```r
# packages
install.packages("topicmodels")
install.packages("pdftools")
install.packages("tidytext")
install.packages("ggplot2")
install.packages("dplyr")
install.packages("reshape2")
library(topicmodels)
library(pdftools)
library(tm)
library(tidytext)
library(ggplot2)
library(dplyr)
library(reshape2)

# load all the PDFs
All_Files<- list.files(pattern = "pdf$")
All_opinions<- lapply(All_Files, pdf_text)

# create the corpus-the database of words
document<-Corpus(VectorSource(All_opinions))

# clean up the "document"-includes 4 files-using transformations
document <-tm_map(document, content_transformer(tolower)) # convert to all lower case
document <-tm_map(document, removeNumbers) # remove numbers
document <-tm_map(document, removePunctuation, preserve_intra_word_dashes = TRUE)
document <-tm_map(document, stripWhitespace) # remove white spaces
document <-tm_map(document, removeWords, stopwords("english")) #remove english stopwords

stopwords("english") # if you want to check what these are

# define custom stopwords to exclude
custom_stopwords <- c("tuft", "lyndsey", "nyeah", "nlyndsey", "nthe", "going", "sort", "will",
"things","nolson", "nyeahnherring","probably", "ngarcia", "gonna", "something", "wanna", "want",
"vera", "nherring", "might", "get", "can", "nso", "nand", "nramos", "lot" , "maybe", "think",
"like", "amy", "herring", "just", "kind", "fns", "sara", "know", "kristin", "say", "jaime",
"mean", "really", "yeah", "one")
#combine custom stopwords with the english ones
all_stopwords <- c(stopwords("english"), custom_stopwords)
# clean up the "document" to remove all stop words
document <-tm_map(document, removeWords, all_stopwords)

# create a document term matrix
DTM <- DocumentTermMatrix(document)

# create Latent Dirichlet allocation model with 4 topics. Set a seed to ensure reproducible
results
model_lda <- LDA(DTM, k=4, control = list(seed = 1234))
model_lda

# get beta values for per-topic-per-word probabilities
beta_topics <- tidy(model_lda, matrix = "beta") # create the beta model
beta_topics # reveal the information in beta_topics

# visualize the associations
# group the terms by topic
beta_top_terms <- beta_topics %>%
  group_by(topic) %>%
  slice_max(beta, n=10) %>%
  ungroup() %>%
  arrange(topic, -beta)

# display on a bar chart
beta_top_terms %>%
mutate(terms = reorder_within(term, beta, topic)) %>%
ggplot(aes(beta, term, fill = factor(topic))) +
geom_col(show.legend = FALSE) +
facet_wrap(~ topic, scales = "free") +
scale_y_reordered()
```