BMI 500: A Report on Natural Language Processing Part 2

Alireza Rafiei[1]

[1] Department of Biomedical Informatics, Emory University, Atlanta, Georgia
Email: alireza.rafiei@emory.edu

Problem 1:

First, using the CountVectorizer sklearn tool, the text1 and text2 documents were converted to a matrix of token counts to create a vectorizing method with an n-gram size of 1-3 for the vectorize. Second, the Cosine similarity and the Jaccard similarity between text1 and text2 were calculated. While the similarity score for the former was 0.39745, the latter just showed a 0.02374 score. This might stem from the fact that Jaccard similarity takes only a unique set of words in the files; however, Cosine similarity takes a total length of the vectors in the files.

Problem 2:

To answer this problem, I compared the performance of a unigram tagger, a bigram tagger, and a combined tagger using the brown corpus dataset provided by the nltk package. Of the three tested taggers, the combined tagger had the best performance with 0.8452 accuracy. The developed unigram tagger took the second best place in terms of performance with an accuracy of 0.8121. However, the bigram tagger reached just accuracy of 0.1020. This sharp difference is likely due to the effect of the increment in the combination of consecutive words, which leads to a decrement in the likelihood of encountering those specific n-words. Based on the results, a combined tagger was utilized for training a POS tagging to tag all the words from text1.

Lowercasing can affect the performance of the tagger. Back to Lecture 5, lowercasing is not considered for tasks such as comparison of different contents. Nevertheless, in the current task, it would result in the elimination of clues that can be provided. Take names that are usually in uppercase as the most patently example. Therefore, lowercasing can affect the performance of the POS tagger.

Problem 3:

Generally, a question answering system (QAS) has three main components: question classification, information retrieval, and answer extraction. Question classification as a query processing module plays a primary role in a QAS to categorize the question based on context using conducting preprocessing on the original input data. It also aims to extract keywords and reformulates a question into semantically equivalent multiple questions. Information retrieval focuses on extracting applicable answers from the relevant documents. Information retrieval searches for information in documents, for documents themselves, for metadata that describes

documents, or even within databases to find specific pieces of information (keywords) to provide a concise, comprehensible, and correct answer. After the precision and ranking of candidate information, the answer extraction module gives the top N relevant documents from the retrieval module. It performs a detailed analysis and pinpoints the answer to the question. Usually answer extraction module produces a candidate list of answers and ranks them according to specific scoring functions. As a result, a QAS provides an answer to the specified question.

References:

https://scikit-learn.org/stable/modules/generated/sklearn.feature_extraction.text.CountVectorizer.html

https://www.nltk.org/book/ch05.html

Gupta, P. and Gupta, V., 2012. A survey of text question answering techniques. International Journal of Computer Applications, 53(4).