User tutorial for HGTphyloDetect

Installation

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:

```
cd HGTphyloDetect
pip install -r requirements.txt
```

Example – 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 db.

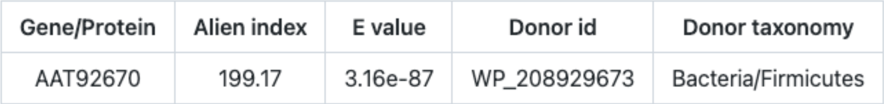
(1) If you are now in the HGTphyloDetect directory, just enter into the folder example via the command line:

```
cd example
```

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

```
python HGT_workflow.py input.fasta
```

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



Gene/Protein	Alien index	E value	Donor id	Donor taxonomy
AAT92670	199.17	3.16e-87	WP_208929673	Bacteria/Firmicutes

High-throughput running

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, i.e., 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 at the same folder.

To identify potential genes that have been horizontally acquired from evolutionary distant organisms (i.e., bacteria to yeast)

Users can run enter into the folder (main) and run this command:

```
python HGT_workflow_distant.py input.fasta
```

This python script mainly executes these three steps as follows.

- Step 1: BLAST hits were parsed to retrieve associated taxonomic information using the NCBI's taxonomy database.
- Step 2: Calculate Alien Index (AI) values and out_pct based on the above information.
- Step 3: 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 folder (main) and run this command:

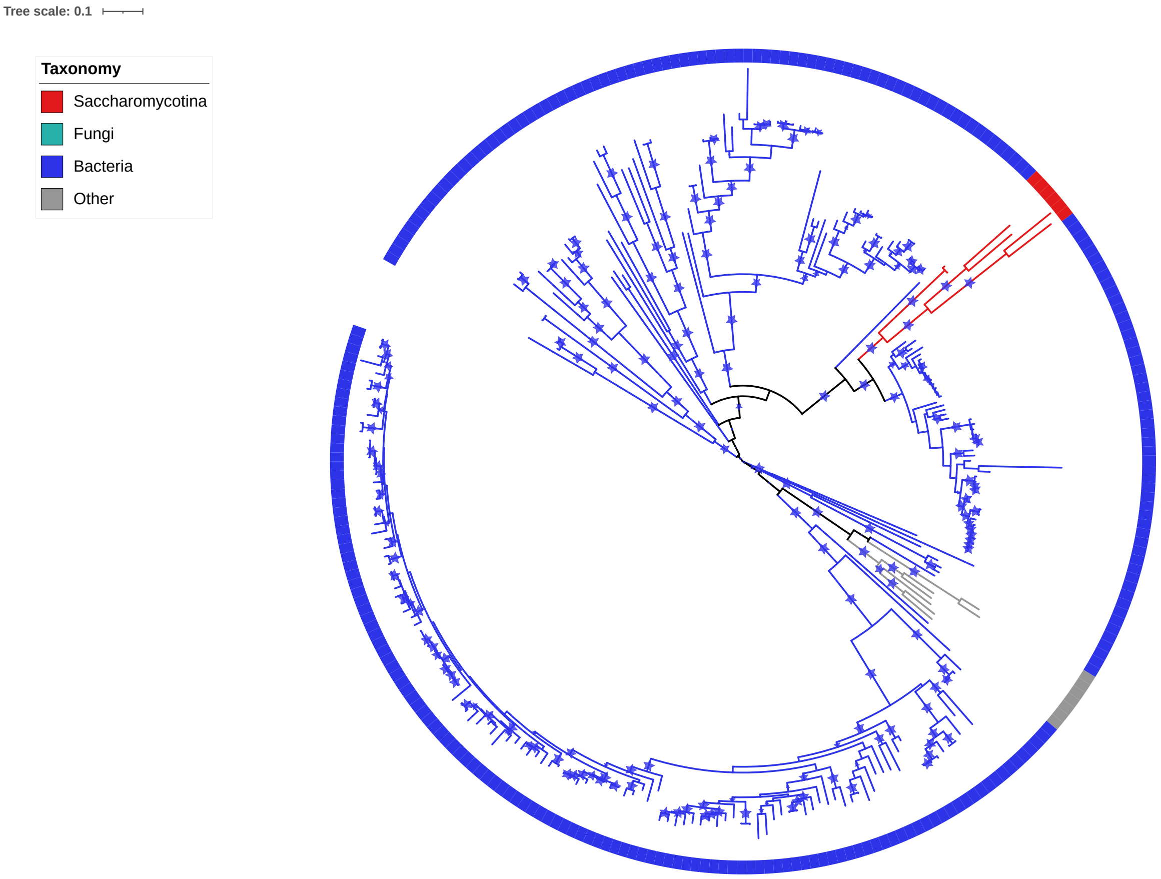
```
python HGT_workflow_close.py input.fasta
```

Phylogenetic analysis

To corroborate the accurate identification of HGT genes by their AI as described above, we extended HGTphyloDetect with a 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, each 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 one example (the important protein YOL164W in *Saccharomyces cerevisiae*) for easy understanding. Users could directly run the scripts provided here, and then three files could be output: YOL164W_midpoint.tree, YOL164W_midpoint-font.txt, YOL164W_midpoint-color.txt. Note that the first file is the generated phylogenetic tree, 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. It is very easy, users just need to drag these three files to the website, and then the phylogenetic tree could be output.

If users choose ‘circle’ for tree visualization, then the tree would be like this:


Tree scale: 0.1
Taxonomy
Saccharomycotina
Fungi
Bacteria
Other