Session 6 - Twitter Data

Alex Mounsey

21/11/2020

Exercise

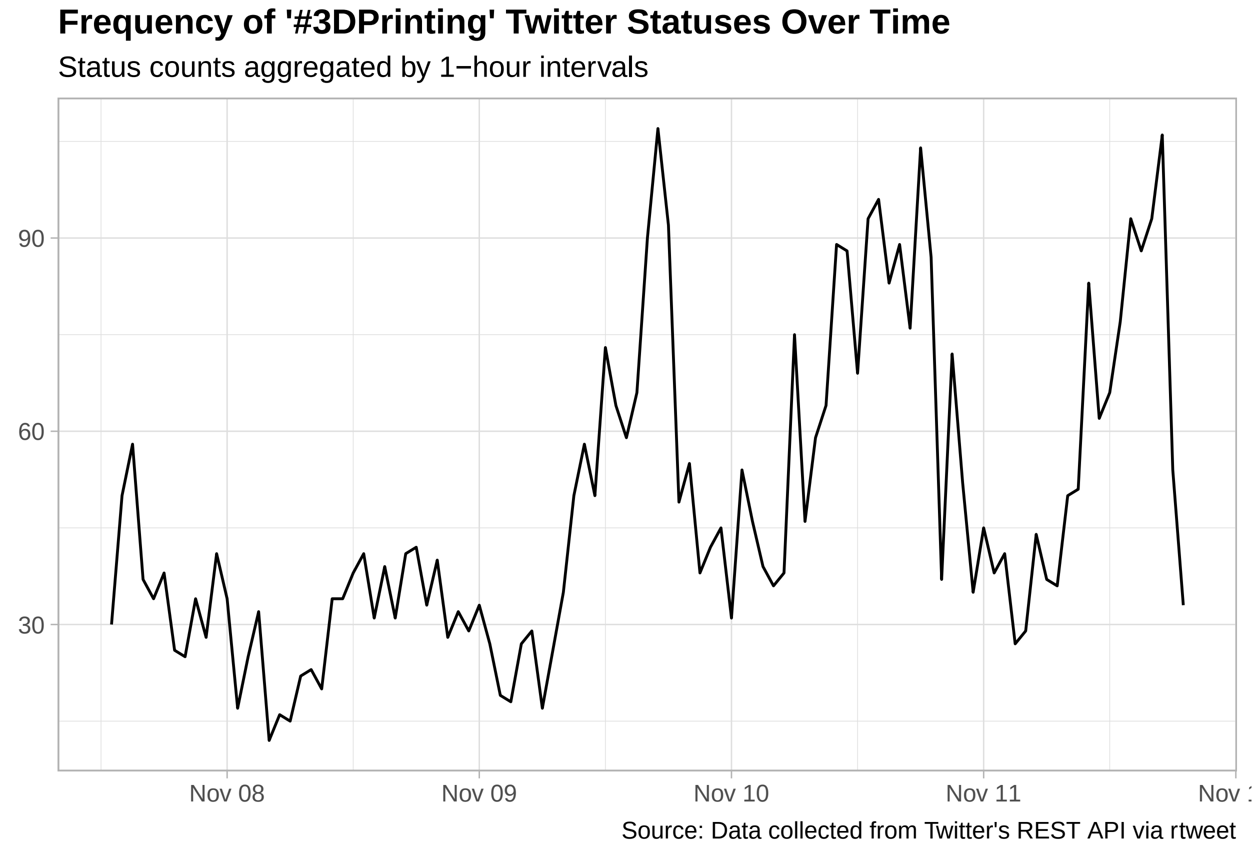
Extract Twitter data on a topic of your choice. Perform the necessary data manipulation to produce interesting data visualizations and analysis of your Twitter data.

Importing Data

```
tweets <- read.csv(’./tweets.csv’)
users <- read.csv(’./users.csv’)
```

Plotting Time-Series Data

```
ts_plot(tweets, ’hours’) +
  theme_light() + theme(plot.title = element_text(face = ’bold’)) +
  labs(x = NULL, y = NULL,
       title = "Frequency of ’#3DPrinting’ Twitter Statuses Over Time",
       subtitle = "Status counts aggregated by 1-hour intervals",
       caption = "Source: Data collected from Twitter’s REST API via rtweet")
```


Frequency of '#3DPrinting' Twitter Statuses Over Time

Status counts aggregated by 1−hour intervals

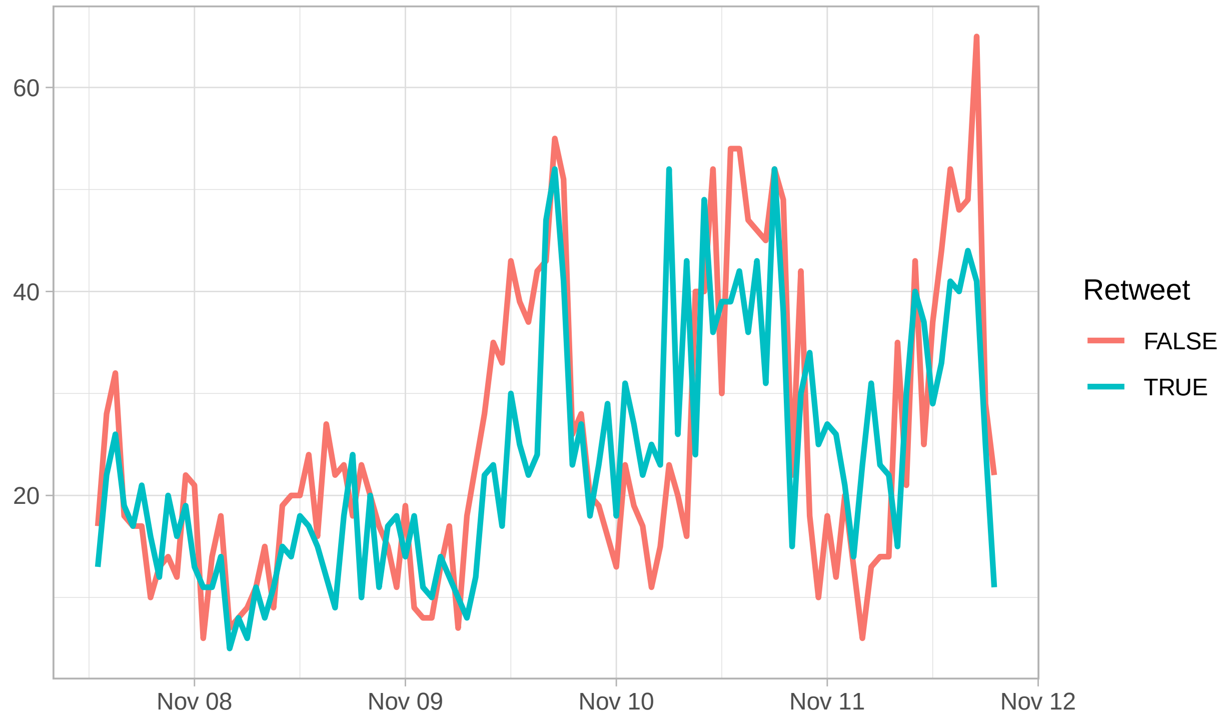
Source: Data collected from Twitter's REST API via rtweet

Plotting Tweets vs. Re-Tweets

```
tweets %>%
  group_by(is_retweet) %>%
  ts_plot(’hours’, lwd = 1) +
    theme_light() + theme(plot.title = element_text(face = ’bold’)) +
    labs(x = NULL, y = NULL,
         title = "Frequency of ’#3DPrinting’ Twitter Statuses Over Time",
         subtitle = "Status counts aggregated by 1-hour intervals",
         caption = "Source: Data collected from Twitter’s REST API via rtweet",
         colour = "Retweet")
```

Frequency of '#3DPrinting' Twitter Statuses Over Time

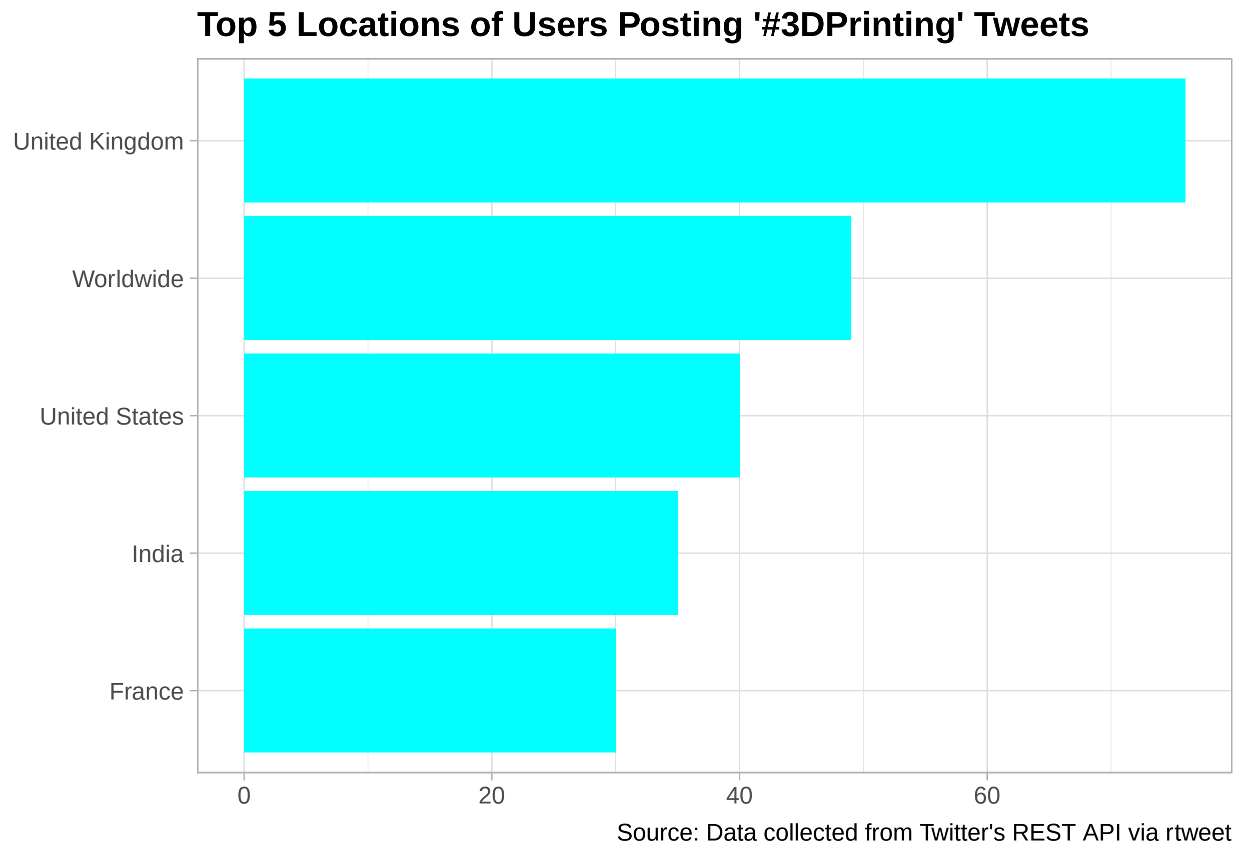
Status counts aggregated by 1−hour intervals



Source: Data collected from Twitter's REST API via rtweet

Plotting User Locations

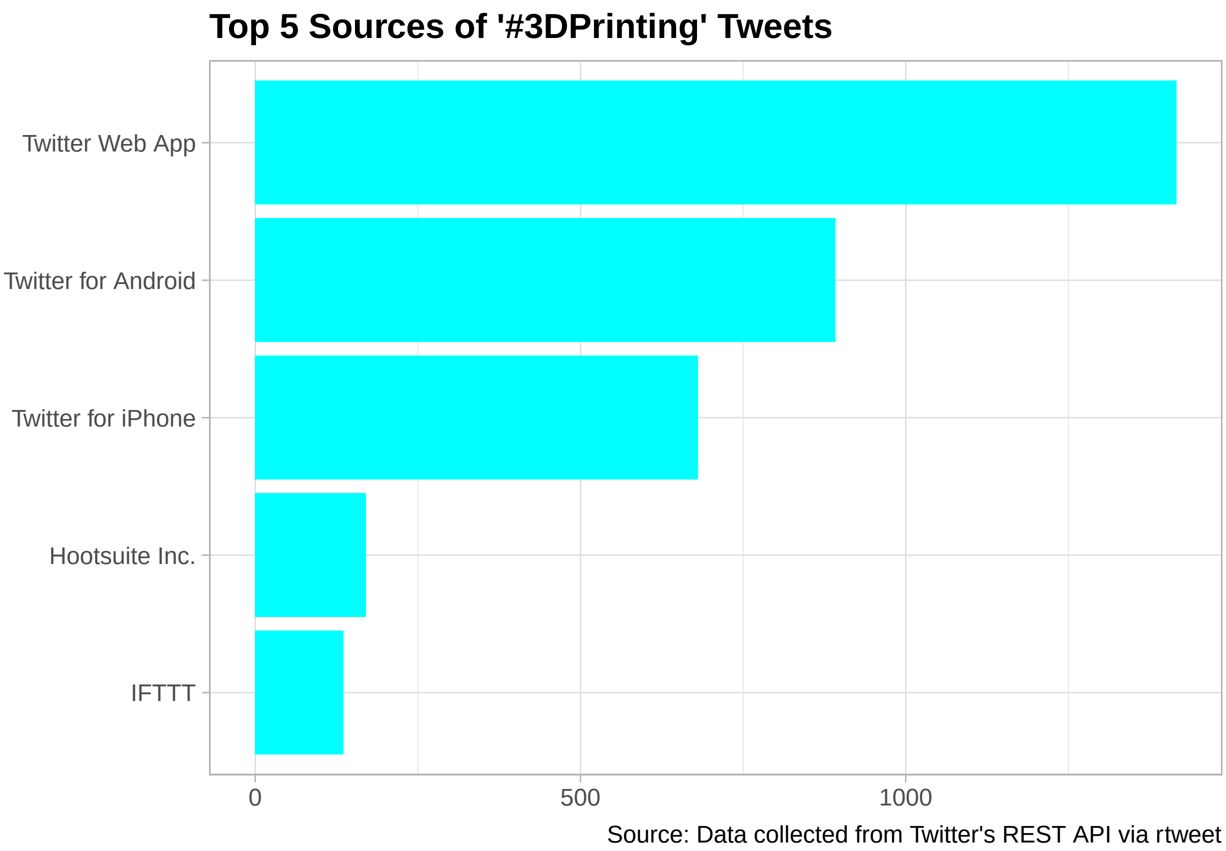
```
# Combine similar locations
users <- users %>%
  mutate(location_rec = recode(location,
                               ’Paris, France’ = ’France’,
                               ’Earth’ = ’Worldwide’, ’Global’ = ’Worldwide’,
                               ’International’ = ’Worldwide’,
                               ’Europe’ = ’Worldwide’,
                               ’UK’ = ’United Kingdom’,
                               ’London’ = ’United Kingdom’,
                               ’London, UK’ = ’United Kingdom’,
                               ’London, England’ = ’United Kingdom’,
                               ’Bengaluru, India’ = ’India’,
                               ’England, United Kingdom’ = ’United Kingdom’,
                               ’Hyderabad, India’ = ’India’,
                               ’Pune, India’ = ’India’,
                               ’San Francisco, CA’ = ’United States’,
                               ’Seattle, WA’ = ’United States’))

# Plot user location frequency
users %>%
  count(location_rec, sort = TRUE) %>% # Count the frequency of each location
  mutate(location_rec = reorder(location_rec, n)) %>% # Order locations by frequency
  na.omit() %>% # Remove NA values
  head(5) %>% # Select the top locations
  ggplot(aes(x = location_rec, y = n)) +
    geom_col(fill = 'cyan') + coord_flip() +
    theme_light() + theme(plot.title = element_text(face = 'bold')) +
    labs(x = NULL, y = NULL,
         title = "Top 5 Locations of Users Posting '#3DPrinting' Tweets",
         caption = "Source: Data collected from Twitter's REST API via rtweet")
```



Plotting the Top Devices Used to Tweet

```
tweets %>%
  group_by(source) %>%
  summarise(Total = n()) %>%
  arrange(desc(Total)) %>%
  head(5) %>%
  ggplot(aes(reorder(source, Total), Total, fill = source)) +
    geom_col(fill = 'cyan') + coord_flip() +
    theme_light() + theme(plot.title = element_text(face = 'bold')) +
    labs(x = NULL, y = NULL,
         title = "Top 5 Sources of '#3DPrinting' Tweets",
         caption = "Source: Data collected from Twitter's REST API via rtweet")
```

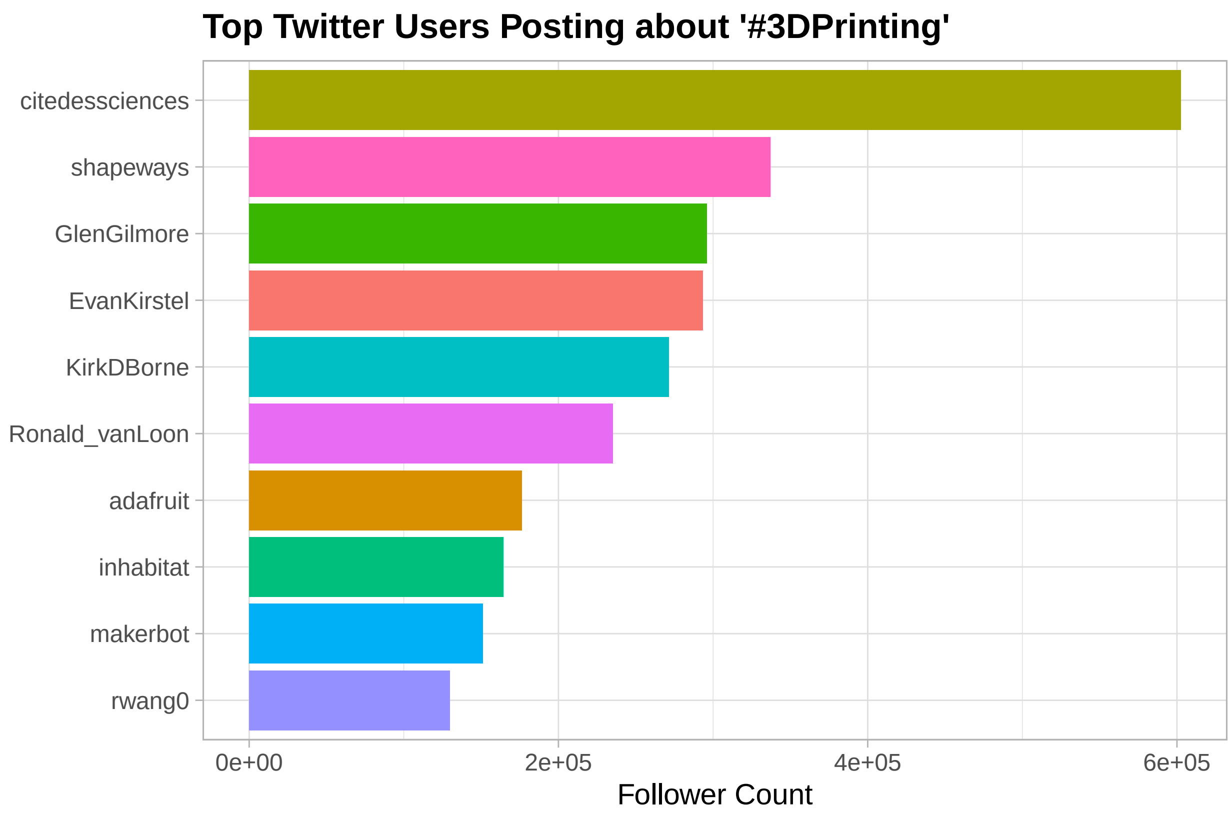
```
## `summarise()` ungrouping output (override with `.groups` argument)
```


Top 5 Sources of '#3DPrinting' Tweets

Source: Data collected from Twitter's REST API via rtweet

Plotting User Follower Counts

```
users %>%
  group_by(screen_name) %>%
  arrange(desc(followers_count)) %>%
  head(10) %>%
  ggplot(aes(reorder(screen_name, followers_count), followers_count, fill = name)) +
    geom_col() + coord_flip() +
    theme_light() + theme(plot.title = element_text(face = ’bold’), legend.position = ’none’) +
    labs(x = NULL, y = ’Follower Count’,
         title = "Top Twitter Users Posting about ’#3DPrinting’",
         caption = "Source: Data collected from Twitter’s REST API via rtweet")
```

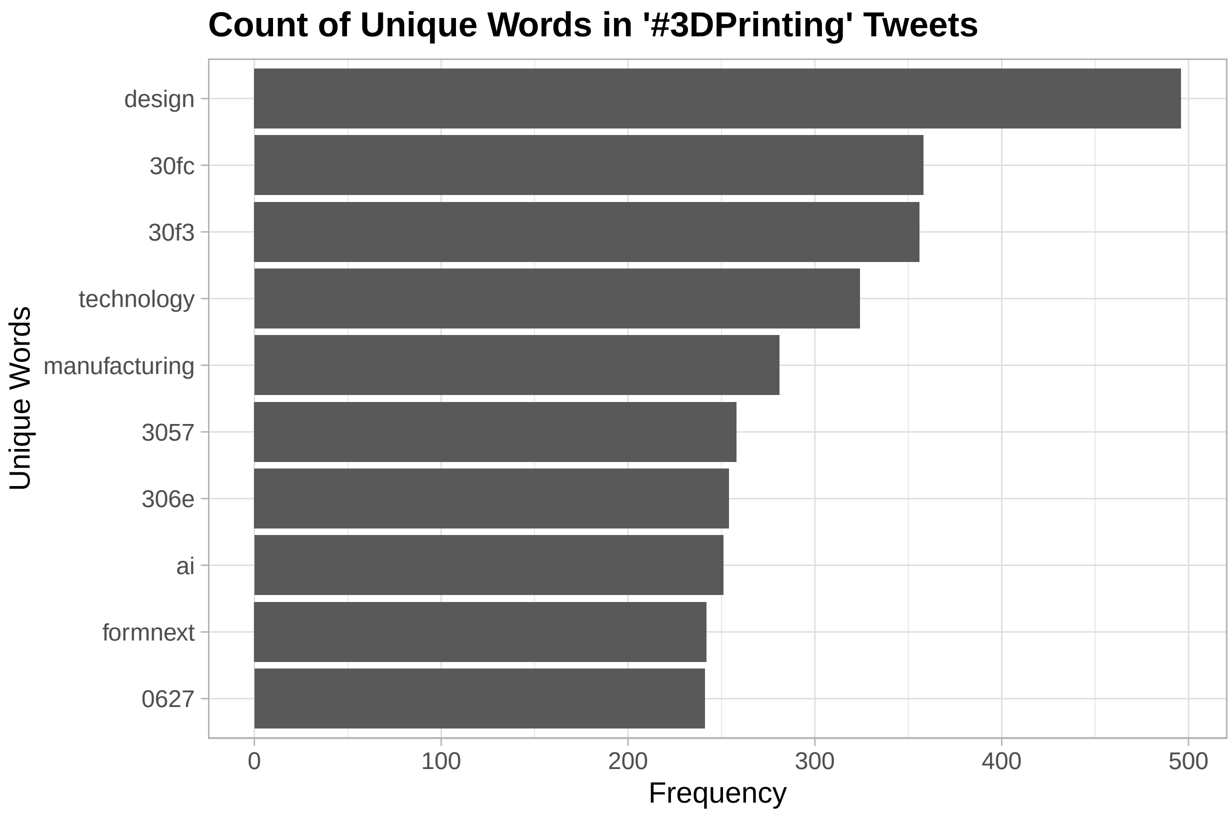


Source: Data collected from Twitter's REST API via rtweet

Word Frequencies

```r
# Remove http elements
tweets$stripped_text <- gsub('http.*', '', tweets$text)
tweets$stripped_text <- gsub('https.*', '', tweets$stripped_text)
tweets$stripped_text <- gsub('amp', '', tweets$stripped_text)

# Remove punctuation, convert to lowercase, and add an ID for each tweet
tweets_clean <- tweets %>%
  select(stripped_text) %>%
  mutate(tweetnumber = row_number()) %>%
  unnest_tokens(word, stripped_text)

# Load a list of stop words, and remove them from the list of words
data("stop_words")
cleaned_tweet_words <- tweets_clean %>%
  anti_join(stop_words)

# Create a custom list of stop words, and remove them from the list of words
custom_stop_words <- data.frame(word = c('3dprinting', '3d', 'printing', 'printed', 'printer', 'print',
cleaned_tweet_words <- cleaned_tweet_words %>%
  anti_join(custom_stop_words)

# Plot word frequencies
```

```
cleaned_tweet_words %>%
  # Count the number of occurrences of each word, and sort by that count
  count(word, sort = TRUE) %>%
  head(10) %>%
  mutate(word = reorder(word, n)) %>%
  ggplot(aes(x = word, y = n)) +
    geom_col() + coord_flip() +
    theme_light() + theme(plot.title = element_text(face = ’bold’)) +
    labs(x = "Unique Words", y = "Frequency",
         title = "Count of Unique Words in ’#3DPrinting’ Tweets",
         caption = "Source: Data collected from Twitter’s REST API via rtweet")
```


Count of Unique Words in '#3DPrinting' Tweets

Source: Data collected from Twitter's REST API via rtweet