SOFTWARE TEST DOCUMENT

SÜZGEÇ
TURKISH TEXT SUMMARIZATION

SUPERVISORS

Ayşenur BİRTURK
Itır ÖNAL

Team Default

Abdullah Göktuğ MERT
Baran Barış KIVILCIM
Enes Uğur ŞEKERCİ
Yağız ARKAYIN

Table Of Contents

1. INTRODUCTION

This report includes the test report and results of the software (Turkish Text Summarization Chrome Extension and Web Application) that Team Default had been developed.The overall testing phase determine that the software can handle the requirements that determined in SRS.

This document is prepared by IEEE Std 829-2008 standart.

2. SCOPE

The software product contains two service both of which developed with server-client architecture. The main components are the browser extension, the summarization algorithm and the web application. The tests of each of these components are independent. But, the final services are combination of two components. The unit testing were done in each sprint and not included in this document.

3. REFERENCES

a. Suzgec Software Requirement Document
b. Suzgec Software Design Document
c. IEEE Std 829-2008

4. INTRODUCTION TO SYSTEM TEST APPROACH

This section is provided to introduce next to sections that system test design and system test cases. The first one, details of the system test design is aim to show how to define test approach. The second one, system test cases are provided the test case scenarios that Team Default tested on.

Team Default coded a test module to specify these task and iterate them easily. Especially in the part of classifier tests, these step was very important, because classier tests are taking too much time and must be iterated too many times.

In some of the test cases, Black Box testing method is used, like Sending PDF URL Test and Extension-Server Communication Test. The analysis method is used for Classifier and Clustering tests.

5. DETAILS OF THE SYSTEM TEST DESIGN

The test case identifiers are:

1. Feature Representation Tests
2. Word2Vec with Wikidump Test
3. Text to Sentence Parser Test
4. Web Page Summarization via Chrome Extension Test
5. Clustering Test
6. Summarizing contents by URL
7. Implementation of Support Vector Machine Test
8. Implementation of Logistic Regression Test

6. SYSTEM TEST CASES

This section presents test cases associated with tests designed in the previous section.

6.1 - Test Case 1

Test case identifier / associated test no	tc1/ 1
Objective	Verification of the structural feature values, which are explained in the previous retrospective documents.
Inputs	Sentences
Outcomes	Sentence feature functions returned correct values for each sentence.
Environmental needs	Any device that can run Python codes
Special procedural requirements	The text must obey the basic orthographic and punctuation rules.
Intercase dependencies	-

6.2 - Test Case 2

Test case identifier / associated test no	tc2 / 2
Objective	Parsing a text written in Turkish to Sentences correctly
Inputs	Text
Outcomes	A list of sentences of given text
Environmental needs	Any device that can run Python codes
Special procedural requirements	The text must obey the basic orthographic and punctuation rules.
Intercase dependencies	-

6.3 - Test Case 3

Test case identifier / associated test no	tc3 / 3
Objective	Verification of word existence in the word vector and getting correct values of cosine distance between two words
Inputs	Words
Outcomes	We have a sufficient number of words in word vector and their relation with each other is as it should be.
Environmental needs	A trained word vector
Special procedural requirements	-
Intercase dependencies	-

6.4 - Test Case 4

Test case identifier / associated test no	Tc4 / 4
Objective	Sending website HTML codes to server through extension and get the response from it
Inputs	Webpages
Outcomes	Extension can send the page content and show the response from server
Environmental needs	A running instance of Süzgeç Web Application Back End, Internet connection, Google Chrome Browser, Süzgeç Chrome Extension
Special procedural requirements	-
Intercase dependencies	-

6.5 - Test Case 5

Test case identifier / associated test no	Tc5 / 5
Objective	Creating clusters by K-Means algorithm and select their centroids as summaries.
Inputs	Texts
Outcomes	As our final result, we have 0.39 f-score.
Environmental needs	Any device that can run Python codes and its Scipy module.
Special procedural requirements	-
Intercase dependencies	-

6.6 - Test Case 6

Test case identifier / associated test no	Tc6 / 6
Objective	Summarize any URL's web content on the web application
Inputs	A web page URL
Outcomes	Our web application accomplished to access the given URL, get its HTML content and return the summary as an output.
Environmental needs	Any web browser, a running instance of Süzgeç Web Application Back End, Internet connection
Special procedural requirements	-
Intercase dependencies	-

6.7 - Test Case 7

Test case identifier / associated test no	Tc7 / 7
Objective	Testing the accuracy of the implementation of Support Vector Machine, a method we used to classify summary sentences.
Inputs	A dataset of texts along with their summaries.
Outcomes	The implementation tested with different configurations. However, in each case, the dataset did not fit in SVM. As a result, the prediction of the classifier was not sufficient.
Environmental needs	A dataset of texts and their summaries.
Special procedural requirements	-
Intercase dependencies	-

6.8 - Test Case 8

Test case identifier / associated test no	Tc8 / 8
Objective	Testing the accuracy of the implementation of Logistic Regression, a method we used to classify summary sentences.
Inputs	A dataset of texts along with their summaries.
Outcomes	The implementation was tested on different programming languages with different settings of the parameters. The accuracy results of the prediction of this regression method was low for our dataset.
Environmental needs	A dataset of texts and their summaries.
Special procedural requirements	-
Intercase dependencies	-

7. DETAILED TEST RESULTS

Tc1	**Passed**
Tc2	**Passed**
Tc3	**Passed**
Tc4	**Passed**
Tc5	**Passed**
Tc6	**Passed**
Tc7	**Failed**
Tc8	**Failed**