Mansoura University
Faculty of Computers and Information sciences
Department of medical informatics

Error correction
"lighter tool"

Submitted by:

Student Name	Student Email	Section
Zeinab Mohamed Elazony	Zalazony@gmail.com	Sec 14
Salma Alzeky Ahmed	salmaalzekey@gmail.com	Sec 4
Aya Elsayed Elmetwaly	aya.sayed5777@gmail.com	Sec 4
Somaya Ragab Kamal	eng.somayaragab@gmail.com	Sec 4
Souad Borham Anwar	soaad.brham@gmail.com	Sec 4

Project Abstract:

Current sequencing technologies, especially next-generation sequences, provide us with massive amounts of sequencing data, which are full of errors.

Correcting these errors is important for achieving high accuracy in various next-generation sequencing applications, such as genome assembly, genome sequencing, and SNP haplotyping (single nucleotide polymorphism).) More accurate error correction methods for these sequencer reads are highly desired. In this project, we present a new algorithm for error-correction of sequencer reads. As many of previous error correcting algorithms do, we also utilize k-mer frequency table, but we achieve higher accuracy by considering non-exact matching between k-mers. To evaluate the accuracy of our algorithm, we compared it to the most recent and well-known error-correction algorithm.
Experiments showed that our algorithm outperforms the previous best-known algorithm in terms of accuracy.

Whatever the genome sizes and coverage ranges are, in both the case of short reads and long reads.
In our project, we'll use the lighter tool as an error-correcting algorithm.

For whole genome sequencing results, Lighter is a kmer-based

error correction tool.
Lighter obtains a collection of kmers that are possibly from the genome through sampling (rather than counting).
Using this knowledge, Lighter can correct the reads containing sequence errors.

Project objectives:

1-Obtain a more precise DNA sequencing.

2-Decide on error-correcting software that is efficient in terms of time and memory.

3-Cover a large trade-off space in the use tool between accuracy, speed, and memory- and storage-efficiency.

4- increase the efficiency of genome sequencing and lower the cost of genome sequencing.

Introduction:

Error correction is the process by which a computer automatically correct mistakes in DNA.In NGS (next generation sequence)technologies, there are two types of errors: substitution errors and indels. Substitution errors occur when a single base is moved to a different base, and indels occur when bases in a read are added or removed.(inserted or deleted) Correcting errors in the reads can greatly increase the performance of applications that use NGS data such as genome assembly and variant calling. The Illumina platform predominantly makes substitution errors, and since it is the dominant technology most software developed for correcting errors focuses on substitution errors. Some of the most successful error correction programs that correct substitution errors include BLESS , Coral , HiTEC, Musket , RACER, SGA , and SHREC . Most error correction programs make use of “k-mers” in a read to find and correct errors. A k-mer is a sub-sequence of a read with length k.

Lighter (Light error correction) :

- Lighter is a tool for correcting sequencing errors that is fast and memory-efficient.Lighter does not count k-mers.Instead, it employs two Bloom filters, one of which holds a sample of the input k-mers and the other of which holds k-mers that are most likely to be right.Lighter is parallelized and Uses no secondary storage and achieves equivalent accuracy while being quicker and more memory-efficient than competing approaches.
- The most important advantage is that Lighter's parameters can be set such that memory footprint and accuracy are near constant with respect to depth of sequencing. That is, no matter how deep the coverage, Lighter can allocate the same sized Bloom filters and achieve nearly the same: (a) Bloom filter occupancy, (b) Bloom filter false positive rate and (c) error correction accuracy. Lighter does this without using any disk space or other secondary memory. This is in contrast to BLESS and Quake/Jellyfish, which use secondary memory to store some or all of the k-mer counts.
- Lighter is free open-source software that has been compiled and tested on Linux, Mac OS X, and Windows computers under the GNU GPL licence.The source code for the programme is also available.

- Lighter has three parameters the user must specify: the k-mer length k, the genome length G and the sub-sampling fraction α. While the performance of Lighter is not overly sensitive to these parameters , it is not desirable to leave these settings to the user. In the future, we plan to extend Lighter to estimate G, along with appropriate values for k and α, from the input reads. This could be accomplished with methods proposed in the K-merGenie and Kmer Stream studies.
- Light Assembler uses only two passes over the sequenced reads to identify the approximate set of trusted nodes without error correction or intensive graph simplification modules. Also, one of the efficient representations of De Bruijn graph based on a Bloom filter is implemented in Minia and uses k-mer counting module to identify the set of trusted k-mers. Minia's counting algorithm follows a divide and conquer paradigm and utilizes the disk space as secondary memory storage. Without the use of a counting module or additional disc space, our method will define the collection of trustworthy k-mers.

Using simulated and benchmarked datasets, we will compare our findings to current state-of-the-art sequence assemblers as well as resource-efficient one.

Method:

- The first pass obtains a sample of the k-mers present in the input reads, storing the sample in Bloom filter A.
- The second pass uses Bloom filter A to identify solid k-mers, which it stores in Bloom filter B
- The third pass uses Bloom filter B and a greedy procedure to correct errors in the input reads.

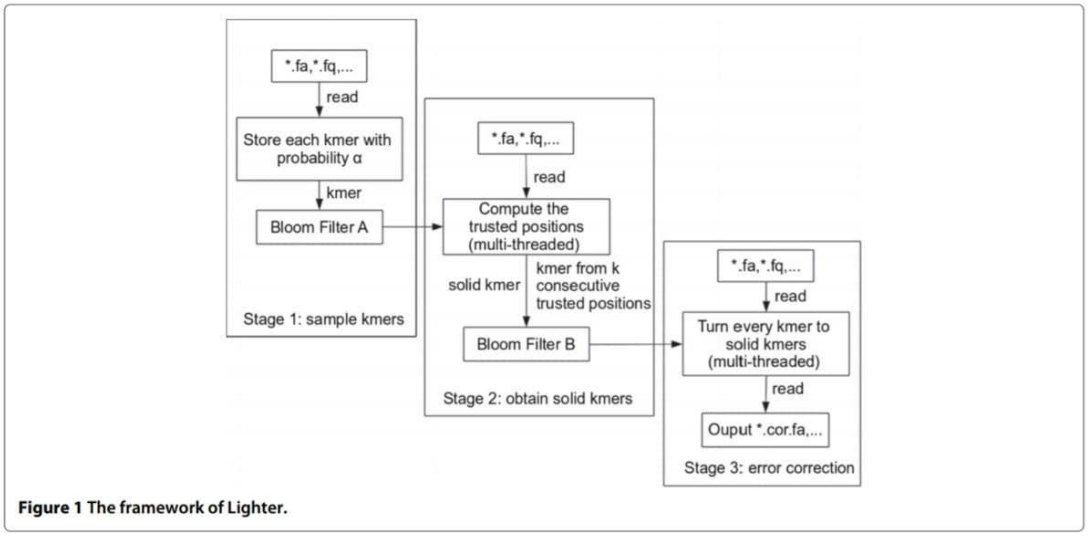

Figure 1 The framework of Lighter.

Stages of the method

First pass

In the first pass, Lighter examines each k-mer of each read. With probability 1 − α,the k-mer is ignored. k-mers containing ambiguous nucleotides (e.g. 'N') are also ignored. O otherwise, the k-mer is cannibalized and added to Bloom filter A
The sub sampling fraction α is set by the user. We suggest adjusting α in inverse proportion to depth of sequencing, for reasons discussed below. For experiments described here, we set α = 0.1 when the average coverage is 70-fold. That is, we set α to 0.1(70/C),where C is average coverage.

Second pass

A read position is overlapped by up to xk-mers, 1 ≤ x ≤k,where x depends on how close the position is to either end of the read . For a position altered by sequencing error,the overlapping k-mers a re all incorrect and are unlikely to appear in A. We apply a threshold such that if the number of k-mers overlapping the position and appearing in Bloom filter A is less than the threshold, we say the position is untrusted.Otherwise we say it is trusted.

Third pass:

In the third pass, Lighter applies a simple, greedy error correction procedure read r of length |r|,contains|r|−k + 1 k-mers. kidenotes the kmer starting at read position i,1≤ i ≤|r|−k + 1. We first identify the longest stretch of consecutive k-mers in the read that appear in Bloom filter B. Let k band kebethe k-mers at the left and right extremes of the stretch. If e < |r|−k + 1, we examine successive k-mers to the right starting at ke+ 1. For a k-mer kithat does not appear in B,we assume the nucleotide at off set i + k − 1 is incorrect.

Literature Review:

In recent years, the rapid advancement of next-generation DNA sequencing has revolutionized biological and ecological science.

Scientists tested lighter,Quake,SOApec,Musket, and Bless tools for error correction on Caenorhabditis elegans, GAGE human chromosome 14, and Escherichia coli and discovered that lighter is quicker and more effective than other tools and does not require counting k-mers.
(Li Song, Liliana Florea & Ben Langmead, 509 (2014))

Reference:

Light Assembler: fast and memory-efficient assembly algorithm for

high-throughput sequencing readsSara El-Metwally 1, Magdi Zakaria 2, Taher Hamza 2

Lighter: fast and memory-efficient sequencing error correction without counting Li Song, Liliana Florea & Ben Langmead 509 (2014)

Pollux: platform independent error correction of single and mixed genomesEric Marinier, Daniel G Brown & Brendan J McConkey

Chaisson M, Pevzner P, Tang H: Fragment assembly with short reads . Bioinformatics. 2004, 20: 2067-2074. 10.1093/bioinformatics/bth205.

Glenn TC: Field guide to next-generation DNA sequencers . Mol Ecol Resour. 2011, 11: 759-769. 10.1111/j.1755-0998.2011.03024.x.