16S processing on AWS

Thomas Gurry

1 Introduction

This document describes the process of going from raw 16S data on in an S3 bucket to processed data. Each dataset must be accompanied by a machine-readable text file, called a summary file, described below.

2 Preliminary notes - read this first!!

2.1 Raw FASTQ files

The 16S pipeline is designed to take individual FASTQ files. If you have many files for a single dataset, merge them into a single file. The pipeline will split them for parallelization and recombine them for OTU calling.

FASTQ files have different ASCII encodings for the quality characters. It is often ASCII base 33 or ASCII base 64. Check this using `usearch -fastq_chars yourFASTQfile.fastq` and specify the encoding in the summary file. If left unspecified, it will default to base 64, which could cause problems.

2.2 Summary files

This file, named `summary_file.txt`, is a machine-readable, **tab-delimited** file that must accompany any dataset directory when uploaded to the cloud. It should be found in the highest directory level for the dataset directory in the S3 bucket. It is a text file with descriptors for the data and paths to all relevant datafiles within the directory. It can include a True or False flag for whether any associated raw 16S data has already been processed.

The order in which items are listed between the lines `#16S_start` and `#16S_end` appear does not matter. For an exhaustive list of summary file attributes, see the table below.

2.3 Metadata

Metadata is uploaded as is to S3 for storage with the raw data. Ideally, we want to capture as much of the metadata as possible in the database and into file objects describing the data

(e.g. OTU tables in BIOM format). We can produce a file with three columns: `sample_ID`, `disease_label` and `keywords`. `sample_ID` corresponds to the name ID of the sample as is listed in the OTU table. `disease_label` is a specific label used for each subject and can be more general than disease, but is used by the analytics layer for building classifiers and such. There is a list on the Personal Analytics AWS Google drive describing these labels. `keywords` can be a list of any keywords you think should be associated with the sample. These are listed separated by commas and no spaces (e.g. `keyword1,keyword2,keyword3`). These will be searchable from the database interface so it's worth giving them some thought!

2.4 Dataset master list

Once a dataset is uploaded, please add an entry on the master list on the PA google drive.

3 Examples

3.1 Case 1: raw FASTQ file, still includes primers and barcodes

The simplest case is if you have the following files: a raw FASTQ file; a file specifying the map between barcode sequences and IDs; and a file specifying the primers used. Your summary file would look something like this:

```
DATASET_ID myDataset

#16S_start
RAW_FASTQ_FILE myData.fastq
ASCII_ENCODING ASCII_BASE_33
PRIMERS_FILE primers.txt
BARCODES_MAP barcodes_map.txt
BARCODES_MODE 2
METADATA_FILE metadata.txt
PROCESSED False
#16S_end
```

Note that you must also specify the place where barcodes are to be found, i.e. either in the "¿" sequence ID lines (mode 1) or in the sequences themselves (mode 2). The `PROCESSED` flag tells the processing instance that the dataset needs to be processed into OTU tables.

3.2 Case 2: raw FASTQ file, primers and barcodes have been removed

In the case where the 'raw' data has actually had primers and barcodes previously removed, the sample IDs must be listed in the sequence ID lines of the FASTQ file. When the pipeline removes barcodes itself and replaces them with sample IDs, individual sequence reads for

a given sampleID will be annotated as sampleID;1, sampleID;2, etc., where we note here that the BARCODES_SEPARATOR is ';'. However, in a dataset where the barcodes have previously been removed, you will have to look into the FASTQ file to check the 'separator' character. Your summary file would look something like this:

```
DATASET_ID myDataset

#16S_start
RAW_FASTQ_FILE myData.fastq
ASCII_ENCODING ASCII_BASE_33
PRIMERS_FILE None
BARCODES_MAP None
BARCODES_SEPARATOR ;
METADATA_FILE metadata.txt
PROCESSED False
#16S_end
```

3.3 Case 3: no raw data, only OTU table

If you wanted to upload an OTU table without any associated raw data, your summary file would look something like this:

```
DATASET_ID myDataset

#16S_start
OTU_TABLE otu_table.txt
OTU_SEQUENCES_FASTA otu_sequences.fasta
METADATA_FILE metadata.txt
PROCESSED True
#16S_end
```

4 Python modules

4.1 Modules

- `Formatting.py` - Module to house miscellaneous formatting methods, e.g. conversion from classic dense format to BIOM format, OTU table transposition, etc.
- `DepositDB.py` - Methods for depositing data into the database.
- `preprocessing_16S.py` - Methods and wrappers for raw 16S sequence data processing.

4.2 Scripts

- `Master.py` - Master script that calls relevant processing pipelines, e.g. `raw2otu.py`.
- `raw2otu.py` - Pipeline for converting raw 16S FASTQ sequence files to OTU tables. Handles parallelization requirements in these processing steps automatically. Takes as input a directory that contains a summary file and the raw data.