SENTIMENT ANALYSIS OF HOTEL REVIEW

A UG PROJECT PHASE-I REPORT

Submitted to

JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY, HYDERABAD

In partial fulfilment of the requirements for the award of the degree of

BACHELOR OF TECHNOLOGY
IN
COMPUTER SCIENCE AND ENGINEERING

Submitted by

PADIDALA SRINIKHIL	**19UK1A05M5**
GADDAM DEEKSHITHA	**19UK1A05P2**
DONTHULA ARUN	**20UK5A0506**

Under the esteemed guidance of

Dr. K. SHARMILA REDDY

(Associate Professor)

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
VAAGDEVI ENGINEERING COLLEGE
(Affiliated to JNTUH, HYDERABAD)
Bollikunta, Warangal - 506005

2019-2023

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
VAAGDEVI ENGINEERING COLLEGE
BOLLIKUNTA, WARANGAL---506005
2019-2023

CERTIFICATE OF COMPLETION

A UG PROJECT PHASE-I

This is to certify that the UG Project Phase-I entitled "**SENTIMENT ANALYSIS OF HOTEL REVIEW"** is being submitted by **PADIDALA.SRINIKHIL(19UK1A05M5), GADDAM.DEEKSHITHA(19UK1A05P2), DONTHULA.ARUN(20UK5A0506**) in partial fulfilment of the requirements for the award of the degree of **Bachelor of Technology** in **Computer Science and Engineering** to **Jawaharlal Nehru Technological University Hyderabad** during the academic year 2019-23, is a record of work carried out by them under the guidance and supervision.

Project Guide
Dr. K. SHARMILA REDDY
(Associate Professor)

Head of the Department
Dr. R. NAVEEN KUMAR
(Professor)

External

ACKNOWLEDGEMENT

We wish to take this opportunity to express our sincere gratitude and deep sense of respect to our beloved **Dr.P.PRASAD RAO,** Principal, Vaagdevi Engineering College for making us available all the required assistance and for his support and inspiration to carry out this UG Project Phase-1 in the institute.

We extend our heartfelt thanks to **Dr.R.NAVEEN KUMAR**, Head of the Department of CSE, Vaagdevi Engineering College for providing us necessary infrastructure and thereby giving us freedom to carry out the UG Project Phase-1.

We express heartfelt thanks to Smart Bridge Educational Services Private Limited, for their constant supervision as well as for providing necessary information regarding the UG Project Phase-1 and for their support in completing the UG Project Phase-1.

We express heartfelt thanks to the guide, **Dr.K.SHARMILA REDDY** Associate professor, Department of CSE for her constant support and giving necessary guidance for completion of this UG Project Phase-1.

Finally, we express our sincere thanks and gratitude to my family members, friends for their encouragement and outpouring their knowledge and experience throughout the thesis.

P.SRINIKHIL (19UK1A05M5)
G.DEEKSHITHA (19UK1A05P2)
D.ARUN (20UK5A0506)

ABSTRACT

We consider the problem of classifying a hotel review as a positive or negative and thereby analysing the sentiment of a customer. Using Hotel review data from Kaggle, we find that standard Machine Learning techniques can definitely outperform human-produced sentiment analysis baselines. We will explore wide range of probabilistic models including Naive Bayes Classifier, Logistic Regression Classifier to classify a review. To extract the frequent words from the reviews we have used stop words , word net. We conclude by comparing accuracy of different strategic models and discuss about scope for future work.

Keywords: Hotel Review , NLTK, Naïve Bayes Classifier , Logistic Regression Classifier.

TABLE OF CONTENTS :-

LIST OF FIGURES **PAGE NO**

1. INTRODUCTION

1.1 MOTIVATION:

There are many researchers trying to surpass the latest best results and achieve the state-of-the-art in English sentiment analysis by using handcrafted features. This approach may result into overfitting the data. However, sentiment analysis in Czech has not yet been thoroughly targeted by the research community.

Czech as a representative of a inflective language is an ideal environment for the study of various aspects of sentiment analysis (overview or breadth study of sentiment analysis if you will) for inflectional languages. It is challenging because of its very flexible word order and many different word forms. We conceive this study to deal with several aspects of sentiment analysis. The breadth of this study can lead to more general view and better understanding of sentiment analysis. We can reveal and overcome unexpected obstacles, create necessary evaluation datasets and even come up with new creative solutions to sentiment analysis tasks.

1.2 DEFINITION:

Sentiment analysis (also known as opinion mining or emotion AI) is the use of natural language processing, text analysis, computational linguistics, and biometrics to systematically identify, extract, quantify, and study affective states and subjective information. Sentiment analysis is widely applied to voice of the customer materials such as reviews and survey responses, online and social media, and healthcare materials for applications that range from marketing to customer service to clinical medicine. With the rise of deep language models, such as RoBERTa, also more difficult data domains can be analysed, e.g., news texts where authors typically express their opinion/sentiment less explicitly.

1.3 OBJECTIVE OF PROJECT:

By the end of this project, you will:

- Know fundamental concepts and techniques used for machine learning.
- Gain a broad understanding of data.
- Have knowledge on pre-processing textual data and classification algorithms.

1.4 PURPOSE:

Recent years have seen rapid growth in online discussion groups and review sites (e.g.www.tripadvisor.com) where a crucial characteristic of a customer's review is their sentiment or overall opinion — for example if the review contains words like 'great', 'best', 'nice', 'good', 'awesome' is probably a positive comment. Whereas if reviews contains words like 'bad', 'poor', 'awful', 'worse' is probably a negative review. However, Trip Advisor's star rating does not express the exact experience of the customer. Most of the ratings are meaningless, large chunk of reviews fall in the range of 3.5 to 4.5 and very few reviews below or above. We seek to turn words and reviews into quantitative measurements. We extend this model with a supervised sentiment component that is capable of classifying a review as positive or negative with accuracy (Section 4). We also determine the polarity of the review that evaluates the review as recommended or not recommended using semantic orientation. A phrase has a positive semantic orientation when it has good associations (e.g., "excellent, awesome") and a negative semantic orientation when it has bad associations (e.g., "terrific, bad"). Next step is to assign the given review to a class, positive or negative, based on the average semantic orientation of the phrases extracted from the review. If the average is positive, the prediction is that the review posted is positive. Otherwise, the prediction is that the item is negative.

2.PROBLEM STATEMENT

The project "Hotel Management System" is used for maintaining the information for each and every customer, employee, driver and product. Each and every customer has own personal details and the products are transferred through the driver. Before providing the product to customer, the administrator gathering necessary information about the product meanwhile the invoice is also raised for the customer product.

The Hotel Manager software is a web-based application that is designed to be implemented for hotels of any size(scale). The software product will automate the major hotel operations. The first subsystem is a Reservation and Booking System to keep track of reservations and room availability. The second subsystem is the Tracking and Selling Food System that charges the current room. The third subsystem is a General Management Services and Automated Tasks System which generates reports to audit all hotel operations and allows modification of subsystem information. These three subsystems' functionality will be described in detail in section 2-Overall Description.

There are two end users for the HMS. The end users are the hotel staff (customer service representative) and hotel managers. Both user types can access the Reservation and Booking System and the Food Tracking and Selling System. The General Management System will be restricted to management users.

The Hotel Management System's objectives is to provide a system to manage a hotel that has increased in size to a total of 100 rooms. Without automation the management of the hotel has become an unwieldy task. The end users' day-to-day jobs of managing a hotel will be simplified by a considerable amount through the automated system. The system will be able to handle many services to take care of all customers in a quick manner. The system should be user appropriate, easy to use, provide easy recovery of errors and have an overall end user high subjective satisfaction.

3.LITERATURE SURVEY

3.1 EXISTING PROBLEM:

Aiming at the lack of specific domain corpus in text sentiment polarity analysis, the inaccurate classification accuracy of the naïve Bayes algorithm due to the independence assumption and the sparse word vector matrix, a text sentiment analysis method based on the improved naïve Bayes algorithm is proposed. Combining machine learning methods with domain sentiment dictionary weighting methods. The improved word frequency inverse file frequency algorithm is used to extract the feature word vector of hotel review text, and the weight of the feature word vector of the domain dictionary after regression test is introduced to weaken the influence of the independence assumption. The singular value decomposition algorithm realizes the dimensionality reduction of the word vector sparse matrix and eliminates redundancy. The remaining features are used to construct a polynomial model of Naïve Bayes. The results of simulation research show that this method can effectively improve the effect of text sentiment classification.

3.2 PROPOSED SYSTEM:

Machine learning is a subfield of artificial intelligence (AI). The goal of machine learning generally is to understand the structure of data and fit that data into models that can be understood and utilized by people. Although machine learning is a field within computer science, it differs from traditional computational approaches. In traditional computing, algorithms are sets of explicitly programmed instructions used by computers to calculate or problem solve. Machine learning algorithms instead allow for computers to train on data inputs and use statistical analysis in order to output values that fall within a specific range. Because of this, machine learning facilitates computers in building models from sample data in order to automate decision making processes based on data inputs. In machine learning, tasks are generally classified into broad categories. These categories are based on how learning is received or how feedback on the learning is given to the system developed.

You must have prior knowledge of the following topics to complete this project.

- Supervised learning
- Unsupervised learning
- Regression and classification
- Naïve Bayes Classifier
- Logistic Regression
- Evaluation metrics (Precision, recall, fbeta)

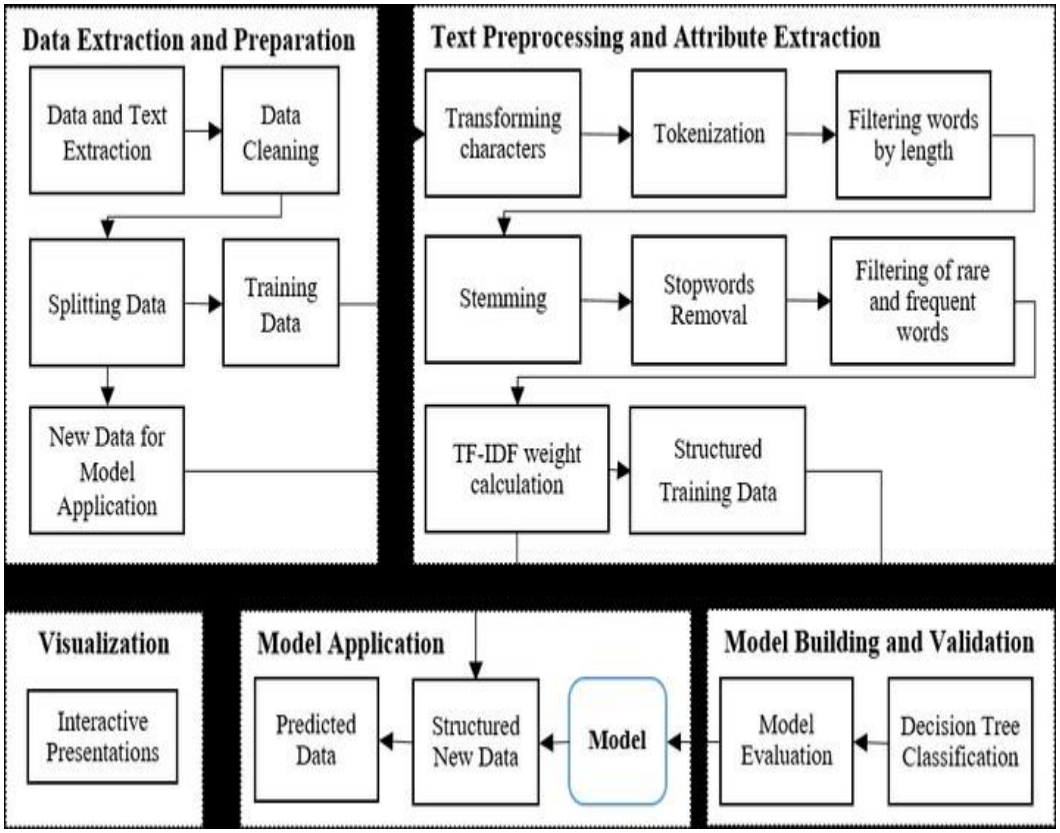

Figure 1: system overview

4.EXPERIMENTAL INVESTIGATION

For the implementation of the project, we have gone through several research papers from the "Google Scholar" website. We have gone through several websites including towardsdatascience.com, tutorials point, geeks for geeks, etc

RESEARCH PAPER 1:

Sentiment analysis is the task of identifying opinions expressed in any form of text. With the widespread usage of social media in our daily lives, social media websites became a vital and major source of data about user reviews in various fields. The domain of tourism extended activity online in the most recent decade. In this paper, an approach is introduced that automatically perform sentiment detection using Fuzzy C-means clustering algorithm, and classify hotel reviews provided by customers from one of the leading travel sites. Hotel reviews have been analysed using various techniques like Naïve Bayes, K-Nearest Neighbour, Support Vector Machine, Logistic Regression, and Random Forest. An ensemble learning model was also proposed that combines the five classifiers, and results were compared.

RESEARCH PAPER 2:

In today's scenario online reviews on various digital platforms plays a vital role for customers to buy products. Based on the reviews and ratings by the consumer on E-commerce platform like flipkart, amazon etc. products are widely accepted or rejected. Apart from products people also look for the reviews of the services provided from restaurants, hotels, airlines etc. Sentiment analysis helps the developers to easily analyse the reviews and categorize them as positive or negative. In this paper, service of a hotel is analysed by finding out the polarity of the reviews in order to get the subject information. Aspect detection and sentiment classification are the main tasks focused here. For aspect detection latent irichlet allocation (LDA) is used for building the topics. Different machine learning classifiers like naïve bayes classifier, SVM, decision tree and logistic regression are used for classification of reviews. Evaluation is done by computing the accuracy, recall, precision and F score of these algorithms.

4.1 PROJECT ARCHITECTURE:

The Project Architecture briefly explains the procedure involved:

- Firstly, Collect the dataset and split them into Training and Testing datasets.
- Preprocess both training and testing datasets.
- Pre-process or clean the data.
- Analyse the pre-processed data.
- Train the machine with pre-processed data using an appropriate machine learning algorithm.
- Save the model and its dependencies.
- Build a Web application using flask that integrates with the model built.
- Open the anaconda prompt from the start menu.
- Navigate to the folder where your app.py resides.
- Now type ―python app.py‖ command.
- It will show the local host where your app is running on **http://127.0.0.1.5000/**
- Copy that local host URL and open that URL in the browser. It does navigate me to where you can view your web

  page. • Enter the values, click on the predict button and see the result/prediction on the web page.

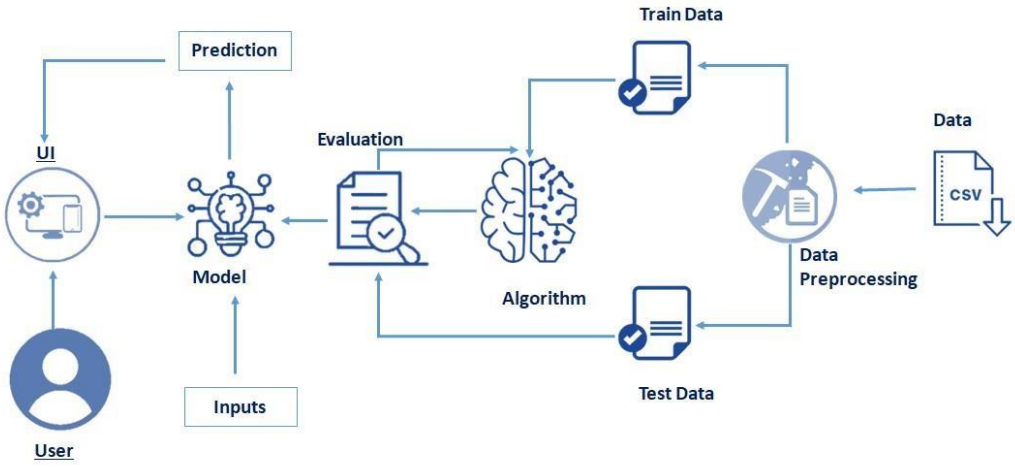

Figure 3 : Project Architecture

4.2 BLOCK DIAGRAM:

- Initially, Labelled datasets are collected.
- Preprocessing the data.
- Training using machine learning algorithms.
- Using the Navia bayes classifier models build them.
- Classify them using logistic regression.
- Again pre-process for selecting the dataset for prediction.
- Finally predict them in webpage

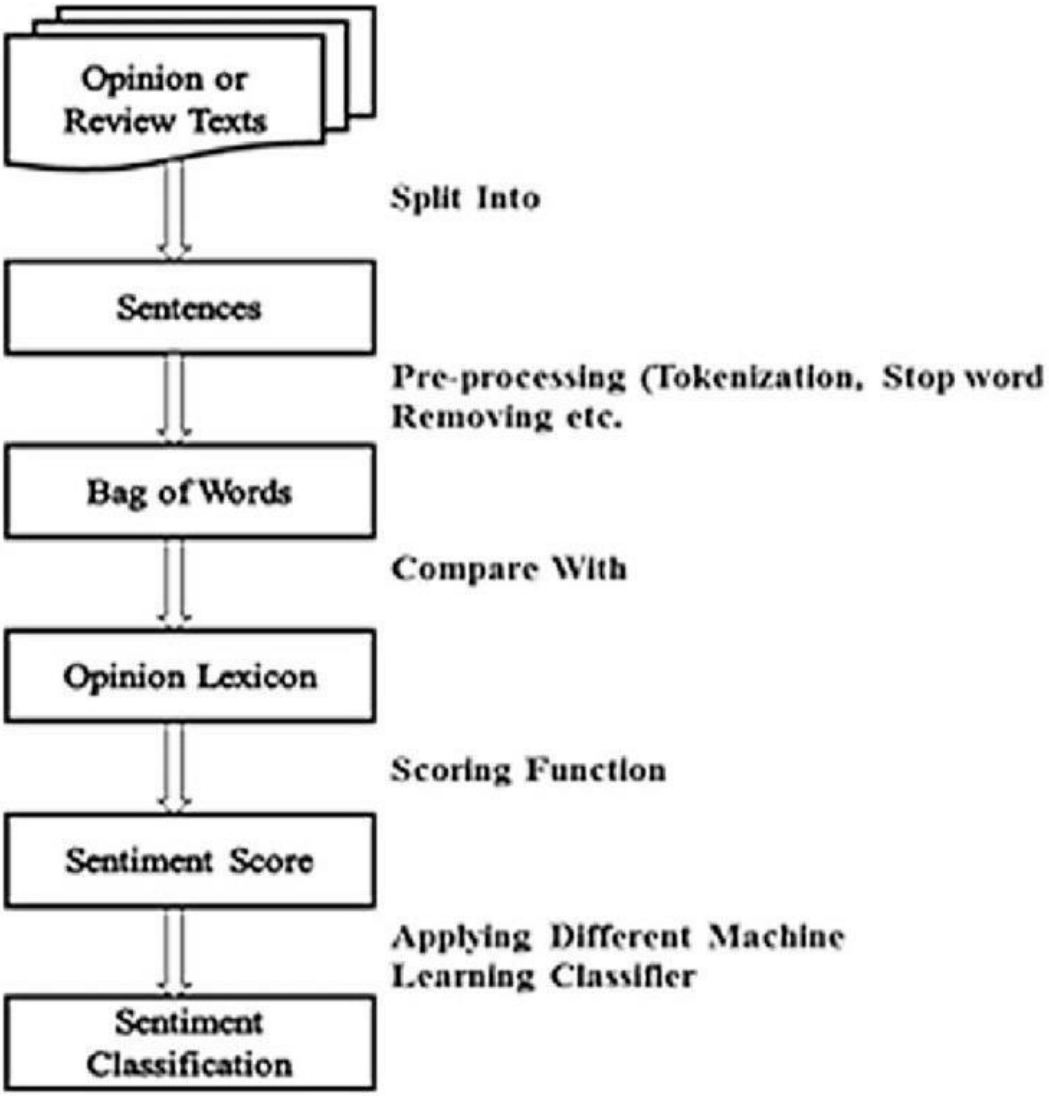

Figure 3: Block diagram representing process of Machine learning

4.3 SOFTWARE REQUIREMENTS:

- Python 3.9:
- Python is an interpreted high-level general-purpose programming language.
- Python can be used on a server to create web applications.
- Visual Studio Code:
- Visual studio code is a source-co-editor made by Microsoft for Windows, Linux and macOS.
- Features include support for debugging, syntax highlighting, intelligent code completion, snippets, code refactoring, and embedded Git.
- Anaconda Environment

- The default environment base (path) is used because it consists of multiple libraries and modules.
- Pandas and NumPy:

- Pandas and NumPy is used for the purpose of linear regression model building.

- Flask:

- Flask is the module used for web framework.

- Flask provides you with tools, libraries and technologies that allow you to build a web application.

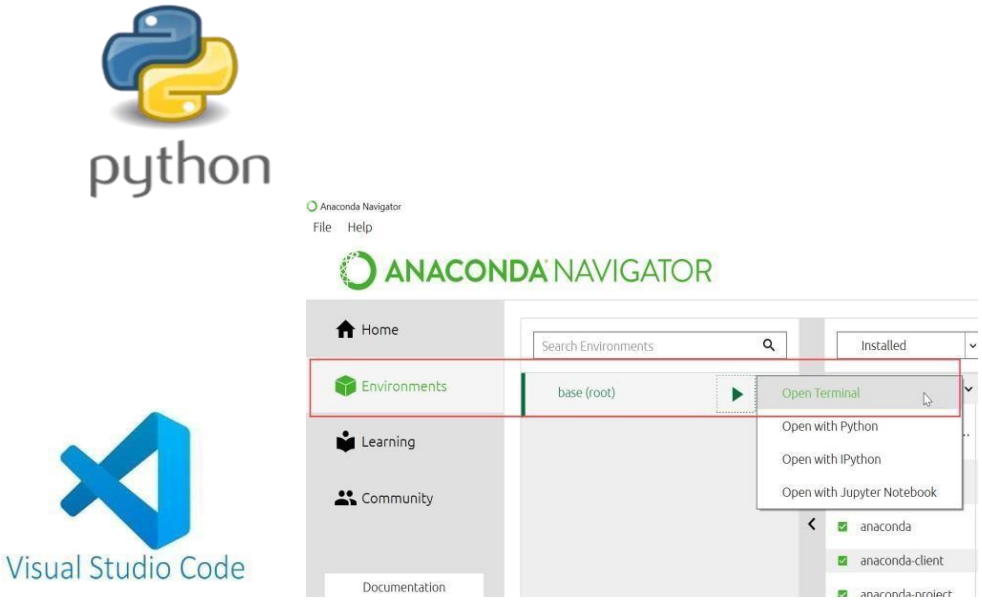

Figure 4: Logos of python and VSCode and the base environment location in Anaconda

4.4 PROJECT FLOW

4.4.1. Data Collection:

a) In our project according to project structure, create train & test folders with 5 folders of skin diseases named Acne, Melanoma, Psoriasis, Rosacea, Vitiligo in each test and train folders.

4.4.2. Data Preprocessing:

- Import dataset data generator library and configure it
- Apply data generator functionality to train and test datasets
- Import the Libraries.
- Importing the dataset.
- Checking for Null Values.

a) Data Visualization.

b) Taking care of Missing Data.

c) Label encoding.

d) One Hot Encoding.

e) Feature Scaling.

f) Splitting Data into Train and Test.

4.4.3. Model Building:

- Training and testing the model
- Evaluation of Model

4.4.4. Test the Model:

- Import the saved model:
- Import the model that is saved in a plain text file (.h5).
- Load the test data, preprocess it and then predict and check for results:
- Preprocessing the data and predicting the image which is required.

4.4.5. Application Building:

- Build a FLAK application:
- Flask provides you with tools, libraries and technologies that allow you to build a web application.
- Build the HTML page and execute it:
- HTML page is used for developing the webpage to display the result in webpage.
- Run the app:

Run the python file such that the pages are rendered and linked to webpage‘s with a local host.

- User interacts with the UI (User Interface) to upload the input features.
- Uploaded features/input is analysed by the model which is integrated.
- Once a model analyses the uploaded inputs, the prediction is showcased.

5. DESIGN

5.1 USE CASE DIAGRAM:

A use case diagram is usually simple. It does not show the detail of the use cases:

- It only summarizes some of the relationships between use cases, actors, and systems.
- It does not show the order in which steps are performed to achieve the goals of each use case. A use case is **a methodology used in system analysis to identify, clarify and organize system requirements** A use case document can help the development team identify and understand where errors may occur during a transaction so they can resolve them. Every use case contains three essential elements.

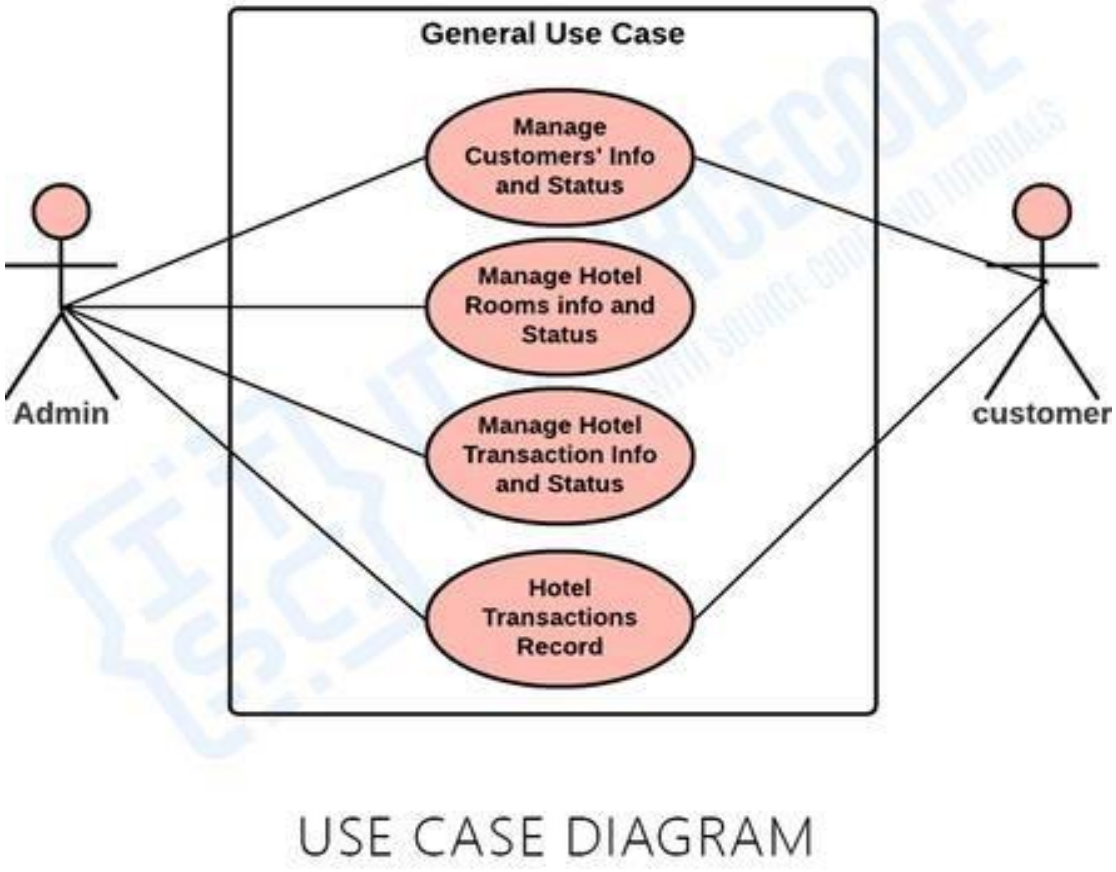

Figure 5: Use case Diagram

5.2 FLOWCHART:

A flowchart is a picture of the separate steps of a process in sequential order.

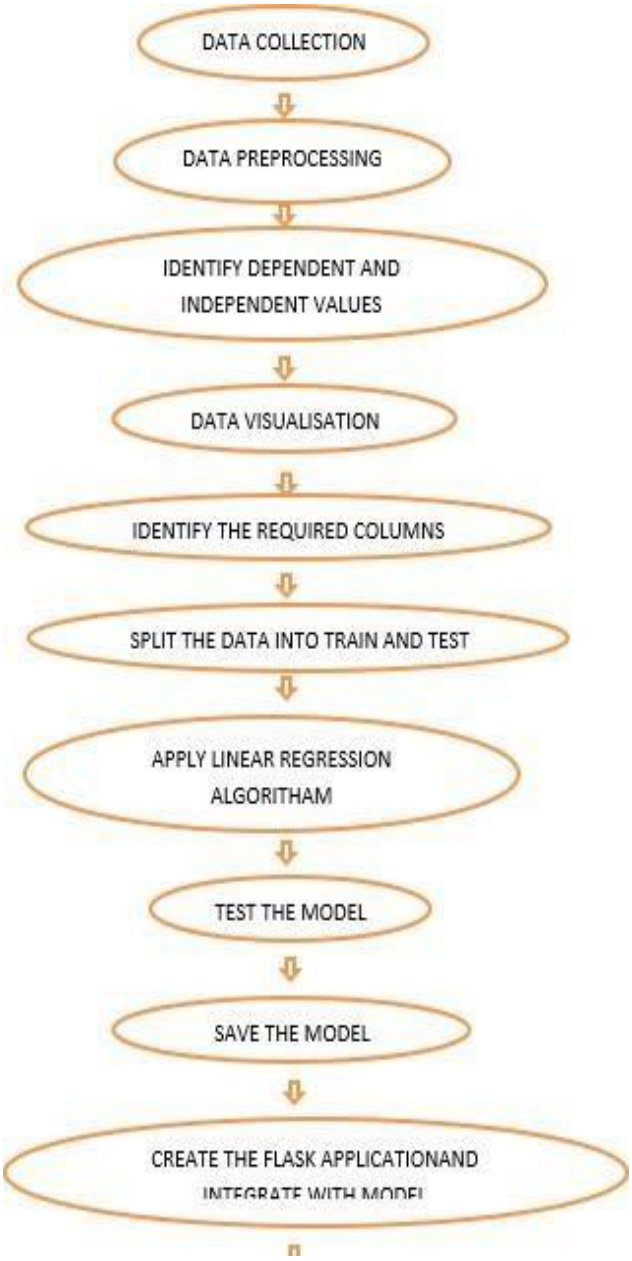

Figure 6: Flowchart

5.3 CLASS DIAGRAM :

Class diagram **describes the attributes and operations of a class and also the constraints imposed on the system**. The class diagrams are widely used in the modeling of objectoriented systems because they are the only UML diagrams, which can be mapped directly with object-oriented languages.

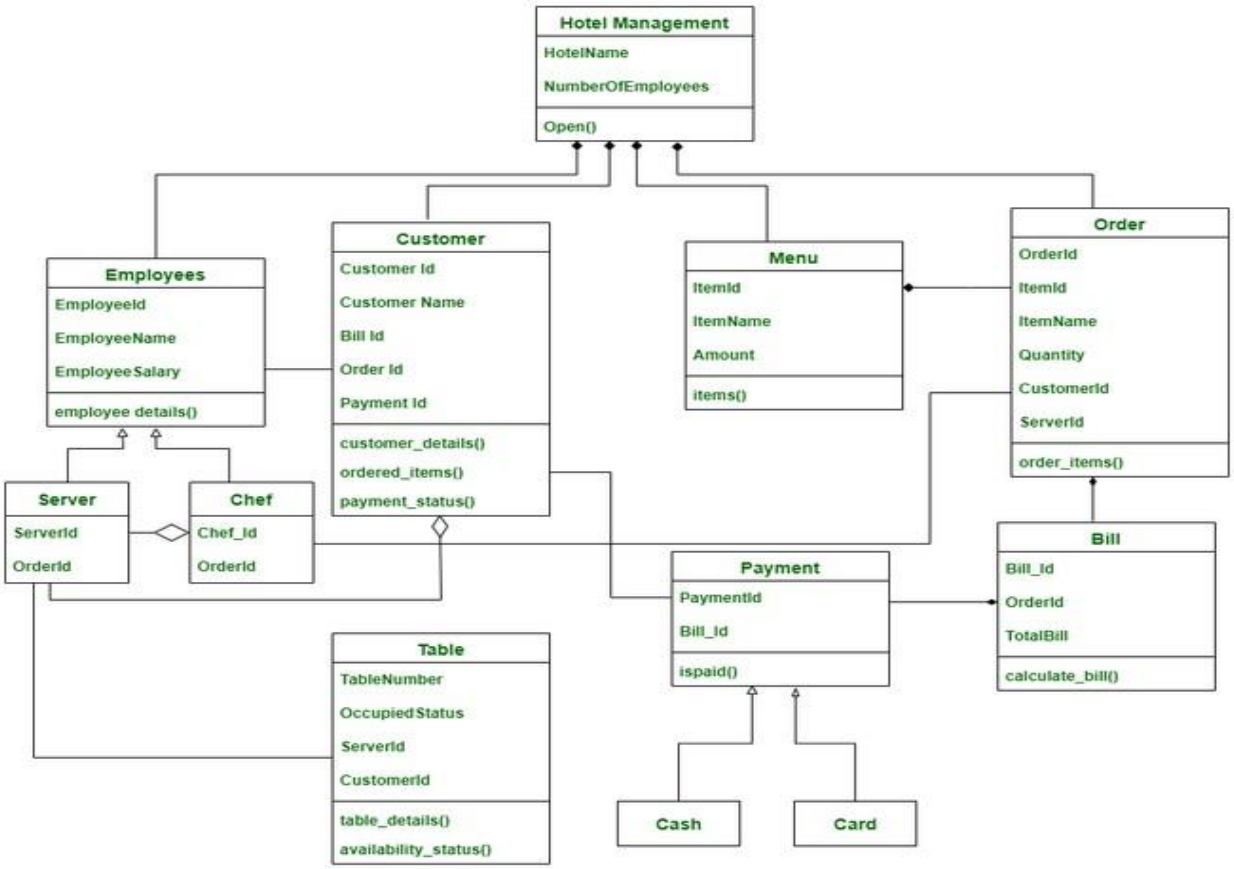

Figure 6: Class Diagram

6.CONCLUSION

In UG Project Phase-1, we have worked on problem statement, literature survey and also done the experimental analyses which are required for the project to move forward. In experimental analysis we have discussed about the machine learning concepts and models and explained the algorithms to be used in the project. We also discussed about the flowcharts, use case diagrams, and class diagrams which are used in the project. Based on the experimental analysis we have designed the model for the project. Entire designing part is involved in UG Project Phase-1.

7.FUTURE SCOPE

UG Project Phase-2 is the extension of UG Project Phase-1. UG Project Phase-2 involves all the coding and implementation of the design which we have retrieved from UG Project Phase-1. All the implementation is done and conclusions will be retrieved in the phase. We will also work on the applications, advantages, and disadvantages of the project in this phase.
Future scope of the project will be also discussed in the UG Project Phase-2.

SENTIMENTAL ANALYSIS OF HOTEL REVIEW

A UG PROJECT PHASE-2 REPORT

Submitted to

JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY, HYDERABAD

In partial fulfilment of the requirements for the award of the degree of

BACHELOR OF TECHNOLOGY

IN

COMPUTER SCIENCE AND ENGINEERING

Submitted by

PADIDALA SRINIKHIL	**19UK1A05M5**
GADDAM DEEKSHITHA	**19UK1A05P2**
DONTHULA ARUN	**20UK1A0506**

Under the esteemed guidance of

Dr. K. SHARMILA REDDY

(Associate Professor)

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

VAAGDEVI ENGINEERING COLLEGE

(Affiliated to JNTUH, HYDERABAD)

Bollikunta, Warangal - 506005

2019-2023

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
VAAGDEVI ENGINEERING COLLEGE
BOLLIKUNTA, WARANGAL -- 506005
2019-2023

CERTIFICATE OF COMPLETION
A UG PROJECT PHASE-2

This is to certify that the UG PROJECT PHASE-2 entitled "**SENTIMENTAL ANALYSIS OF HOTEL REVIEW"** is being submitted by **PADIDALA.SRINIKHIL (19UK1A05P2),GADDAM.DEEKSHITHA(19UK1A05P2),DONTHULA.ARUN(20UK1A0506)** in partial fulfilment of the requirements for the degree of **Bachelor of Technology** in **Computer Science and Engineering to Jawaharlal Nehru Technological University Hyderabad** during the academic year **2022-23,** is a record of work carried out by them under the guidance and supervision.

Project Guide
Dr. K. SHARMILA REDDY
(Associate Professor)

Head of the Department
Dr .R. NAVEEN KUMAR
(Professor)

EXTERNAL

ACKNOWLEDGEMENT

We wish to take this opportunity to express our sincere gratitude and deep sense of respect to our beloved **Dr.P.PRASAD RAO,** Principal, Vaagdevi Engineering College for making us available all the required assistance and for his support and inspiration to carry out this UG Project Phase-2 in the institute.

We extend our heartfelt thanks to **Dr.R.NAVEEN KUMAR**, Head of the Department of CSE, Vaagdevi Engineering College for providing us necessary infrastructure and thereby giving us freedom to carry out the UG Project Phase-2.

We express heartfelt thanks to Smart Bridge Educational Services Private Limited, for their constant supervision as well as for providing necessary information regarding the UG Project Phase-2 and for their support in completing the UG Project Phase-2.

We express heartfelt thanks to the guide, **Dr.K.SHARMILA REDDY** Assistant professor, Department of CSE for her constant support and giving necessary guidance for completion of this UG Project Phase-2.

Finally, we express our sincere thanks and gratitude to my family members, friends for their encouragement and outpouring their knowledge and experience throughout the thesis.

P.SRINIKHIL (19UK1A05M5)
G.DEEKSHITHA (19UK1A05P2)
D.ARUN (20UK5A0506)

TABLE OF CONTENTS:

LIST OF FIGURES PAGE NO

1.INTRODUCTION

1.1 MOTIVATION:

There are many researchers trying to surpass the latest best results and achieve the state-of-the-art in English sentiment analysis by using handcrafted features. This approach may result into overfitting the data. However, sentiment analysis in Czech has not yet been thoroughly targeted by the research community.

Czech as a representative of a inflective language is an ideal environment for the study of various aspects of sentiment analysis (overview or breadth study of sentiment analysis if you will) for inflectional languages. It is challenging because of its very flexible word order and many different word forms.

We conceive this study to deal with several aspects of sentiment analysis. The breadth of this study can lead to more general view and better understanding of sentiment analysis. We can reveal and overcome unexpected obstacles, create necessary evaluation datasets and even come up with new creative solutions to sentiment analysis tasks.

1.2 DEFINITION:

Sentiment analysis (also known as opinion mining or emotion AI) is the use of natural language processing, text analysis, computational linguistics, and biometrics to systematically identify, extract, quantify, and study affective states and subjective information. Sentiment analysis is widely applied to voice of the customer materials such as reviews and survey responses, online and social media, and healthcare materials for applications that range from marketing to customer service to clinical medicine. With the rise of deep language models, such as RoBERTa, also more difficult data domains can be analyzed, e.g., news texts where authors typically express their opinion/sentiment less explicitly.

1.3 OBJECTIVE OF PROJECT:

By the end of this project, you will:

- Know fundamental concepts and techniques used for machine learning.
- Gain a broad understanding of data.
- Have knowledge on pre-processing textual data and classification algorithms.

1.4 PURPOSE:

Recent years have seen rapid growth in online discussion groups and review sites (e.g.www.tripadvisor.com) where a crucial characteristic of a customer's review is their sentiment or overall opinion — for example if the review contains words like 'great', 'best', 'nice', 'good', 'awesome' is probably a positive comment. Whereas if reviews contains words like 'bad', 'poor', 'awful', 'worse' is probably a negative review. However, Trip Advisor's star rating does not express the exact experience of the customer. Most of the ratings are meaningless, large chunk of reviews fall in the range of 3.5 to 4.5 and very few reviews below or above. We seek to turn words and reviews into quantitative measurements. We extend this model with a supervised sentiment component that is capable of classifying a review as positive or negative with accuracy (Section 4). We also determine the polarity of the review that evaluates the review as recommended or not recommended using semantic orientation. A phrase has a positive semantic orientation when it has good associations (e.g., "excellent, awesome") and a negative semantic orientation when it has bad associations (e.g., "terrific, bad"). Next step is to assign the given review to a class, positive or negative, based on the average semantic orientation of the phrases extracted from the review. If the average is positive, the prediction is that the review posted is positive. Otherwise, the prediction is that the item is negative.

2.CODE SNIPPETS

2.1 MODEL CODE:

2.1.1. Data Preprocessing:

Data Pre-processing includes the following main tasks

1.Import the Libraries.

2.Importing the dataset.

3.Checking for Null Values.

4.Data Visualization.

5.Label Encoder.

6.Spilliting Data into Train and Test

```
import pandas as pd

[ ] #read the csv review dataset
    trip = pd.read_csv("hotel_reviews.csv")

[ ] trip.head()

[ ] # Let's create a new data frame

    trip = trip[(trip['Rating']==5)|(trip['Rating']==2)|(trip['Rating']==1)][['Review','Rating']]

    # Lets modify the Rating column
    trip['Rating'] = trip['Rating'].apply(lambda rating: 'Pos' if rating==5 else 'Neg')

[ ] # reseting the index because after removing some rows, the index gets crowded
    trip.reset_index(inplace=True)
    trip.head()

[ ] trip['Rating'].value_counts()

[ ] #Data cleaning and preprocessing
    import re
    import nltk

    from nltk.corpus import stopwords
    from nltk.stem.porter import PorterStemmer
    from nltk.stem import WordNetLemmatizer
```

FIGURE 1: ipynb code describing importing libraries and displaying the few rows from the Set

```
# Lemmatization object
ps = WordNetLemmatizer()
corpus = []
```

```
# Text preprocessing
# keep only text based
# lower all the letters
# split the words
for i in range(0,len(trip)):
    review = re.sub('[^a-zA-Z]'," ",trip['Review'][i])
    review = review.lower()
    review = review.split()
    review = [ps.lemmatize(word) for word in review if not word in stopwords.words('english')]
    review = ' '.join(review)
    corpus.append(review)
```

```
trip.to_csv('tdata.csv', index = False)
```

```
from sklearn.feature_extraction.text import CountVectorizer
cv = CountVectorizer()
```

```
X = cv.fit_transform(corpus).toarray()
```

```
X.shape
```

```
(12268, 34569)
```

```
y = pd.get_dummies(trip['Rating'])
y = y.iloc[:,1].values
y
```

```
array([0, 1, 1, ..., 0, 0, 0], dtype=uint8)
```

```
#train test split
from sklearn.model_selection import train_test_split
X_train, X_test,y_train,y_test = train_test_split(X,y,test_size=0.20,random_state=3)
```

FIGURE 2: ipynb code describing Text processing and splitting datasets for training and testing.

Naive Bayes

```
#Naive bayes classifier

from sklearn.naive_bayes import MultinomialNB
model1 = MultinomialNB().fit(X_train,y_train)

# for the accuracy
model1.score(X_test,y_test)

y_pred = model1.predict(X_test)
```

FIGURE 3: ipynb code describing the Creation of model MultinomialNB from Sklearn.

```
#compare y test and y_pred
#confusion matrix is a 2x2 matrix and it tells,
#how many number of elements are correctly predicted.

from sklearn.metrics import confusion_matrix
confusion_m = confusion_matrix(y_test,y_pred)

confusion_m
```

```
array([[ 574,   69],
       [  44, 1767]])
```

```
#checking accuracy score

from sklearn.metrics import accuracy_score
accuracy = accuracy_score(y_test,y_pred)

accuracy
```

```
0.9539527302363489
```

```
#checking precision score

from sklearn.metrics import precision_score
precision_score(y_test,y_pred)
```

```
0.9624183006535948
```

```
#checking recall score

from sklearn.metrics import recall_score
recall_score(y_test,y_pred)
```

```
0.9757040309221424
```

```
#checking f-beta score

from sklearn.metrics import fbeta_score
fbeta_score(y_test,y_pred,beta=1)
```

```
0.9690156292843433
```

FIGURE 4: ipynb code describes testing of model with test data and finding accuracy.

```
from sklearn.linear_model import LogisticRegression

logreg = LogisticRegression(solver='liblinear')

[ ] model2 = logreg.fit(X_train, y_train)
```

FIGURE 5: ipynb code describing the creation of model LogisticRegression.

```
[ ] y_pred_class = logreg.predict(X_test)

[ ] from sklearn import metrics

    metrics.accuracy_score(y_test, y_pred)

    0.9539527302363489

[ ] metrics.precision_score(y_test, y_pred)

    0.9624183006535948

[ ] metrics.precision_score(y_test, y_pred)

    0.9624183006535948

[ ] metrics.recall_score(y_test, y_pred)

    0.9757040309221424

[ ] metrics.fbeta_score(y_test, y_pred, beta=1)

    0.9690156292843433

[ ] metrics.roc_auc_score(y_test, y_pred)

    0.9341972720707136

[ ] metrics.confusion_matrix(y_test, y_pred)

    array([[ 574,   69],
           [  44, 1767]])
```

FIGURE 6: ipynb code describes testing of new model with test data and finding accuracy.

```
!pip install pickle
# Dump the machine learning model outsite so you can use outsite and not retrain again and again
import pickle

#pickle file for logistic regression

filename = 'logistic_regression_model.pkl'
pickle.dump(model2, open(filename, 'wb'))
pickle.dump(cv, open('tranform_logistic.pkl', 'wb'))

[ ] #pickle file for naive bayes

filename = 'naive_bayes_model.pkl'
pickle.dump(model1, open(filename, 'wb'))
pickle.dump(cv, open('transform_naive.pkl', 'wb'))
```

FIGURE 7: .ipynb code describes importing pickle and creating the checkpoint to save the progess of the model.

2.2 HTML CODE AND PYTHON CODE

2.2. app.py code :

```
from flask import Flask,render_template,url_for,request
import pandas as pd
import pickle
from sklearn.feature_extraction.text import CountVectorizer
from sklearn.naive_bayes import MultinomialNB
import pickle
from sklearn.feature_extraction.text import TfidfVectorizer

# load the model
naive_bayes_model = pickle.load(open('naive_bayes_model.pkl', 'rb'))
logistic_regression_model = pickle.load(open('logistic_regression_model.pkl', 'rb'))
cv_naive=pickle.load(open('tranform_naive.pkl','rb'))
cv_logistic=pickle.load(open('tranform_logistic.pkl','rb'))

app = Flask(__name__)

@app.route('/')
def home():
    return render_template('home.html')

@app.route('/predict',methods=['POST','GET'])
def predict():
    if request.method == 'POST':
        message = request.form['message'].lower()
        data = [message]
        vect_naive = cv_naive.transform(data).toarray()
        vect_logistic = cv_logistic.transform(data).toarray()
        # for naive bayes classifier
        my_nb_prediction = naive_bayes_model.predict(vect_naive)
        nb_percentage = naive_bayes_model.predict_proba(vect_naive)
        if my_nb_prediction==1:
            nb_percentage = nb_percentage[0][1]
        else:
            nb_percentage = nb_percentage[0][0]
        # for logistic regression classifier
        my_lg_prediction = logistic_regression_model.predict(vect_logistic)
        lg_percentage = logistic_regression_model.predict_proba(vect_logistic)
        if my_lg_prediction ==1:
            lg_percentage = lg_percentage[0][1]
        else :
            lg_percentage = lg_percentage[0][0]
    return render_template('result.html',
                message=message,
                my_nb_prediction = my_nb_prediction,
                nb_percentage=nb_percentage,
                my_lg_prediction = my_lg_prediction,
                lg_percentage=lg_percentage
                )

if __name__ == '__main__':
    app.run(debug=True)
```

FIGURE 8: python code used for rendering all HTML pages.

2.1. HOME.HTML:

```
<!DOCTYPE html>
<html>
<head>
    <title>Hotel Review Sentiment Analysis</title>
    <link rel="icon" href="../static/hotel.png">
    <link rel="stylesheet" type="text/css" href="../static/styles.css">
    <link rel="stylesheet" href="../static/w3.css">
    <link rel="stylesheet" type="text/css" href="{{ url_for('static', filename='css/styles.css') }}">
</head>
<body>
    <div class="w3-center w3-padding-16 w3-red w3-text-white w3-xxlarge">
        Hotel Review Sentiment Analysis
    </div>
    <div class="ml-container container w3-center w3-margin-top">
        <form action="{{ url_for('predict')}}" method="POST">
        <textarea name="message" style="outline: none;" placeholder="Enter your review here" rows="6" cols="50" required></textarea>
        <br/>
        <input type="submit" class="btn-info w3-button w3-red w3-xlarge w3-round-large w3-hover-brown" value="Predict">
    </form>
    </div>
</body>
</html>
```

FIGURE 9: home.html page is the code for home page of our Web Application.

```
<!DOCTYPE html>
<html>
<head>
    <title>Hotel Review Sentiment Analysis</title>
    <link rel="icon" href="../static/hotel.png">
    <link rel="stylesheet" type="text/css" href="../static/styles.css">
    <link rel="stylesheet" href="../static/w3.css">
</head>
<body>
<div class="w3-center w3-padding-16 w3-red w3-xxlarge">
        Hotel Review Sentiment Analysis
    </div>
    <div class="results w3-center w3-bold w3-panel">
    <h3 class=" w3-padding-16 w3-blue">"{{message}}"</h3>
    {% if my_nb_prediction == 1%}
        <div class="w3-green padding-large w3-border w3-round-large">
            <h2>Naive Bayes classifier</h2>
            <h2><b>POSITIVE REVIEW <br> with a predicted probability of {{('%.2f'%(nb_percentage*100))}}%</b></h2>
        </div>
        {% elif my_nb_prediction == 0%}
        <div class="w3-red padding-large w3-border w3-round-large">
            <h2>Naive Bayes classifier</h2>
            <h2><b>NEGATIVE REVIEW <br>with a predicted probability of {{('%.2f'%(nb_percentage*100))}}%</b></h2>
        </div>
    {% endif %}
    {% if my_lg_prediction == 1%}
        <div class="w3-green padding-large w3-round-large">
            <h2>Logistic Regression classifier</h2>
            <h2><b>POSITIVE REVIEW <br> with a predicted Probability of {{('%.2f'%(lg_percentage*100))}}%</b></h2>
        </div>
        {% elif my_lg_prediction == 0%}
        <div class="w3-red padding-large w3-round-large">
            <h2>Logistic Regression classifier</h2>
            <h2><b>NEGATIVE REVIEW <br>with a predicted Probability of {{('%.2f'%(lg_percentage*100))}}%</b></h2>
        </div>
    {% endif %}
    </div>
</body>
</html>
```

FIGURE 10: result.html page is the code for result page of our Web Application

3.CONCLUSION

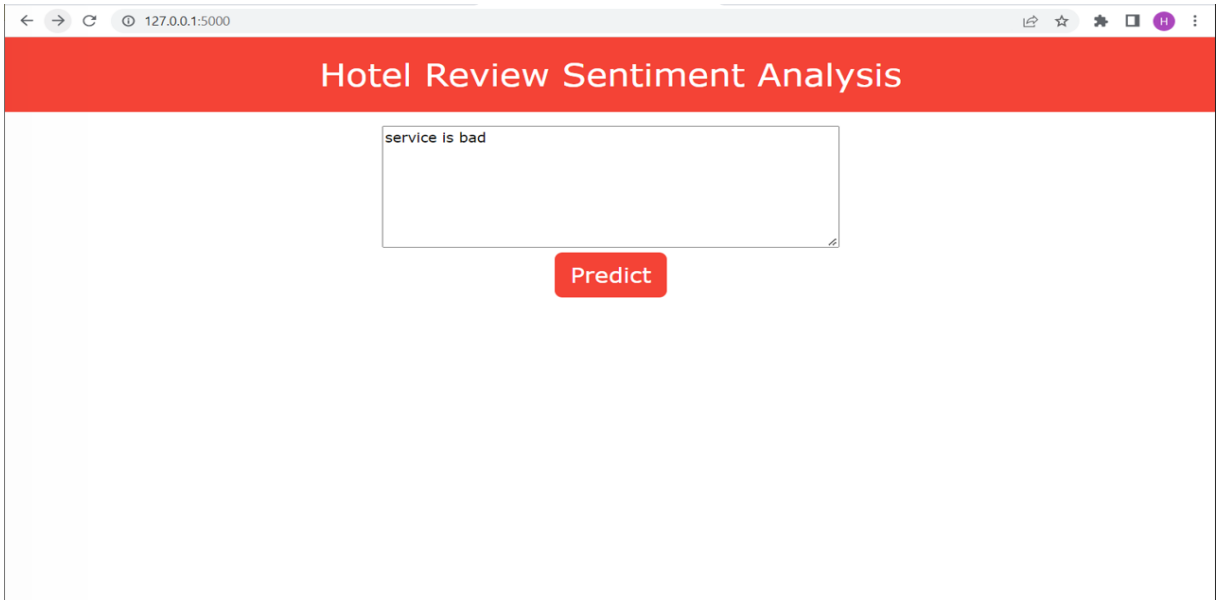

FIGURE 11: The Hotel Review Sentiment Analysis Homepage Provides A Textbox With a Button named Predict You Enter The review and Click Predict.

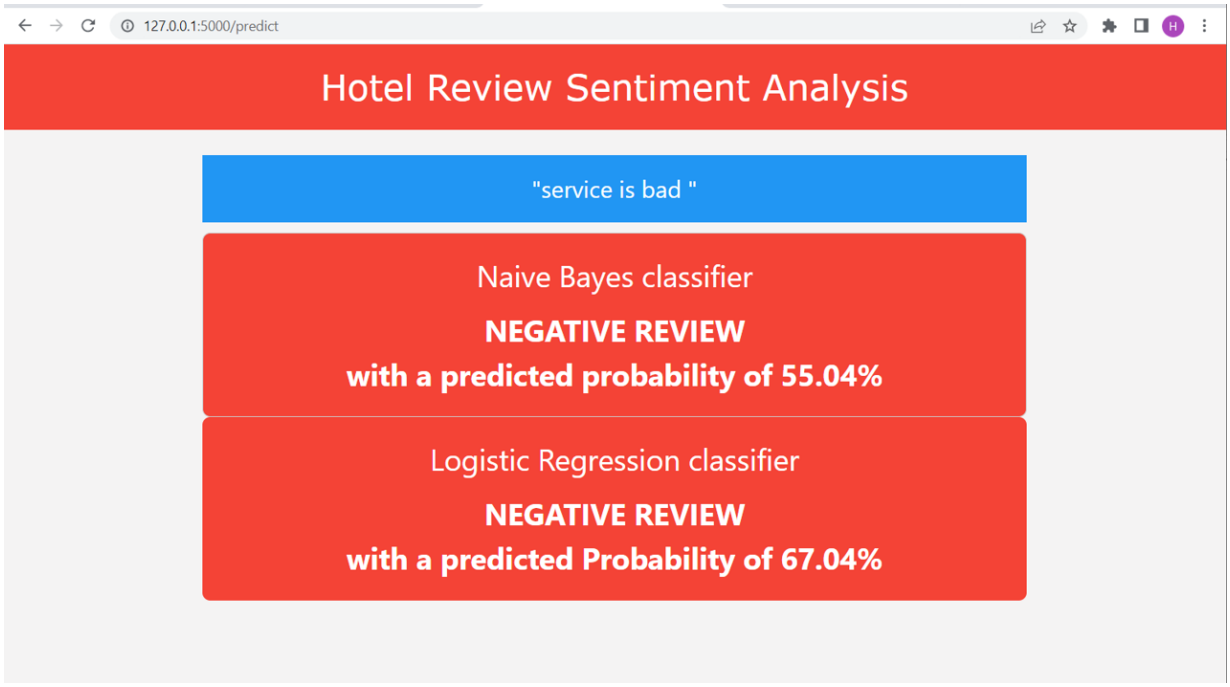

FIGURE 12: Predicting the User review is good or bad using the trained Models the Model give results in probability (BAD).

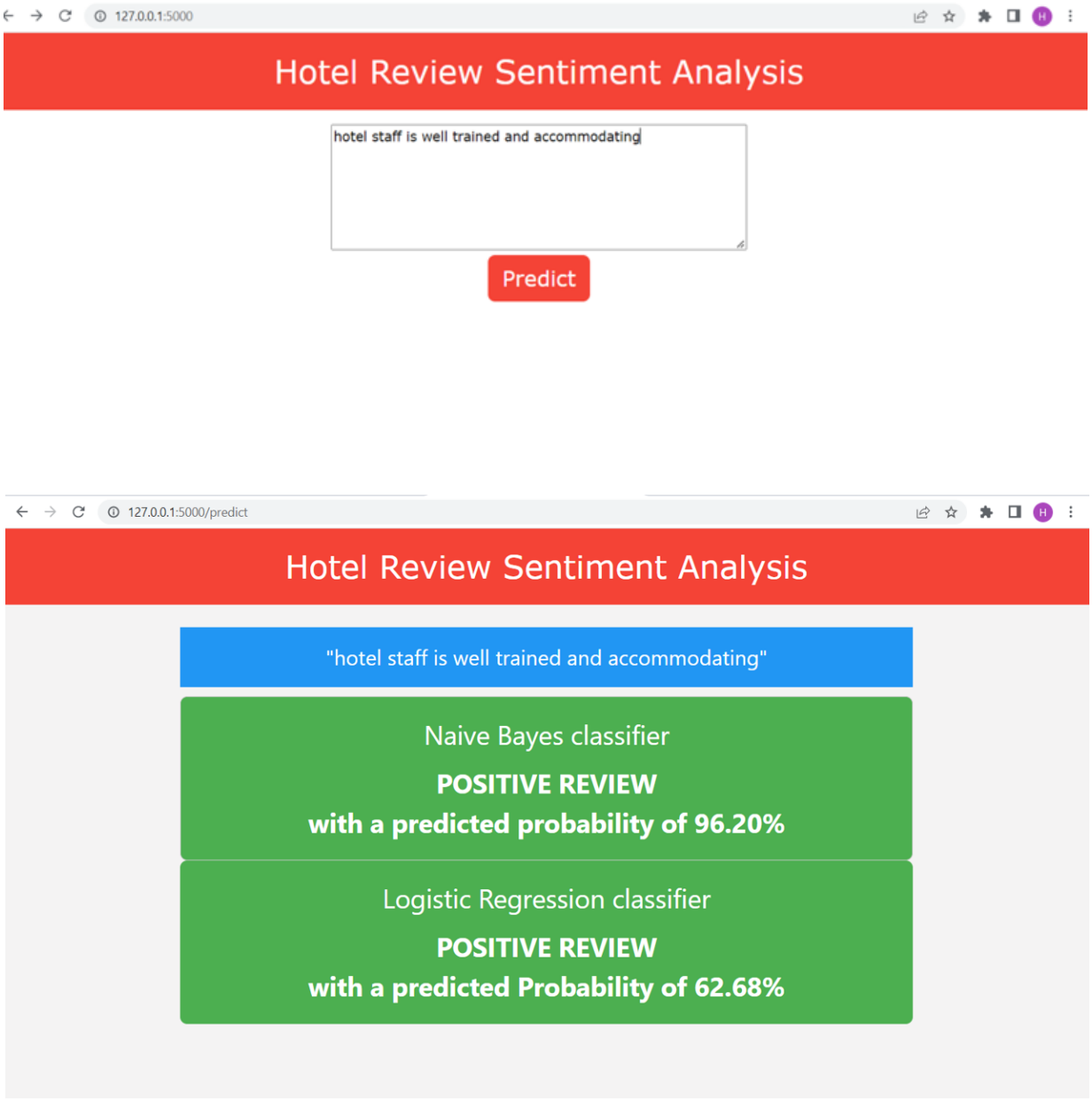

FIGURE 13: Predicting the User review is good or bad using the trained Models the Model give results in probability (GOOD).

5.ADVANTAGES

- They Predict Faster and More Accurately
- The model is trained to predict the bad and negative probability of the Reviews
- Thus with a rapid growth in the deep learning architecture, an objective of reviews of how much good and bad is easy to predict
- This will allow a non-contact, easy to use and low cost that can be performed routinely anywhere in the world
- The Trained model and also used anywhere remotely

6.DISADVANTAGES

- For Training We need some good dataset from reputed sites like trip advisor
- It is not completely possible to use only raw text as input for making predictions
- We need to extract the relevant features from this raw source of data
- This kind of data can often come as a good complementary source in data science projects in order to extract more learning features and increase the predictive power of the models
- The result of accuracy is only 89%

7.FUTURE SCOPE

Know fundamental concepts and techniques used for machine learning.Gain a broad understanding of data.Have knowledge on pre-processing textual data and classification algorithms.This can be used to help the websites to understand The user reviews with a single click which helps more accurately and improve their websites and hotel amenities according to the user needs.This will allow a noncontact, easy to use and low cost that can be performed routinely anywhere in the world

Enhancements that can be made in the future:

In this project we have used the deep learning with NLTK, Naïve Bayes Classifier , Logistic regression classifier but Sentiment analysis and AI could be the answer to mental health treatment, according to **TDWI** With the ability to read emotions and learn responses, it is believed to be possible.Some think that it might be dangerous to use AI in the mental health field. However, this trend is popping up more as a serious consideration. Consumers are the most important part of a business. With unhappy customers, a company can receive a bad reputation. Sentiment analysis can help with monitoring customer service, and experience. For instance, using AI technology to analyze customer feedback and customer service exchanges, a company can adjust their service to improve customer satisfaction and loyalty.

8.BIBILOGRAPHY

- H X Shi and X J Li "A sentiment analysis model for hotel reviews based on supervised learning," in in International Conference on Machine Learning and Cybernetics China 2011
- P Goyal "What is Laplacian smoothing and why do we need it in a Naive Bayes classifier?," Quora 27 September 2017 [Online] Available:https://www.quora.com/What-is-Laplacian-smoothing-and-why-do-we-need-it-in-a-Naive-Bayes-classifier [Accessed 22 April 2018]
- A Goel J Gautam and S Kumar "Real time sentiment analysis of tweets using naïve bayes," 2nd International Conference on Next Generation Computing Technologies (NGCT) pp 257-261 2016
- "6 Tren Wisata Utama Tahun 2016," tripadvisor 14 December 2015 [Online] Available:https://www.tripadvisor.co.id/TripAdvisorInsights/w665 [Accessed 19 February 2018]
- T Ghorpade and L Ragha "Featured Based Sentiment Classification for Hotel Reviews using NLP and Bayesian Classification," in international conference on communication , information & computing technology (ICCICT) Mumbai india 2012

9.HELP FILE

PROJECT EXECUTION:

STEP-1: Go to Start,Open GOOGLE CHROME.

STEP-2: After Opening GOOGLE CHROME,Then Search For GOOGLE COLAB.

STEP-3: Open “Major project code” IPYNB file.

STEP-4: Then run all the cells.

STEP-5: All the data preprocessing, training and testing, model building, accuracy of the model can be showcased.

STEP-6: And a pickle file will be generated.

STEP-7: Create a Folder named FLASK . Extract the pickle file into this Flask Folder.

STEP-8: Extract all the html files (home.html, index1.html, indexnew.html, result.html) and python file(app.py) into the FLASK Folder.

STEP-9: Then Open COMMAND PROMPT.

STEP-10: After Opening follow the below steps: cd File Path□click enter python app.py□click enter (We could see running of files).

STEP-11: Then open BROWSER, at the URL area type ―localhost:5000”.

STEP-12: Home page of the project will be displayed.

STEP-13: Click on ―Go to Predict”. Directly it will be navigated to index page.

STEP-14:A index page will be displayed where the user needs to give the inputs and then click on ―Predict” and see the result/prediction on the web is positive or negative.