User tutorial for HGTphyloDetect

Table of Contents

How to install HGTphyloDetect?

Install HGTphyloDetect:

git clone https://github.com/SysBioChalmers/HGTphyloDetect.git

or

pip install git+https://github.com/SysBioChalmers/HGTphyloDetect.git

To install all the required Python packages, run the command line:

cd HGTphyloDetect

pip install -r requirements.txt

Any examples for quick start?

We provide a user-friendly example for small test, users just need to prepare a FASTA file including protein id and protein sequence, note that protein id should be from the GenBank protein database.

(1) If you are now in the HGTphyloDetect directory, just enter into the folder (example) via the command line:

cd example

(2) Then users can run the script for the input file:

python HGT_workflow.py input.fasta

(3) Finally, our software could generate the output results as a file under the folder (example) for this gene/protein. The output file includes some important information, i.e., Alien index, E value and donor information. For example:

Gene/Protein	Alien index	E value	Donor id	Donor taxonomy
AAT92670	199.18	3.15e-87	WP_208929673	Bacteria/Firmicutes

How to run high-throughput HGT identification?

Here, we recommend users to run BLASTP process separately, because it may cost too much time especially if users want to execute BLASTP for many genes/proteins, such as 1,000 or 10,000. For BLASTP, users can enter into the folder (main), and run:

python blastp.py input.fasta

Then users could get BLASTP result under the generated folder (blastp_files).

To identify potential genes that have been horizontally acquired from evolutionarily distant organisms (i.e., prokaryote to eukaryote)

Users can enter into the directory (main) and run the command line:

```
cd main
python HGT_workflow_distant.py input.fasta
```

This python script mainly executes these three steps as follows:

- **Step I**: BLASTP hits were parsed to retrieve associated taxonomic information using the NCBI's taxonomy database.
- **Step II**: Calculate Alien Index (AI) values and out_pct based on the above information.
- **Step III**: Output the horizontal gene transfer (HGT) high-throughput identification results.

To identify potential genes that have been horizontally acquired from more closely related organisms (e.g., eukaryote to eukaryote)

Users can enter into the directory (main) and run the command line:

```
cd main
python HGT_workflow_close.py input.fasta
```

How to run phylogenetic analysis?

Overview: To corroborate the accurate identification of HGT genes by their Alien Index (AI) values and out_pct, we extended HGTphyloDetect with a comprehensive phylogenetic analysis pipeline. First, the top 300 homologs with different taxonomic species names are selected from the BLASTP hits for each query sequence. These homologs are aligned with MAFFT v7.310 (Katoh, et al., 2005) using default settings for multiple sequence alignment, while ambiguously aligned regions are removed with trimAl using its '-automated1' option (Capella-Gutiérrez, et al., 2009). Phylogenetic trees are constructed from these alignments using IQ-TREE v1.6.12 (Nguyen, et al., 2015) with 1000 ultrafast bootstrapping replicates. Subsequently, the phylogenetic tree is rooted at the midpoint using the R packages ape v5.4-1 (Paradis, et al., 2004) and phangorn v2.5.5 (Schliep, 2011). Finally, the resulting phylogenies are visualized using iTol v5 (https://itol.embl.de/) to assess the mode of transmission of each gene.

To run phylogenetic analysis, we provided all the scripts under this folder (phylogenetics) in the repository. Here, we used an example (one important protein YOL164W in *Saccharomyces cerevisiae*) for easy understanding. Users could directly run the scripts provided here, and finally

three files could be output: YOL164W_midpoint.tree, YOL164W_midpoint-font.txt and YOL164W_midpoint-color.txt. Note that the first file is the generated phylogenetic tree via the open-source software, another two additional files are the iTol annotation files for this tree. After that, users can use the iTol website (https://itol.embl.de/) to visualize the phylogenetic tree. The detailed steps are shown as follows:

- **Step I**

If you are now located in the HGTphyloDetect directory, first enter into the phylogenetics directory via the command line:

```
cd phylogenetics
```

- **Step II**

Obtain a specific FASTA file including a list of homolog identifiers and related sequences (sequences with more than 80% similarity are eliminated in this step) via the following command line:

```
python scripts/HGT_homologs_sequence.py input/YOL164W.fasta
```

Input file: the basic FASTA file YOL164W.fasta under the input folder.

Output file: YOL164W_homologs.fasta under the input folder.

- **Step III**

Install the MAFFT (Multiple alignment program for amino acid or nucleotide sequences) software according to the tutorial (https://mafft.cbrc.jp/alignment/software/source.html)

And then run multiple sequence alignment by MAFFT via the command line:

```
mafft --thread 6 --
auto ./input/YOL164W_homologs.fasta > ./intermediate/YOL164W_aln.fasta
```

Input file: YOL164W_homologs.fasta under the input folder.

Output file: YOL164W_aln.fasta under the intermediate folder.

- **Step IV**

Install the trimAl (a tool for automated alignment trimming in phylogenetic analysis) program according to the tutorial (https://github.com/inab/trimal). Note that the trimAl should be added to PATH.

And then remove ambiguously aligned regions with trimAl using its '-automated1' option via the command line:

trimal -in ./intermediate/YOL164W_aln.fasta -
out ./intermediate/YOL164W_aln_trimmed.fasta -automated1

Input file: YOL164W_aln.fasta under the intermediate folder.

Output file: YOL164W_aln_trimmed.fasta under the intermediate folder.

- **Step V**

Install the IQ-TREE (an efficient software for phylogenetic inference) software based on the documentation (http://www.iqtree.org/doc/Quickstart). Note that the IQ-TREE should be added to PATH.

Phylogenetic tree construction from the sequence alignments using IQ-TREE with 1000 ultrafast bootstrapping replicates via the command line:

iqtree -nt 6 -st AA -s ./intermediate/YOL164W_aln_trimmed.fasta -m TEST -mrate G4 -keep-ident -bb 1000 -pre ./intermediate/YOL164W

Input file: YOL164W_aln_trimmed.fasta under the intermediate folder.

Output file: a list of files related to the phylogenetic tree.

- **Step VI**

Install R programming language at first. Then, copy & paste the following commands to your R command prompt to install the R packages ape and phangorn:

install.packages("ape",repos="https://cloud.r-project.org",quiet=TRUE)

install.packages("phangorn",repos="https://cloud.r-project.org",quiet=TRUE)

Next, the phylogenetic tree is rooted at the midpoint using the R packages ape and phangorn via the command line:

Rscript scripts/midpoint_tree.R YOL164W

Input file: YOL164W.treefile under the intermediate folder.

Output file: YOL164W_midpoint.tree under the intermediate folder.

- **Step VII**

Install Perl and BioPerl, and then run the following command line to generate two annotation files for the visualization of the phylogenetic tree:

perl scripts/create_iTOL_config.pl ./intermediate/YOL164W_midpoint.tree

Input file: YOL164W_midpoint.tree under the intermediate folder.

Output file: YOL164W_midpoint-font.txt and YOL164W_midpoint-color.txt under the intermediate folder.

- **Step VIII**

In order to facilitate the visualization of the phylogenetic tree, users just need to log into the iTol website (https://itol.embl.de/) and then upload these three files (YOL164W_midpoint.tree, YOL164W_midpoint-font.txt and YOL164W_midpoint-color.txt) to the website, and then the phylogenetic tree could be output by the 'Export' function on the website.

Input file: YOL164W_midpoint.tree, YOL164W_midpoint-font.txt and YOL164W_midpoint-color.txt under the intermediate folder. Here, I also put them into the output folder.

Output file: YOL164W_circle_tree.pdf or YOL164W_rectangular_tree.pdf under the output folder.

If users choose the 'Circular' mode for the phylogenetic tree visualization, then the tree would be generated like this (**Figure 1**):

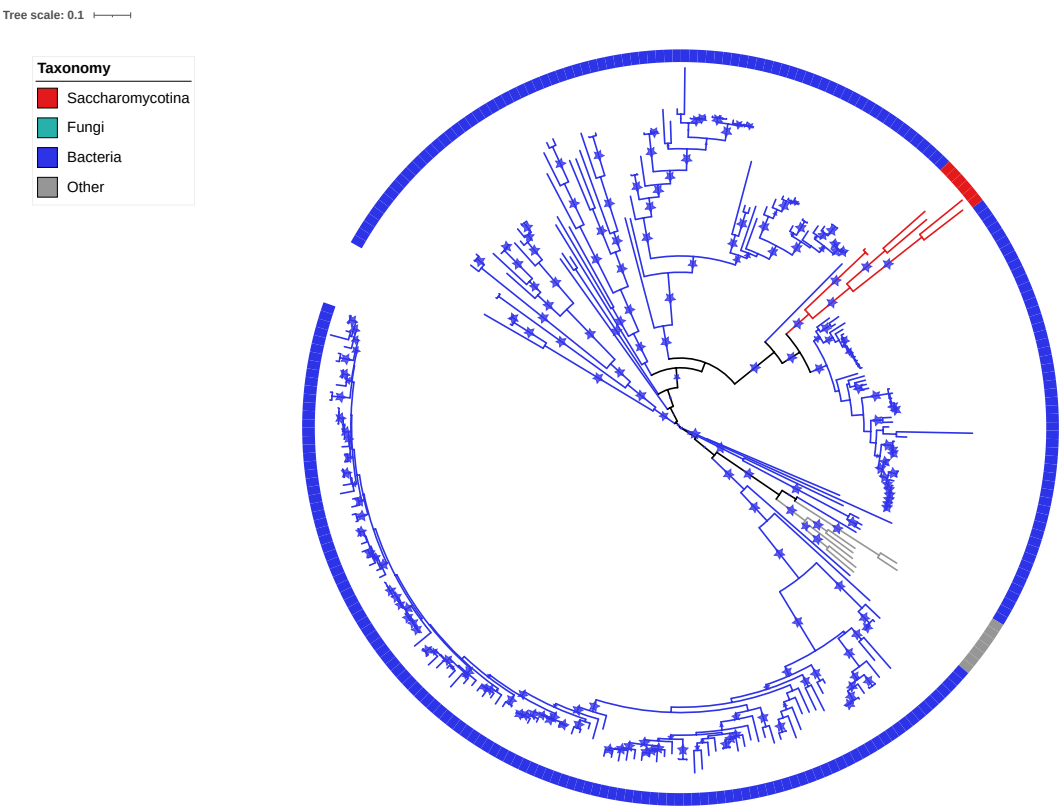

Figure 1. Maximum Likelihood (ML) phylogeny of a protein YOL164W in *Saccharomyces cerevisiae*. Branches with bootstrap support higher than 80% are shown by star.

If users choose the 'Rectangular' mode for the phylogenetic tree visualization, then the tree would be generated like this (**Figure 2**):

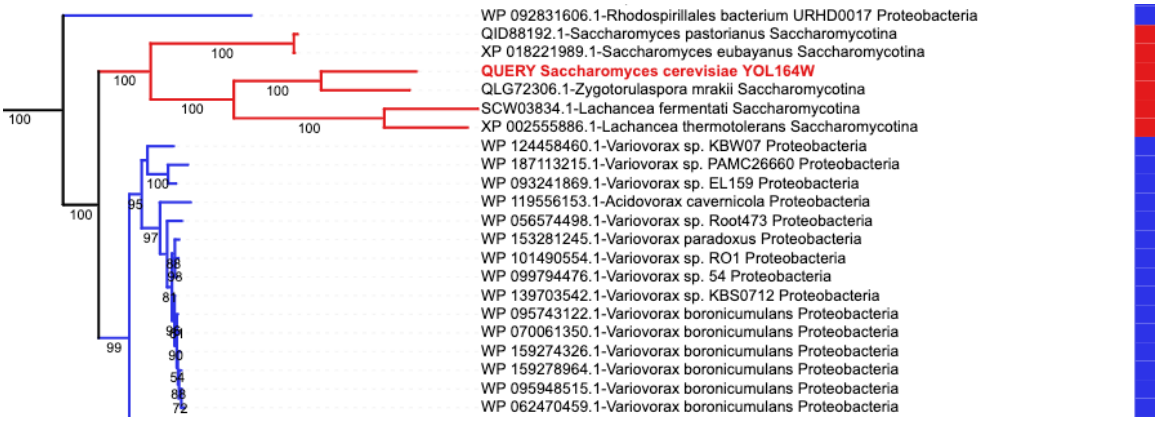

Figure 2. The detailed phylogenetic tree represents pruned ML phylogeny depicting the phylogenetic relationship between the protein YOL164W from the *Saccharomycotina* subphylum and its close relatives from other bacteria.

Besides, we have also compiled the separate steps (step I to step VII) into one shell script (generate_tree.sh) shown in **Figure 3**. Users can directly run these steps by using the command line:

sh generate_tree.sh

But the prerequisite is that users have installed all the external software regarding phylogenetic analysis.

```
#!/bin/bash

echo 'Begin to run!!!'
cd  ../
python scripts/HGT_homologs_sequence.py input/YOL164W.fasta
mafft --thread 6 --auto ./input/YOL164W_homologs.fasta > ./intermediate/YOL164W_aln.fasta
trimal -in ./intermediate/YOL164W_aln.fasta -out ./intermediate/YOL164W_aln_trimmed.fasta -automated1
iqtree -nt 6 -st AA -s ./intermediate/YOL164W_aln_trimmed.fasta -m TEST -mrate G4 -keep-ident -bb 1000 -pre ./intermediate/YOL164W
Rscript scripts/midpoint_tree.R YOL164W
perl scripts/create_iTOL_config.pl ./intermediate/YOL164W_midpoint.tree
echo 'Yep, finish!!!'
```

Figure 3. Screenshot of the integrative shell script about running phylogenetic analysis for horizontal gene transfer (HGT) event.

Looking for help?

If you encounter any issues, feel free to contact the software developer: Dr. Le Yuan (leyu@chalmers.se)