

Test Plan

The Testing Process

We will be focusing on conducting tests for the five main methods of the fuzzywuzzy project. These five methods are each different ways to compare strings.

Requirements Traceability

Fuzzywuzzy is designed as a way to compare strings. The methods must compare the accuracy of strings and how closely two strings are to each other in different fashions, from comparing if they are exactly the same or are similar. What is testable is this number that is produced from each method and if the different methods compare the strings as they are designed.

Testable Items

Simple Ratio

Checks of the strings are exactly alike

Partial Ratio

Checks that the strings are mostly alike

Token Sort Ratio

Checks that the same words are presents in both strings

Token Set Ratio

Checks that most of the same words are present in both strings

Process

Includes the ability to pull a compared string from an array and show which strings are the closes match

Testing Schedule

25 total tests will be designed, 5 for each method presented. The total number of tests will be divided approximately between the three group members to come up with. Over the rest of the semester, the schedule will be divided out as such:

9-23: Test plan and 5 tests created

10-14: Automated testing framework used to implement test plan and at least 5/25 test cases.

11-6: Completed testing framework and all 25 test cases created that will be used automatically by the framework.

11-18: Design and inject 5 faults that cause 5 tests to fail.

Test Recording Procedures

The test results will be sent to a file, either a single file for all test cases with indicators of which test are being performed or into separate files for each separate test item. These will be formatted on individual rows for easy legibility.

Hardware and Software Requirements

Current testing environments include:

Ubuntu 14.04,
Python 2.7 or better

Git
Difflib
Optionally python-Levenshtein

Constraints

One of the largest constraints is the group members' schedules along with other work that must be completed. Most of the issues with come from trying to find time in order to meet. Frequent communication is encouraged through Google Hangouts, Google Docs, textings, and face-to-face meetings. Through as much frequent communication as possible it is hoped to avoid too much miscommunication and last minute problems, even if it becomes difficult to meet up as a group due to schedules and other work.

System Tests

Since we are testing strings, we thought it appropriate for a set of tests to be done with empty strings.

Ratio

Test:1
function: fuzz.ratio
input: "", "testing empty string"
output: 0

Partial Ratio

Test:1
function: fuzz.partial_ratio
input: "", "testing empty string"
output: 0

Token Sort Ratio

Test:1
function: fuzz.token_sort_ratio
input: "", "testing empty string"
output: 0

Token Set Ratio

Test:1
function: fuzz.token_set_ratio
input: "", "testing empty string"
output: 0

Test:2
function: fuzz.token_set_ratio
input: "Fun Stuff", ""
output: 0