BEHAVIOUR ANALYSIS IN CHAT MESSAGE

Mini Project Report

Submitted by

KAVYA SUDHEER

Submitted in partial fulfillment of the requirements for the award of the degree of

Master of Computer Applications
Of
A P J Abdul Kalam Technological University

FEDERAL INSTITUTE OF SCIENCE AND TECHNOLOGY (FISAT)®
ANGAMALY-683577, ERNAKULAM(DIST)
MARCH 2022

DECLARATION

I **Kavya Sudheer**, hereby declare that the report of this project work, submitted to the Department of Computer Applications, Federal Institute of Science and Technology (**FISAT**), Angamaly in partial fulfillment of the award of the degree of Master of Computer Application is an authentic record of our original work.

The report has not been submitted for the award of any degree of this university or any other university.

Date : 04-03-2022

Place: Angamaly

FEDERAL INSTITUTE OF SCIENCE AND TECHNOLOGY (FISAT)®

ANGAMALY, ERNAKULAM-683577

DEPARTMENT OF COMPUTER APPLICATIONS

CERTIFICATE

This is to certify that the project report titled **"BEHAVIOUR ANALYSIS IN CHAT MESSAGE"** submitted by **KAVYA SUDHEER** towards partial fulfillment of the requirements for the award of the degree of Master of Computer Applications is a record of bonafide work carried out by them during the year 2022.

Project Guide **Head of the Department**

Submitted for the viva-voice held on ..... at .....

Examiner1 :

Examiner2 :

ACKNOWLEDGEMENT

Gratitude is a feeling which is more eloquent than words, more silent than silence. To complete this project work we needed the direction, assistance and cooperation of various individuals, which is received in abundance with the grace of God.

We hereby express our deep sense of gratitude to **Dr. Manoge George**, Principal of FISAT and **Dr. C Sheela**, Vice principal of FISAT, for allowing us to utilize all the facilities of the college.

Our sincere thanks to **Dr. Deepa Mary Mathew**, Head of the department of Computer Applications FISAT and scrum master and our Internal guide for this project **Dr.Rakhi Venugopal** for giving as valuable guidance, constructive suggestions and comment during our project work. We also express our boundless gratitude to all the lab faculty members for their guidance.

Finally we wish to express a whole heart-ed thanks to our parents, friends and well-wishers who extended their help in one way or other in preparation of our project. Besides all, We thank GOD for everything.

ABSTRACT

Nowadays, the mode of communication is mainly through messages.. It has been widely used by all, especially among the business people and youngsters.
Using several analyzing tools, users can analyse the group chat or personal chat for several purpose.
The aim of our project is people's behaviour analysis in chat message for any platform based on their chat.Using this system,we Analysis the messages for people's with the help of Communication platform.Machine learning algorithms are used for this project and the works based on the NLP technique.By using NLP,the system can easily identify ,whether the message is either suspecious or fair should be given based on their chat.This project has many use cases like the parent, who wants to analyze their child chat; the police, who want to get valuable information from culprit chat; the business people, who wants to know the status of the business in the group chat.
In this we use Support Vector Algorithm. A support vector machine is a discriminative classifier defined by hyper-plane that seprates diffrent classes.In other words,given a set of training data to the algorithm,SVM outputs an optimal hyperplane that can be used for further classification of test data.

Contents

Chapter 1

INTRODUCTION

The present situation a lot of information has been conveyed through watsapp.It is a platform of instant message and voice-over-internet protocol service.Using several analyzing tools,users can analyse the watsapp group chat or personal chat. This project work is perform a flirt analysis and time analysis. This project has many use cases like parent, who wants to analyze their child chat; the police, who want to get valuable information from culprit chat; the business people, who wants to know the status of the business in the group chat. Using the Deep Learning Model (NLP),sentimental analysis has been performed for each text.This helps to find the state of mind of the chatters. It also calculates the number of positive and negative statements for each text using the text mining concept.

Machine learning algorithms are used for this project and the works based on the NLP technique.By using NLP,the sys- tem can easily identify ,whether the message is either suspecious or fair should be given based on their chat.This project has many use cases like the parent, who wants to analyze their child chat; the police, who want to get valuable information from culprit chat; the business people, who wants to know the status of the busi- ness in the group chat.In this we use Support Vector Algorithm. A support vector machine is a dis- criminative classifier defined by hyper-plane that seprates diffrent classes.

Chapter 2

PROOF OF CONCEPT

2.1 Existing System

The usage of social media networks now became a common mode of information sharing. A large set of users were now adapting to this model of the technological era.For the privacy and safety of the users, as of now mostly our chat conversation is through WhatsApp and by the sentimental analyze model we can analyze and we can conclude whether the chat we are going safe or going correctly.
It is possible through have a telephonic communication,but there are a number of limitations regarding it. Sometimes,instant reply may not be received for the patients.

2.2 Proposed System

This research work is intended to perform a flirt analysis. This system has many use cases like the parent, who wants to analyze their child chat; the police, who want to get valuable information from culprit chat; the business people, who wants to know the status of the business in the group chat. Using the Natural language processing. sentimental analysis has been performed for each text. This helps to find the state of mind of the chatters. Further, this research work calculates the

number of positive and negative statements that are used by each person in the text by using the text mining concept.

2.3 Objectives

The privacy and safety of the users, as of now mostly our chat conversation is through WhatsApp and by the sentimental analyze model we can analyze and we can conclude whether the chat we are going safe or going correctly and also the parents can track their child move and also the business personality can also check how they got their review in the group works. The text mining concept is also used for mining the text data, which was used in the Whatsapp conversation.

For every process of calculation, the data will be collected and then for the pre-processing technique, the data collected was given to certain classifying techniques for performing the sentimental analysis . The individual emotions were taken for the analyzing purpose and then it was made for the test and by which they get the information across the social platform and the required response was noted. Based on the sentimental analysis results we can come to know about the analysis report whether the views were positive and loved by everyone or else negative or they are neutral from which they will conclude whether it was liked by everyone or not .

Chapter 3

IMPLEMENTATION

The dataset we took from Kaggle.com. Shape of dataset (5572).We have isolated the label column from the data frame. Preprocessing steps are applied and then we split the dataset into train and test data set.Feature extraction is done after completing pre-processing on the given input data.The extracted features are then given as input to the Classification Algorithm(SVM).The output of this classification algorithm shows if the given message is fair or suspecious.This is a supervised approach because it uses a tagged or labeled training data set.

3.1 TOOLS OR PROGRAMMING LANGUAGE

•FRONT END :
– Html
• BACK END :
– Python

NLP

Natural language processing (NLP) refers to the branch of computer science—and more specifically, the branch of artificial intelligence—concerned with giving computers the ability to understand text and spoken words in much the same way

human beings can. The Python programming language provides a wide range of tools and libraries for attacking specific NLP tasks. Many of these are found in the Natural Language Toolkit, or NLTK, an open source collection of libraries, programs, and education resources for building NLP programs. The NLTK includes libraries for many of the NLP tasks listed above, plus libraries for subtasks, such as sentence parsing, word segmentation, stemming and lemmatization (methods of trimming words down to their roots), and tokenization (for breaking phrases, sentences, paragraphs and passages into tokens that help the computer better understand the text). It also includes libraries for implementing capabilities such as semantic reasoning, the ability to reach logical conclusions based on facts extracted from text.

3.2 FRAME WORK

FLASK

Flask is a micro web framework written in Python. It is classified as a microframework because it does not require particular tools or libraries.It has no database abstraction layer, form validation, or any other components where pre-existing third-party libraries provide common functions. However, Flask supports extensions that can add application features as if they were implemented in Flask itself. Extensions exist for object-relational mappers, form validation, upload handling, various open authentication technologies and several common framework related tools.

Flask is a web application framework written in Python. It was developed by Armin Ronacher, who led a team of international Python enthusiasts called Poocco. Flask is based on the Werkzeg WSGI toolkit and the Jinja2 template engine.Both are Pocco projects.

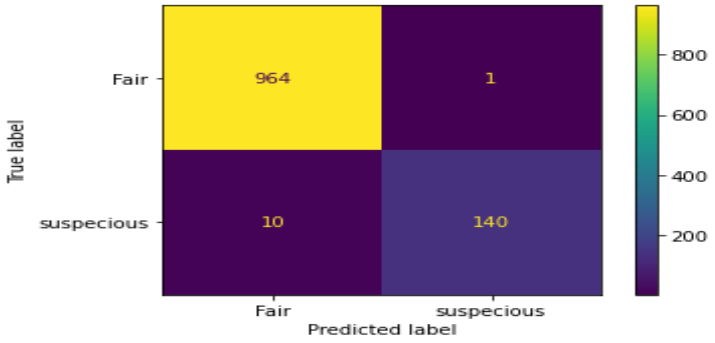
3.3 ALGORITHM

Support Vector Machine

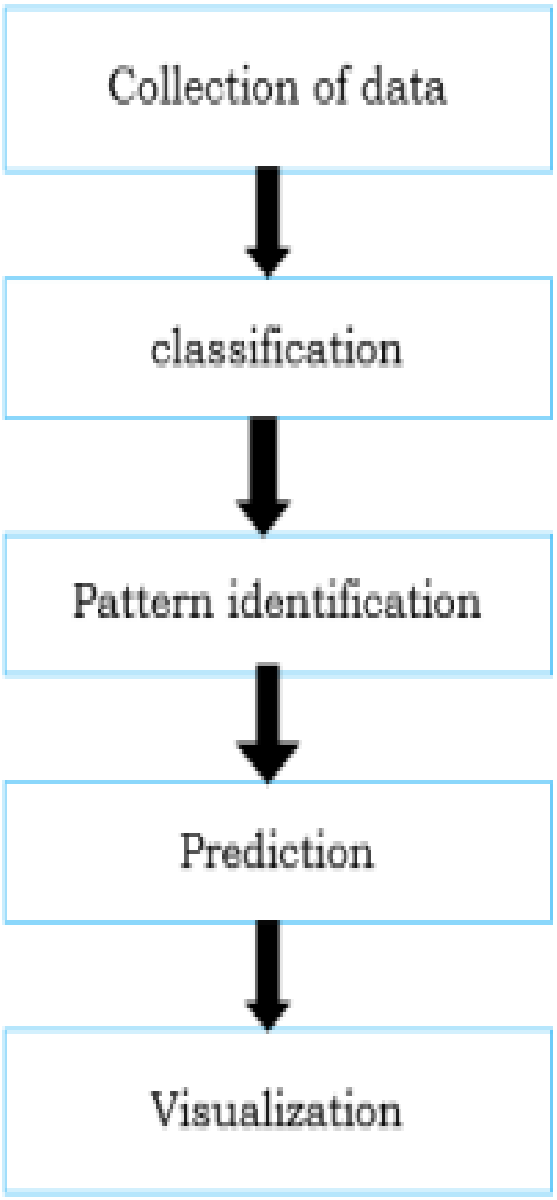
Support Vector Machine or SVM is one of the most popular Supervised Learning algorithms, which is used for Classification as well as Regression problems. However, primarily, it is used for Classification problems in Machine Learning.

The goal of the SVM algorithm is to create the best line or decision boundary that can segregate n-dimensional space into classes so that we can easily put the new data point in the correct category in the future. This best decision boundary is called a hyperplane.

SVM chooses the extreme points/vectors that help in creating the hyperplane. These extreme cases are called as support vectors, and hence algorithm is termed as Support Vector Machine.



3.4 ARCHITECTURE



3.5 MODULES

Data Preprocessing

The data collected had to be preprocessed since it had traces of unstructured contents. It basically meant we needed to clean or trim the data in order to obtain a higher accuracy. There were various steps that were needed to be followed for preprocessing the data such as data cleaning, stop word removal, tokenization. The term stop words mean those words that don't give any helpful data to decide in which category a text should be classified. For facilitating the further processes with the motive of not distin- guishing among capital letters and lowercase letters, we transformed the whole data into lower case. Furthermore tokenization had to be practiced on these text contents to facilitate the feature extraction step. Tokenization can be defined as a way of separating or isolating every word that compiles in a document or even a conversation.

Feature Extraction

A primary concern of ours was to examine the effect of feature set size on text categorization effectiveness. All potential features were ranked for each category by expected mutual information between assignment opposite effects on the properties of a text representation, which led us to investigate combining the two techniques. However, the small size of standard text retrieval test collections, and the variety of approaches available for query interpretation, made it difficult to study purely representational issues in text retrieval experiments. In this work we examine indexing language properties using two text categorization data sets. We obtain much clearer results, as well as producing a new text categorization method capable of handing multiple, overlapping categories.

Training the Model

A training model is a dataset that is used to train an ML algorithm. It consists of

the sample output data and the corresponding sets of input data that have an influence on the output. The training model is used to run the input data through the algorithm to correlate the processed output against the sample output. The result from this correlation is used to modify the model.

Evaluation

Model evaluation techniques in machine learning are helping us to find a bet- ter model among all other models in machine learning. It is simply the selection of machine learning models or measuring the performance of machine learning models.

3.6 DATASET

The dataset used to test the efficiency of the model is produced by kaggle, containing 5572 data.

• Attributes of dataset :

– Label

– Content

	Label	Content
2	Fair	Go until jurong point, crazy.. Available only in bugis n great world la e buffet... Cine there got amore wat...
3	Fair	Ok lar... Joking wif u oni...
4	suspecious	Free entry in 2 a wkly comp to win FA Cup final tkts 21st May 2005. Text FA to 87121 to receive entry question(std txt rate)T&C's apply 08452810075over18's
5	Fair	U dun say so early hor... U c already then say...
6	Fair	Nah I don't think he goes to usf, he lives around here though
7	suspecious	FreeMsg Hey there darling it's been 3 week's now and no word back! I'd like some fun you up for it still? Tb ok! XxX std chgs to send, å£1.50 to rcv
8	Fair	Even my brother is not like to speak with me. They treat me like aids patent.
9	Fair	As per your request 'Melle Melle (Oru Minnaminunginte Nurungu Vettam)' has been set as your callertune for all Callers. Press *9 to copy your friends Callertune
10	suspecious	WINNER!! As a valued network customer you have been selected to receivea å£900 prize reward! To claim call 09061701461. Claim code KL341. Valid 12 hours only.
11	suspecious	Had your mobile 11 months or more? U R entitled to Update to the latest colour mobiles with camera for Free! Call The Mobile Update Co FREE on 08002986030
12	Fair	I'm gonna be home soon and i don't want to talk about this stuff anymore tonight, k? I've cried enough today.
13	suspecious	SIX chances to win CASH! From 100 to 20,000 pounds txt> CSH11 and send to 87575. Cost 150p/day, 6days, 16+ TsandCs apply Reply HL 4 info
14	suspecious	URGENT! You have won a 1 week FREE membership in our å£100,000 Prize Jackpot! Txt the word: CLAIM to No: 81010 T&C www.dbuk.net LCCLTD POBOX 4403LDNW1A7RW18
15	Fair	I've been searching for the right words to thank you for this breather. I promise i wont take your help for granted and will fulfil my promise. You have been wonderful and a blessing at all
16	Fair	I HAVE A DATE ON SUNDAY WITH WILL!!
17	suspecious	XXXMobileMovieClub: To use your credit, click the WAP link in the next txt message or click here>> http://wap. xxxmobilemovieclub.com?n=QJKGIGHJJGCBL
18	Fair	Oh k...i'm watching here:)
19	Fair	Eh u remember how 2 spell his name... Yes i did. He v naughty make until i v wet.
20	Fair	Fine if thatåÕs the way u feel. ThatåÕs the way its gota b
21	suspecious	England v Macedonia - dont miss the goals/team news. Txt ur national team to 87077 eg ENGLAND to 87077 Try:WALES, SCOTLAND 4txt/Ì¼1.20 POBOXox36504W45WQ 16+
22	Fair	Is that seriously how you spell his name?
23	Fair	I‰Û÷m going to try for 2 months ha ha only joking

Chapter 4

RESULT ANALYSIS

Accuracy is often the most used metric representing the percentage of correctly predicted observations, either true or false. To calculate the accuracy of a model performance, the following equation can be used: In most cases, high accuracy value represents a good model, but considering the fact that we are training a classification model in our case, an article that was predicted as true while it was actually false (false positive) can have negative consequences; similarly, if an article was predicted as false while it contained factual data, this can create trust issues. Accuarcy = TP+TN/TP+TN+FP+FN

Confusion Matrix

A Confusion matrix is an N x N matrix used for evaluating the performance of a classification model, where N is the number of target classes. The matrix compares the actual target values with those predicted by the machine learning model. This gives us a holistic view of how well our classification model is performing and what kinds of errors it is making. For a binary classification problem, we would have a 2 x 2 matrix as shown below with 4 values:

1. TP = True Positives
2. FP = False Positives
3. TN = True Negatives
4. FN = False Negatives

Chapter 5

CONCLUSION AND FUTURE SCOPE

5.1 Conclusion

The present situation a lot of information has been conveyed through watsapp.It is a platform of instant message and voice-over-internet protocol service.Using several analyzing tools,users can analyse the watsapp group chat or personal chat. This project work is perform a flirt analysis and time analysis. This project has many use cases like parent, who wants to analyze their child chat; the police, who want to get valuable information from culprit chat; the business people, who wants to know the status of the business in the group chat.In this paper the sentimental flirt analysis method was given out the respective result can obtained.

5.2 Future Scope

In this paper the sentimental flirt analysis methods was implemented in the all the entire chats and was given out the respective result can be obtained The entire methodology can be implemented but for the storage the cost was little high for using the cloud storage and can be reduced in the future works The respective

chat messages can be taken out from various users with the approval of the respective users it was having some delay in execution can be used effectively in future works.

Chapter 6

APPENDIX

In [1]:
```
import pandas as pd
from pandas import read_excel
```

In [2]:
```
trainData=read_excel('messages.xlsx')
dfhead=trainData.head()
dfhead
```

Out[2]:

	Label	Content
0	Fair	Go until jurong point, crazy.. Available only ...
1	Fair	Ok lar... Joking wif u oni...
2	suspecious	Free entry in 2 a wkly comp to win FA Cup fina...
3	Fair	U dun say so early hor... U c already then say...
4	Fair	Nah I don't think he goes to usf, he lives aro...

In [3]:
```
trainData.head(5)
```

Out[3]:

	Label	Content
0	Fair	Go until jurong point, crazy.. Available only ...
1	Fair	Ok lar... Joking wif u oni...
2	suspecious	Free entry in 2 a wkly comp to win FA Cup fina...
3	Fair	U dun say so early hor... U c already then say...
4	Fair	Nah I don't think he goes to usf, he lives aro...

In [4]:
```
from bs4 import BeautifulSoup
import re
```

In [5]:
```
def html_tag(phrase):
    http_remove = re.sub(r"http\S+", "",phrase)
    html_remove = BeautifulSoup(http_remove, 'lxml').get_text()
    return html_remove
```

In [6]:
```
def decontracted(phrase):
    # specific
    phrase = re.sub(r"won't", "will not", phrase)
    phrase = re.sub(r"can\'t", "can not", phrase)

    # general
    phrase = re.sub(r"n\'t", " not", phrase)
    phrase = re.sub(r"\'re", " are", phrase)
    phrase = re.sub(r"\'s", " is", phrase)
    phrase = re.sub(r"\'d", " would", phrase)
    phrase = re.sub(r"\'ll", " will", phrase)
    phrase = re.sub(r"\'t", " not", phrase)
    phrase = re.sub(r"\'ve", " have", phrase)
    phrase = re.sub(r"\'m", " am", phrase)
    return phrase
```

In [7]:
```
stopwords= set(['br', 'the', 'i', 'me', 'my', 'myself', 'we', 'our', 'ours', 'ourselves', 'you', "you're", "you've",\
            "you'll", "you'd", 'your', 'yours', 'yourself', 'yourselves', 'he', 'him', 'his', 'himself', \
            'she', "she's", 'her', 'hers', 'herself', 'it', "it's", 'its', 'itself', 'they', 'them', 'their',\
            'theirs', 'themselves', 'what', 'which', 'who', 'whom', 'this', 'that', "that'll", 'these', 'those', \
            'am', 'is', 'are', 'was', 'were', 'be', 'been', 'being', 'have', 'has', 'had', 'having', 'do', 'does', \
            'did', 'doing', 'a', 'an', 'the', 'and', 'but', 'if', 'or', 'because', 'as', 'until', 'while', 'of', \
            'at', 'by', 'for', 'with', 'about', 'against', 'between', 'into', 'through', 'during', 'before', 'after',\
            'above', 'below', 'to', 'from', 'up', 'down', 'in', 'out', 'on', 'off', 'over', 'under', 'again', 'further',\
            'then', 'once', 'here', 'there', 'when', 'where', 'why', 'how', 'all', 'any', 'both', 'each', 'few', 'more',\
            'most', 'other', 'some', 'such', 'only', 'own', 'same', 'so', 'than', 'too', 'very', \
            's', 't', 'can', 'will', 'just', 'don', "don't", 'should', "should've", 'now', 'd', 'll', 'm', 'o', 're', \
            've', 'y', 'ain', 'aren', "aren't", 'couldn', "couldn't", 'didn', "didn't", 'doesn', "doesn't", 'hadn',\
            "hadn't", 'hasn', "hasn't", 'haven', "haven't", 'isn', "isn't", 'ma', 'mightn', "mightn't", 'mustn',\
            "mustn't", 'needn', "needn't", 'shan', "shan't", 'shouldn', "shouldn't", 'wasn', "wasn't", 'weren', "weren't", \
            'won', "won't", 'wouldn', "wouldn't"])
```

In [8]:
```
from tqdm import tqdm
processed_review = []
for i in tqdm(trainData["Content"].values):
    sentance = html_tag(i)
    sentance = decontracted(sentance)
    sentance = re.sub("\S*\d\S*", "", sentance)
    sentance = re.sub('[^A-Za-z]+', ' ', sentance)
    sentance = " ".join(i.lower() for i in sentance.split() if i.lower() not in stopwords)
    processed_review.append(sentance)
```

```
100%|██████████| 5571/5571 [00:01<00:00, 5026.83it/s]
```

In [9]:
```
trainData["Cleantext"] = processed_review
trainData.head(5)
```

Out[9]:

	Label	Content	Cleantext
0	Fair	Go until jurong point, crazy.. Available only ...	go jurong point crazy available bugis n great ...
1	Fair	Ok lar... Joking wif u oni...	ok lar joking wif u oni
2	suspecious	Free entry in 2 a wkly comp to win FA Cup fina...	free entry wkly comp win fa cup final tkts may...
3	Fair	U dun say so early hor... U c already then say...	u dun say early hor u c already say
4	Fair	Nah I don't think he goes to usf, he lives aro...	nah not think goes usf lives around though

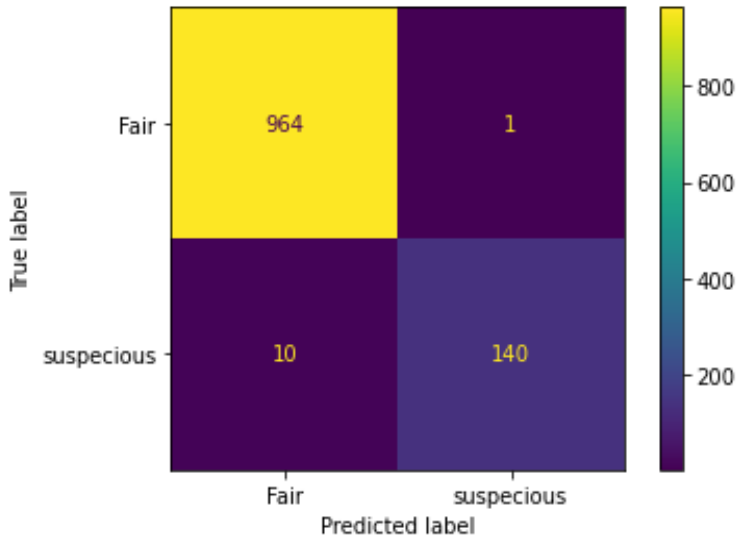
In [10]:
```
from sklearn.feature_extraction.text import TfidfVectorizer
# Create feature vectors
vectorizer = TfidfVectorizer(min_df = 5,
                             max_df = 0.8,
                             sublinear_tf = True,
                             use_idf = True)
train_vectors = vectorizer.fit_transform(trainData['Cleantext'])
```

In [11]:
```
from sklearn.model_selection import train_test_split
X_train, X_test, y_train, y_test = train_test_split(train_vectors, trainData['Label'], test_size = 0.20, random_state = 0)
```

In [12]:
```
tot=train_vectors.shape[0]
tot
```

Out[12]: 5571

```
In [13]: trsize=X_train.shape[0]
         trsize

Out[13]: 4456

In [14]: tstsize=X_test.shape[0]
         tstsize

Out[14]: 1115

In [15]: from sklearn.metrics import plot_confusion_matrix
         from sklearn.metrics import confusion_matrix
         import matplotlib.pyplot as plt

In [16]: from sklearn import svm
         classifier_linear = svm.SVC(kernel='linear')
         classifier_linear.fit(X_train, y_train)
         prediction_linear = classifier_linear.predict(X_test)

In [17]: from sklearn.metrics import accuracy_score
         acc_svm=accuracy_score(y_test, prediction_linear)
         print(acc_svm)

         0.9901345291479821

In [18]: from sklearn.metrics import plot_confusion_matrix
         import matplotlib.pyplot as plt
         plot_confusion_matrix(classifier_linear, X_test, y_test)
         plt.savefig('static\\acc.png')
```



In [19]:
```
Message= 'SMS AUCTION - A BRAND NEW Nokia 7250 is up 4 auction today! Auction is FREE 2 join & take part! Txt NOKIA to 86021 now!
trainData=read_excel('messages.xlsx')
df2 = {'Label': 'Fair', 'Content': Message}
trainData = trainData.append(df2, ignore_index = True)
processed_review = []
for i in trainData["Content"].values:
    sentance = html_tag(i)
    sentance = decontracted(sentance)
    sentance = re.sub("\S*\d\S*", "", sentance)
    sentance = re.sub('[^A-Za-z]+', ' ', sentance)
    sentance = " ".join(i.lower() for i in sentance.split() if i.lower() not in stopwords)
    processed_review.append(sentance)
trainData["Cleantext"] = processed_review
vectorizer = TfidfVectorizer(min_df = 5,
                             max_df = 0.8,
                             sublinear_tf = True,
                             use_idf = True)
train_vectors = vectorizer.fit_transform(trainData['Cleantext'])
X_test1=train_vectors[train_vectors.shape[0]-1]
# y_score = model.predict(X_test)
y_score = classifier_linear.predict(X_test1)
res="This Message is :"+ y_score[0]
print(res)
```

```
This Message is :suspecious
```

In [ ]:
```
from flask import Flask,redirect,render_template,request
app=Flask(__name__)

@app.route("/tst")
def test():
    return render_template("testview.html",tssize=tstsize,accdt=acc_svm)
@app.route("/mn",methods=['GET','POST'])
def mn():
    if request.method=="POST":
        Message= request.form['msg']
        trainData=read_excel('messages.xlsx')
        df2 = {'Label': 'Fair', 'Content': Message}
        trainData = trainData.append(df2, ignore_index = True)
        processed_review = []
        for i in trainData["Content"].values:
            sentance = html_tag(i)
            sentance = decontracted(sentance)
            sentance = re.sub("\S*\d\S*", "", sentance)
            sentance = re.sub('[^A-Za-z]+', ' ', sentance)
            sentance = " ".join(i.lower() for i in sentance.split() if i.lower() not in stopwords)
            processed_review.append(sentance)
        trainData["Cleantext"] = processed_review
        vectorizer = TfidfVectorizer(min_df = 5,
                                     max_df = 0.8,
                                     sublinear_tf = True,
                                     use_idf = True)
        train_vectors = vectorizer.fit_transform(trainData['Cleantext'])
        X_test1=train_vectors[train_vectors.shape[0]-1]
        # y_score = model.predict(X_test)
        y_score = classifier_linear.predict(X_test1)
        res="This Message is :"+ y_score[0]
        return render_template("mntest.html",res=res,ms=Message)
```

```
    return render_template("mntest.html",res="",ms="")
@app.route("/rg")
def rg():
    return render_template("register.html")
@app.route("/rv")
def rv():
    return render_template("resultview.html")
@app.route("/trv")
def trv():
    return render_template("trainingview.html",ttsize=tot,trnsize=trsize,tssize=tstsize)
@app.route("/")
def user():
    return render_template("dataset.html",tables=[dfhead.to_html(classes='data')],titles=dfhead.columns.values)
@app.route("/Loadds")
def Loadds():
    return render_template("dataset.html",tables=[dfhead.to_html(classes='data')],titles=dfhead.columns.values)


if __name__=='__main__':
    from werkzeug.serving import run_simple
    run_simple('localhost',9000,app)
```

Figure 6.1: HOME PAGE

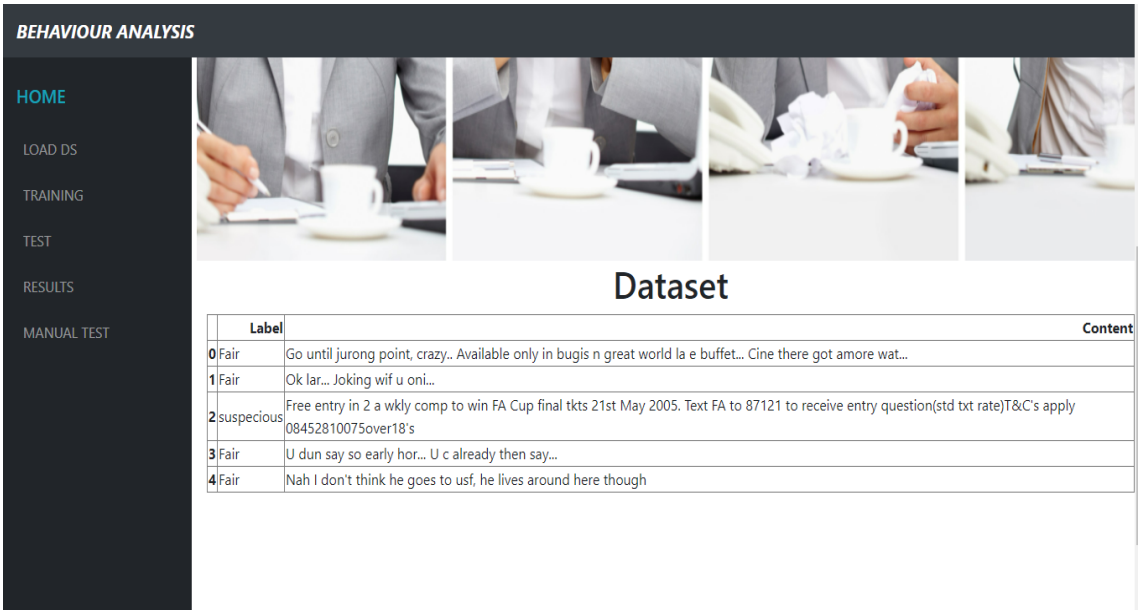


Figure 6.2: DATASET VIEW

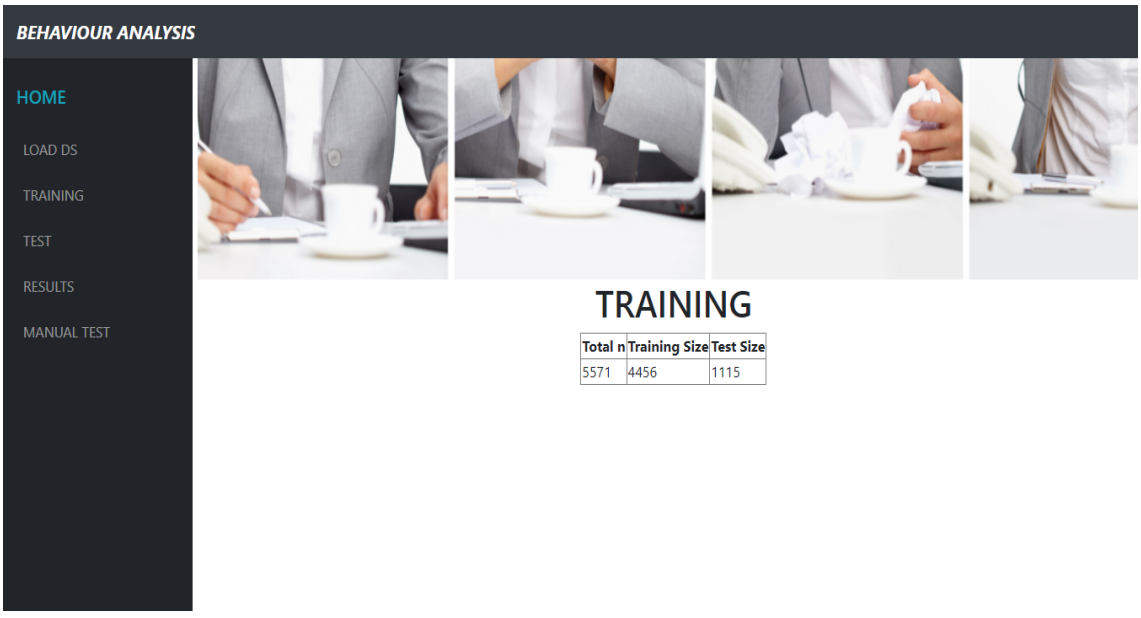


Figure 6.3: TRAINING SET

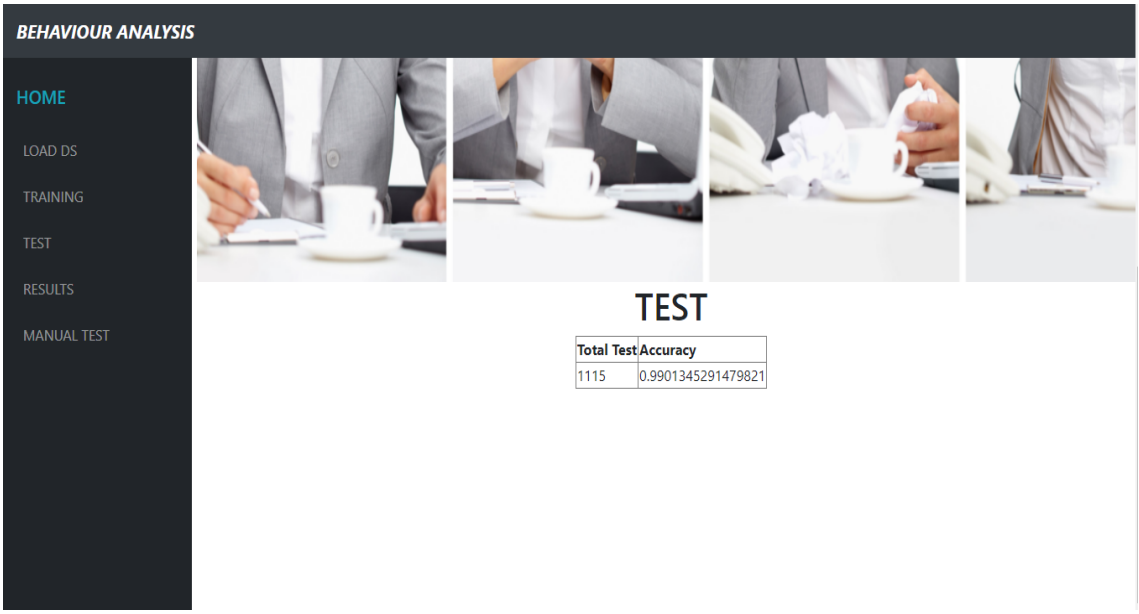


Figure 6.4: TEST SET

Figure 6.5: RESULT

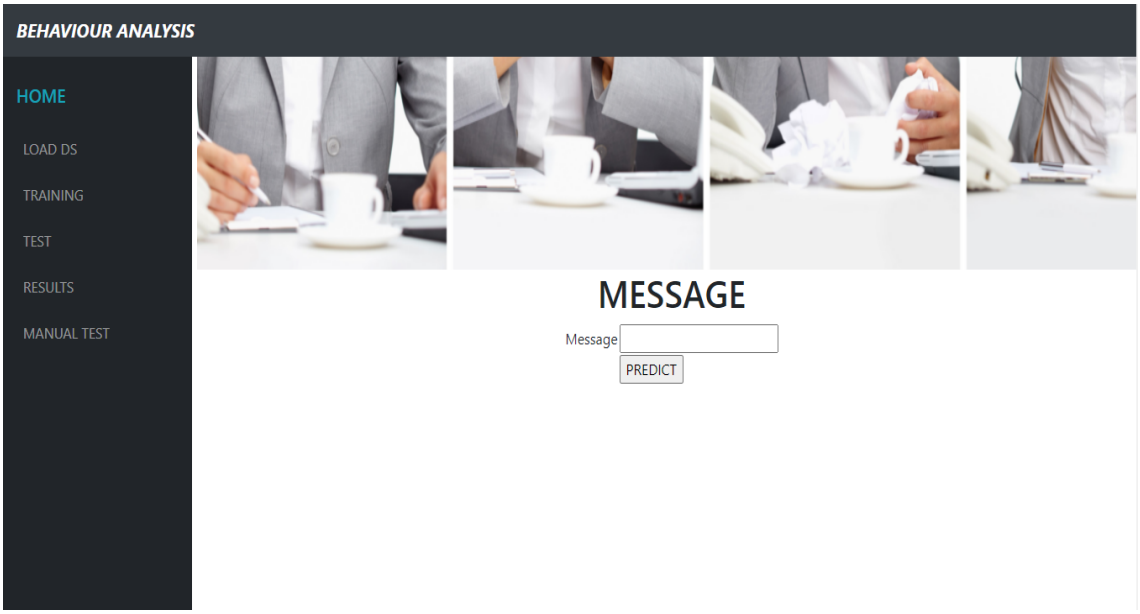


Figure 6.6: MANUAL TEST

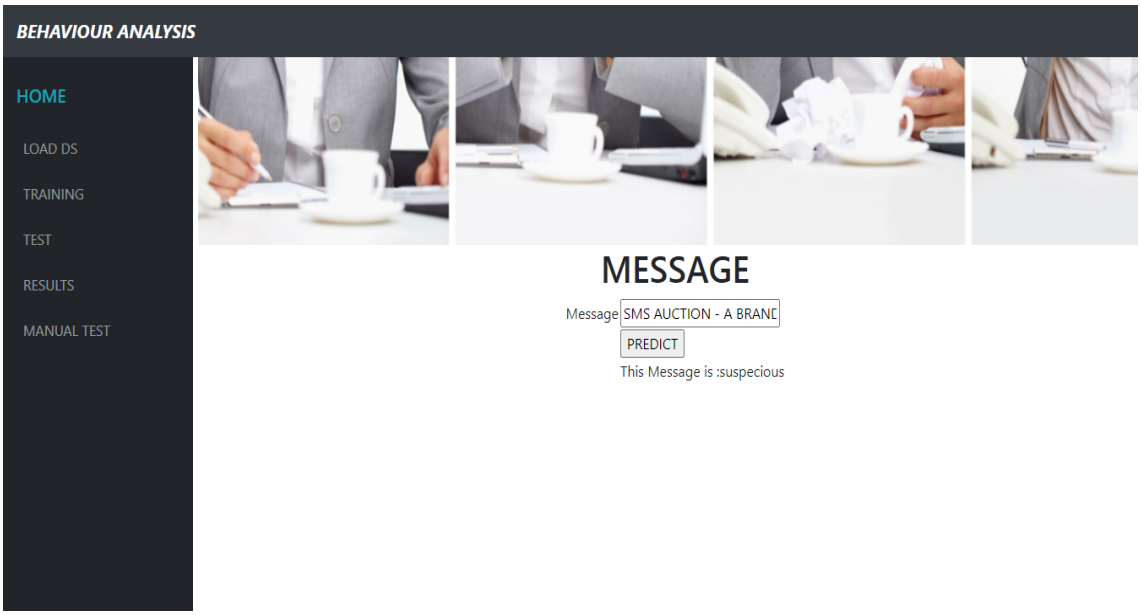


Figure 6.7: MANUAL TEST

Chapter 7

REFERENCES

- www.youtube.com
- www.wikipedia.com

1 Proceedings of the Third International Conference on Intelligent Communication Technologies and Virtual Mobile Networks (ICICV 2021). IEEE Xplore Part Number: CFP21ONG-ART; 978-0-7381-1183-4

[2] “Sentiment analysis of Twitter data during critical events through Bayesian networks classifiers”,Gonzalo A.RuzabPablo A.Henríqueza AldoMascareño-8 January 2020