Predictive Analysis Code Base Breakdown

Running the graph-model:

REQUIRES BIFROST VERSION: 1.0.5

REQUIRES BLASTFROST (First Release)

Example Syntax:

nextflow run graph.nf --dname anthrax --customtargets true --customtargetspath ‘/home/USER/clustering/anthrax\_hclust.txt’ --bifrost ‘/home/user/bifrost/bifrost/build/src/Bifrost’

--blastfrost ‘/home/user/BlastFrost/build/BlastFrost’ -with-trace -bg

Workflow settings:

‘--genera’ : used to specify the genus (or multiple separated with ,) to be downloaded from NCBI when the ‘--download’ flag is set.

‘--download’ : set to true to download genomes for the specified genus from NCBI.

‘--customtargets’ : set to true to use custom labels rather than species labels from NCBI genomes (such as phylogenetic group)

‘--customtargetspath’ : path to file of custom labels to be used. File must be a text file with 1 label on each line. Number of labels in file must match number of samples in dataset.

‘--dname’ : name for the dataset you wish to run through the graph model. (Required)

‘--bifrost’ : path to Bifrost on your machine. (Required)

‘--blastfrost’ : path to BlastFrost on your machine. (Required)

Dataset information:

If you wish to use your own dataset not downloaded from the NCBI it must be in its own folder (named with the dname you desire) containing an individual folder for each sample. Each sample folder must contain a ‘.fasta’ of the genomic sequence data. If you want to include species information for the sample it should be in a file ending in ‘.meta’ with 3 lines. Line 1 contains the accession number/id you wish to give it, Line 2 contains the species, and line 3 contains the genus. Alternatively you could use ‘--customtargets’ and specify the target attribute one per line for each sample in the dataset (see workflow settings)

Workflow description:

DOWNLOAD: Step to download files from NCBI if selected.

METADATA: Step to index and catalogue files in dataset for use in future steps.

CUSTOMTARGETS: Step to set custom target labels for samples in dataset if selected.

PANGENOMES: Uses catalogued dataset to create pangenome graph of the samples, as generate and record splits used in cross-fold validation for model (These splits are also used in the kmer model so this workflow should be run first if intending to use the kmer model as well.

MAKEGRAPHS: Makes individual bifrost graphs for each sample in the dataset.

MAKEFASTA: Creates fasta files from these individual graphs which are used later on to make queries to the pangenome graph and develop the feature-graph used in model training/testing.

QUERY: Uses previously created fasta files with BlastFrost to query the pangenome graph for presence/absence of unitigs in dataset samples.

DATASET: Builds feature-graphs from queries obtained in previous step and saves it as a pytorch dataset. This is also where the code to make the graphs for the shuffled species experiment, although it is commented out in the live code base.

MODEL: This is where the actual model is run, first the dataset is loaded which is then used to train then test the graph model. This finishes by saving a .csv file of each accuracies for each fold named in the convention: (--dname)\_(foldnumber)graphsummary.csv

Running the kmer model:

REQUIRES jellyfish 2.3.0

Example Syntax:

nextflow run kmer.nf --dname coronavirus --download false --features true -with-trace -bg

Workflow settings:

‘--genera’ : used to specify the genus (or multiple separated with ,) to be downloaded from NCBI when the ‘--download’ flag is set.

‘--download’ : set to true to download genomes for the specified genus from NCBI.

‘--features’ : specifies whether you want to run kmer counting and build the matrix from those kmers. Typically you’ll want to set this to true unless you already have a kmer matrix build from a previous run/something else.

‘--customtargets’ : set to true to use custom labels rather than species labels from NCBI genomes (such as phylogenetic group)

‘--customtargetspath’ : path to file of custom labels to be used. File must be a text file with 1 label on each line. Number of labels in file must match number of samples in dataset.

‘--dname’ : name for the dataset you wish to run through the graph model. Only setting which is required to be set for this workflow.

Dataset information:

If you wish to use your own dataset not downloaded from the NCBI it must be in its own folder (named with the dname you desire) containing an individual folder for each sample. Each sample folder must contain a ‘.fasta’ of the genomic sequence data. If you want to include species information for the sample it should be in a file ending in ‘.meta’ with 3 lines. Line 1 contains the accession number/id you wish to give it, Line 2 contains the species, and line 3 contains the genus. Alternatively you could use ‘--customtargets’ and specify the target attribute one per line for each sample in the dataset (see workflow settings)

Workflow description:

DOWNLOAD: Step to download files from NCBI if selected.

METADATA: Step to index and catalogue files in dataset for use in future steps.

CUSTOMTARGETS: Step to set custom target labels for samples in dataset if selected.

FEATURES: Runs k-mer counting followed by building a matrix of k-mer counts for every sample in the dataset, followed by filtering out redundant k-mers to reduce matrix size on disk. This can be time consuming and memory intensive but is required to run the model.

TRAIN: Loads k-mer matrix into memory and runs the XGB model with it. This is completed by saving a ‘.csv’ file with the class accuracies of the dataset for each fold in the naming convention of: (--dname)\_(foldnumber)xgbsummary.csv

Running the filtermers workflow (counts kmers in a dataset and builds a matrix which then filters out redundant kmers:

Workflow settings:

‘--genera’ : used to specify the genus (or multiple separated with ,) to be downloaded from NCBI when the ‘--download’ flag is set.

‘--download’ : set to true to download genomes for the specified genus from NCBI.

‘--dname’ : name for the dataset you wish to run through the graph model. Only setting which is required to be set for this workflow.

‘--virus’ : true if you have a file of multiple viral genomes downloaded from the NCBI virus explorer.

‘--virusfile’ : path to file of viral genomes if specified yes to ‘--virus’

Dataset information:

If you wish to use your own dataset not downloaded from the NCBI it must be in its own folder (named with the dname you desire) containing an individual folder for each sample. Each sample folder must contain a ‘.fasta’ of the genomic sequence data. If you want to include species information for the sample it should be in a file ending in ‘.meta’ with 3 lines. Line 1 contains the accession number/id you wish to give it, Line 2 contains the species, and line 3 contains the genus. Alternatively you could use ‘--customtargets’ and specify the target attribute one per line for each sample in the dataset (see workflow settings)

Workflow description:

SEPARATE: Seperates file of multiple viral genomes into separate files to form a dataset.

DOWNLOAD: Step to download files from NCBI if selected.

INDEX: Step to index and catalogue files in dataset for use in future steps.

COUNTS: Perform k-mer counting on each sample in the dataset

MATRIX: Builds a matrix from k-mer counts and filters out redundant kmers.

Running the Unitig Matrix workflow (Builds bifrost graph and extracts unitigs from it to build a matrix of unitig presence/absence in your dataset):

Workflow settings:

‘--dname’ : name for the dataset you wish to run through the graph model. (Required)

‘--virus’ : true if you have a file of multiple viral genomes downloaded from the NCBI virus explorer.

‘--virusfile’ : path to file of viral genomes if specified yes to ‘--virus’

‘--bifrost’ : path to Bifrost on your machine. (Required)

‘--blastfrost’ : path to BlastFrost on your machine. (Required)

Dataset info:

Since no metadata is required/used the dataset must simply consist of a folder of name (--dname) containing individual folders with ‘.fasta’ files in them for each sample in the dataset.

Workflow description:

SEPARATE: Seperates file of multiple viral genomes into separate files to form a dataset.

GRAPH: Indexes the files in your dataset and builds a bifrost graph with them. After graph is built it used BlastFrost to enhance the graph with colour information so unitigs for each sample can be obtained easily.

UNITIGS: Extracts unitigs from the built pangenome graph and uses them to form a presence/absence matrix of unitigs in your dataset (saved in a .pkl file which can be loaded as you wish with pandas).

Running the reads\_pipeline workflow (processing, storing and further analysis of nanopore reads):

Workflow settings:

‘--move’ : default true, compresses then moves files from target machine (sequencer machine) to machine this is being run on for further processing/analysis.

‘--targetdir’ : the folder containing reads on the target machine that you wish to move/process.

‘--targetmachine’ : the ip address of the machine your desired reads are stored on (sequencer machine)

‘--flowcell’ : flowcell used for the sequencer. Only required if doing additional basecalling.

‘--kit’ : kit used in sequencer run. Only required if doing additional basecalling.

‘--basecalling’ : set to true if additional basecalling is desired. Default false.

Workflow Description:

MOVEFILES: Moves files from target machine to local machine and backs them up to the NAS.

GUPPY: Performs additional basecalling with guppy if (--basecalling) is set to true.

PYCOQC: Does quality and control and analysis with PycoQC.

NANOPLOT: Does data analysis/makes charts using nanoplot.

ASSEMBLE: Assembles reads using flye on the --meta option for metagenomics reads.

File Locations on Disk:

The four datasets for the recent experiments are located at ‘/home/liam/220802pan/pangraph/samples/’

Singularity image files for the graph/kmer model (cuda.sif and pipeline.sif respectively) are located in ‘/home/liam/220802pan/pangraph/’ as well as on the NAS under ‘singularity\_images/’

Unitig matrices for each dataset are located in ‘/home/liam/Unitigs/processed\_data/’ labeled as name\_unitigs.pkl

The script I used to do hierarchal/k-means clustering is located in ‘pangraph/pyfiles/clustering.py’ in the repo.