# Movie Plots Topic Modeling Analysis

Chang Lu

## Table of contents

```
# Load data
movie_plots <- read.csv("movie_plots_with_genres.csv")
names(movie_plots)
```

```
[1] "row"        "Movie.Name" "Genre"      "Plot"
```

```
# Data preprocessing
movie_plots_clean <- movie_plots %>%
  rename(Movie = Movie.Name) %>%  # Rename column for simplicity
  unnest_tokens(word, Plot) %>%
  anti_join(stop_words, by = "word") %>%  # Ensure correct join
  filter(!is.na(word)) %>%  # Remove any NA words
  count(Movie, word, sort = TRUE) %>%
  cast_dtm(Movie, word, n)
```

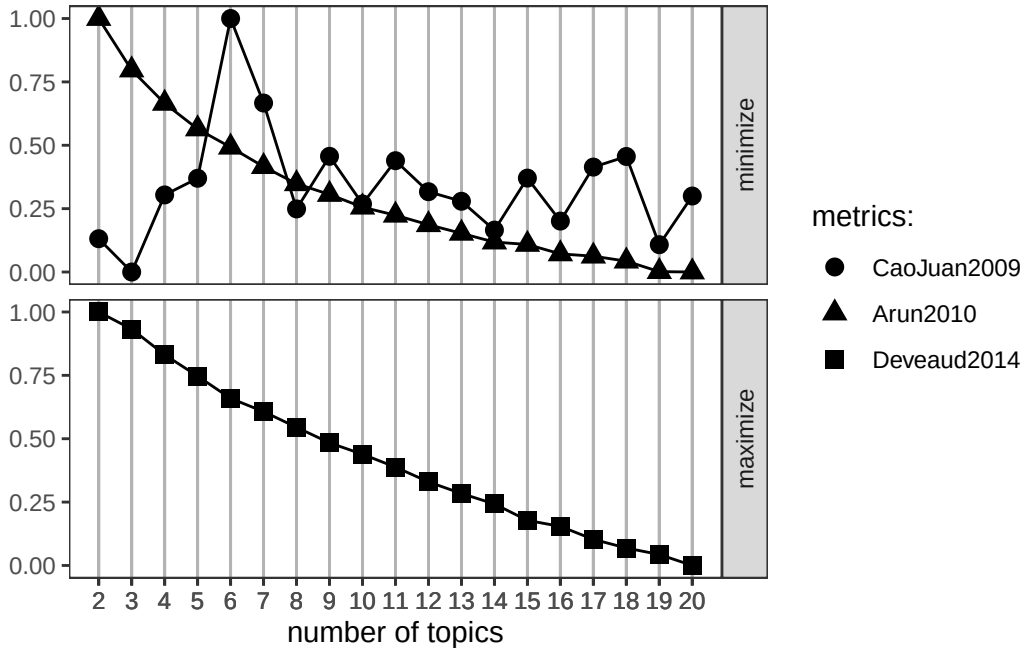
```
# Determine the optimal number of topics using ldatuning
result <- FindTopicsNumber(
  movie_plots_clean,
  topics = seq(2, 20, by = 1),
  metrics = c("CaoJuan2009", "Arun2010", "Deveaud2014"),
  method = "Gibbs",
  control = list(seed = 1234),
  verbose = TRUE
)
```

```
fit models... done.
calculate metrics:
```

```
  CaoJuan2009... done.
  Arun2010... done.
  Deveaud2014... done.
```

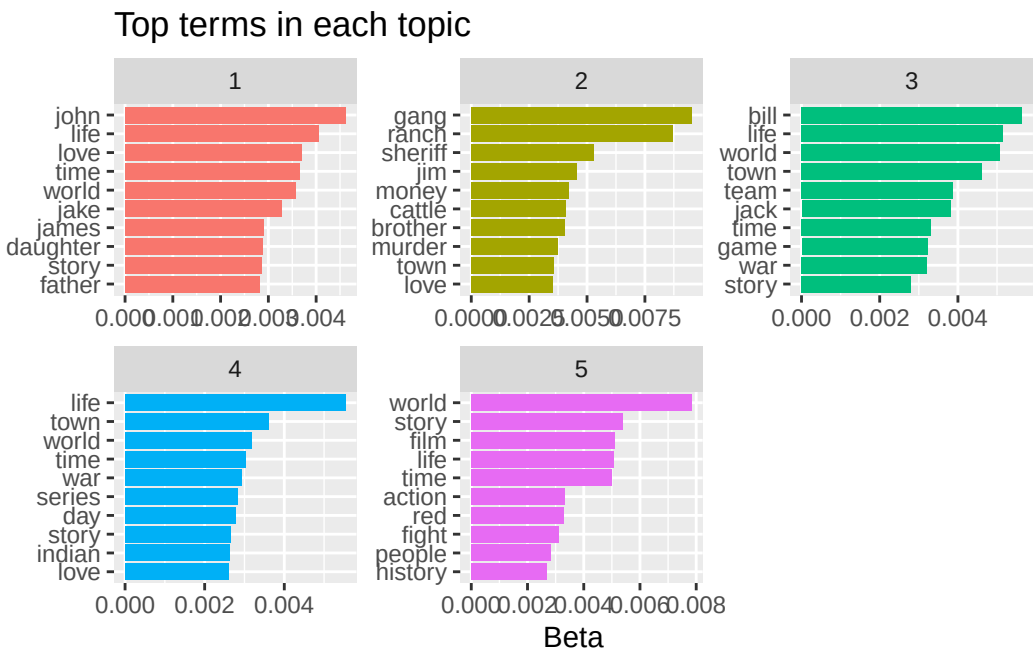
```r
# Plot the result to choose k
FindTopicsNumber_plot(result)
```

```
Warning: The `<scale>` argument of `guides()` cannot be `FALSE`. Use "none" instead as
of ggplot2 3.3.4.
i The deprecated feature was likely used in the ldatuning package.
  Please report the issue at <https://github.com/nikita-moor/ldatuning/issues>.
```




```r
# Fit the LDA model
optimal_k <- 5  # Replace with chosen k based on the scree plot
lda_model <- LDA(movie_plots_clean, k = optimal_k, control = list(seed = 1234))

# Extract topics
topics <- tidy(lda_model, matrix = "beta")

top_terms <- topics %>%
  group_by(topic) %>%
  slice_max(beta, n = 10) %>%
```

```
  ungroup() %>%
  arrange(topic, -beta)

# Visualize topics
top_terms %>%
  mutate(term = reorder_within(term, beta, topic)) %>%
  ggplot(aes(term, beta, fill = factor(topic))) +
  geom_col(show.legend = FALSE) +
  facet_wrap(~ topic, scales = "free") +
  coord_flip() +
  scale_x_reordered() +
  labs(title = "Top terms in each topic", x = NULL, y = "Beta")
```

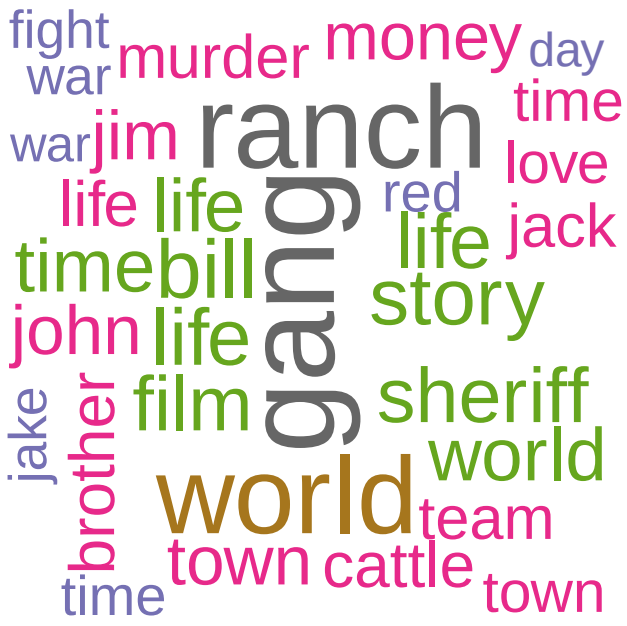



```
# Create an artsy word cloud
wordcloud(top_terms$term, top_terms$beta, max.words = 100, random.order = FALSE, colors = bre
```

```
Warning in wordcloud(top_terms$term, top_terms$beta, max.words = 100,
random.order = FALSE, : world could not be fit on page. It will not be plotted.
```

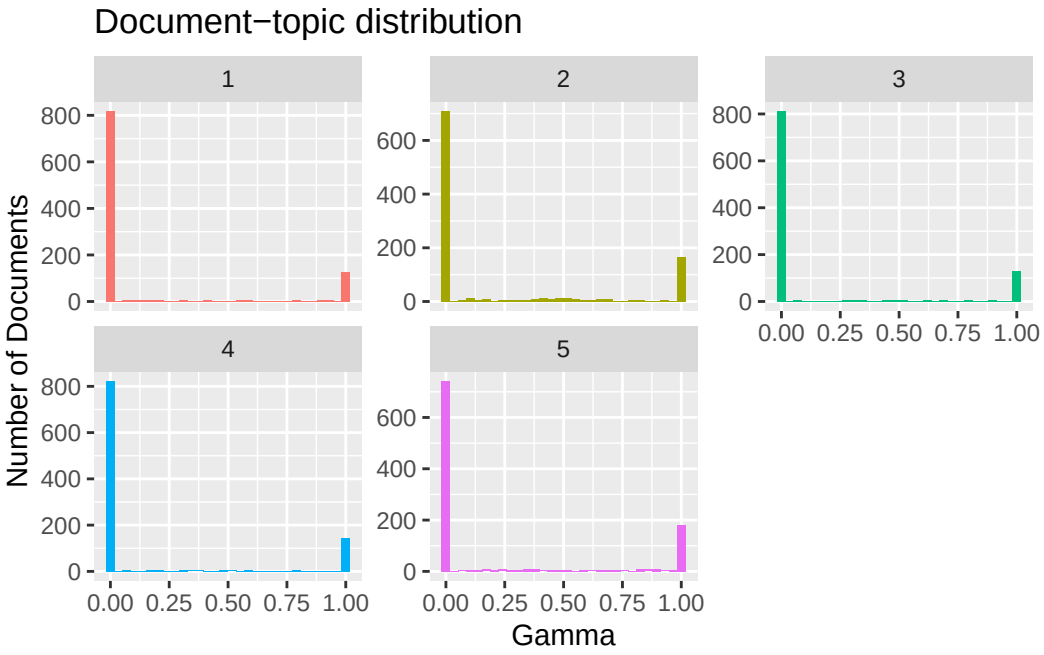
```
Warning in wordcloud(top_terms$term, top_terms$beta, max.words = 100,
random.order = FALSE, : town could not be fit on page. It will not be plotted.
```

```
Warning in wordcloud(top_terms$term, top_terms$beta, max.words = 100,
random.order = FALSE, : love could not be fit on page. It will not be plotted.

Warning in wordcloud(top_terms$term, top_terms$beta, max.words = 100,
random.order = FALSE, : action could not be fit on page. It will not be
plotted.

Warning in wordcloud(top_terms$term, top_terms$beta, max.words = 100,
random.order = FALSE, : game could not be fit on page. It will not be plotted.

Warning in wordcloud(top_terms$term, top_terms$beta, max.words = 100,
random.order = FALSE, : world could not be fit on page. It will not be plotted.

Warning in wordcloud(top_terms$term, top_terms$beta, max.words = 100,
random.order = FALSE, : time could not be fit on page. It will not be plotted.

Warning in wordcloud(top_terms$term, top_terms$beta, max.words = 100,
random.order = FALSE, : james could not be fit on page. It will not be plotted.

Warning in wordcloud(top_terms$term, top_terms$beta, max.words = 100,
random.order = FALSE, : daughter could not be fit on page. It will not be
plotted.

Warning in wordcloud(top_terms$term, top_terms$beta, max.words = 100,
random.order = FALSE, : story could not be fit on page. It will not be plotted.

Warning in wordcloud(top_terms$term, top_terms$beta, max.words = 100,
random.order = FALSE, : people could not be fit on page. It will not be
plotted.

Warning in wordcloud(top_terms$term, top_terms$beta, max.words = 100,
random.order = FALSE, : series could not be fit on page. It will not be
plotted.

Warning in wordcloud(top_terms$term, top_terms$beta, max.words = 100,
random.order = FALSE, : father could not be fit on page. It will not be
plotted.

Warning in wordcloud(top_terms$term, top_terms$beta, max.words = 100,
random.order = FALSE, : story could not be fit on page. It will not be plotted.
```




```
Warning in wordcloud(top_terms$term, top_terms$beta, max.words = 100,
random.order = FALSE, : history could not be fit on page. It will not be
plotted.
Warning in wordcloud(top_terms$term, top_terms$beta, max.words = 100,
random.order = FALSE, : story could not be fit on page. It will not be plotted.

Warning in wordcloud(top_terms$term, top_terms$beta, max.words = 100,
random.order = FALSE, : indian could not be fit on page. It will not be
plotted.

Warning in wordcloud(top_terms$term, top_terms$beta, max.words = 100,
random.order = FALSE, : love could not be fit on page. It will not be plotted.
```

```r
# Extract gamma values for each document
gamma <- tidy(lda_model, matrix = "gamma")

# Visualize document-topic distribution
gamma %>%
  ggplot(aes(gamma, fill = factor(topic))) +
  geom_histogram(show.legend = FALSE, bins = 30) +
  facet_wrap(~ topic, scales = "free_y") +
  labs(title = "Document-topic distribution", x = "Gamma", y = "Number of Documents")
```


Document−topic distribution
1
2
3
4
5
Number of Documents
Gamma
0
200
400
600
800
0.00
0.25
0.50
0.75
1.00