Ideatory DBS Text Mining Submission

Submitted by

Pronojit Saha

Ideatory username: pronojitsaha@gmail.com

Best Hotels

- Which are the best 5 hotels?

hotel_218524

hotel_478252

hotel_149399

hotel_150841

hotel_247957

Worst Hotels

- Which are the worst 5 hotels?

hotel_85003

hotel_100584

hotel_252969

hotel_305809

hotel_306174

Approach

- Combine and build a uniform dataset containing all hotel reviews.
- Perform text data transformations on the reviews for better analysis.
- An unsupervised learning problem: No labelled training data available
- Score the hotel reviews using a lexicon based review scoring algorithm built from scratch. Add each individual Review Score to compute total hotel Review Score.
- Aggregate scores for all hotels and rank as per the Review Score calculated to obtain the best and worst 5 hotels.

Problem Definition

- Hotel customers often leave reviews of their experience on websites.
- The problem is to use the reviews and build a learning model in an effective way to determine the underlying sentiment polarity of the reviews and thus determine the best and the worst hotels.

Methodology

1. Data Understanding

- Sources of data: list of 1500 files in '.dat' format made available by the client
- Data Description: Each hotel file has a number of customer reviews. Each review has the following structure:
    - Content: the actual content of the review
    - Date: the time the review was posted

2. Data Transformation

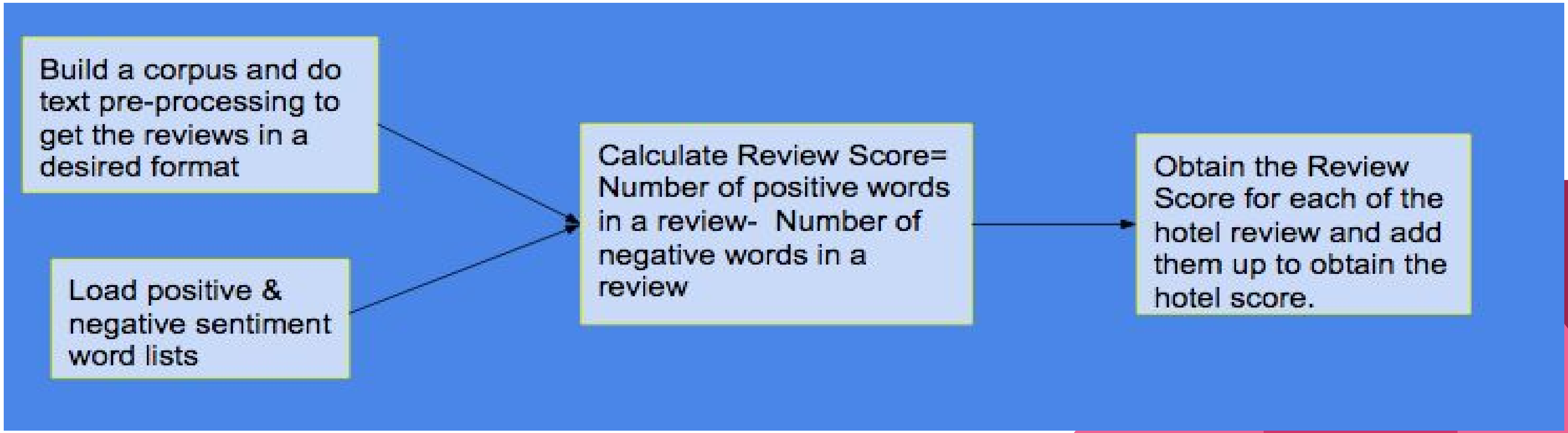
- build a corpus of hotel reviews
- lowercasing of review text
- remove punctuations
- remove stop words
- word stemming

Methodology (Contd..)

3. Algorithm Selection

- Supervised Machine Learning Algorithms (Naive Bayes, SVM, Tree based methods, etc): They have good predictive power but suffer from the drawback of requiring a labelled dataset for training the models, which is not available in the present task.
- Unsupervised Machine Learning Algorithms: These are primarily lexicon (a dictionary of words) based methods which are intuitive and easy to build. Most importantly they do not need any labelled data and hence was used for the task at hand.

4. Model Building



Build a corpus and do text pre-processing to get the reviews in a desired format

Load positive & negative sentiment word lists

Calculate Review Score= Number of positive words in a review- Number of negative words in a review

Obtain the Review Score for each of the hotel review and add them up to obtain the hotel score.

Methodology (Contd..)

5. Build Review Scoring Algorithm

- Utilize the opinion lexicon in English as built by Hu and Liu[1] containing a database of positive and negative sentiment words list.
- Define Review Score= Number of positive words in a review- Number of negative words in a review
- Find the Review Score of each hotel review. A positive review is expected to have more positive words and hence a higher positive Review Score and vice-versa.
- Add the review scores of all the reviews for a particular hotel and arrive at the final Review Score for the hotel.
- Do the above iteratively for each of the 1500 hotels.

6. Results

- Sort the hotels by their Review Scores and report the top 5 (best) and bottom 5 (worst) hotels.

Implementation

1.Analytical Tools/Software Used

○ R for data transformation, implementing the review scoring algorithm and final visualization of results

○ Domino online platform to execute all the R scripts on virtual machines at scale. Total execution time was 12 mins (approx.) on a 30GB 8 core virtual machine.

2.Data Construction

○ Build a common data frame in R containing all the 1500 hotel reviews along with hotel name. Total number of reviews: 212,308

3.Data Transformation

○ Tm package & SnowballC package in R were used for building corpus of all hotel reviews, lowercasing, removing punctuations, removing stop words and word stemming.

4.Model Build

○ Review Scoring algorithm built in R. The Stringr package was used to check the presence of positive and negative words in a review.

5.Visualization

○ ggplot and wordcloud package in R were used for creating visualizations.

Results

Best 5 Hotels

Total Reviews: 10,153

Hotel Name	Review Score
hotel_218524	33482
hotel_478252	19414
hotel_149399	14146
hotel_150841	13364
hotel_247957	10276

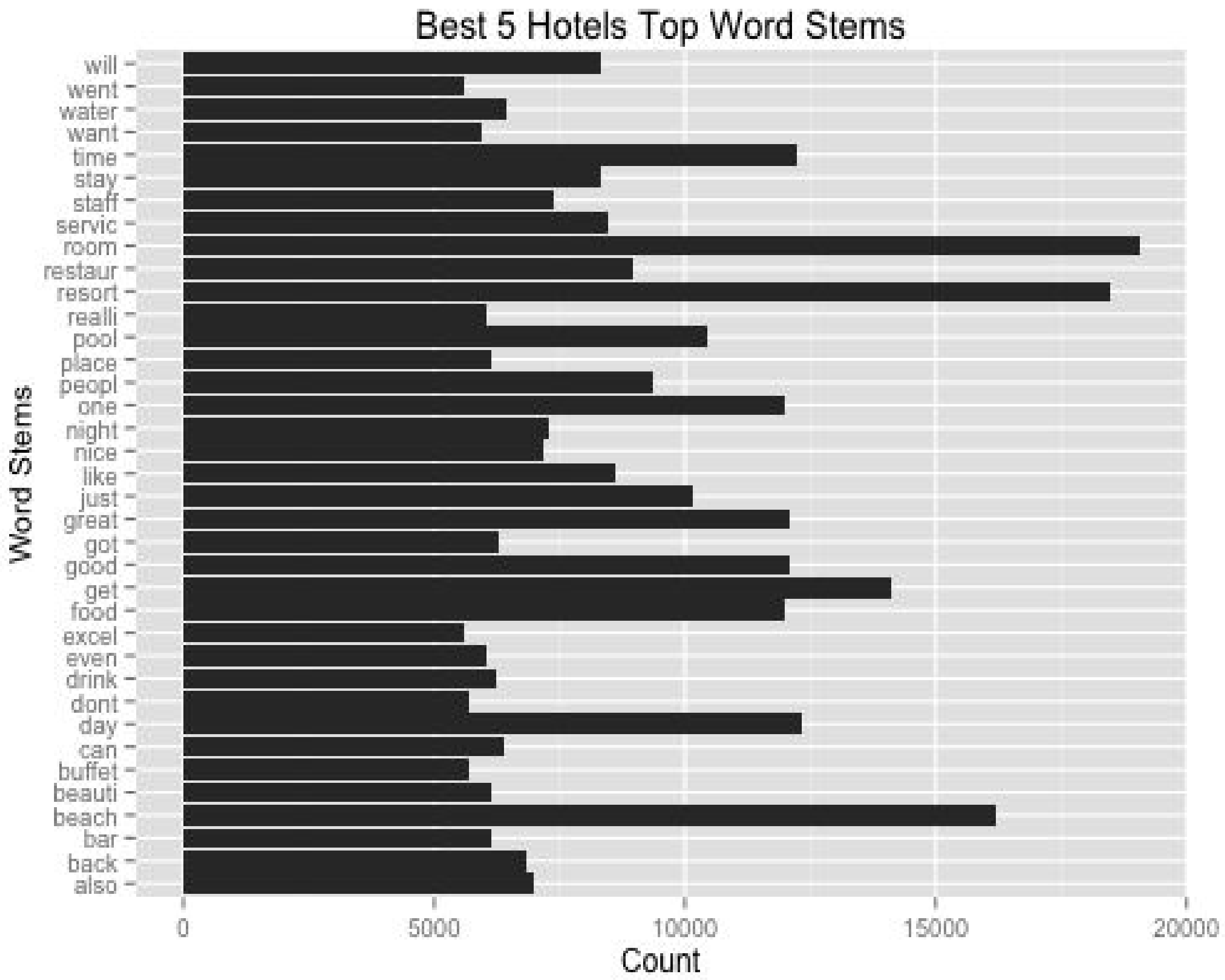
Worst 5 Hotels

Total Reviews: 343

Hotel Name	Review Score
hotel_85003	-138
hotel_100584	-99
hotel_252969	-82
hotel_305809	-59
hotel_306174	-58

- From total number of reviews, it is very clear that the best hotels garner a whole lot of reviews from their patrons.
- The top most hotel's score is more than 1.5 times the second one hence clearly the best by a good margin.
- The worst hotel's scores on the other hand have a more close range and also not much negative in absolute value w.r.t to the best hotel's score, indicating customers on the overall are satisfied with the services of the group of 1500 hotels.

Visualization of Results

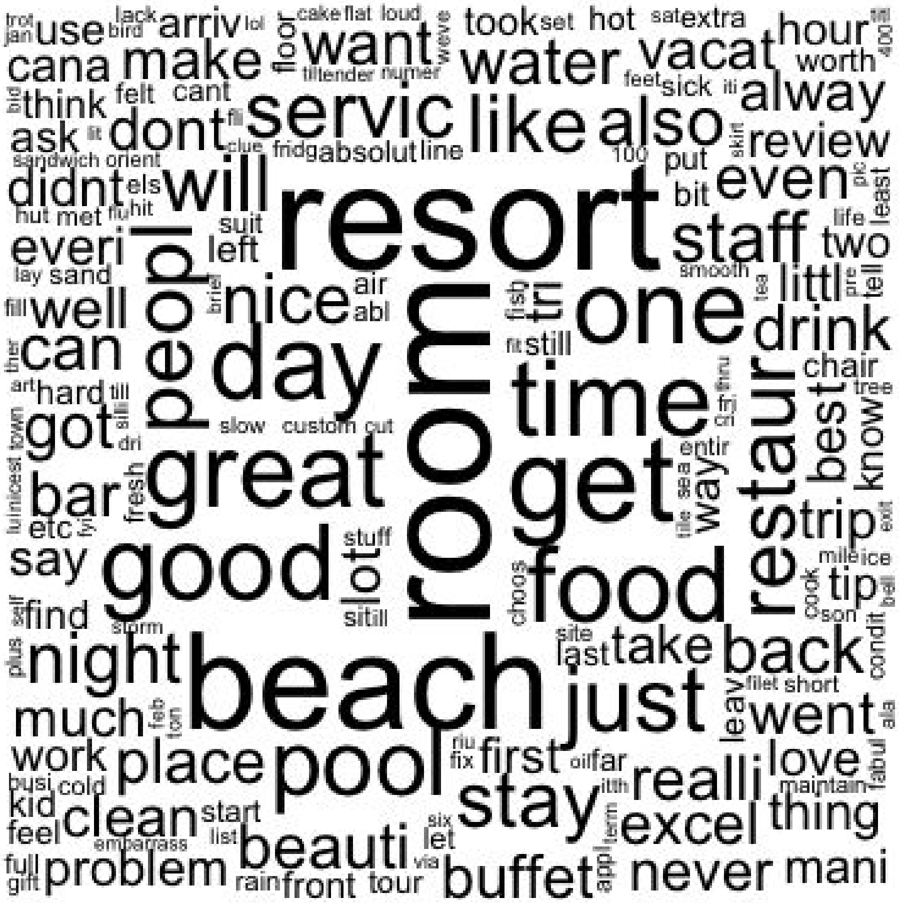


Top 5 word stems:

1. room
2. resort
3. beach
4. get
5. time

Visualization of Results

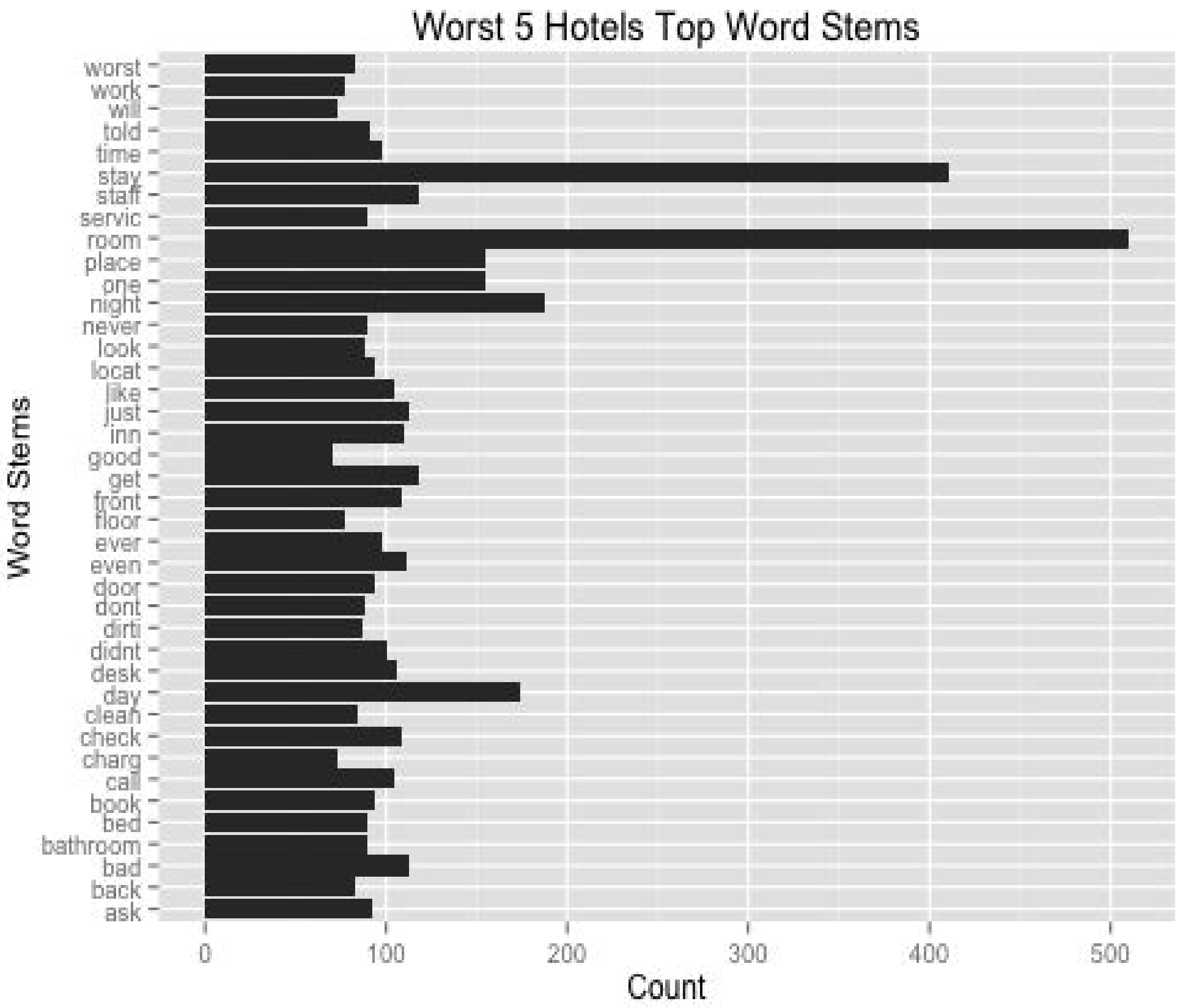
Best 5 Hotels Word Cloud



Words to take notice:

1. trip
2. kids
3. friend
4. clean
5. service, staff
6. fun
7. food, buffet, bar
8. pool
9. kid
10. club

Visualization of Results

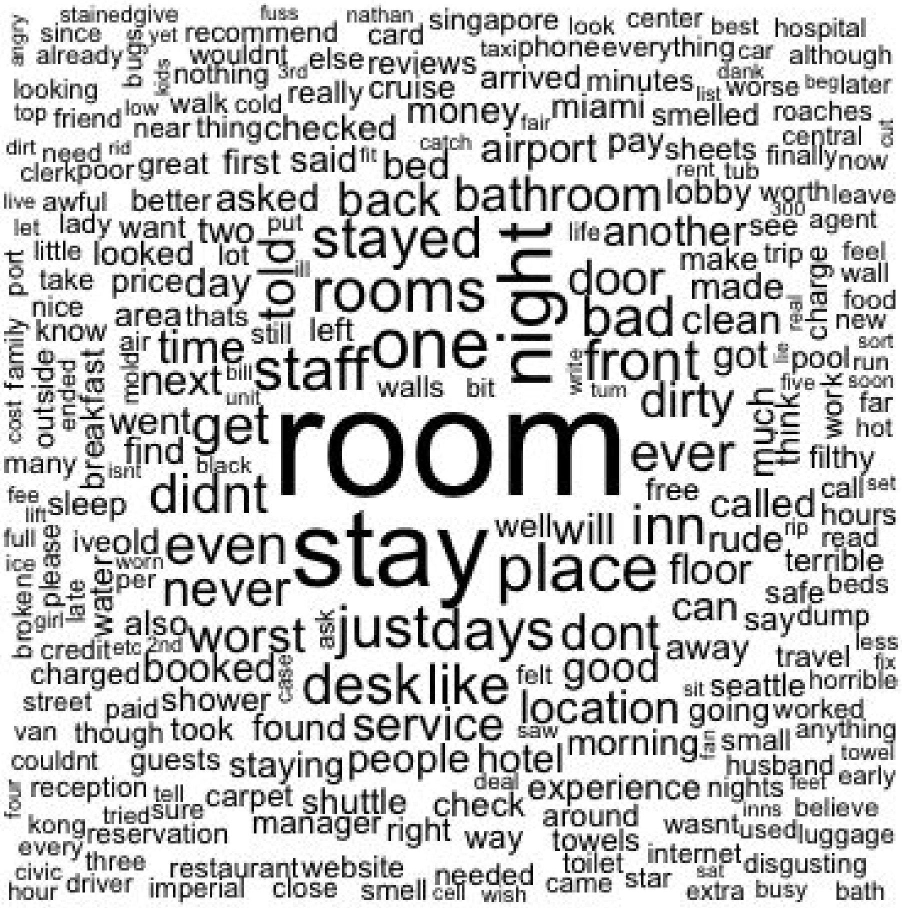


Top 5 words:

1. room
2. stay
3. night
4. day
5. place

Visualization of Results

Worst 5 Hotels Word Cloud



Words to take notice:

1. smelled
2. roaches
3. dirty
4. restaurant
5. driver
6. manager
7. towels, bed
8. breakfast
9. bathroom
10. water

Limitations/Issues

- Inability to find any reliable source of positive/negative sentiment labelled hotel reviews dataset
- Absence of labelled data set, lead to various limitations
  - Supervised machine learning methods which generally have better predictive power could not be used.
  - Advantage of n-gram feature information could not be utilized
  - Advantage of parts of speech feature information could not be utilized
- The hotel reviews don't have any emoticons, hence possible information gain through them couldn't be utilized

Other Approaches Tried

- Twitter tweets is a good proxy for analyzing the sentiment of text data such as hotel reviews.
- Used a corpus of sentiment labelled (positive and negative) dataset from Edinburgh Twitter corpus[1] containing 222,570 tweets.
- Trained a supervised Naive Bayes classifier on the above data set with text transformations.
- Used the resulting model to classify the hotel reviews as positive or negative.
- Calculated a Hotel Score as the difference between the total number of positive reviews and negative reviews for that hotel
- Ranked the hotels as per their scores.
- Limitations:
  - This approach led to the scores being tightly stacked in a narrow range. Hence may be susceptible to noise.
  - As such went ahead with Lexicon based approach.

THANK YOU