**Gene-Disease associations from Comparative Toxicogenomics Database (CTD) and PharmGKB**

**From CTD:**

Download the latest version of gene-disease association from CTD [http://ctdbase.org/downloads/;jsessionid=02DD8D4FEDB4D115A3884A314F35AE97#cg](http://ctdbase.org/downloads/;jsessionid=02DD8D4FEDB4D115A3884A314F35AE97" \l "cg)

**To remove introduction section in the source file**

Java program Preprocessor.java deletes the introduction lines prior to file processing. The program can be executed with the following command:

*$ javac Preprocessor.java*

*$ java Preprocessor INPUT\_FILE OUTPUT\_FILE\_noIntroSection*

**To get selected columns from CTD source file**

We filtered column 1-4 from the source file for further processing. These columns correspond to Chemical name, Chemical ID (MeSH identifier), Disease name and Disease ID (MeSH or OMIM identifier)*.*

*$ cut -f1,2,3,4 OUTPUT\_FILE\_noIntroSection | sort | uniq > CTD\_SOURCE\_FILE\_selectedColumns*

**To check whether the genes are associated with human**

We use the resource **Homo\_sapiens.gene\_info.gz** from Entrez Gene for filtering the records with genes related to human. We use only gene symbol for matching, because of the overlap of synonyms with the gene symbols of other organisms. For example, 11-BETA-HSD3 is a gene symbol in species *Gasterosteus aculeatus* (three-spined stickleback), and the same is an aliase for gene symbol HSD11B1L in *Homo sapiens*. The latest version is downloaded from <ftp://ftp.ncbi.nlm.nih.gov/gene/DATA/GENE_INFO/Mammalia/>.

Java program HumanRecordsRetriever.java retrieves the records with human related genes. The program can be executed with the following commands:

*$ javac HumanRecordsRetriever.java*

*$ java HumanRecordsRetriever Homo\_sapiens.gene\_info CTD\_SOURCE\_FILE\_selectedColumns CTD\_SOURCE\_FILE\_selectedColumns\_human*

**To replace MeSH ID with CUI for diseases**

CTD uses MeSH ID for representing diseases and their synonyms. However, we are interested in unique concept identifier (CUI) from UMLS Metathesaurus for our text mining projects. Java program Conceptname2CUIMapper.java replaces MeSH ID to CUI from UMLS Metathesaurus. The program can be executed with the following command:

*$ javac Conceptname2CUIMapper.java*

*$ java Conceptname2CUIMapper CTD\_SOURCE\_FILE\_selectedColumns\_human CTD\_SOURCE\_FILE\_selectedColumns\_human\_CUIForDiseases*

**To remove the records without matching CUI for diseases**

We observed that **certain records returned null value** for disease CUI while mapping from MeSH ID to CUI. Such records can be removed with the following Linux command.

*$ awk -F"\t" '$3 != "null" { print $1 "\t" $2 "\t" $3 }' CTD\_SOURCE\_FILE\_selectedColumns\_human\_CUIForDiseases > CTD\_SOURCE\_FILE\_selectedColumns\_human\_CUIForDiseases\_filtered*

Output file format: Gene\_symbol \t Disease\_name \t Disease\_CUI

**From PharmGKB:**

Download **relationships.tsv** file from <https://www.pharmgkb.org/>. The resource requires license and can be obtained by contacting PharmGKB team.

**To retrieve gene-disease records those are not annotated only in literature**

Java program GeneDiseaseAssociationProcessorExcludingLiteratureAnnotation.java retrieves gene-disease associations that are not annotated only from literature through text mining approaches. The program can be executed with the following commands:

*$ javac GeneDiseaseAssociationProcessorExcludingLiteratureAnnotation.java*

*$ java GeneDiseaseAssociationProcessorExcludingLiteratureAnnotation relationships.tsv OUTPUT\_FILE\_literatureAnnotationExcluded*

**To assign CUI for diseases**

Java program Conceptname2CUIMapper.java assigns CUI to diseases. The program can be executed with the following commands:

*$ javac Conceptname2CUIMapper.java*

*$ java Conceptname2CUIMapper OUTPUT\_FILE\_literatureAnnotationExcluded OUTPUT\_FILE\_literatureAnnotationExcluded\_CUIForDiseases*

**To remove the records without matching CUI for diseases**

We observed that **certain records are not assigned with CUI** for diseases. Such records can be removed with the following Linux command.

*$ awk -F"\t" '$3 != "null" { print $1 "\t" $2 "\t" $3 }' OUTPUT\_FILE\_literatureAnnotationExcluded\_CUIForDiseases > OUTPUT\_FILE\_literatureAnnotationExcluded\_CUIForDiseases\_filtered*

Output file format: Gene\_symbol \t Disease\_name \t Disease\_CUI

**Unique list of disease-gene associations from both resources**

The output from both resources are combined.

*$ cat CTD\_SOURCE\_FILE\_selectedColumns\_human\_CUIForDiseases\_filtered OUTPUT\_FILE\_literatureAnnotationExcluded\_CUIForDiseases > OUTPUT\_FILE*

**To get unique list based on gene symbol and disease CUI:**

Java program DiseaseGeneAssociationUniqueList.java retrieves a unique list of associations based on gene symbol and disease CUI. The program can be executed with the following command:

*$ javac DiseaseGeneAssociationUniqueList.java*

*$ java DiseaseGeneAssociationUniqueList OUTPUT\_FILE OUTPUT\_FILE\_uniquelist*

Output file format: Gene\_symbol \t Disease\_name \t Disease\_CUI

**Alternate way:** Linux command can be used to get a unique list of association. However, we observed that Linux command was not efficient for the big volume of data from both resources.

**Summary (Oct 2017)**

|  |  |  |  |
| --- | --- | --- | --- |
|  | **CTD** | **PharmGKB** | **Both (unique list)** |
| **# of associations** | 17,819,528 | 2,487 | 17,762,335 |
| **# of diseases** | 5,720 | 443 | 5,774 |
| **# of genes** | 21,990 | 940 | 22,028 |