BIOINFORMATICS APPLICATIONS NOTE
Vol. 28 no. 16 2012, pages 2182–2183
doi:10.1093/bioinformatics/bts355

Sequence analysis
Advance Access publication June 23, 2012

DySC: software for greedy clustering of 16S rRNA reads

Zejun Zheng, Stefan Kramer and Bertil Schmidt*
Institut für Informatik, Johannes Gutenberg University Mainz, Mainz 55099, Germany

Associate Editor: Michael Brudno

ABSTRACT

Summary: Pyrosequencing technologies are frequently used for sequencing the 16S ribosomal RNA marker gene for profiling microbial communities. Clustering of the produced reads is an important but time-consuming task. We present Dynamic Seed-based Clustering (DySC), a new tool based on the greedy clustering approach that uses a dynamic seeding strategy. Evaluations based on the normalized mutual information (NMI) criterion show that DySC produces higher quality clusters than UCLUST and CD-HIT at a comparable runtime.

Availability and implementation: DySC, implemented in C, is available at http://code.google.com/p/dysc/ under GNU GPL license.

Contact: bertil.schmidt@uni-mainz.de

Supplementary Information: Supplementary data are available at *Bioinformatics* online.

Received on January 3, 2012; revised on June 1, 2012; accepted on June 16, 2012

1 INTRODUCTION

Pyrosequencing technologies are frequently used for microbial sequencing studies based on sequencing of hyper-variable regions of the 16S ribosomal RNA (rRNA) marker gene (e.g. Turnbaugh *et al.*, 2009) Typical dataset sizes range from a few tens of thousands up to several million reads. The taxonomy-independent approach to their computational analysis typically performs a hierarchical clustering and then bins the reads into Operational Taxonomic Units (OTUs) based on a distance threshold (Huse *et al.*, 2010). Clustering is usually computed on a pairwise distance matrix that is derived either from an all-against-all read alignment (Sun *et al.*, 2009) or a (profile-based) multiple sequence alignment (Nawrocki *et al.*, 2009; Schloss *et al.*, 2009). However, this approach has a high complexity in terms of both time and space and thus does not scale up to several million reads.

The classical greedy sequence clustering method (Holm and Sander, 1998; Li *et al.*, 2001) builds up clusters in an iterative incremental fashion. Each cluster is represented by one sequence (called seed). An unseen read is then aligned to all seeds. If all alignments return distances above a given threshold, the read is used as the seed of a new cluster. Otherwise, the read is added to one of the clusters for which the distance is within the threshold. Examples of existing software used for read clustering based on the greedy approach are CD-HIT (Li and Godzik, 2006) and UCLUST (Edgar, 2010) Although this approach is scalable and efficient it generally produces clusters of lower quality than hierarchical clustering (Sun *et al.*, 2011). In this article, we present a new tool called Dynamic Seed-based Clustering (DySC) that uses a dynamic seeding strategy to achieve higher accuracy while preserving scalability and efficiency.

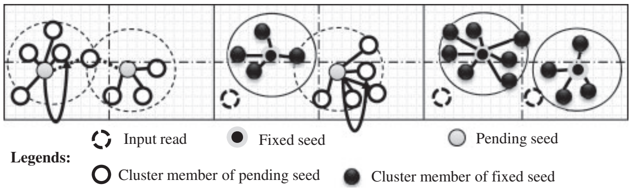

Fig. 1. Pending cluster conversion example using two neighboring clusters reaching the threshold size *max_pending* = 5 (left). Fixed seeds and the fixed cluster members are determined (center and right). After optimization, the seed of the cluster can be shifted to another member of the original pending cluster (see arrows)

2 METHODS AND IMPLEMENTATION

DySC uses the following concepts in order to reduce the amount of inaccurately formed clusters in the early stages of greedy clustering.

2.1 Fixed cluster

A fixed cluster consists of a set of assigned reads and is represented by one of the assigned reads, called 'fixed seed'. Read membership to a fixed cluster is exclusive and final. A fixed cluster can still grow in size, but the seed is fixed.

2.2 Pending cluster

A pending cluster consists of a set of assigned reads and is represented by one of the assigned reads, called 'pending seed'. Pending clusters are used at the early stages of cluster construction. Read membership to a pending cluster is neither exclusive nor final. Once a pending cluster reaches a threshold size (*max_pending* reads), it is converted into a fixed cluster.

2.3 Pending cluster to fixed cluster conversion

First, a fixed seed is determined as the assigned read that maximizes a *k*-mer spectrum-based inner cluster link score, which is defined as the sum of the *k*-mers shared between the selected read and all other assigned reads. Second, all pending reads are re-aligned to the new fixed seed in order to decide whether they are members of the new fixed cluster. If a read is assigned to the new fixed cluster, its membership to any pending clusters will be removed. Figure 1 illustrates the conversion process.

2.4 Read assignment

Consider a set of fixed clusters (each represented by a fixed seed) and a set of pending clusters (each represented by a pending seed). The DySC clustering workflow iteratively loads a batch of reads from the input file and tries to assign them to the fixed clusters first. The assignment procedure calculates pairwise *k*-mer distances (see Supplementary 'Read assignment') between the batch of input reads and the set of fixed seeds. For each identified read-seed pair with a *k*-mer distance below a given threshold, a dissimilarity

*To whom correspondence should be addressed.

value (defined by one minus the fractional sequence identity derived from an optimal global alignment) is calculated in quadratic time. If the dissimilarity is below a given threshold, the read is assigned to the corresponding fixed cluster.

To identify promising read-seed pairs quickly and with high sensitivity, two *k*-mer indices are used as follows. First, the *k*-mer index of the batch of input reads is built. This index is queried by each fixed seed. The returned list containing sorted seed-read pairs ordered by *k*-mer distance is searched from top to bottom to identify homologues by computing the corresponding dissimilarities. Each read assigned to a fixed cluster is removed from the *k*-mer index. If a certain number of dissimilarity computations (*max_low*) is above another given threshold (*thresh_low*), the search terminates in order to limit the time-consuming global alignment computations. Second, each remaining read is used as a query to the fixed seed *k*-mer index. The returned list of sorted promising read-seed pairs is then processed in a similar way. Subsequently, each input read that has not been assigned to any fixed cluster is compared with all pending seeds using *k*-mer distance and dissimilarity computation. A read is assigned to all pending clusters for which the dissimilarity is below a given threshold. Each remaining read that has not been assigned to any pending cluster is considered as a new pending seed. If a pending cluster reaches the size of *max_pending* reads, the conversion procedure described above is triggered. The supplement provides more details about our clustering procedure.

3 PERFORMANCE EVALUATION

We have compared DySC with CD-HIT-OTU (v0.0.2, with CD-HIT-EST as clustering engine), UCLUST (USEARCH v5.0.151) and the hierarchical clustering-based tool ESPRIT-Tree (Cai and Sun, 2011) in terms of accuracy and runtime. ULCUST is executed within the otupipe script (http://drive5.com/otupipe). All four tools use default parameters. DySC has the following default parameters: $k = 10$, $max\_pending = 10$, $max\_low = 10$ and $thresh\_low = 1\text{-}(SIM\text{-}0.04)$, where *SIM* is the given clustering similarity (e.g. $SIM = 0.97$). For the accuracy comparison we have downloaded the well-known human gut microbiome dataset (Turnbaugh *et al.* 2009) generated by pyrosequencing of the 16S rRNA gene V6 region (SRX001445, 464K reads of average length 121 bp) and V2 region (SRX001447, 535K reads of average length 265 bp) from the NCBI Sequence Read Archive (SRA) The normalized mutual information (NMI) score (Sun *et al.*, 2011) with ground-truth clusters established by Megablast comparison to the RDP Database R10.26 (Cole *et al.*, 2009) is used to determine clustering accuracy. Unaligned reads (similarity <97% or blast region <120 bp) and chimeric reads (using UCHIME v4.2; Edgar *et al.*, 2011) were removed from the datasets. We then randomly sampled 30K reads from the remaining V2 (V6) dataset. Each tested tool then performs a clustering with similarity threshold settings ranging from 90 to 99% with a 1% step-size. Each experiment is repeated 10 times with different read sampling. Table 1 shows the average NMI scores as well as the average number of identified OTUs at 97% and 95% similarity levels, which are often used to reflect species- and genus-level community structure profiles. Furthermore, the NMI scores for all tests are shown in Figure S1.

We have further compared the clustering runtime of the four tools for performing a four-level clustering at 97, 95, 93 and 91% similarity levels on a workstation with a Xeon E5504 processor and 16 GB RAM using a single thread. In addition to the gut microbiome V6 and V2 datasets, we have also tested a bigger artificial bacterial community dataset (SRR069029, NCBI SRA) generated by sequencing of a defined mixture of genomic DNA containing 2.7 M reads of average length 125 bp. For CD-HIT-OTU and otupipe, only the runtime of the respective clustering engine (CD-HIT-EST and UCLUST) is considered. It took ESPRIT-Tree 23 min and each of other three tools took less than 5 min to finish the four-level clustering of the complete V6 dataset. For the complete V2 dataset (SRR069029 dataset), UCLUST finishes the task within 17 min (182 min), DySC within 67 min (129 min), CD-HIT-EST within 296 min (223 min) and ESPRIT-Tree within 1324 min (1571 min). An empirical evaluation of DySC using the SRR069029 dataset indicates a quasi-linear time complexity (Supplementary Fig. S2) and a memory consumption of less than 1 GB for a dataset with a few million input reads (Supplementary Fig. S3).

Table 1. Clustering accuracy evaluation

Tool	SIM	NMI-score		No. OTU-genus/species	
		V2	V6	V2	V6
otupipe	97%	0.73	0.64	8457/183	2422/114
CD-HIT-OTU		0.68	0.64	6801/369	2222/190
ESPRIT-Tree		0.78	0.68	5633/350	2092/184
DySC		0.78	0.78	2976/345	745/184
otupipe	95%	0.61	0.64	3872/347	699/184
CD-HIT-OTU		0.61	0.67	2899/170	628/110
ESPRIT-Tree		0.69	0.74	2548/331	523/105
DySC		0.70	0.82	886/329	215/177

In conclusion, DySC is a scalable tool for clustering of 16S rRNA reads for microbial profiling generated by high-throughput sequencing machines, which can cluster millions of reads on a standard workstation within a few hours. Benchmarking results show that extending the classical greedy algorithm by delaying the production of fixed clusters through the usage of pending clusters can improve accuracy significantly while keeping efficiency high.

REFERENCES

Cai,Y. and Sun,Y. (2011) ESPRIT-Tree: hierarchical clustering analysis of millions of 16S rRNA pyrosequences in quasilinear computational time. *Nucleic Acid Res.*, **39**, e95.

Cole,J.R. *et al.* (2009) The Ribosomal Database Project: improved alignments and new tools for rRNA analysis. *Nucleic Acid Res.*, **37**, D141–D145.

Edgar,R.C. (2010) Search and clustering orders of magnitude faster than BLAST. *Bioinformatics*, **26**, 2460–2461.

Edgar,R.C. *et al.* (2011) UCHIME improves sensitivity and speed of chimera detection. *Bioinformatics*, **27**, 2194–2200.

Holm,L. and Sander,C. (1998) Removing near-neighbour redundancy from large protein sequence collections. *Bioinformatics*, **14**, 423–429.

Huse,S.M. *et al.* (2010) Ironing out the wrinkles in the rare biosphere through improved OTU clustering. *Environ. Microbiol.*, **12**, 1889–1898.

Li,W. and Godzik,A. (2006) Cd-hit: a fast program for clustering and comparing large sets of protein or nucleotide sequences. *Bioinformatics*, **22**, 1658–1659.

Li,W. *et al.* (2001) Clustering of highly homologous sequences to reduce the size of large protein database. *Bioinformatics*, **17**, 282–283.

Nawrocki,E.P. *et al.* (2009) Infernal 1.0: inference of RNA alignments. *Bioinformatics*, **25**, 1335–1337.

Schloss,P.D. (2009) A high-throughput DNA sequence aligner for microbial ecology studies. *PLoS One*, **4**, e8230.

Sun,Y. *et al.* (2009) ESPRIT: estimating species richness using large collections of 16S rDNA pyrosequences. *Nucleic Acids Res.*, **37**, e76.

Sun,Y. *et al.* (2011) A large-scale benchmark study of existing algorithms for taxonomy-independent microbial community analysis. *Brief Bioinform.*, **13**, 107–121.

Turnbaugh,P.J. *et al.* (2009) A core gut microbiome in obese and lean twins. *Nature*, **457**, 480–484.