|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Document Version** | **Prepared By** | **Compiled On** | **Reviewed By** | **Date Of Approval** |
| 1.0 |  | 19/6/2018 |  |  |

|  |
| --- |
| Hi,  Thanks for allowing us to forward you a proposal and your help.  Please take a moment to review the complete proposal as well as a few questions that we would like to know. Your timely answers are greatly appreciated as in the development process it is often the unknown, which can lead to issues and ultimately dissatisfaction during or after completion of project.  As always, please feel free to contact us anytime via email or phone with any questions or concerns you may have.  Thanks again for your time and choosing Reubro!  With Kind Regards,  Philip Abraham  **CEO**  Reubro International  **+91 484 2 357 256**  [www.reubro.com](http://www.reubro.com/%2200d0c9ea79f9bace118c8200aa004ba90b0200000003000000e0c9ea79f9bace118c8200aa004ba90b2e00000068007400740070003a002f002f007700770077002e00720065007500620072006f002e0063006f006d002f00000) |

|  |
| --- |
| **PROBLEM STATEMENT** |

The need to find phrases in different datasets that mean/convey the same thing but are not identical to each other 100% word-to-word.

|  |
| --- |
| **CURRENT STATE** |

At present, VLOOKUP on excel is being used but is incapable in finding very close phrases that aren’t 100% identical. Therefore, other methods like manual searching is employed.

|  |
| --- |
| **DESIRED STATE** |

A software that matches test phrases that aren’t 100% identical word-to-word but mean/convey the same thing. This software should determine the closeness of the phrases and find matches on the basis of this closeness.

|  |
| --- |
| **OUR PERSPECTIVE** |

Reubro views this project as a “standard” application build, which shall be an original layout not a reproduction or purchased template.

|  |
| --- |
| **ARCHITECTURE** |

The Architecture of the project relates to the type of frameworks and/or methodologies used for the purpose of development and the release of the final project.

What we plan to do is use a slightly modified form of Jaccard Index to find the similarity of products given in the two datasets. The Jaccard index measures similarity between finite sample sets, and is defined as the size of the intersection divided by the size of the union of the sample sets.

For example,

Assume we have two products, “Indoor and outdoor foldable table for comfort and easy use” and “Home furniture indoor study table, foldable and easy to use”.

We first ignore the commonly used adjectives like ‘the’, ‘and’, ‘to’, etc. Then we use Jaccard index by calculating the number of words which appear in both divide by total number of words combined.

Therefore, the above given sentences have a coefficient of 0.5. Depending on the results of the first implementation, this should give a percent match as around 70-80%.

**Advantages**

* Because the number of items in the dataset are huge, we cannot use more complex algorithm as it can lead to nasty run-time constraints. This method should work for such big datasets.
* This gives good results especially for this case where we compare similar products from two datasets.

**Disadvantages**

* For example, let us take two products to be “Blue headphone with mic” and “Over-the-ear headphone without mic”. According to this algorithm, these will have high Jaccard index even though ‘with mic’ and ‘without mic’ are the complete opposite.

We can check for such examples and try to include a catch mechanism for the same in subsequent iterations.

* This only works as long as similar products have similar words in the product description. If the two items uses synonyms or definitions of the same word, this algorithm will not be able to find those.

We can also try to include some dictionary type of function to find synonyms but due to

run-time constraint not sure if this is viable.

|  |
| --- |
| **DESIGN** |

The following elements are standard items in which Reubro will employ during the development this application:

* The application will run on PC (desktop and laptop) on most operating system.

|  |
| --- |
| **FUNCTIONAL SPECIFICATIONS** |

The following elements are standard items in which Reubro will employ during the development this application:

**1. Main Page**

Here, the user will add the two datasets with phrases that will be compared for similarity. Currently, the documents should be in .xlsx format. User can give the compare options in this page.

**2. Results Page**

Here, the results of comparison are shown to the user. Each phrase from the first dataset is compared to every other one in the second dataset based on closeness of the statements. Thus, statements meaning the same will have higher similarity than others (thus higher percentage similarity is printed).

|  |
| --- |
| **TECHNOLOGY USED** |

* Python 3.0 using packages like Tkinter, Numpy etc.

|  |
| --- |
| **REQUIREMENTS FROM CLIENT** |

* Test dataset and required results.

|  |
| --- |
| **COMPLETION CRITERIA** |

User acceptance testing will consist of verifying the completion of the criteria outlined in this document. Alterations to app functionality that are required after the completion of user acceptance testing will have additional incremental costs associated with them. This does not include bug fixes to allow proper site functionality (i.e., items that cause error messages to return are bugs. However, all other alterations to site functionality, regardless of their appearance within this initial scope document, will result in billable charges after the completion of user acceptance testing.

*This proposal remains the property of Reubro International. Any imagery such as but not limited to mock designs, pragmatic topographies, sketches, layouts, otherwise defined as “work product” shall not be distributed, used, reproduced, or modified without expressed written permission from Reubro.*

|  |
| --- |
| **TIME ESTIMATES** |

|  |  |  |
| --- | --- | --- |
| **Sl. No.** | **Module** | **Time (Days)** |
| 1 | Documentation | 5 |
| 4 | Development | 20 |
| 6 | Testing | 5 |
|  | **Total** | **30 Days** |