Optimizing error correction of RNAseq reads
Matthew D. MacManes \( ^1 \ast \)

\( ^1 \)Department of Molecular Cellular and Biomedical Sciences, University of New Hampshire, Durham, NH 03824. USA

ABSTRACT
Motivation: The correction of sequencing errors contained in Illumina reads derived from genomic DNA is a common pre-processing step in many de novo genome assembly pipelines, and has been shown to improved the quality of resultant assemblies. In contrast, the correction of errors in transcriptome sequence data is much less common, but can potentially yield similar improvements in mapping and assembly quality. This manuscript evaluates several popular read-correction tool's ability to correct sequence errors commonplace to transcriptome derived Illumina reads.

Results: I evaluated the efficacy of correction of transcriptome derived sequencing reads using using several metrics across a variety of sequencing depths. This evaluation demonstrates a complex relationship between the quality of the correction, depth of sequencing, and hardware availability which results in variable recommendations depending on the goals of the experiment, tolerance for false positives, and depth of coverage. Overall, read error correction is an important step in read quality control, and should become a standard part of analytical pipelines.

Availability: Results are non-deterministically repeatable using AMI:ami-3dae4956 (MacManes_EC,2015) and the Makefile available here: https://goo.gl/oVIuE0

Contact: matthew.macmanes@unh.edu and @PeroMHC

1 INTRODUCTION
Genome-enabled biology – the study of biological phenomenon empowered by the use of high throughput sequencing of transcriptomes [MacManes and Eisen, 2014, Ferreira et al., 2013, Balakrishnan et al., 2014], genomes [Castoe et al., 2013, Bactrian Camels Genome Sequencing and Analysis Consortium et al., 2012], and epigenomes [Lyko et al., 2010, Lin et al., 2014] has grown in popularity over the past several years. Much of this growth has been driven by relatively cheap sequencing generated on the Illumina platform. Unlike the previous generation of sequence data (e.g., Sanger) where error rates were far below 1%, the rate of error typical of the Illumina platform is between 1% and 3% [Wang et al., 2012]. This higher error rate is often considered mitigated by high sequencing coverage, which may often result in each nucleotide being sequenced more than 100 times (100x coverage). When depth of coverage \( \gg \) the number or expected sequencing errors, these errors may be efficiently detected by assemblers and eliminated from the assembly, often early during the creation of the assembly graph [Compeau et al., 2011, Pevzner et al., 2001]. Though shotgun sequencing is expected produce uniform coverage, certain genomic features (e.g., biased GC content) may inhibit the library construction or sequencing process, resulting in coverage valleys. The reconstruction of these regions may be significantly improved by read-error correction.

Concomitant with rapid improvement in quality and quantity of genomic data, and release of genome assemblies has been the development of novel algorithms for the correction of genomic data. These tools (reviewed in [Yang et al., 2013, Molnar and Ilie, 2014]) typically cluster identical and nearly-identical subreads of length k (kmers) then, in a probabilistic framework, attempt to minimize the Hamming (edit) distance of the reads to a consensus kmer. These algorithms can be brown down into four general classes (kmer-spectra, suffix array, multiple sequence alignment and Hidden Markov Model based methods) [Heo et al., 2014]. These algorithms assume uniform sequence coverage, and therefore have been applied most successfully to genomic data, where the have been shown to improve the quality of genome assembly [Salzberg et al., 2012].

In contrast to DNA sequencing of genomes, RNA sequencing of the expressed parts of the genome (e.g., the transcriptome) offers unique challenges. Chief amongst these challenges include coverage that is variable with patterns of expression and the reconstruction of splice-isoforms, each of which may erode error correction algorithm’s ability to accurately distinguish sequencing error from meaningful variation. This complexity may result in higher than expected false-positive rate, or lower than expected rate of error correction. In the face of these challenges, error correction of transcriptome reads has been shown to improve transcriptome assembly [MacManes and Eisen, 2013]. Since it’s publication, new correction algorithms have been developed (e.g., BLESS [Heo et al., 2014], and older methods have benefitted from new software implementations. In addition, newer evaluation algorithms had been developed (for example, see evaluation in Li, 2015). Given this, the current work aims to extend on previous work to additional error correction tools and newer analytics, while providing concrete recommendations to researchers interested in selecting an optimal software package for the correction of RNA sequencing reads.

2 METHODS
To evaluate the efficacy of read-based error correction of transcriptome data, I used a well characterized [Han et al., 2013] publicly available (SRR797058) Mus RNAseq dataset. Because the efficacy of error correction may vary with depth of sequencing, I randomly subsampled the full dataset to 10, 20, 50, 100 million paired end reads using the subsampler.py tool.
script available here (https://goo.gl/llf3zm). The resultant subsets were trimmed with the software package Trimmomatic version 0.32 [Bolger et al., 2014] using recommendations from [MacManes, 2014]. Reads were located on GitHub (https://goo.gl/oVIuE0). Note the RAM requirements for the image (AMI: ami-3dae4956). A makefile for recreating the analysis is script contained in the

### 3 RESULTS

Error correction of RNA sequencing resulted in a dramatic improvement in the number of error contains in sequence reads. This effect is highly variable depending on the specific error correction algorithm and kmer used as well as the depth of sequencing coverage. When reflecting on the different metrics, it becomes clear they vary with respect to the aggressiveness of correction.

Researchers interested in selecting a tool may choose based on different metrics. For instance, some researchers may choose based on the corrector that makes the fewest mistakes while others may decide on a different optimality criteria. Despite the fact that the correctors are variable in efficacy, several patterns emerge. First, SGA is very aggressive when applied to transcriptome data. It makes, often by an order of magnitude, more reads better. This improvement however, if buffered by the fact that it makes the most reads worse and therefore is generally not an appropriate choice for the correction of transcriptome data. Next, the correction tool Lighter makes, in all of the tests conducted, the fewest number of reads worse (e.g., increases the nm tag value infrequently). For researchers concerned about erroneous correction, this appears to be an optimal choice. Lastly, both correctors bfc and Seecer appear to preform well for a variety of metrics and all tested sequencing depths.

### 4 DISCUSSION

Lighter, bfc, SGA and BLESS were all developed for genomic data – that many of them performed quite well was somewhat surprising, given the unique characteristics of transcriptome data. In contrast to these, Seecer was developed specifically for transcriptome sequence data. While it was most efficacious when applied to high coverage data (though with high RAM requirement), it also performed well for lower coverage datasets. bfc was most efficacious at low coverages, and was slightly worse than Seecer at higher coverage.
5 CONCLUSION

In conclusion, I offer the following recommendations for researchers interested in selecting an optimal tool for error correction.

1. For sequencing experiments where less than 50 million paired end reads are collected, the software bfc appears to offer an optimal solution, with SEECER running a close second.
2. For sequencing experiments where more than 50 million paired end reads are collected, the software Seecer is best, though at the cost of high RAM requirement. Bfc runs a close second.
3. In higher coverage data where a large amount of RAM is not available, bfc should be chosen.

4. If the research is sensitive to erroneous correction, even in the face of overall poorer performance, the Lighter package should be optimal.

ACKNOWLEDGEMENT

Funding: This work was supported by start up funds provided by the College of Life Science and Agriculture at the University of New Hampshire.

REFERENCES


Table 3. 50 million paired end reads. SEECER offers the best overall correction, with bfc close behind.

<table>
<thead>
<tr>
<th>RAM</th>
<th>kmer</th>
<th>perfect</th>
<th>better</th>
<th>worse</th>
<th>gain_perf</th>
</tr>
</thead>
<tbody>
<tr>
<td>raw</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>&lt;16Gb</td>
<td>33</td>
<td>5.15E+06</td>
<td>6.00E+06</td>
<td>6.30E+05</td>
<td>6.94E+06</td>
</tr>
<tr>
<td>&lt;16Gb</td>
<td>55</td>
<td>4.89E+06</td>
<td>4.99E+06</td>
<td>3.21E+05</td>
<td>5.98E+06</td>
</tr>
<tr>
<td>lighter</td>
<td>31</td>
<td>4.64E+06</td>
<td>5.21E+06</td>
<td></td>
<td>2.03E+05</td>
</tr>
<tr>
<td>SGA</td>
<td>&lt;16Gb</td>
<td>33</td>
<td>3.62E+06</td>
<td>1.95E+07</td>
<td>1.91E+07</td>
</tr>
<tr>
<td>&lt;16Gb</td>
<td>55</td>
<td>3.74E+06</td>
<td>1.95E+07</td>
<td>1.91E+07</td>
<td>5.70E+07</td>
</tr>
<tr>
<td>SEECER</td>
<td>≈69Gb</td>
<td>31</td>
<td>5.43E+06</td>
<td>6.44E+06</td>
<td>3.37E+05</td>
</tr>
<tr>
<td>bfc</td>
<td>&lt;16Gb</td>
<td>33</td>
<td>5.12E+06</td>
<td>6.63E+06</td>
<td>2.43E+05</td>
</tr>
<tr>
<td>&lt;16Gb</td>
<td>55</td>
<td>5.02E+06</td>
<td>5.96E+06</td>
<td>2.21E+05</td>
<td>7.18E+06</td>
</tr>
</tbody>
</table>

Same description as Table 1

Table 4. 100 million paired end reads. SEECER offers the best overall correction.

<table>
<thead>
<tr>
<th>RAM</th>
<th>kmer</th>
<th>perfect</th>
<th>better</th>
<th>worse</th>
<th>gain_perf</th>
</tr>
</thead>
<tbody>
<tr>
<td>raw</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>&lt;16Gb</td>
<td>33</td>
<td>1.04E+07</td>
<td>1.20E+07</td>
<td>1.66E+06</td>
<td>1.36E+07</td>
</tr>
<tr>
<td>&lt;16Gb</td>
<td>55</td>
<td>9.92E+06</td>
<td>1.03E+07</td>
<td>7.71E+05</td>
<td>1.23E+07</td>
</tr>
<tr>
<td>lighter</td>
<td>31</td>
<td>9.35E+06</td>
<td>1.06E+07</td>
<td></td>
<td>4.03E+07</td>
</tr>
<tr>
<td>SGA</td>
<td>&lt;16Gb</td>
<td>33</td>
<td>7.08E+06</td>
<td>3.90E+07</td>
<td>3.82E+07</td>
</tr>
<tr>
<td>&lt;16Gb</td>
<td>55</td>
<td>7.17E+06</td>
<td>3.89E+07</td>
<td>3.82E+07</td>
<td>7.25E+05</td>
</tr>
<tr>
<td>SEECER</td>
<td>≈126Gb</td>
<td>31</td>
<td>1.09E+07</td>
<td>1.31E+07</td>
<td>6.78E+05</td>
</tr>
<tr>
<td>bfc</td>
<td>&lt;16Gb</td>
<td>33</td>
<td>9.88E+06</td>
<td>1.22E+07</td>
<td>4.61E+05</td>
</tr>
<tr>
<td>&lt;16Gb</td>
<td>55</td>
<td>9.81E+06</td>
<td>1.15E+07</td>
<td>4.33E+05</td>
<td>1.37E+07</td>
</tr>
</tbody>
</table>

Same description as Table 1
MacManes, M.D.

Frank Lyko, Sylvain Foret, Robert Kucharski, Stephan Wolf, Cassandra Falckenhayn, and Ryszard Maleszka. The Honey Bee Epigenomes: Differential Methylation of Brain DNA in Queens and Workers. PLOS Biology, 8(11):e1000506, November 2010.