Xiao_Lin_solutions_lab3

Lin Xiao

September 8, 2015

Team members: Yulin Lei, Wei(Emily) Shao Task 1

```
load("/Users/LinXIAO/Downloads/lab3.RData")
require(tm)
```

```
## Loading required package: tm
## Loading required package: NLP
```

```
length(shakespeare)
```

```
## [1] 182
```

Task 2

```
corp <- Corpus(VectorSource(shakespeare))
```

Task 3

```
corp <- tm_map(corp, content_transformer(tolower))
corp <- tm_map(corp, removePunctuation)
corp <- tm_map(corp, removeNumbers)
dtm <- as.matrix(DocumentTermMatrix(corp))
```

Task 4

```
myQuery<-c("something","rotten", "state","denmark")
```

Task 5

```
#Input would be a query and a documentname#
#Output would be a subset of normalized DTM with columns that#
#are shared with the query, and a column contains l2 distance#
myTextMiner<-function(query,corpus){
  corpus <- Corpus(VectorSource(corpus))
  corpus <- tm_map(corpus, content_transformer(tolower))
  corpus <- tm_map(corpus, removePunctuation)
  corpus <- tm_map(corpus, removeNumbers)    #All above are processing the corpus
  dtm <- as.matrix(DocumentTermMatrix(corpus)) #Turn it into a matrix
  query1<-c()
  for(i in colnames(dtm)){
    if(i %in% query){#Table function might help if there are repeated words in the query
      query1[which(colnames(dtm)==i)]<-table(query)[[i]]
    }else{
      query1[which(colnames(dtm)==i)]<-0
```

```
    }
  } #We create a query1 vector counting the words in it
  mydtm<-rbind(dtm,query1)
  mydtm.dl=mydtm/rowSums(mydtm) #Normalized by the length of each document
  #Compute the Euclidean distance
  distanceMetric<-sqrt(rowSums((scale(mydtm.dl,center=mydtm.dl[nrow(mydtm),],scale=F))^2))
  mat.l2<-cbind(mydtm.dl,distanceMetric)
  final_matrix<-mat.l2[,c(unique(query),"distanceMetric")]
  # return the subset
  return(final_matrix)
}
#And we test it with myQuery and shakespeare
ans<-myTextMiner(myQuery,shakespeare)
write.csv(ans,file="Xiao_Lin_DTM.csv")# Write it into a csv file
```