**Expert curated disease-drug associations (DDA)**

**Resources:**

* Comparative Toxicogenomics Database (CTD)
* National Drug File – Reference Terminology (NDF-RT)
* DrugBank
* ClinicalKey

**Retrieving DDA from CTD and NDF-RT:**

See our GitHub project <https://github.com/CutaneousBioinf/LiteratureMiningTool/tree/master/RelatGold/Disease2DrugAssociationGoldStandard> for generating DDA from CTD and NDF-RT.

For any disease of interest, we retrieved the expert curated DDAs using Linux command.

$ grep 'UMLS\_CUI' DDA\_FROM\_CTD\_NDFRT | cut -f1 > OUTPUT\_FILE1

**Retrieving DDA from DrugBank:**

DDA information in DrugBank is in indication section. DrugBank indications are in unstructured format. For any disease of interest, we retrieved the indications using Linux command. We manually read the indications and collected a list of drugs suggested by the experts. We didn’t include the drugs under investigation to our DDA.

$ grep -i 'disease\_of\_interest' DRUG\_BANK\_FILE > OUTPUT\_FILE2

**Retrieving DDA from ClinicalKey:**

The resource ClinicalKey can only be access online (<https://www.clinicalkey.com/#!/>). For any disease of interest, we queried ClinicalKey web interface to retrieve the expert curated DDAs. We saved the DDAs in a file (OUTPUT\_FILE3) for further processing.

**To combine DDAs from all the resources and to get a unique list:**

We developed a Java program to combine the drugs from all resources. The program also gets the ID and synonyms from our drugs lexicon.

$ javac KnownDrugsCompiler.java

$ java KnownDrugsCompiler OUTPUT\_FILE1 OUTPUT\_FILE2 OUTPUT\_FILE3 ALL\_DDA\_OUTPUT\_FILE

Certain DDAs may not be mapped with ID from our drugs lexicon. We noticed that such DDAs are combinations of two or more drugs. In the current study, we focus on repurposing drugs, not combinations of drugs. Therefore, we removed the drug combinations for a disease of interest.

**Post processing to flip the columns and filter drugs with ID**

$ cat ALL\_DDA\_OUTPUT\_FILE | awk -F'\t' '{ if($3!= "-" && ($2!= "-" && $2!= "")) {print($3"\t"$2)} else if($3!= "-" && ($2== "-" || $2== "")) {print($3"\t"$1)} }' > OUTPUT\_FILE4