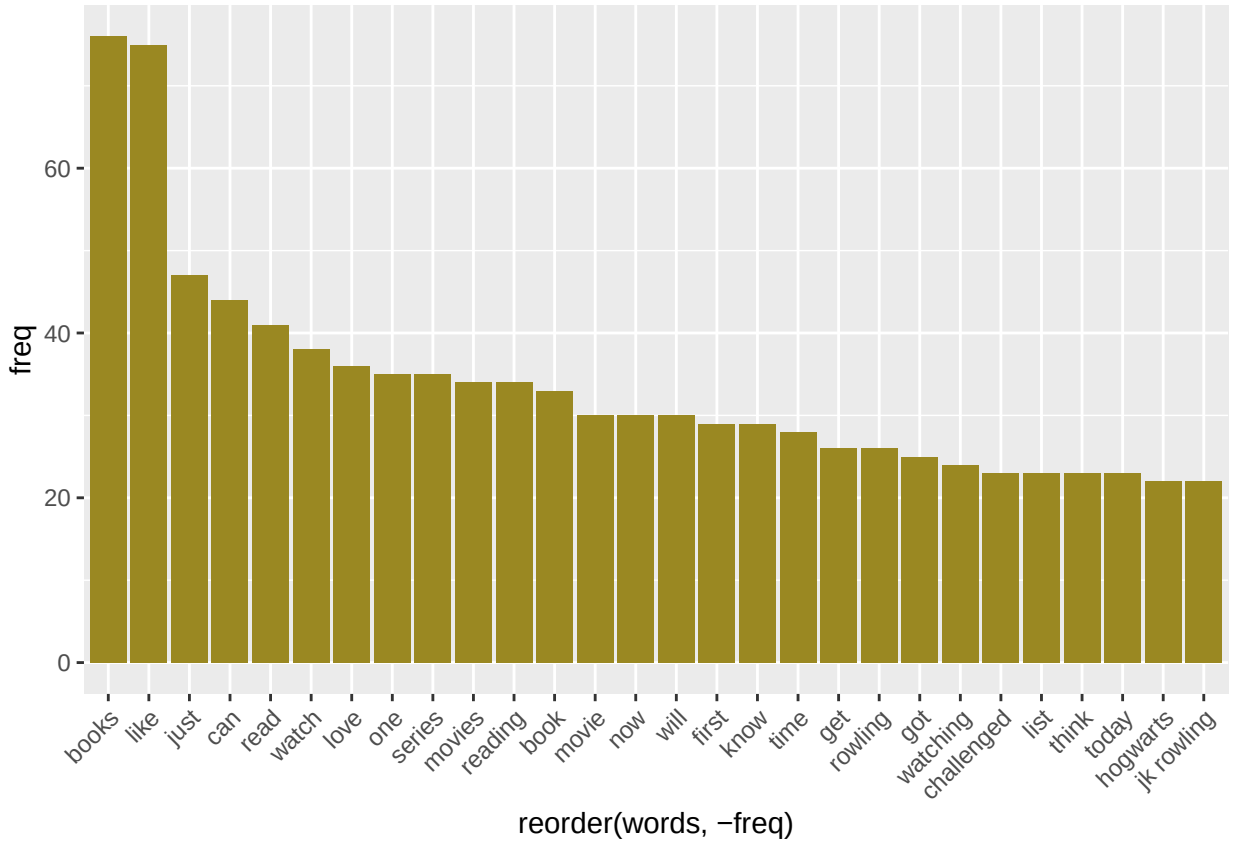
Data Import and Cleaning

Read in 1000 non-retweet tweets and removed emoticons and emojis and saved extracted data. Preprocessed tweets by * creating plain text document * replacing abbreviations and contractions with full words * turning all letter to lower case * removing any URLs, numbers, or punctuation * removing stopwords as well as the words “harry” and “potter” because they are included in every tweet by definition * stripping any white space * turning words into lemmas * converting into DTM * removing sparse terms (0 in 98% of documents) * combining with original dataset (favorite count) and removing cases with no tokens retained

Visualization

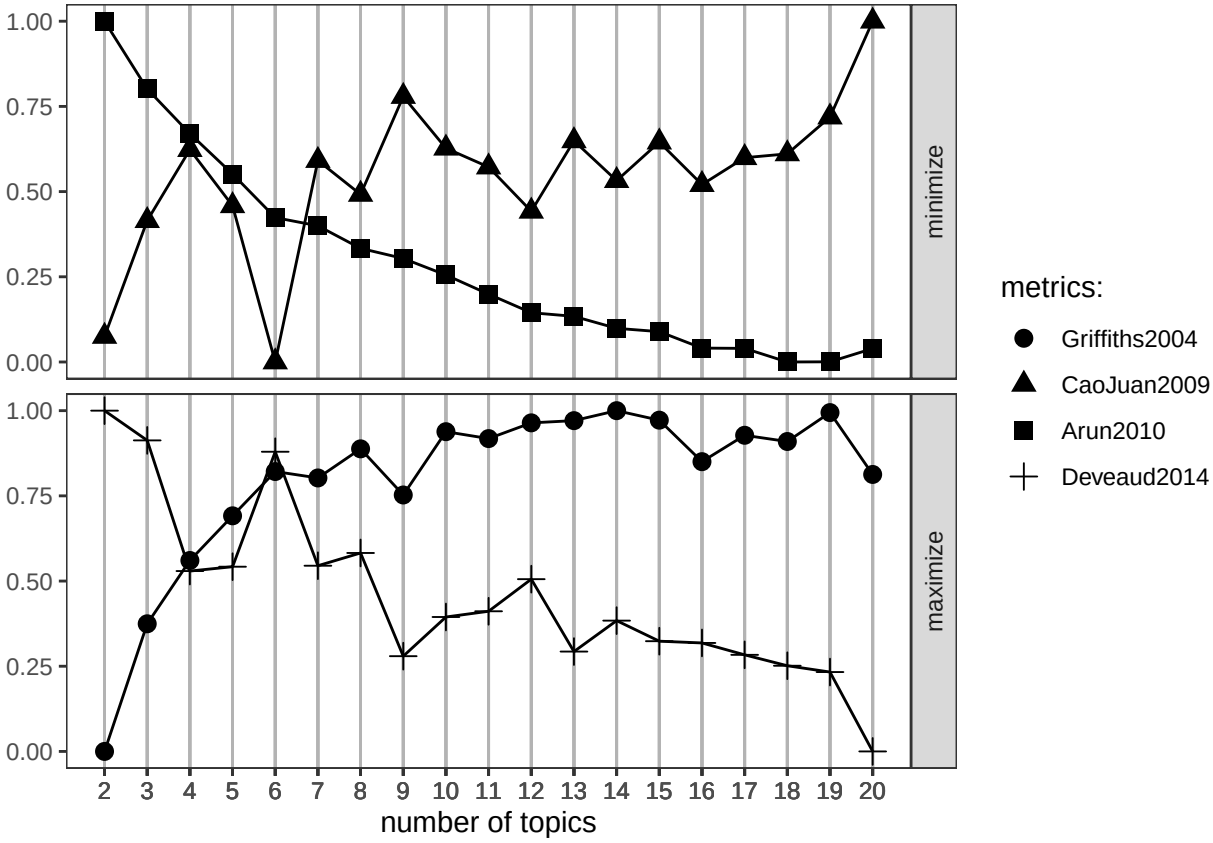
```
twitter_tbl <- dropped_tbl[,3:30]
count <- tibble(words=colnames(twitter_tbl),
                freq=apply(twitter_tbl,2,sum)) %>%
  arrange(desc(freq))
wordcloud(words=count$words,
          freq=count$freq,
          colors=wes_palette("Royal2"),
          scale=c(3,.5), # range of font sizes
          random.order=F)
```


```
ggplot(count,aes(x=reorder(words,-freq),y=freq))+
  geom_bar(stat="identity",fill=wes_palette("Royal2",1))+
  theme(axis.text.x=element_text(angle=45,hjust=1))
```



Analysis: Topic Modeling

```
tuning <- FindTopicsNumber(cleaned_dtm,
                           topic=seq(2,20,1),
                           metrics=c("Griffiths2004",
                                     "CaoJuan2009",
                                     "Arun2010",
                                     "Deveaud2014"),
                           verbose=F)
FindTopicsNumber_plot(tuning)
```



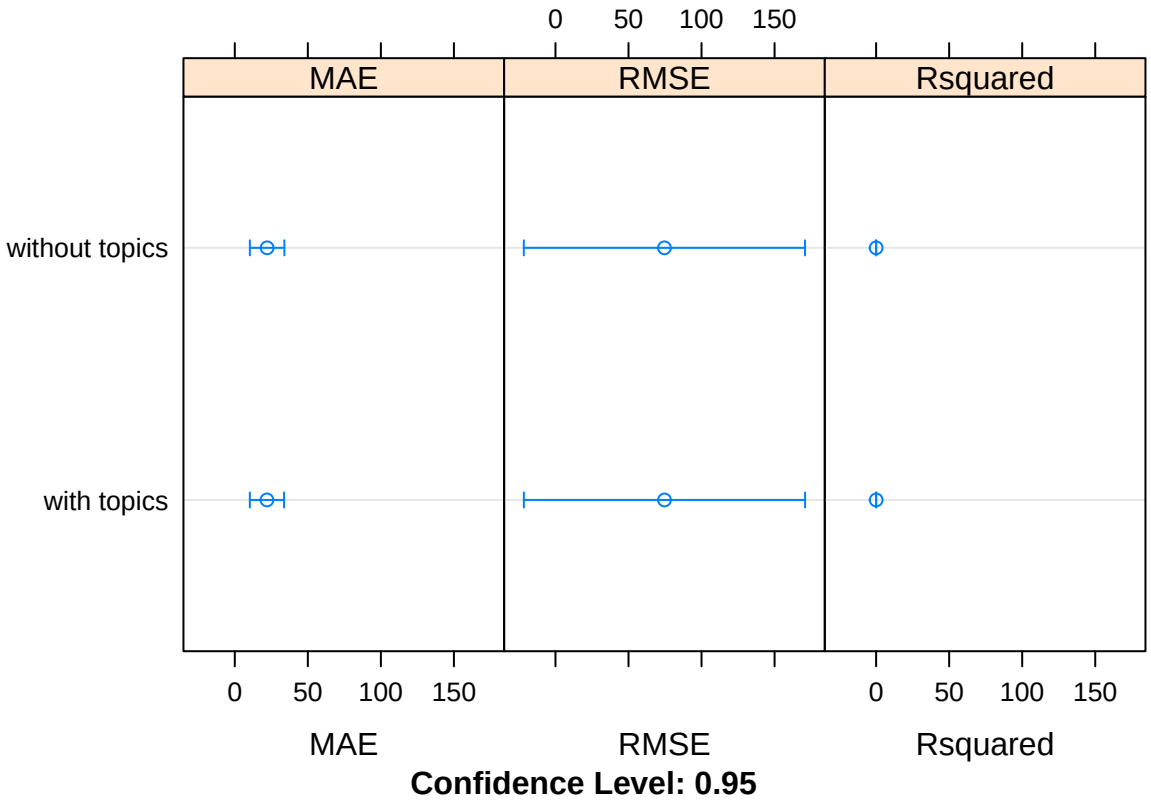

```
## run LDA
lda_10 <- LDA(cleaned_dtm,k=10)
top_terms <- terms(lda_10,10) # top 10 terms in each topic
lda_betas <- tidy(lda_10,matrix="beta") # probability that a word belongs to a topic
lda_betas %>%
  group_by(topic) %>%
  top_n(10,beta) %>%
  arrange(topic,beta) %>%
  View
lda_gammas <- tidy(lda_10,matrix="gamma") # probability that tweets contain topics
lda_gammas$document <- rep(1:530,10)
lda_gammas %>%
  group_by(topic) %>%
  top_n(10,gamma) %>%
```

```
  arrange(topic,gamma) %>%
  View # can generate topic names based on this
categories <- lda_gammas %>%
  group_by(document) %>%
  top_n(1,gamma) %>%
  slice(1) %>%
  ungroup %>%
  mutate(document=as.numeric(document)) %>%
  arrange(document) %>%
  select(topic) # determine the most popular topic for each tweet
twitter_tbl <- cbind(twitter_tbl,categories)
```

In choosing the number of topics to model, the goal is to minimize Griffiths2004 and CaoJuan2009 indices and maximize Arun2010 and Deveaud2014 indices. Therefore, 10 topics are chosen. LDA was performed to explore topics undering tweets. A variable representing the most likely topic per tweet in added.

Analysis: Machine Learning

```
twitter_tbl$favoriteCount <- dropped_tbl$favoriteCount
local_cluster <- makeCluster(detectCores()-1)
registerDoParallel(local_cluster)
svm1 <- train(
  favoriteCount~.-topic,
  twitter_tbl,
  method="svmLinear",
  trControl=trainControl(method="cv",number=10,verboseIter=F),
  na.action=na.pass
)
svm2 <- train(
  favoriteCount~.,
  twitter_tbl,
  method="svmLinear",
  trControl=trainControl(method="cv",number=10,verboseIter=F),
  na.action=na.pass
)
stopCluster(local_cluster)
dotplot(resamples(list("without topics"=svm1,"with topics"=svm2)))
```




```
summary(resamples(list("without topics"=svm1,"with topics"=svm2)))
```

```
##
## Call:
## summary.resamples(object = resamples(list(`without topics` = svm1,
##  `with topics` = svm2)))
##
## Models: without topics, with topics
## Number of resamples: 10
##
## MAE
##                    Min. 1st Qu.   Median     Mean  3rd Qu.     Max. NA's
## without topics 13.81664 14.4233 14.58461 22.11978 17.18890 63.76893    0
## with topics    14.16463 14.3704 14.52183 22.02159 17.10091 63.92158    0
##
## RMSE
##                    Min.  1st Qu.   Median     Mean  3rd Qu.     Max. NA's
## without topics 13.91313 14.52275 14.71099 74.59186 26.10783 424.2855    0
## with topics    14.38298 14.51332 14.62953 74.63913 26.07720 424.2875    0
##
## Rsquared
##                         Min.     1st Qu.      Median       Mean     3rd Qu.
## without topics 5.863767e-04 0.001241932 0.003344215 0.01472926 0.006630139
## with topics    2.355017e-05 0.000994339 0.003936787 0.01459530 0.009833912
##                      Max. NA's
## without topics 0.08093657    0
## with topics    0.09928506    0
```

Ran two support vector regression models and 10-fold CV, the first without topic and the second with topic. The model with topic assignment yielded higher Rsquared and lower RMSE and therefore performed better.

Analysis: Final Interpretation

Topics can be important in predicting tweet popularity. However, topic categorizations in this case were generated based on a fairly small number of tweets and top terms could not be easily distinguished from each other. There was also larger overlap in top terms among different topics. Results should be interpreted with caution.