PeerJ Preprints NOT PEER-REVIEWED

MerCat: a versatile k-mer counter and diversity estimator for database-independent property analysis obtained from metagenomic and/or metatranscriptomic sequencing data

Richard Allen White III[1*], Ajay Panyala[2], Kevin Glass[3], Sean Colby[1], Kurt R Glaesemann[2], Christer Jansson[3], Janet K Jansson[1*]

[1]Biological Sciences Division, Pacific Northwest National Laboratory, Richland, Washington 99352, USA

[2]Infomation technology, High Performance Computing (HPC) and Cloud Services, Pacific Northwest National Laboratory, Richland, Washington 99352, USA

[3]Environmental and Molecular Sciences Laboratory, Pacific Northwest National Laboratory, Richland, WA 99352, USA

*To whom correspondence should be addressed

PeerJ Preprints | https://doi.org/10.7287/peerj.preprints.2825v1 | CC BY 4.0 Open Access | rec: 21 Feb 2017, publ: 21 Feb 2017

PeerJ Preprints NOT PEER-REVIEWED

Abstract

Summary: MerCat ("Mer - Catenate") is a parallel, highly scalable and modular property software package for robust analysis of features in next-generation sequencing data. Using assembled contigs and raw sequence reads from any platform as input, MerCat performs k-mer counting of any length k, resulting in feature abundance counts tables. MerCat allows for direct analysis of data properties without reference sequence database dependency commonly used by search tools such as BLAST for compositional analysis of whole community shotgun sequencing (e.g., metagenomes and metatranscriptomes).

Availability and implementation: MerCat is written in Python and distributed under a BSD license. The source code of MerCat is freely available at https://github.com/pnnl/mercat

MerCat is compatible with Python 2 and 3 and works on both Mac OS X and Linux. MerCat can also be easily installed using bioconda: conda install mercat

Contact: Richard Allen White III and Janet Jansson, Biological Sciences Division, Pacific Northwest National Laboratory, Richland, Washington 99352, USA. Email: Richard.white@pnnl.gov or raw937@gmail.com, Janet.jansson@pnnl.gov

Keywords:
K-mer counting
Database-independent property analysis (DIPA)
Metagenomic analysis
Metatranscriptomic analysis
Diversity-estimation

PeerJ Preprints | https://doi.org/10.7287/peerj.preprints.2825v1 | CC BY 4.0 Open Access | rec: 21 Feb 2017, publ: 21 Feb 2017

1 Introduction

Whole community sequencing of total DNA (i.e., metagenomics) and total RNA (i.e., metatranscriptomics) have provided windows into the composition, functions and potential roles of microbial communities residing in complex ecosystems (e.g., soil) (White III *et al.,* 2016a). The throughput of next generation sequencing (NGS) technologies is continuously increasing: sequence data currently requires terabytes of storage (White III *et al.,* 2016a) and read lengths can exceed 90 kbp (Laver *et al.,* 2015). Therefore, developments of robust bioinformatics tools are needed to analyze these data.

Reference sequence databases and tools that classify sequences are critical bottlenecks in metagenomics and metatranscriptomics. For example, tools that search reference sequence databases against query data such as homology-based BLAST are computationally slow against large databases (e.g., KEGG) (Silva *et al.,* 2016). In addition, metagenome assembly approaches, although recently improved for complex data types (Howe *et al.,* 2014; Li *et al.,* 2015; White III *et al.,* 2016b), are not able to assemble all data. Open-source reference sequence databases are facing a number of challenges, including finding lasting funding, many are moving to a subscription-based access (e.g., KEGG www.kegg.jp/kegg/), slowed development (e.g., COG https://www.ncbi.nlm.nih.gov/COG/), or discontinuation (e.g., CAMERA http://camera.calit2.net/).

Database-independent property analysis (i.e., DIPA) which utilizes counting of k-mers subsequences (of length k) from sequence reads obtained from NGS platforms without a reference sequence database for matching query data. DIPA-based k-mer counting provides rapid and robust microbial community analysis and characterization without the

PeerJ Preprints | https://doi.org/10.7287/peerj.preprints.2825v1 | CC BY 4.0 Open Access | rec: 21 Feb 2017, publ: 21 Feb 2017

biases or limitations of sequence databases (Jiang *et al.,* 2012) and/or *de novo* assembly in order to compare and contrast sequence datasets. K-mers are critical to assembly (Li *et al.,* 2015), counting (Zhang *et al.,* 2014), partitioning (Howe *et al.,* 2014), genomic binning (Wu *et al.,* 2015) and classification (Jiang *et al.,* 2012). K-mer based counting is amongst the fastest approaches for profiling metagenomic and/or metatranscriptomic data (Lindgreen *et al.,* 2015).

There are many k-mer counters (Zhang *et al.,* 2014), and even database dependent k-mer profilers (Koslicki and Falush, 2016). MerCat provides only k-mer counting tool for assembled contigs (.fna), translated protein-coding ORFs (.faa) and NGS reads (.fastq) for any size k-mer. Alpha diversity metrics for microbial ecology including chao1, ace, simpson, goods coverage, dominance and fishers alpha are generated by MerCat. Nucleotide properties (e.g., %G+C, %A+T) and protein properties of translated protein-coding open reading frames (ORFs) (e.g., protein isoelectric point, pI, and hydrophobicity metrics) are also generated.

Here we describe MerCat, a tool that can accommodate any size sequence file by utilizing a 'divide and conquer' approach and then performs k-mer analysis. MerCat can be employed for rapid, robust, versatile analysis of NGS microbial community data using DIPA.

2 DESCRIPTION OF THE TOOL

MerCat is a modular and highly-scalable Python-based open-source software package. MerCat computes k-mer frequency counting to any length k on assembled contigs as nucleotide fasta, raw reads (e.g., fastq), and translated protein-coding ORFs (e.g., protein

PeerJ Preprints | https://doi.org/10.7287/peerj.preprints.2825v1 | CC BY 4.0 Open Access | rec: 21 Feb 2017, publ: 21 Feb 2017

fasta). The package also allows for user-defined custom analyses. Although raw read inputs can be used in MerCat, it is not recommended due to low quality and sequencing errors, thus we utilize Trimmomatic (Bolger *et al.,* 2014) for quality control trimming of low-quality data obtained by fastq formats (default trimming is base pair quality score >Q_{30}). K-mer counting in MerCat has two modes: DNA mode, which can analyze nucleotide contigs directly, and Protein mode, wherein nucleotide contigs are translated into protein-coding ORFs with Prodigal (Hyatt *et al.,* 2012), using the metagenomic option (default) (Figure 1). Individual sequence files or many files within a single folder can be analyzed by MerCat. Tabular outputs include overall feature files (e.g., many files within a folder), or per-file feature analysis based on k-mer frequency counts tables for either nucleotides and/or proteins fasta files. Tabular file outputs are stored as comma-separated files for downstream analysis. MerCat can also calculate Alpha diversity metrics for each file in both Protein and DNA mode. As a default, we provide k-mer frequency stacked bar plots for individual samples and MDS plots for many samples.

MerCat can handle input files >10 gigabyte by splitting them into multiple files. MerCat computes on the individual files, then combines the resulting data, analyzes data and produces the final output (as mentioned previously) for large input files. The combined overall output generated may be too large to fit in the available memory of a standard computer. For this reason, we used Dask, a Python-based parallel-computing library that enables processing data that does not fit into available memory. Dask stores the data on a hard disk, then loads portions of it back and forth into memory as needed for analysis. This enables MerCat to scale from laptop to high-performance computing resources, all within the same

PeerJ Preprints | https://doi.org/10.7287/peerj.preprints.2825v1 | CC BY 4.0 Open Access | rec: 21 Feb 2017, publ: 21 Feb 2017

user friendly-package.

3 SUMMARY

MerCat provides DIPA for metagenomic and metatranscriptomic data, starting from nucleotide and protein sequence files and ending with tabular files for downstream analysis and visualization. MerCat is scalable, accommodating for large input files, is user-friendly, easy to install and is user customizable. MerCat fills a major computational bottleneck by enabling rapid analysis of many datasets and large datasets in a database independent manner.

Acknowledgments

We thank Nathan Johnson for his assistance in preparing excellent figures.

Funding

This research was provided by PNNL Laboratory-Directed Research and Development (LDRD) Program at PNNL; Microbiomes in Transition (MinT) Initiative, and PNNL Initiative integrated Plant-Atmosphere-Soil System (iPASS; PNNL Project # 204412), a multiprogram national laboratory operated by Battelle for the Department of Energy (DOE) under contract DE-AC06-76RL01830. A portion of the research was conducted using PNNL Institutional Computing (PIC) at PNNL and at EMSL, a national scientific user facility sponsored by the DOE Office of Biological and Environmental Research and located at PNNL.

PeerJ Preprints | https://doi.org/10.7287/peerj.preprints.2825v1 | CC BY 4.0 Open Access | rec: 21 Feb 2017, publ: 21 Feb 2017

References

Bolger, A.M. *et al.* (2014) Trimmomatic: a flexible trimmer for Illumina sequence data. *Bioinformatics,* **30**, 2114-2120.

Hyatt, D. *et al*. (2012) Gene and translation initiation site prediction in metagenomic sequences. *Bioinformatics,* **28**, 2223-2230.

Howe, A.C. *et al.* (2014) Tackling soil diversity with the assembly of large, complex metagenomes. *Proc Natl Acad Sci USA.,* **111**, 4904-4909.

Jiang, B. *et al.* (2012) Comparison of metagenomic samples using sequence signatures. *BMC Genomics,* **13**, 730. doi: 10.1186/1471-2164-13-730.

Koslicki, D. and Falush, D. (2016) MetaPalette: a k-mer Painting Approach for Metagenomic Taxonomic Profiling and Quantification of Novel Strain Variation. *Msystems*, **3**, pii: e00020-16.

Laver, T. *et al.* (2015) Assessing the performance of the Oxford Nanopore Technologies MinION. *Biomol. Detect Quantif.,* **3**, 1-8.

Lindgreen, S. *et al.* (2016) An evaluation of the accuracy and speed of metagenome analysis tools. *Sci. Rep.,* **6**, 19233.

Li, D. *et al.* (2015) MEGAHIT: an ultra-fast single-node solution for large and complex metagenomics assembly via succinct de Bruijn graph. *BioInformatics* **31**, 1674-1676.

Silva, G.G. *et al.* (2016) SUPER-FOCUS: a tool for agile functional analysis of shotgun metagenomic data. *Bioinformatics.* **32**, 354-361.

White, R.A. III, *et al.* (2016) (a) The past, present and future of microbiome analyses. *Nature Protocols* **11**, 2049-2053.

White, R.A. III, *et al.* (2016) (b) Moleculo long-read sequencing facilitates assembly and genomic binning from complex soil metagenomes. *Msystems*, **3**, e00045-16. doi:10.1128/mSystems.00045-16.

Wu, Y.W. *et al.* (2016) MaxBin 2.0: an automated binning algorithm to recover genomes from multiple metagenomic datasets. *Bioinformatics* **32**, 605-607.

Zhang, Q. *et al.* (2014) These Are Not the K-mers You Are Looking For: Efficient Online K-mer Counting Using a Probabilistic Data Structure. *PLoS ONE* **9**, e101271.

PeerJ Preprints | https://doi.org/10.7287/peerj.preprints.2825v1 | CC BY 4.0 Open Access | rec: 21 Feb 2017, publ: 21 Feb 2017

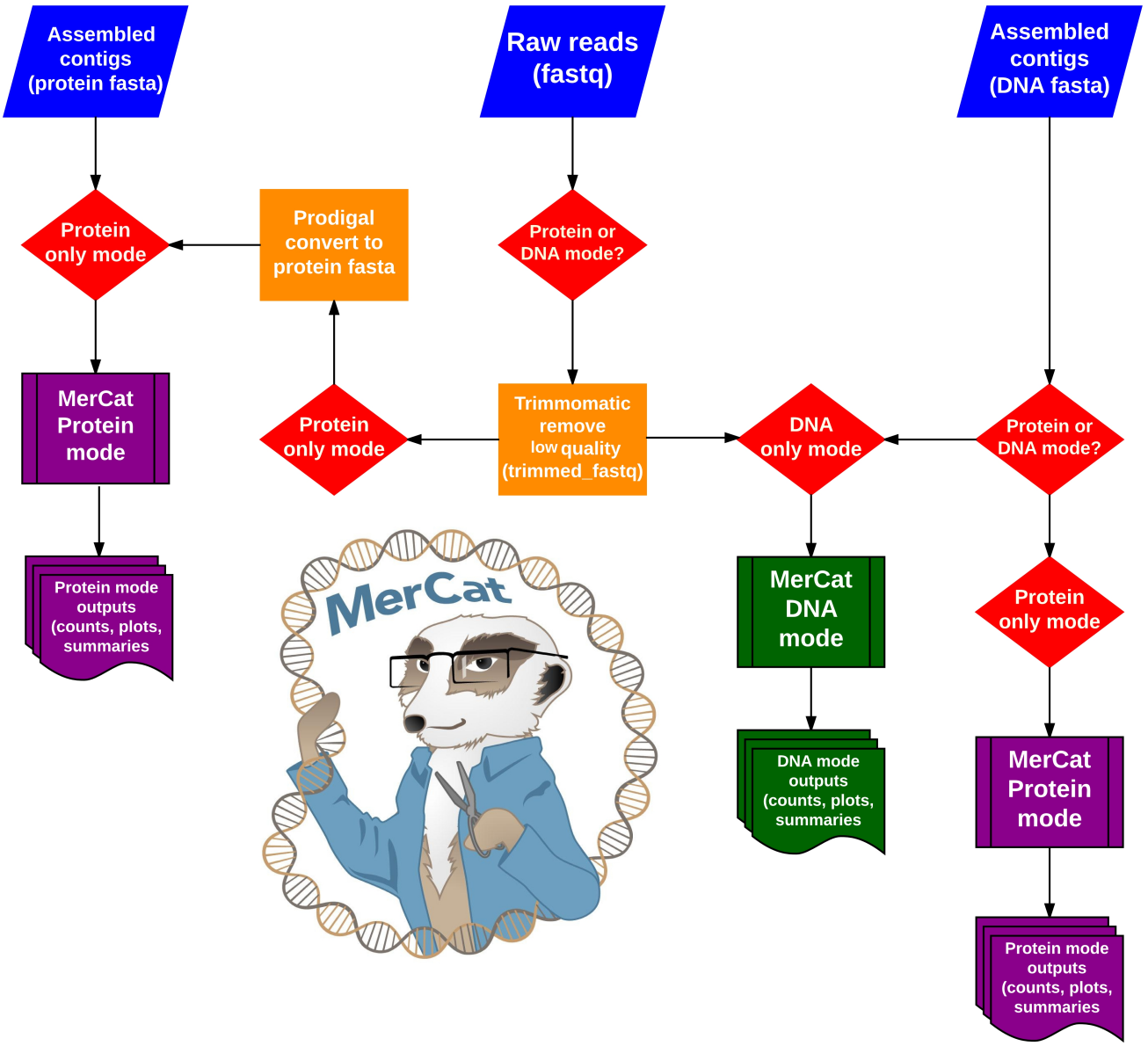

Figure 1: MerCat workflows. Inputs include assembled contigs (.fna), assembled contigs previously translated protein-coding ORFs (.faa) and NGS reads (.fastq) for any size k-mer. Outputs include tabular count tables for individual mers, stacked bar and MDS plots, alpha diversity statistics. Prodigal uses metagenomic mode as default for translating assembled nucleotide contigs into protein-coding ORFs (.faa). Trimmomatic as default requires base quality >Q_{30}.

PeerJ Preprints | https://doi.org/10.7287/peerj.preprints.2825v1 | CC BY 4.0 Open Access | rec: 21 Feb 2017, publ: 21 Feb 2017