HOTEL RATING CLASSIFICATION

Project presented by **Group 4**

MENTOR : SRI VINOD

17/11/2020

01

Group Members

Group 4

AKSHAY JADHAV

ANIKET POUL

PRASANNA BHAGAT

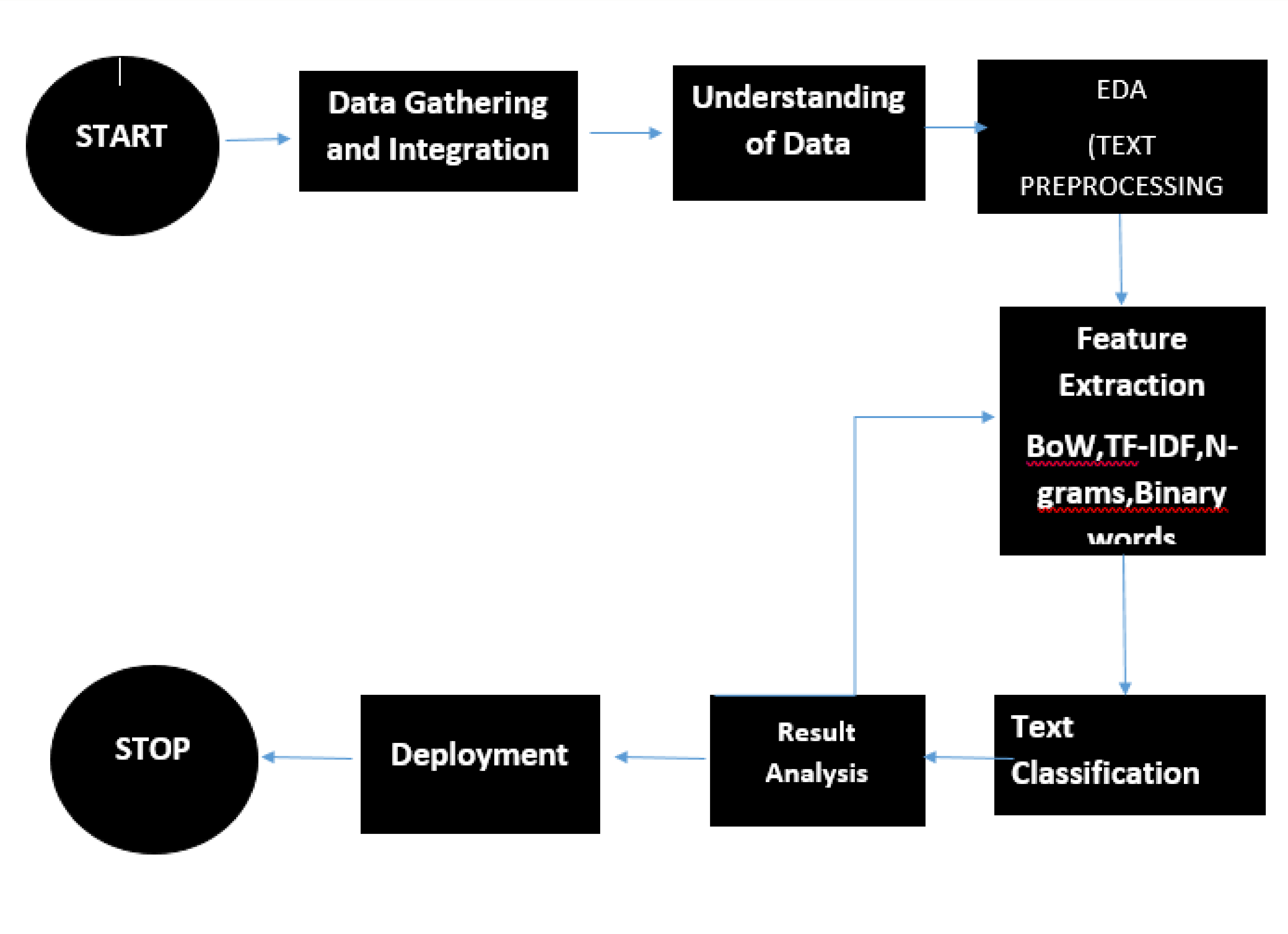
SHASHIKIRAN.M

SRUTHI SANKAR P.M

Business Objective

This is a sample dataset which consists of 20,000reviews and ratings for different hotels and our goal is to examine how travellers are communicating their positive and negative experiences in online platforms for staying in a specific hotel and major objective is what are the attributes that travellers are considering while selecting a hotel. With this manager can understand which elements of their hotel influence more in forming a positive review or improves hotel brand image

Project Architecture / Project Flow



Understanding of Data

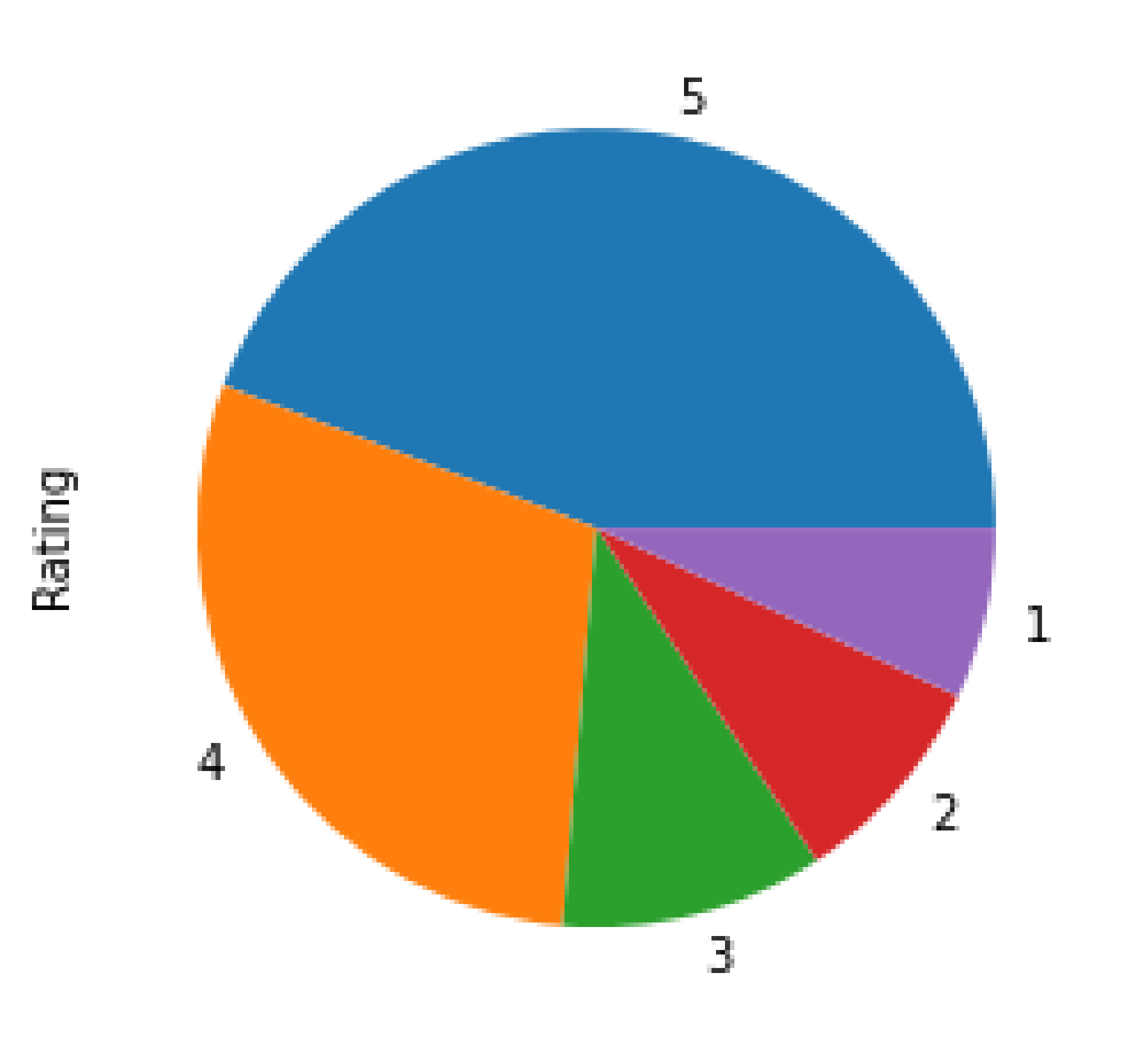
1.Dataset contains 20491 observations and 2 variables(Review and Rating)

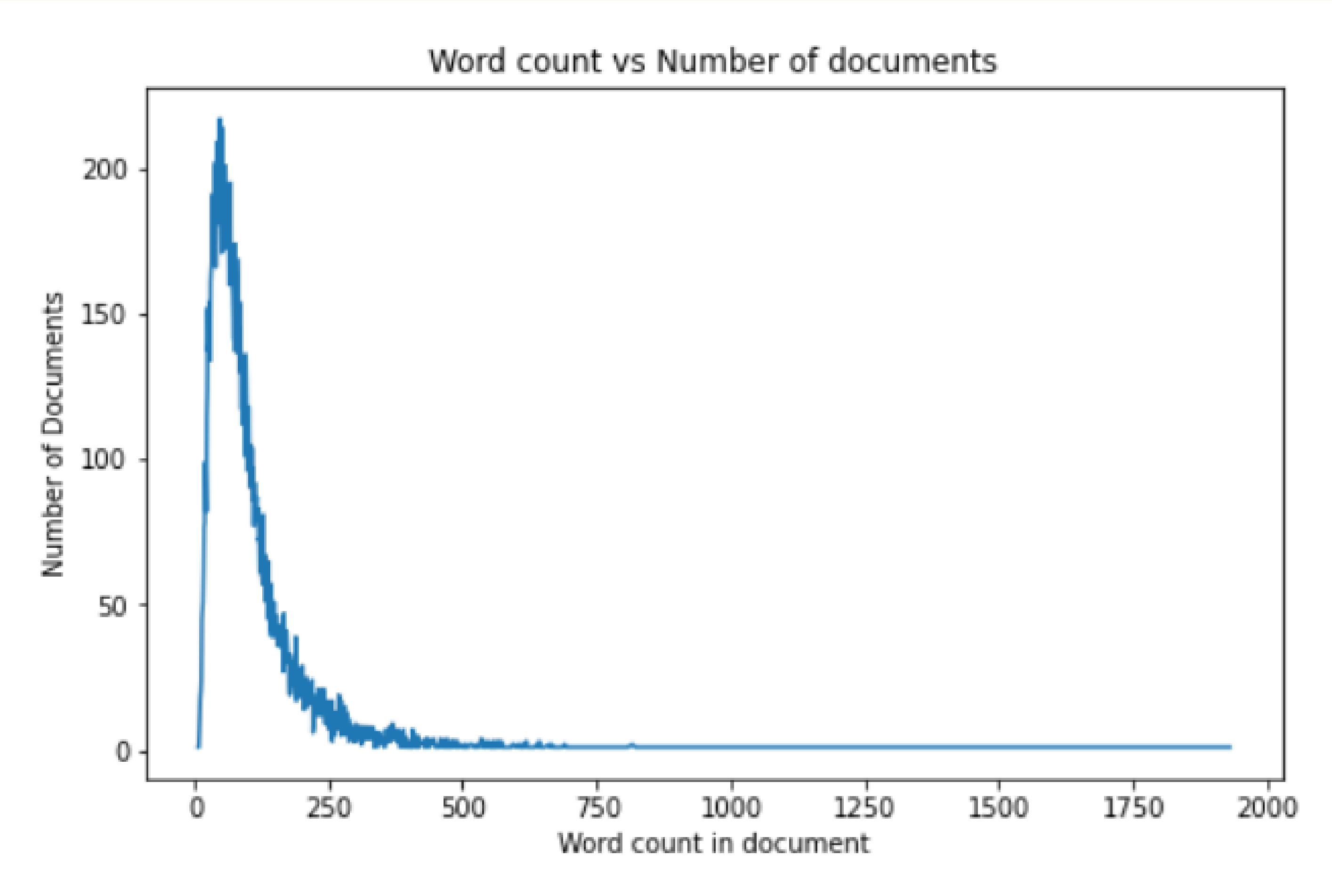
2.Review is object type and Rating is integer type

3.Dataset has no null values

1.Rating column has 5 unique values. Rating 1 to 5

2.This dataset is biased or imbalanced



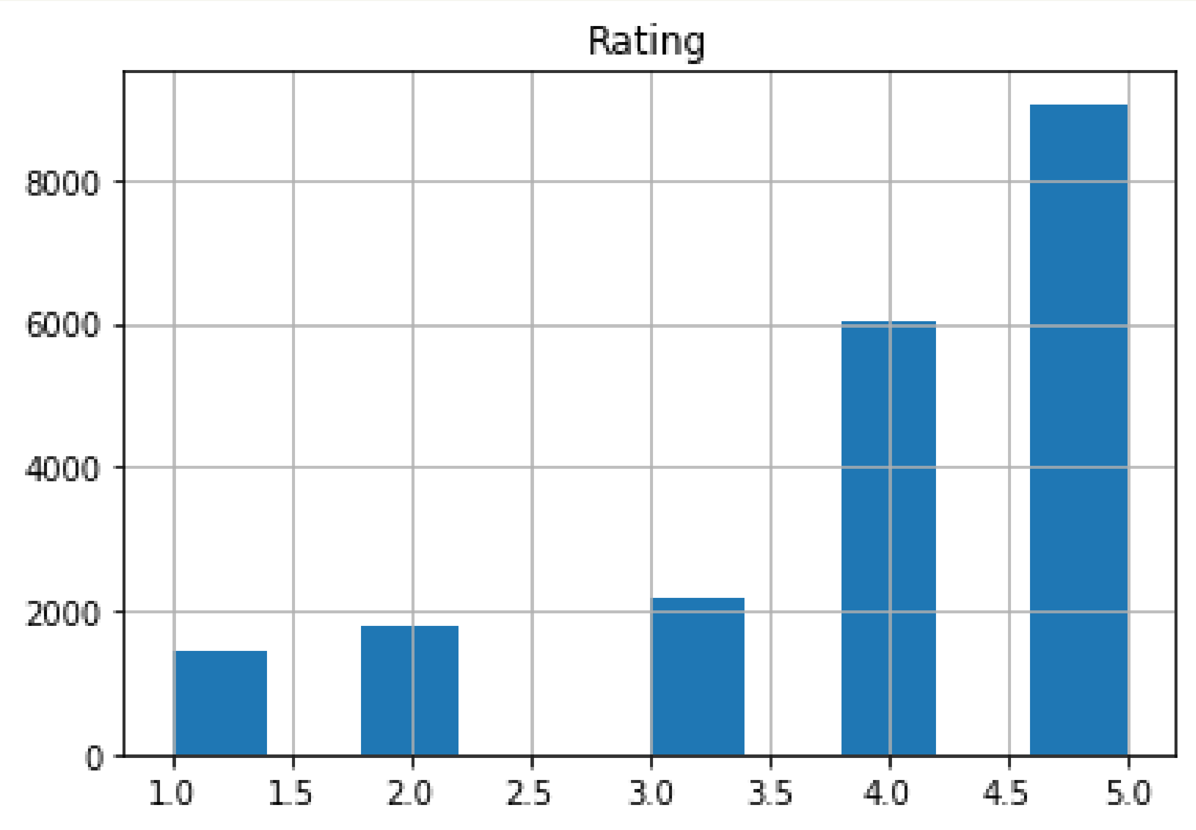


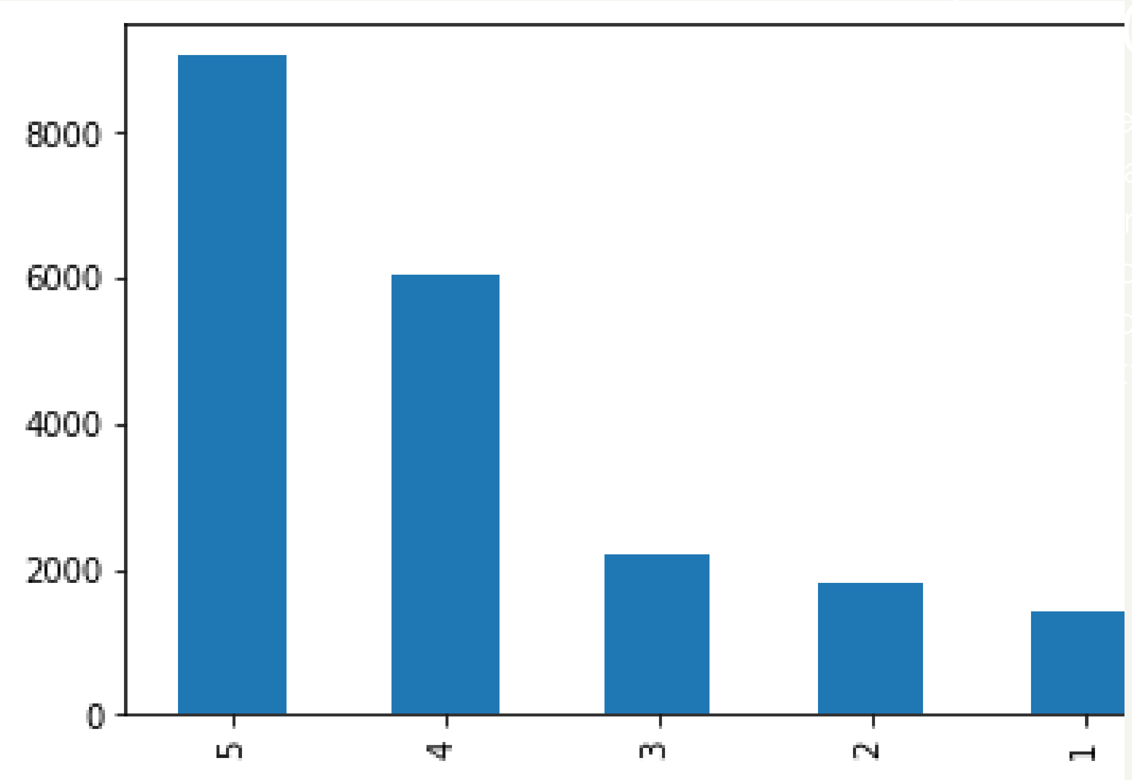
Most of the reviews have wordlength below 150 and a few have more than 250 words

Visualization

Histogram

Barplot





Text Preprocessing

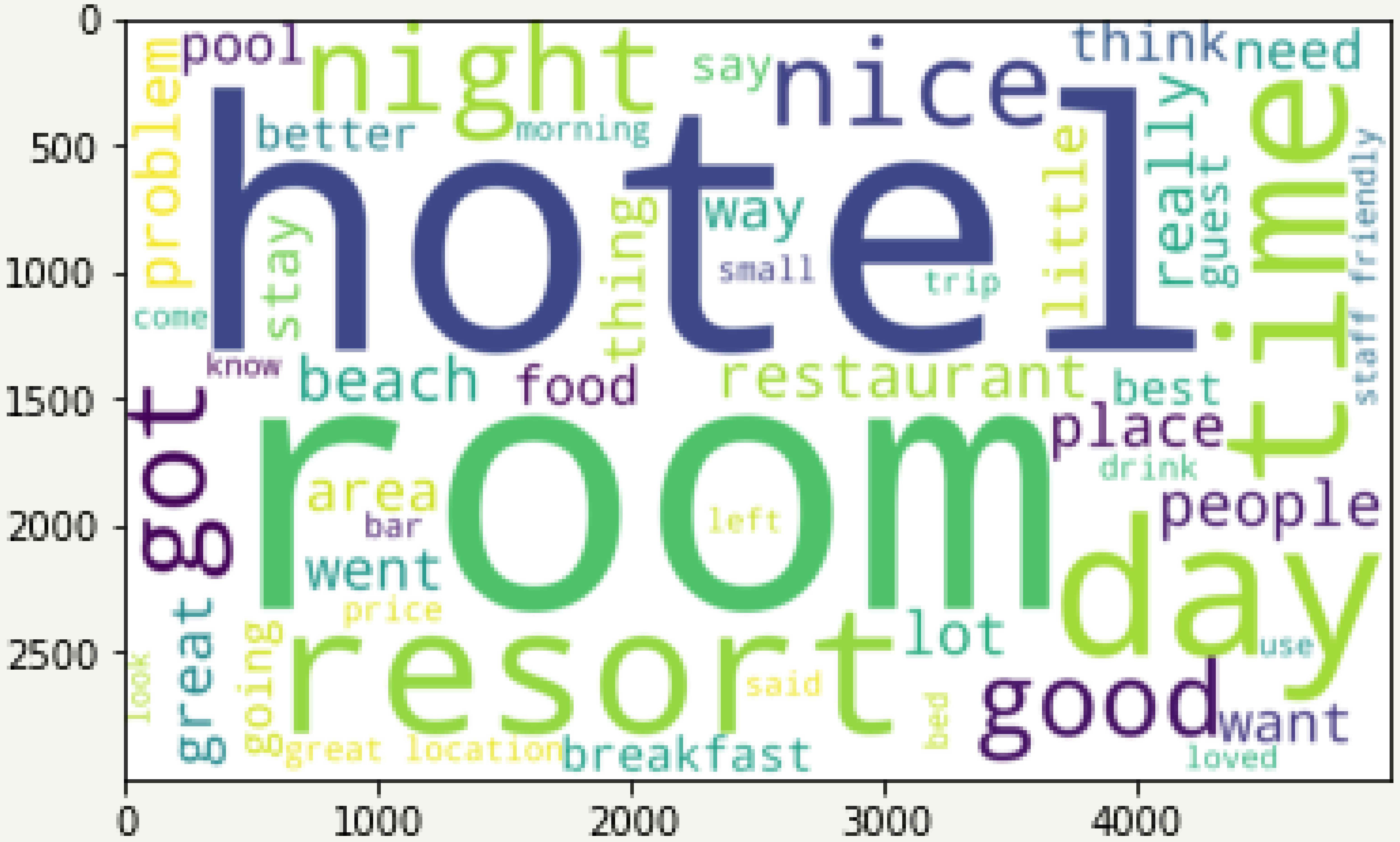
In Text classification, text preprocessing is the first step in the process of building a model.Whenever we have textual data, we need to apply several pre-processing steps to the data to transform words into numerical features that work with machine learning algorithms.We used NLTK library for text preprocessing

The various text preprocessing steps are:

1 - Tokenization
2 -Normalization
3 - Removing Numbers
4 - Removing Punctuations
5 - Stopwords Removal
6- Lemmatization

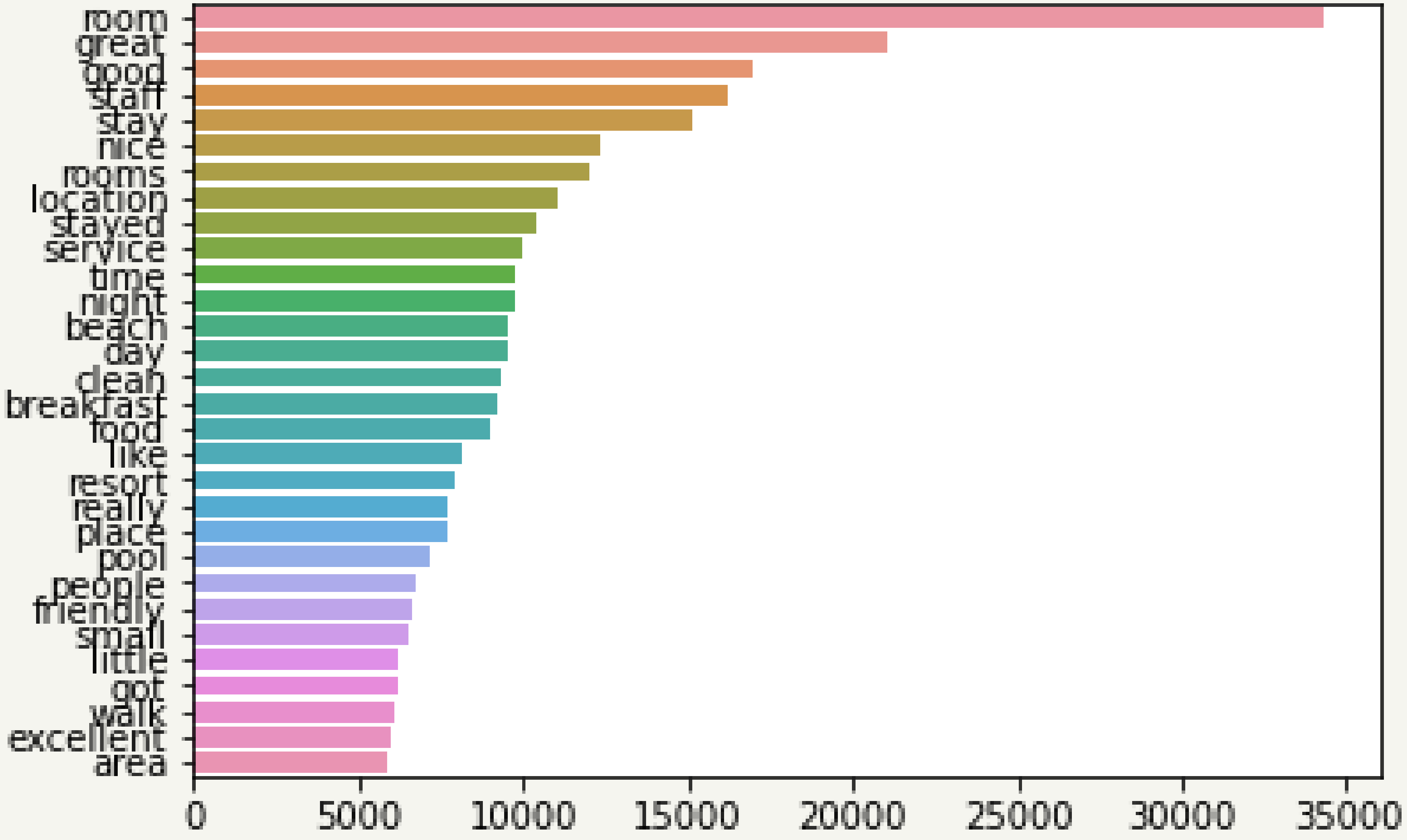
WordCloud

- Visualization Technique in which the size of each word will represent the frequency of that text data.
- From cleaned review data - the most repeated words are 'room','time','day','great','resort','nice','beach','good','night', 'breakfast','restaurant'.



hotel
room
resort
time
day
nice
night
good
got
restaurant
beach
food
place
people
breakfast
thing
little
really
guest
pool
problem
better
morning
say
think
need
way
stay
small
trip
come
know
area
bar
went
left
price
lot
best
drink
want
use
said
going
great
great location
look
bed
loved
staff
friendly
0
500
1000
1500
2000
2500
0
1000
2000
3000
4000

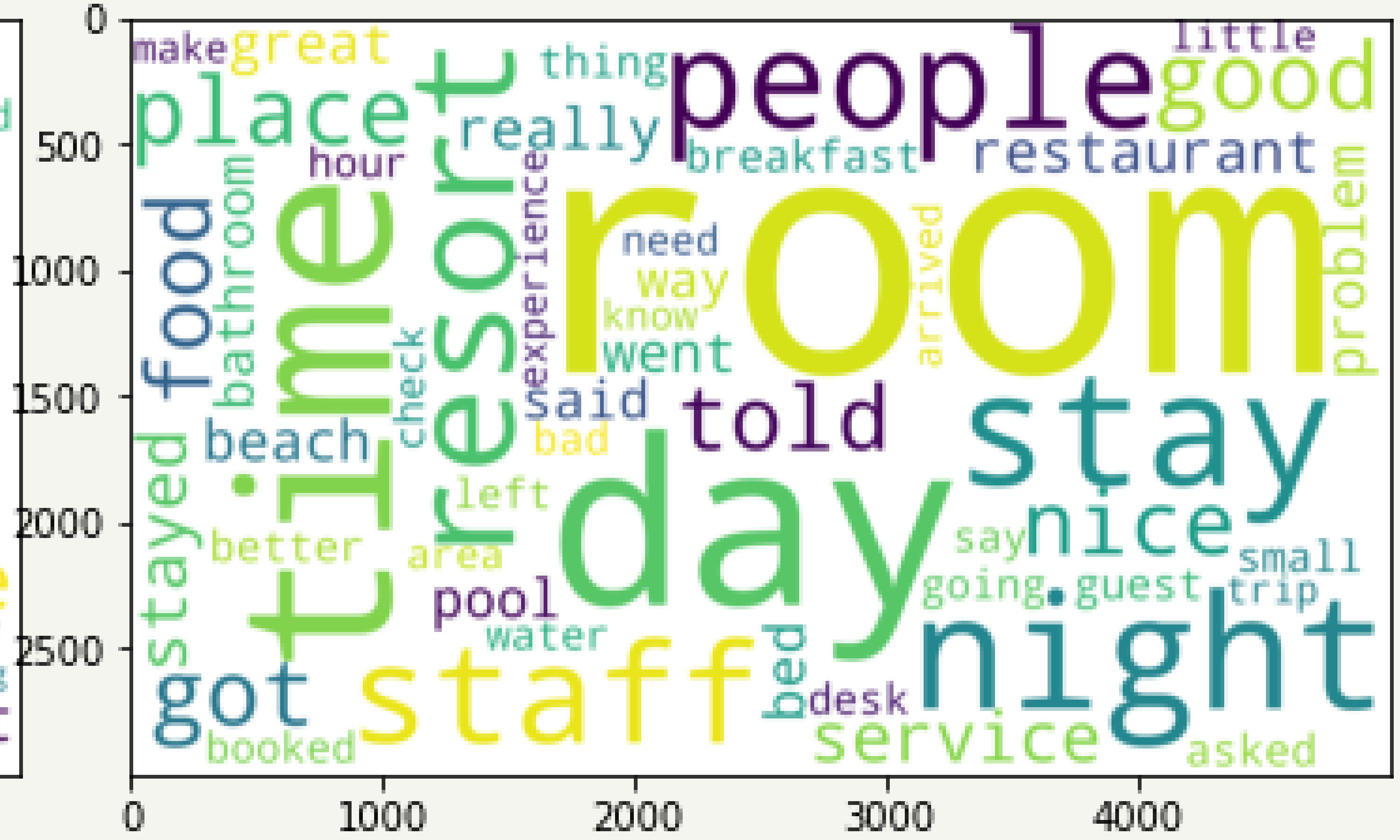
Bar Plot using Counter Function
Most frequent 30 words in reviews



In reviews people are talking more about Room,Staff,Stay,Location,Food, Beach, breakfast,resort etc

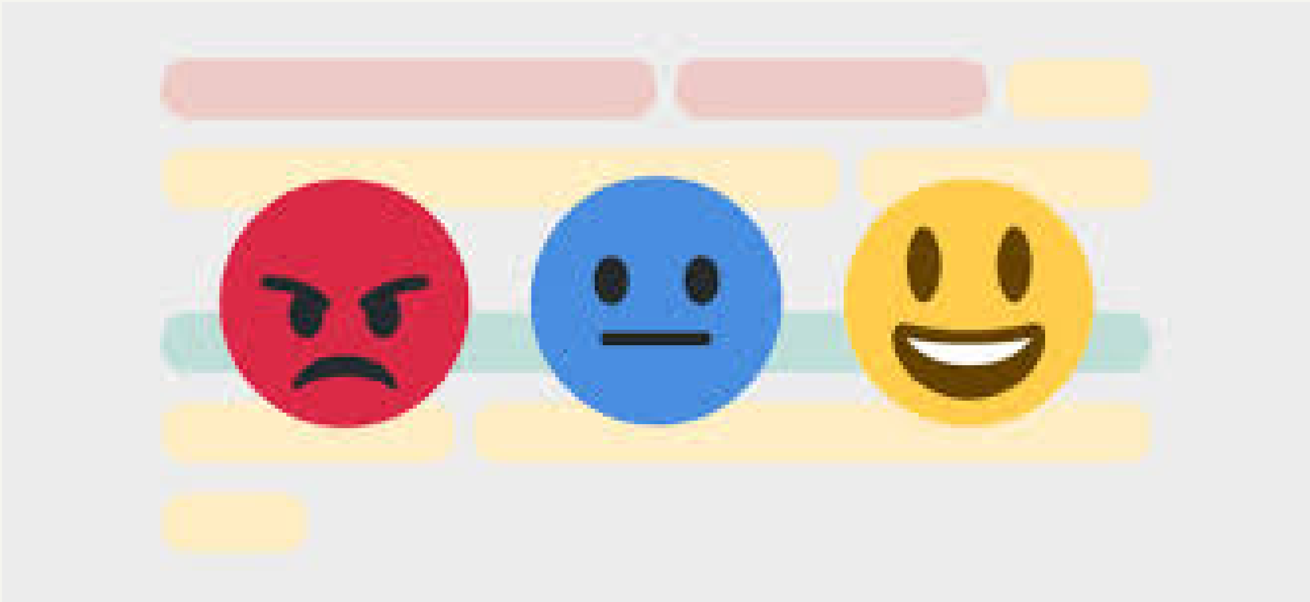
Positve word_cloud based on rating

Negative wordcloud based on rating



From wordcloud based on rating we can say taht there are positive and negative comments on Room,resort,night,stay etc

Sentiment analysis using TextBlob



Sentiment Analysis is the process of determining whether a piece of writing is positive, negative or neutral.

TextBlob gives two scores Polarity scores and Subjectivity scores

Polarity is float which lies in the range of [-1,1] where 1 means positive statement and -1 means a negative statement. subjectivity is a matric between 0 and 1(1 is highly subjective)

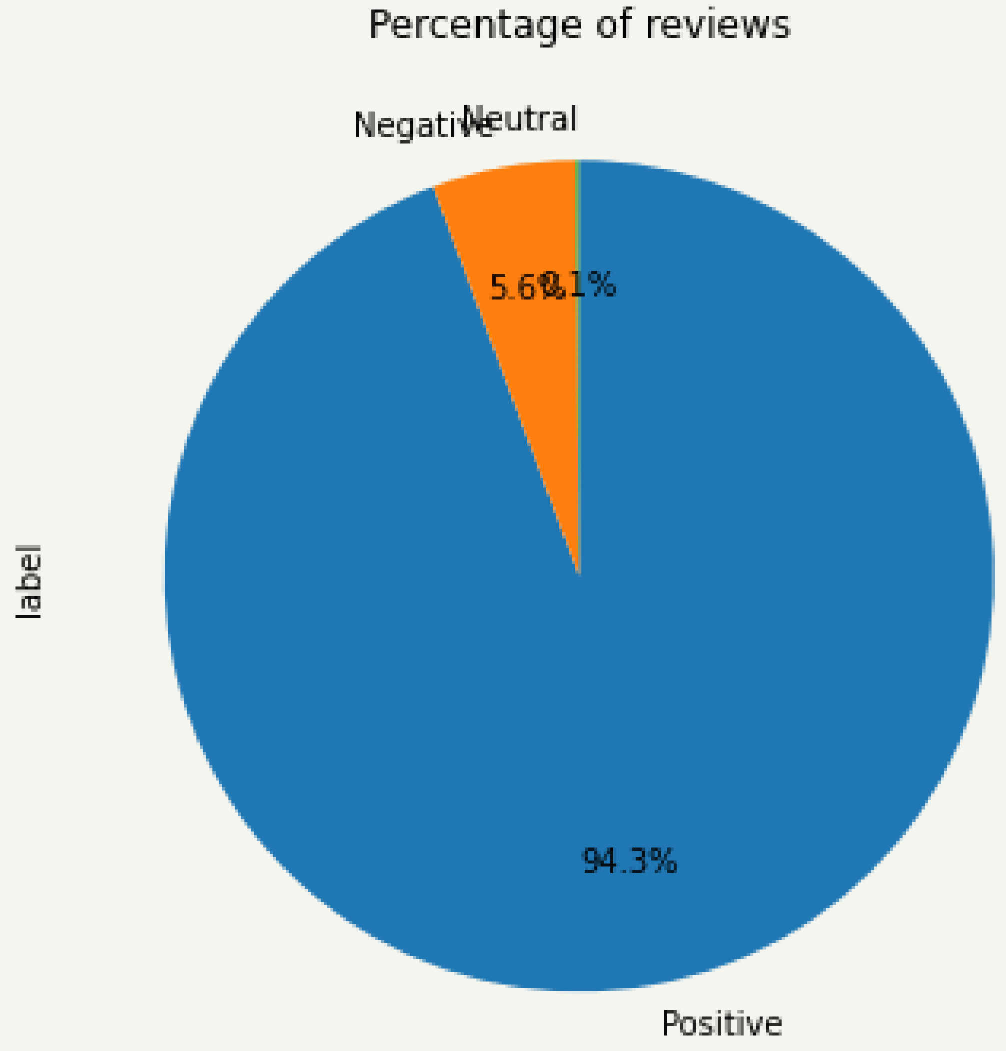
some positive reviews based on polarity score

	Review	Rating	polarity	subjectivity
0	nice expensive parking got good deal stay anni...	4	0.208744	0.687000
1	ok nothing special charge diamond member hilto...	2	0.248633	0.523295
2	nice rooms experience monaco seattle good nt ...	3	0.294420	0.605208
3	unique great stay wonderful time monaco locati...	5	0.504825	0.691228
4	great stay great stay went seahawk game awesom...	5	0.471154	0.629396

some negative reviews baesd on the polarity score

	Review	Rating	polarity	subjectivity
42	warwick bad good reviews warwick shocks staff ...	2	-0.080000	0.633333
44	austin powers decor familiar seattlewhere shee...	2	-0.043056	0.533333
65	hated inn terrible roomservice horrible staff ...	1	-0.633333	0.725000
76	stay clear internet reservation friday rang ho...	1	-0.142857	0.547619
77	single rooms like hospital rooms single rooms ...	1	-0.164947	0.330026

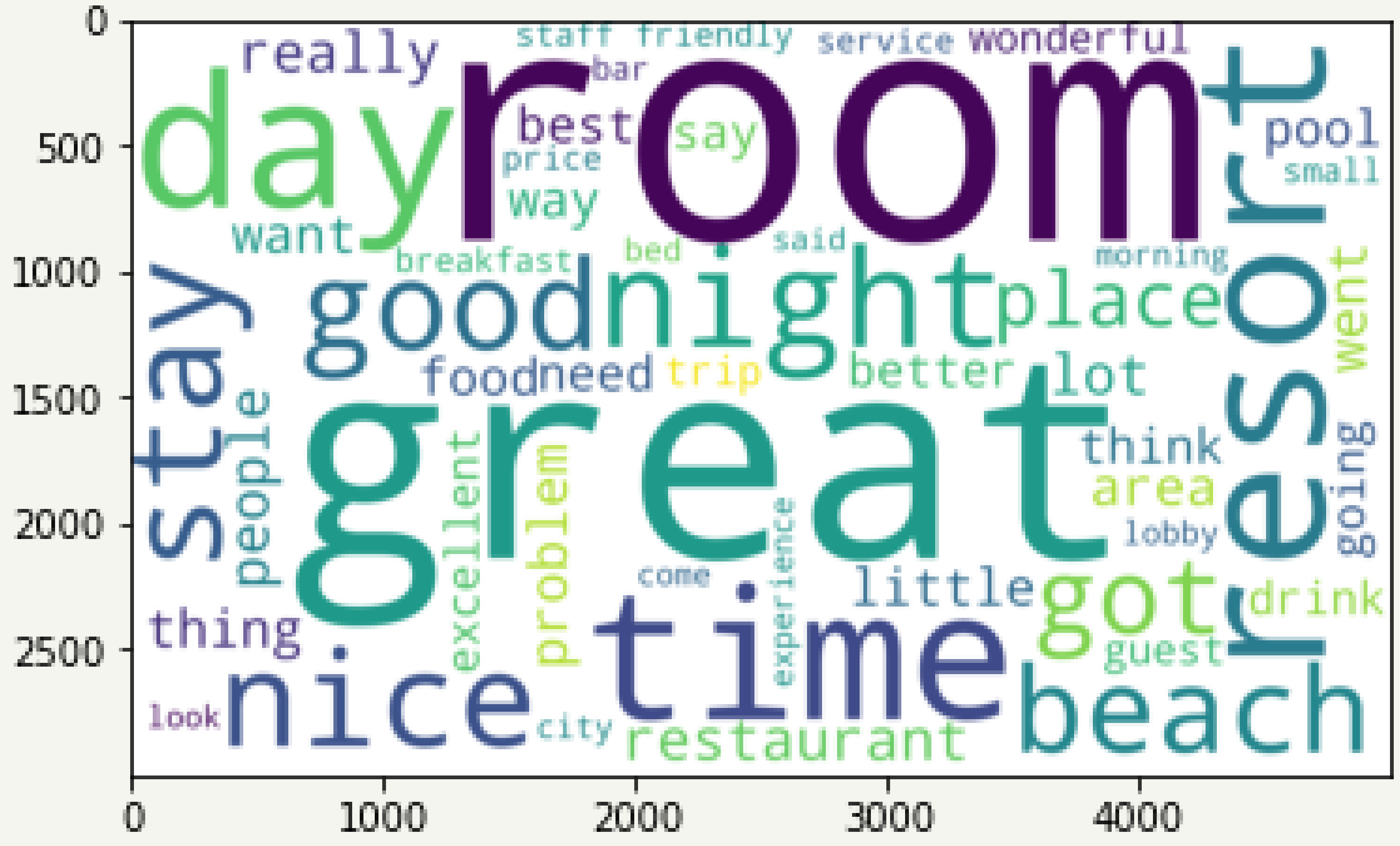
Pie-Chart
percentage of Sentiments

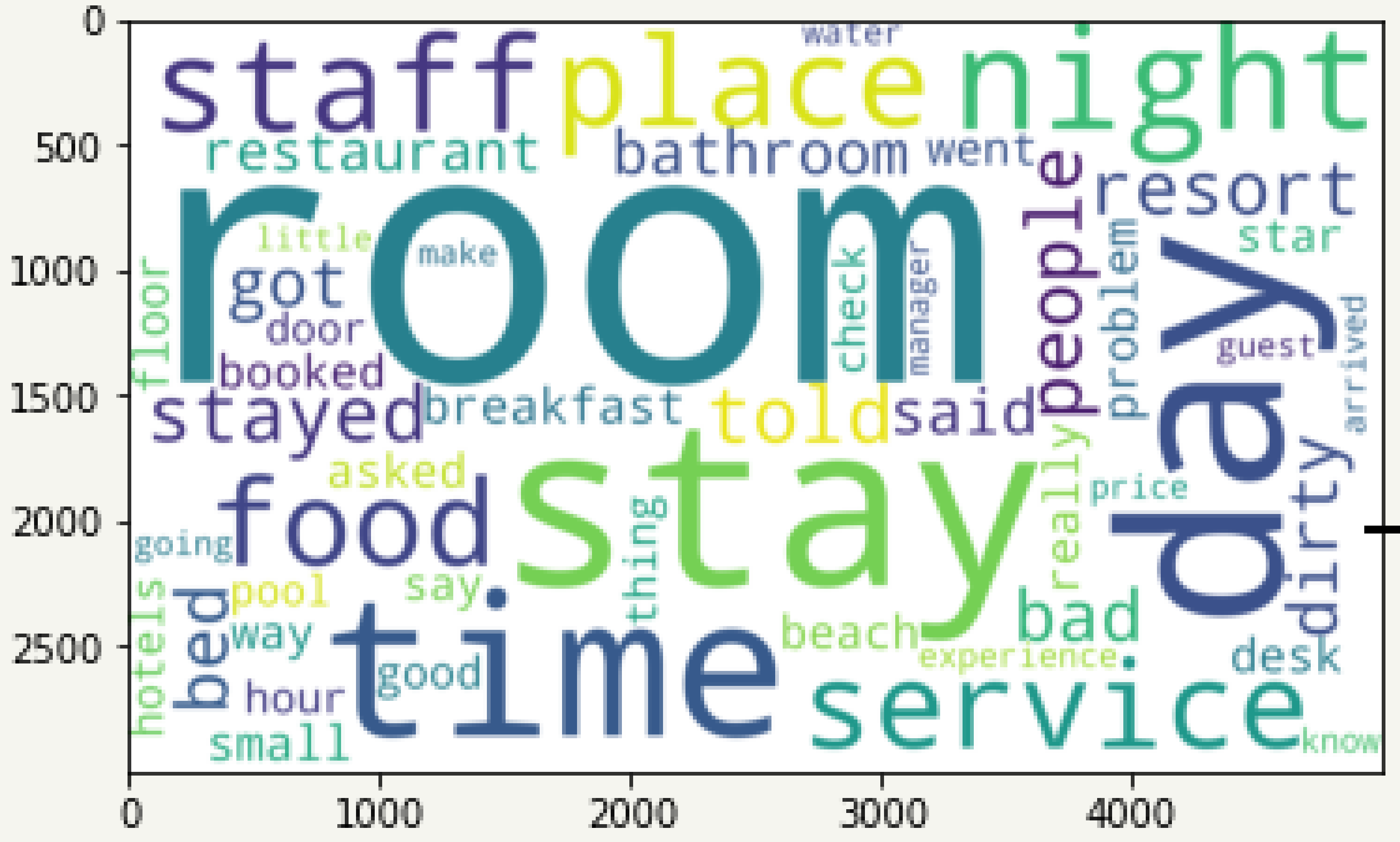


94.3%reviews are positive based on the polarity score.
So data is highly biased or imbalnced

Wordcloud _positive reviews
Based on polarity

Wordcloud_Negative Reviews
Based on polarity

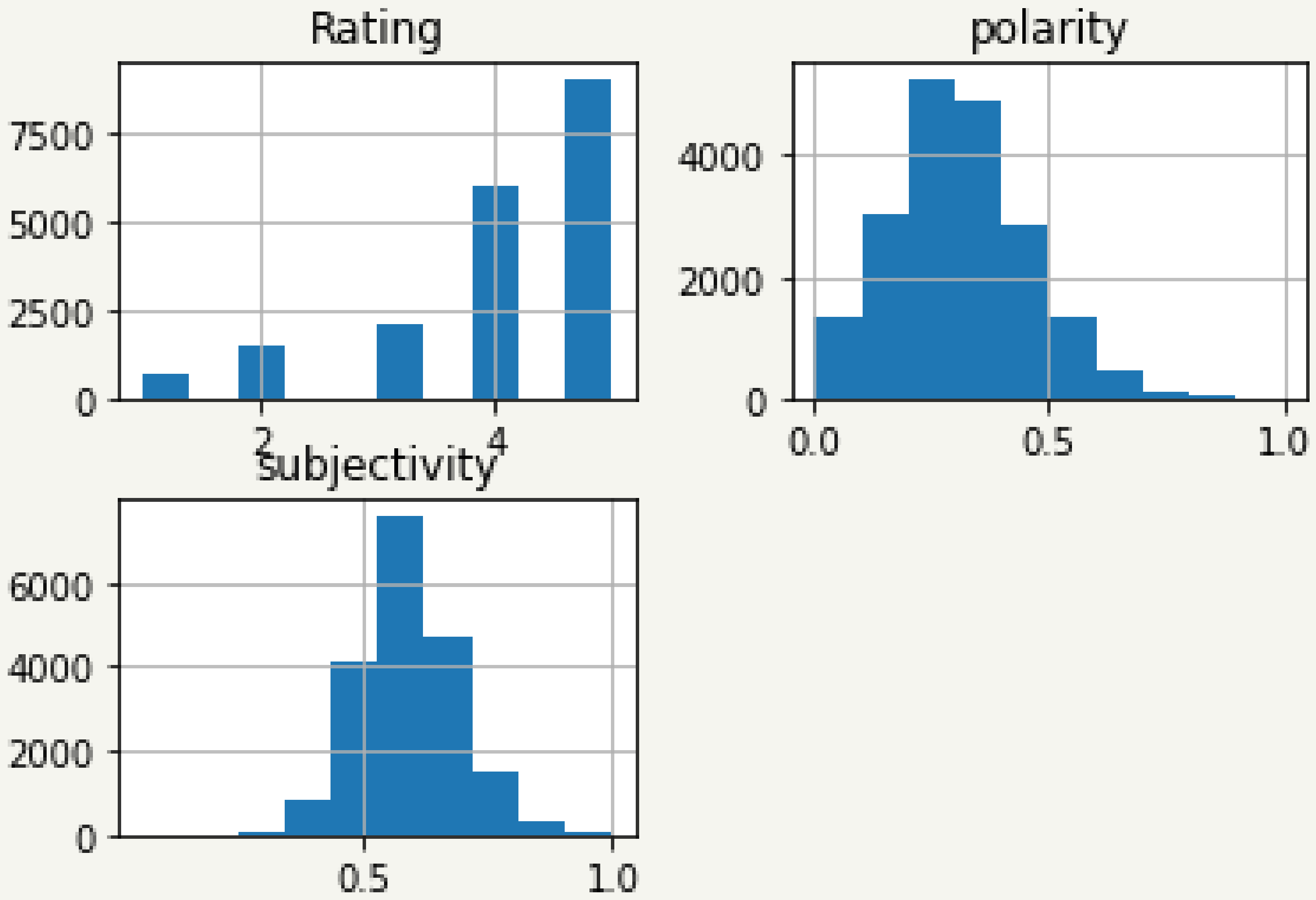




There are some positive comments on room,beach,stay,restaurant,food

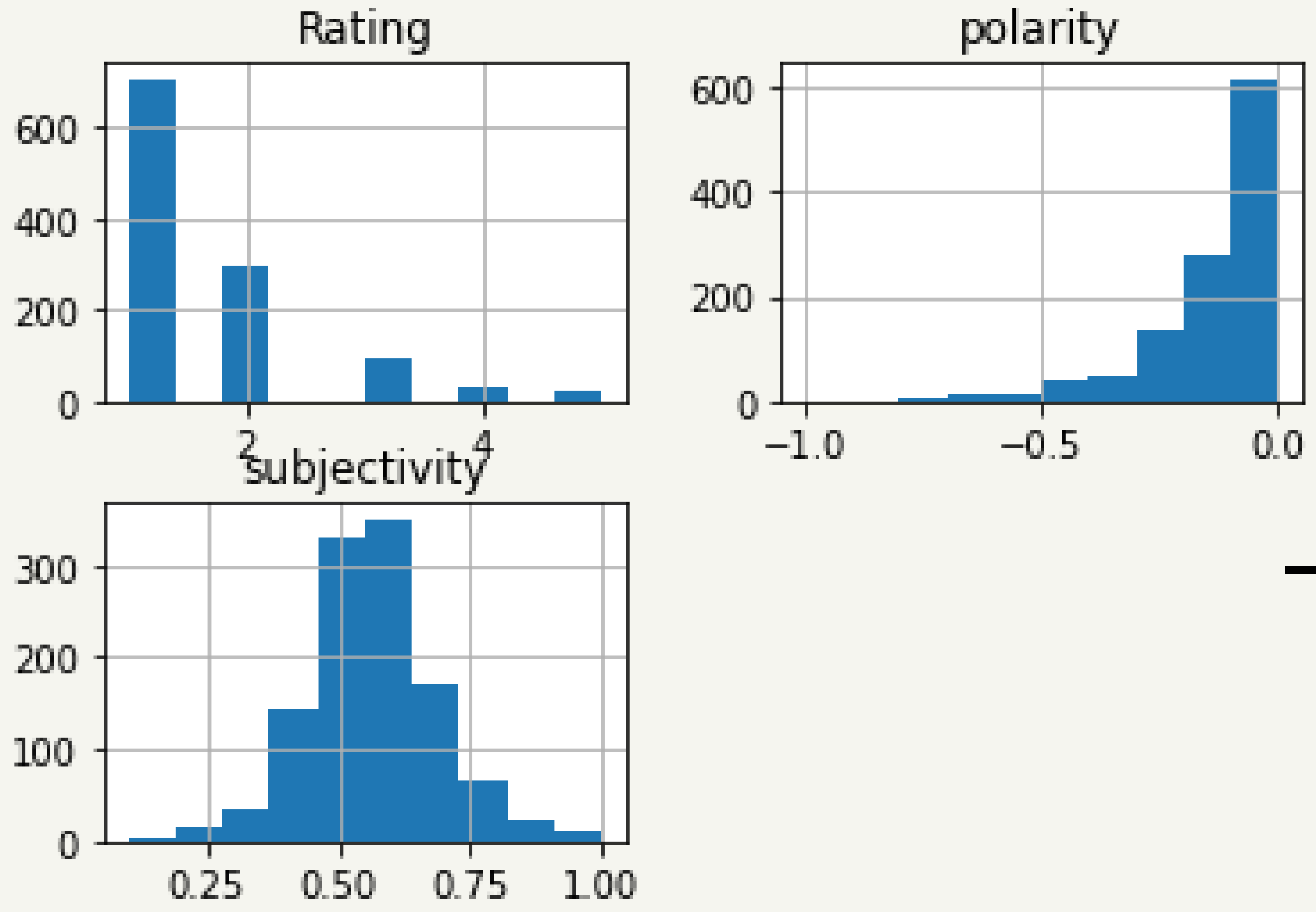
There are some negative comments on room,stay,staff,place,night,service,food, bathroom,small,dirty, restaurant,manager

Postive reviews -Histogram



polarity is right skewed

Negative reviews_Histogram



Polarity left skewed

positive review Bigrams

```
[('nice', 'hotel'),
 ('hotel', 'expensive'),
 ('expensive', 'parking'),
 ('parking', 'got'),
 ('got', 'good'),
 ('good', 'deal'),
 ('deal', 'stay'),
 ('stay', 'hotel'),
 ('hotel', 'anniversary'),
 ('anniversary', 'arrived'),
 ('arrived', 'late'),
 ('late', 'evening'),
 ('evening', 'took'),
 ('took', 'advice'),
 ('advice', 'previous'),
 ('previous', 'reviews'),
 ('reviews', 'valet'),
 ('valet', 'parking'),
 ('parking', 'check'),
 ('check', 'quick'),
 ('quick', 'easy'),
 ('easy', 'little'),
 ('little', 'disappointed'),
 ('disappointed', 'nonexistent'),
 ('nonexistent', 'view'),
 ('view', 'room'),
 ('room', 'room'),
 ('room', 'clean'),
 ('clean', 'nice'),
 ('nice', 'size'),
```

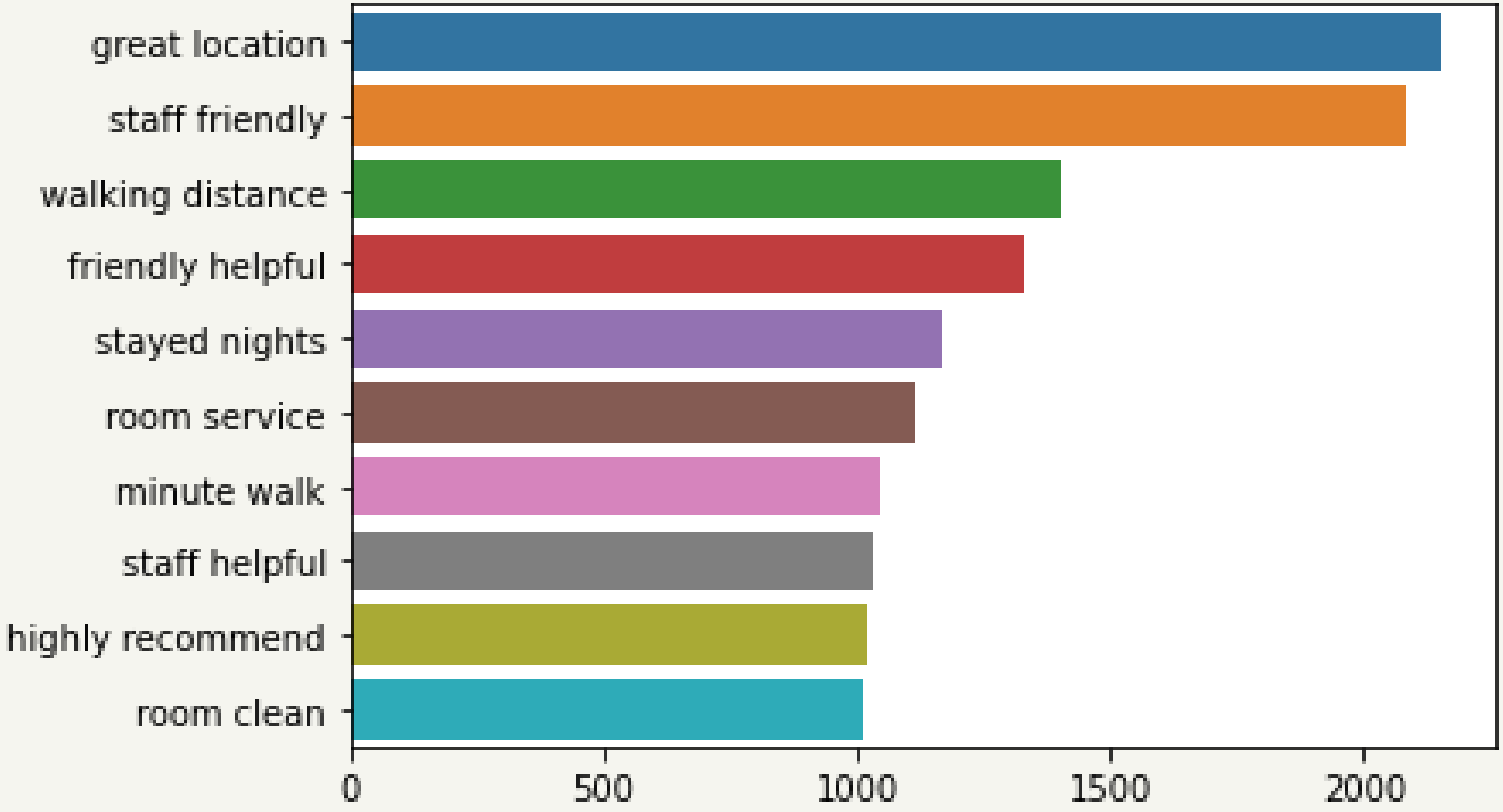
Top 15
most frequent Bigrams

```
[(('staff', 'friendly'), 2043),
 (('punta', 'cana'), 1550),
 (('great', 'location'), 1494),
 (('hotel', 'great'), 1435),
 (('walking', 'distance'), 1387),
 (('friendly', 'helpful'), 1307),
 (('hotel', 'staff'), 1161),
 (('stayed', 'hotel'), 1128),
 (('room', 'service'), 1102),
 (('recommend', 'hotel'), 1048),
 (('minute', 'walk'), 1044),
 (('staff', 'helpful'), 1006),
 (('room', 'clean'), 998),
 (('stayed', 'nights'), 962),
 (('highly', 'recommend'), 915)]
```

Here wecan see that most frequent positive comments are

- Friendly and helpful Staff
- Great Location
- Room Service
- Room Clean
- Highly recommend

Bar Plot using Counter Function
Most frequent Positive bigrams



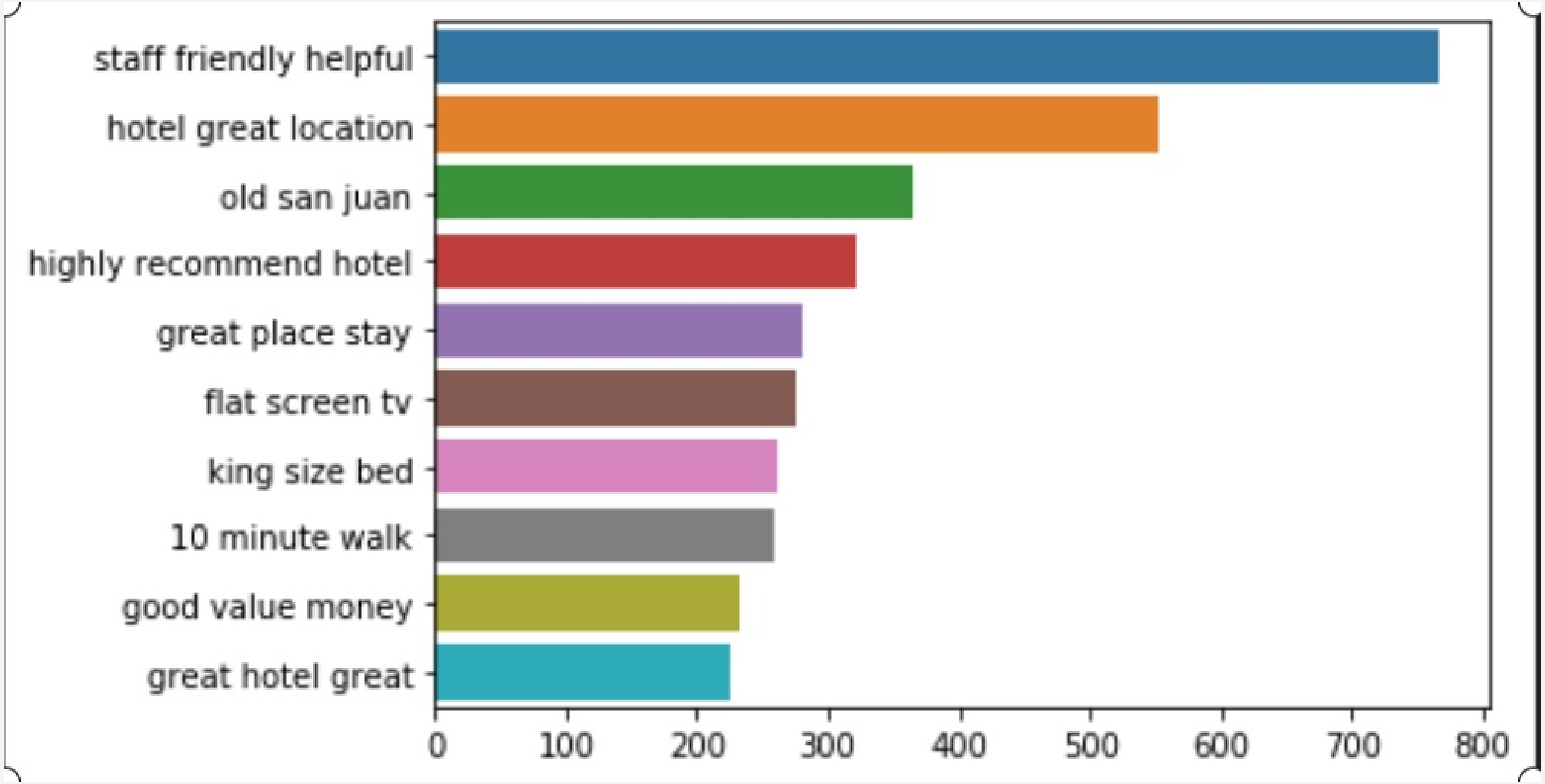
Polarity based
Positive review Trigrams

```
[('nice', 'hotel', 'expensive'),
 ('hotel', 'expensive', 'parking'),
 ('expensive', 'parking', 'got'),
 ('parking', 'got', 'good'),
 ('got', 'good', 'deal'),
 ('good', 'deal', 'stay'),
 ('deal', 'stay', 'hotel'),
 ('stay', 'hotel', 'anniversary'),
 ('hotel', 'anniversary', 'arrived'),
 ('anniversary', 'arrived', 'late'),
 ('arrived', 'late', 'evening'),
 ('late', 'evening', 'took'),
 ('evening', 'took', 'advice'),
 ('took', 'advice', 'previous'),
 ('advice', 'previous', 'reviews'),
 ('previous', 'reviews', 'valet'),
 ('reviews', 'valet', 'parking'),
 ('valet', 'parking', 'check'),
 ('parking', 'check', 'quick'),
 ('check', 'quick', 'easy'),
 ('quick', 'easy', 'little'),
 ('easy', 'little', 'disappointed'),
 ('little', 'disappointed', 'nonexistent'),
 ('disappointed', 'nonexistent', 'view'),
 ('nonexistent', 'view', 'room'),
 ('view', 'room', 'room'),
 ('room', 'room', 'clean'),
 ('room', 'clean', 'nice'),
 ('clean', 'nice', 'size'),
 ('nice', 'size', 'bed'),
```

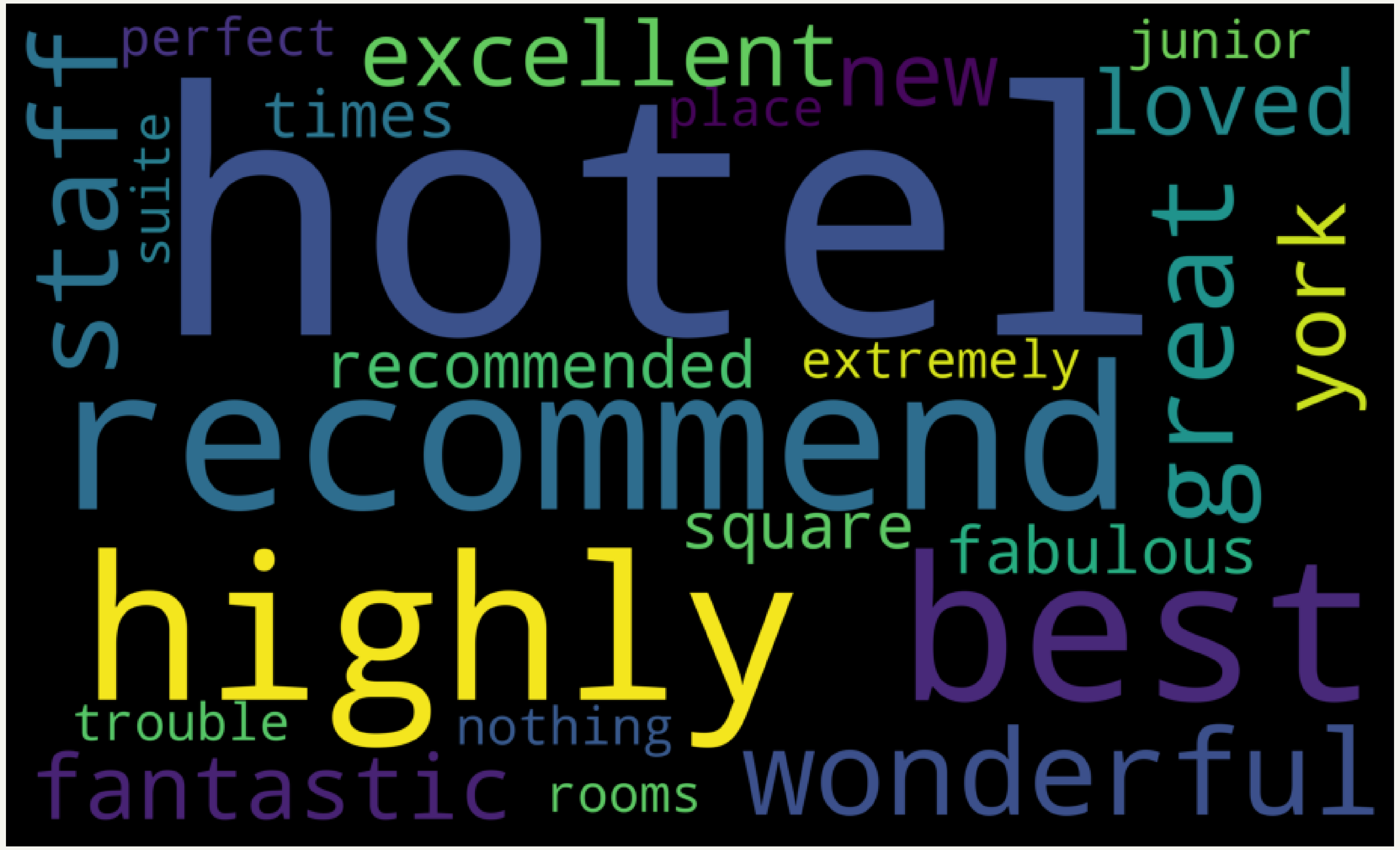
Most frequent Trigramst

```
[(('staff', 'friendly', 'helpful'), 687),
 (('hotel', 'great', 'location'), 528),
 (('old', 'san', 'juan'), 341),
 (('highly', 'recommend', 'hotel'), 245),
 (('flat', 'screen', 'tv'), 227),
 (('king', 'size', 'bed'), 223),
 (('stayed', 'hotel', 'nights'), 202),
 (('hotel', 'staff', 'friendly'), 200),
 (('easy', 'walking', 'distance'), 186),
 (('free', 'internet', 'access'), 183),
 (('hotel', 'good', 'location'), 172),
 (('la', 'carte', 'restaurants'), 165),
 (('staff', 'helpful', 'friendly'), 157),
 (('returned', 'night', 'stay'), 157),
 (('good', 'value', 'money'), 153)]
```

Bar Plot using Counter Function
Most frequent Positive trigrams



Positive Bigram_Wordcloud



Positive trigram WordCloud

Positive features/Reviews

Top positive features/reviews related to hotel on the basis of N-grams and wordcloud are

- staff helpful, friendly and efficient
- Great Location
- Room clean and nice
- Highly recommend
- Flat screen Tv
- Free Wifi and wonderful server
- free and best breakfast
- wonderful stay
- Free wine,tea and coffee service
- Bathroom attractive ,large with great soaking tub
- Car service price is reasonable
- Huge open space
- good lighting
- large and good building

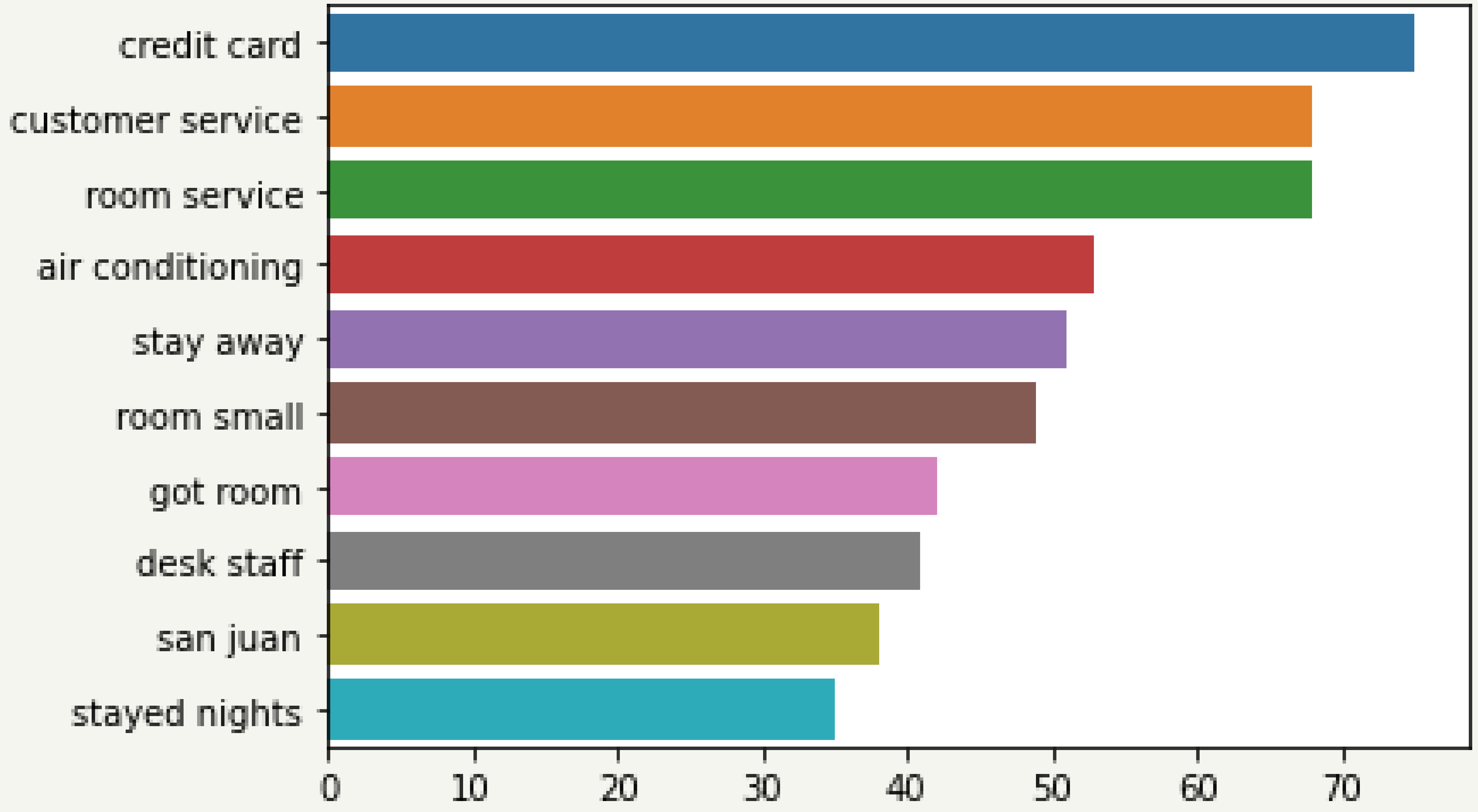
Polarity based Negative Bigrams

```
[('warwick', 'bad'),
 ('bad', 'good'),
 ('good', 'reviews'),
 ('reviews', 'warwick'),
 ('warwick', 'shocks'),
 ('shocks', 'staff'),
 ('staff', 'quite'),
 ('quite', 'rude'),
 ('rude', 'rooms'),
 ('rooms', 'fairly'),
 ('fairly', 'dirty'),
 ('dirty', 'cut'),
 ('cut', 'asked'),
 ('asked', 'bandaid'),
 ('bandaid', 'requested'),
 ('requested', 'bottle'),
 ('bottle', 'opener'),
 ('opener', 'better'),
 ('better', 'serviceaustin'),
 ('serviceaustin', 'powers'),
 ('powers', 'decor'),
 ('decor', 'familiar'),
 ('familiar', 'hotel'),
 ('hotel', 'seattlewhere'),
```

Most Frequent negative bigrams

```
[(('star', 'hotel'), 86),
 (('punta', 'cana'), 83),
 (('credit', 'card'), 75),
 (('room', 'service'), 68),
 (('customer', 'service'), 67),
 (('hotel', 'room'), 63),
 (('stay', 'hotel'), 62),
 (('stayed', 'hotel'), 56),
 (('air', 'conditioning'), 52),
 (('room', 'small'), 48),
 (('hotel', 'stayed'), 46),
 (('worst', 'hotel'), 46),
 (('hotel', 'staff'), 43),
 (('got', 'room'), 42),
 (('desk', 'staff'), 41)]
```

Bar Plot using Counter Function
Most frequent Negative bigrams



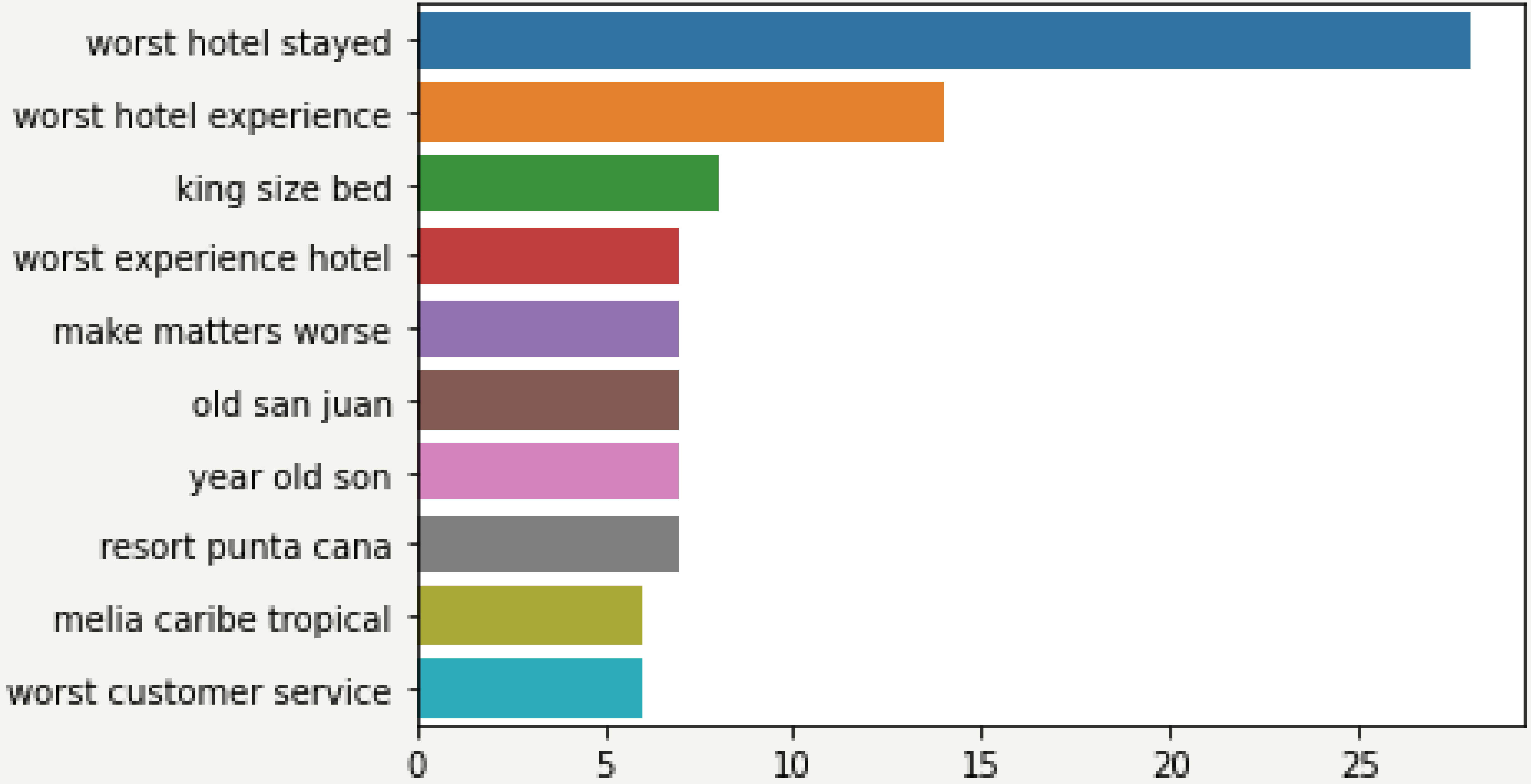
Polarity based
Negative review trigrams

```
[('warwick', 'bad', 'good'),
 ('bad', 'good', 'reviews'),
 ('good', 'reviews', 'warwick'),
 ('reviews', 'warwick', 'shocks'),
 ('warwick', 'shocks', 'staff'),
 ('shocks', 'staff', 'quite'),
 ('staff', 'quite', 'rude'),
 ('quite', 'rude', 'rooms'),
 ('rude', 'rooms', 'fairly'),
 ('rooms', 'fairly', 'dirty'),
 ('fairly', 'dirty', 'cut'),
 ('dirty', 'cut', 'asked'),
 ('cut', 'asked', 'bandaid'),
 ('asked', 'bandaid', 'requested'),
 ('bandaid', 'requested', 'bottle'),
 ('requested', 'bottle', 'opener'),
 ('bottle', 'opener', 'better'),
 ('opener', 'better', 'serviceaustin'),
 ('better', 'serviceaustin', 'powers'),
 ('serviceaustin', 'powers', 'decor'),
 ('powers', 'decor', 'familiar'),
 ('decor', 'familiar', 'hotel'),
 ('familiar', 'hotel', 'seattlewhere'),
```

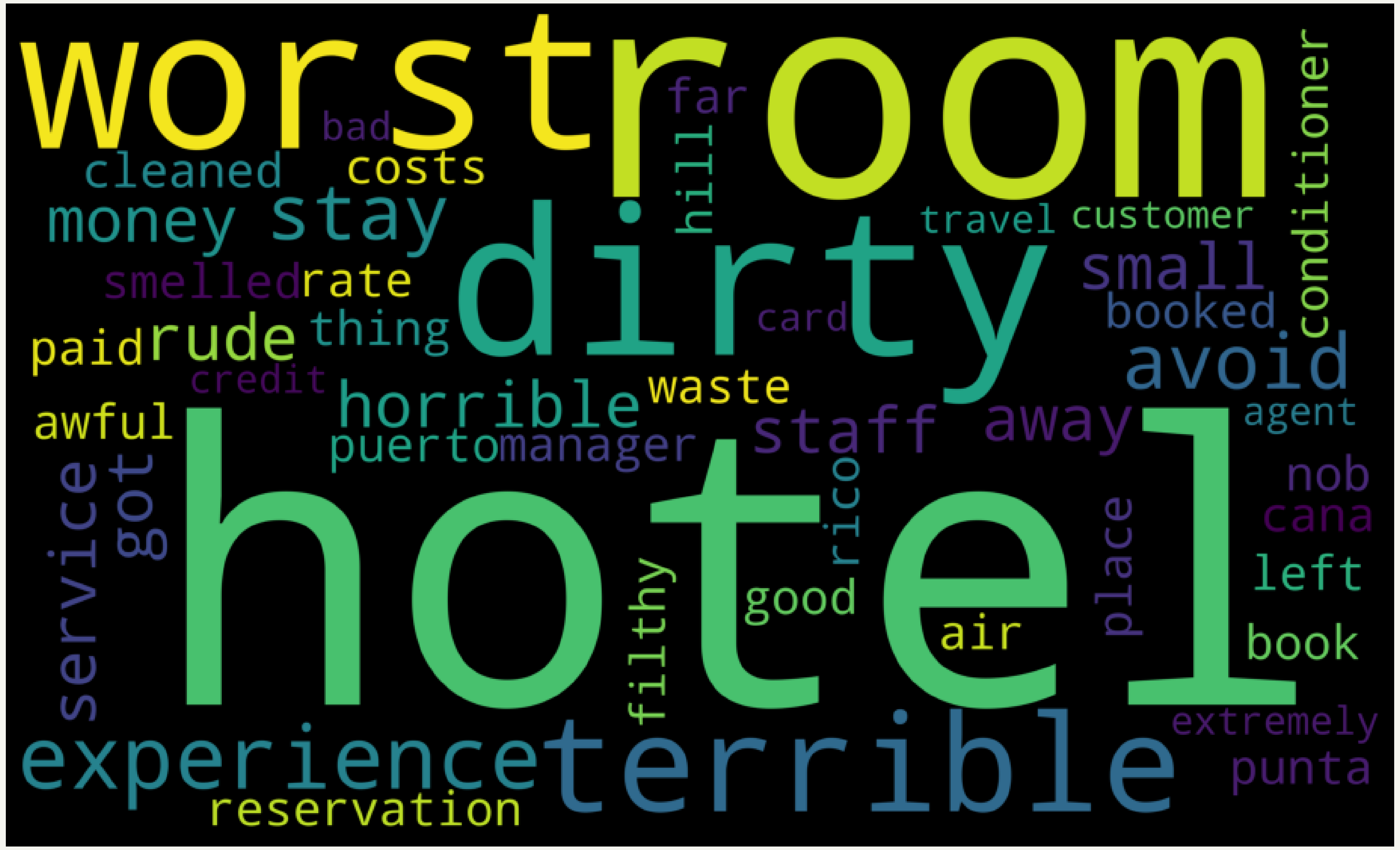
Most frequent
Negative trigrams

```
[(('worst', 'hotel', 'stayed'), 21),
 (('king', 'size', 'bed'), 8),
 (('worst', 'hotel', 'experience'), 7),
 (('make', 'matters', 'worse'), 7),
 (('old', 'san', 'juan'), 7),
 (('year', 'old', 'son'), 7),
 (('resort', 'punta', 'cana'), 7),
 (('melia', 'caribe', 'tropical'), 6),
 (('long', 'story', 'short'), 6),
 (('stayed', 'hotel', 'nights'), 6),
 (('far', 'worst', 'hotel'), 6),
 (('husband', 'stayed', 'hotel'), 5),
 (('charges', 'credit', 'card'), 5),
 (('needless', 'say', 'sleep'), 5),
 (('better', 'places', 'stay'), 5),
 (('room', 'called', 'desk'), 5),
 (('disturb', 'sign', 'door'), 5),
 (('desk', 'staff', 'rude'), 5),
 (('called', 'desk', 'told'), 5),
 (('water', 'pressure', 'shower'), 5),
 (('hotel', 'great', 'location'), 5),
 (('la', 'carte', 'restaurants'), 5),
 (('spoke', 'little', 'english'), 5),
 (('credit', 'card', 'details'), 5),
```

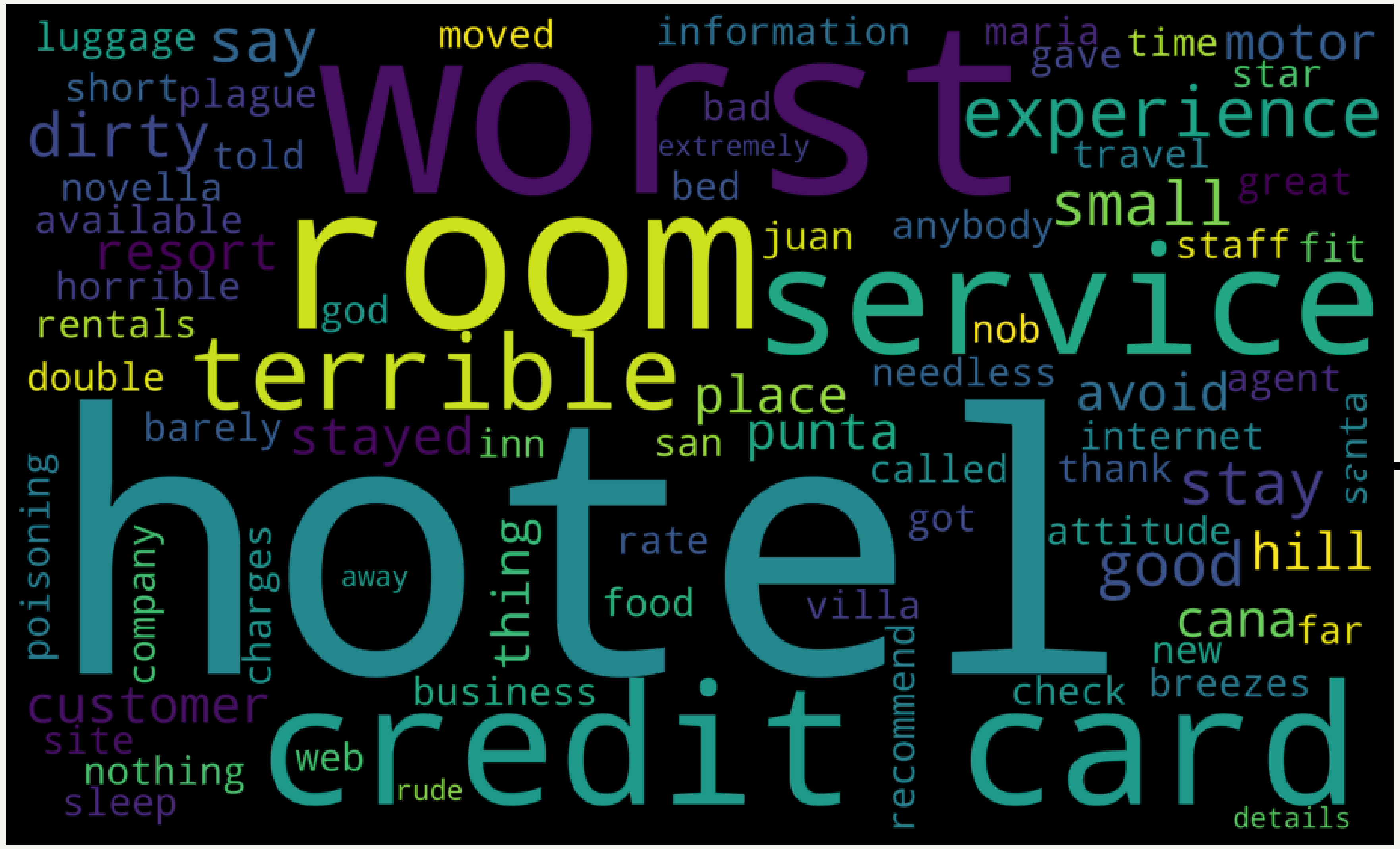
Bar Plot using Counter Function
Most frequent Negative trigrams



Negative Bigram_Wordcloud



Negative-Trigram-Wordcloud



Negative features/Review

Top negative points to be noted to improve the hotel's brand image on the basis of Ngrams and wordcloud are

- Worst hotel stayed,Awful night stay,Hotel charges additional charges on credit card,Worst vacation,Horrible experience
- Staff and room service related:
  Staff unwelcoming and rude
  Staff smoking intentionally,cigerette smell in room
  Terrible service
  worst
- Room and infrastructure related
  Rooms like hospital rooms
  beds hard,blanket rough
  Small double bed
  nasty frige with odor of rotten vegetables
  Dirty and tiny carpet
  Geyser issue-took cold shower
  Noisy elevator
  Tiles loose
  Broken switches and sagged LCD Tv
  Digital box not working
  AC not working
  Dirty bathroom
- Noise issue:
  Cant sleep in night due to the sounds of walking and talking
  Noisy neighbours
  Noise from road ,can't sleep,wink night

TOPIC MODELING

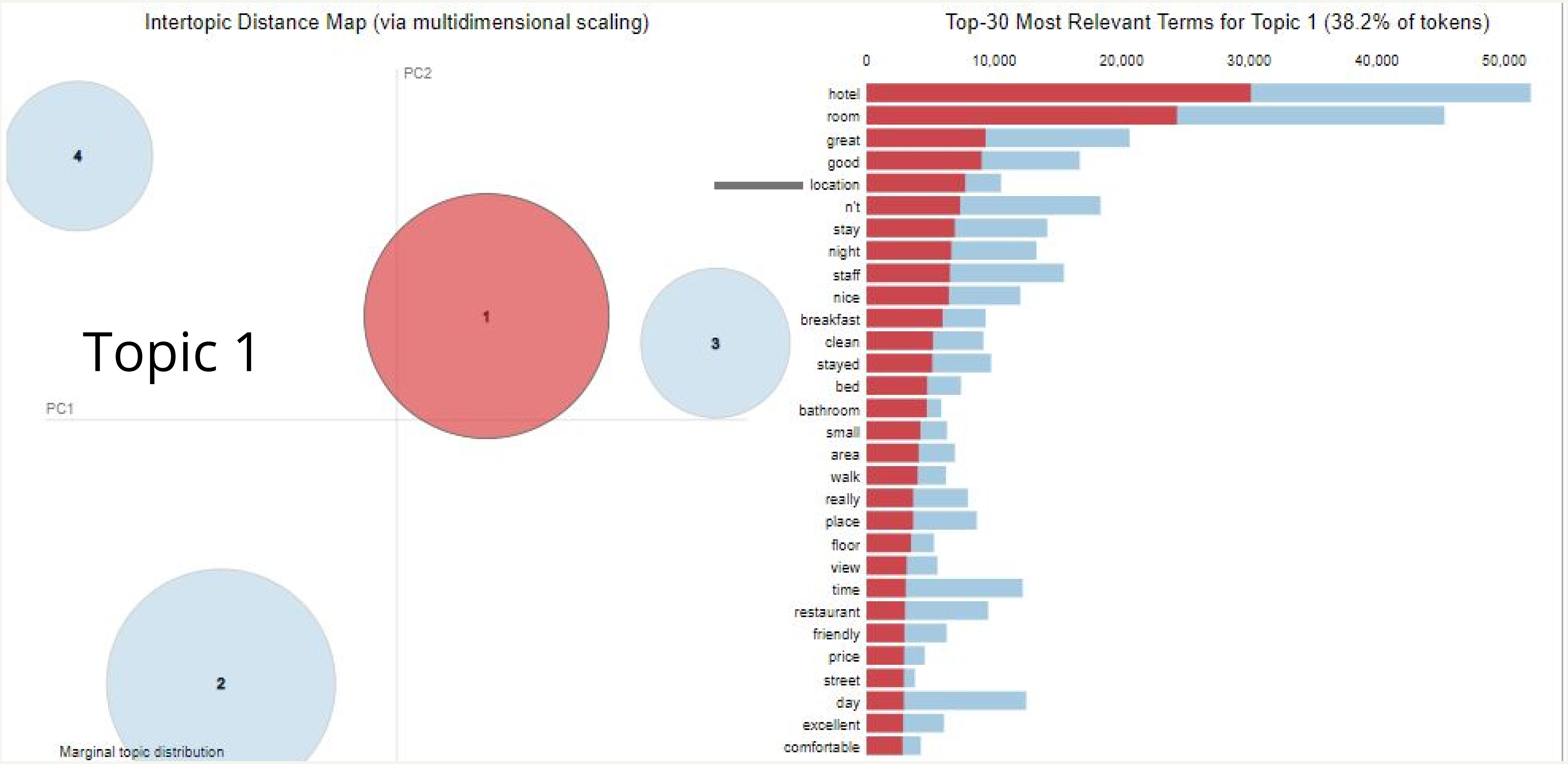
Topic modeling is a statistical modeling for discovering te abstract that occur in a collection of documents. It helps to uncover the hidden structure in a collection of texts

Most common algorithms used to perform topic modeling are

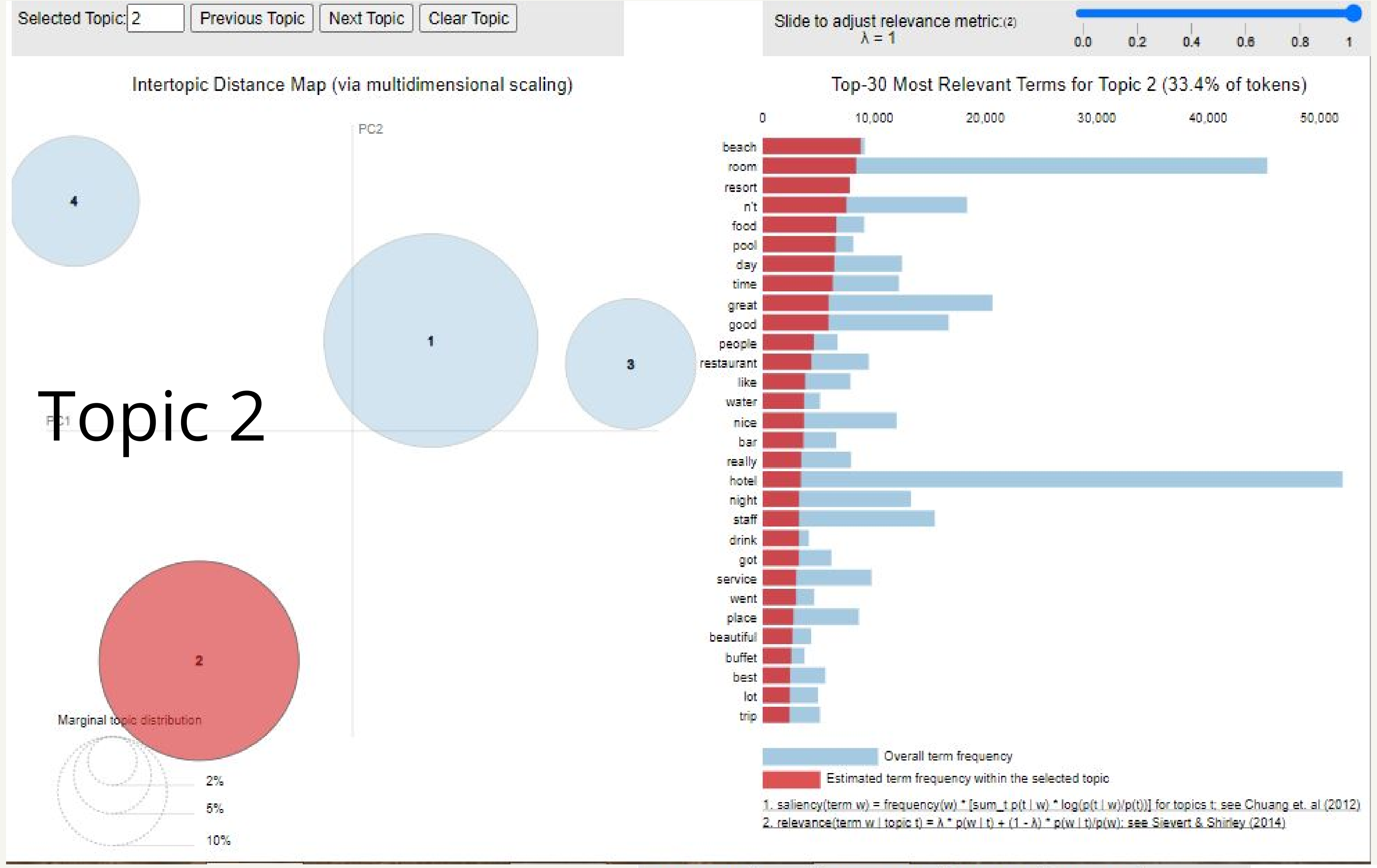
- Latent Dirichlet Allocation(LDA)
- Latent Semantic Analysis (LSA)
- Probabilistic Latent Semantic Analysis(PLSA)

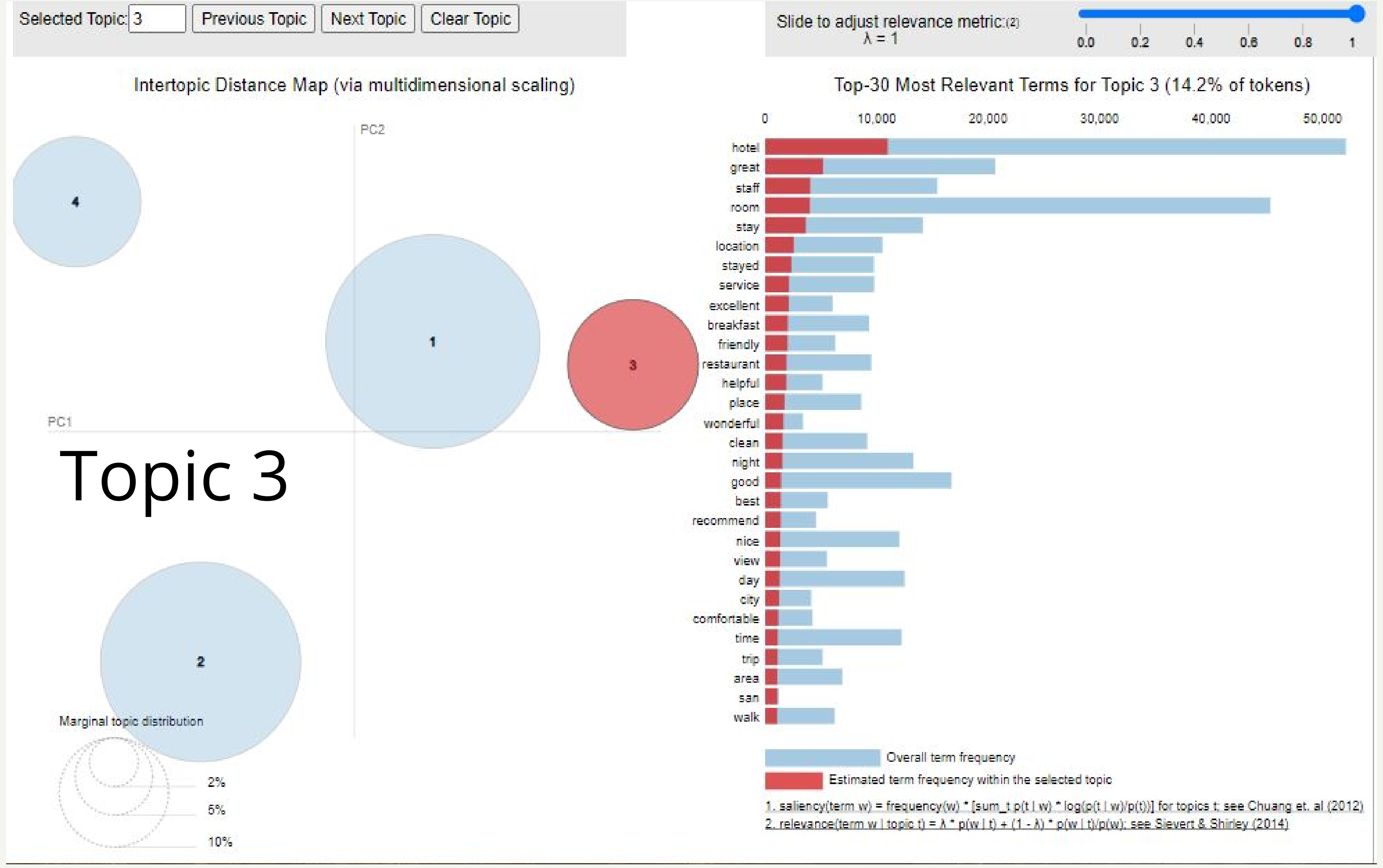
TOPIC MODELING - LDA

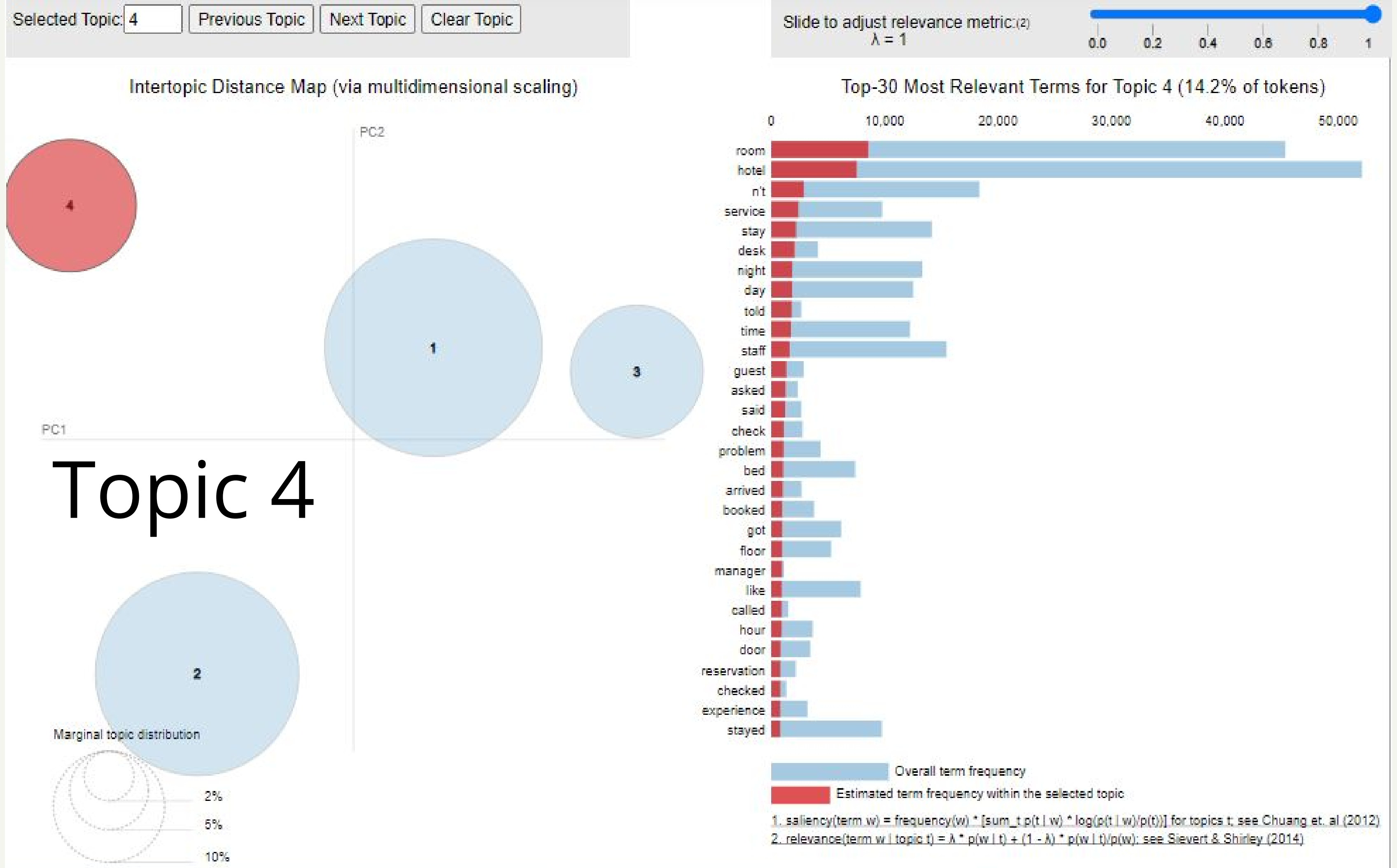
- 4 topics seletced by using Topic Modeling'
- Topic 1 - **Room** + **Location**
- Topic 2 - **Beach** + Room+ **Resort** + **Food**
- Topic 3 - **Staff** + Room + Stay
- Topic 4 - **Room Service** and **Hotel service**



- Area of the circle represents the importance of the topic.
- Distance between the center of the circle represents the similarity between the topic.
- Bar represents total frequency of term in entire dataset.
- Dark Bar represents the extent to which it belongs to that topic.


Selected Topic: 2
Previous Topic
Next Topic
Clear Topic
Slide to adjust relevance metric:(2)
λ = 1
Intertopic Distance Map (via multidimensional scaling)
Top-30 Most Relevant Terms for Topic 2 (33.4% of tokens)
Topic 2
PC2
PC1
Marginal topic distribution
2%
5%
10%
beach
room
resort
n't
food
pool
day
time
great
good
people
restaurant
like
water
nice
bar
really
hotel
night
staff
drink
got
service
went
place
beautiful
buffet
best
lot
trip
Overall term frequency
Estimated term frequency within the selected topic
1. saliency(term w) = frequency(w) * [sum_t p(t | w) * log(p(t | w)/p(t))] for topics t; see Chuang et. al (2012)
2. relevance(term w | topic t) = λ * p(w | t) + (1 - λ) * p(w | t)/p(w); see Sievert & Shirley (2014)


Selected Topic: 3
Previous Topic
Next Topic
Clear Topic
Slide to adjust relevance metric:(2)
λ = 1
Intertopic Distance Map (via multidimensional scaling)
Top-30 Most Relevant Terms for Topic 3 (14.2% of tokens)
Topic 3
PC2
PC1
Marginal topic distribution
2%
5%
10%
hotel
great
staff
room
stay
location
stayed
service
excellent
breakfast
friendly
restaurant
helpful
place
wonderful
clean
night
good
best
recommend
nice
view
day
city
comfortable
time
trip
area
san
walk
Overall term frequency
Estimated term frequency within the selected topic
1. saliency(term w) = frequency(w) * [sum_t p(t | w) * log(p(t | w)/p(t))] for topics t; see Chuang et. al (2012)
2. relevance(term w | topic t) = λ * p(w | t) + (1 - λ) * p(w | t)/p(w); see Sievert & Shirley (2014)


Selected Topic: 4
Previous Topic
Next Topic
Clear Topic
Slide to adjust relevance metric:(2)
λ = 1
Intertopic Distance Map (via multidimensional scaling)
Top-30 Most Relevant Terms for Topic 4 (14.2% of tokens)
Topic 4
PC2
PC1
Marginal topic distribution
2%
5%
10%
room
hotel
n't
service
stay
desk
night
day
told
time
staff
guest
asked
said
check
problem
bed
arrived
booked
got
floor
manager
like
called
hour
door
reservation
checked
experience
stayed
Overall term frequency
Estimated term frequency within the selected topic
1. saliency(term w) = frequency(w) * [sum_t p(t | w) * log(p(t | w)/p(t))] for topics t; see Chuang et. al (2012)
2. relevance(term w | topic t) = λ * p(w | t) + (1 - λ) * p(w | t)/p(w); see Sievert & Shirley (2014)

FEATURE EXTRACTION

Machine only knows numbers .So here we have to convert text data into a numerical format called vectors (Words in reviews are converting into vectors). The process of conversion of unstructured text data into structured format is called Feature Extraction

Feature extraction techniques:

- **Bag Of Words(Count Vectorization)**
- **TFIDF Vectorization**
- **Word Embedding(Word2Vec)**

TFIDF -Feature Extraction

In TFIDF we get diifferent values rather than zeroes and 1's
TFIDF follows weight scheme &it gives an idea about how important a word is.

WORD EMBEDDING(Word2Vec)

In which each word is represented by using a vector in three dimensional space. Words with similar meaning should have similar representation. These representation helps to identify synonyms, antonyms and various other relationship between words .
Word2Vec uses simple neural network.
in Word2Vec feature extraction a word is known by the neighborhood words.
Word2Vec helps to predict what word comes next and what is the context of the word to be used. Word2Vec predict the target word based on the context word and viceversa

MODEL BUILDING

Machine learning algorithms used for the text classification are

1. **Random Forest Classifier**
2. **Support Vector Machine**
3. **Logistic Regression**
4. **Boosting**
5. **KNN**
6. **Decision Tree**
7. **Naive Bayes**

Model Building

we classified reviews based on the rating and labelled the reviews , positive (Rating>3) , neutral(Rating=3) and negative(Rating<3)

	Review	Rating	label
0	nice hotel expensive parking got good deal sta...	4	positive
1	ok nothing special charge diamond member hilto...	2	negative
2	nice rooms experience hotel monaco seattle go...	3	neutral
3	unique great stay wonderful time hotel monaco ...	5	positive
4	great stay great stay went seahawk game awesom...	5	positive
...	...	...	...
20486	best kept secret rd time staying charm star be...	5	positive
20487	great location price view hotel great quick pl...	4	positive
20488	ok looks nice modern outside desk staff partic...	2	negative
20489	hotel theft ruined vacation hotel opened sept ...	1	negative
20490	people talking believe excellent ratings hotel...	2	negative

MODEL BUILDING-TFIDF Vectorization

Model (TF-IDF approach)	Classification Report

Random Forest Classifier
Train accuracy=100%

	precision	recall	f1-score	support
negative	0.93	0.19	0.32	649
neutral	1.00	0.01	0.01	408
positive	0.77	1.00	0.87	3042
accuracy			0.77	4099
macro avg	0.90	0.40	0.40	4099
weighted avg	0.82	0.77	0.70	4099

SUPPORT VECTOR MACHINE
Train accuracy= 99.7

	precision	recall	f1-score	support
negative	0.89	0.35	0.50	649
neutral	1.00	0.02	0.03	408
positive	0.79	1.00	0.88	3042
accuracy			0.80	4099
macro avg	0.89	0.45	0.47	4099
weighted avg	0.83	0.80	0.74	4099

Logistic Regression
Train accuracy= 84

	precision	recall	f1-score	support
negative	1.00	0.06	0.11	649
neutral	0.00	0.00	0.00	408
positive	0.75	1.00	0.86	3042
accuracy			0.75	4099
macro avg	0.58	0.35	0.32	4099
weighted avg	0.71	0.75	0.65	4099

eXtreme Gradient Boosting (XG Boosting)
Train accuracy=75
Good model

	precision	recall	f1-score	support
negative	0.86	0.10	0.17	649
neutral	0.69	0.02	0.04	408
positive	0.76	1.00	0.86	3042
accuracy			0.76	4099
macro avg	0.77	0.37	0.36	4099
weighted avg	0.77	0.76	0.67	4099

KNN
Train accuracy=81.6

	precision	recall	f1-score	support
negative	0.74	0.31	0.44	649
neutral	0.26	0.06	0.10	408
positive	0.79	0.97	0.87	3042
accuracy			0.78	4099
macro avg	0.60	0.45	0.47	4099
weighted avg	0.73	0.78	0.73	4099

DECISION TREE
Train accuracy= 100

	precision	recall	f1-score	support
negative	0.48	0.43	0.45	649
neutral	0.23	0.14	0.18	408
positive	0.82	0.88	0.85	3042
accuracy			0.74	4099
macro avg	0.51	0.48	0.49	4099
weighted avg	0.71	0.74	0.72	4099

Comparatively better model is XGB model

MODEL BUIDING-Word2Vec Feature Extraction

Word Embedding-Models (Word2Vec)

Classification Report

Random Forest Classifier
Train accuracy=100%

	precision	recall	f1-score	support
negative	0.76	0.68	0.72	649
neutral	0.38	0.01	0.02	408
positive	0.85	0.98	0.91	3042
accuracy			0.83	4099
macro avg	0.66	0.56	0.55	4099
weighted avg	0.79	0.83	0.79	4099

Naïve Bayes
Train accuracy=70.99

	precision	recall	f1-score	support
negative	0.64	0.41	0.50	649
neutral	0.15	0.17	0.16	408
positive	0.80	0.85	0.83	3042
accuracy			0.71	4099
macro avg	0.53	0.48	0.49	4099
weighted avg	0.71	0.71	0.71	4099

Logistic Regression
Train accuracy=84

	precision	recall	f1-score	support
negative	0.75	0.73	0.74	649
neutral	0.34	0.15	0.21	408
positive	0.89	0.96	0.92	3042
accuracy			0.84	4099
macro avg	0.66	0.61	0.62	4099
weighted avg	0.81	0.84	0.82	4099

eXtreme Gradient Boosting (XG Boosting)
Train accuracy=84

	precision	recall	f1-score	support
negative	0.75	0.70	0.73	649
neutral	0.32	0.06	0.10	408
positive	0.86	0.97	0.91	3042
accuracy			0.84	4099
macro avg	0.64	0.57	0.58	4099
weighted avg	0.79	0.84	0.80	4099

DECISION TREE
Train accuracy=100

	precision	recall	f1-score	support
negative	0.53	0.54	0.54	649
neutral	0.14	0.16	0.15	408
positive	0.85	0.83	0.84	3042
accuracy			0.72	4099
macro avg	0.51	0.51	0.51	4099
weighted avg	0.73	0.72	0.72	4099

KNN
Train accuracy=85

	precision	recall	f1-score	support
negative	0.66	0.64	0.65	649
neutral	0.20	0.11	0.14	408
positive	0.86	0.93	0.89	3042
accuracy			0.80	4099
macro avg	0.58	0.56	0.56	4099
weighted avg	0.77	0.80	0.78	4099

Both Logistic Regression and XG boosting have high accuracy score=0.84

Model-Accuracy

MODEL (TFIDF)	TRAIN ACCURACY	TEST ACCURACY
RANDOM FOREST CLASSIFIER	100	77
SUPPORT VECTOR MACHINE	99.7	80
LOGISTIC REGRESSION TRAIN ACCURACY	84	75
EXTREME GRADIENT BOOSTING (XG BOOSTING)	75	76
KNN	81.6	78
DECISION TREE	100	74
MODEL (Word2VEC)	**TRAIN ACCURACY**	**TEST ACCURACY**
RANDOM FOREST CLASSIFIER	100	83
NAÏVE BAYES	70.99	71
LOGISTIC REGRESSION	84	84
EXTREME GRADIENT BOOSTING (XG BOOSTING)	84	84
DECISION TREE	100	72
KNN	85	80

RESAMPLING-SMOTE

We tried to improve the model accuracy by using resampling technique named SMOTE(Synthetic Minority Oversampling Technique).

We got good accuracy for Random Forest Classifier(96.4) and Logistic Regression(98.34)

RESAMPLED DATA (Balanced Data)-MODEL ACCURACY

RESAMPLED-MODEL (TFIDF)	TRAIN ACCURACY	TEST ACCURACY
RANDOM FOREST CLASSIFIER	100	96.4
LOGISTIC REGRESSION TRAIN ACCURACY	99.94	98.34
EXTREME GRADIENT BOOSTING (XG BOOSTING)	73.71	72.6
KNN	66.47	66
DECISION TREE	100	84.46
RESAMPLED_MODEL (Word2VEC)	TRAIN ACCURACY	TEST ACCURACY
RANDOM FOREST CLASSIFIER	100	83
NAÏVE BAYES	100	93.32
LOGISTIC REGRESSION	74.5	74.3
EXTREME GRADIENT BOOSTING (XG BOOSTING)	76.31	74.80
DECISION TREE	100	79.03
KNN	87.25	82.69

Classification Report- Balanced Data

Random Forest
Train accuracy 100%

	precision	recall	f1-score	support
negative	0.99	0.94	0.97	3011
neutral	1.00	0.95	0.98	3024
positive	0.90	0.99	0.95	3021
accuracy			0.96	9056
macro avg	0.97	0.96	0.96	9056
weighted avg	0.97	0.96	0.96	9056

Logistic Regression
Train Accuracy= 99.9

	precision	recall	f1-score	support
negative	0.99	0.99	0.99	3011
neutral	0.99	0.99	0.99	3024
positive	0.98	0.97	0.98	3021
accuracy			0.98	9056
macro avg	0.98	0.98	0.98	9056
weighted avg	0.98	0.98	0.98	9056

Decision Tree
Train Accuracy=100

	precision	recall	f1-score	support
negative	0.87	0.86	0.87	3011
neutral	0.85	0.88	0.87	3024
positive	0.84	0.81	0.83	3021
accuracy			0.85	9056
macro avg	0.85	0.85	0.85	9056
weighted avg	0.85	0.85	0.85	9056

KNN
Train Accuracy-66.47

	precision	recall	f1-score	support
negative	0.79	0.98	0.88	3011
neutral	0.57	1.00	0.72	3024
positive	0.00	0.00	0.00	3021
accuracy			0.66	9056
macro avg	0.45	0.66	0.53	9056
weighted avg	0.45	0.66	0.53	9056

eXtreme Gradient Boosting (XG Boosting) Train accuracy=73.36

	precision	recall	f1-score	support
negative	0.80	0.63	0.71	3011
neutral	0.76	0.68	0.72	3024
positive	0.66	0.87	0.75	3021
accuracy			0.73	9056
macro avg	0.74	0.73	0.72	9056
weighted avg	0.74	0.73	0.72	9056

Deployment

Deployment done by using

- streamlit
- Flask

THANK YOU