ÇANKAYA UNIVERSITY
FACULTY OF ENGINEERING
COMPUTER ENGINEERING DEPARTMENT

CENG 408
Sentiment Analysis of the Feedback from Airplane Passengers

Test Plan

İrem Şahingöz - 201611051
Merve Şirin - 201626404

2020-2021

Contents

1. INTRODUCTION

This project enables the analysis of the comments made by the passengers on Twitter as a result of the services offered to the passengers of American airports with artificial intelligence. As stated in the project report, various methods and algorithms selected as a result of the researches were used. Additions will be made to the project when more efficient methods are found in our project process. Details are given in other sections.

1.1 Version Control

Version No	Description of Changes	Date
1.0	First Version	April 16, 2020

1.2 Overview

The accuracy of our prediction will be tested by dividing the comments made by the airport passengers into words and classifying the emotions they contain as positive, negative, or neutral.

1.3 Scope

This document contains the test plan of the use cases, test cases, and test design features. The scope of the project and the targeted result expectations are explained in detail in the project document, SRS (Software Requirements Specification) and SDD (Software Design Description). During the test phase, all cases specified in the document will be tested.

1.4 Terminology

Acronym	Definition
DC	Data Collection
DP	Data Preparation
TP	Text Preprocessing
SA	Sentiments Annotation

2. FEATURES TO BE TESTED

The performance and success of our model are tested by calculating the accuracy of our data set.

2.1 Data Collection (DC)

Our data set was provided by our advisor.

2.2 Data Preparation (DP)

Filtering redundancies in data.

2.3 Text Preprocessing (TP)

Cleaning and normalizing, tokenizing, stopwords removal, stemming.

2.4 Sentiment Annotaton (SA)

This statement will be testing the polarity determination of the comments.

3. FEATURES NOT TO BE TESTED

In the dataset specified in the project, no test procedure was applied to the airline sentiment section.

4. ITEM PASS/FAIL CRITERIA

4.1 Exit Criteria

%100 of the test cases are executed

%99.9 of the test cases passed

All High Priority test cases passed

5. REFERENCES

[1] SRS

[2] SDD

6. Test Design Specifications

6.1 Data Collections

6.2 Data Preparation

6.2.1 Filtering (DP.F)

With this subfeature, filtering redundancies in data.

6.3 Text Preprocessing

6.3.1 Cleaning (TP.C)

With this subfeature, cleaning the text data.

6.3.2 Normalizing (TP.N)

With this subfeature, arrangements have been made in sentence structures.

6.3.3 Tokenizing (TP.T)

With this subfeature, tokenize the text data.

6.3.4 Stopwords Removal (TP.SR)

With this subfeature, remove the stopwords.

6.3.5 Stemming (TP.S)

With this subfeature, find the root of the word with stemming.

6.4 Sentiment Annotation

6.4.1 Positive (SA.PO)

With this subfeature, positive polarity determination of the comments.

6.4.2 Negative (SA.NEG)

With this subfeature, negative polarity determination of the comments.

6.4.3 Neutral (SA.NEU)

With this subfeature, neutral polarity determination of the comments.

6.5 Test Cases

TC ID	Give corresponding requirement no	High or Medium or Low	A brief description
DP.F	6.2	H	Redundancies in the data are detected and filtered.
TP.C	6.3	H	Expressions that do not express emotion in the text data are cleared.
TP.N	6.3	H	Normalized by making adjustments in sentence structures.
TP.T	6.3	H	Text data is tokenized.
TP.SR	6.3	H	Stopwords are removed from the text data.
TP.S	6.3	H	Finding the root of the word.
SA.PO	6.4	H	Comments with positive polarity are detected.
SA.NEG	6.4	H	Comments with negative polarity are detected.
SA.NEU	6.4	H	Comments with neutral polarity are detected.

7. Detailed Test Cases

7.1 DP.F

TC_ID	DP.F
Purpose	Filter the data
Requirements	6.2
Priority	High.
Estimated Time Needed	15 Second
Dependency	The system is executed
Setup	-
Procedure	Filter the redundancies in data
Cleanup	-

7.2 TP.C

TC_ID	TP.C
Purpose	Cleaning
Requirements	6.3
Priority	High.
Estimated Time Needed	15 Second
Dependency	The system is executed
Setup	-
Procedure	Clean the text data
Cleanup	-

7.3 TP.N

TC_ID	TP.N
Purpose	Normalizing
Requirements	6.3
Priority	High.
Estimated Time Needed	15 Second
Dependency	The system is executed
Setup	-
Procedure	Normalized the text data
Cleanup	-

7.4 TP.T

TC_ID	TP.T
Purpose	Tokenizing
Requirements	6.3
Priority	High.
Estimated Time Needed	15 Second
Dependency	The system is executed
Setup	-
Procedure	Tokenize the txt data
Cleanup	-

7.5 TP.SR

TC_ID	TP.SR
Purpose	Remove stopwords
Requirements	6.3
Priority	High.
Estimated Time Needed	15 Second
Dependency	The system is executed
Setup	-
Procedure	Remove the stopwords in the text data
Cleanup	-

7.6 TP.S

TC_ID	TP.S
Purpose	Stemming
Requirements	6.3
Priority	High.
Estimated Time Needed	15 Second
Dependency	The system is executed
Setup	-
Procedure	Find root of the word in the text data
Cleanup	-

7.7 SA.PO

TC_ID	SA.PO
Purpose	Detect the positive polarity
Requirements	6.4
Priority	High.
Estimated Time Needed	2 Minutes
Dependency	The system is executed
Setup	-
Procedure	Positive polarity comments are detected
Cleanup	-

7.8 SA.NEG

TC_ID	SA.NEG
Purpose	Detect the negative polarity
Requirements	6.4
Priority	High.
Estimated Time Needed	2 Minutes
Dependency	The system is executed
Setup	-
Procedure	Negative polarity comments are detected
Cleanup	-

7.9 SA.NEU

TC_ID	SA.NEU
Purpose	Detect the neutral polarity
Requirements	6.4
Priority	High.
Estimated Time Needed	2 Minutes
Dependency	The system is executed
Setup	-
Procedure	Neutral polarity comments are detected
Cleanup	-

8. OBJECTIVES AND TASKS

8.1 Objectives

Consistency of analyzed data with targeted text data. (The score value of the train data set and test data set fitted should close to the targeted level.)

8.2 Tasks

Correcting errors on the data and adapting other predetermined options to the algorithm.

9. TESTING STRATEGY

Correct completion of the analysis phase of words and letters. Making sure that the merge and extraction phase is working correctly. The result of the estimation should make close to the determined accuracy rate.

9.1 Model Creating

In this project, we used a supervised learning type model where we need to cluster some given spaces according to their characteristics or categories.

In this code, what we do is load a sample data set, separate training and test pieces, and then train our SVM model with our training data. Our model is a Support Vector Classifier (SVC) classifier produced with a linear kernel.

9.2 Model Testing

APIs of scikit-learn were used to test our model after training with training data. In the prediction list in our code, the estimates of our model for each test sample are included.

9.3 Performance Testing

The performance of our model was evaluated using different measurement methods: The accuracy of our data model and the success rate were measured. In this case, the performance of our model was examined by looking at the score value.

10. HARDWARE REQUIREMENTS

There may be changing in the performance of the computer depending on the hardware specifications. It can be run on all operating systems.

11. RESOURCES/ROLES & RESPONSIBILITIES

Throughout the project, the results are evaluated by using different methods with my group friend, and studies are carried out to increase the efficiency of the project.

12. RISKS

When errors that may occur during data training are not noticed, the results become inefficient. This, in turn, will lead to a decrease in the success rate of the project.

13. TOOLS

Errors that occur can be corrected in python software development environments. Python code development environments such as Colab and Jupyter Notebook are used.