rRNAFilter: A Fast Approach for Ribosomal RNA Read Removal Without a Reference Database

YING WANG,¹ HAIYAN HU,¹ and XIAOMAN LI²

ABSTRACT

Metatranscriptomics studies the transcriptome of all microbial species in a habitat. Removing ribosomal RNA (rRNA) reads in metatranscriptomic data is essential for the study of microbial gene expression. Although several methods are developed, all of them rely on rRNA databases that contain a limited number of known rRNA sequences and cannot work well on rRNA reads from unknown rRNA sequences. To address this problem, we have developed a novel approach called rRNAFilter. Our method can accurately and rapidly remove rRNA reads from metatranscriptomes without any prior knowledge of known rRNA sequences. Compared with two existing approaches, rRNAFilter has shown comparable performance when working on reads from known rRNA sequences and much better performance when dealing with reads from unknown rRNA sequences.

Keywords: metatranscriptomics, rRNA, rRNA read removal.

1. INTRODUCTION

METATRANSCRIPTOMICS IS WIDELY APPLIED to study microbes from different habitats such as soil and human guts (Bailly et al., 2007; Gosalbes et al., 2011). One of the most challenging tasks in metatranscriptomics is to separate ribosomal RNA (rRNA) reads from non-rRNA reads such as messenger RNA (mRNA) reads. This is because more than 90% of reads may be from rRNAs in a metatranscriptomic dataset (Stewart et al., 2010), which prevents the identification of mRNAs and their protein products. For instance, without removing rRNA, reads can generate 90% of incorrect protein annotations that result in inaccurate downstream analyses (Tripp et al., 2011).

Different experimental and computational approaches have been developed to remove rRNA reads. Several rRNA depletion or mRNA amplification experimental protocols can help to remove rRNA reads (He et al., 2010; Gilbert and Hughes, 2011). However, the metatranscriptomic datasets still contain a large portion of reads from rRNAs after applying these protocols. A few computational methods have also been developed to remove rRNA reads in metatranscriptomes (Kopylova et al., 2012; Schmieder et al., 2012). These methods usually filter rRNA reads by comparing reads with known rRNA sequences in public databases or models derived from known rRNA sequences. Although these methods have shown high

¹Department of Computer Science, University of Central Florida, Orlando, Florida.

²Burnett school of Biomedical Science, College of Medicine, University of Central Florida, Orlando, Florida.

accuracy in previous studies (Kopylova et al., 2012; Schmieder et al., 2012), their accuracy depends on the percentage of reads from known rRNA sequences in a metatranscriptomic dataset. Moreover, as shown in this study, these methods cannot work well on rRNA reads from unknown rRNA sequences.

In this study, we developed a novel rRNA filtering tool, rRNAFilter. Our tool does not require any prior knowledge of known rRNA sequences. It filters rRNA reads based on the difference of the frequency of k-mers (k base pairs long DNA segments) in input reads. The rationale is that rRNA reads are much more abundant than non-rRNA reads and k-mers in input reads should be able to distinguish the two types of reads. Compared with two popular methods for removing rRNA reads, rRNAFilter had a much faster speed and at least a comparable accuracy (see Reference 1 for url).

2. MATERIALS AND METHODS

2.1. Overview of the major steps in rRNAFilter

RRNAFilter applies an expectation-maximization (EM) algorithm (Li and Waterman, 2003; Wang et al., 2015) to k-mers in input reads and separates k-mers of different abundance into different groups (Fig. 1). It then assigns reads comprising k-mers from high-abundant k-mer groups as rRNA reads and reads comprising k-mers from low-abundant k-mer groups as non-rRNA reads. The majority of reads are assigned

FIG. 1. The procedure to assign reads in metatranscriptomic datasets by rRNAFilter. The rRNA reads are represented by solid lines, and the non-rRNA reads are represented by dotted lines. rRNA, ribosomal RNA.

with high confidence at this step. Next, Markov models are trained with the assigned rRNA reads and non-rRNA reads to represent the common characteristics of the rRNA reads and the non-rRNA reads, respectively. Finally, the unassigned reads are compared with the two trained Markov models to assign the remaining reads.

There are two different modes in rRNAFilter. In the first mode, rRNAFilter_m1, it assigns reads by the above procedure without the last two steps. These assigned reads are highly likely rRNA reads or non-rRNA reads. This mode may leave a fraction of reads unassigned. In the second (default) mode, rRNAFilter_m2, rRNAFilter assigns all input reads as either rRNA reads or non-rRNA reads by the above procedure, including the last two steps. The details are in the following.

2.2. EM algorithm

RRNAFilter applies an EM algorithm to separate rRNA reads from non-rRNA reads. The EM algorithm assumes that the frequency of k-mers in input reads, $X = \{x_1, x_2, \dots, x_n\}$, follows a mixture of m Poisson distributions with the unknown Poisson parameters, $\lambda_1, \lambda_2, \dots, \lambda_m$. For any i from 1 to n and any j from 1 to m , if x_i is from the j -th Poisson distribution, p_j , then

$$P(x_i = x) = \alpha_j p_j(\lambda_j, x) = \alpha_j \frac{\lambda_j^x}{x!} e^{-\lambda_j}$$

where α_j is the unknown probability that a random k-mer is from the j -th distribution and $\sum_{j=1}^m \alpha_j = 1$. We define the missing binary variables, Z_{ij} , where $Z_{ij} = 1$ indicates that x_i is from the j -th Poisson distribution and $Z_{ij} = 0$ otherwise.

With the above notations, the log complete likelihood function of the observed data X and the missing data $Z = \{Z_{ij}\}$ is $\log(L(\theta; X, Z)) = \sum_{i=1}^n \sum_{j=1}^m Z_{ij} \log(\alpha_j * p_j(\lambda_j, x_i))$. The E-step of the EM algorithm is to estimate Z_{ij} as $Z_{ij} = P(y_i = j | X, \theta) = \frac{\alpha_j * p_j(\lambda_j, x_i)}{\sum_{r=1}^m \alpha_r * p_r(\lambda_r, x_i)}$, where the parameter $\theta = \{\alpha_1, \alpha_2, \dots, \alpha_m; \lambda_1, \lambda_2, \dots, \lambda_m\}$.

The M-step is to estimate the parameters in the following manner: $\alpha_j = \frac{1}{n} \sum_{i=1}^n Z_{ij}$, $\lambda_j = \frac{\sum_{i=1}^n Z_{ij} x_i}{\sum_{i=1}^n Z_{ij}}$.

To apply the above EM algorithm, rRNAFilter initializes $\alpha_j = 1/m$, $\lambda_j = j * 10 + 10$ for j from 1 to m , m is set to be 10 and $k = 20$. The initial value of the two parameters, m and k , is based on our experience. It then iterates the E-steps and M-steps until the difference between the updated θ and the current θ is small ($< 1e-5$). Finally, rRNAFilter outputs the current θ .

The predicted k-mer coverage $\lambda_1, \lambda_2, \dots, \lambda_m$ are sorted in an ascending order. The abundance of non-rRNA reads will be most likely close to the smallest λ and the abundance of rRNA reads will be most likely close to a large λ . We will then assign rRNA reads and non-rRNA reads based on these predicted abundances in the following sections.

2.3. Assign reads with the first mode, rRNAFilter_m1

For each read, rRNAFilter calculates the average frequency of all its k-mers in input reads as the estimated abundance of this read. Since the average abundance of the non-rRNA reads is much smaller than that of rRNA reads, rRNAFilter assumes that k-mers with an abundance close to the smallest estimated abundance, λ_1 , are highly likely from non-rRNA reads. If the average abundance of a read is not larger than λ_1 , rRNAFilter identifies this read as a non-rRNA read. Because there exist high-abundance mRNAs as well, rRNAFilter defines rRNA reads with a slightly larger abundance, λ_5 . If the average abundance of a read is larger than λ_5 , rRNAFilter identifies this read as an rRNA read. The λ_5 is chosen based on our observations from experimental datasets. The reads with the abundance between λ_1 and λ_5 are labeled as unassigned.

2.4. Assign remaining reads with the second mode, rRNAFilter_m2

To assign the remaining reads, rRNAFilter evaluates whether an unassigned read is more similar to rRNA reads than to non-rRNA reads. RRNAFilter trains two ninth-order Markov chains using the above grouped rRNA reads and non-rRNA reads to describe their common characteristics, respectively. The ninth-order Markov chain is used based on our previous study (Wang et al., 2016). Note that the Markov chain trained from rRNA reads is more informative than that trained from non-rRNA reads since

the number of rRNA reads in the dataset is usually much larger than that of non-rRNA reads. We have also tested lower order Markov chains. The ninth order of Markov chains can in general help to assign rRNA reads more accurately. For each group of reads (rRNA reads or non-rRNA reads), rRNAFilter calculates the stationary and transition probabilities of the ninth-order Markov chain by counting the 9-mer and 10-mer frequencies on both positive and negative strands of the reads assigned to the corresponding read group. A pseudocount, 0.0001, is added to each count to avoid any count to be zero. With the trained Markov chains, the similarity score of a single-end read $a_1a_2 \dots a_n$ to a Markov chain with the stationary probability matrix, S , and transition matrix, T , is calculated as the maximum of the following two items: $S(a_1a_2 \dots a_9) * T(a_{10}|a_1a_2 \dots a_9) * T(a_{11}|a_2a_3 \dots a_{10}) * \dots * T(a_n|a_{n-9}a_{n-8} \dots a_{n-1})$ and $S(b_1b_2 \dots b_9) * T(b_{10}|b_1b_2 \dots b_9) * T(b_{11}|b_2b_3 \dots b_{10}) * \dots * T(b_n|b_{n-9}b_{n-8} \dots b_{n-1})$, where $b_1 b_2 \dots b_n$ is the reverse complement read of this read. For each unassigned read, rRNAFilter_m2 assigns it to the group to which it has a higher similarity score. For a paired-end read, its score will be the larger of the following two scores, $(a+b)/2$ and $(a'+b)/2$, where a , a' , b , and b' are the score of one end of this read, the score of the reverse complement of this end, the score of the other end of this read, and the score of the reverse complement of the other end, respectively.

2.5. Simulated datasets and real datasets

We evaluated rRNAFilter with nine simulated datasets, each containing 1 million Illumina 100 nucleotide long reads. The first three datasets, one to three, were obtained from a previous study and comprised only rRNA reads (Kopylova et al., 2012). The second three datasets, each of which contained 500,000 rRNA reads and 500,000 mRNA reads, had reads randomly selected from the reads used in the same previous study (Kopylova et al., 2012). The last three datasets contained only rRNA reads generated from rRNA sequences randomly selected from a more comprehensive rRNA database. To make this a more comprehensive database, we combined nonredundant rRNA sequences from the following databases: SILVA (Quast et al., 2013), HOMD (Chen et al., 2010), Greengenes (DeSantis et al., 2006), and RDP (Cole et al., 2009). A fraction of rRNA sequences in these more comprehensive databases was not included in the rRNA database used by SortMeRNA (Kopylova et al., 2012). The rRNA reads were then simulated by the MetaSim tool with the randomly selected rRNA sequences (Richter et al., 2008).

We also tested rRNAFilter with six publicly available metatranscriptomic datasets. Two datasets (SRR106861 and SRR013513) were from the study of SortMeRNA (Kopylova et al., 2012) and four additional datasets (SRR1013758, SRR933607, SRR3441864, SRR3441820) were downloaded from www.ncbi.nlm.nih.gov/sra and preprocessed by the tool PRINSEQ (Schmieder and Edwards, 2011), which can filter, reformat, or trim reads, such as removing sequence copies, sequences comprising N's, and low-quality sequences.

2.6. Comparison with other tools

We compared rRNAFilter with two popular tools, SortMeRNA and riboPicker (Kopylova et al., 2012; Schmieder et al., 2012). Both tools have shown to have high recall. They were run with the default setting by using the same rRNA reference database from SortMeRNA.

3. RESULTS

3.1. rRNAFilter reliably identified rRNA reads on simulated datasets

To see how well rRNAFilter could identify rRNA reads, we tested it on the first three simulated datasets. These datasets were from the study of SortMeRNA, with all reads in each dataset from rRNA sequences. We calculated the recall to measure the accuracy of rRNAFilter as the precision would be one on these datasets. To compare, we also applied two popular tools, SortMeRNA and riboPicker (Kopylova et al., 2012; Schmieder et al., 2012), to the same datasets. The recall of the three methods on the three datasets was shown in Table 1. Note that since rRNAFilter_m1 did not consider all input reads, we only used all its assigned reads to calculate its recall. On average, rRNAFilter_m2 had a 3.14% higher recall than riboPicker and a 0.22% lower recall than SortMeRNA. Moreover, rRNAFilter_m1 confidently filtered more than 74% of total rRNA reads in each dataset. Its recall was 0.1% higher than that of SortMeRNA and 3.47% higher

TABLE 1. THE TOOL COMPARISON ON THE SIMULATED DATASETS

	<i>rRNAFilter_m1</i> (%)	<i>%filtered</i> <i>reads by</i> <i>rRNAFilter_m1</i>	<i>rRNAFilter_m2</i> (%)	<i>SortMeRNA</i> (%)	<i>riboPicker</i> (%)
Dataset1 (rRNA)	99.97	80.46	99.56	99.94	96.22
Dataset2 (rRNA)	100	89.17	99.99	99.85	94.23
Dataset3 (rRNA)	99.998	74.85	99.45	99.88	99.12
Dataset4 (mix)	100/100	87.71	100/97.22	100.00/99.94	98.03/72.27
Dataset5 (mix)	100/100	89.99	100/96.94	100.00/99.94	96.56/72.20
Dataset6 (mix)	100/100	86.75	100/97.09	100.00/99.94	99.41/72.27
Dataset7 (new rRNA)	99.99	37.90	95.06	1.99	0.16
Dataset8 (new rRNA)	99.96	42.91	96.60	18.97	3.14
Dataset9 (new rRNA)	100	79.89	99.996	13.94	1.06

The recalls are provided for the first and last groups of three datasets since the precision is one for all datasets for all tools. The recall/precision are provided for the second group of three datasets in the format of recall/precision.
rRNA, ribosomal RNA.

than that of riboPicker. This indicated that most reads in a metatranscriptomic dataset could be reliably filtered by rRNAFilter with high confidence and without any reference database.

We further tested rRNAFilter with the second three datasets. Although the reads in these three datasets were also from the study of SortMeRNA, half of the reads in each dataset were from non-rRNA sequences. We calculated both recall and precision for the three methods on these three datasets (Table 1). We found that rRNAFilter_m2 had the same recall as SortMeRNA, which was higher than the recall of riboPicker. Its precision was better than that of riboPicker while slightly lower than SortMeRNA. Moreover, rRNAFilter_m1 had the same recall as SortMeRNA, but a better precision than SortMeRNA and riboPicker.

Since rRNAFilter does not depend on known rRNA sequences, while other methods do, we further compared the three methods on three additional datasets that contained rRNA reads from rRNA sequences that may not be used to train SortMeRNA and riboPicker. We found that SortMeRNA and riboPicker could hardly work on these datasets, while rRNAFilter had similar performance as that on the above six datasets. This demonstrated that rRNAFilter can filter rRNA reads without relying on any rRNA reference database. It also suggested the wide applications of rRNAFilter to the current metatranscriptomic datasets, which commonly contain unknown rRNA sequences.

3.2. rRNAFilter performs well on experimental datasets

We also tested the tools on six publicly available metatranscriptomic datasets, including two (SRR106861 and SRR013513) from the study of SortMeRNA (Materials and Methods section). With not much annotation of the reads in these datasets, we evaluated the tools using BLASTN (Camacho et al., 2009) with the default parameters and the same rRNA reference database used by SortMeRNA. The rRNA reads identified by BLASTN were considered as true rRNA reads.

rRNAFilter showed a similar recall as SortMeRNA and a better recall than riboPicker (Table 2). On average, rRNAFilter_m2 had a 3.18% higher recall than riboPicker and a 0.007% lower recall than SortMeRNA.

TABLE 2. THE TOOL COMPARISON ON EXPERIMENTAL DATASETS

Dataset	BLASTN		<i>%filtered</i> <i>reads by</i>		<i>SortMeRNA</i> (%)	<i>riboPicker</i> (%)
	(#reads)	<i>rRNAFilter_m1</i> (%)	<i>rRNAFilter_m1</i>	<i>rRNAFilter_m2</i> (%)		
SRR106861	96,702 (90,077)	100/99.60	87.85	(100,520) 99.998/96.20	(97,151) 99.999/99.54	(94,204) 97.34/99.92
SRR1013758	243,596 (274,822)	99.998/73.84	74.48	(345,432) 99.92/70.46	(250,381) 99.75/97.04	(235,692) 96.41/99.65
SRR933607	2,312,264 (2,293,483)	100/98.49	90.02	2,428,310 99.80/95.03	(2,340,901) 99.98/98.75	(2,284,631) 98.41/99.60
SRR013513	146,497 (134,009)	99.99/94.47	72.48	175,561 99.97/83.42	(150,619) 100/97.26	(130,820) 89.1/99.78
SRR3441864	749,697 (733,358)	100/99.91	73.11	(752,730) 99.999/99.60	(750,546) 100/99.89	(747,340) 99.66/99.97
SRR3441820	131,1481	1,307,825 100/99.93	97.45	(1,314,212) 100/99.79	1,312,764 100/99.90	1,307,825 99.70/99.97

For each tool, the number of identified rRNA reads is provided, followed by the recall/precision of the tool.

TABLE 3. THE COMPARISON OF ADDITIONALLY MAPPED rRNA AND NON-rRNA READS

Dataset	<i>rRNAFilter_m2</i>		<i>SortMeRNA</i>	
	#additionally predicted rRNA reads	#new mapped rRNA reads	#additionally predicted rRNA reads (%shared reads)	#new mapped rRNA reads (%shared mapped reads)
SRR106861	3820	1563	450 (99.11)	449 (99.33)
SRR1013758	102,026	74,275	7404 (84.72)	7105 (85.62)
SRR933a607	120,777	54,966	29,173 (98.99)	29,087 (99.07)
SRR013513	29,102	14,533	4122 (97.26)	4043 (97.80)
SRR3441864	3043	2152	852 (96.95)	814 (97.67)
SRR3441820	2742	1937	1284 (98.36)	1275 (98.43)

The #additionally predicted rRNA reads are the number of predicted rRNA reads that cannot be mapped to rRNA sequences with the default BLASTN and the default rRNA database. The #new mapped rRNA reads are the number of additionally predicted rRNA reads that are mapped to rRNA sequence with the new BLASTN parameter and the more comprehensive rRNA database. The percentages shown in the fourth and fifth column are the percentages of reads from SortMeRNA that are shared by the reads from rRNAFilter.

RRNAFilter_m1 confidently filtered more than 72% of reads, with its average recall 3.34% higher than that of riboPicker and 0.04% higher than that of SortMeRNA. This indicated that rRNAFilter_m1 could be used to initially filter most rRNA reads in a metatranscriptomic dataset with high accuracy and fast speed. For these datasets, rRNAFilter_m1 showed a higher precision than SortMeRNA for three of the six datasets, while rRNAFilter_m2 showed a lower precision than the other two methods for almost all datasets.

The lower precision of rRNAFilter_m2 may be due to the fact that there exist unknown rRNA reads that could not be detected by the above procedure. To see whether this hypothesis was true, we compared the predicted rRNA reads with rRNA sequences in a more comprehensive database (Materials and Methods section). We mapped these predicted rRNA reads by each method to this new rRNA database by BLASTN with a smaller word_size parameter, 7. We found that a large number of predicted rRNA reads that were not identified as rRNA reads by the default BLASTN were detected as rRNA reads by the new procedure (Table 3).

To test the significance of the observed large number of additional rRNA reads in each dataset, we also mapped the predicted non-rRNA reads in each dataset with the same new procedure. By Fisher's exact test, we found that the observed number of predicted rRNA reads that were mapped to rRNA sequences in this

FIG. 2. The running time comparison on six experimental datasets.

new database is not by chance ($p < 2.9\text{e-}122$), suggesting that these additionally mapped rRNA reads were likely true rRNA reads.

Since the above new procedure made sense in mapping unknown rRNA reads, we further compared how well rRNAFilter_m2 and SortMeRNA predicted the additional rRNA reads (Table 3). We did not consider riboPicker since SortMeRNA was superior to riboPicker on all datasets. In all six datasets, much more number of predicted rRNA reads by rRNAFilter could be mapped to rRNA sequences in the more comprehensive rRNA database than that by SortMeRNA. Moreover, almost all predicted rRNA reads that could be mapped to rRNA sequences in the new database by SortMeRNA were also predicted by rRNAFilter. Interestingly, in all six datasets, the percentage of the SortMeRNA-predicted rRNA reads that were also predicted by rRNAFilter was smaller than the percentage of the SortMeRNA-predicted rRNA reads that could be mapped to new rRNA sequences and were also predicted by rRNAFilter, supporting the good quality of the predicted rRNA reads by rRNAFilter. It is also worth pointing out that rRNAFilter_m2 had a higher recall than SortMeRNA, with the additionally mapped reads (Supplementary Table S1).

We also compared the speed of the tools. Because SortMeRNA and riboPicker compare reads with known rRNA databases, they are time-consuming. On the other hand, rRNAFilter, which does not compare sequences, has a much faster speed, about 26 times faster than BLASTN, 5 times faster than SortMeRNA, and 6 times faster than riboPicker (Fig. 2).

4. DISCUSSION AND CONCLUSION

We developed a novel approach rRNAFilter to remove rRNA reads from metatranscriptomic datasets. Our method is different from existing approaches that rely on rRNA reference databases. It considers the abundance difference of rRNA reads and non-rRNA reads to separate them, without depending on any reference database. Compared with other methods, rRNAFilter has a comparable accuracy and much faster speed.

Due to the limitation of the current rRNA databases, rRNAFilter showed a lower precision than other methods. This was especially evident in the experimental dataset, SRR1013758. Compared with a more comprehensive rRNA database, we demonstrated that rRNAFilter indeed predicted a much larger number of true rRNA reads than existing methods and its precision was significantly improved with the more comprehensive database (Supplementary Table S1).

We also want to point out that although the additionally predicted rRNA reads may be from real rRNA sequences, some of them may be from unknown abundant mRNA, or other types of RNA sequences. In fact, we noticed that some predicted rRNA reads by rRNAFilter and SortMeRNA can be mapped to the ribosomal protein gene (RPG) mRNA sequences (Nakao et al., 2004) in the six experimental datasets (Supplementary Table S2). We also observed that slightly fewer rRNA reads predicted by rRNAFilters were similar to RPG mRNA sequences than those predicted by SortMeRNA, although rRNAFilter predicted much more rRNA reads than SortMeRNA.

In practice, one can run both rRNAFilter and existing tools and combine the predicted rRNA reads. When run existing tools, we recommend to run rRNAFilter with the first mode first and then apply the existing tools to the remaining unassigned reads. In this way, we may be able to remove rRNA reads from unknown rRNA sequences confidently. At the same time, we will be able to employ the known rRNA sequence information to further remove rRNA reads.

ACKNOWLEDGMENTS

This work has been supported by the National Science Foundation [Grant Nos. 1356524, 1149955, and 1218275] and the National Institutes of Health [Grant No. 2R01HL048044]. Funding for open access charge was provided by The National Science Foundation grant 1218275.

AUTHOR DISCLOSURE STATEMENT

No competing financial interests exist.

REFERENCES

- rRNAFilter can be freely downloaded at <http://hulab.ucf.edu/research/projects/rRNAFilter/rRNAFilter.html>.
- Bailly, J., Fraissinet-Tachet, L., Verner, M.-C., et al. 2007. Soil eukaryotic functional diversity, a metatranscriptomic approach. *ISME J.* 1, 632–642.
- Camacho, C., Coulouris, G., Avagyan, V., et al. 2009. BLAST+: Architecture and applications. *BMC Bioinformatics.* 10, 421.
- Chen, T., Yu, W.-H., Izard, J., et al. 2010. The Human Oral Microbiome Database: A web accessible resource for investigating oral microbe taxonomic and genomic information. *Database.* 2010, baq013.
- Cole, J.R., Wang, Q., Cardenas, E., et al. 2009. The Ribosomal Database Project: Improved alignments and new tools for rRNA analysis. *Nucleic Acids Res.* 37, D141–D145.
- Desantis, T.Z., Hugenholtz, P., Larsen, N., et al. 2006. Greengenes, a chimera-checked 16S rRNA gene database and workbench compatible with ARB. *Appl. Environ. Microbiol.* 72, 5069–5072.
- Gilbert, J.A., and Hughes, M. 2011. Gene expression profiling: Metatranscriptomics. In *High-Throughput Next Generation Sequencing: Methods and Applications*, pp. 195–205.
- Gosalbes, M.J., Durbán, A., Pignatelli, M., et al. 2011. Metatranscriptomic approach to analyze the functional human gut microbiota. *PLoS One.* 6, e17447.
- He, S., Wurtzel, O., Singh, K., et al. 2010. Validation of two ribosomal RNA removal methods for microbial metatranscriptomics. *Nat. Methods.* 7, 807–812.
- Kopylova, E., Noé, L., and Touzet, H. 2012. SortMeRNA: Fast and accurate filtering of ribosomal RNAs in metatranscriptomic data. *Bioinformatics.* 28, 3211–3217.
- Li, X., and Waterman, M.S. 2003. Estimating the repeat structure and length of DNA sequences using ℓ -tuples. *Genome Res.* 13, 1916–1922.
- Nakao, A., Yoshihama, M., and Kenmochi, N. 2004. RPG: The ribosomal protein gene database. *Nucleic Acids Res.* 32, D168–D170.
- Quast, C., Pruesse, E., Yilmaz, P., et al. 2013. The SILVA ribosomal RNA gene database project: Improved data processing and web-based tools. *Nucleic Acids Res.* 41, D590–D596.
- Richter, D.C., Ott, F., Auch, A.F., et al. 2008. MetaSim—A sequencing simulator for genomics and metagenomics. *PLoS One.* 3, e3373.
- Schmieder, R., and Edwards, R. 2011. Quality control and preprocessing of metagenomic datasets. *Bioinformatics.* 27, 863–864.
- Schmieder, R., Lim, Y.W., and Edwards, R. 2012. Identification and removal of ribosomal RNA sequences from metatranscriptomes. *Bioinformatics.* 28, 433–435.
- Stewart, F.J., Ottesen, E.A., and DeLong, E.F. 2010. Development and quantitative analyses of a universal rRNA-subtraction protocol for microbial metatranscriptomics. *ISME J.* 4, 896–907.
- Tripp, H.J., Hewson, I., Boyarsky, S., et al. 2011. Misannotations of rRNA can now generate 90% false positive protein matches in metatranscriptomic studies. *Nucleic Acids Res.* 39, 8792–8802.
- Wang, Y., Hu, H., and Li, X. 2015. MBBC: An efficient approach for metagenomic binning based on clustering. *BMC Bioinformatics.* 16, 36.
- Wang, Y., Hu, H., and Li, X. 2016. MBMC: An effective Markov chain approach for binning metagenomic reads from environmental shotgun sequencing projects. *OMICS.* 20, 470–479.

Address correspondence to:

Dr. Haiyan Hu

Department of Computer Science

University of Central Florida

Orlando, FL 32816

E-mail: haihu@cs.ucf.edu

Dr. Xiaoman Li

Burnett School of Biomedical Science

College of Medicine

University of Central Florida

Orlando, FL 32816

E-mail: xiaoman@mail.ucf.edu