For dependencies check ICE\_Project/README.md.

The medrec algorithm manual is in rxnormparse/medrecmanual.md

Must create the rxnorm ontology for medrec as per its manual. Rxnorm pickle is in rxnormparse/

python generate\_rxnorm\_file.py /path/to/umls/metathesaurus rxnorm.pickle.bz2

This pipeline takes a user defined immunization list (userInput.txt) and reconciles it with the output of two other immunizations lists. The other immunization lists would be provided by the output of Matt’s portion, for testing it defaults to testvMRout1.txt and testvMRout2.txt. userInput.txt can be freely modified and tested just tab delimt the imm name and the date (month-day-year) Fuzzy dates of (month-year) or just year will work as well, discussed later.

The wrapper script wrapper.sh is the main wrapper and will run the entire pipeline to view the pipeline you can open it.

First, the preprocessing to get the data into medrec suitable format is performed, seacrh.py does this for the user defined list. The user list is parsed and the immunization names are used to query rxnorm api for the proper rxnorm names, since the api search function is average performing sometimes the search is not perfect so in this case the raw string is fed to medrec instead. This search function is interactive and runs on the command line allowing the user to choose the best search result from the API query that matches what they typed in the txt file, minor constarints are added such as out of range and yes/no safegaurds.

Second, preprocess.py is ran which queries the rxnorm API using the code and codetype from the “VMR” to get the proper names for the “official source” records to be fed to med rec.

Third, the first reconciliation is done on the two official lists and the output json from medrec is put into rxnormparse/out1/ this json is parsed and our fuzzy date logic is performed. Atleast a year is required for the date. Imm’s that were considered as reconciled by the algorithm are considered as the same immunization if the full date OR the year and month are identical. Fuzzy dates get defaulted to month 6 if no month and day 15 if no day are given. Fuzzy dates that get reconciled are also output into a separate fuzzy.txt file displaying the two original given dates. A final reconciled list from the two “VMR” sources is created.

Fourth, the reconciled list from the previous steps is then reconciled with the user defined list using medrec.

Finally, the same post processing and fuzzy date reconciling is performed on this list and a final reconciliation is output to /rxnormparse/FINAL\_RECONCILE.txt. All fuzzy dates from both steps will be in FINAL\_FUZZY\_DATES.txt.

Delete the json files in rxnormparse/out1/ and rxnormparse/FINAL\_OUTPUT/ before running the pipeline each time, these are intermediate files that are referred to by name.

The medrec algorithm we leveraged operates in four steps for each individual medication; 1st a string match is attempted, 2nd the medication is looked up in the UMLS rxnorm ontology and if the ingredients are >30% identical it is reconciled, 3rd generics are considered, and lastly intent to treat based on a pre-built literature review is attempted. The algorithm has limitations and is not 100% accurate but it is published and was tested at University of Texas Health Science Center at Houston with an accuracy of 23.4%. However, they tested using strictly raw English language feeding directly into medrec.

We attempted to improve this accuracy for our case by taking the raw language and first standardizing it via the rxnorm API search queries into the same language as the algorithm uses (rxnorm ontology names).