Words distribution in Hotel reviews

Alejandro Moreno Díaz

Introduction to the problem

Hotel reviews are used by people to decide in which hotel they are going to stay in. These reviews help users to understand the positive and negative aspects of the hotel during their statement, so new customers who decide to stay, have an idea of how is going to be their experience. It also can help the staff of the hotel to solve the problems that customers have experienced before during their stay.

Usually, the users use a rating system of 5 stars, being 5 the best experience in the hotel and 1 the worst. However, not all the reviews follow this system and there could be some rating that does not match the description given; for instance, a user who had a specific problem could give a 4-star rating but the other users who had the same problem gave a rating less than 3 stars. Also, some reviews are just the description without any kind of rating.

To try to solve these problems it could be useful for analyzing the distribution of words in positive and negative reviews. Knowing the distributions could help identify outliers (for example, a positive review that most of the words are employed in negative reviews) and classify reviews without any kind of rating.

All the code used can be found in: https://github.com/alechunchun98/NLP-Project

Preprocessing the data

The libraries imported to perform the analysis on the data were:

```
library(tidyverse) # general utility & workflow functions
library(tidytext) # tidy implimentation of NLP methods
library(tm) # general text mining functions, making document term matrixes
library(SnowballC) # for stemming
library(wordcloud) # for making a wordcloud
```

The data used in this project consists of 3000 reviews of hotels, with a rating between 1 and 5 stars. The original dataset was obtained from Kaggle: https://www.kaggle.com/andrewmvd/trip-advisor-hotel-reviews/notebooks?datasetId=897156&sortBy=voteCount which has more than 20000 reviews.

Before starting the analysis of the distributions, the data was preprocessed. The first step was to eliminate the reviews with missing values. Next, the reviews with 4 or 5 stars were classified as positive, and the reviews with 1 or 2 stars as negative. The reviews with 3 stars are were dropped from the dataset having a total of 2642.

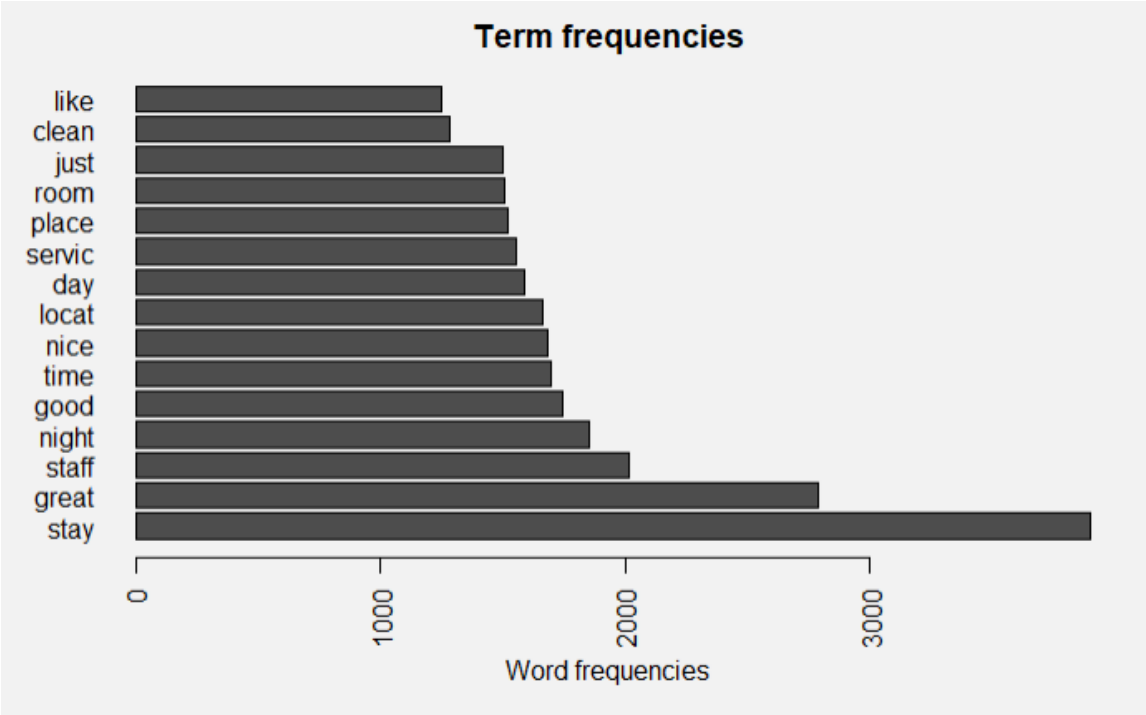
The general distribution of words

Then a corpus was generated using all the positive and negative reviews (removing the stopwords and numbers) and after that a matrix (Table 1) with the word frequencies and different types of visualizations (**Image 1** and more in the Markdown document).

	word	freq
stay	stay	3902
great	great	2785
staff	staff	2014
night	night	1850
good	good	1745
time	time	1693
nice	nice	1685
locat	locat	1660
day	day	1590
servic	servic	1553
place	place	1517
room	room	1506
just	just	1500

Table 1 (left): Matrix shows the frequency of the different words considering all the positive and negative reviews. The words displayed show the frequencies in a descending order until the frequency 1500.

Image 1 (bottom): Graphic bar to display the frequency of the different words considering all the positive and negative reviews. The words displayed are the first 15 words with the highest frequencies.



This graph shows the most used words in all the reviews so we can remove for the future analysis the ones with doesn't have an influence on the rating.

The differential distribution between positive and negative reviews

This time, we create two different corpora, one for the positive reviews and another one for the negative ones. After that, with each corpus, we follow the same process as the analysis of the general distribution. In the end, there were 2063 positive reviews and 579 negative reviews. Then, the same graph as the previous one was generated (**Image 2)** for the positive reviews and (**Image 3)** for the negative ones and show that the distribution of words is different for each graph (also, the matrixes can be viewed in the R-Markdown document).

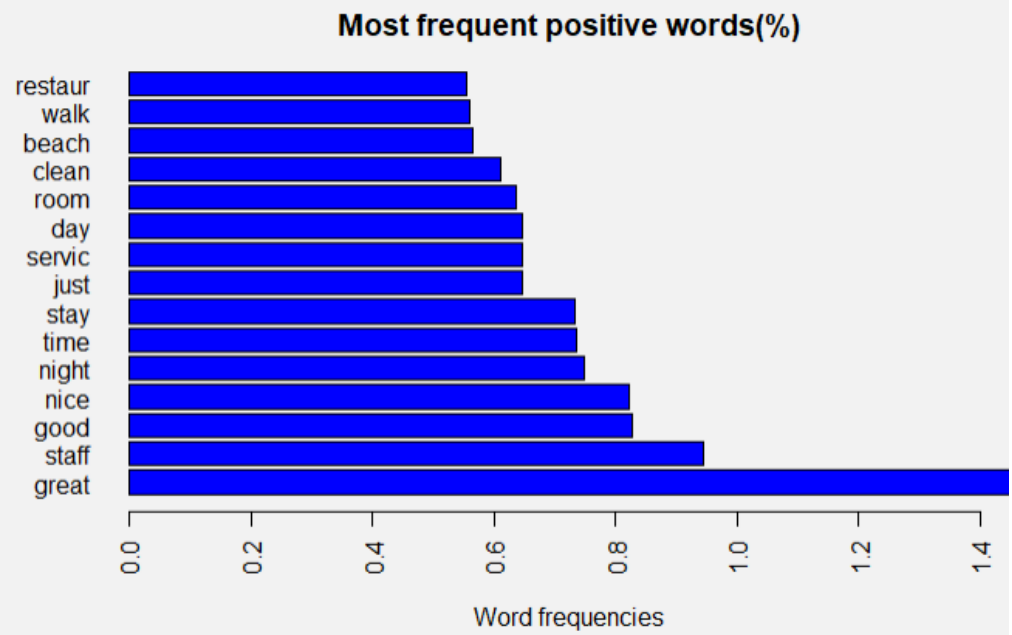


Image 2: Graphic bar to display the frequency of the different words considering all the positive reviews. The words displayed are the first 15 words with the highest percentages.

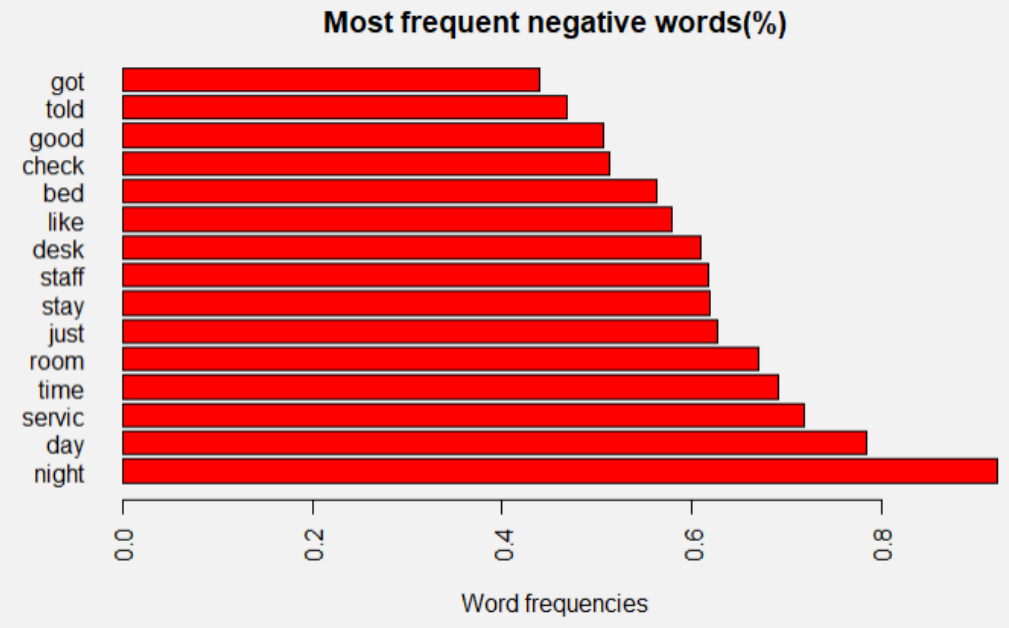


Image 3: Graphic bar to display the frequency of the different words considering all the negative reviews. The words displayed are the first 15 words with the highest percentages.

If we combine the graphs displayed to look for the highest difference for the distribution of a word we obtain the following graph (**Image 4**).

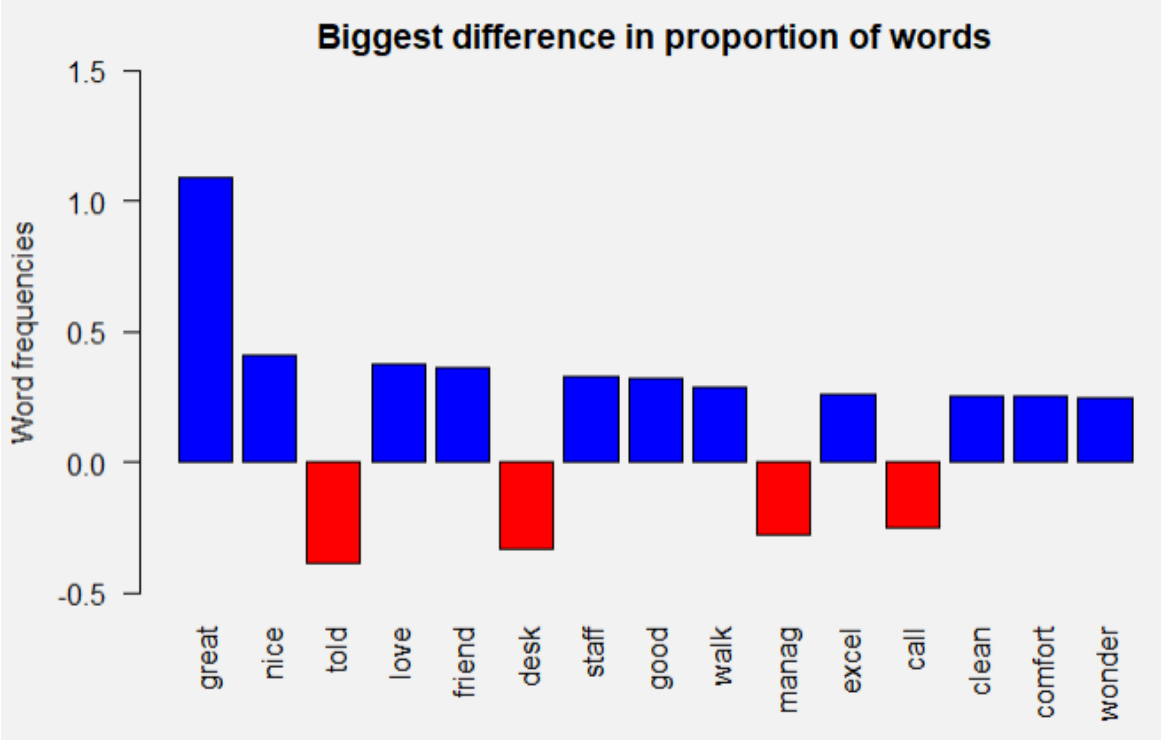


Image 4: The bars in blue are the ones that the proportion of positive use is greater than the negative. The red bars are the opposite case.

Looking and the hugest bars, the words “great, nice, love, and friend” have usually a positive meaning so it makes sense they are used more in positive reviews. The word “night” could be used more in negative reviews because the customers of the hotel could have some trouble sleeping at night. To go further, it could be analyzed the usage of some expressions that employ these words, like “bad night” Given the data frame df_prop (the one created that combines positive and negative relative frequencies in different columns) and the last graph, we can see the distribution of positive and negative reviews and which words differ in their relative frequency the most.

Conclusions and future projects

During the analysis of positive and negative reviews, it is shown that the number of positive reviews is greater than the negative ones. The last graphs comparing the distribution between positive and negative reviews shows clearly that the usage of words in positive and negative hotel reviews are different.

To improve the distribution of words it could be considered some of the stopwords like “should” that is saying that there is something to improve, so it will be strange that the review has a 5-star rating. Additionally, an association analysis could be done and see if the most used words have a negative word near(not, can´t, don´t, didn´t.), in this case, the word would have the opposite meaning.

Calculating the a priori probabilities using the length of the positive and negative corpus, and the a posteriori probabilities that can be obtained from the data frame df_prop; a simple Naive-Bayes classifier could be built. The data used in this project contains the first 3000 reviews from the original dataset, the rest of the dataset could be used to verify and improve the model. To go further, the reviews with a 3-star rating could be classified as positive and negative.