READ ME

You should have a “clinical\_case\_reports” file that contains the following files:

* ExtractionScript.py (This is the main script you will be using)
* impact\_dict.txt (This is necessary for ExtractionScript.py, and was created by “prep\_journals.py.” If you do not have this, once you activate the virtual environment you can run “prep\_journals.py” to create this.)
* ji\_list.txt[[1]](#footnote-1) (This is necessary for prep\_journals.py)
* journal\_dict.txt (This is necessary for ExtractionScript.py, and was created by “prep\_journals.py.” If you do not have this, once you activate the virtual environment you can run “prep\_journals.py” to create this.) (Currently not correct.)
* nlmcatalog\_result.xml[[2]](#footnote-2) (This is necessary for prep\_journals.py)
* prep\_journals.py (This was used to create “impact\_dict.txt”, and “journal\_dict.txt,” which ExtractionScript.py reads in as dictionaries. “impact\_dict.txt” maps journal names to journal impact factors, and is taken directly from “ji\_list.txt,” while “journal\_dict.txt” maps journal name synonyms to the version of the journal name used in “impact\_dict.txt.”) (Currently there is an error that is producing an incorrect version of journal\_dict.txt. I have not determined what the error is yet)
* test\_impact.py (This was used to test prep\_journals.py)

And the following directories:

* my\_env (Virtual environment created using \_\_\_\_[[3]](#footnote-3) )
* TEST (Contains a test dataset of Excel files)
* 1000 (Contains a dataset of almost 1000 metadata Excel files. Contains a few duplicates and problem files.)

You will also need to have another directory that contains the Excel sheets you want to process.

To activate the virtual environment using the command line:

* Change directories until you are inside “clinical\_case\_reports”
* Run the following line:
  + source my\_env/bin/activate

To run prep\_journals.py:

* Make sure you are still inside “clinical\_case\_reports”
* Run the following line:
  + python prep\_journals.py

To run ExtractionScript.py:

* Make sure you are still inside “clinical\_case\_reports” and that this directory also contains your directory containing desired Excel files, or that you have a file path that leads to wherever those files are stored. For this example, we will run the script on the files in the “TEST” directory.
* Decide on a base name for your output files (In this case we will use “testing.txt”)
* Run the following line:
  + python ExtractionScript.py TEST/ testing.txt
  + (Make sure you include the extra “/” at the end of the file path where the Excel files are stored)
* This will create the following output files:
  + “testing.txt” (Main output summary table. Contains the most important pieces of information from the dataset. It also parses some of the data (ie demographics) into a more analyzable form[[4]](#footnote-4).) (NOTE: We may want to edit which pieces of information is contained here slightly)
  + “AGES\_testing.txt”
  + “SCORE\_testing.txt” (Score table. Contains all the scores from the dataset)
  + “ALL\_testing.txt” (I have not created this yet, but I intend to make this to gather all the information from the dataset into one file, because currently the main output table is a summary. For example, it only includes one version of each piece of information even if it was present in the “Contained in Context” and “Indexed by MeSH” columns. The one exception is that the demographic information may in some cases combines data from both columns.<fix—it should grab genders from contained in context and only from mesh column if can’t>)

1. SOURCE THIS [↑](#footnote-ref-1)
2. AND THIS [↑](#footnote-ref-2)
3. SOURCE [↑](#footnote-ref-3)
4. SEE ADDITIONAL PARAGRAPH [↑](#footnote-ref-4)