SENTIMENT ANALYSIS OF HOTEL REVIEWS

Department of Information Technology
Veer Surendra Sai University of Technology
December-2019

Submitted by
Aanchal Singh

ABSTRACT

With everything shifting online, brands have started giving utmost importance to Sentiment Analysis. It's the only gateway to thoroughly understand customer-base and their expectations from the brand. It can help organizations from any domain by understanding the grievances and tackle the exact problems and concerns of their customers which eventually help the organizations scale up their services. It is helpful even for brands suffering from data overload as huge amount of data are collected from customers, those can't be manually analyzed. So in Sentiment Analysis the data goes throw a number of processes and algorithms which finally score as positive, negative, and neutral data which reduces a lot of time and effort in analyzing each and every data from the corpus.

User reviews and comments on hotels on the web are an important information source in travel planning. Therefore, knowing about these comments is important for quality control to the hotel management, too. We present a system that collects such comments from the web and creates classified and structured overviews of such comments and facilitates access to that information.

Sentiment Analysis is the analysis of the feelings (i.e. emotions, attitudes, opinions, thoughts, etc.) behind the words by making use of Natural Language Processing (NLP) tools. Sentiment Analysis also uses Natural Language Processing and Machine Learning to help organizations look far beyond just the numbers and texts.

Chapter 1

Introduction

1.1 Objective:

To reduce human efforts to analyse the sentiment of hotel aspects and identification of high rated hotels as per convenient person to person.

1.2 How customer Reviews are important?

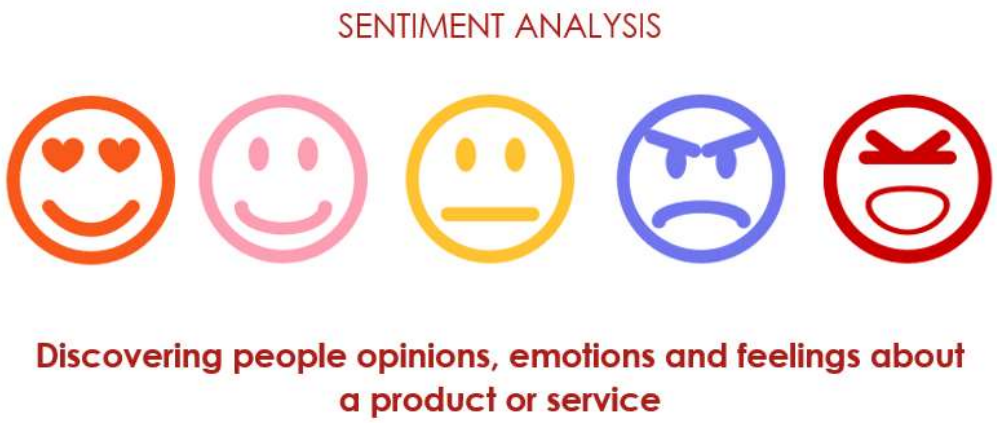
The internet has brought all things to our fingertips, from buying groceries to researching our next new automobile purchase. The internet is now evolving to be a forum where consumers evaluate products and services based on impressions and feedback from other, like-minded consumers. Nothing highlights more than objective data just how reviews are used and how they impact business. Statistics analyse how customers behave before and after using services or buying products, which can help develop plans to improve business.

Reviews not only have the power to influence consumer decisions but can strengthen a company's credibility. Reviews have the power to gain customer trust, and they encourage people to interact with the company. Customer interaction ultimately leads to improved profits for businesses.

BrightLocal (2017) studies indicate that 97% of consumers use reviews to search for local services. Specifically, 60% of consumers read reviews for restaurants and cafes, 40% for B&B's, and 33% for medical services.

1.3 What is Sentiment Analysis and Why Does it Matter?

Sentimentanalysis, also called opinion mining, is a text mining technique that could extract emotions of a given text — whether it is positive, negative or neutral, and return a sentiment score. This technique is usually used on reviews or social media texts. It is a Natural Language Processing and Information Extraction task that aims to obtain writers feelings expressed. Natural language processing (NLP) is a field of computer science, artificial intelligence, and linguistics concerned with the interactions between computers and human (natural) languages.



It's estimated that 80% of the world's data is unstructured and not organized in a pre-defined manner. Most of this comes from text data, like emails, support tickets, chats, social media, surveys, articles, and documents. These texts are usually difficult, time-consuming and expensive to analyze, understand, and sort through.

Sentiment analysis systems allows companies to make sense of this sea of unstructured text by automating business processes, getting actionable insights, and saving hours of manual data processing, in other words, by making teams more efficient.

1.4 Advantages of Sentiment Analysis:

Ideally, sentiment analysis can be put to use by any brand looking to:

- Target specific individuals to improve their services.
- Track customer sentiment and emotions over time.
- Determine which customer segment feels more strongly about your brand.
- Track the changes in user behaviour corresponding to the changes in your product.
- Find out your key promoters and detractors.
- Reduces a lot of time and effort.

Chapter 2

LITERATCHER REVIEW

Natural Language Processing is a field at the intersection of computer science, artificial intelligence, and linguistics. The goal is for computers to process or "understand" natural language in order to perform various human like tasks like language translation or answering questions.With the rise of voice interfaces and chatbots, NLP is one of the most important technologies of the 4th Industrial Revolution and become a popular area of AI. There's a fast-growing collection of useful applications derived from the NLP field.

Sentiment analysis is part of the Natural Language Processing (NLP) techniques that consists in extracting emotions related to some raw texts. This is usually used on social media posts and customer reviews in order to automatically understand if some users are positive or negative and why.
Though, it is quite difficult to find the true relation between sentiment analysis, natural language processing and machine learning as all they are different from each other in terms of their use and processing at different places.

However, there is one thing common between them that they work on the principles of Artificial Intelligence based technology to respond or work automatically without too much human intervention. So, to make it stronger and more precise let's find out what is the technical relation between sentiment analysis, machine learning and natural language processing.

Machine learning is an area of computer science that allows computer systems a unique ability to learn with data without reprogramming and respond accordingly. Machine Learning is a kind of self-learning process that when exposed to new data or computer programs it learns, grow, change and develop themselves without human intervention.

Sentiment analysis is a kind of opinion mining of customers, users and audience through social media and other online platforms towards a particular product, services, brand and companies. Sentiment analysis is a kind of measurement of positive and negative languages and it helps you to see what customers like and dislike about you, your brand and your company. Sentiment Analysis helps to investigate the feelings prevail about a certain things and reviewing your customer's feedback on your business regularly.

Natural Language processing is superset of sentiment analysis that means there are more categories of problems that comes into various classes. The sentiment analysis is an application for extracting the sentiments from certain amount of texts. While Machine learning may not be used in NLP Sentiment analysis but if ML is used correctly, if can help you to boost the performance of NLP systems or sentiment analysis software used for such thing.

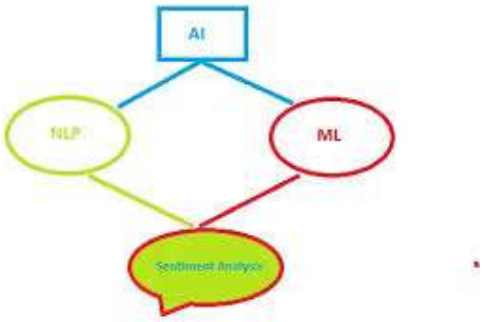


FIG 2.1: RELATION SHIP BETWEEN AI, NLP, ML, SENIMENT ANALYSIS

Most existing studies, for determining sentiment analysis, can be classified into two types, i.e. supervised methods and lexicon based methods. Classifiers those are most likely used by the aforementioned are supervised methods. The supervised methods are mainly based on training classifiers such as Naïve Bayes, Support Vector Machine, Random Forest etc.

Lexicon based methods determine the overall sentiment tendency of a given text by utilizing pre-established lexicon of words weighted with their sentiment orientation, such as SentiWordNet. These methods depend on the presence of lexical or syntactical feature that explicitly express the sentiment information. Though, in lot of cases, the sentiment of tweet is implicitly related with the context semantic.

On the other hand, Deep Learning (DL) methods do not depend upon extensive manual feature engineering and extracts the features simultaneously. They have the advantage of inherently taking into account the word ordering and by using word vectors they encompass syntactic and semantic meaning of words.

In this project we have use Naïve Bayes learning method which is a supervised learning method for training of our model.

Supervised learning is a learning model built to make prediction, given an unforeseen input instance. A supervised learning algorithm takes a known set of input dataset and its known responses to the data (output) to learn the regression/classification model. A learning algorithm then trains a model to generate a prediction for the response to new data or the test dataset. Supervised learning usesclassification algorithms and regression techniques to develop predictive models. The algorithms include linear regression, logistic regression and neural networkas decision tree, Support vector machine (SVM), random forest, naive Bayes, and *k*-nearestneighbour.

Linear regression and Linear classifier. Despite an apparent simplicity, they are very useful on a huge amount of features where better algorithms suffer from overfitting.

Logistic regression is the simplest non-linear classifier with a linear combination of parameters and nonlinear function (sigmoid) for binary classification.

Decision trees is often similar to people's decision process and is easy to interpret. But they are most often used in compositions such as Random forest or Gradient boosting.

K-means is more primal, but a very easy to understand algorithm, that can be perfect as a baseline in a variety of problems.

PCA is a great choice to reduce dimensionality of your feature space with minimum loss of information.

Neural Networks are a new era of machine learning algorithms and can be applied for many tasks, but their training needs huge computational complexity.

Chapter 3

PROPOSED ALGORITHM

3.1 PROBLEM STATEMENT

In the given project work, the problem lies in detection of sentiment of a given input text corpus. The problem is to calculate the accuracy of proposed algorithm against the large hotel review dataset. In the end we calculate the accuracy test data with the trained classification model. We classify all the given hotel reviews based on the ratings given by the customers. At the end, the graphs were plotted to demonstrate the obtained outputs.

3.2 PRELIMINARIES

3.2.1 Data Collection

Before designing a model, first we have to collect the historical data using which the model is going to be trained. For this project, the hotel review dataset is scrapped from 'Booking.com'. This dataset contains 515,000 customer reviews and scoring of 1493 luxury hotels across Europe. Meanwhile, the geographical locations of hotels are also provided for further analysis.

The dataset contains 17 attributes. The description of each field is as below:

- Hotel_Address: Address of Hotel.
- Review_Date: Date when reviewer posted the corresponding review
- Average_Score: Average score of the hotel, calculated based on the latest comment in the last year.
- Hotel_Name: name of the hotel.
- Reviewer_Nationality: Nationality pf the reviewer.
- Negative_Review: Negative review of the reviewer gave to the hotel.If the reviewer does not provide the negative review, then it should be: 'No Positive'.
- Review_Total_Negetive_Word_Counts: Total number of words in the negative review.
- Positive_Review: Positive review of the reviewer gave to the hotel. If the reviewer does not provide the negative review, then it should be: 'No Negative'.
- Review_Total_Positive_Word_Counts: Total number of words in the positive review.
- Reviewer_Score: Score the reviewer has given to the hotel based on his/her experience.
- Total_Number_Of_Reviews_Reviewer _Has_Given: Number of reviews the reviewer has given.
- Total_Number_Of_Reviews: Total number of valid reviews the hotel has.
- Tags: Tags the reviewer gave the hotel.
- days_since_review: Duration between the review date and scrape date.
- Additional_Number_of_Scoring: There are also some guests who just made a scoring on the service rather than a review.This number indicates how many valid scores without review in there.
- lat: Latitude of the hotel.

- lng: Longitude of the hotel

3.2.2 Data pre-processing

Data preprocessing is a technique that involves transforming of raw data into an understandable form. Real world data is often incomplete, inconsistent and lacks certain behavior. The dataset used here also may lack attribute values and may contain only aggregated data. It may contain missing values or heterogenous values and have a lot of noise. After preprocessing the data is prepared to me trained. The steps to data pre-processing is as follows:

3.2.2.1 Tokenization

In tokenization step, tokens are generated from the textual file (in this case it is Hotelreview dataset). Tokens are individual words in one or many sentences. A text or a group of text is taken and broken up to it's individual words. These tokens are then used as input for further analysis. Tokenization can be done in sentence level or word level. In this project we are going use word tokenization for pre-processing.

For example:

Given a sentence: "This is a final year B.tech project"

After word tokenization:

['This','is','a','final',' year', 'B.tech',' project']

3.2.2.2 Removal of Stopwords

One of the major forms of pre-processing is to filter out unwanted data. In natural language processing, useless words are referred as stop word. A stop word is a commonly used word (such as 'the','a','an','in'etc)that a search engine has been programed to ignore, both when indexing entries for searching and when retrieving them as the result of a search query. We don't want these words taking up spaces in our database, or taking up valuable processing time.

For Example:

Given a sentence: "This is a final year B.tech project"

After word tokenization:

['This','is','a','final',' year', 'B.tech',' project']

After stop words removal:

['final','year','B-tech','project']

3.2.2.2 Removal of Special characters

In this step special characters like '@,URL which are creating only noises are removed from the sentence because those words are not contributing towards sentiment analysis. Use of regular expression was employed to the task.

3.2.3 Libraries Used:

Numpy: Itis a general-purpose array-processing package. It provides a high-performance multidimensional array object, and tools for working with these arrays. It is the fundamental package for scientific computing with Python.
Besides its obvious scientific uses, Numpy can also be used as an efficient multi-dimensional container of generic data.

Pandas: Pandas is an opensource library that allows to you perform data manipulation in Python. Pandas library is built on top of Numpy, meaning Pandas needs Numpy to operate. Pandas provide an easy way to create, manipulate and wrangle the data. Pandas is also an elegant solution for time series data.

Matplotlib: Matplotlib is an amazing visualization library in Python for 2D plots of arrays. Matplotlib is a multi-platform data visualization library built on NumPy arrays and designed to work with the broader SciPy stack. It was introduced by John Hunter in the year 2002.

One of the greatest benefits of visualization is that it allows us visual access to huge amounts of data in easily digestible visuals. Matplotlib consists of several plots like line, bar, scatter, histogram etc.

Seaborn: Seaborn is a library for making statistical graphics in Python. It is built on top of matplotlib and closely integrated with pandas data structures. Seaborn aims to make visualization a central part of exploring and understanding data. Its dataset-oriented plotting functions operate on dataframes and arrays containing whole datasets and internally perform the necessary semantic mapping and statistical aggregation to produce informative plots.

NLTK: The Natural Language Toolkit (NLTK) is a platform used for building Python programs that work with human language data for applying in statistical natural language processing (NLP).It contains text processing libraries for tokenization, parsing, classification, stemming, tagging and semantic reasoning. It also includes graphical demonstrations and sample data sets as well as accompanied by a cook book and a book which explains the principles behind the underlying language processing tasks that NLTK supports.

The Natural Language Toolkit is an open source library for the Python programming language originally written by Steven Bird, Edward Loper and Ewan Klein for use in development and education.

It comes with a hands-on guide that introduces topics in computational linguistics as well as programming fundamentals for Python which makes it suitable for linguists who have no deep knowledge in programming, engineers and researchers that need to delve into computational linguistics, students and educators.

RE: The Python module **re** provides full support for Perl-like regular expressions in Python. The re module raises the exception re.error if an error occurs while compiling or using a regular expression.

We would cover two important functions, which would be used to handle regular expressions. But a small thing first: There are various characters, which would have special meaning when they are used in regular expression. To avoid any confusion while dealing with regular expressions, we would use Raw Strings as **r'expression'**.

String: This module contains a number of functions to process standard Python strings.In Python 1.6 and later, most string operations are made available as string methods as well, and many functions in the string module are simply wrapper functions that call the corresponding string method.

Time: The Python time module provides many ways of representing time in code, such as objects, numbers, and strings. It also provides functionality other than representing time, like waiting during code execution and measuring the efficiency of your code.

3.2.4 Designing Classification Model:

Definition: Classification is the technique related to categorization that is the process in which idea and objects are recognized, differentiated and understood based on certain criteria. Classification divides different data into labelled groups. The data in each group are similar with each other and differs from data of other groups.

Based on the criteria there can be number of classes and each class can have number of sub classes. It may so happen an object which belong to a particular class may change it's orientation to another class if the criteria changes.

For example: During admission student of both boys and girls may belong to same class if the criteria is 'department'. By changing the criteria to 'gender' the same boys and girls who used to belong to same class will now be in different class.

Classification is a two steps process:

1. classifier is built based on previous data.
2. The model is tested with a new set of similar data.

When the classifier is built, a predetermined set of data is provided as input. This data is known as learning data and the process is known as learning process.

The learning data is used to train the model that is update the parameters of the model. For testing purpose, a new set of data is provided as input to test the accuracy of the model.

If the accuracy is accepted then the model is used for scientific and commercial purpose

Training DataIt is made up of dataset consisting of tuples and a class associated with each tuple. A tuple is a vector of n-dimension,

Example: X= {x1,x2,x3,……xn}

The training tuples are randomly selected for training the parameter. Classification used supervised learning.

Testing Data:Testing dataset is used to check the accuracy of the model. During the testing the tuples which are not used in training are used. The output of the model is compared with the class value of each model so as to decide the accuracy of the model.

In a general concept, 70% of the data is used for training and 30% is used for testing.

In this project 75% of the hotel review dataset is used for training and 25% is used for testing purpose.

Classifier: A classifier is a machine learning model that is used to discriminate different objects based on certain features.

For this project we are going to use Naïve Bayes algorithm.

Naïve Bayes Algorithm

This algorithm is a machine learning model based on probabilistic classification. It is also called as statistical classifier.

Principle of Naïve Bayes Classifier

The principle of this algorithm is based on Bayes Theorem. Thus, this algorithm is also called as Bayesian classifier.

Let there be a data tuple X which belongs to a class C. Given that we know the attributes of X, then probability of X belonging to class C is given as:

$$p(H|x) = (P(x|H) * P(H))/P(x)$$

Here H = Some Hypothesis

x= Attribute value of a particular row or pattern

X = The whole data tuple

C= One of the class in the dataset

Step-1:Let B be a dataset consisting of tuples, attributes, class values

Step-2: Let there are 'm' number of classes denoted by C1, C2….Cm.

Given a tuple 'X'. The theorem can predict to which class the tuple belongs to. A tuple will belong to the class having highest probability. The inputs to the classifiers are the attribute values. They are represented as:

A= {A1,A2,A3…..An}

Step-3: According to above, the Bayes Theorem can be written as:

$$p(Ci|x) = (P(x|Ci) * P(Ci))/P(x)$$

The variable 'Ci' is the class variable and 'x' represent the feature or attribute.

As probability of a particular attribute that is $P(x)$ is constant for all the classes, only $(P(x|Ci) * P(Ci))$ need to be calculated. Because we have to maximize the probability of classes that is $P(Ci)$.

So, probability of class can be given as:

$P(Ci)$= |Cid| / |d|

Where, Cid= number of time a class, Ci has occurred in the dataset

|d| = number of existing tuples

Step-4: If the attribute values are conditionally independent then,

$P(x|Ci) = \prod_{k=1}^{n} P(Xk|Ci)$

$= P(X1|Ci)\ P(X2|Ci)$…….

As X is a tuple consisting of a set of attributes {X1,X2,X3……Xk}

Step-5: To predict the class lable of X , P(Ci) is evaluated for each class, where

0 <i<m . The tuple X will belong to class Ci if the following condition is satisfied:

$(P(x|Ci) * P(Ci))>(P(x|Cj) * P(Cj))$

Where, i ≤ j ≤ m ,j≠i

Using the above five steps, the class for the given predictors can be obtained.

Chapter 4

Implementation & Results:-

4.1 PROBLEM STATEMENT

In the given project work, the problem lies in detection of sentiment of a given input text corpus. The problem is to calculate the accuracy of proposed algorithm against the large hotel review dataset. In the end we calculate the accuracy test data with the trained classification model. We classify all the given hotel reviews based on the ratings given by the customers. At the end, the graphs were plotted to demonstrate the obtained outputs.

4.2 COLLECT AND LOAD DATA

The dataset for the Hotel Review is collected and loaded from the below link:
https://www.kaggle.com/pacogiu/sentiment-analysis-from-hotel-reviews

It is loaded as:
df=pd.read_csv('Hotel_Reviews.csv')

The dataset consists of 17 features (attributes) and each feature has 515378 samples. It is stored in a tabular form. The rows being the samples and the columns being the features. We first start by loading the raw data. Each textual review is splitted into a positive and a negative part. We group them together in order to start with the raw data and no other information.

The shape of the dataset is viewed as:

```
display(df.shape)
```

```
(515738, 17)
```

4.3 DIVIDING THE DATASET:-

The dataset is divided into positive and negative review based on feature Reviewer_Score. If Reviewer_Score<5, then it falls under negative review. If Reviewer_Score>8, then it falls under positive review. It can be shown as:

```
neg_rev=df[df.Negative_Review!='No Negative'].reset_index().drop('index',1)
pos_rev=df[df.Positive_Review!='No Positive'].reset_index().drop('index',1)
neg_rev = neg_rev[neg_rev['Reviewer_Score']<5].reset_index().drop('index',1)
pos_rev = pos_rev[pos_rev['Reviewer_Score']>8].reset_index().drop('index',1)
```

4.4 DATA CLEANING:-

Data cleaning is the process of filtering and modifying the dataset such that it is easier to explore. Filtering out the parts you don't want or need such that you don't need to process them. Modifying the parts you need but are not in the format you need them to be so that you can properly use them.

4.4.1 WORD TOKENZATION:-

In this step, it takes the reviews and gives back the clean list of words. We use TokTok Tokenizer, to tokenize the given reviews. It is given as:

```
from nltk.tokenize import ToktokTokenizer
toktok = ToktokTokenizer()
```

TokTok tokenizer is faster than Word_tokenize. For large dataset TokTok tokenizer works better.

4.4.2 STOPWORD REMOVAL:-

Stopword- A stop word is a commonly used word (such as “the”, “a”, “an”, “in”) that a search engine has been programmed to ignore, both when indexing entries for searching and when retrieving them as the result of a search query.

All the stopwords, punctuation and numbers are removed. It is given as:

```
from nltk.corpus import stopwords
```

```
remove_punct_and_digits = dict([(ord(punct), ' ') for punct in string.punctuation + st
ring.digits])
stopWords = set(stopwords.words('english'))
```

4.5 SPLIT DATASET FOR TRAIN AND TEST:-

The dataset is divided into two groups, negative and positive review. We split our data into training and testing purpose. We set a cut-off of 75% for both negative and positive review for training purpose and 25% for testing purpose. It is given as:

```
negcutoff = int(len(neg_set)*3/4)
poscutoff = int(len(pos_set)*3/4)

trainfeats = neg_set[:negcutoff] + pos_set[:poscutoff]
testfeats = neg_set[negcutoff:] + pos_set[poscutoff:]
```

We use a Naïve Bayes Classifier to train our model. It is given as:

```
from nltk.classify import NaiveBayesClassifier
classifier = NaiveBayesClassifier.train(trainfeats)
```

4.6 CALCULATION OF SENTIMENT SCORE:-

The score of the word are calculated according to their sentiment.

- If *sentiment*=1, then *word* =positive and *score* <200
- If *sentiment*=0, then *word* =negative and *score* >200

It is given as:

```
word_scores=pd.DataFrame({'word':word,'sentiment':sentiment,'score':score})
neg_word_scores=word_scores[word_scores.sentiment==0]
pos_word_scores=word_scores[word_scores.sentiment==1]
display(word_scores[word_scores['sentiment']==1].head())
display(word_scores[word_scores['sentiment']==0].head())
neg_given=df[df.Negative_Review!='No Negative'].reset_index().drop('index',1)
pos_given=df[df.Positive_Review!='No Positive'].reset_index().drop('index',1)
```

	word	sentiment	score
0	worst	0	929.9
1	rude	0	691.1
2	dirty	0	648.6
3	disgusting	0	361.3
4	terrible	0	302.3

	word	sentiment	score
7	superb	1	198.0
20	delicious	1	127.1
47	excellent	1	77.8
54	fab	1	72.0
56	beautifully	1	70.8

4.7 TESTING THE SENTIMENT OF A PARTICULAR REVIEW:-

After the calculation of sentiment score, we give input to the model to check the sentiment of a particular review, whether it is a positive or negative review along with score.

```
print('negative score:',neg_sentiment_sum(df.Negative_Review[6584]),\
      'positive score:', pos_sentiment_sum(df.Positive_Review[6584]))
```

```
negative score: 0 positive score: 77.8
```

4.8 REVIEW GROUPED BY RATE BAND:-

Here the reviews are grouped based on their Reviewer_Score. It is given as:

```
score_9=df[df.Reviewer_Score>9].copy()
score_4=df[df.Reviewer_Score<4].copy()
score_6=df[df.Reviewer_Score<7].copy()
score_7=df[(df.Reviewer_Score>7)&(df.Reviewer_Score<8)].copy()
score_8=df[(df.Reviewer_Score>8)&(df.Reviewer_Score<9)].copy()
```

```
(86851, 17)
(92830, 17)
(88104, 17)
(246390, 17)
```

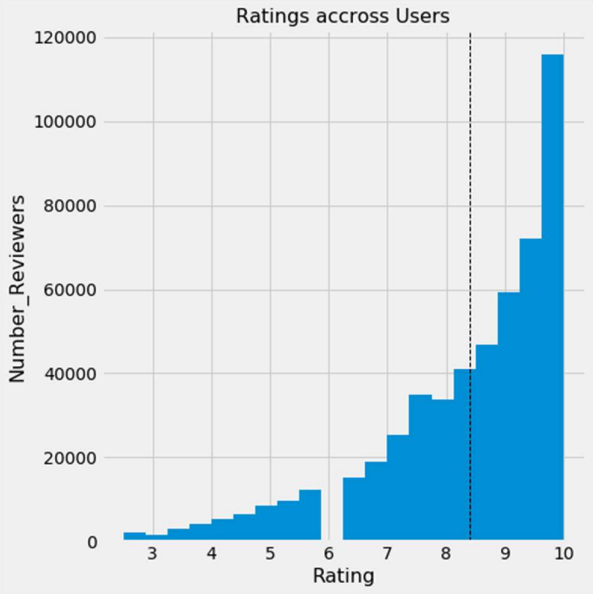
4.9 PLOTTING THE GRAPH AND RESULT:-

In python, the graphs are plotted by importing Matplotlib library. Matplotlib is a plotting library for the Python programming language and its numerical mathematics extension NumPy. rcParams is used so that one can dynamically change the default rc settings to customize the default parameters. It is imported as follows:

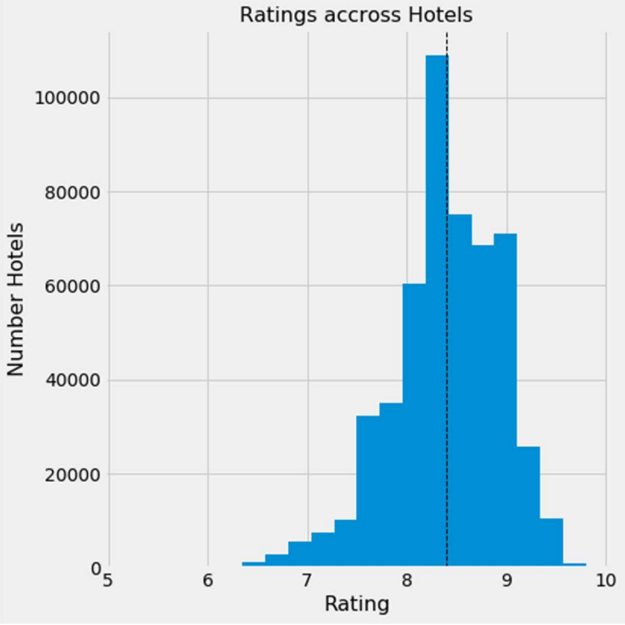
```
from matplotlib import rcParams
```

The results are shown in the following graphs:

- First graph is given as Rating Across Users- Number_of_Reviewer VS Ratings



- Second graph is given as Rating Across Hotel- Number of Hotels VS Rating



5. CONCLUSION AND FUTURE WORK:-

In today's language, sentiment analysis is a hot topic, but we are still unable to determine the sentiment of a corpus accurately because of the complexity of English language. The increase in in user-generated content has proved to be a rich source of subjective knowledge hence the challenge of understandings, analyzing, classifying and predicting the sentiment of these written languages accurately have become ever so important. In this project, the dataset is divided into two groups of negative and positive review set, further preprocess the two groups by tokenization and removing stopwords, used a Naïve Bayes Classifier to train the model, analyzing the sentiment of each word and plot the graph to determine the best rated hotel according to the number of reviewers and number of hotels.

Future work is to propose different Deep Learning approaches with some improved feature extraction methods to increase the accuracy of these deep learning model in order to create a simple recommendation engine to the guest who is fond of a special characteristic of hotel.