

Aug 28, 2024

Version 2

Annotation for Fungi V.2

DOI

<https://dx.doi.org/10.17504/protocols.io.e6nvw14nwlmk/v2>

Sebastian Bassi¹, Virginia Gonzalez¹, Tristan Yang²

¹Toyoko; ²Keck Graduate Institute



Sebastian Bassi

Toyoko

Create & collaborate more with a free account

Edit and publish protocols, collaborate in communities, share insights through comments, and track progress with run records.

Create free account

OPEN  ACCESS



DOI: <https://dx.doi.org/10.17504/protocols.io.e6nvw14nwlmk/v2>

Protocol Citation: Sebastian Bassi, Virginia Gonzalez, Tristan Yang 2024. Annotation for Fungi. protocols.io <https://dx.doi.org/10.17504/protocols.io.e6nvw14nwlmk/v2> Version created by **Sebastian Bassi**

License: This is an open access protocol distributed under the terms of the **[Creative Commons Attribution License](#)**, which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited

Protocol status: Working

We use this protocol and it's working

Created: August 28, 2024

Last Modified: August 28, 2024

Protocol Integer ID: 106542

Keywords: docker, bioinformatics, dnalinux, fungi, annotation for fungi protocol, fungi genome, fungi protocol, annotation

Abstract

Protocol to annotate a fungi genome

Setup

1 Install Docker

If you don't have Docker already, install it. There are two versions, Docker Engine (also known as CE) and Docker Desktop. The Desktop version is more user friendly but since may require commercial license for large enterprise, this tutorial is based on the Docker engine. Both version will work in this protocol. Linux users can install both Docker CE and Desktop, while macOS and Windows users should install Docker Desktop.

Follow the installation instructions from <https://docs.docker.com/engine/install/>

2 Get your data ready

You will need fastq data (long reads), short reads, and the assembly data. In the following code, the assembly data file is called *assembly.fasta*. The long reads file is called *ID.fastq*. The short reads should be two files (*ID_R1.fastq.gz* and *ID_R2.fastq.gz*). If you have more files for short reads, you can concatenate them so you end up with 2 files. For example, if you have *ID_L001_R1.fastq.gz*, *ID_L002_R1.fastq.gz*, *ID_L001_R2.fastq.gz*, *ID_L002_R2.fastq.gz*, you can concatenate them with these commands:

```
cat ID_L001_R1.fastq.gz ID_L002_R1.fastq.gz > ID_R1.fastq.gz
cat ID_L001_R2.fastq.gz ID_L002_R2.fastq.gz > ID_R2.fastq.gz
```

All files should be inside a directory, for example: *your_dir*

Inside *your_dir* there should be three directories: *funannotate_prep*, *funannotate* and *funannotate_ipsout*.

You can create them with this command:

```
mkdir -p your_dir/funannotate/ipsout && mkdir
your_dir/funannotate_prep
```

3 Download FamDB HDF5 database and GeneMark license

FamDB HDF5 database is needed for the **RepeatMasker** step. This database is partitioned by taxonomic groups, the partition needed for Fungi is partition number 0, for



more information about partitions read this file:

 README.txt 2.1KB

FamDB HDF5 database can be downloaded [from here](#).

Bash commands to download, unzip and mv the database to */your_dir*:

```
wget
https://www.dfam.org/releases/Dfam_3.8/families/FamDB/dfam38_full.
0.h5.gz
gunzip dfam38_full.0.h5.gz
mv dfam38_full.0.h5 /you_dir
```

If you don't have a **GeneMark** license, get it [from this page](#). License key file should be named *gm_key* and located in */your_dir*. This license is need to run the **Funannotate Predict** step.

Run sspace_longread

- 4 Run the following command (replace */your_dir* for the base directory where you have your data

```
docker run -it -v /your_dir:/ftmp dnalinux/sspace_longread:latest
perl SSPACE-LongRead.pl -c /ftmp/assembly.fasta -p /ftmp/ID.fastq
-b /ftmp/outputperlID
```

Run Gapcloser

- 5 Run the following command (replace */your_dir* for the base directory where you have your data

```
docker run -it -v /your_dir:/ftmp dnalinux/lr_gapcloser:latest
bash /LR_Gapcloser/src/LR_Gapcloser.sh -i
/ftmp/outputperlID/scaffolds.fasta -l /ftmp/ID.fastq -o
/ftmp/ID_lr-gapcloser
```



Run BWA Index

- 6 Run the following command (replace /your_dir for the base directory where you have your data)

```
docker run -it -v /your_dir:/ftmp dnalinux/bwa:0.7.17-3-deb bwa
index /ftmp/ID_lr-gapcloser/iteration-1/gapclosed.fasta
```

Run fastp

- 7 Run the following command (replace /your_dir for the base directory where you have your data)

```
docker run -it -v /your_dir:/ftmp dnalinux/fastp:0.23.4 fastp --
in1 /ftmp/ID_R1.fastq.gz --in2 /ftmp/ID_R2.fastq.gz --out1
/ftmp/ID_R1_trim.fastq.gz --out2 /ftmp/ID_R2_trim.fastq.gz
```

Run BWA mem

- 8 Run the following command (replace /your_dir for the base directory where you have your data). Replace CPU for your CPU count.

```
docker run -it -v /your_dir:/ftmp dnalinux/bwa:0.7.17-3-deb
/bin/bash -c "bwa mem -t CPU /ftmp/ID_lr-gapcloser/iteration-
1/gapclosed.fasta /ftmp/ID_R1_trim.fastq.gz
/ftmp/ID_R2_trim.fastq.gz > /ftmp/ID_aligned_reads.sam"
```

Run SAMTOOLS

- 9 SAMTOOLS View, Sort and Index

Run the following command (replace /your_dir for the base directory where you have your data).



```
docker run -it -v /your_dir:/ftmp dnalinux/samtools:1.20-3-deb
/bin/bash -c "samtools view -Sb /ftmp/ID_aligned_reads.sam >
/ftmp/ID_aligned_reads.bam"

docker run -it -v /your_dir:/ftmp dnalinux/samtools:1.20-3-deb
/bin/bash -c "samtools sort /ftmp/ID_aligned_reads.bam -o
/ftmp/ID_sorted_aligned_reads.bam"

docker run -it -v /your_dir:/ftmp dnalinux/samtools:1.20-3-deb
/bin/bash -c "samtools index /ftmp/ID_sorted_aligned_reads.bam"
```

Pilon

- 10 Run the following command (replace /your_dir for the base directory where you have your data).

```
docker run -it -v /your_dir:/ftmp dnalinux/pilon:1.24-3-deb pilon
--genome /ftmp/ID_lr-gapcloser/iteration-1/gapclosed.fasta --frags
/ftmp/ID_sorted_aligned_reads.bam --output /ftmp/ID_polished
```

Funannotate

- 11 Funannotate Clean and Sort

Run the following command (replace /your_dir for the base directory where you have your data).

```
docker run -it -v /your_dir:/ftmp dnalinux/funannotate:latest
/bin/bash -c "funannotate clean -i /ftmp/ID_polished.fasta -o
/ftmp/funannotate_prep/ID_polished_clean.fasta"

docker run -it -v /your_dir:/ftmp dnalinux/funannotate:latest
/bin/bash -c "funannotate sort -i
/ftmp/funannotate_prep/ID_polished_clean.fasta -o
/ftmp/funannotate_prep/ID_polished_clean_sort.fasta --minlen 1000"
```

RepeatMasker

- 12 Run the following command (replace `/your_dir` for the base directory where you have your data). Remember that this step requires the `dfam38_full.0.h5` database installed in `/your_dir`

```
docker run -it -v /your_dir:/ftmp dnalinux/repeatmasker:latest
/usr/local/RepeatMasker/RepeatMasker -s -species Fungi
/ftmp/funannotate_prep/ID_polished_clean_sort.fasta -xsmall
```

Fuannotate Predict

- 13 Run the following command (replace `/your_dir` for the base directory where you have your data). Replace CPU for your CPU count.

```
docker run -it -v /your_dir:/ftmp dnalinux/funannotate-gmes-
dikarya:latest /bin/bash -c "funannotate predict -i
/ftmp/funannotate_prep/ID_polished_clean_sort.fasta.masked -s ID -
o /ftmp/funannotate --cpus CPU"
```

Interproscan

- 14 Run the following command (replace `/your_dir` for the base directory where you have your data). Replace CPU for your CPU count.

```
docker run -it -v /your_dir:/ftmp -v
/home/sebastian/ips/interproscan-5.69-
101.0/data:/opt/interproscan/data -v /tmp:/temp
dnalinux/interproscan:5.69-101.0 --input
/ftmp/funannotate/predict_results/ID.proteins.fa --disable-precalc
--output-dir /ftmp/funannotate/ipsout --cpu CPU
```



Funannoate annotate

- 15 Run the following command (replace /your_dir for the base directory where you have your data)

```
docker run -it -v /your_dir:/ftmp dnalinux/funannotate-gmes-  
dikarya /bin/bash -c "funannotate annotate -i /ftmp/funannotate --  
fasta /ftmp/funannotate/predict_results/ID.proteins.fa --species  
ID --out /ftmp/FA_results --iprscan  
/ftmp/funannotate/ipsout/ID.proteins.fa.xml"
```