

# **CS6611 CREATIVE AND INNOVATIVE PROJECT**

## **B.E CSE VI Q - BATCH**

**LAB NO: 12**

**DATE: 26/04/2021**

**TEAM MEMBERS:**

**TEAM NO: 20**

<b>S.No.</b>	<b>REG. NO.</b>	<b>NAME</b>
1	2018103515	Aparna G
2	2018103521	Darrshana R
3	2018103596	S. Soumya

**Project Title:** Model generation to identify public opinion towards tourism form tweets

Observation document (5)	
On the Spot exercise (5)	
Laboratory exercises identified (15)	
Total (25)	
Signature	

## **1.ABSTRACT**

Travel and Tourism industry has emerged as one of the largest and fastest growing economic sectors globally. Its contribution to the global Gross Domestic Product and employment has increased significantly. Tourism in India is a sun rise industry, an employment generator, a significant source of foreign exchange for the country and an economic activity that helps local and host communities. Rising income levels and changing lifestyles, development of diverse tourism offerings and policy and regulatory support by the government are playing a pivotal role in shaping the travel and tourism sector in India.

Social media has changed every single aspect of human lives, it has become an integral part of our everyday life. Their influence is significant, both for personal aspect and for business. The use of social networks is leading individuals and businesses towards a new era for the global economy. Social media is omnipresent. Twitter is the most popular micro blogging platform. Individuals, companies, organizations, and even governments use Twitter on a daily bases and get vast benefits from it. Twitter has been valuable for the tourism sector also, especially in developing business strategies, planning and studying tourist decision-making processes. With the growing use of social media in the tourism industry, a substantial amount of user-generated content containing tourism information and sentiment is readily available. However, a large amount of this data goes unanalyzed. This paper attempts to address this problem in the context of tweets made in India with the use of natural language processing and machine learning techniques.

### **1.1 OBJECTIVE OF THE PROJECT**

What it does?

The first task is the extraction of target data about tourism from one of the most famous

micro blogging service – twitter. In twitter we post tweets in real time. The tweets often contain significant information of events for tourism as lifelog data.

## **1.2 Information Extraction:**

### **1.2.1. Extraction of Tweets using Tweepy:**

Twitter is a popular social network where users share messages called tweets. Twitter allows us to mine the data of any user using Twitter API or Tweepy. The data will be tweets extracted from the user. The first thing to do is get the consumer key, consumer secret, access key and access secret from twitter developer available easily for each user. These keys will help the API for authentication.

Tweepy is one of the libraries that should be installed using pip. Now in order to authorize our app to access Twitter on our behalf, we need to use the OAuth Interface. Tweepy provides the convenient Cursor interface to iterate through different types of objects. Twitter allows a maximum of 3200 tweets for extraction. These all are the prerequisite that have to be used before getting tweets of a user.

### **1.2.2 Acquisition of basic queries and section of related words:**

The basic information for this process is extracted from twitter regarding tourism from tweets of people who visited the tourist spots. There might be a few issues with basic queries. Tourists do not always mention the basic queries, i.e., facility or event names, in tweets. Moreover, they might mention information which is related to the location or event names and does not appear in the portal site. Therefore, we need to acquire related words of the basic query, i.e., query expansion. So we divide each sentence into words to perform morphological analysis.

### **1.2.3 Extraction and filtering of words:**

We retrieve tweets with Twitter API by using the manually produced query list, namely

the union of basic queries and related words. Final queries for the retrieval are combination of words in the list. But the problem here is that tweets are informal and do not always contain the official name of a place. To solve this problem, we manually generate abbreviations of queries. Also, the tweets might contain noisy data. At time ‘Palani’ or ‘Tirupati’ might refer to a person rather than a town. We need to remove them. The second task is to perform sentiment analysis or feature-based opinion mining on the formed dataset and print the overall opinion of tourists who tweeted about them.

### **1.3 Sentiment Analysis:**

Sentiment analysis is the automated process of identifying and classifying subjective information in text data. This might be an opinion, a judgment, or a feeling about a particular topic or product feature. It’s also known as opinion mining, deriving the opinion or attitude of a speaker.

We plan on classifying the tweets into 2 categories – positive and negative. The words “excellent” and “poor” can be used as the seed words, and compute the semantic orientation of a phrase by using the Point wise Mutual Information (PMI) between the seed word and the phrase. The paper proposes an unsupervised machine learning approach based on Naive Bayes method as a means to classify tweets into positive or negative. The model extracts tweets containing the seed words by an exact match approach. These tweets can be used as the training set for the classifier and then the vector space can be constructed from the data.

In a nutshell, the model takes tweets as input and produces the general opinion on a tourist spot/city.

## **2.INTRODUCTION**

### **2.1 Social Media**

Tourism is a rapidly growing practice of travelling across international and national borders to obtain various objectives. Tourism is growing to be an important source of income across the globe, it is a key contributor towards the global gross Domestic Product. Tourism and urban everyday life are deeply connected in a mutually constitutive way. Tourism industry is probably most affected by the introduction of social media and micro-blogging websites. Social media has become a part and parcel of our lives today. The emergence of Social media platforms has provided a new dais for communication for both the consumers and marketers. They are now able to communicate with each other and build relationships. The use of social networks is leading individuals and businesses towards a new era for the global economy. Reason for this status of social media sites is because social networking sites like Twitter have become a preferred channel for consumers to display their feelings and thoughts through reviews, posts, tweets etc. People share their opinions, perceptions and views on various topics, current issues and also their life experiences. Due to easy accessibility of these platforms, people from diverse domain and spheres are using these platforms aggressively as compared to traditional medium.

### **2.2 Social media in Tourism**

The trend of going to internet and finding information from travel destinations, finding hotels, places to visit, and restaurants and doing bookings has become very common in the travel and tourism industry. Social media plays a predominant role not only during the planning of the trip but also during the trip as people tend to share their experiences, reviews and photos. After the trip they usually leave reviews on popular websites and social media platforms also. Social media helps the travelers to get the latest updates and current trends about the destinations, the activities they can engage in and famous

festivals they can attend which will add fun and excitement in their travel plan. With the growing use of social media in the tourism industry, a substantial amount of user-generated content containing tourism information and sentiment is readily available. However, a large amount of this data goes unanalyzed.

This data can be efficiently used to know the personality of the consumer, their past experiences, their responses to services, the places they have visited etc. Customers tend to trust their friend's and family's opinions more than that of the brand advertisements. These social media reviews by family and friends work like electronic word of mouth and become major factors during their buying decision. People even tend to change their travel plans if they get bad or negative reviews from someone they know. Hence, Consumer generated content has more significance than the brand information.

### **2.3 Sentiment Analysis**

Sentiment analysis plays vital role in the internet era due to extensive range of business applications and social media. Inspiration behind sentiment analysis is that it provides people's opinion about the product, which helps to improve the product quality. It also supports to take purchase/manufacturing decisions. This enables people to understand the customer sentiments towards their brand to better strategize their marketing plan and budget. Sentiment analysis which is also known as opinion mining is the automated process of identifying and classifying subjective information in text data. This might be an opinion, a judgment, or a feeling about a particular topic or product feature. This paper aims at classifying the tweets made by people in twitter into positive and negative based on the polarity of words. We have planned to extract tourism related tweets from the Twitter API. The tweets will be first preprocessed by performing morphological analysis, and then we abbreviate the queries which were earlier used to retrieve data from API. Sometimes the data contains unwanted or irrelevant data which

needs to be removed. Hence after performing noisy data removal we move on to the sentiment analysis phase. The words “good” and “dirty” can be used as the seed words, and compute the semantic orientation of a phrase by using the Pointwise Mutual Information (PMI) between the seed word and the phrase. This paper proposes an unsupervised machine learning approach based on Naive Bayes method as a means to classify tweets into positive or negative. The model extracts tweets containing the seed words by an exact match approach. These tweets can be used as the training set for the classifier and then the vector space can be constructed from the data. Finally, we provide an opinion based on the sentiment score of the data.

### **3.PROBLEM STATEMENT**

Usually when people travel to a particular tourist spot they cast their opinion about the place in twitter in form of tweets. Most people find twitter to be an effective platform where they can cast their opinion without any hinderance. These tweets can be effectively analyzed to deduce a general review about the particular tourist spot so that they can be useful to the tourism recommendation system and to confused users who are unable to decide on a place to visit to come to a wise conclusion.

In this paper we aim at classifying the tweets made by people in twitter into positive and negative based on the polarity of words. And finally generate a opinion based on these tweets. The proposed model includes extraction of tweets from the Twitter API using tweepy, performing sentiment analysis on the dataset and finally generating an opinion about the particular tourist spot.

### **4.LITERATURE SURVEY**

#### **4.1.1 Papers [1,2]: Tourism Recommendation System**

These papers have a good explanation about Vector Space Generation and Naïve Bayes Classification. The model is trained using tweets containing predefined seed/keywords.

The vector space is generated from training the model. The Naive Bayes Algorithm is going to be applied to the vector space for positive-negative classification. The same methods are going to be adopted for our model. Data Filtration methods were not very clear. The number of training inputs were not sufficient hence the accuracy of Naïve Bayes classification was low. The authors have explained about the various criteria to consider while collecting data for the project. The Tourism Contextual Information consists of information about tourists, tour destination, location, time, social and weather. It explains about the literature-based study that was conducted by collecting, reviewing and analyzing the topic related to the tourism recommendation system. These papers have explained in detail how to effectively integrate and distribute heterogeneous data sources i.e. the data coming from any source and in the form of various format. However, the authors have collected data based on six factors only which limits the correctness of the suggestion provided to the users.

#### **4.1.2 Papers[3,4]:Opinion Mining**

In these papers, they've reviewed various significant sentence-based opinion mining techniques. The major focus of this is tourist places sentences extraction and tourist places sentences classification techniques. The authors have proposed a tourism information analysis system that performs analysis on the tweets and then divides them on the basis of positive and negative sentiments. System firstly takes tweets from the twitter as input. Secondly, extracts the tourism related tweets and discards the remaining ones. Thirdly, apply sentiment analysis on extracted tweets. The strengths of these papers are that the proposed method applies post-processing on the tweets before applying sentiment analysis. As a result tourism related tweets separate to the other remaining tweets. The classification of tweets into positive and negative sentiments related tweets has accuracy of approximately 0.92 that is nearly the human ability. The authors have clearly explained about the usage of feature-based opinion mining, where features refer to the categories of opinions and opinion mining is the process of



classifying them into different scales. Features and polar words are used as the lexicons and they have extracted these from the opinion text based on syntactic pattern analysis and by calculating the sentiment scores. However they have done opinion mining based on the feedback a hotel collects from their customer which maybe limited and which may not be true as the customer might not express his/her true opinion. Also, the method extracts some irrelevant sentences from the tourist tweets that do not have not any relationship with the targeted tourist place and cause to noise in tweets classification.

#### **4.1.3 Paper[5]:Sentiment Analysis**

Sentiment analysis of tweets using Naive Bayes can be extended to any review related website for example product review to understand products popularity , movie review etc.. It can also be highly useful in sub component technology such as detecting antagonistic, heated language in mails, context sensitive information detection, spam detection etc. However, determining consumer attitudes and trends is one of the major applications of sentiment analysis of data. So this sentiment analysis model has a flaw that it takes a lot of time in fetching data from twitter and in data management.

#### **4.2. Summary:**

In the first paper they have focused on an unsupervised machine learning approach, which does not need manually annotated training data. It is based on seed words and pseudo training data extracted from a non-tagged corpus. They have used naive bayes method as a means to classify tweets into positive or negative. The naive bayes classifier generates the model from the vector space.

The second paper has explained in detail how to effectively integrate and distribute heterogeneous data sources i.e. the data coming from any source and in the form of various format. It explains about the literature-based study that was conducted by

collecting, reviewing and analyzing the topic related to the tourism recommendation system.

The positive aspect of the third paper is that the proposed method applies post-processing on the tweets before applying sentiment analysis. As a result tourism related tweets separate to the other remaining tweets. The classification of tweets into positive and negative sentiments related tweets has accuracy of approximately 0.92 that is nearly the human ability.

In the fourth paper the authors have clearly explained about the usage of feature-based opinion mining, where features refer to the categories of opinions and opinion mining is the process of classifying them into different scales. Features and polar words are used as the lexicons and they have extracted these from the opinion text based on syntactic pattern analysis and by calculating the sentiment scores.

The fifth paper clearly explains the implementation of Naïve Bayes for sentiment analysis. This paper mentions that this method can be extended to any review related website for example product review to understand products popularity , movie review etc.Determining consumer attitudes and trends is one of the major applications of sentiment analysis of data

## **5.ISSUES IDENTIFIED**

The varies issues that were identified from the IEEE base and reference papers were listed above. Each paper had a different kind of issue, for instance ,in the base paper 1, the data Filtration methods were not mentioned clearly.Another paper had a disadvantage that they have performed sentiment analysis on the reviews collected from the customers in their official portal. Oftentimes, these reviews won't be the real reflection of what the customer might feel.

We have tried to overcome the issues as follows,

## **5.1 Data filtration is done in multiple steps:**

### **5.1.1. Breaking each tweet in words and performing Morphological Analysis.**

Tweets that contain unwanted information are discarded. Example: I am waiting at a bus stop in Ooty. Tweets like these do not contain any information regarding tourism. These are removed. The “name” might not always refer to a tourist location. Example: Palani might also refer to a person. In such cases remove that particular tweet.

### **5.1.2. The sizes of the training and the testing datasets were increased.**

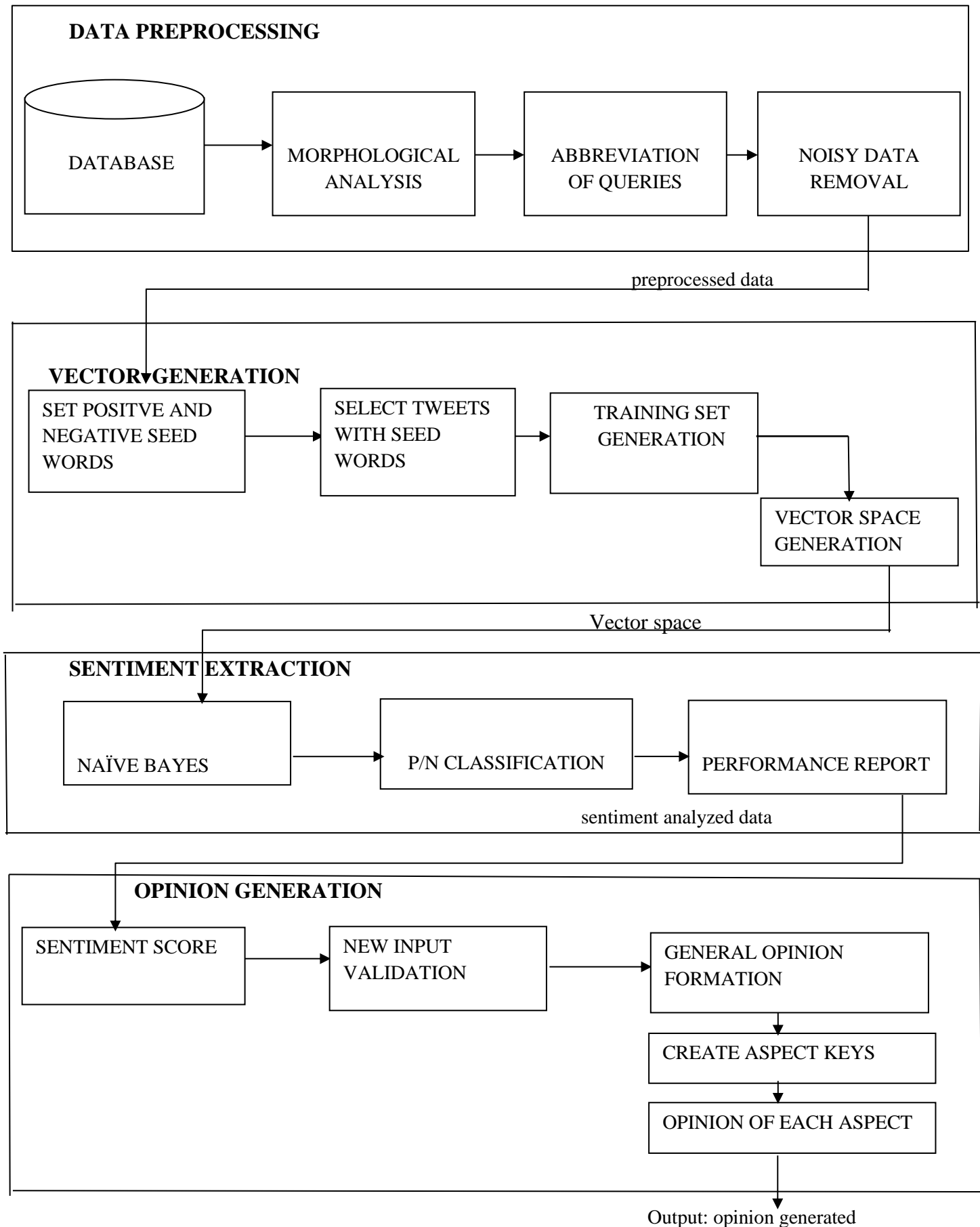
### **5.1.3 Using tweets from twitter API**

Instead of performing sentiment analysis on reviews collected by a hotel, we have performed them on tweets collected from the Twitter API. The twitter is considered as the voice of the people as most people find it comfortable to cast their opinions in twitter.

### **5.1.4. To resolve the issue of unwanted tweets** that were collected, we done noisy data removal using Regex.

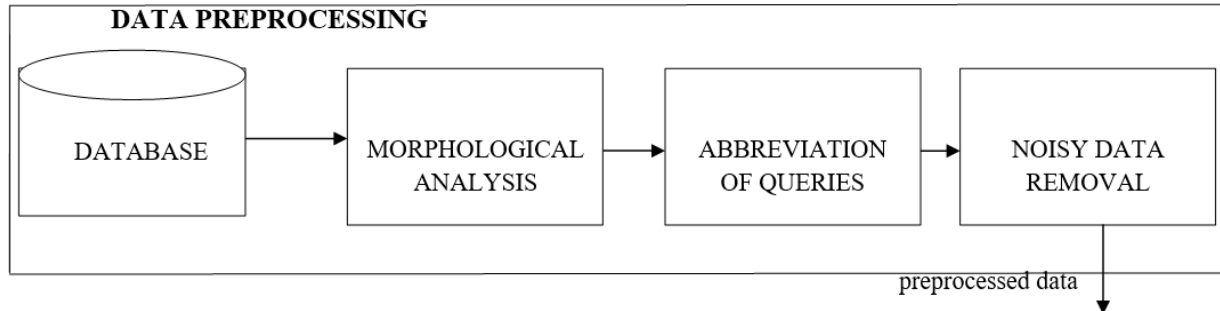
We have also incorporated the idea of classifying tweets based on the score of polar words in this paper. All polar words that are indicated by positive and negative will be mapped with a score. The score of polar words is used in statistical analysis for opinion mining.

## 6.ARCHITECTURE BLOCK DIAGRAM



## 7.MODULES WITH I/O

### 7.1.Data pre-processing



**7.1.1.Explanation:**First module in our project's block diagram is **Data preprocessing**. We have done data extraction from the Twitter API and then filter it by tokenizing it and using regex expression. Here, Tokenization refers to the process of splitting the tweets into smaller units, such as individual words or terms. Each of these smaller units are called tokens.

#### Data Extraction

We have extracted Tweets pertaining to various famous tourist spots across India from the twitter API using Tweepy code in python. First we have imported the necessary python scripts and then used the Twitter API Authentication details to extract the tweets.

- We have used several parameters in order to obtain the most suitable dataset for our project. The StartDate corresponds to starting date range of the tweets i.e we have collected all tweets starting from 2008-01-01 till now.
- The places array is the manually prepared query list i.e and is used to collect tweets which contain the keywords mentioned in this array.
- We have restricted to collect the retweeted tweets by using a filter.

The output of the first stage is Noisy data from Twitter API in csv file format.

#### Example:

One of the cities that we considered is shimla, the various tourists spots that we included in shimla were kulfri, shimla mall and himachal Pradesh kalpa. These keywords were identified as a part of identification of basic queries.

## Noisy data removal

- Next we have performed Data filtering in order to remove the unwanted and irrelevant tweets. The tweets from Twitter usually consist of hash tags, mentions, emojis and other symbols, URLs and text in regional language as well. Noisy data are unwanted to our project and removing them is very important. Hence by filtering the data using Regex we have removed such unwanted data.
  - The Regex is used to remove the hashtags, spaces, URLs and other unwanted symbols.
  - We have used tokenization to remove unwanted tweets. An array named stop containing the unwanted words to be removed was declared.
  - First we filter the tweets that contain words specified in the 'stop' array are removed. Then we again filter these tweets based on their length. The terms with length > 3 are only retained. The tokenized tweets are then printed.
  - Finally we store the cleaned tweets in a new csv final. This forms the final dataset.
- We have shown the tabular format for the processes we have done namely getting details about a place, extracting corresponding tweets from Twitter API, tokenization and noisy data removal.

### 7.1.2.Pseudo Code:

#### Get tweets:

**Input:** Tweets from Twitter API( extracted using tweepy)

**Output:** Cleaned Tweets

Authenticate keys

Get API connection

Get PlaceName from user

Get StartDate from user

Set places list

Ask if want to add spots in PlaceName

If PlaceName.csv exist :

```
Index = len(PlaceName.csv)
Else
    Index = 0
If yes
    Ask for number of spots n
    Get n spots
```

```
Append to places list
Open PlaceName.csv file
for i = index to places.length:
    Tweet = Get english tweets with places[i] in it
    If tweet is not retweeted:
        Append to PlaceName.csv
```

### **Clean Tweets:**

```
function clean_tweets(tweets)
    tweets = tweets - urls
    tweets = tweets - '@'
    tweets = tweets - 'RT'

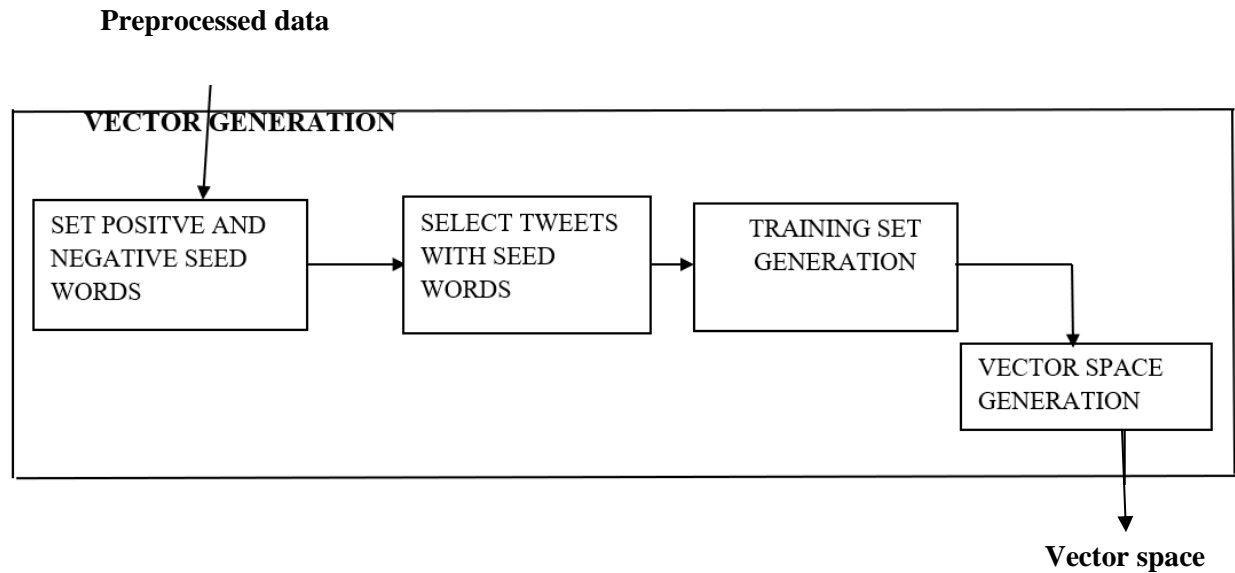
    wordlist.extend(tokenizer.tokenize(tweet))
    set stopwords
endfunction
newFile = "cleaned"+PlaceName.csv

csvFile = open(newFile)

csvWriter = csv.writer(csvFile)
myFile = openRead("example.txt")

while NOT myFile.endOfFile()
    csvreader = csv.reader(myFile)
    For each row in csvreader:
        String clean_tweets(row)
        csvWriter.writerow([string])
    endwhile
myFile.close()
```

## 7.2: Vector Space Generation



**7.2.1.Explanation:** The second module in this paper is Vector Generation. After extracting tweets from the Twitter API and after pre-processing the same using regex we have used the dataset in the second module to generate vectors. The goal of the second module is to find the training dataset. We have identified our training data set by setting **‘seed words’** and we have proceeded by generating the vector space for the dataset.

As we are performing sentiment analysis, to prepare an ideal dataset we have collected both positive and negative tweets about the tourist places in equal proportion. The seed words are the basis of classification. We have given a list of the positive seed words and negative seed words in separate arrays and the classification of the cleaned dataset into positive and negative happens on the basis of the words in these two arrays.

Some of the **positive seed words** that we have used include **‘good’**, **‘amazing’**, **‘beautiful’**, **‘serene’**, **‘love’** and **‘enjoy’**. Some of the **negative seed words** that we have used include **‘bad’**, **‘worst’**, **‘dirty’**, **‘unsafe’**, **‘unworthy’**, **‘horrible’** and **‘hate’**.

Processing natural language text and extracting useful information from a given word or a sentence using machine learning and deep learning techniques requires the string/text to be converted into a set of real numbers (a vector). This process is referred to as word embeddings



or word vectorization. These are then used to find word predictions, word similarities/semantics etc. Simply put, the processes of converting words into numbers are called Vectorization.

There are many methods that can be applied to perform vectorization. We have used the TF-IDF vectorizer. **Tf-idf** stands for Term frequency-inverse document frequency. Tf-idf is a weighting scheme that assigns each term in a document a weight based on its term frequency (tf) and inverse document frequency (idf). The terms with higher weight scores are considered to be more important.

### **7.2.2.Pseudo Code:**

**Input:** Pre-processed tweets

**Output:** Vector Space

**Set seed words**

**Set positive seed array**

**Set negative seed array**

```
function positive_tweets(tweet):  
    for i from 0 to positive_seed.length()  
        if tweet has positive_seed[i]  
            return tweet  
    return “ ”  
endfunction
```

```
Function negative_tweets(tweet):  
    for i from 0 to negative_seed.length()  
        if tweet has negative_seed[i]  
            return tweet  
    return “ ”  
endfunction
```

**Vector Space Generation:**

```
function vector_space(wordlist)  
    rows, cols = (2,len(wordlist))  
    m = [[1 for i in range(cols)] for j in range(rows)]
```

```

for i in range(0,int(len(wordlist))):
    if( positive_seed has wordlist[i]) :
        m[0][i] = 1
        m[1][i] = 0
    else if ( negative_seed has wordlist[i]):
        m[0][i] = 0
        m[1][i] = 1
    else:
        m[0][i]=0.5
        m[1][i]=0.5

```

```

pd.DataFrame(m, columns=wordlist)
tf = TfidfTransformer()
m = tf.fit_transform(m).todense()
Print vector space

```

### **Separating positive and negative tweets:**

Ask for filename

Get input as n

If n>=0

    Newfile1 = positive+filename

Else

    Newfile1 = negative+filename

csvFile = openRead("Newfile1.csv")

csvWriter = csv.writer(csvFile)

csvreader = csv.reader(csvFile)

row = myFile.readLine()

tokenizer = RegexpTokenizer(r'#\w+')

wordlist = [ ]

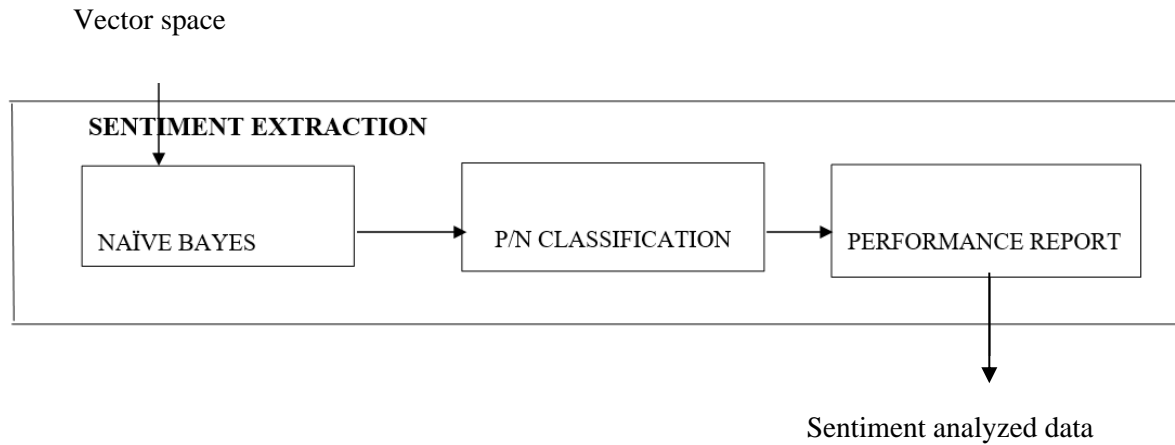
wordlist.extend(tokenizer.tokenize(str(row)))

for w in wordlist :

    words.append(w)

vector\_space(words)

### 7.3.Sentiment Extraction



The third module in this paper is sentiment extraction and it focuses on Naive Bayes Classifier and PN Classification.

#### **Naive Bayes Classifier:**

The tweets of all tourist places that were earlier split into positive and negative in the previous module were saved in a new file. We have used +1 to denote positive tweets and -1 to denote the negative tweets. Naive Bayes Classifier cannot understand text format. So, the tweets in text format are converted to a matrix of TF-IDF features. We have used TfidfVectorizer which gives a better accuracy. The dataset is split into training and testing sets (X\_train, X\_test, y\_train, y\_test). The testing is 20 percent of the entire dataset.

Naive Bayes classifiers are a collection of classification algorithms based on Bayes' Theorem. It is not a single algorithm but a family of algorithms where all of them share a common principle, i.e. every pair of features being classified is independent of each other. Naive Bayes classifier is used in Text Classification, Spam filtering and Sentiment Analysis. It has a higher success rate than other algorithms.

#### **Naive Bayes Theorem:**

$$P(A|B) = P(B|A) P(A)/(P(B))$$

There are various types of Naive Bayes Classifier like Bernoulli Naive Bayes, Gaussian Naive Bayes, Multinomial Naive Bayes.

We have trained the model using **MultinomialNB**. The multinomial Naive Bayes classifier is suitable for classification with discrete features. This works well for data which can easily be turned into counts. Multinomial Naive Bayes classifiers has been used widely in NLP problems compared to the other Machine Learning algorithms, such as SVM and neural network because of its fast learning rate and easy design. In text classification these are giving more accuracy rate despite their strong naive assumption.

We have imported the inbuilt classifier from sklearn.

### 7.3.2 Pseudocode:

**Input:** Training and testing sets

**Output:** Trained Naive Bayes Classifier(sentiment analyzed data)

**Combine all datasets so that it would be easier to feed to the model**

**Naive Bayes:**

```
df1 = pd.read_csv('datasets.csv')
```

```
X = df1[df1.columns[0]]
```

```
y = df1[df1.columns[1]]
```

Apply TfidfVectorizer

Transform vectorized data

Split into training and testing sets = X\_train, X\_test, y\_train, y\_test

#Apply Naive Bayes

```
model_naive = MultinomialNB().fit(X_train, y_train)
```

Predict for testing set

#Performance metrics

Print confusion matrix

Print accuracy

Print classification report

Test model for new Tweets:

Input: Tweets in text format

Output: PN classification

```
df2 = pd.read_csv('testinputs.csv')
```

```
X_pred = df2[df2.columns[0]]
```

```
temp_X = X_pred
```

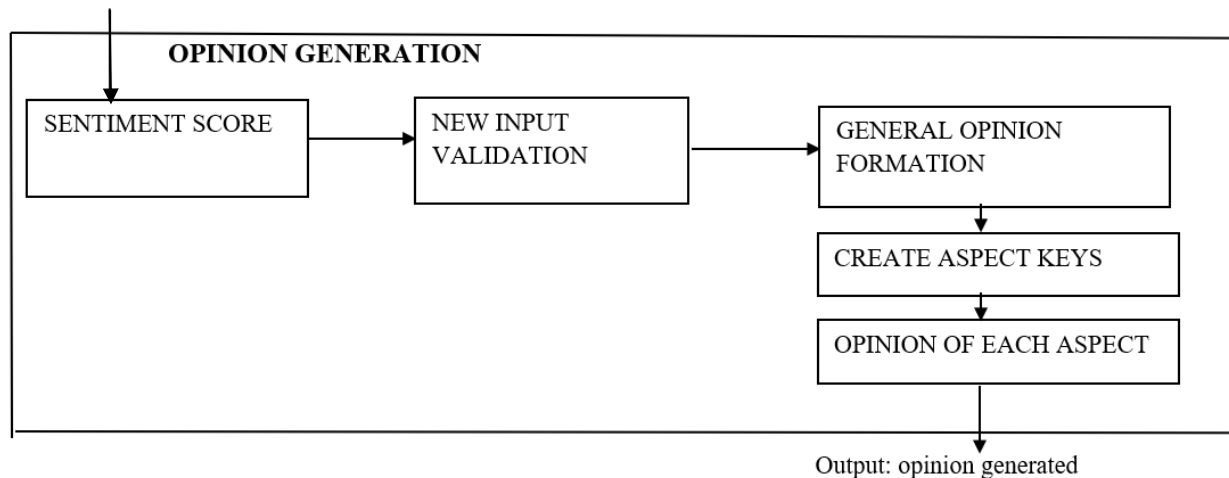
```

Transform new inputs <= final_vector_data
Predict for final_vector_data <= new_output
For i from 0 to new_output.length
    If new_output = 1
        print('Positive')
    Else
        print('Negative')

```

#### 7.4. Opinion Generation(Innovative Module) :

sentiment analyzed data



The main purpose of this module is to calculate the sentiment score ,validate the user inputs and generate summary for the tourist spots.

##### Sentiment Score Calculation:

In this module we have calculated the rating from the sentiment analysis done in the previous modules,. The calculation is simple. Count the number of positive tweets for a place. Sentiment score = number of positive tweets for a place/ total number of tweets for that place rating = round(Sentiment Score)/10

A place is considered as a 'good' place if it has a rating in the range 8-10.

A place is considered as an 'average' place if it has a rating in the range 6-8

A place is considered as a 'poor' place if it has a rating below 5.

## **Input Validation:**

A user can also add their comments about any of the tourist spots. The opinions are first validated to check if they can be helpful to the system. The new command line input is appended to testinputs.csv file. The new inputs are then vectorised and transformed and fed to the model. They are analysed as positive or negative. The input and their sentiment are then appended to the particular place's dataset. The model has correctly identified the sentiment of the new inputs.

## **Summarisation of tweets:**

In this module we have also generated a small summary of the positive and negative aspects of each place.

The summary has been generated using T5 Summarisation. T5 can take input in the text format and generate a summary of the input. T5 is an abstractive summarization algorithm. It can rephrase sentences or use new words to generate the summary. The data augmentation technique is useful for NLP tasks involving long text documents. Since the number of tweets are more, T5 Summariser is much more efficient. Both the positive and the negative aspects of each spot have been generated. The tweets are read from the respective CSV files, fed to the summarizer, cleaned and then printed.

Now to be even more specific, we have created a dictionary with 4 keys: **‘transportation’** , **‘safety’** , **‘clean’** , **‘local’** to give users a much more detailed and clear review about a given place. The user can get the summary by entering the place name and one of the key names.

T5 summarizer has been used to generate the summary. T5 can take input in the text format and generate a summary of the input. T5 is an abstractive summarization algorithm .It can rephrase sentences or use new words to generate the summary. The data augmentation technique is useful for NLP tasks involving long text documents.

Since the number of tweets are more, T5 Summariser is much more efficient. Both the positive and the negative opinion of each aspect of each spot has been generated. The tweets are read from the respective CSV files, formed the dictionary, fed to the summarizer, cleaned and then printed.

#### **7.4.2.Pseudo code:**

**Input:** Sentiments Analysed data

**Output:** Sentiment Score and Input Validation

##### **Calculate Sentiment Score:**

```
Toursit_places[0] = mumbai'  
Toursit_places[1] = 'northeast'  
Tourist_places[2] = 'kashmir'  
Toursit_places[3] = 'ladakh'  
Toursit_places[4] = 'kerala'  
Toursit_places[5] = 'delhi'
```

```
For i from 0 to tourist_places.length  
    df1 = read_csv(tourist_places[i]+'dataset.csv')  
    X = df1[df1.columns[0]]  
    y = df1[df1.columns[1]]  
    pos = 0  
  
    for j in range(0, X.length) :  
        if(y[j]=1) :  
            pos+=1  
  
senti_score <= (pos/X.length*100)  
rating+= senti_score rounded to 2 decimal places  
Sort placenames and ratings according to senti_score
```

##### **Find rating of places:**

```
Ask for placename input  
Check if valid input  
If rating[placename] > 8  
    Print great tourist spot  
Else if rating[placename] > 5  
    Print average tourist spot
```

Else

Print poor tourist spot

### **Validating user input:**

Function clean\_tweets(tweet)

if(tweet has @, # , url)

Remove them

Set stop words

Tokenize tweets and store in wordlist

if wordlist.length<2

Print 'Tweets are not related to tourism'

Exit

For i from 0 to stopwords.length()

If tweets has stopwords[i]

Print 'Tweets are not related to tourism'

Exit

Print 'Thankyou for your opinion'

Return tweet

Endoffunction

Set allowed\_places

Created a dictionary of allowed places and their tourist spots

Keep placename input

Tweet <=Get input opinion

Validate opinion clean\_tweets(tweet)

if(not valid) :

exit(1)

csvFile = openRead("testinput.csv")

csvWriter = csv.writer(csvFile)

csvWriter.write(testinputs.csv)

**Input :** New Tweet(Text)

**Output:** Sentiment Analysed text

### **Predict Sentiment of user input:**

Transform user input

Send to naive bayes model

if(sentiment == 1)

Positive opinion

Else

Negative opinion

Add tweet and sentiment in dataset



**Input:** Tweets

**Output:** Positive and Negative summary of the given place

**General Summary:**

Get input for PlaceName

Collect positive tweets of PlaceName

Create T5 summarizer model

Feed tweets to model

Summary\_p = Generate model summary with unwanted symbols

Collect negative tweets of PlaceName

Create T5 summarizer model

Feed tweets to model

Summary\_n = Generate model summary with unwanted symbols

```
function remove_unwanted(tweets)
```

```
    Tweet.remove('<extra_id_1>')
```

```
    tweet.remove('<pad>')
```

```
    tweet.remove('</s>')
```

```
    Return tweet
```

```
Final_p = remove_unwanted(Summary_p)
```

```
Print Final_p
```

```
Final_n = remove_unwanted(Summary_n)
```

```
Print Final_n
```

**Input:** Place name and aspect

**Output:** General summarised opinion

**Specific Aspect Summary:**

Get PlaceName

Set transport\_keys []

Set safety\_keys []

Set clean\_keys[]

Set local\_keys[]

```
Function collect_opinion(PlaceName, arr,flag )
```

```
    sign = ""
```

```
    if flag == -1:
```

```
        sign = "negative"
```

Else:

```
        sign = "positive"
        opinion = ""
        with openRead(sign+'cleaned'+PlaceName+'.csv') as csvfile:
            csvreader = csv.reader(csvfile)
            for each row in csvreader:
                for i from 0 to arr.length:
                    if arr[i] in row[0]
                        opinion += str(row[0])
                        opinion += ' '
                        continue
        return opinion
Endoffunction
```

Function generate\_summary(tweet\_data):

```
    Create T5 summarizer model
    Feed tweets to model
    Summary = Generate summary with unwanted symbols
    Summary = remove_unwanted(summary)
```

```
    function remove_unwanted(tweets)
        Tweet.remove('<extra_id_1>')
        tweet.remove('<pad>')
        tweet.remove('</s>')
        Return tweet
```

Endoffunction

Input category

```
Generate (positive_dict[category] )
Generate(neagative_dict[category])
```

```
positive_dict = {
    "transport" : collect_opinion(PlaceName,transport_keys,1),
    "safety" : collect_opinion(PlaceName,safety_keys,1),
    "sight" : collect_opinion(PlaceName, sight_keys,1),
    "clean" : collect_opinion(PlaceName, clean_keys,1),
    "local" : collect_opinion(PlaceName, local_keys,1)
}
```

```
negative_dict = {  
    "transport" : collect_opinion(PlaceName,transport_keys,-1),  
    "safety" : collect_opinion(PlaceName,safety_keys,-1),  
    "sight" : collect_opinion(PlaceName, sight_keys,-1),  
    "clean" : collect_opinion(PlaceName, clean_keys,-1),  
    "local" : collect_opinion(PlaceName, local_keys,-1)  
}
```

## 8.INNOVATION

None of our papers have any information about generating reviews. All papers have information only till classification of tweets. As per our base and reference papers, we have classified the tweets extracted from the Twitter API into positive and negative categories. Moreover, all the papers focused only on classification of tweets for one city. We have done it for multiple locations.

As an innovation, we are also getting inputs from the user and appending them to the corresponding spot's CSV data file and classifying them into positive and negative categories. Then, as another innovation, from the sentiment analysis we have done for all the tourist spots we are calculating the sentiment score.

Sentiment score is the rounded off likeness of a place divided by 10. A place is considered to be a 'good' place if its sentiment score is in the range 8-10. A place is considered as an 'average' place if its sentiment score is in the range 6-8. A place is considered as a 'bad' place if its sentiment score is below 6.

Based on the sentiment score and the tweets, the model generates an overall opinion for the tourist spot. So, when an end user enters a tourist spot as an input, they get the general public opinion on it. Further, we have added a specific review about certain features in the dataset, we have considered cleanliness, safety, local ambience and transportation and if a user specifies a particular feature and a spot, the review is generated. Thus apart from sentiment analyzing the tweets from Twitter API and user inputs we are calculating the sentiment score of the tourist spots and generating an overall review for the tourist spots.

Our model with all the innovations we have made can be of immense use to the tourism recommendation system. It'll also help confused people who aren't able to decide on the tourist place to visit to come to a wise conclusion. **These innovations are handled in module 4.**

## 9.DATA DESCRIPTION

### 9.1.DATASET DETAILS

<b>DATASET NAME</b>	<b>CUSTOM DATASET (TOURISM RELATED TWEETS FROM TWITTER API)</b>
<b>INPUT DATASET COLLECTED</b>	<b>TOURISM RELATED TWEETS FOR THE FOLLOWING TOURIST SPOTS:</b>  <b>1.Mumbai</b> <b>2.Delhi</b> <b>3.Kashmir</b> <b>4.Kerala</b> <b>5.NorthEast India</b> <b>6.Ladakh</b>
<b>TYPE OF DATA</b>	<b>TEXT</b>
<b>SIZE OF THE DATA COLLECTED (CSV file size)</b>	<b>1.Mumbai - 7KB</b>  <b>2.Delhi - 10KB</b>  <b>3.Kashmir – 10KB</b>  <b>4.Kerala – 9 KB</b>  <b>5.Ladakh – 7 KB</b>  <b>6.NorthEast India – 7KB</b>
<b>SOURCE OF DATA</b>	<b>TWEETS EXTRACTED FROM THE TWITTER API</b>
<b>DURATION FOR WHICH THE DATA HAS BEEN COLLECTED</b>	<b>FROM 2008-01-01 - PRESENT</b>

## 9.2.Data Extraction:

The tweets regarding 6 tourist spots - **Mumbai, Delhi, Kashmir, Ladakh, Northeast India, Kerala were extracted.** The tweets are extracted using Tweepy. Tweepy is an open source Python package that gives a very convenient way to access the Twitter API with Python.

Twitter API requires that all requests use OAuth to authenticate. So we need to create the required authentication credentials to be able to use the API. These credentials are four text strings: Consumer key, Consumer secret, Access token, Access secret. Hence to access these keys, we have created a developer's account. The cursor method has been used for extracting the tweets. Using cursor, we can specify attributes like the search words, date since and date until which represent the time interval of the tweets that can be extracted. The language was set to "en" to extract only the tweets in english language.

The start date was specified as 2008-01-01. This means that the tweets tweeted from 1st January 2008 related to tourism were collected. In order to collect only useful tweets that are related to travel and tourism and leave out the unwanted information, few keywords like "tourism", "beautiful", "amazing", "dirty", "experience" were used. These words are often used by travelers to describe their visit to any tourist place. To make the tweet extraction more accurate, tourist spots in PlaceName were also entered. The algorithm works in such a way that any sentence(tweets) which contain the PlaceName and any of the keywords or the spot names as a subsequence will be extracted.

But twitter has a feature of retweeting - i.e. reposting a tweet. Collecting the same tweets more than once leads to redundancy. Hence we have removed all the retweets retaining only the original tweet.

## 9.3.Cleaning the extracted tweets:

But even these extracted tweets contain many symbols like #, URLs, emojis, mentions etc. These symbols are removed using Regular Expressions. After removing them, the tweets only contain plain texts. But these tweets can still be unrelated to tourism. Hence they are removed

using stopwords. Stopwords is an array whose elements are words like “election” , “protest” , “campaign” , “congress” , “BJP” and 15 other words. Since these words are not related to tourism, any tweet which contains at least one of the stopwords are not taken into further consideration and are discarded. The remaining ‘cleaned tweets’ are definitely related to tourism and are now stored in a new file names - cleanedPlaceName.csv.

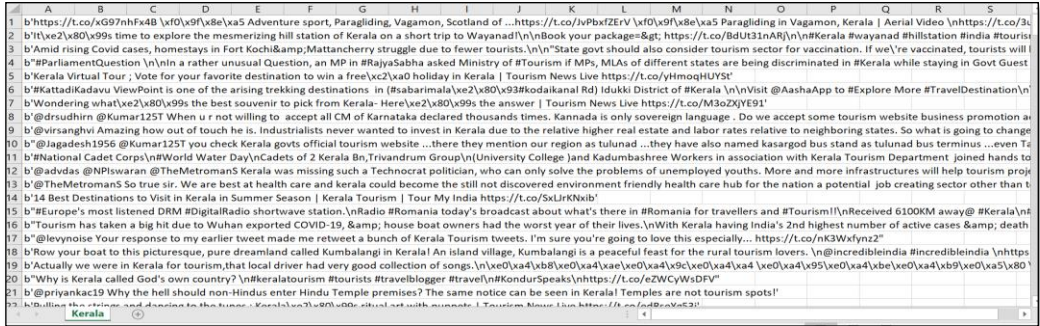
The above processes were repeated for all the 6 considered tourist places.

#### **9.4.Features or attributes used:**

The only attribute in our dataset is the tweet tweeted by the user. The main aim of our project is to analyze the sentiment of people about a tourist spot and generate a summary based on that. In Order to do so, all the required data can be obtained from the tweets alone. Hence a minimum of 50 tweets are extracted for all the 6 considered tourist places.

## 10.RESULTS AND INTERMEDIATE RESULTS OF EACH MODULES

### 10.1:DATA PREPROCESSING

Input	Output
<b>1. Place details of Kerala</b>	<div>Enter the spot you want the tweets to be downloaded for: kerala</div> <div>Enter the start date in this format yyyy-mm-dd: 2008-01-01</div> <div>Enter number of places in kerala 3</div> <div>Enter places in kerala</div> <div>Alleppey</div> <div>munnar</div> <div>kovalam</div>
<b>2.Noisy data</b>	
<b>3.Tokenization</b>	<pre>[ 'Beautiful', 'location' ] [ 'Munnar', 'Cool', 'Classic', 'Heaven' ] [ 'just', 'Munnar', 'this', 'district', 'Kerala', 'many', 'touristy', 'gems' ] [ 'Munnar' ] [ 'Munnar', 'Plantation', 'Truly', 'Delightful' ] [ 'just', 'Munnar', 'this', 'district', 'Kerala', 'many', 'touristy', 'gems' ] [ 'delighted', 'that', 'enjoyed', 'your', 'trip', 'experience', 'staying', 'County', 'Thank', 'words', 'appreciation', 'regarding', 'resort', 'x99s', 'ambience', 'hospitality', 'look', 'forward', 'seeing', 'again' ] [ 'Vishram', 'Business', 'Review', 'been', 'quite', 'some', 'time', 'since', 'traveled', 'Hyderabad', 'last', 'went', 'Kochi', 'Munnar' ] [ 'beautiful', 'natural', 'surroundings', 'plan', 'your', 'next', 'getaway', 'with' ] [ 'Munnar', 'hill', 'station', 'India' ] [ 'Nature', 'Zone', 'Resorts', 'Munnar', 'perfect', 'blend', 'nature', 'wildlife', 'Incorporating', 'Tree', 'House', 'Munnar', 'Jungle', 'Resorts', 'ideal', 'spot', 'connect', 'with', 'nature', 'serene', 'beauty', 'Want', 'enjoy', 'stay', 'here' ] [ 'BOOKING', 'nCoimbatore', 'Ooty', 'Kodaikanal', 'Munnar', 'Alleppey', 'Drop', 'Cochin', 'Coimbatore', '7N8D', 'Trips', 'n1Couple s', 'nVehicles', 'Accommodation', 'House', 'Boat', 'nBreakfast', 'nsouthtoursbe', 'nWhatsApp', 'Call', 'n08682070707', 'n08682848586' ]</pre>
<b>4.Cleaned data</b>	<pre>122: build more resorts churches Munnar other hilly 123: Catch 124: Summer rains interior Kerala slowly picking readings till 49mm nMunnar 16mm 125: Munnar heaven used cool place anymore Environmental encroachment killed beauty Still Keralite would like invite tourists have scenic areas still left 126: Easy Hill Munnar Kerala 127: Wild Elephant Ecofriendly Resort Munnar Pool view early morning 128: Rather Kamaraj order retain Nadar dominant Kanakumari overhance Munnar Other nart Idukki dist which Tamil majority reelin with wa resources</pre>



Place Name	Cleaned Tweets stored in csv file
Kerala	Really bad day.So sad.Ws disaapointed with boat house stay
	your boat this picturesque pure dreamland called Kumbalangi Kerala island village Kumbalangi peaceful feast rural tourism lovers
	Beach Tourism Beaches Kerala Beautiful
	Kerala is a Overhyped place.

## 10.2: VECTOR GENERATION

### INPUT TABLE:

Enter spot name	Enter any positive number for Positive classification or negative number for negative classification
cleanedkashmir.csv	1
cleanedkashmir.csv	-1
cleanedladakh.csv	1
cleanedladakh.csv	-1
cleanedshimla.csv	1
cleanedshimla.csv	-1
cleanedmumbai.csv	1
cleanedmumbai.csv	-1

## **OUTPUT (POSITIVE AND NEGATIVE CSV FILES):**

### **KASHMIR:**

#### **Positive:**

gulmarg trekking! amazing experience... pumped with adrenaline
gulmarg - perfect skiing location.
Srinagar is a heaven on earth.. So beautiful!
visited dal lake in srinagar. Such a soothing and calm place. So peaceful
Love from core heart from paradise Kashmir

#### **Negative:**

interior kashmir not safe for tourists.
india pok border unsafe
pulwama is filled with terrorists
landslides in gulmarg leave people stranded. No food or water.

### **MUMBAI:**

#### **Positive:**

stunning and beautiful woodwork bhaja caves near lonavala mumbai tourism marine drive serene,quiet and blissful place suitable for a morning walk
mumbai is a city of dreams and wonderful tourist spots
street food is at its best at mumbai streets I love Mumbai
gateway of India is beautiful
gateway of India great historical monument good architecture

## Negative:

mumbai traffic makes it one of the worst cities
elephanta caves at mumbai are not maintained properly and are untidy and worst nowadays
mumbai traffics makes it unfit and unworthy of living
dharavi slum is one of the worst living spaces in the world
street food at mumbai are poorly maintained and disgusting at times they cause health problems for tourists

## VECTOR SPACE GENERATION:

### INPUT

Enter spot name
positivecleanedkashmir.csv
negativecleanedkashmir.csv
positivecleanedladakh.csv
negativecleanedladakh.csv
positivecleanedshimla.csv
negativecleanedshimla.csv
positivecleanedmumbai.csv
negativecleanedmumbai.csv
positiveCleanednortheast India.csv
negativeCleanednortheast India.csv

## VECTOR SPACE:

All positive words are given numerical values above 0.5 and all negative words are given values less than 0.5.

## KASHMIR POSITIVE:

	gulmarg	trekking	amazing	experience	pumped	with	adrenaline	\
0	0.025120	0.025120	0.070611	0.025120	0.025120	0.025120	0.025120	
1	0.032965	0.016482	0.000000	0.032965	0.032965	0.032965	0.032965	
	gulmarg	perfect	skiing	location	Srinagar	is	a	\
0	0.025120	0.070611	0.025120	0.025120	0.025120	0.025120	0.025120	
1	0.032965	0.000000	0.032965	0.032965	0.032965	0.032965	0.032965	
	heaven	on	earth	So	beautiful	visited	dal	\
0	0.025120	0.025120	0.025120	0.025120	0.070611	0.025120	0.025120	
1	0.032965	0.032965	0.032965	0.032965	0.000000	0.032965	0.032965	
	lake	in	srinagar	Such	a	soothing	and	\
0	0.025120	0.025120	0.025120	0.025120	0.025120	0.025120	0.025120	
1	0.032965	0.032965	0.032965	0.032965	0.032965	0.032965	0.032965	
	calm	place	So	peaceful	have	amazing	spring	\
0	0.070611	0.025120	0.025120	0.070611	0.025120	0.070611	0.025120	
1	0.000000	0.032965	0.032965	0.000000	0.032965	0.000000	0.032965	

## KASHMIR NEGATIVE:

	nCourtesy	interior	kashmir	not	safe	for	tourists	\
0	0.053482	0.053482	0.053482	0.053482	0.053482	0.053482	0.053482	
1	0.055457	0.055457	0.055457	0.055457	0.055457	0.055457	0.055457	
	india	pok	border	unsafe	pulwama	is	filled	\
0	0.053482	0.053482	0.053482	0.000000	0.053482	0.053482	0.053482	
1	0.055457	0.055457	0.055457	0.077943	0.055457	0.055457	0.055457	
	with	terroists	landslides	in	glumarg	leave	people	\
0	0.053482	0.000000	0.053482	0.053482	0.053482	0.053482	0.053482	
1	0.055457	0.077943	0.055457	0.055457	0.055457	0.055457	0.055457	
	stranded	No	food	or	water			
0	0.000000	0.053482	0.053482	0.053482	0.053482			
1	0.077943	0.055457	0.055457	0.055457	0.055457			

## LADAKH POSITIVE:

	travelpics	ladakh	follow	diio_hub	ndiiohub	travel	traveler	\
0	0.027844	0.027844	0.027844	0.027844	0.027844	0.027844	0.027844	
1	0.033706	0.033706	0.033706	0.033706	0.033706	0.033706	0.033706	
	travelingram	traveltheworld		travelagency	travelphotography		travelgram	\
0	0.027844		0.027844	0.027844		0.027844	0.027844	
1	0.033706		0.033706	0.033706		0.033706	0.033706	
	tourism	tour	tourist	photogram	photographer	video	reels	\
0	0.027844	0.027844	0.027844	0.027844	0.027844	0.027844	0.027844	
1	0.033706	0.033706	0.033706	0.033706	0.033706	0.033706	0.033706	
	travelling	beautiful	nature	naturephotography		naturelover	story	\
0	0.027844	0.078267	0.027844		0.027844	0.027844	0.027844	
1	0.033706	0.000000	0.033706		0.033706	0.033706	0.033706	

## LADAKH NEGATIVE:

[illegible]

### 10.3:P/N CLASSIFICATION

#### TWEETS

```
50      chathrapathi sivaji terminus most photographd ...
51              love shivaji park and the greenery
52              shivaji park beautiful place for me
53      blissful walks at shivaji park beautiful place...
54              juhu beach love
55              love juhu beach
56              love my time spent at hanging gardens
57              good food at mumbai
58      mumbai famous vadapav at streets enjoying with...
59              love mumbai and juhu beach
60      street food enjoy at mumbai India misal and va...
61      mumbai India best for beaches I love my time here
62              love the unity in diversity of mumbai
63      love mumbai street food vada pav yummy at chea...
64              I love juhu beach enjoy alone time only me
65              mumbai is love I love mumbai India very much
66              mumbai traffic makes it one of the worst cities
67      elephant caves at mumbai are not maintained pr...
68      mumbai traffics makes it unfit and unworthy of...
69      dharavi slum is one of the worst living space...
```

#### SENTIMENTS:

Here -1 indicates negative sentiment

and 1 indicates positive sentiment

52	1
53	1
54	1
55	1
56	1
57	1
58	1
59	1
60	1
61	1
62	1
63	1
64	1
65	1
66	-1
67	-1
68	-1
69	-1
70	-1
71	-1

```
print(X_train)
```

```
(0, 1335)    0.3571101232386054
(0, 4261)    0.3571101232386054
(0, 1275)    0.3571101232386054
(0, 1334)    0.3571101232386054
(0, 4260)    0.3571101232386054
(0, 1274)    0.3571101232386054
(0, 1333)    0.32155098228266515
(0, 4251)    0.22112590691841885
(0, 2642)    0.28730203334750054
(1, 3839)    0.12122945403585089
(1, 7286)    0.12122945403585089
(1, 4032)    0.12122945403585089
(1, 4426)    0.12122945403585089
(1, 5854)    0.12122945403585089
(1, 1997)    0.12122945403585089
(1, 6406)    0.12122945403585089
(1, 4950)    0.12122945403585089
(1, 2479)    0.12122945403585089
(1, 5273)    0.12122945403585089
(1, 5782)    0.12122945403585089
(1, 6593)    0.12122945403585089
(1, 5798)    0.12122945403585089
(1, 2206)    0.12122945403585089
(1, 2495)    0.12122945403585089
(1, 3656)    0.12122945403585089
:
(323, 5666)  0.16338247553723542
(323, 7355)  0.16338247553723542
(323, 6308)  0.16338247553723542
(323, 6851)  0.16338247553723542
(323, 5278)  0.16338247553723542
(323, 6334)  0.15288925017335472
(323, 3629)  0.16338247553723542
```

```
print(y_train)
```

```
310    1
354    1
44     1
233   -1
181    1
281    1
338    1
300    1
53     1
99    -1
138    1
124    1
162    1
25     1
106   -1
62     1
207    1
110   -1
189   -1
206    1
```

```
print(X_test)
```

```
(0, 5043)    0.3955892789294944:  
(0, 6811)    0.3955892789294944:  
(0, 4281)    0.352156260228695  
(0, 5042)    0.352156260228695  
(0, 4280)    0.352156260228695  
(0, 5041)    0.352156260228695  
(0, 4251)    0.2180584289066456!  
(0, 6810)    0.3267495729956067  
(0, 156)     0.1914349927494919  
(1, 7491)    0.2365263185850293:  
(1, 401)     0.2365263185850293:  
(1, 1662)    0.2365263185850293:  
(1, 4547)    0.2213354362260696:  
(1, 1263)    0.2105573336667960!  
(1, 3085)    0.2213354362260696:  
(1, 4212)    0.2365263185850293:  
(1, 6656)    0.2365263185850293:  
(1, 7490)    0.2365263185850293:  
(1, 400)     0.2365263185850293:  
(1, 1660)    0.2213354362260696:  
(1, 4546)    0.2105573336667960!  
(1, 1262)    0.2105573336667960!  
(1, 3084)    0.2213354362260696:  
(1, 4208)    0.1895911458722207  
(1, 5855)    0.2365263185850293:  
:  
(80, 1508)   0.1071122636957514:  
(80, 661)    0.1071122636957514:
```

```
print(y_test)
```

```
349    1  
2      1  
393   -1  
402   -1  
155    1  
4      1  
276    1  
232   -1  
80     -1  
47     1  
57     1  
32     1  
97    -1  
18     1  
331    1  
103   -1  
309    1
```



## OUTPUT:

```
#predict for testing set
predicted_naive = model_naive.predict(X_test)

print(predicted_naive)
print(type(predicted_naive))

[ 1  1 -1 -1  1  1  1 -1 -1  1  1  1  1  1  1  1  1  1  1  1  1
  1  1 -1  1 -1  1  1  1  1  1 -1  1  1  1 -1  1  1 -1 -1  1  1  1  1
  1 -1  1  1  1  1 -1  1  1  1 -1  1 -1 -1  1  1  1  1 -1  1 -1  1  1
  1  1  1  1  1  1  1  1]
<class 'numpy.ndarray'>
```

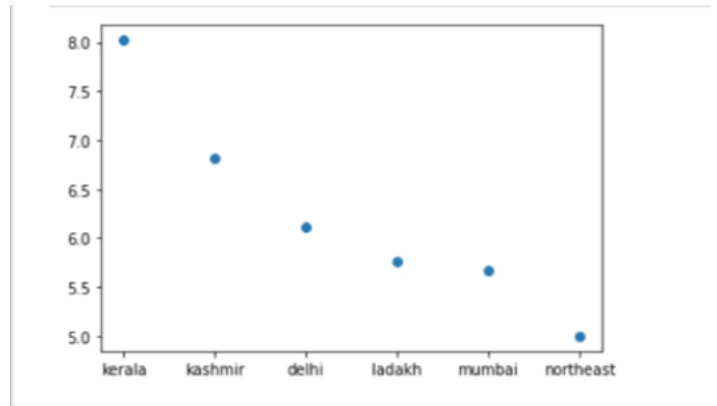
```
Testing Against New inputs:
1 . mumbai is the worst city in the world the traffic is so annoying   Negative
2 . had an amazing day - marine drive & some amazing temples. the whole of mumbai had light rains & heard my favorite song
   s. now family time! :)   Positive
3 . My family truly enjoy visiting Mumbai, Udwada and Navsari to pray. Mumbai for a holiday is a fun destination.   Positiv
   e
4 . Stuck in a train since an hour reaching d station one hour late no place to sit no place to stand either ppl all around
   too much heat on a december say mumbai is always a bad experience when u travel   Negative
5 . I love Mumbai the most It's a beautiful city it has an energy in the air people are more lively most importantly the ci
   ty which taught me to fight back in spirit, it has got it's unique culture which you won't find anywhere and it's the most s
   afest city   Positive
6 . cannot stay in pahalgam for more than a day there is a constant fear of terrorits poor management very bad roads. Hone
   stly the view does not makeup for these cons   Negative
7 . dal lake is unclean authorities need to take some action very poor maintainence   Negative
8 . made a donation to kazhiranga national park today. These people are doing some really good work. Happy to be a part of
   it   Positive
9 . I am sad to say that all those stereotypes about northeast india are actually true I visited assam last week and the who
   le state was underdeveloped. Poor roads bad management dirty streets and the list just goes on   Negative
10 . visited dal lake today I drank tea in a floating boat such a serene view   Positive
11 . I visited mizoram last week the people are not friendly and did not treat me well I regretted my trip   Negative
12 . munnar is an amazing tourism spot in kerala. plantations of tea cardomom etc can be seen in munnarhills.   Positive
[-1  1  1 -1  1 -1 -1  1 -1  1 -1  1]
```

## 10.4: REVIEW GENERATION

### 10.4.1.Sentiment Score Calculation:

Input -Spot Name	Output - Ratings
<b>Kashmir</b>	Enter place name kashmir Rating of kashmir : 6.73 kashmir is an average tourist spot The following are better tourist spots kerala
<b>Ladakh</b>	Enter place name ladakh Rating of ladakh : 5.67 ladakh is an average tourist spot The following are better tourist spots kerala kashmir
<b>Mumbai</b>	Enter place name mumbai Rating of mumbai : 5.59 mumbai is an average tourist spot The following are better tourist spots kerala kashmir ladakh
<b>NorthEast India</b>	Enter place name northeast Rating of northeast : 4.79 northeast is a poor tourist spot The following are better tourist spots kerala kashmir ladakh mumbai
<b>Kerala</b>	Enter place name kerala Rating of kerala : 8.02 kerala is a great tourist spot kerala is the best tourist place according to us
<b>Delhi</b>	Enter place name delhi Rating of delhi : 6.12 delhi is an average tourist spot The following are better tourist spots kerala kashmir

### 10.4.2.Rating Graph:



From the above graph it is evident that the state of Kerala has been a very popular and great destination for tourism with a very high rating of 8.02. Next to Kerala stands Kashmir with a rating of 6.73. Delhi, Mumbai, Ladakh follow Kashmir with ratings of 6.12, 5.67, 5.59 respectively. Though these might be great cities, they do not shine very well in the aspect of tourism. Hence they are considered as "Average tourist" spots. Finally Northeast India has a very low rating of 4.79. This implies that even though a few tourists had amazing experience in the northeast, majority of the people have expressed a lot of negative opinions about northeast India. Hence it is considered as a poor tourist spot.

### 10.4.3.Summarization of tweets:

Positive and Negative summary of the given place is generated.

Place Name	Input file	Positive and Negative aspects of the place
Mumbai	Positivecleaned mumbai.csv	<b>Positive Aspects of Mumbai:</b> bollywood cinemas are famous in mumbai and mumbai gave life to many people. vada pav at streets is good but not hygienic. mumbai is arguably one of the best cities in india. juhu beach is great experience for me blissful walks at shivaji park. juhu beach is a great place to spend time with friends and family. chowpatty beach is a great place to visit with a view of the sea.
	Negativecleaned mumbai.csv	<b>Negative Aspects of Mumbai:</b> poor unclean city suburbs have never been shown mumbai slums are dirty and poorly managed slums are dirty and disgusting at times. traffic at mumbai is making it unfit and



		destroying bikes and harassing people. the nubra valley is terrible most unsafe, dangerous for tourists.
--	--	--

Place Name	Input file	Positive and Negative aspects of the place
<b>Kerala</b>	<b>Positivecleaned kerala.csv</b>	<b>Positive Aspects of Mumbai:</b> thekkady is a very good tourist spot kerala is so beautiful and amazing. the jatayu nature park is a must visit place kerala is so beautiful and welcoming. thekkady is really underrated the widelife si beautiful. thekkady is a must visit place for a kerala trip. the jatayu nature park is a must visit place for bird lovers. thekkady is a must visit.
	<b>Negativecleaned kerala.csv</b>	<b>Negative Aspects of Mumbai:</b> kerala is really unsafe for north indian tourists. roads are really bad and dirty kerala food is really bad and disgusting. kerala people are bad drivers.
<b>NorthEast India</b>	<b>Positivecleaned Noertheast.csv</b>	<b>Positive Aspects of northeast:</b> kaziranga is one of the most amazing places to be in North East India. visit the beautiful river island Majuli in Assam. visit the ancient silk route and the Kanchenjunga waterfalls. A day trip from Gangtok is a must. A visit to pelling is a must. a small town in west Sikkim is popular among tourists. a visit to the ruins of the 2nd capital of Kingdom of sikkim
	<b>Negativecleaned Noertheast.csv</b>	<b>Negative Aspects of northeast:</b> arunachal Pradesh is so unclean and unhygienic Illegal immigrants attack tourists in Tripura Kaziranga National Park is a cleanest national park in the world majuli unsafe to travel for every single passenger visited majuli last year guwahati roads are in worst state possible Gangtok roads are life threatening and immensely dangerous Gangtok to gurudongmar and Rohtang Roads built every year but same condition Gangtok

#### 10.4.4.Final General Review for the Tourist Spots:

**Transportation-** Gives the general idea(both pros and cons) regarding the traffic, taxi, bus, train , air and water transportation services in the tourist spot.

**Safety-**This tells how safe the tourist spot is.

**Local** - Gives the general idea about the ambience , local people and local tourist spots of the place.

**Cleanliness-**This tells how clean the tourist spot is.

Place Name	Summary
Mumbai	<p><b>Transportation:</b></p> <p>How good is transportation in mumbai: great airport is connected via air to all the cities in india.black and yellow taxi is perfect to get around mumbai for tourists. great city centre is also connected via air to all the cities in india. But remember these facts about mumbai before visiting: mumbai traffic makes it one of the worst cities in the world. traffic from mahim bandra road is causing headaches. people who spend time at traffic makes mumbai worst. traffic is causing headaches.</p> <p><b>Safety:</b></p> <p>Yes it is safe in mumbai travel as a group in mumbai then it is safe crimes against tourists are low in mumbai But here's why people think it's unsafe: street food at mumbai are poorly maintained and disgusting at times. they cause health problems for tourists. travellers need to avoid traveling alone on public transport or in taxis. if you are travelling alone on public transport or in taxis, you can easily become victim to crime.</p> <p><b>Local:</b></p> <p>Here's why you should really visit mumbai mumbai is a beautiful city and the city of dreams is good for bollywood starsgateway of India monument stands beautiful in mumbai city. mumbai serves scrumptious food for the tourists. juhu beach refreshing and rejuvenating beautiful place. Why you should avoid mumbai bollywood always shown beautiful mumbai poor unclean city suburbs have never been shown. local trains are always crowded, unclean and suffocating. if you wish to spend a peaceful time at Nariman Point then it's going to be a bad idea</p> <p><b>Cleanliness:</b></p> <p>Colaba is absolutely stunning. Clean streets and clean air. But is mumbai really clean? bollywood always shown beautiful mumbai poor unclean city suburbs have never been shown mumbai beaches are just dirty. too much sand, sea and sunset to deal with!the local trains are always crowded, unclean and suffocating!</p>

Place Name	Summary
<p><b>Kerala</b></p>	<p><b>Transportation:</b>  ooty munnar road Such beautiful trip unforgettable experience munnar hills are easy to get to roads are pretty good thrilling road jeep safari with lovely valley views along walk into long made hike tangled jungles. frequency of buses is high in big cities like cochin and ernakulam.  But remember these facts about kerala before visiting:  kerala roads are really bad needs more maintenance self drive to munnar is not advisable. just mud hilly areas in kerala have poor roads state government buses are in bad condition.</p> <p><b>Safety:</b>  Yes it is safe in kerala  Kerala is absolutely safe to travel big cities like cochin, trivandrum are very safe for women.  But here's why people think it's unsafe:  kerala is unsafe for north indian tourists. kerala is not safe for women.</p> <p><b>Local:</b>  Kerala is known for traditional arts and people enjoy traditional, percussion-filled music. Kerala is one of the most beautiful places in india. kerala is a great destination for tourists.  Why you should avoid kerala  kerala food is really bad and disgusting kerala food is really bad and disgusting. there are few places where theres literally no road. just mud</p> <p><b>Cleanliness:</b>  cochin is the cleanest town in kerala. the island village of Kumbalangi is a pure dreamland.  But is kerala really clean?  kerala is a very dirty place and many places are littered and remain dirty for days. local hotels in munnar are really unhygienic and unclean.</p>

Place Name	Summary
<p><b>NorthEast India</b></p>	<p><b>Transportation:</b>  guwahati roads have improved significantly in the last 5 years. The roadworld is a great place to explore guwahati.  But remember these facts about northeast before visiting: guwahati roads are unsafe and indisciplined traffic. gangtok roads are life threatening and immensely dangerous</p> <p><b>Safety:</b>  Yes it is safe in northeast  India's northeast is not as bad as it is shown in movies. big cities are safer compared to interior villages.  But here's why people think it's unsafe:  Bangaldesh illegal immigrants attack tourists in tripura. gangtok roads are life threatening and immensely dangerous.</p> <p><b>Local:</b>  kaziranga national park is one of the most amazing places to be in North East India. Kaziranga national park is the wildlife destination of North East India. Majuli river island located along the Brahmaputra River is currently listed as the world's largest river island in the Guinness Book of World Records.  Why you should avoid northeast  arunachal Pradesh is a dangerous place and people are hostile. the locals are unfriendly and we hated the food. Horrible experience in a very bad place. Assam is exactly as it is stereotyped.</p> <p><b>Cleanliness:</b>  banjhakri waterfalls is so pure tsongmo lake is so clean that it's practically transparent. mawlynnong is the cleanest place in the whole world.  But is northeast really clean?  tripura is so unclean and unhygienic. activities of roadside dhabas and markets make the areas adjacent to Kaziranga National Park quite dirty.</p>



Place Name	Summary
<b>Kashmir</b>	<p><b>Transportation:</b>  state transport in kashmir is really good. nice improvement over the last few years. srinagar roads are not a problem if you are a good driver. state transport in kashmir is really good.  But remember these facts about kashmir before visiting:  patnitop roads are in bad condition. stayed 2 days in pahalgam very dirty bad roads concrete everywhere. it is a terrible journey. patnitop fails to attract tourists from all over the country due to bad condition of the road.</p> <p><b>Safety:</b>  Yes it is safe in kashmir  kashmir: tourists visiting kashmir need to stay together to be safe. kashmir: tourists visiting kashmir need to stay together to be safe. kashmir: tourists visiting kashmir need to stay together to be safe.  But here's why people think it's unsafe:  kashmir is unsafe for tourism sullied in political turmoil. walking on icy Dal Lake can be life threatening &amp; can result in slips, falls, fractures &amp; fatal injuries.</p> <p><b>Local:</b>  Here's why you should really visit kashmir  gulmarg trekking amazing experience pumped with adrenaline. kashmir is an amazing place shown wrongly in movies. srinagar is a heaven on earth so beautifulvisited dal lake in srinagar.  Why you should avoid kashmir  kashmir is extremely unsafe and full of terrorists never visit kashmir. kashmiri people really don't know how to behave with tourists.</p> <p><b>Cleanliness:</b>  The lakes in kashmir are crystal clear,stunning and pristine.The valleys of Kashmir are untamed and unspoilt.  But is Kashmir really clean?  Kashmir's iconic Dal lake is dying a slow death.Cleaning drives on the Dal are not new to Kashmir.</p>

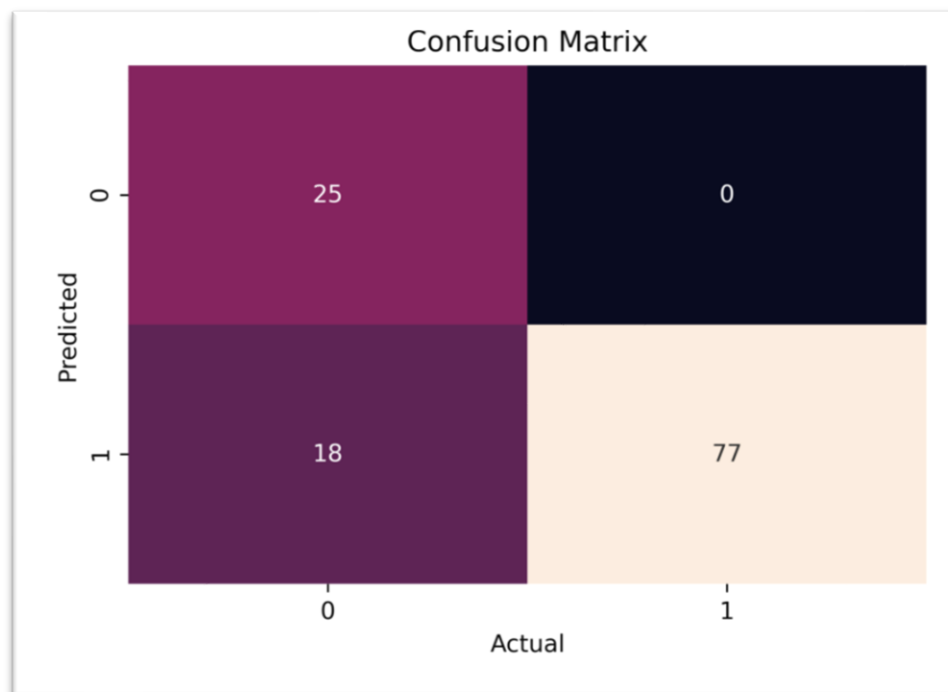
Place Name	Summary
<p><b>Ladakh</b></p>	<p><b>Transportation:</b>  How good is transportation in ladakh:  ladakh bike trip takes enchanting lands most coveted booming destinations indian tourism world most thrilling adventurous bike trip withmazing ladakh roadtrip unforgettable experience. ladakh roadtrip takes enchanting lands most coveted booming destinations indian tourism world most thrilling.  But remember these facts about ladakh before visiting:  quality of roads in and around Ladakh must be improved. ladakh is unsafe for tourism with unions destroying bikes. there are so many thieves in Ladakh...</p> <p><b>Safety:</b>  Yes it is safe in ladakh  ladakh region doesn't require any special permission to visit and it's absolutely safe too. Definitely give it a consideration if you're looking for adventurous bike trip through most beautiful places.  But here's why people think it's unsafe:  Ladakh is unsafe for tourism with unions destroying bikes, harassing people and forcefully troubling them. Safe trekking trails water stress and intense pressures on a fragile ecosystem are just some of the by-products of Ladakh embrace of unregulated tourism.</p> <p><b>Local:</b>  Here's why you should really visit ladakh  ladakh is incredibly beautiful in every season but i have somehow fallen in love with its white expanse more. enjoy the sheer beauty of ladakh and pangong tso lake. Amongst the most starkly beautiful places on earth.  Why you should avoid ladakh  ladakh is overhyped Locals are not friendly and never help. had a horrible experience. nubra valley highways terrible most unsafe dangerous for tourists. for anyone the nubra Valley an unsafe place to go alone. rain ruined my ladakh trip.</p> <p><b>Cleanliness:</b>  Markha Valley is very clean With no contact with the outer world Markha Valley is very pure. Clean air. Fresh water in small lakes. Leh is very clean. Clean air. Clean water in small lakes.  But isladakh really clean?  pangong lake and its surroudings where 3 idiots was shot have become so unclean. there is no difference between the lake and sewages now public facilities in ladakh are extremly unhygienic garbage everywhere.</p>

## 10.5. Inference:

Thus we've collected tourism related tweets for Delhi ,Mumbai ,Kerala ,Kashmir, Ladakh and northeast India from the Twitter API using appropriate keywords. We have cleaned the tweets using regex and stop words. We have classified the tweets using Multinomial naive Bayes classifier. The classifier has provided us with a good accuracy of 85 percent. We have also generated rating scores for the tourist spots and have compared the ratings to provide good suggestions to the users on which tourist spot to visit. We have generated an overall and specific summary for all the tourist places. Thus we have built a model that can be useful to the tourists and the tourism recommendation system.

## 11.EVALUATION METRICS

### 11.1.Confusion Matrix:



The model has correctly predicted 77 inputs as positive and 25 inputs as negative. The model has not predicted any positive input as negative but it has predicted 18 negative inputs as positive

## 11.2. Accuracy:

### Accuracy Score

```
▶ from sklearn.metrics import accuracy_score #2
score_naive = accuracy_score(predicted_naive, y_test)
print("Accuracy: ",score_naive*100,"%")
```

Accuracy: 85.0 %

The accuracy of this model is now 85%. It can be increased by increasing the size of the dataset.

## 11.3. Classification Report:

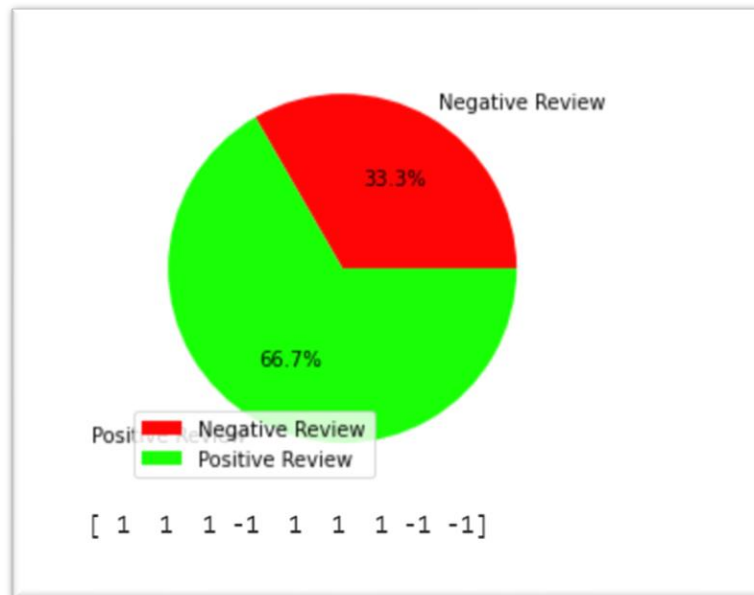
### Classification Report

```
▶ from sklearn.metrics import classification_report #3
print(classification_report(y_test, predicted_naive))
```

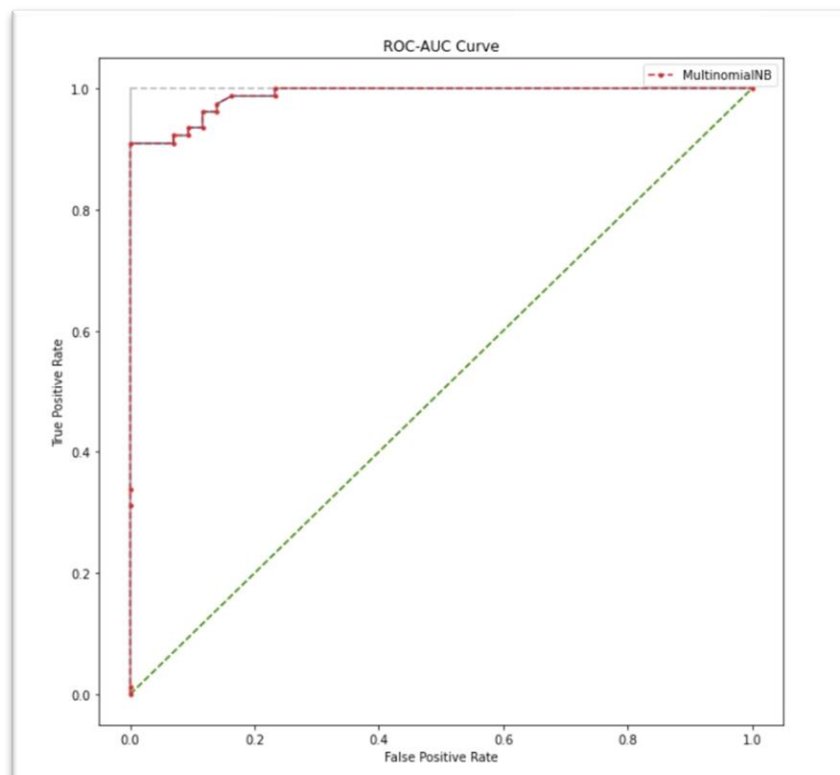
	precision	recall	f1-score	support
-1	1.00	0.58	0.74	43
1	0.81	1.00	0.90	77
accuracy			0.85	120
macro avg	0.91	0.79	0.82	120
weighted avg	0.88	0.85	0.84	120

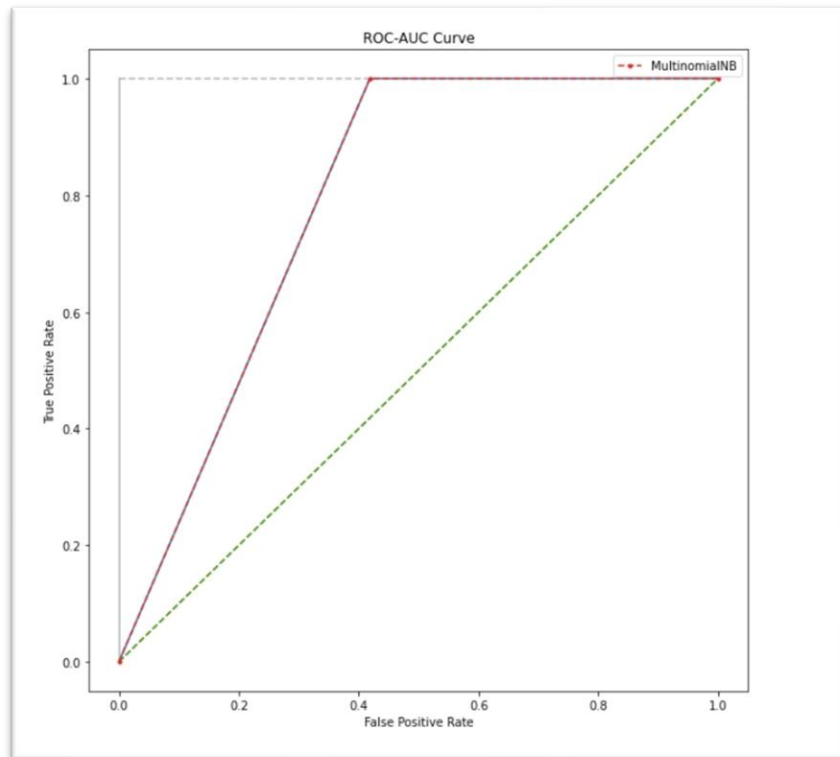
We can see that the classification report shows an accuracy of 85%.

#### 11.4 Pie chart indicating the percentage of positive and negative tweets in the test inputs:

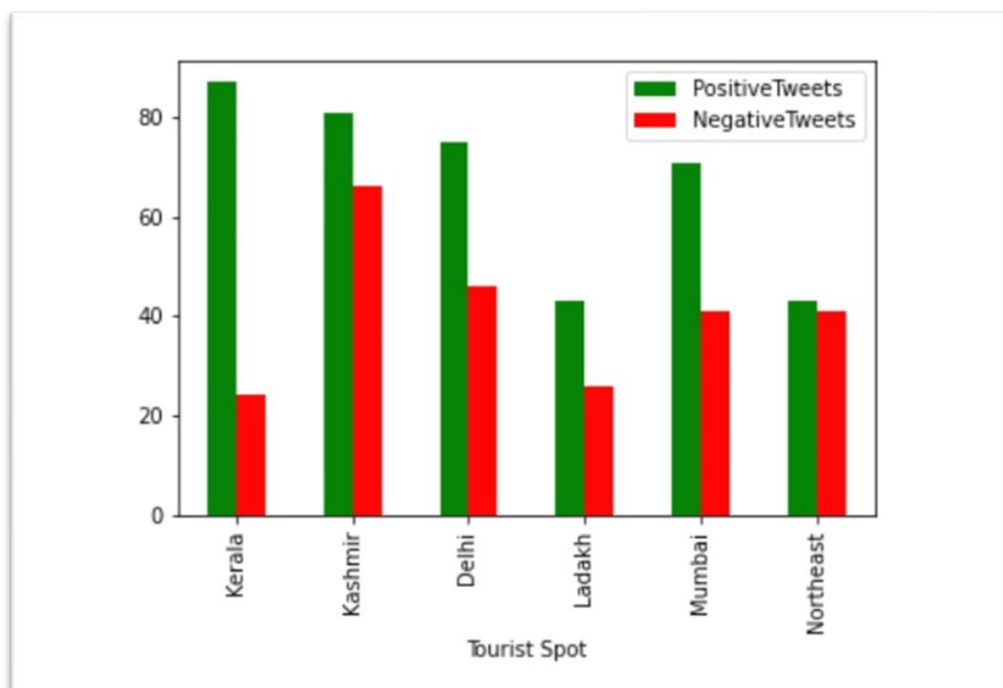


#### 11.5 ROC curves for the multinomialNB classifier for the training set:





**11.6 Bar graph indicating the number of positive and negative tweets for the tourist places.**



## **12.CONCLUSION AND FUTURE DIRECTION**

We have completed all the 4 modules in the block diagram .We have successfully extracted tweets pertaining to various tourist spot across India from the Twitter API and have successfully performed sentiment analysis on them using Naïve Bayes classifier. We have calculated the sentiment score and validated the user inputs for the tourist spots. We have also generated a general opinion or review about the particular tourist spot based on the user input. The general review provides information regarding the safety, cleanliness, transportation and local places of the tourist spots.

Further more our model can be extended to work for many other tourist spots as well. It can also be made into a website with a good user interface for ease of use for the layman. Thus our model can be of immense use to the tourism recommendation system and for confused people who are not able to decide a good tourist place to visit to come to a wise conclusion.

## **13.REFERENCES**

1. Kazutaka Shimada, Shunsuke Inoue, Hiroshi Maeda and Tsutomu Endo. “Analyzing Tourism Information on Twitter for a Local City.” *In The First ACIS International Symposium on Software and Network Engineering*,pages 61-66,2011.
2. Rico Yudha Saputra,Lukito Edi Nugroho,Sri Suning Kusumawardani. “Collecting the Tourism Contextual Information data to support the tourism recommendation system.” *In The International Conference on Information and Communications Technology (ICOIACT)*,pages 79-84,2019.
3. Muhammad Afzaal,Muhammad Usman. “A Novel Framework for Aspect-based Opinion Classification for Tourist Places.” *In The Tenth International Conference on Digital Information Management (ICDIM)* ,pages 1-9,2015.
4. Ananchai Muangon, Sotarath Thammaboosadee,Choochart Haruechaiyasak. “A Lexiconizing Framework of Feature-based Opinion Mining in Tourism Industry.” *In International Conference on Digital Information and Communication Technology and its Applications(DIC TAP)*,pages 169-173,2014.

5. Ankur Goel,Jyoti Gautam,Sitesh Kumar. “Real Time Sentiment Analysis of Tweets Using Naive Bayes.” *In The 2nd International Conference on Next Generation Computing Technologies Dehradun, India*, pages 257-261,2016.