Nicholas DaRosa
ndarosa2
CS410 Text Information Systems
23 October 2022

Project Proposal: Natural Language Processing Extension for Firefox

This proposed project will be an individual project completed by me, Nicholas DaRosa (NetID: ndarosa2), so I will also be the team captain. The primary goal of the project is to create a browser extension in Mozilla Firefox that performs natural language processing techniques (e.g. sentiment classification) on text that has been highlighted or inputted by the user. After the processing has been completed, the results of the natural language processing techniques will be presented to the user for inspection. The planned techniques that will be performed on the text are lemmatization, lexical analysis (specifically part-of-speech tagging), semantic analysis (specifically entity recognition), and sentiment classification.

The project's main theme is intelligent browsing. This project relates to the theme of intelligent browsing because it revolves around adding functionality to existing browsers, specifically Mozilla Firefox although other browsers will be considered. Furthermore, this project relates to the class because it implements the techniques for several topics discussed in the class such as lemmatization, part-of-speech tagging, entity recognition, and binary sentiment classification (i.e. positive or negative).

Although a myriad of topics are discussed in this class, one problem is that many of the discussed topics are not reinforced through hands-on demonstration and exploration. This project aims to contribute a partial solution to this problem by providing students hands-on experience in the topics of lemmatization, part-of-speech tagging, entity recognition, and sentiment classification. Moreover, the demonstration of part-of-speech tagging and sentiment classification on inputted text will also reinforce to students the state of natural language processing in terms of how accurate/inaccurate modern natural language processing techniques are currently.

In this project, the user will provide their own datasets since they will be highlighting or copying text that will then be used as the input into the program. As stated previously, the techniques that will be employed are lemmatization, part-of-speech tagging, entity recognition, and sentiment classification. The main programming language used to implement the project will be Javascript with specific use of the Javascript libraries Compendium [1] and Compromise [2].

Upon completion, a brief video will be recorded of a user using the extension to demonstrate that the implementation is working as expected. The user will highlight text (e.g.

“Joe really likes dogs”) which will then be processed by the extension to display results to the user such as the text’s sentiment classification (e.g. positive with a score of 0.89) and part-of-speech tags (e.g. “dogs” is a plural noun). A report will also be produced comparing the text’s expected results (as determined by a human) in terms of recognized entities, part-of-speech tags, lemmatization, and sentiment classification versus the results provided by the extension. This report will demonstrate how accurate the techniques employed in the extension actually are and if they are working as expected.

Since this project is being completed individually, the workload of the project should take at least 20 hours. I do not have much experience in Javascript and have no experience in creating browser extensions, so the time required to successfully complete the project may take longer than expected compared to those with more experience. It is planned that a basic webpage version of the project will be created first to demonstrate it is feasible and then related scripts will be converted to be used as part of a browser extension.

In order to successfully complete the project, several tasks need to be completed. The first task is to create a basic webpage in which a user can enter in text, hit submit, and then the lemmatization results of the text will be displayed. If possible depending on the required processing time, the processing scripts will rerun whenever there is a change in the inputted text. It is estimated this step will take eight hours as I have no experience with the previously mentioned Javascript libraries. The next step is then to also implement the part-of-speech tagging, sentiment classification, and entity recognition for the inputted text, which will hopefully take around six hours. Once a bare bones version of the project is working, the scripts will then be ported to be used as part of a Firefox browser extension. As I have never created a browser extension before, it is estimated a basic working version of the extension will take eight hours. To implement automatic processing of text through highlighting and to customize the display of the extension, it is estimated four hours is required. Lastly, it is estimated quality assurance testing and subsequent fixes will require four hours. Assuming the estimated task times are approximately accurate, the programming portion of the project should take approximately 30 hours which is greater than the at least 20 hours requirement.

References

1. https://github.com/Ulflander/compendium-js
2. https://github.com/spencermountain/compromise/