Topic Modeling

Haochen_Li

2024-11-09

Load the Dataset:

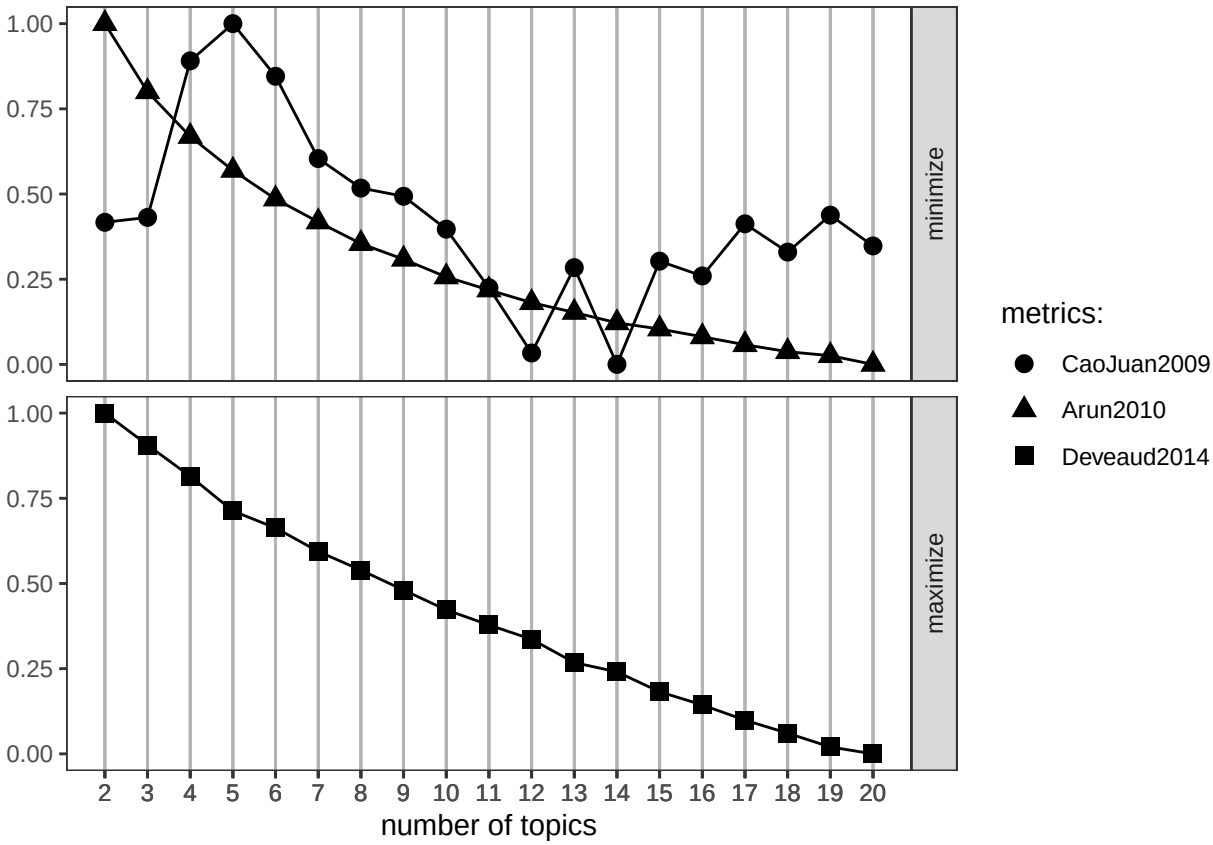
Dataset processing:

```
movie_data_clean <- movie_data %>%
  unnest_tokens(word, Plot) %>%
  anti_join(stop_words) %>%
  filter(!word %in% c("movie", "film")) # Add more domain-specific stopwords if needed
```

```
## Joining with `by = join_by(word)`
```

```
# Create a DTM
dtm <- movie_data_clean %>%
  count(row, word) %>%
  cast_dtm(row, word, n)

# Testing a range of topic numbers from 2 to 20
lda_results <- FindTopicsNumber(
  dtm,
  topics = seq(2, 20, by = 1),
  metrics = c("CaoJuan2009", "Arun2010", "Deveaud2014"),
  method = "Gibbs",
  control = list(seed = 1234)
)

# Plot the results to find the optimal number of topics
FindTopicsNumber_plot(lda_results)
```

```
## 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>.
## This warning is displayed once every 8 hours.
## Call `lifecycle::last_lifecycle_warnings()` to see where this warning was
## generated.
```



Top Plot (Minimize Metrics - CaoJuan2009 and Arun2010):

As the number of topics increases, the values for CaoJuan2009 (circles) and Arun2010 (triangles) fluctuate but generally decrease, suggesting that fewer topics (e.g., around 6-8) might offer good coherence.

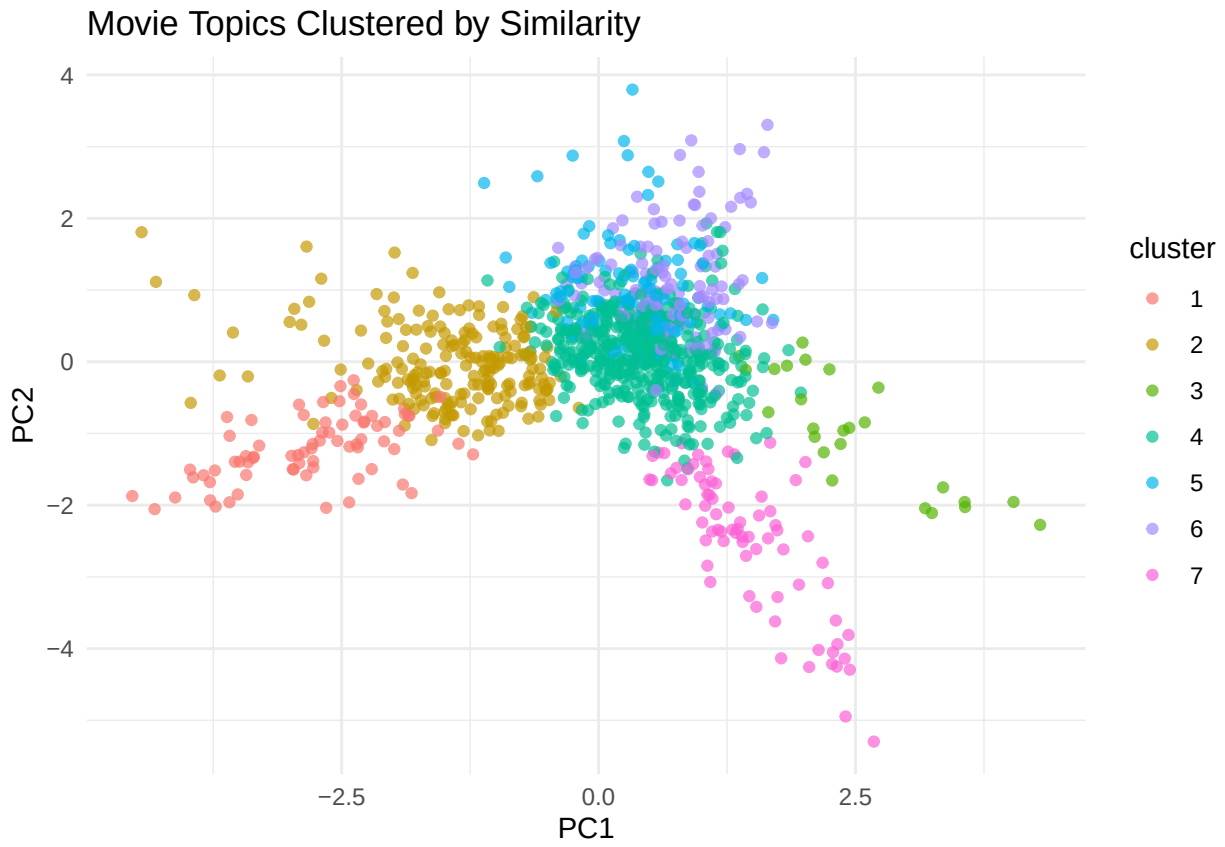
For Arun2010, the values stabilize around 6-10 topics before starting to rise again, indicating an optimal range around this topic count.

Bottom Plot (Maximize Metric - Deveaud2014):

The Deveaud2014 metric (squares) shows a gradual decline as the number of topics increases, suggesting that a lower number of topics might be optimal.

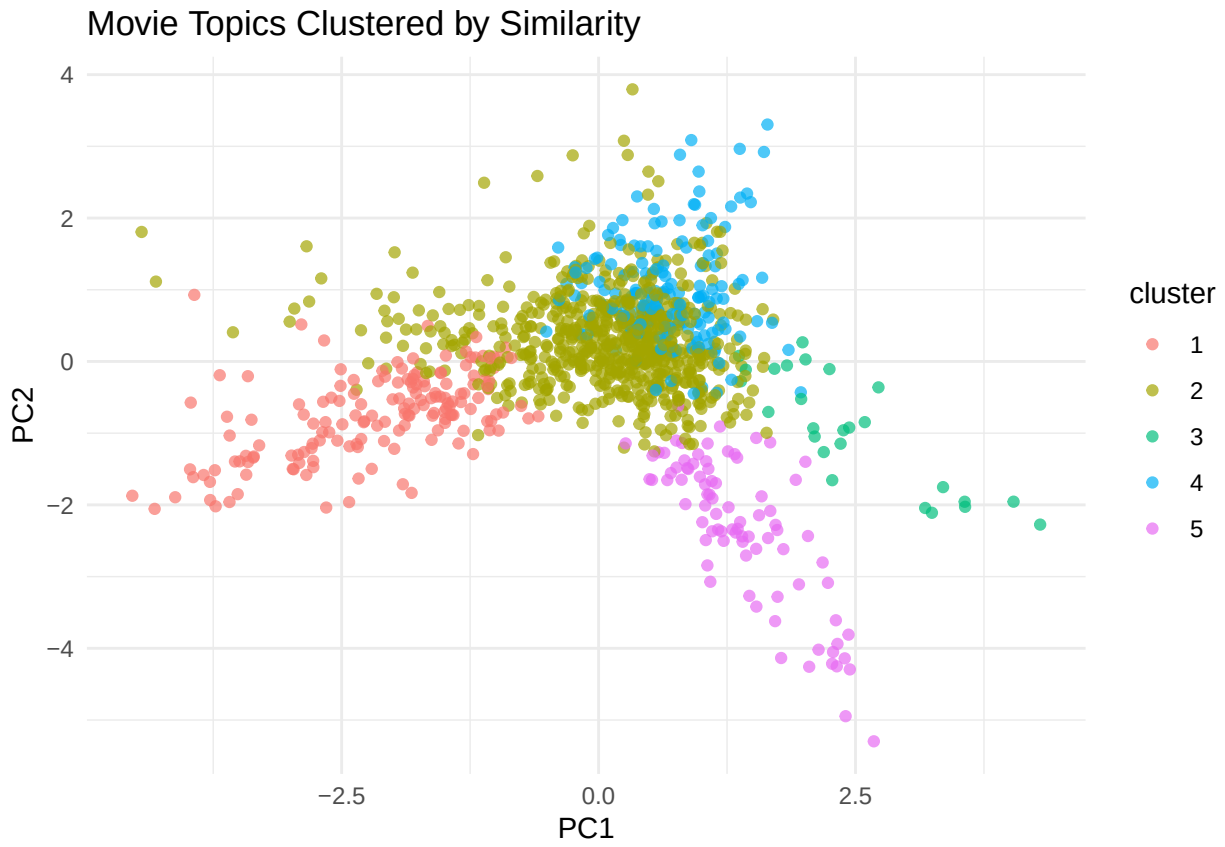
Higher values around 3-6 topics show better coherence, so the ideal number of topics could likely fall within this range.

```
optimal_k <- 7
lda_model <- LDA(dtm, k = optimal_k, method = "Gibbs", control = list(seed = 1234))

# Extract topic distributions for each document
topic_distributions <- posterior(lda_model)$topics

k_means <- kmeans(topic_distributions, centers = 7)

# Add cluster assignments to your dataset
movie_data$cluster <- as.factor(k_means$cluster)

pca_result <- prcomp(topic_distributions, scale = TRUE)
pca_data <- as.data.frame(pca_result$x[, 1:2]) # Use first two principal components
pca_data$cluster <- movie_data$cluster

# Plot clusters
ggplot(pca_data, aes(x = PC1, y = PC2, color = cluster)) +
  geom_point(alpha = 0.7) +
  labs(title = "Movie Topics Clustered by Similarity", x = "PC1", y = "PC2") +
  theme_minimal()
```




```
movie_data_tfidf <- movie_data_clean %>%
  count(row, word) %>%
  bind_tf_idf(word, row, n) %>%
  cast_dtm(row, word, tf_idf)
```

```
#######

optimal_k <- 7
lda_model <- LDA(dtm, k = optimal_k, method = "Gibbs", control = list(seed = 1234))

# Extract topic distributions for each document
topic_distributions <- posterior(lda_model)$topics

k_means <- kmeans(topic_distributions, centers = 5)

# Add cluster assignments to your dataset
movie_data$cluster <- as.factor(k_means$cluster)

pca_result <- prcomp(topic_distributions, scale = TRUE)
pca_data <- as.data.frame(pca_result$x[, 1:2]) # Use first two principal components
pca_data$cluster <- movie_data$cluster

# Plot clusters
ggplot(pca_data, aes(x = PC1, y = PC2, color = cluster)) +
  geom_point(alpha = 0.7) +
  labs(title = "Movie Topics Clustered by Similarity", x = "PC1", y = "PC2") +
  theme_minimal()
```



This plot shows movie topics clustered by similarity, reduced to two main components (PC1 and PC2). Each color represents a cluster of movies with similar themes:

Cluster 1 (Red): Distinct theme, on the left side.

Cluster 2 (Green): Central and dense, likely common themes.

Cluster 3 (Blue): Top-right, unique themes.

Clusters 4 (Teal) & 5 (Purple): On the plot's edges, indicating niche themes.

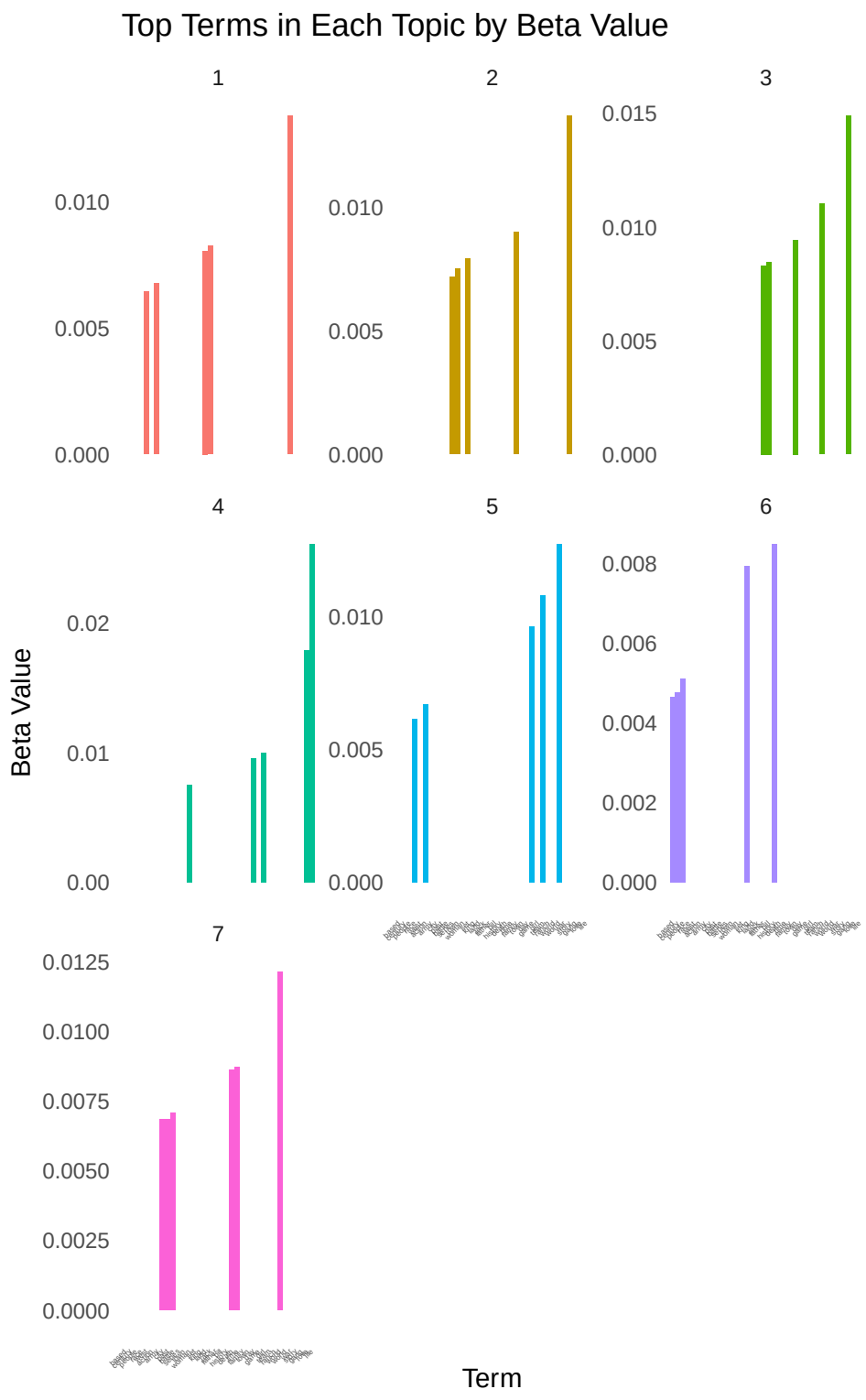
Movies close together share similar topics, while those further apart differ more in theme. This helps visualize the thematic grouping among movies.

The plot with 5 clusters is likely the better option here. It has a more

compact and balanced distribution with better separation between clusters,

making it easier to interpret the overall clustering structure.

```
lda_terms <- tidy(lda_model, matrix = "beta")

# Select the top 5 terms per topic
top_terms <- lda_terms %>%
  group_by(topic) %>%
  top_n(5, beta) %>%
  ungroup() %>%
  arrange(topic, -beta)


ggplot(top_terms, aes(x = reorder_within(term, beta, topic), y = beta, fill = factor(topic))) +
  geom_col(show.legend = FALSE) +
  facet_wrap(~ topic, scales = "free_y") +
  scale_x_reordered() +
  labs(title = "Top Terms in Each Topic by Beta Value",
       x = "Term",
       y = "Beta Value") +
theme_minimal() +
theme(axis.text.x = element_text(angle = 45, hjust = 1, vjust = 0.5, size = 3),
      panel.grid.major = element_blank(),
      panel.grid.minor = element_blank())
```



Topic 1 (Red):

The most prominent terms in Topic 1 have relatively high beta values, suggesting they are strongly associated with this topic.

The terms could represent a specific theme, such as “action” or “war,” depending on the terms themselves

(though the text is hard to read in this image).

Topic 2 (Gold):

Topic 2 also has clear top terms with distinct beta values, indicating that certain terms are more representative of this topic than others.

This could represent a unique theme (e.g., “family” or “romance” if the words reflect those concepts).

Topic 3 (Green):

This topic shows a small range of top terms, suggesting that the content may be more specific or focused. Based on the actual terms, you might infer a theme like “adventure” or “exploration.”

Topic 4 (Turquoise):

Topic 4 has a high beta value for a few terms, indicating a strong association with specific words.

If these terms relate to a theme like “mystery” or “crime,” it would suggest that these are central to this topic.

Topic 5 (Blue):

The beta values here indicate a moderate association with certain terms, possibly reflecting a genre or theme like “science fiction” or “fantasy,” depending on the actual terms.

Topic 6 (Purple):

Topic 6 has fewer terms with higher beta values, meaning this topic is likely well-defined with specific key terms.

This might represent a narrow theme, such as “supernatural” or “horror,” depending on the terms.

Topic 7 (Pink):

This topic has distinctive terms with higher beta values, which could suggest a unique theme, perhaps something like “historical” or “biographical.”

General Observations

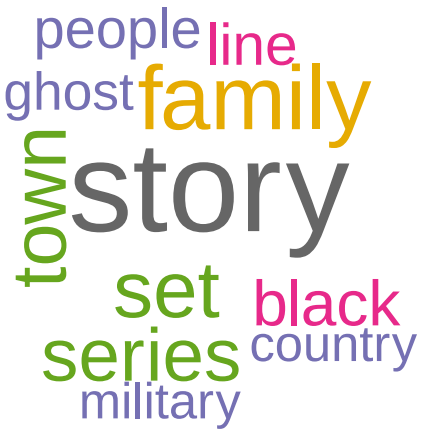
Distinctiveness: Each topic has terms with relatively high beta values, indicating that the terms selected are fairly representative of their respective topics.

Overlap: If you notice common terms across topics (e.g., “war” in multiple topics), it may indicate that the topics share some thematic overlap.

Interpret Themes Based on Terms: While the beta plot shows the importance of each term in its respective topic, you’ll need to look at the actual words (terms) to infer specific themes for each topic.

```
lda_terms <- tidy(lda_model, matrix = "beta")

# Top terms for each topic
top_terms <- lda_terms %>%
  group_by(topic) %>%
  top_n(10, beta) %>%
  ungroup() %>%
  arrange(topic, -beta)

# top_terms


######## word cloud #######
for (i in 1:optimal_k) {
  wordcloud(words = filter(top_terms, topic == i)$term,
```

```
            freq = filter(top_terms, topic == i)$beta,
            scale = c(4, 0.5),
            random.order = FALSE,
            colors = brewer.pal(8, "Dark2"))
}
```


people
line
ghost
family
town
story
set
black
series
country
military


brother
money
gang
john
town
ranch
bill
killed
sheriff
father


woman
love
home
secret
life
wife
real
girl
day
meets


james
2
time
road
race
game
world
american
team
3
action
star
wrestling


story
century
time
people
dance
history
king
based
events
journey


past
city
fight
time
world
death
earth
battle
power
save