BITS :: Call for Abstracts 2023 - Oral communication

Type	Oral communication
Session	Algorithms for Bioinformatics
Title	BFQzip: lossy compression of FASTQ through extended BWT
All Authors	Guerrini V(1), Louza F. A(2), Rosone G(1)
Affiliation	

(1) Department of Computer Science, University of Pisa, Italy
(2) Faculty of Electrical Engineering, Federal University of Uberlandia, Brazil

Motivation

The growing interest in applications of genome sequencing, together with the reduced cost of these technologies has led to the generation of unprecedented amounts of increasing large genomic datasets. The raw data produced by sequencer are stored in a special text-based format called FASTQ, in which each sequenced DNA fragment is stored in a record of three main components: the header that identifies it and contains information related to the sequencing process; the nucleotide sequence of which it is composed; and the sequence of quality scores that encode a per-base estimation of the sequencer's error probability. The increase of genomic data produced daily has drawn attention to develop compressed representations which can reduce storage space and bandwidth requirements for data exchange. The majority of the state-of-the-art compressors for FASTQ files focus on compressing only one of the two main components, (i) nucleotide sequence or (ii) quality score sequence, applying known techniques or third tools to the other one. The approaches that compress only the nucleotides are lossless, since they do not perform modifications on that component, but exploit its inherent redundancy to compress them, and may use or not a shared reference sequence to encode them. Differently, the quality score component is generally compressed lossy: modifications are applied at the cost of a little distortion effect, by using or not the related biological information contained in the FASTQ file. Among the lossy approaches that evaluate the related biological information to compress only the quality score component are BEETL [Janin et al., 2014] and LEON [Benoit et al., 2015].

Methods

In [1], we proposed a novel alignment-free approach for the lossy compression of FASTQ files that performs modifications in both components, nucleotides and quality scores, by processing them at the same time. The compression scheme introduced is called BFQzip, and uses the biological information contained in the FASTQ file without combining it with any external information (it is a reference-free approach). At the core of our strategy is the Burrows-Wheeler Transform (BWT) (precisely, its extension to string collections) and its properties. The BWT is a reversible text transformation that performs a symbol permutation of the input string. The main property of the BWT is the clustering effect, i.e., the fact that equal symbols occurring in similar context are grouped together in the permuted string, which is employed by BFQzip in the basic idea: each nucleotide can with high probability be predicted by its context, and its quality score can be smoothed without producing distortion effects on downstream analysis. BFQzip uses contexts of variable length using the eBWT positional clustering recently introduced in [Prezza et al., 2019]. Thus, the workflow of BFQzip consists in first computing the BWT and its clusters, then performing modifications on both components (noise reduction and quality smoothing) at the same time, and finally output a compressed FASTQ file. We exploited the possibility to split the input FASTQ file in parts to design a parallel version of our framework, and analyzed how reordering reads before splitting the input can improve the compression ratio as the number of threads increases. Now, we plan to use the reordering of the reads internally while building the BWT to reduce the number of equal-symbol-runs in the BWT. For instance, SAP- or RLO-order are shown to be good heuristics for a BWT with fewest number of runs, a relevant compressibility measure that has been increasingly used for space and time complexity of BWT-based data structures and algorithms. We also plan to include in our strategy a way to compress the headers by difference, exploiting their systematic structure in fields. In [1], we presented two implementations of our approach: in internal and in semi-external memory.
[1] Guerrini V. Louza F. and Rosone G. Lossy Compressor Preserving Variant Calling through Extended BWT. BIOSTEC-BIOINFORMATICS 2022

Results

To test the effectiveness of our approach, we performed experiments on FASTQ datasets from human short reads, and compared the results to the tools BEETL and LEON, although (to the best of our knowledge) none of the existing lossy compressors modifies both components at the same time, and thus no comparison with existing tools is completely fair. The FASTQ files obtained by BFQzip achieved a better compression than the original data, and compression rates comparable to the other tools. Performing a lossy compression, it is important to consider the impact that the modified data may have on downstream analysis. Thus, we evaluated the effects of the modified data on variant calling, and we showed that our approach preserved the called variants with respect to the original data more than the other tools. We plan to perform more extensive experiments including larger datasets, non-human datasets, and also testing long reads.

Info

Research partially funded by PNRR - M4C2 - Investimento 1.5, Ecosistema dell'Innovazione ECS00000017 - "THE - Tuscany Health Ecosystem" - Spoke 6 "Precision medicine &

personalized healthcare", funded by the European Commission under the NextGeneration EU programme.	
filename	-
Figure	
-	
Availability	https://github.com/veronicaguerrini/BFQzip

Corresponding Author	
Name, Surname	Veronica, Guerrini
Email	veronica.guerrini@unipi.it
Submitted on	21.04.2023