Sentiment Analysis

CHINDU

The dataset used for this analysis is the "Twitter US Airline Sentiment" dataset from Kaggle. We will be analysing the positive and negative sentiments in this data.

Data Preparation

```
library(tidytext)
library(dplyr)
```

```
##
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':
##
##     filter, lag

## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
library(ggplot2)
tweet <- read.csv('Tweets.csv')
prop.table(table(tweet$airline_sentiment))
```

```
##
##  negative   neutral  positive
## 0.6269126 0.2116803 0.1614071
```

```
tweet$text <- gsub("^@\\w+ *", "", tweet$text)  # remove @airline
```

unnest_tokens deal with punctuations and lowercase, we just need to worry about the other preprocessings requiried on the data

```
library(tm)
```

```
## Loading required package: NLP

##
## Attaching package: 'NLP'
```

```
## The following object is masked from 'package:ggplot2':
##
##     annotate
```

```
tweet_data <- subset(tweet, airline_sentiment != 'neutral')
#tweet_data <- subset(tweet_data, select=c('tweet_id','airline_sentiment', 'text','airline'))
tweet_data$text<-  gsub("\\W|\\d|http\\w?", " ", tweet_data$text, perl = T)
# Change special characters to english letters
library(stringi)
tweet_data$text<-stringi::stri_trans_general(tweet_data$text, "latin-ascii")
```

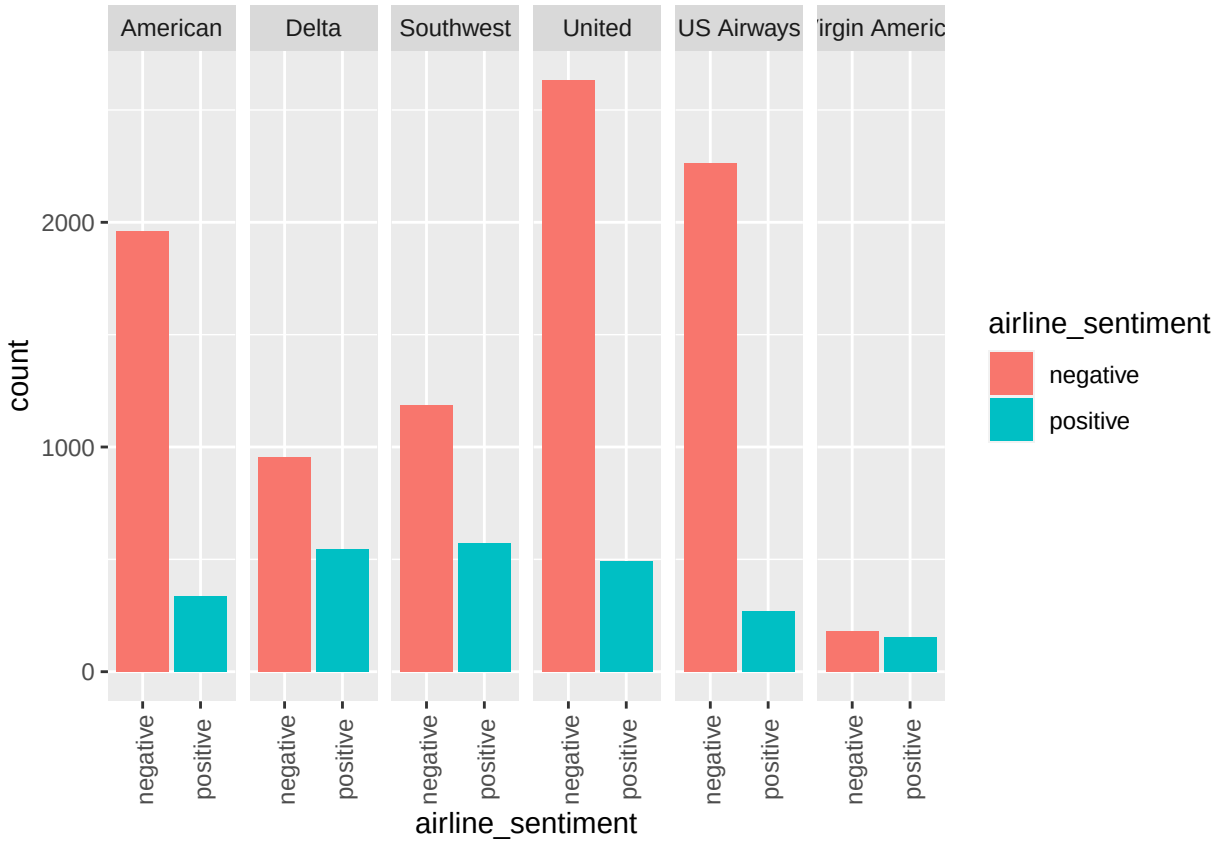
Understanding how many tweets are there for each airline

```
tweet_data %>% group_by(airline) %>%
  summarise(Total_tweets=n_distinct(tweet_id))
```

```
## # A tibble: 6 x 2
##   airline        Total_tweets
##   <fct>                 <int>
## 1 American               2180
## 2 Delta                  1499
## 3 Southwest              1756
## 4 United                 3125
## 5 US Airways             2532
## 6 Virgin America          333
```

A breakdown of negative and positive tweets by each airline

```
ggplot(tweet_data, aes(x = airline_sentiment, fill = airline_sentiment)) +
  geom_bar() +
  facet_grid(. ~ airline) +
  theme(axis.text.x = element_text(angle=90, vjust=0.6))
```



Unnest token and remove stop words for further analysis

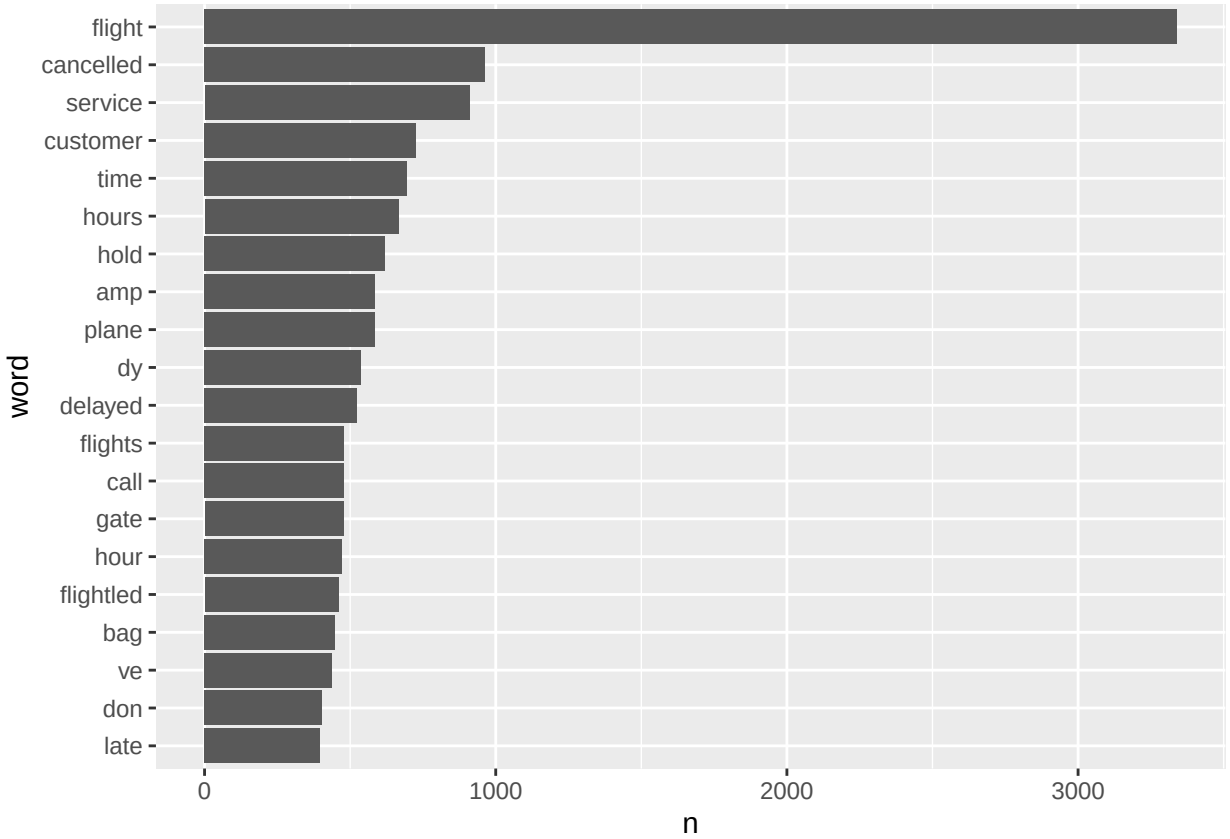
```
tweet_data_token<- tweet_data%>%
  unnest_tokens(word,text)%>%
  anti_join(stop_words)
```

```
## Joining, by = "word"
```

Checking the common words

```
library(ggplot2)
word_count<-tweet_data_token %>%
  count(word,sort=TRUE)
word_count %>%
  top_n(20) %>%
  mutate(word = reorder(word, n)) %>%
  ggplot(aes(word, n)) +
  geom_col() +
  coord_flip()
```

```
## Selecting by n
```



The common words are flight, cancelled,service,customer,time,hours,hold etc.

Since we are looking at the Airline data, the flight is common and should not be on much value in this analysis and hence removed.

```
tweet_data_token2 <- subset(tweet_data_token, word != 'flight')
tweet_data_token2 <- subset(tweet_data_token2, word != 'dy')
```

#Creat a word cloud of the positive words

```
library(wordcloud)
```

```
## Loading required package: RColorBrewer
```

```
Positive_tweet<-tweet_data_token2 %>%
  filter(airline_sentiment == "positive")
pal = brewer.pal(9,"BuGn")
  wordcloud(Positive_tweet[,15],
          max.words = 50,
          random.order=FALSE,
          rot.per=0.30,
          use.r.layout=FALSE,
          colors = (pal))
```

```
## Warning in tm_map.SimpleCorpus(corpus, tm::removePunctuation): transformation
## drops documents

## Warning in tm_map.SimpleCorpus(corpus, function(x) tm::removeWords(x,
## tm::stopwords())): transformation drops documents
```

Creat a word cloud of the Negative words

```
library(wordcloud)
Negative_tweet<-tweet_data_token2 %>%
  filter(airline_sentiment == "negative")
pal = brewer.pal(8,"Dark2")
  wordcloud(Negative_tweet[,15],min.freq=1,
          max.words = 50,
          random.order=FALSE,
          rot.per=0.35,
          colors = (pal))
```

```
## Warning in tm_map.SimpleCorpus(corpus, tm::removePunctuation): transformation
## drops documents

## Warning in tm_map.SimpleCorpus(corpus, function(x) tm::removeWords(x,
## tm::stopwords())): transformation drops documents
```

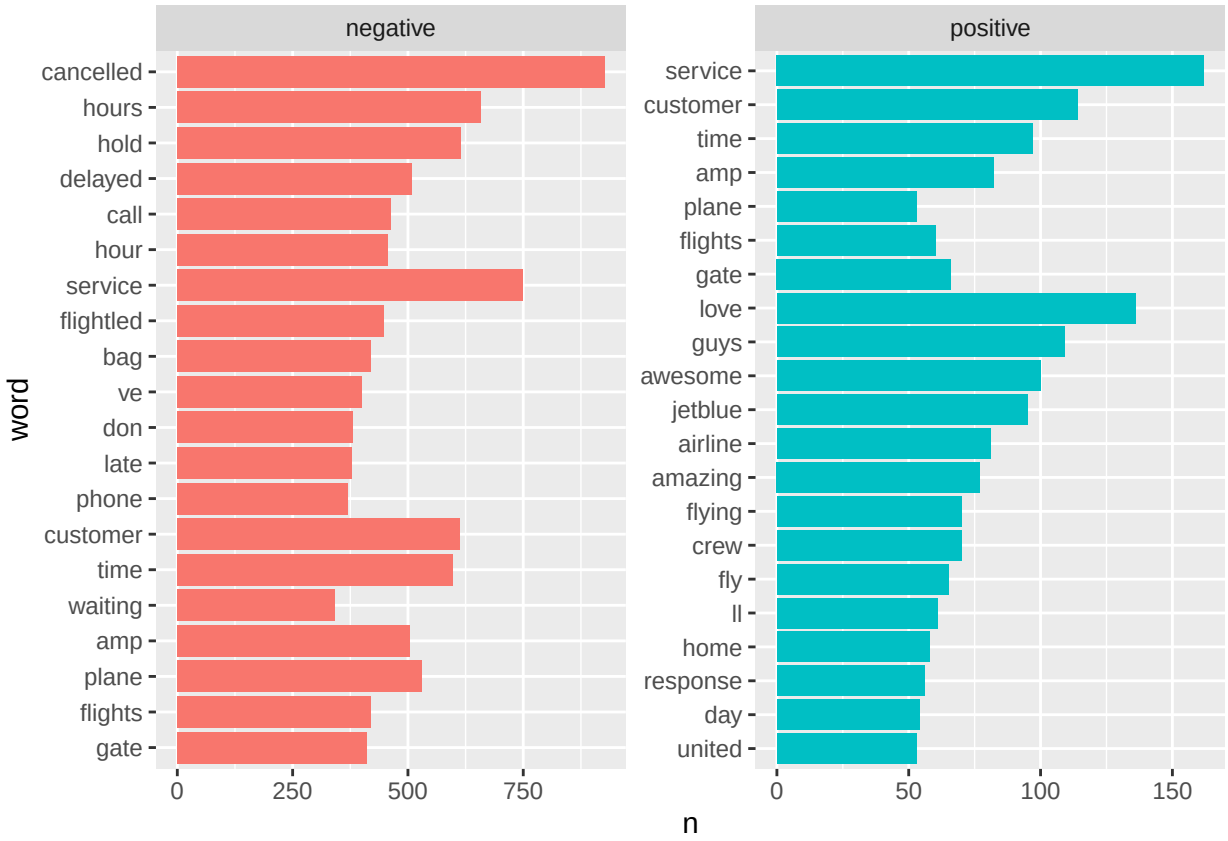
Understanding the words that are influencing the sentiment score

```
word_counts<- tweet_data_token2 %>%
  count(word,airline_sentiment)

top_words <- word_counts %>%
  group_by(airline_sentiment) %>%
  top_n(20) %>%
  ungroup() %>%
  mutate(word = reorder(word, n))
```

```
## Selecting by n
```

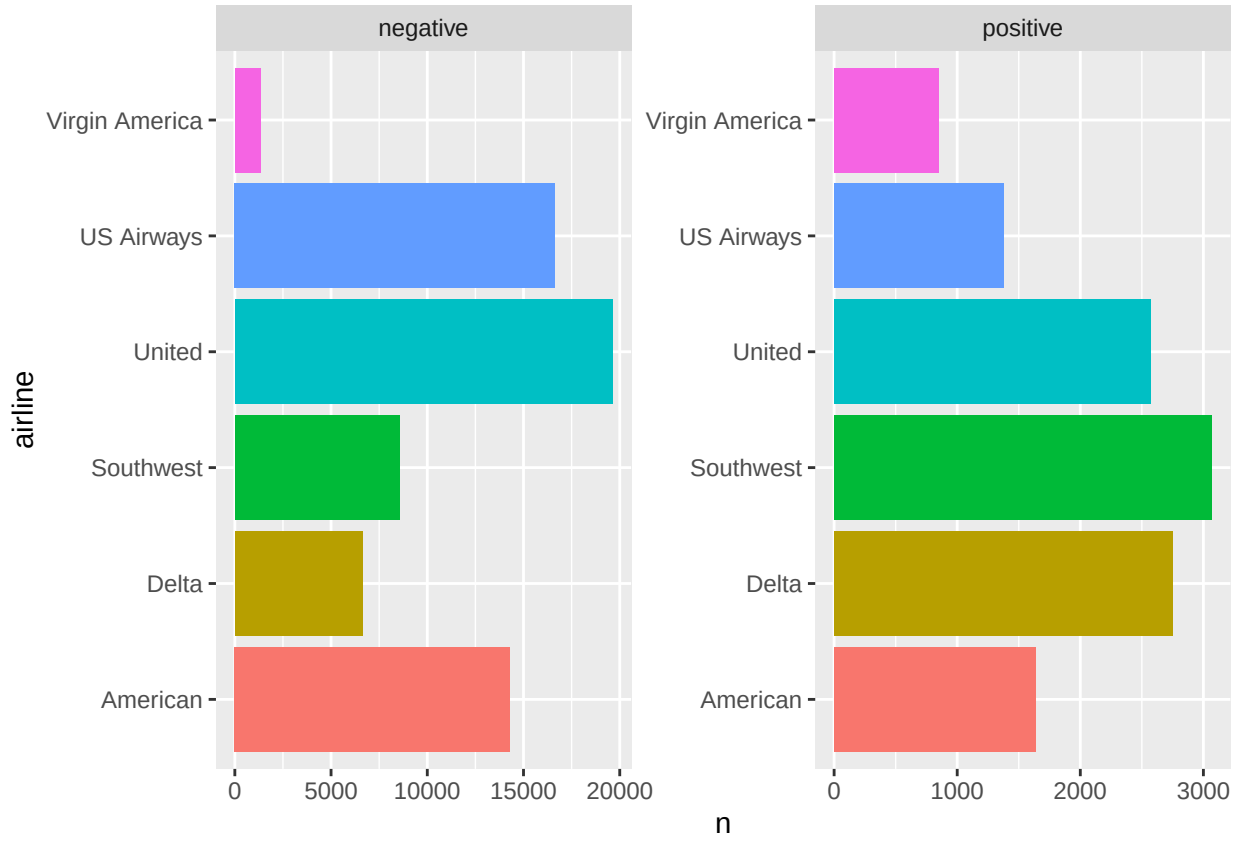
```
ggplot(top_words, aes(word, n, fill = airline_sentiment)) +
  geom_col(show.legend = FALSE) +
  facet_wrap(~airline_sentiment, scales = "free") +
  coord_flip()
```



We should always check which words are contributing to the sentiment scores. Depending on the dataset it may not be what you want.

Comparison of positive and negarive reation by airline

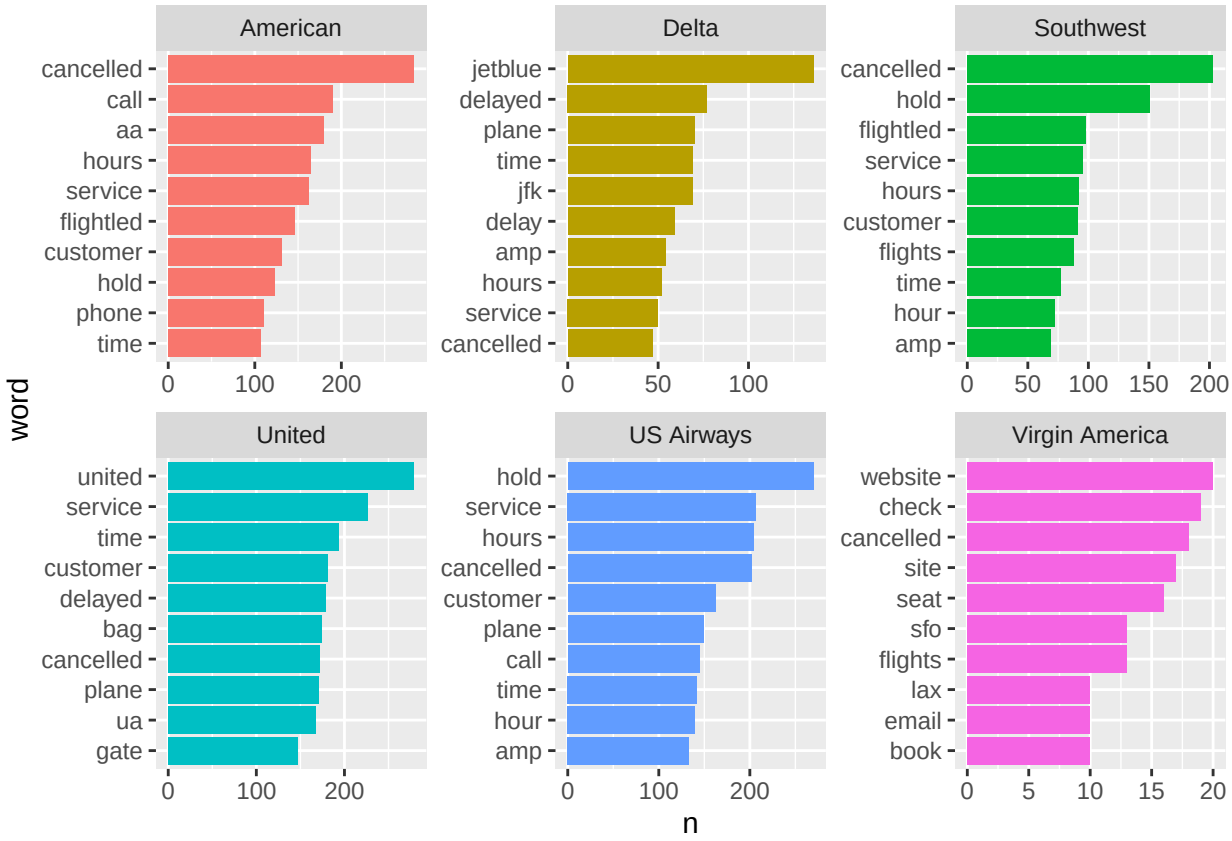
```
tweet_data_token2 %>%
    count(airline, airline_sentiment) %>%
    group_by(airline) %>%
    ggplot(aes(airline, n, fill = airline)) +
    geom_col(show.legend = FALSE) +
    facet_wrap(~ airline_sentiment, scales = "free") +
    coord_flip()
```




```
new2 <- subset(tweet_data_token2, select=c('airline','airline_sentiment','word'))
```

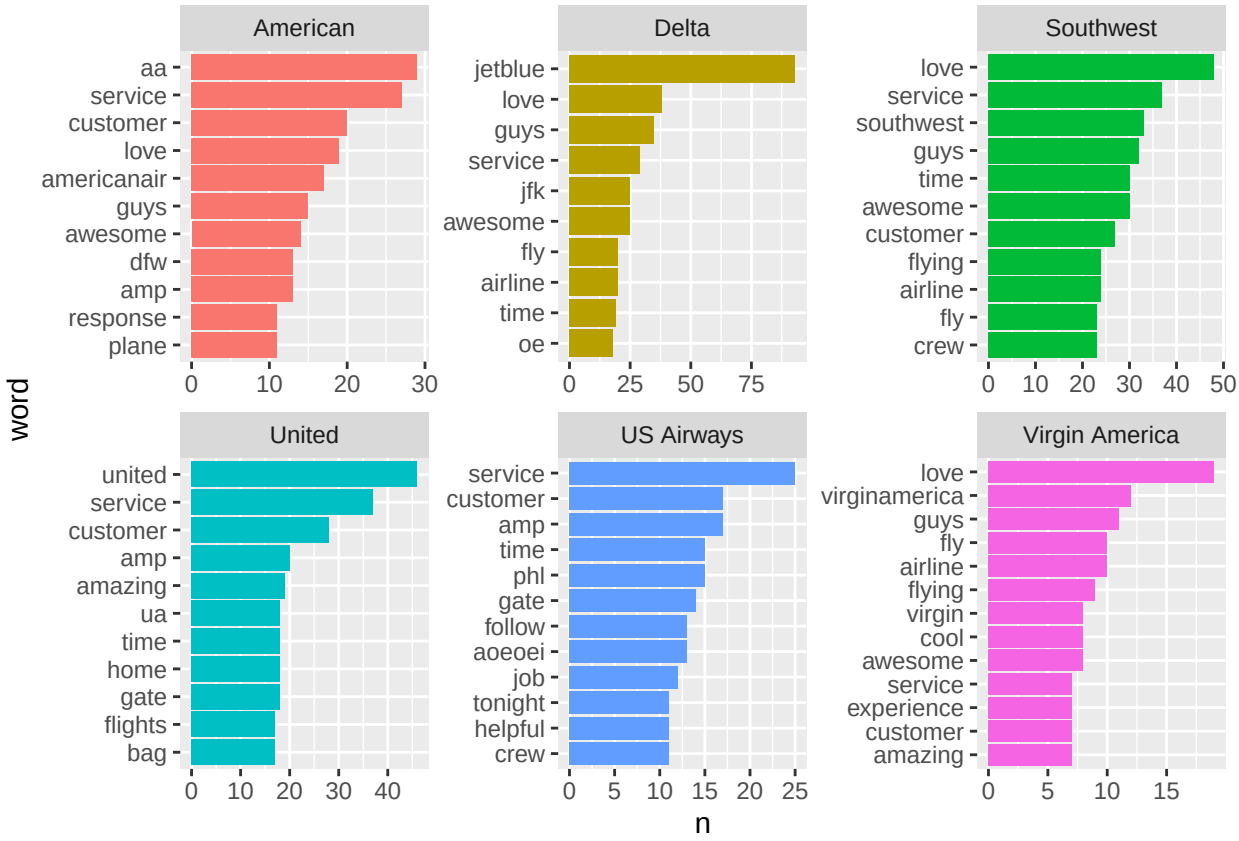
Analyse top negative words for each airline

```
new2 %>%
    filter(airline_sentiment == "negative") %>%
    count(word, airline) %>%
    group_by(airline) %>%
    top_n(10, n) %>%
    ungroup() %>%
    mutate(word = reorder(paste(word, airline, sep = "__"), n)) %>%
    ggplot(aes(word, n, fill = airline)) +
    geom_col(show.legend = FALSE) +
    scale_x_discrete(labels = function(x) gsub("__.+$", "", x)) +
    facet_wrap(~ airline, nrow = 2, scales = "free") +
    coord_flip()
```



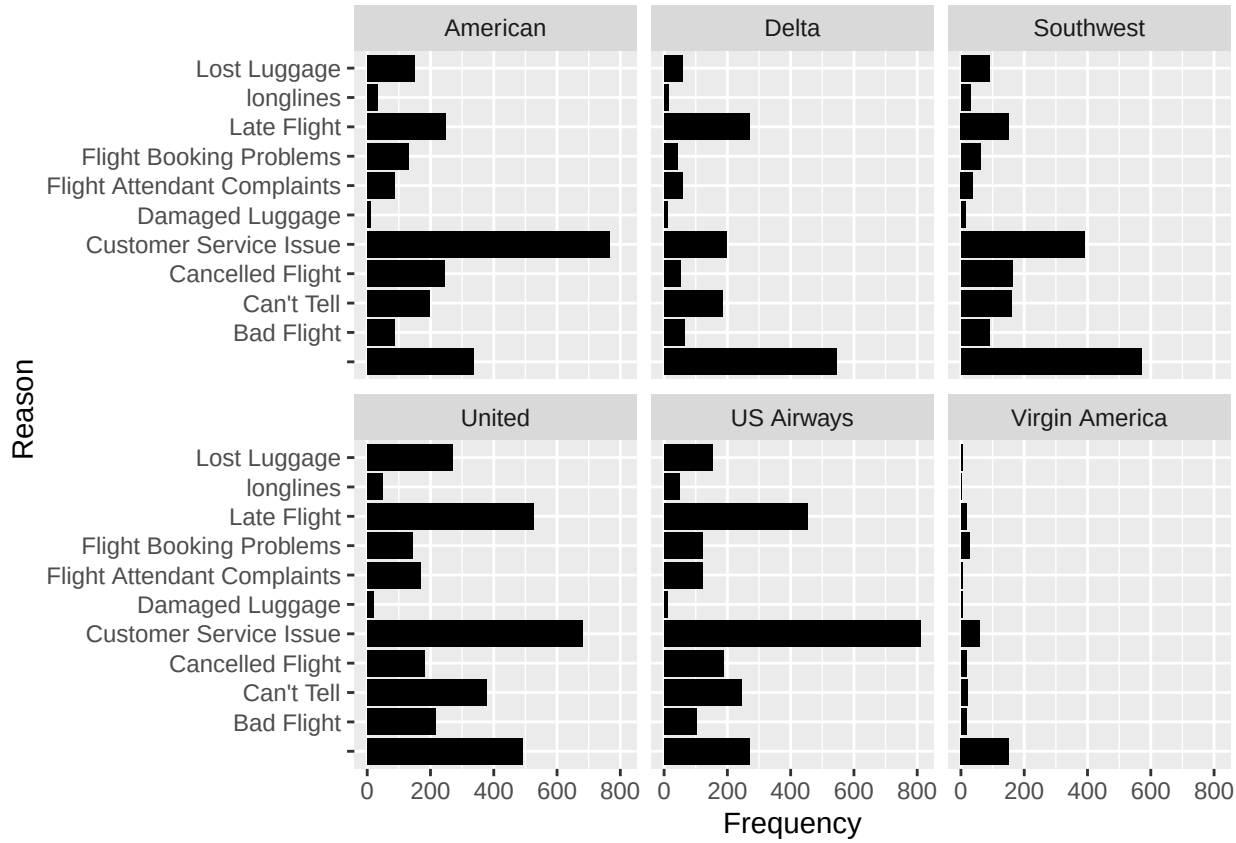
Analyse top Positive words for each airline

```
new2 %>%
    filter(airline_sentiment == "positive") %>%
    count(word, airline) %>%
    group_by(airline) %>%
    top_n(10, n) %>%
    ungroup() %>%
    mutate(word = reorder(paste(word, airline, sep = "__"), n)) %>%
    ggplot(aes(word, n, fill = airline)) +
    geom_col(show.legend = FALSE) +
    scale_x_discrete(labels = function(x) gsub("__.+$", "", x)) +
    facet_wrap(~ airline, nrow = 2, scales = "free") +
    coord_flip()
```



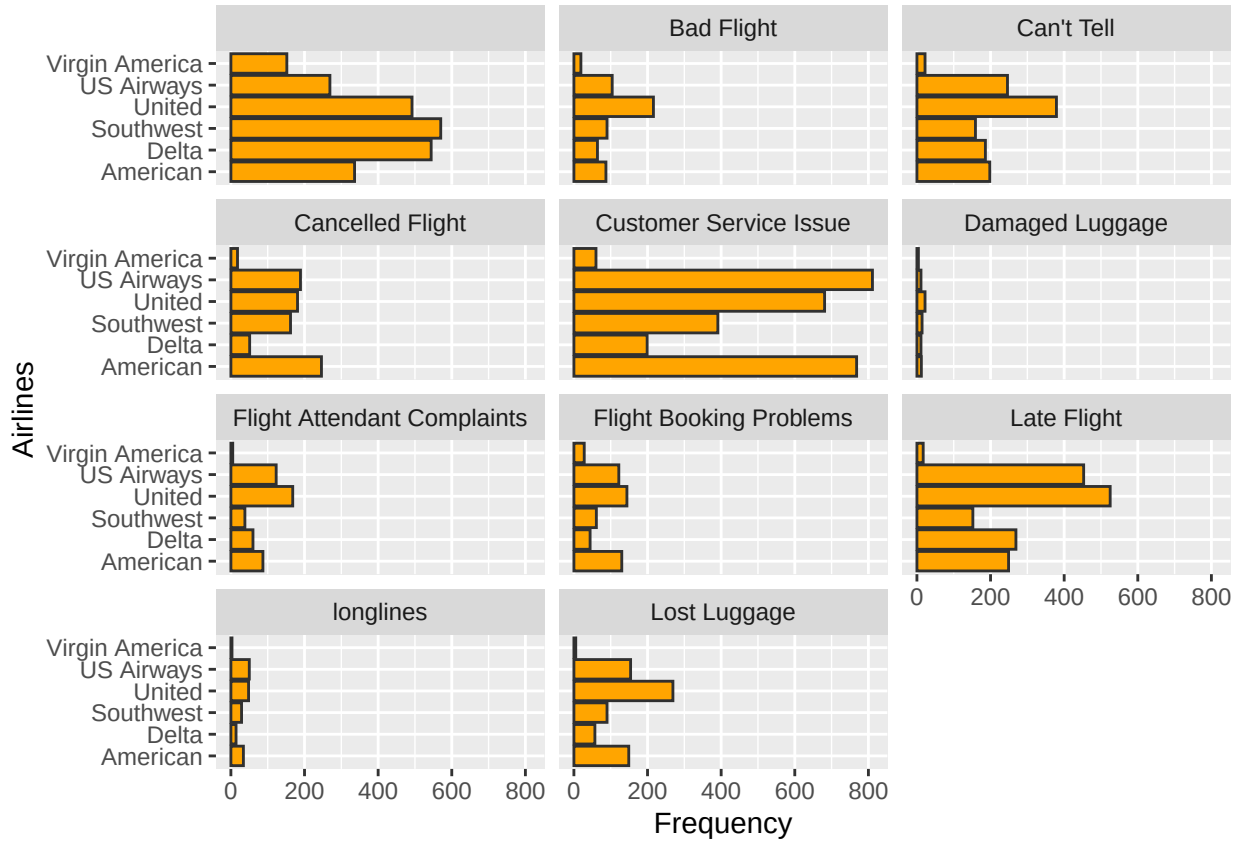
Understanding the reasons for negative sentiments

```
tweet_data %>%
filter(!is.na(negativereason)) %>%
count(airline, negativereason) %>%
ggplot(aes(negativereason, n))+
geom_bar(stat = "identity", fill = "black")+
facet_wrap(~airline, ncol = 3)+
labs(x = "Reason", y = "Frequency")+
coord_flip()
```



This bar chart helps us to identify the main issues related to each airlines. We cant really compare which airline is better from this data as the distribution of tweets for each airline is severely imbalanced. We can however conclude that one of the main issue is the terrible Customer service.

```
tweet_data %>%
filter(!is.na(negativereason)) %>%
count(airline, negativereason) %>%
ggplot(aes(airline, n))+
geom_bar(stat = "identity", colour = "grey19", fill = "orange")+
facet_wrap(~negativereason, ncol = 3)+
labs(x = "Airlines", y = "Frequency") +
  coord_flip()
```



Such plot can be useful in a situation where we extract samples of equal distribution of data for each airline. Ideally from this plot we could understand which airline is performing well or poorly for each reason.