807097_WSOA3000_ExamProjectDocumentation

Date: 4 June 2017

Rationale

The aim of my project is to extend the notions of glitch art, as seen in the works, "My Boyfriend Came Back From The War" (1996), "Shredder" (1998) and "Net.Flag" (2002). At its core, my project is a response to Mark Napier's "Shredder". Just as Napier has designed an application which seeks to apply the concepts of glitch art to a web site, my application aims at glitching text. By rewriting and reordering text, based on definitions and properties of each word, a user is offered a glimpse into what a machine might understand or interpret from a given body of text. This process often leads to the sort of glitched, incomprehensible data which may be seen in other works of the glitch art movement.

How it works

Using a combination of APIs, each word in a body of text (provided by user) is analyzed. For every word, the application makes a call to an API which returns information about each word. In this returned information, a synonym for each word is included. Here, it may be more than one synonym that is returned per word. My application chooses a synonym at random and replaces the reference word with an appropriate synonym in the original text. Once the application has completed its conversion of the text, four words are chosen at random from the original text. These words are then used to generate images and links to other sites which focus on each word individually.

In the end, the application will return a version of the original text. This text may hold the same meaning as the original to another machine, but may make no sense to another human being. Further, since there may be more than one synonym per word, a user can provide the same text and receive a different output every time. Also, however unlikely, a user may also provide previously converted text and receive the text that was originally converted.

Usage

1. Enter text into input section on left of page.
2. Click on button at bottom of input section to convert all provided text.
3. Converted text is displayed in an output section to the right of the page and includes iframes and links to other sites.

Bugs and Problems

Due to limited API access, the number of words that may be converted in a second and in a day is restricted. Thus, in code, two API keys have been provided (Should one key have already expired, please use other key). (note: "key0" and "key1" in code). This also means that the length of the body of text is limited to the amount of calls which can be made.

As almost every word may be written in a different form, the API cannot perfectly convert all the given text. Words which cannot be processed by the API are marked in bold in the converted text.

API calls are often slow and output is not immediate. However, the application is designed to show expose the process of converting the text by updating text in real time.

APIs Used

- Word analyser: https://words.bighugelabs.com/
- Images: http://developers.gettyimages.com/en/