stm-okcupid

Li Liu

November 24, 2019

load packages and data

```
library(stm)
```

```
## stm v1.3.4 successfully loaded. See ?stm for help.
##  Papers, resources, and other materials at structuraltopicmodel.com
```

```
library(quanteda)
```

```
## Package version: 1.5.1
```

```
## Parallel computing: 2 of 8 threads used.
```

```
## See https://quanteda.io for tutorials and examples.
```

```
##
## Attaching package: 'quanteda'
```

```
## The following object is masked from 'package:utils':
##
##     View
```

```
library(topicmodels)
library(tidytext)
library(ggplot2)
library(dplyr)
```

```
##
## Attaching package: 'dplyr'
```

```
## The following objects are masked from 'package:stats':
##
##     filter, lag
```

```
## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
library(tidyr)
library(scales)
library(tm)
```

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

```
##
## Attaching package: 'NLP'
```

```
## The following object is masked from 'package:ggplot2':
##
##     annotate
```

```
##
## Attaching package: 'tm'
```

```
## The following objects are masked from 'package:quanteda':
##
##     as.DocumentTermMatrix, stopwords
```

```
library(grid)
library(wordcloud)
```

```
## Loading required package: RColorBrewer
```

```
library(wordcloud2)
library(tidyverse)
```

```
## -- Attaching packages ---------------------------------------------------------------------- tidyverse
```

```
## v tibble  2.1.3     v stringr 1.4.0
## v readr   1.3.1     v forcats 0.4.0
## v purrr   0.3.3
```

```
## -- Conflicts ------------------------------------------------------------------------- tidyverse_conflicts()
## x NLP::annotate()     masks ggplot2::annotate()
## x readr::col_factor() masks scales::col_factor()
## x purrr::discard()    masks scales::discard()
## x dplyr::filter()     masks stats::filter()
## x dplyr::lag()        masks stats::lag()
```

```
#sample data with essay 0 and demo clusters
essay <- read.csv('essay0.csv')
```

data wrangling

```
#convert factor type to character
essay$essay0 <- as.character (essay$essay0)

list_of_values <- c('love','people','life','time','enjoy','friends','fun', 'people','music')

'%ni%' <- Negate('%in%')

tidy_essay <- essay %>%
  mutate(kmeanscluster = factor(kmeanscluster, levels = unique(kmeanscluster)))%>%
  mutate(line = row_number()) %>%
  unnest_tokens(word, essay0) %>%
  anti_join(stop_words) %>%
  filter(word %ni% list_of_values)
```

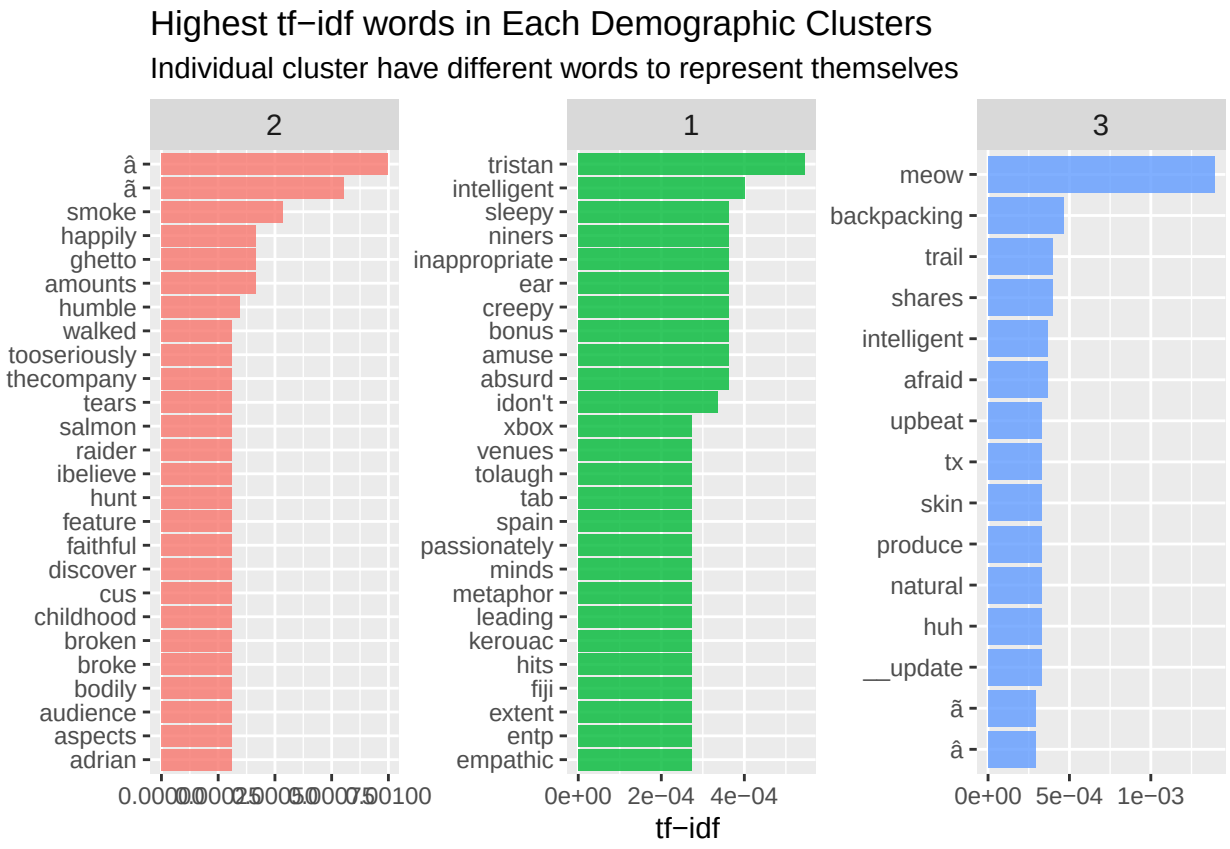
```
## Joining, by = "word"
```

exploring tf-idf

```
essay_tf_idf <- tidy_essay %>%
  count(kmeanscluster, word, sort = TRUE) %>%
  bind_tf_idf(word, kmeanscluster, n) %>%
  arrange(-tf_idf) %>%
  group_by(kmeanscluster) %>%
  top_n(15) %>%
  ungroup
```

```
## Selecting by tf_idf
```

```
essay_tf_idf %>%
  mutate(word = reorder_within(word, tf_idf, kmeanscluster)) %>%
  ggplot(aes(word, tf_idf, fill = kmeanscluster)) +
  geom_col(alpha = 0.8, show.legend = FALSE) +
  facet_wrap(~ kmeanscluster, scales = "free", ncol = 3) +
  scale_x_reordered() +
  coord_flip() +
  theme(strip.text=element_text(size=11)) +
  labs(x = NULL, y = "tf-idf",
       title = "Highest tf-idf words in Each Demographic Clusters",
       subtitle = "Individual cluster have different words to represent themselves")
```



build document-term matrix

```
essay_dfm <- tidy_essay %>%
  count(kmeanscluster, word, sort = TRUE) %>%
  cast_dfm(kmeanscluster, word, n)

essay_sparse <- tidy_essay %>%
  count(kmeanscluster, word, sort = TRUE) %>%
  cast_sparse(kmeanscluster, word, n)
```

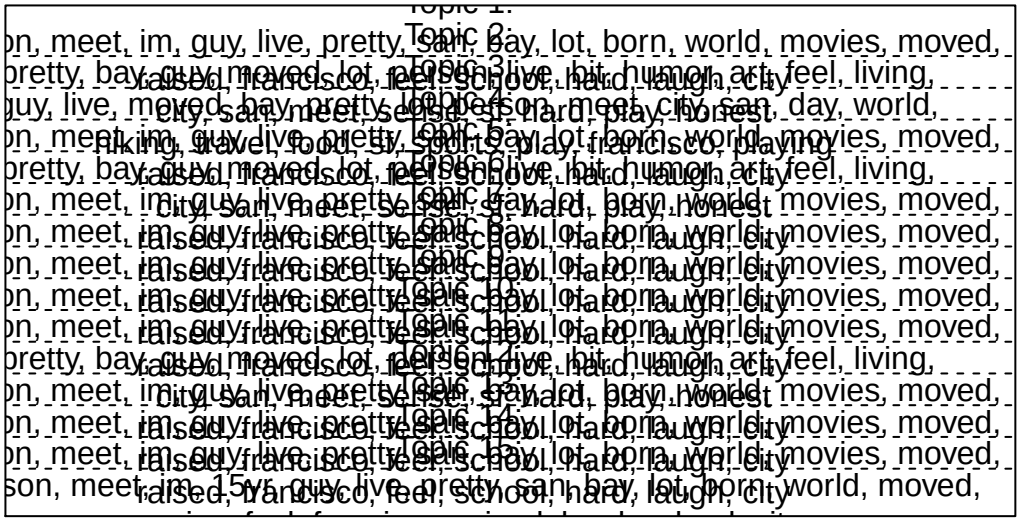
structural topic model

```
topic_num <- 15

essay_topic_model <- stm(essay_dfm, K = topic_num,
                  verbose = FALSE,
                  init.type = "Spectral")

summary(essay_topic_model)
```

```
## A topic model with 15 topics, 3 documents and a 14909 word dictionary.
## Topic 1 Top Words:
##       Highest Prob: person, meet, im, guy, live, pretty, san
##       FREX: person, meet, im, guy, live, pretty, san
##       Lift: 1.5, 35, abilities, absorb, accounting, adrian, aid
##       Score: Ã¢, 1.5, Ã£, smoke, mexican, amounts, ghetto
## Topic 2 Top Words:
##       Highest Prob: pretty, bay, guy, moved, lot, person, live
##       FREX: intelligent, idon't, sense, outdoors, personality, learned, afraid
##       Lift: 0736922466, 110, 12so, 140, 145, 150, 150,000
##       Score: intelligent, idon't, tristan, afraid, introvert, transplant, power
## Topic 3 Top Words:
##       Highest Prob: guy, live, moved, bay, pretty, lot, person
##       FREX: meow, backpacking, shares, trail, afraid, outdoors, sense
##       Lift: __update, alto, architecture, bbqs, benefit, bringing, broaden
##       Score: meow, afraid, intelligent, backpacking, skiing, cats, france
## Topic 4 Top Words:
##       Highest Prob: person, meet, im, guy, live, pretty, san
##       FREX: person, meet, im, guy, live, pretty, san
##       Lift: 105, 1984, 1month, 1rst, 2012looks, 2377388just, 240lbs
##       Score: Ã¢, 105, Ã£, smoke, mexican, amounts, ghetto
## Topic 5 Top Words:
##       Highest Prob: pretty, bay, guy, moved, lot, person, live
##       FREX: intelligent, idon't, sense, outdoors, personality, learned, afraid
##       Lift: 101, 110, 12so, 140, 145, 150, 150,000
##       Score: intelligent, idon't, tristan, afraid, introvert, transplant, power
## Topic 6 Top Words:
##       Highest Prob: person, meet, im, guy, live, pretty, san
##       FREX: person, meet, im, guy, live, pretty, san
##       Lift: 11sheeaaat, 35, abilities, absorb, accounting, adrian, aid
##       Score: Ã¢, 11sheeaaat, Ã£, smoke, mexican, amounts, ghetto
## Topic 7 Top Words:
##       Highest Prob: person, meet, im, guy, live, pretty, san
##       FREX: person, meet, im, guy, live, pretty, san
##       Lift: communistrevolutionaries, 1984, 1month, 1rst, 2012looks, 2377388just, 240lbs
##       Score: Ã¢, communistrevolutionaries, Ã£, smoke, mexican, amounts, ghetto
## Topic 8 Top Words:
##       Highest Prob: person, meet, im, guy, live, pretty, san
##       FREX: person, meet, im, guy, live, pretty, san
##       Lift: 12oz, adrian, aspects, audience, bodily, broke, broken
##       Score: Ã¢, 12oz, Ã£, smoke, mexican, amounts, ghetto
## Topic 9 Top Words:
##       Highest Prob: person, meet, im, guy, live, pretty, san
##       FREX: person, meet, im, guy, live, pretty, san
```

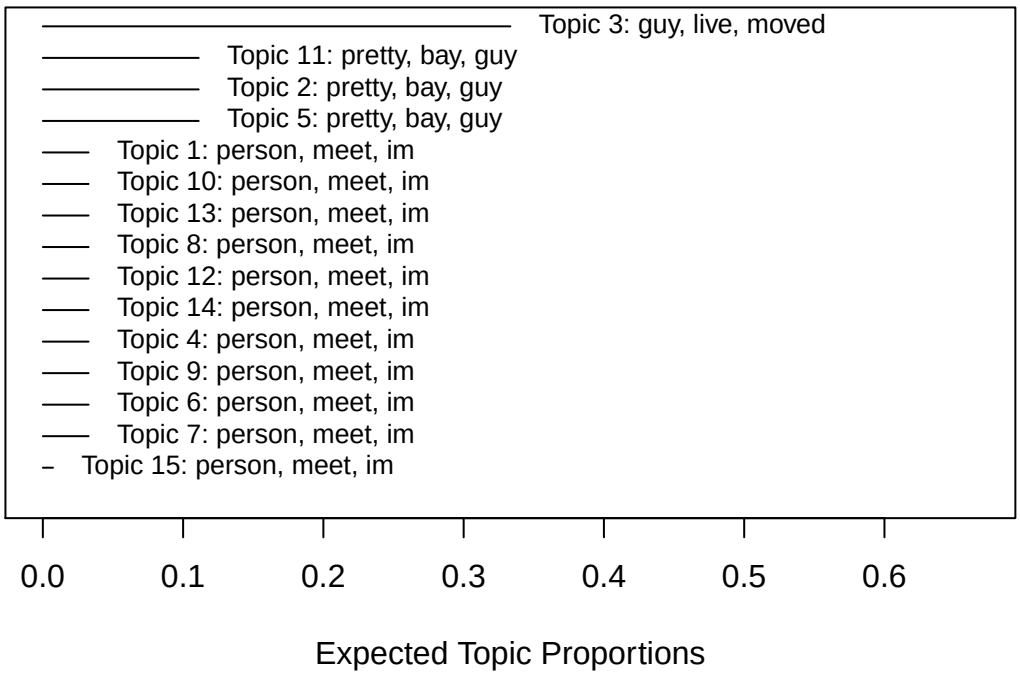
```
##       Lift: generalsituational, 1984, 1month, 1rst, 2012looks, 2377388just, 240lbs
##       Score: Ã¢, generalsituational, Ã£, smoke, mexican, amounts, ghetto
## Topic 10 Top Words:
##       Highest Prob: person, meet, im, guy, live, pretty, san
##       FREX: person, meet, im, guy, live, pretty, san
##       Lift: 13x9, 35, abilities, absorb, accounting, adrian, aid
##       Score: Ã¢, 13x9, Ã£, smoke, mexican, amounts, ghetto
## Topic 11 Top Words:
##       Highest Prob: pretty, bay, guy, moved, lot, person, live
##       FREX: intelligent, idon't, sense, outdoors, personality, learned, afraid
##       Lift: gotlost, tristan, 110, 12so, 140, 145, 150
##       Score: intelligent, idon't, tristan, afraid, introvert, transplant, power
## Topic 12 Top Words:
##       Highest Prob: person, meet, im, guy, live, pretty, san
##       FREX: person, meet, im, guy, live, pretty, san
##       Lift: 13yo, 1984, 1month, 1rst, 2012looks, 2377388just, 240lbs
##       Score: Ã¢, 13yo, Ã£, smoke, mexican, amounts, ghetto
## Topic 13 Top Words:
##       Highest Prob: person, meet, im, guy, live, pretty, san
##       FREX: person, meet, im, guy, live, pretty, san
##       Lift: 14,265mountain, 1984, 1month, 1rst, 2012looks, 2377388just, 240lbs
##       Score: Ã¢, 14,265mountain, Ã£, smoke, mexican, amounts, ghetto
## Topic 14 Top Words:
##       Highest Prob: person, meet, im, guy, live, pretty, san
##       FREX: person, meet, im, guy, live, pretty, san
##       Lift: 14am.i'm, 1984, 1month, 1rst, 2012looks, 2377388just, 240lbs
##       Score: Ã¢, 14am.i'm, Ã£, smoke, mexican, amounts, ghetto
## Topic 15 Top Words:
##       Highest Prob: person, meet, im, 15yr, guy, live, pretty
##       FREX: person, meet, im, 15yr, guy, live, pretty
##       Lift: 15yr, 1984, 1month, 1rst, 2012looks, 2377388just, 240lbs
##       Score: 15yr, Ã¢, Ã£, smoke, mexican, amounts, ghetto
```

```
plot.STM(essay_topic_model, type = "labels")
```



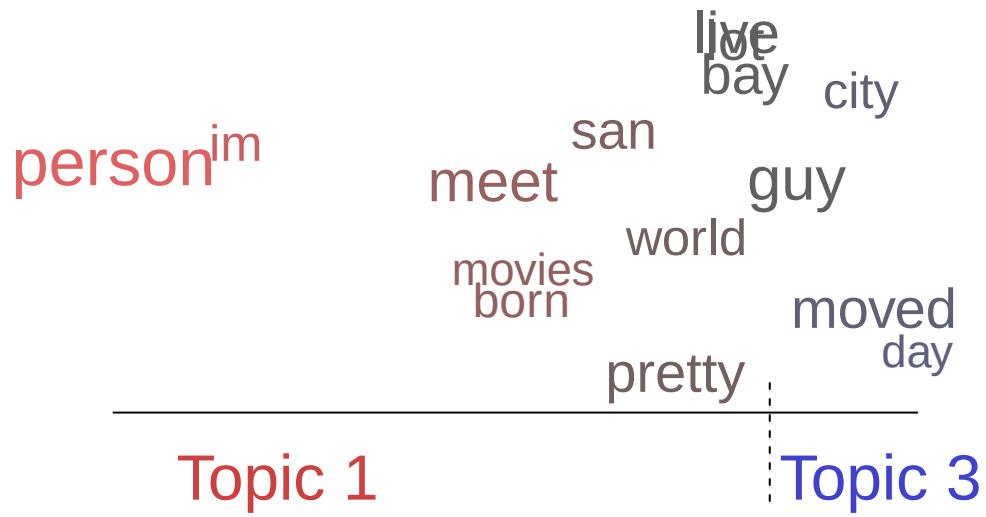
plot total topic share

```
plot(essay_topic_model, type = "summary", text.cex = 0.8)
```



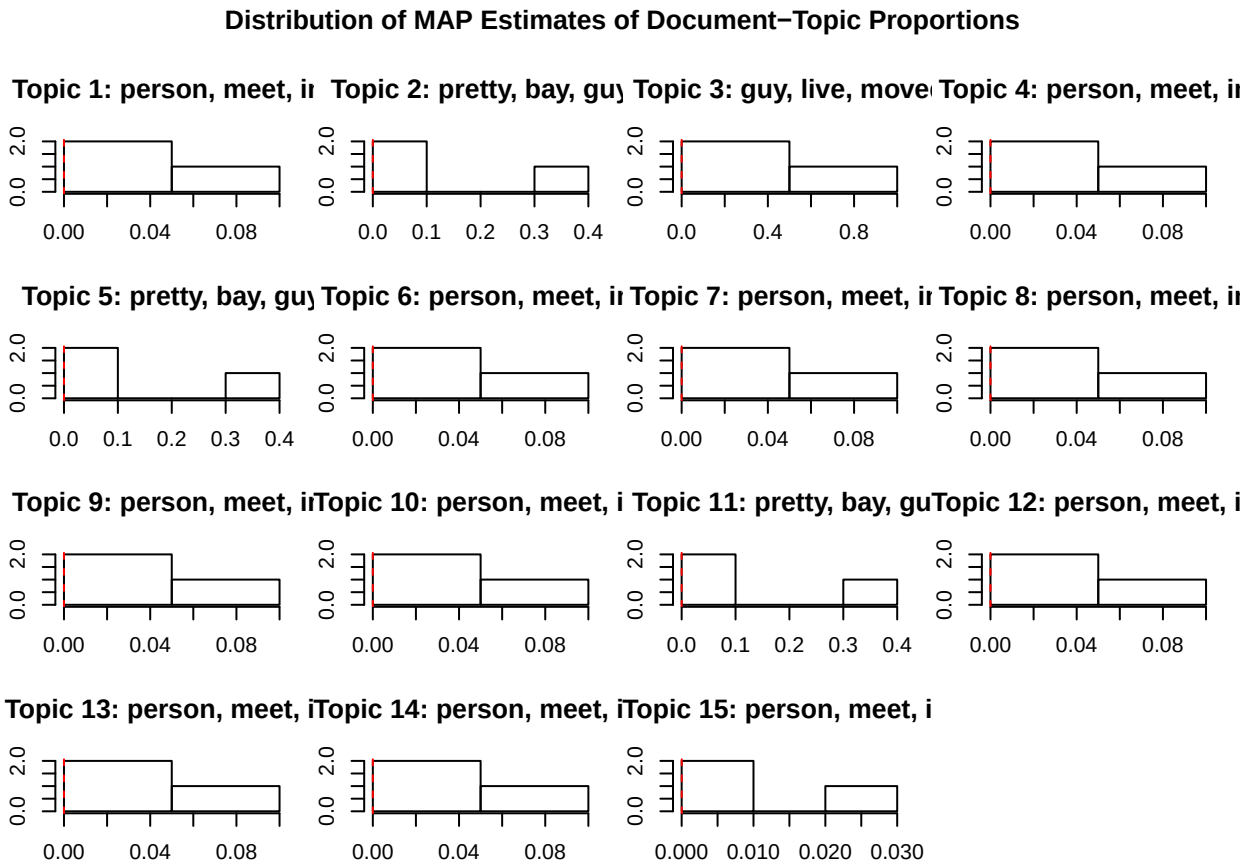
visualize topic constrast between two topics

```
plot(essay_topic_model, type = "perspectives", topics = c(1,3))
```



plot topic proportions within documents

```
plot(essay_topic_model, type = "hist")
```

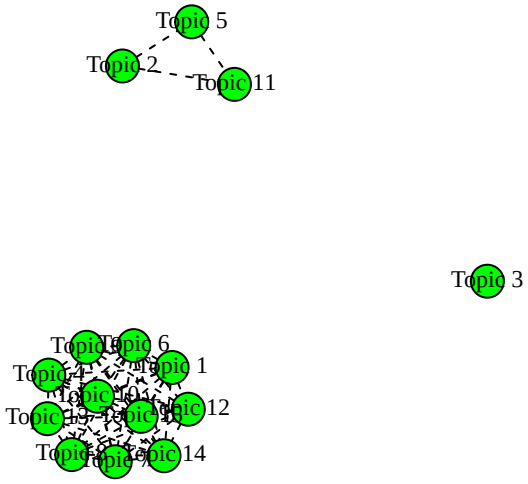


network of topics

Positive correlations between topics indicate that both topics are likely to be discussed within a document. A graphical network display shows how closely related topics are to one another (i.e., how likely they are to appear in the same document). This function requires igraph R package.

Source: https://github.com/dondealban/learning-stm

```
mod.out.corr <- topicCorr(essay_topic_model)
plot(mod.out.corr)
```



word cloud of certain topic

```
cloud(essay_topic_model, topic=2)
```

```
## Warning in wordcloud::wordcloud(words = vocab, freq = vec, max.words =
## max.words, : pretty could not be fit on page. It will not be plotted.

## Warning in wordcloud::wordcloud(words = vocab, freq = vec, max.words =
## max.words, : personality could not be fit on page. It will not be plotted.

## Warning in wordcloud::wordcloud(words = vocab, freq = vec, max.words =
## max.words, : person could not be fit on page. It will not be plotted.

## Warning in wordcloud::wordcloud(words = vocab, freq = vec, max.words =
## max.words, : meet could not be fit on page. It will not be plotted.

## Warning in wordcloud::wordcloud(words = vocab, freq = vec, max.words =
## max.words, : playing could not be fit on page. It will not be plotted.

## Warning in wordcloud::wordcloud(words = vocab, freq = vec, max.words =
## max.words, : school could not be fit on page. It will not be plotted.

## Warning in wordcloud::wordcloud(words = vocab, freq = vec, max.words =
## max.words, : day could not be fit on page. It will not be plotted.

## Warning in wordcloud::wordcloud(words = vocab, freq = vec, max.words =
## max.words, : message could not be fit on page. It will not be plotted.

## Warning in wordcloud::wordcloud(words = vocab, freq = vec, max.words =
## max.words, : laugh could not be fit on page. It will not be plotted.
```

```
## Warning in wordcloud::wordcloud(words = vocab, freq = vec, max.words =
## max.words, : outdoors could not be fit on page. It will not be plotted.

## Warning in wordcloud::wordcloud(words = vocab, freq = vec, max.words =
## max.words, : guy could not be fit on page. It will not be plotted.

## Warning in wordcloud::wordcloud(words = vocab, freq = vec, max.words =
## max.words, : world could not be fit on page. It will not be plotted.

## Warning in wordcloud::wordcloud(words = vocab, freq = vec, max.words =
## max.words, : san could not be fit on page. It will not be plotted.

## Warning in wordcloud::wordcloud(words = vocab, freq = vec, max.words =
## max.words, : spend could not be fit on page. It will not be plotted.

## Warning in wordcloud::wordcloud(words = vocab, freq = vec, max.words =
## max.words, : funny could not be fit on page. It will not be plotted.

## Warning in wordcloud::wordcloud(words = vocab, freq = vec, max.words =
## max.words, : grew could not be fit on page. It will not be plotted.

## Warning in wordcloud::wordcloud(words = vocab, freq = vec, max.words =
## max.words, : talk could not be fit on page. It will not be plotted.

## Warning in wordcloud::wordcloud(words = vocab, freq = vec, max.words =
## max.words, : college could not be fit on page. It will not be plotted.

## Warning in wordcloud::wordcloud(words = vocab, freq = vec, max.words =
## max.words, : francisco could not be fit on page. It will not be plotted.

## Warning in wordcloud::wordcloud(words = vocab, freq = vec, max.words =
## max.words, : watching could not be fit on page. It will not be plotted.

## Warning in wordcloud::wordcloud(words = vocab, freq = vec, max.words =
## max.words, : hang could not be fit on page. It will not be plotted.

## Warning in wordcloud::wordcloud(words = vocab, freq = vec, max.words =
## max.words, : moved could not be fit on page. It will not be plotted.

## Warning in wordcloud::wordcloud(words = vocab, freq = vec, max.words =
## max.words, : hiking could not be fit on page. It will not be plotted.
```

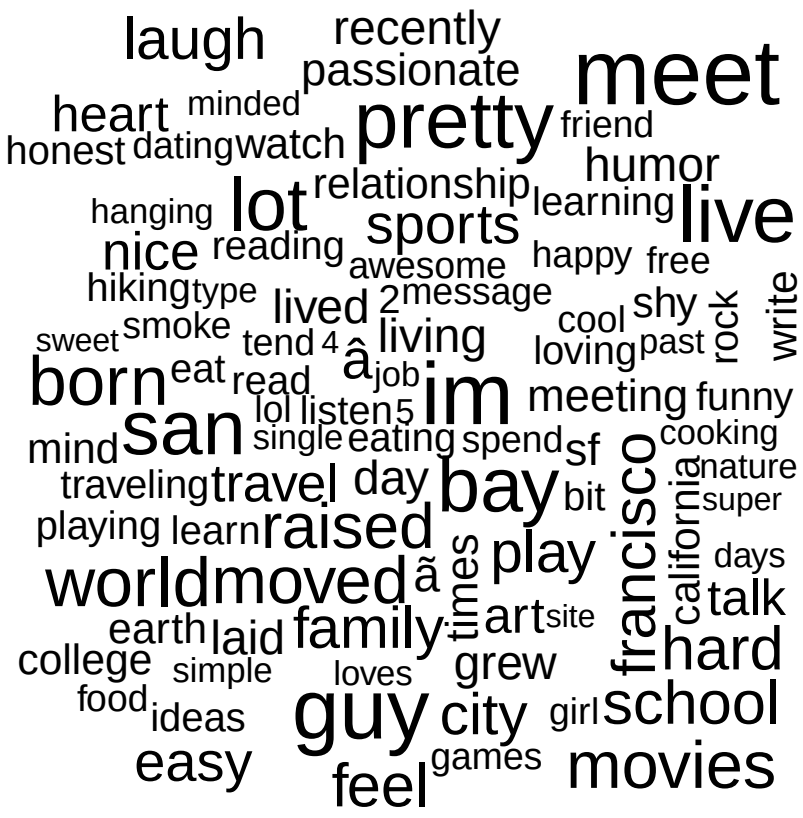

```
cloud(essay_topic_model, topic=4)
```

```
## Warning in wordcloud::wordcloud(words = vocab, freq = vec, max.words =
## max.words, : person could not be fit on page. It will not be plotted.
```

```
## Warning in wordcloud::wordcloud(words = vocab, freq = vec, max.words =
## max.words, : conversation could not be fit on page. It will not be plotted.
```

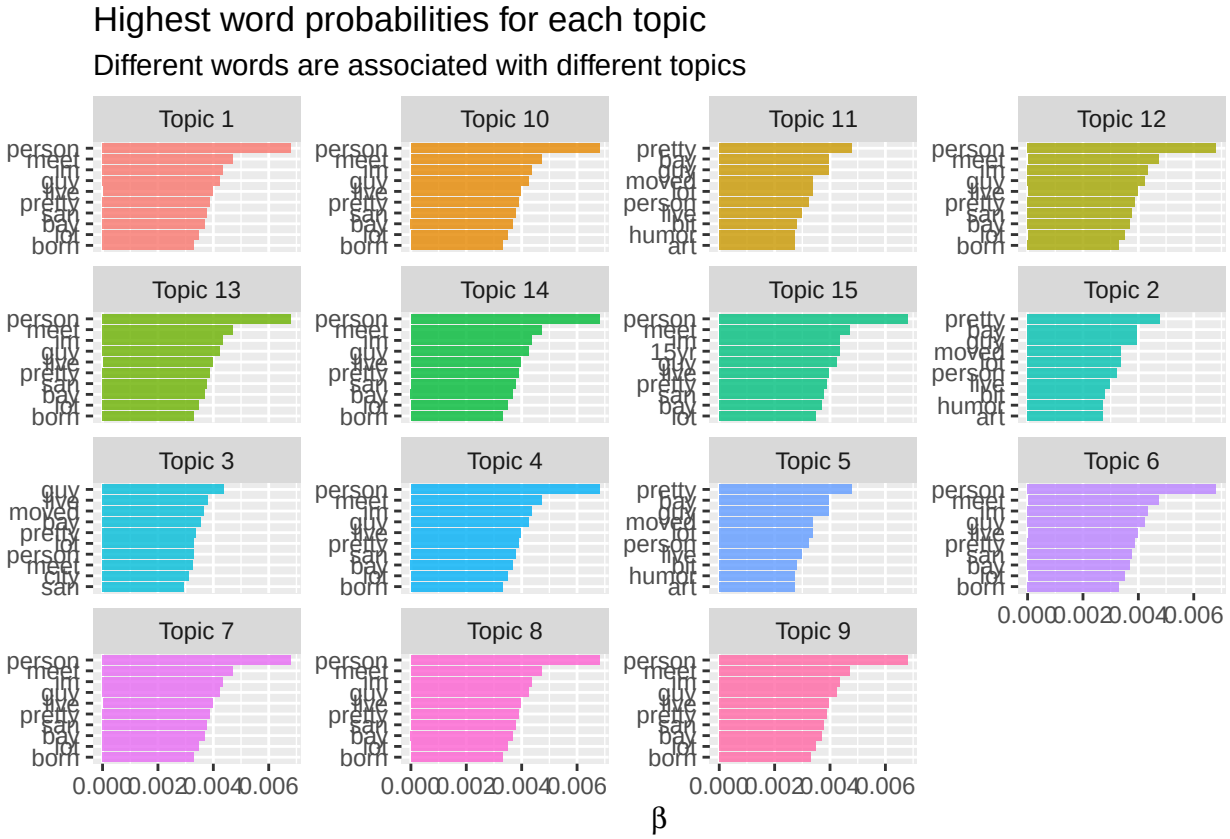
```
## Warning in wordcloud::wordcloud(words = vocab, freq = vec, max.words =
## max.words, : watching could not be fit on page. It will not be plotted.
```

```
## Warning in wordcloud::wordcloud(words = vocab, freq = vec, max.words =
## max.words, : home could not be fit on page. It will not be plotted.
```



beta prob: Distribution of word probabilities for each topic

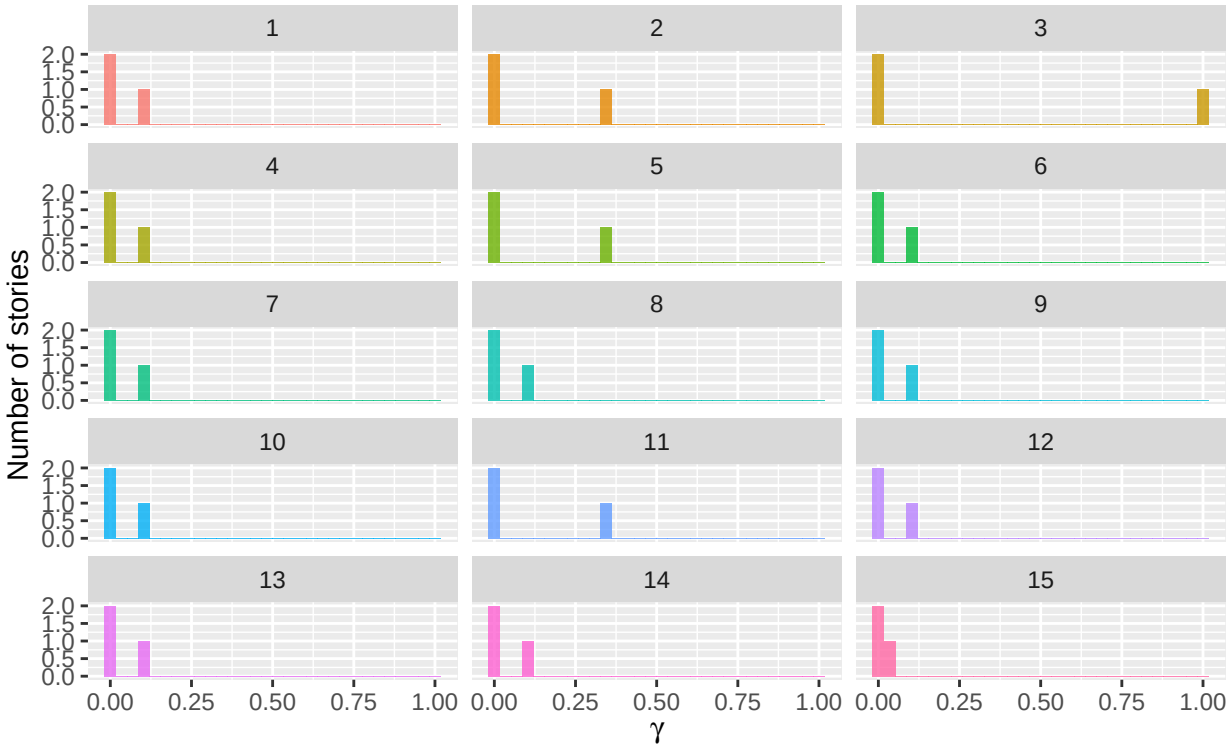
```
td_beta <- tidy(essay_topic_model)

td_beta %>%
  group_by(topic) %>%
  top_n(10, beta) %>%
  ungroup() %>%
  mutate(topic = paste0("Topic ", topic),
         term = reorder_within(term, beta, topic)) %>%
  ggplot(aes(term, beta, fill = as.factor(topic))) +
  geom_col(alpha = 0.8, show.legend = FALSE) +
  facet_wrap(~ topic, scales = "free_y") +
  coord_flip() +
  scale_x_reordered() +
  labs(x = NULL, y = expression(beta),
       title = "Highest word probabilities for each topic",
       subtitle = "Different words are associated with different topics")
```



gamma prob: Distribution of document probabilities for each topic

```
td_gamma <- tidy(essay_topic_model, matrix = "gamma",
                 document_names = rownames(essay_dfm))

ggplot(td_gamma, aes(gamma, fill = as.factor(topic))) +
  geom_histogram(alpha = 0.8, show.legend = FALSE) +
  facet_wrap(~ topic, ncol = 3) +
  labs(title = "Distribution of document probabilities for each topic",
       subtitle = "Each topic is associated with 1-2 clusters",
       y = "Number of stories", x = expression(gamma))
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



Reference

https://www.tidytextmining.com/topicmodeling.html

https://rpubs.com/cbpuschmann/un-stm

https://juliasilge.com/blog/sherlock-holmes-stm/

https://juliasilge.shinyapps.io/sherlock-holmes/#section-documents-by-topic

https://github.com/dondealban/learning-stm

https://blogs.uoregon.edu/rclub/2016/04/05/structural-topic-modeling/

Roberts, M.E., Stewart, B.M. Tingley, D. & Benoit, K. (2017) stm: Estimation of the Structural Topic Model. (https://cran.r-project.org/web/packages/stm/index.html)

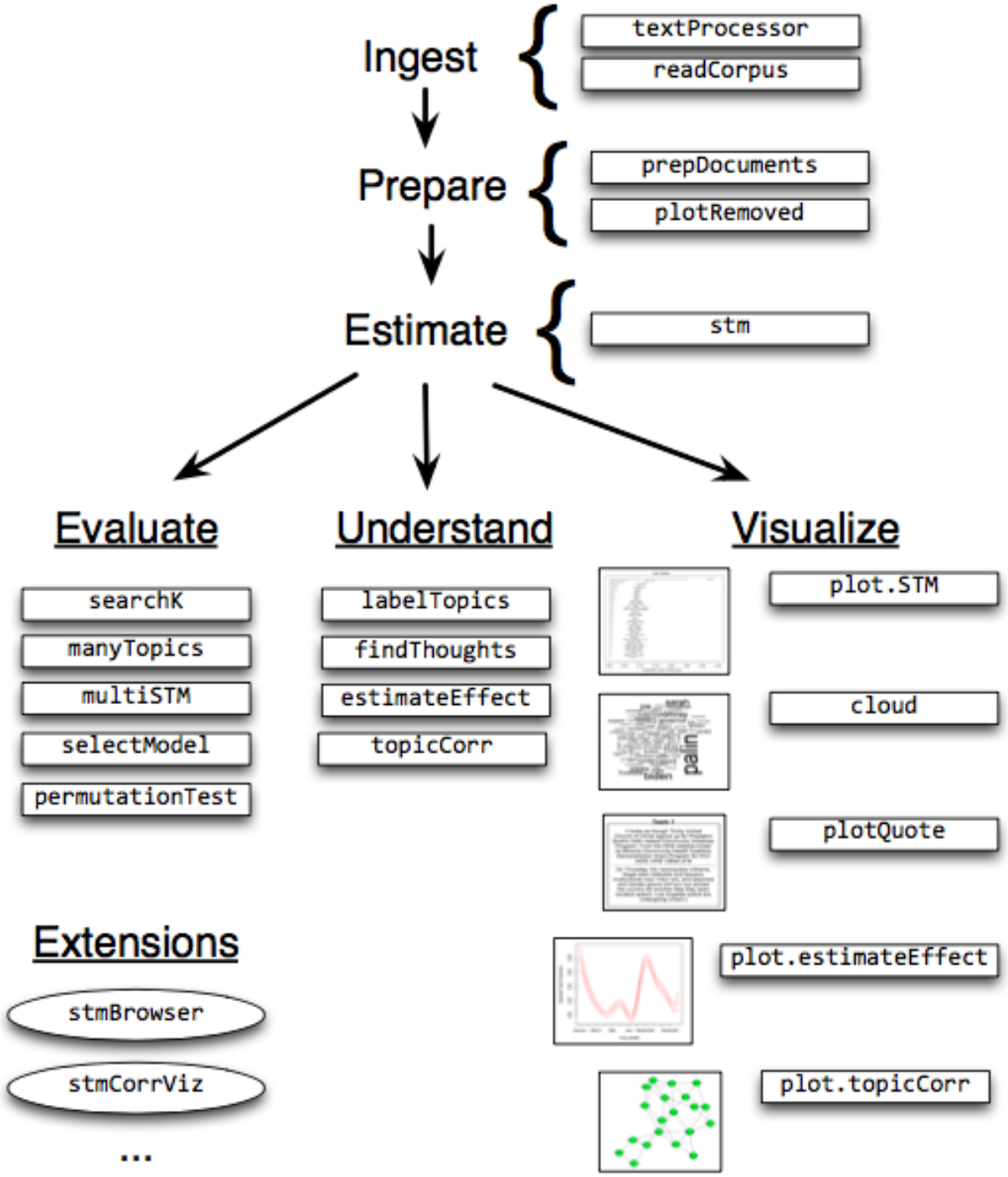


Figure 2: Heuristic description of **stm** package features.

Figure 1: stm_diagram