```
# Install and load required packages
install.packages(c("rtweet", "tidytext", "dplyr", "ggplot2", "stringr", "tidyr"))
library(rtweet)
library(tidytext)
library(dplyr)
library(ggplot2)
library(stringr)
library(tidyr)

# Fetch recent tweets containing a specific keyword
tweets <- search_tweets("climate change", n = 500, lang = "en", include_rts = FALSE)

# View the structure
head(tweets$text)

# Clean and tokenize the text
tweet_words <- tweets %>%
  select(status_id, text) %>%
  unnest_tokens(word, text)

# Remove stop words
data("stop_words")
cleaned_tweets <- tweet_words %>%
  anti_join(stop_words, by = "word")

# Perform sentiment analysis using Bing lexicon
```

```
bing_sentiments <- cleaned_tweets %>%
  inner_join(get_sentiments("bing")) %>%
  count(word, sentiment, sort = TRUE)

head(bing_sentiments)

# Sentiment Count (Positive vs Negative)
sentiment_count <- bing_sentiments %>%
  count(sentiment)

ggplot(sentiment_count, aes(x = sentiment, y = n, fill = sentiment)) +
  geom_bar(stat = "identity") +
  theme_minimal() +
  labs(title = "Sentiment Distribution of Tweets",
       x = "Sentiment", y = "Count")

# Top Words in Each Sentiment
top_words <- bing_sentiments %>%
  group_by(sentiment) %>%
  top_n(10, n) %>%
  ungroup() %>%
  arrange(sentiment, -n)

ggplot(top_words, aes(x = reorder(word, n), y = n, fill = sentiment)) +
  geom_col(show.legend = FALSE) +
  facet_wrap(~sentiment, scales = "free") +
```

```
coord_flip() +
labs(title = "Top Words by Sentiment",
    x = "Words", y = "Frequency")
```