University of Malta
ARI1102 Documentation
Kian Parnis
0107601L

Specifications:

The following are the tasks that are completed:

- Ubuntu Setup
- Installation and Configuration
- Web interface {*all parts*}
- Python server scripting {*all parts*}
- Notifying the user if text is left empty.

Components and Materials:

- Lecture notes / recording to set up the server. [1]
- Bootstrap template which was used for the base of the site. [2]

References

[1] Dr. Joel Azzopardi lecture slides 2020

[2] https://startbootstrap.com/previews/heroic-features

Core Components explanation:

The user is presented a front-end site where they can enter text and post it through JSON. Upon entering text, the python script is called by the server which first detects if the text is English with the use of the package '*langdetect*', if the text is not English *"1"* is appended to the list and when the text is parsed through the JavaScript the user gets an alert that the text is not English. If the supplied text is English the supplied python script moves on to tokenize the paragraph as sentence's with the use of *nltk.tokenize.sent_tokenize(),* after a for loop goes through the sentences and tokenizes each word which are stored in '*tokenized_text'*, followed by a *universal pos_tag* were this is appended to the list variable *'POS1'.*

The stored words are now also processed using *Named Entity Recognition* with the use of *nltk.pos_tag()* and *nltk.chunk.tree2conlltags()* and this is appended as well in

POS1. This is now processed through JavaScript and since POS1 and POS2 were both stored in the same variable one sentence at a time, a switch (*labelled "swtch"*) was used during the for loop which cycles through both to be presented to the user.

Since the universal tagset has around nine possible configurations, the outputs are taken and displayed via colour-code to be more aesthetically pleasing to the viewer. On the other hand, for the *Named Entity Recognition* (POS2), a table of contents is present to the user for all the possible outcomes that can be given. POS1 and POS2 are now respectively placed inside the inner html in their respective *divs*. Lastly if the user were to leave the text box empty JavaScript would detect this and alert the user to "*Enter some text*". The website was also tested for responsiveness using chrome, internet explorer etc and on handheld screens such as iPhone x, iPad pro, Pixel 2 XL etc were the responsiveness held to being aesthetically pleasing.