MyTaxa Manual

Author: Chengwei Luo, Luis M. Rodriguez-R., and Konstantinos T. Konstantinidis

Citation: Chengwei Luo, Luis M. Rodriguez-R., and Konstantinos T. Konstantinidis, MyTaxa: an advanced taxonomic classifier for genomic and metagenomic sequences, *Nucleic Acids Research*, *in press*

[Installation, standalone only]

To start, you will need a C++ compiler and GNU make to be installed in your computer. To check if you have them, type 'g++' and 'make' in the terminal. If you don't have them, go to http://gcc.gnu.org for g++, and http://www.gnu.org/software/make/ for GNU make.

You can either download the zip archive directly from the github repository:

```
https://github.com/luo-chengwei/MyTaxa/archive/master.zip
```

Or, you can clone it from github (if you want to have git installed, please go to http://git-scm.com/), go to the terminal and run:

```
$ git clone git://github.com/luo-chengwei/MyTaxa
```

This command will create a MyTaxa directory on your local machine, and then you need to 'cd' into this directory and run:

```
$ make
```

This command will create the binary 'MyTaxa'.

There is one more necessary step before you can run MyTaxa, to obtain the database (DB) of gene weights. You can either download DB from the following address:

http://enve-omics.ce.gatech.edu/mytaxa/download/db.latest.tar.gz

or, you can run this utility script:

```
$ python utils/download_db.py
```

Then you need to decompress the DB by running:

$ tar xvzf db.latest.tar.gz

Note that you should have all the DB files at ./MyTaxa/db/

Now you should be ready to run MyTaxa. To verify it, you can run:

$ ./MyTaxa

This should print out a help/usage menu, without any error message.

[Running MyTaxa]

Note: there is a sample input file you can test at ./MyTaxa/example/.

To run MyTaxa, the only input file required is a tabular blast-like file with the results from a search against a reference database; but there are a few things you need to make sure before running the search:

- The input file for the search should be genes encoded on the query sequences. If the query sequences are not annotated yet, you need to call the genes first and then do the search of the genes against the reference database. You can run *ab-initio* gene prediction software such as MetaGeneMark, Prodigal, FragGeneScan, *etc* (1-3) to get the genes encoded on the query sequences on your own; or, you can use utils/metaxa_prep.py for this purpose, which essentially uses Prodigal to call all the genes. **You can use the gene prediction results in .gff format from MetaGeneMark/GeneMark as a start point with the webserver; to follow this route, you can now jump to the webserver subsection below.** (**Note:** since you will rely on the webserver for the homology search, your priority will be low and your input size will be limited; also you cannot customize your reference database.). Otherwise, you can do the search on your own (See below).

- The naming of the genes/entries in the input multi-fastA file is recommend to follow this scheme (you don't have to follow this rule, as long as you can retrieve gene_id).

>contig_xyz|gene_id

for instance, by running **metaxa_prep.py,** the genes predicted have names like:

>contig1|1915-2331|1

which means this gene is from "contig1" (contigID) and the gene name is "1915-2331|1" (gene ID); you can use other alphanumerical schemes to name your contigID and gene ID, as long as the contig ID does not contain the '|' sign, and the '|' sign separates contigID from gene ID.

- Several options for the reference database to search the genes against are available as long as they are gi number referenced (genInfo identifier). For example, in the NR database of NCBI, an entry looks like:

>gi|21305377|gb|AAM45611.1|AF384285_1 (AF384285) envelope protein [Human immunodeficiency virus type 1]

The numeric id "21305377" is the gi number you need in the results, and that's how MyTaxa will recognize which the matching genes is and find its weights in the DB.

In the example sample folder, you can find "example.fa" as an example input file for the search.

After you have the fasta file and reference database ready, you can run the blast search or any other search engine you like, as long as you can produce a blast tabular format-like file (check the "example.blat" as an example). **The output file can be also used as another entry point of the webserver; in this case, you may skip the rest, which are specific to the standalone version, and jump to the webserver subsection.**

Then you should modify the output file from the search by adding three columns to the end of each match. These are:

- contigID
- geneID
- matched gene's gi number

You can use **utils/infile_convert.pl** to modify the tabular blast output into the MyTaxa input file (run "perl utils/infile_convert.pl" to get it print out the usage menu). You can check "example.mytaxa.input.txt" as an example of this file.

From this point, you can either use the standalone or the webserver to generate the output of MyTaxa.

Webserver:

Webserver address: http://enve-omics.ce.gatech.edu/mytaxa/submit

There are two input files you will need for the webserver as described above:

GFF file of gene prediction;
Tabular blast output like file;

You'll need to fill out a simple form (name, email, job name), and you can customize some of the parameters (e.g., bit-score, alignment length). Then you can hit the "submit" button to get the job in queue. When it is finished, you will be notified of the results by email.

Standalone:

In the standalone version, you can run this command on the terminal:

$ ./MyTaxa <input_file> <output_file> <score_cutoff> <num_of_matches_to_use>

where the input file would be the modified blast output-like tabular file, the output file is where the MyTaxa prediction will be found. There are two parameters you need to set are: the score cutoff (recommend value: 0.5) and number of matches to use in the analysis (recommend value: 5). MyTaxa's run will generate an output that is explained below.

[The output format and visualization]

There is a toy example available: ./MyTaxa/example/example.mytaxa

Each query sequence will have two lines in the output, and they are:

- Line 1: <query name> <lowest_level> <score> <lowest_level_taxonomy_id>
- Line 2: the taxonomy path from (super)-kingdom down to the lowest taxon predicted.

You can parse the output file to obtain the relative abundance of identified taxa at the phylum/genus/species levels by running **utils/MyTaxa.distribution.pl**. (run "perl utils/MyTaxa.distribution.pl" to print out detailed usage information). Furthermore, with an additional reads-mapping-to-contig file, you can directly visualize the relative abundance using **utils/MyTaxa.distribution.pl,** please refer to its help menu for details.

In the example folder, you can find example files: "example.mytaxa.Phylum.txt", "example.mytaxa.Genus.txt", and "example.mytaxa.Species.txt".

The output from MyTaxa can be also visualize in Krona (http://sourceforge.net/p/krona/home/krona/) by using the utility script **utils/mytaxa2krona.py:**

$ python utils/mytaxa2krona.py <mytaxa_output> <krona_input>

the krona_input file generated can be used to produce the interactive pie-chart by Krona (4). A toy example for this file is provided at: ./MyTaxa/example/example.krona_input.txt.

Then you can run Krona as:

$ ktImportText –o <output_html> <krona_input>

the output_html would be the output from Krona that you can open in your webbrowser to visualize the composition of the MyTaxa output. A sample example is provided at: ./MyTaxa/example/example.krona_input.txt

[Trouble shooting]

See FAQ page at: http://enve-omics.ce.gatech.edu/mytaxa/faq

You can refer to the wiki page where some of the issues/fix would be reported:

https://github.com/luo-chengwei/MyTaxa/wiki

References

1. Rho M, Tang H, & Ye Y (2010) FragGeneScan: predicting genes in short and error-prone reads. *Nucleic acids research* 38(20):e191.
2. Zhu W, Lomsadze A, & Borodovsky M (2010) Ab initio gene identification in metagenomic sequences. *Nucleic acids research* 38(12):e132.
3. Hyatt D*, et al.* (2010) Prodigal: prokaryotic gene recognition and translation initiation site identification. *BMC bioinformatics* 11:119.
4. Ondov BD, Bergman NH, & Phillippy AM (2011) Interactive metagenomic visualization in a Web browser. *BMC bioinformatics* 12:385.