COMP 10261

Sara Zendehboodi

Assignment 3 Resources Report

Oct 25th, 2024

Justin Triantafilou

000775460

To start the python program is imperfect, although it does block out same article recommendations it does not have a dissimilarity recommendation portion and may recommend articles multiple times. I chose to use TFIDVectorizer to lower the weight of common words and chose a standard tolerance on min/max\_df to save time. I use cosine similarity along with a recommendation I found on stack overflow to actually retrieve articles based on the matrix results. That can be found in the provided link on Answer 2. Additionally, I used various info from the scikit-learn documentation website. Finally, I used Co-Pilot in order to help me find the right slicing values to obtain the n most similar articles, as I was having some trouble with trial and error. Specifically on line 120 where I am slicing the resultant argsort() array. Argsort() is a sorting function that returns an array of indices based on would be sorting positions of the input array, without actually sorting it, thereby saving computation time. I retrieved argsort() from the NumPy package. I also imported panda to make my vector tokens and cosine similarity matrices more readable in the console.

Sources:

1. <https://stackoverflow.com/questions/12118720/python-tf-idf-cosine-to-find-document-similarity/18914884#18914884>
2. https://scikit-learn.org/stable/modules/feature\_extraction.html#common-vectorizer-usage