Step-By-Step Instructions

BMI6115 Biomedical Text Processing Final Project Dave Sant

This is simple to follow, but the document selection must be performed on a remote computer because I do not have the SQL database for the MIMIC-III documents on my own computer. Once the documents are obtained, they will need to be moved to folders with specific names that will be described later in this document. For the deployment stage, I used a downloaded CSV file containing the text of the entire MIMIC-III dataset. The python script for the deployment will need to have the pathway to the NOTEEVENTS.csv file updated or it will not run. This will also be described in this document.

1. Obtain the text documents for training and testing purposes. My file selecting included a SQL query that ordered documents by random order, so you will receive a different list of files than mine did. As long as you have some files in both the test and training folder, the program will run and you will obtain results for the files that you have collected.
   1. Log into the remote computer set up for BMI 6115.
   2. Use the python scripts in “~Document Selection Scripts/” to obtain documents.
   3. Note that I have deleted the password from these document selection scripts because I plan to post this on GitHub and don’t want to post the class’s login password online, so you are going to have to update the password for this to run.
   4. Run the first two python scripts “~Document Selection Scripts/DocumentSelection.A1c.py” and “~Document Selection Scripts/DocumentSelection.A1c.py”. I ran these with the Anaconda prompt on the remote computer.
   5. You don’t have to do this step, but I sorted my documents by size and then selected the smallest 92 of them for annotation.
   6. Run the third python script “~Document Selection Scripts/DocumentSelection.Random.py”. For this one, I deleted two abnormally large documents and kept the rest to make a total of 150 documents.
   7. Once you obtain the final documents, shuffle them and split them into testing and training documents. I performed this with the step using the command line on a Mac. The command on a Mac is   
      ls \*.txt | rl | head -75 > test\_files.txt  
      This can be performed on a linux command prompt using   
      ls \*.txt | shuf | head -75 > test\_files.txt  
      I honestly have no idea how to perform this on a computer with a Window’s operating system. Anyways, if you want the exact files that I used, you can see the list of the files in my text files “Training\_Files.txt” and “Test\_Files.txt”.
   8. Once you have the files, put the training text files in to the folder “~Text\_Files/Training\_Dataset/” and put the test text files into the folder “~Text\_Files/Testing\_Dataset/”.
   9. The final step is obtaining the NOTEEVENTS.csv file from the MIMIC-III dataset. I can’t share this file on GitHub or on Canvas because it still potentially has personal health information that could be used to identify a person, but I obtained this file from here: <https://physionet.org/works/MIMICIIIClinicalDatabase/>. Of note, you will have to update the python script NLP\_Deployment\_A1c\_Values\_and\_Diabetes\_Mentions.py to point to your NOTEEVENTS.csv file or it will throw an error.
2. Install PyContext. I did this on my mac by pip install PyContextNLP but there are more ways to install it shown on the GitHub page: <https://github.com/chapmanbe/pyConTextNLP>
3. The next step is to make sure you have the yaml files that are needed to specify what to mark as a target and what to mark as a modifier. These will be present on the GitHub repository so you should not have to make any of them, you just have to make sure they are present. There should be 4 files that end in .yml in the foler “Yaml\_Files”.
4. Make sure you have the file itemData.py in your working directory. This is not the one from the PyConText download, so you are going to need the one posted with these instructions and the other files on this GitHub repository.
5. Now that you have the files, you can run the scripts. There are a few things to double check:
   1. Remember, your testing files need to be in the folder “~Text\_Files/Testing\_Dataset/” and your training files need to be in the folder “~Text\_Files/Training\_Dataset/”.
   2. The results of the processes will be saved in CSV format in the folder “Output\_Files”, which will be present in the GitHub repository so it should not need to be created.
   3. You will have to modify the deployment script “NLP\_Deployment\_A1c\_Values\_and\_Diabetes\_Mentions.py” to point to your location NOTEEVENTS.csv in order for it to run.
   4. Make sure the folder “Yaml\_Files” is in your working directory.
   5. I will mention that my scripts that look for diabetes mention there is a warning given in one of the steps. I might recommend running the script with python -W ignore to suppress the warnings.
   6. Although it is not important to run them in the same order as I did, I ran them in the following order:
      1. NLP\_Extract\_A1c\_Values\_Training.py
      2. NLP\_Extract\_A1c\_Values\_Test.py
      3. NLP\_Diabetes\_Mention\_Training.py
      4. NLP\_Diabetes\_Mention\_Test.py
      5. NLP\_Deployment\_A1c\_Values\_and\_Diabetes\_Mentions.py