COG annotation guide

Yu-Wei Wu
Graduate Institute of Biomedical Informatics
Taipei Medical University
yuwei.wu@tmu.edu.tw

Introduction

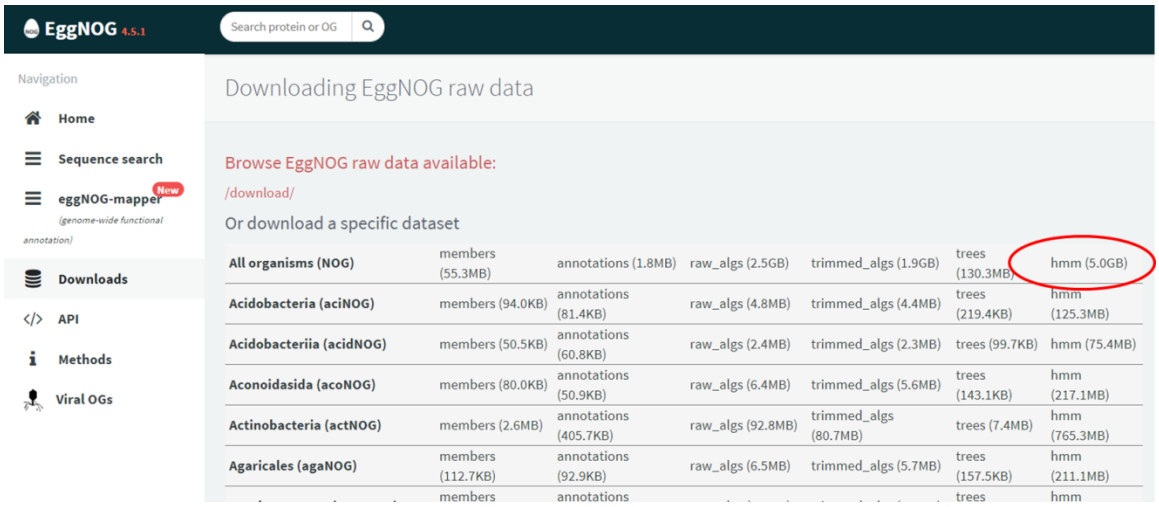
This tutorial provides the detailed steps for annotating COGs for prokaryotic genomes.

Prerequisite:

- HMMER (http://hmmer.org/)
    - Download the latest HMMER for your platform
- Prodigal (http://prodigal.ornl.gov/)

Steps

1. Download the HMM files from eggNOG website (http://eggnogdb.embl.de/#/app/downloads) and unzip it (may take a while as the file is big—we are only use a small fraction of it afterward)



You can also use command “wget” to get the hmm file (remember to use the real link address since eggNOG database will not always stay at version 4.5)

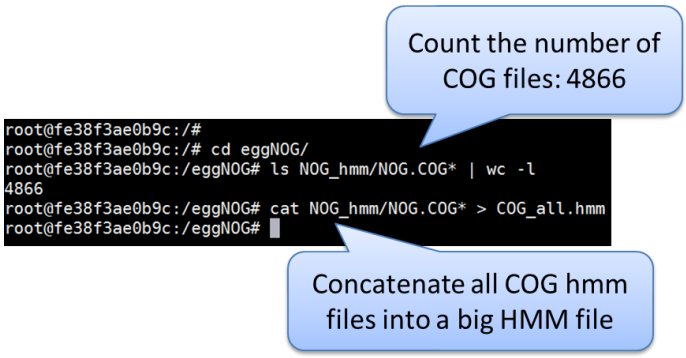
```
root@fe38f3ae0b9c:/#
root@fe38f3ae0b9c:/# wget http://eggnogdb.embl.de/download/eggnog_4.5/data/NOG/NOG.hmm.tar.gz
--2017-07-10 04:09:02--  http://eggnogdb.embl.de/download/eggnog_4.5/data/NOG/NOG.hmm.tar.gz
Resolving eggnogdb.embl.de (eggnogdb.embl.de)... 194.94.44.96
Connecting to eggnogdb.embl.de (eggnogdb.embl.de)|194.94.44.96|:80... connected.
HTTP request sent, awaiting response... 200 OK
Length: 5397320832 (5.0G) [application/octet-stream]
Saving to: 'NOG.hmm.tar.gz'

NOG.hmm.tar.gz                100%[=====================================================>]   5.03G

2017-07-10 04:16:17 (11.9 MB/s) - 'NOG.hmm.tar.gz' saved [5397320832/5397320832]

root@fe38f3ae0b9c:/# tar zxvf NOG.hmm.tar.gz
```

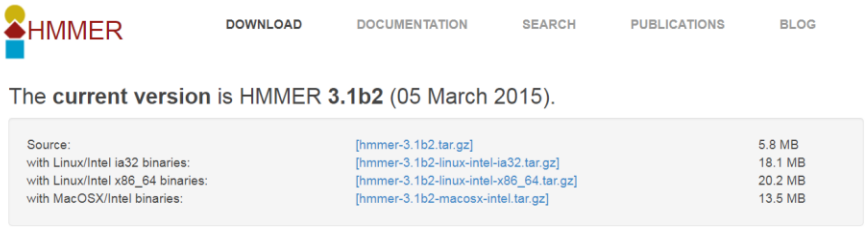
2. There are A LOT OF hmm files in the unzipped folder—possibly over a million. We only want those associated with COGs for now—I still do not know how to make use of other hmm files.



One good attribute about HMMER files is that you can simply concatenate the files into a big file. Later we need to compress this file for faster search.

3. Clone the COGmapper repository from the github website
    - git clone https://github.com/yuwwu/COGmapper.git

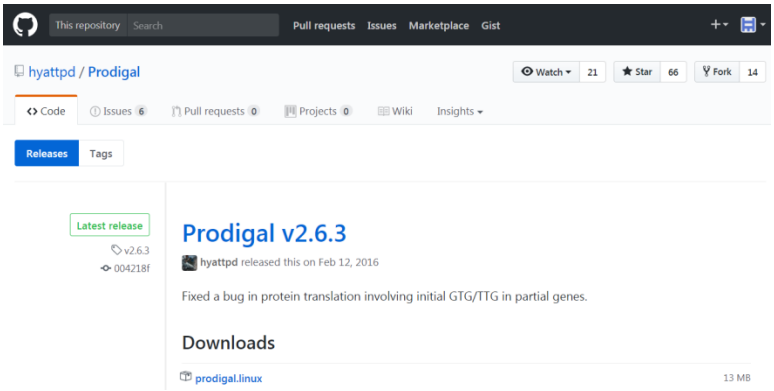
4. Download and install HMMER and Prodigal
    - HMMER
        - Download your HMMER distribution from the HMMER download page (If you are using Linux, most likely the viable choice is Linux/Intel x86_64)
        - http://hmmer.org/download.html



        - You should be able to find all executables in your HMMER binary folders unless you downloaded source files.

```
root@fe38f3ae0b9c:/#
root@fe38f3ae0b9c:/# cd hmmer-3.1b2-linux-intel-x86_64/binaries/
root@fe38f3ae0b9c:/hmmer-3.1b2-linux-intel-x86_64/binaries# ls
alimask       esl-alimerge   esl-compstruct  esl-selectn   esl-ssdraw      hmmc2             hmmlogo    hmmsim       nhmmscan
esl-afetch    esl-alipid     esl-construct   esl-seqrange  esl-stranslate  hmmconvert        hmmpgmd    hmmstat      phmmer
esl-alimanip  esl-alistat    esl-histplot    esl-seqstat   esl-weight      hmmemit           hmmpress   jackhmmer
esl-alimap    esl-cluster    esl-mask        esl-sfetch    hmmalign        hmmerfm-exactmatch hmmscan   makehmmerdb
esl-alimask   esl-compalign  esl-reformat    esl-shuffle   hmmbuild        hmmfetch          hmmsearch  nhmmer
root@fe38f3ae0b9c:/hmmer-3.1b2-linux-intel-x86_64/binaries#
```

- Prodigal
  - Download the most suitable executable from Prodigal github website (https://github.com/hyattpd/prodigal/releases/)



  - Change the executable filename into “prodigal” and change its mode via “chmod”

```
root@fe38f3ae0b9c:/bin#
root@fe38f3ae0b9c:/bin# mv prodigal.linux prodigal
root@fe38f3ae0b9c:/bin# chmod 0755 prodigal
root@fe38f3ae0b9c:/bin#
```

5. Setup HMMER and Prodigal executable locations
   There are two ways to setup the executables.
   (1) Simply place their paths into the system paths so that you can run the programs anywhere, or
   (2) Add their paths into the setting file (explained in step 7)
   - Note: you need to choose either (1) or (2) but not necessarily both. COGmapper will automatically find the programs and will only report errors if both (1) and (2) fails.

6. Clone the github repo (https://github.com/yuwwu/COGmapper.git)

```
root@6346910871b2:/#
root@6346910871b2:/# git clone https://github.com/yuwwu/COGmapper.git
Cloning into 'COGmapper'...
remote: Counting objects: 21, done.
remote: Compressing objects: 100% (18/18), done.
remote: Total 21 (delta 2), reused 18 (delta 2), pack-reused 0
Unpacking objects: 100% (21/21), done.
Checking connectivity... done.
root@6346910871b2:/# cd COGmapper/
root@6346910871b2:/COGmapper# ls
COG.list  COGcategory.txt  COGlist.txt  LICENSE  README  map_COG.pl  setting  test_genomes
root@6346910871b2:/COGmapper#
```

7. (Skip this step if you have already add the Prodigal and HMMER paths into system paths)
   If you want to add the program executables into setting…

(a) Locate the **binary executables** of both HMMER and Prodigal
(b) Use "pwd" to print out the exact paths
(c) Replace the paths with the actual paths in the "setting" file

```
root@6346910871b2:~#
root@6346910871b2:~# cd /hmmer-3.1b2-linux-intel-x86_64/binaries/
root@6346910871b2:/hmmer-3.1b2-linux-intel-x86_64/binaries# pwd
/hmmer-3.1b2-linux-intel-x86_64/binaries
root@6346910871b2:/hmmer-3.1b2-linux-intel-x86_64/binaries# cd /bin
root@6346910871b2:/bin# pwd
/bin
root@6346910871b2:/bin#
root@6346910871b2:/bin#
root@6346910871b2:/bin# cd /COGmapper/
root@6346910871b2:/COGmapper# cat setting
[HMMER3] /usr/bin/hmmer-3.1/binaries
[Prodigal] /usr/bin/prodigal-2.60
root@6346910871b2:/COGmapper#
```

8. Make a directory "data" under COGmapper, move the COG_all.hmm file that we just made into the data folder, and then use "hmmpress" command to press it.

```
root@6346910871b2:/#
root@6346910871b2:/# cd COGmapper/
root@6346910871b2:/COGmapper# perl map_COG.pl
Please download the COG hmm files from eggNOG website and concatenate them into COG_all.hmm as instructed in the tutorial PDF fi
le
Please place them under 'data' folder at the COGmapper folder.
root@6346910871b2:/COGmapper# mkdir data
root@6346910871b2:/COGmapper# mv /eggNOG/COG_all.hmm data/
root@6346910871b2:/COGmapper# cd data
root@6346910871b2:/COGmapper/data# /hmmer-3.1b2-linux-intel-x86_64/binaries/hmmpress COG_all.hmm
Working...
```

9. I have put some test genomes under "test_genomes" folder, in which three are genomic files and one consists of predicted genes (amino acids). I also have a list file listing the four genome files.

```
root@6346910871b2:/COGmapper# cd test_genomes/
root@6346910871b2:/COGmapper/test_genomes# ls -l
total 9404
-rw-r--r-- 1 root root 3548591 Jul 10 10:44 Deinococcus_maricopensis_DSM_21211.fasta
-rw-r--r-- 1 root root  865612 Jul 10 10:44 Thermus_thermophilus_HB8.faa
-rw-r--r-- 1 root root 1876251 Jul 10 10:44 Thermus_thermophilus_HB8.fasta
-rw-r--r-- 1 root root 3307055 Jul 10 10:44 Truepera_radiovictrix_DSM_17093.fasta
-rw-r--r-- 1 root root     139 Jul 10 10:44 list
root@6346910871b2:/COGmapper/test_genomes# cat list
Deinococcus_maricopensis_DSM_21211.fasta
Thermus_thermophilus_HB8.faa
Thermus_thermophilus_HB8.fasta
Truepera_radiovictrix_DSM_17093.fasta
```

To run COGmapper, simply use perl to call "map_COG.pl" file and issue "input list" and "output." The final output will be stored in the output file. The map_COG.pl script can be called anywhere from the system—as long as you provided the actual paths to this script.

```
root@637e5afd4427:/COGmapper/test_genomes# perl ../map_COG.pl list out
Processing file Deinococcus_maricopensis_DSM_21211.fasta
--Predicting genes from Deinococcus_maricopensis_DSM_21211.fasta
--Annotating genes in terms of COGs
Processing file Thermus_thermophilus_HB8.faa
--Annotating genes in terms of COGs
Processing file Thermus_thermophilus_HB8.fasta
--Predicting genes from Thermus_thermophilus_HB8.fasta
--Annotating genes in terms of COGs
```

(Warning: the HMMER step will require a while to run due to the number of COG categories. Please be patient in this step or change the thread number as follows…)

Adjust thread number

The thread number for running HMMER can be adjusted by changing the number in the 6th line of the map_COG.pl file.

Output

The output of the COGmapper script is represented as a table, in which the rows indicate each COG categories and the columns are the input genomes. The COG categories are:

COG category Abbreviations	Category descriptions
J	Translation, ribosomal structure and biogenesis
A	RNA processing and modification
K	Transcription
L	Replication, recombination and repair
B	Chromatin structure and dynamics
D	Cell cycle control, cell division, chromosome partitioning
Y	Nuclear structure
V	Defense mechanisms
T	Signal transduction mechanisms
M	Cell wall/membrane/envelope biogenesis
N	Cell motility
Z	Cytoskeleton
W	Extracellular structures
U	Intracellular trafficking, secretion, and vesicular transport
O	Posttranslational modification, protein turnover, chaperones
C	Energy production and conversion
G	Carbohydrate transport and metabolism
E	Amino acid transport and metabolism
F	Nucleotide transport and metabolism
H	Coenzyme transport and metabolism
I	Lipid transport and metabolism

P	Inorganic ion transport and metabolism
Q	Secondary metabolites biosynthesis, transport and catabolism
R	General function prediction only
S	Function unknown

The output file consists of the **number** of genes hit to each category for each genome.