Topic Models Use Cases

Promothesh Chatterjee*

Topic Models

Topic Models, also called Probabilistic Topic Models, are used to automatically extract information about topics from text. Say, we have a collection of varied type of texts, topic modeling algorithms will group the text into certain somewhat non-overlapping groups.

Here is simple explanation of one of the popular topic models, LDA. Latent Dirichlet Allocation (LDA) is a generative statistical model commonly used in natural language processing to automatically discover topics in a collection of documents. Here's a simple explanation:

Imagine you have a large collection of documents and you want to find out what topics these documents cover, but you don't have time to read each document individually. LDA can help with this.

Here's how it works:

1. **Topics**: In the context of LDA, a 'topic' is a collection of words that frequently appear together. For instance, in a document about baking, words like 'cake', 'bake', 'flour', 'oven' might appear together and these would form a 'baking' topic.
2. **Assumption**: LDA assumes that each document in your collection is made up of a mix of topics. For example, a cookbook might be 30% baking, 20% Italian cuisine, 50% vegetarian recipes.
3. **Working**: When you run LDA on your collection of documents, you tell it how many topics (like the baking, Italian cuisine, vegetarian recipes) you think are in your documents. It then goes through each document and assigns every word in that document to one of the topics. It does this in a smart way, so that words that appear together are likely to be assigned to the same topic.
4. **Output**: The output of LDA is a list of topics, each represented by a collection of words, and a weightage of topics for each document.
5. **Interpretation**: You can look at the words in each topic to get an idea of what the topic is about. And for each document, you can look at the weights to see which topics are most important in that document.

In summary, LDA is a tool for automatically discovering the main topics in a large collection of documents. It's a way of summarizing and understanding large amounts of text data.

Here are several business use cases using topic modeling:

1. **Customer Support Optimization**: Topic modeling can be used to automatically categorize and route customer support tickets based on their content. For example, if a customer raises an issue about "payment failure," a topic model could identify this as related to "Payment Issues" and route it to the appropriate support team.
2. **Content Recommendation**: Online platforms such as news sites or blogs can use topic models to recommend content to users. For example, if a user often reads articles on the "Artificial Intelligence" topic, the system can suggest more articles on this topic.
3. **Market Research**: Businesses can apply topic modeling to social media posts, customer reviews, or other forms of user-generated content to identify trending topics and gain insights about customer preferences and concerns.

4. **Document Management and Search**: In large organizations, documents often get created and stored without much organization. Topic modeling can help in indexing and organizing these documents for easy retrieval. For instance, a business could use topic modeling to categorize internal reports, memos, and emails, thereby improving searchability.
5. **Product Development**: By applying topic modeling to customer feedback and reviews, businesses can identify common themes that may inform product development. For example, if many customers are discussing a specific feature or problem, that could signal an opportunity to introduce a new product or improve an existing one.
6. **Sentiment Analysis**: Topic modeling can be combined with sentiment analysis to understand the sentiments associated with different topics. This can provide valuable insights for brand management and marketing strategies.
7. **Competitor Analysis**: By applying topic modeling to public communications from competitors, such as press releases, blog posts, or social media updates, businesses can identify the key topics their competitors are focusing on and adjust their own strategies accordingly.
8. **Risk Management**: In the financial sector, topic models can be used to identify risky topics from news articles or financial reports. For example, words like "bankruptcy," "scandal," or "fraud" might be associated with higher risk.

These are just a few examples. The specific use cases can vary greatly depending on the industry and the specific needs of the business. Let's look at two use cases:

Business Case Study: Uncovering Consumer Preferences Using Topic Modeling

Introduction

A leading electronics retail chain, "TechGear," had been collecting customer reviews across several product categories including laptops, smartphones, headphones, smartwatches, and cameras. With the aim of gaining deeper insights from this valuable data, the company decided to use topic modeling techniques to understand the key themes that were frequently discussed in these reviews.

Approach

TechGear's data science team set up an R script to analyze the reviews. The reviews were preprocessed, which involved creating a corpus, tokenizing the text, converting all text to lowercase, and removing stopwords. The processed data was then transformed into a Document-Feature Matrix (DFM).

The team used the `seededlda` package to conduct a Latent Dirichlet Allocation (LDA), a popular method for topic modeling. They decided to extract five topics from the reviews. These topics represent clusters of words that frequently appear together in the reviews. Once the model was trained, they predicted the most likely topic for each review.

```
# Load necessary libraries
library(tibble)
library(quanteda)

# Define product categories
```

```r
product_categories <- c("laptop", "smartphone", "headphones", "smartwatch", "camera")

# Define some example reviews for each product category
laptop_reviews <- c("The battery life is great", "It's super fast",
                    "The screen is amazing")
smartphone_reviews <- c("The camera is superb", "Love the sleek design",
                        "It's very user friendly")
headphones_reviews <- c("The sound quality is excellent", "They are very comfortable",
                        "Too much bass")
smartwatch_reviews <- c("Tracks my workouts accurately", "Love the design",
                        "The battery life is excellent")
camera_reviews <- c("Takes high quality pictures", "It's very easy to use",
                    "Love the zoom feature")

# Create a tibble (similar to a data frame) with the reviews and product categories
reviews <- tibble(
  review = c(laptop_reviews, smartphone_reviews, headphones_reviews,
             smartwatch_reviews, camera_reviews),
  product_category = rep(product_categories, each = 3)
)

# Preprocessing steps

corpus <- corpus(reviews, text_field = "review") # create a corpus from the review text
tokens <- tokens(corpus, remove_punct = TRUE, remove_numbers = TRUE, remove_symbols = TRUE)
tokens <- tokens_tolower(tokens) # convert all text to lowercase
tokens <- tokens_remove(tokens, stopwords("en")) # remove English stopwords

# Create a Document-Feature Matrix
dfm <- dfm(tokens)

library(seededlda)
# Train a topic model with 5 topics
lda <- seededlda::textmodel_lda(dfm, k = 5)

# Display the top words in each topic
knitr::kable(seededlda::terms(lda))
```

topic1	topic2	topic3	topic4	topic5
design	love	quality	super	battery
much	friendly	sleek	fast	life
bass	tracks	user	amazing	excellent
workouts	high	sound	love	great
accurately	battery	comfortable	design	screen
easy	life	takes	feature	camera
battery	great	zoom	battery	superb
life	super	battery	life	pictures
great	fast	life	great	use
super	screen	great	screen	super

```
# You can also predict the topics of documents using topics()
dat <- quanteda::docvars(lda$data)
dat$topic <- seededlda::topics(lda)
knitr::kable(head(dat, 10))
```

product_category	topic
laptop	topic5
laptop	topic4
laptop	topic4
smartphone	topic5
smartphone	topic4
smartphone	topic2
headphones	topic3
headphones	topic3
headphones	topic1
smartwatch	topic1

Findings

To interpret the findings, you will look at the top terms in each topic that the LDA model has discovered. These terms should give you an idea of what each topic is about. For example, if the top terms for a topic are "screen", "battery", "camera", it might be about smartphone reviews. If another topic has terms like "sound", "volume", "bass", it might be about speaker or headphone reviews.

Next, look at how the documents (reviews) are assigned to each topic. This gives you an idea of what topics are most prevalent in your dataset. For example, if many reviews are assigned to the smartphone topic, it means a lot of your customers are talking about smartphones.

This analysis can help you understand what products or product features your customers are discussing most frequently. This can inform your product development, marketing, and customer service efforts. You can also conduct further analysis to see if certain topics are associated with positive or negative sentiment, which can give you more detailed insights into customer preferences.

Conclusion and Next Steps

Through this exercise, TechGear was able to gain valuable insights into customer preferences and pain points across their product range. These insights can guide future product development, improve marketing messaging, and even inform inventory decisions.

Competitor Analysis Using Topic Modeling

In the highly competitive consumer electronics market, understanding customer preferences and feedback is crucial for business success. Competitor1 and Competitor2 are two companies producing Smartwatch1 and Smartwatch2. They have been collecting customer reviews for their products to gain insights into customer preferences and identify potential areas of improvement.

The main objective was to analyze customer reviews and identify dominant themes or topics that customers frequently mention. We also wanted to compare the dominant themes between the two competitors to identify their strengths and weaknesses.

Methodology

We used a synthetic dataset consisting of 500 customer reviews for each competitor. The reviews covered a range of topics including battery life, display quality, performance, price, and customer service. We used a topic modeling approach (specifically Latent Dirichlet Allocation, LDA) to identify the most prevalent topics in the reviews.

```
library(readr)

# Create synthetic reviews for two hypothetical competitors
set.seed(12345)
product_types <- c("Smartwatch1", "Smartwatch2")
competitors <- c("Competitor1", "Competitor2")
themes <- list(
    "battery life" = c("Great battery life!", "The battery drains too quickly.",
                       "Battery lasts all day.", "Battery life is not as advertised."),
    "display quality" = c("The display is vibrant and sharp.", "Display is mediocre.",
                          "Love the OLED display!", "Screen resolution is subpar."),
    "performance" = c("Super fast and responsive!", "Performance is laggy.",
                      "Handles all apps and games smoothly.", "Tends to freeze and crash."),
    "price" = c("Excellent value for the price.", "Overpriced for what it offers.",
                "Affordable and well worth it.", "Too expensive."),
    "customer service" = c("Customer service was helpful and prompt.",
                           "Had a bad experience with customer service.",
                           "Customer support is top notch.",
                           "Customer service could be improved.")
)

# Generate synthetic dataset
```

```r
reviews <- data.frame(
    product = sample(product_types, 500, replace = TRUE),
    competitor = sample(competitors, 500, replace = TRUE),
    review = replicate(500, paste(sample(unlist(themes), 5), collapse = " "))
)

# Print the first few rows of the dataset
#head(reviews)

# Preprocessing steps
corpus <- corpus(reviews$review) # create a corpus from the review text
tokens <- tokens(corpus, remove_punct = TRUE, remove_numbers = TRUE, remove_symbols = TRUE)
tokens <- tokens_tolower(tokens) # convert all text to lowercase
tokens <- tokens_remove(tokens, stopwords("en")) # remove English stopwords

# Create a Document-Feature Matrix
dfm <- dfm(tokens)

# Train a topic model with 5 topics
model <- seededlda::textmodel_lda(dfm, k = 5)

# Extract the top terms for each topic
top_terms <- seededlda::terms(model)

# Print the top terms for each topic
print(top_terms)
```

```
##       topic1       topic2
##  [1,] "customer"   "apps"
##  [2,] "service"    "games"
##  [3,] "helpful"    "handles"
##  [4,] "bad"        "smoothly"
##  [5,] "experience" "screen"
##  [6,] "prompt"     "resolution"
##  [7,] "support"    "subpar"
##  [8,] "top"        "battery"
##  [9,] "notch"      "drains"
## [10,] "improved"   "service"
##       topic3      topic4
##  [1,] "display"   "freeze"
##  [2,] "sharp"     "crash"
##  [3,] "vibrant"   "tends"
##  [4,] "love"      "battery"
```

```
##  [5,] "oled"      "well"
##  [6,] "mediocre"  "worth"
##  [7,] "value"     "affordable"
##  [8,] "price"     "life"
##  [9,] "excellent" "excellent"
## [10,] "laggy"     "expensive"
##       topic5
##  [1,] "battery"
##  [2,] "life"
##  [3,] "fast"
##  [4,] "responsive"
##  [5,] "super"
##  [6,] "lasts"
##  [7,] "great"
##  [8,] "day"
##  [9,] "advertised"
## [10,] "offers"
```

```r
# Assign each document to a topic
doc_topics <- model$theta
doc_max_topics <- apply(doc_topics, 1, which.max)

# Add the dominant topic to each review in the dataset
reviews$dominant_topic <- doc_max_topics

# Now, you can analyze the dominant topics for each competitor
table(reviews$competitor, reviews$dominant_topic)
```

```
##
##                1  2  3  4  5
##   Competitor1 69 55 37 52 52
##   Competitor2 72 52 43 35 33
```

```r
# Now, you can analyze the dominant topics for each competitor
topic_distribution <- table(reviews$competitor, reviews$dominant_topic)

# Provide meaningful names to the rows and columns
rownames(topic_distribution) <- paste("Competitor", rownames(topic_distribution))
colnames(topic_distribution) <- paste("Topic", colnames(topic_distribution))

# Add margins (totals for each row and column)
topic_distribution <- addmargins(topic_distribution)

# Print the labeled table
print(topic_distribution)
```

```
##
##                           Topic 1
##   Competitor Competitor1      69
##   Competitor Competitor2      72
##   Sum                        141
##
##                           Topic 2
##   Competitor Competitor1      55
##   Competitor Competitor2      52
##   Sum                        107
##
##                           Topic 3
##   Competitor Competitor1      37
##   Competitor Competitor2      43
##   Sum                         80
##
##                           Topic 4
##   Competitor Competitor1      52
##   Competitor Competitor2      35
##   Sum                         87
##
##                           Topic 5 Sum
##   Competitor Competitor1      52 265
##   Competitor Competitor2      33 235
##   Sum                         85 500
```

Findings

The LDA model identified five topics that were prevalent in the reviews. Each topic was characterized by a set of top terms that frequently appeared together in the reviews.

The distribution of dominant topics varied between the two competitors. For example, Competitor1 had a higher proportion of topic 4 and topic 5, while Competitor2 had more a higher proportion of topic 1 and topic 3.

These insights could help both competitors to focus their efforts on improving specific product aspects and enhance their customer service, thus improving overall customer satisfaction and gaining a competitive edge in the market.

In conclusion, topic modeling is a powerful tool for analyzing customer reviews and understanding customer preferences. By identifying dominant themes in customer feedback, businesses can gain valuable insights and make data-driven decisions to improve their products and services.