Bioinformatics of nanopore sequencing

Abstract :

Nanopore sequencing is one of the most exciting new technologies that undergo dynamic development. With its development, a growing number of analytical tools are becoming available for researchers. To help them better navigate this ever changing field, we discuss a range of software available to analyze sequences obtained using nanopore technology.

Introduction :

Beginning of twenty-first century witnessed dynamic development of sequencing technology. First, so-called “Next Generation Sequencing” brought increased sequencing yield and decline of sequencing cost. However, it was with a certain price, namely the length of the reads, which is much shorter than in traditional Sanger sequencing. As a result, we are flooded with a number of genomes (as of May 30, 2019, there are over 200,000 prokaryotic

and almost 9000 eukaryotic genomes deposited at the NCBI database). However, most of these genomes are in the so-called draft form. It means that chromosomes are presented in rather small pieces for which order and orientation on a chromosome is unknown. Moreover, gene annotation in these genomes is quite poor or does not exist at all.

Base calling :

Base calling is a crucial step in any sequencing method. It is a process of transforming a raw signal obtained from a sequencer into a string of nucleotides. In the case of nanopore sequencing, it is a computational processing of electric signal collected from an ONT instrument (MinION, GridION, or PromethION). The accuracy of base calling is influenced by two factors. First, the chemistry used can affect a signal-to-noise ratio. If the ratio is low, determination of underlying DNA sequence may not be possible .The second factor is how well the signal can be interpreted by a software used for base calling. To discriminate between signal

and noise a specific training dataset is used, which may not be optimal for interpretation of real DNA molecule if the latter has, for instance, strong nucleotide composition bias. For example, the genome of malaria-causing parasite is 80% AT rich and nanopore base calling of reads from this genome is usually far from optimal, especially within homopolymers stretches.Since, the base calling is a very important step in obtaining a useful, for a customer, data, it is not surprising that the ONT is developing such software and is constantly working on its improvement. The original ONT base caller used Hidden Markov Models (HMM) approach but all current programs use neural networks machine learning approach, following the path paved by the DeepNano software .lists ten base callers developed specifically for nanopore sequencing. Since nanopore-sequencing technology is developing dynamically, base callers need to keep up the pace and understandably some of them became obsolete, e.g., albacore or metrichore. The base callers are usually available for a range of operating systems, including Linux, MacOS, and Windows. However,

since base calling is computationally intensive, it is a good idea to run the software on a multiple-processor machine or a server. It is also beneficial to perform a base calling on GPUs (graphics processing unit) instead of CPUs (central processing unit). Their highly parallel structure makes them more efficient in specialized tasks. In fact, ONT's computing unit, MinIT, designed for MinION and computing module of PromethION are equipped with 256-core GPU . For example, Chiron is about 80 times faster on a GPU when compared with a single CPU performance. Wick et al. recently compared several base callers for nanopore sequencing .In short, they all perform very similarly regarding quality score of both single reads and consensus sequences with the exception of Chiron who performed a bit worst on a single-reads level. On the other end of speed distribution lies Guppy .They also tested Guppy with different models and concluded that developing custom-trained models can improve base calling significantly.Furthermore, the accuracy of a consensus sequence can be improved by employing Nanopolish software.

Table 1 Base callers developed for nanopore sequencing

From: Bioinformatics of nanopore sequencing

Tool	Read qscore[a]	Consensus qscore[a#]	Availability
Albacore	9.2	21.9	Only to ONT customers
BasecRAWller	N/A	N/A	https://basecrawller.lbl.gov/ (seems to be down)
Chiron	7.7	21.4	https://github.com/haotianteng/Chiron
DeepNano	N/A	N/A	https://bitbucket.org/vboza/deepnano/src/master/
FastQC	A quality control tool for high throughput sequence data.		https://www.bioinformatics.babraham.ac.uk/projects/fastqc/
Flappie	9.6	22.0	https://github.com/nanoporetech/flappie
Guppy	9.7	23.0	Only to ONT customers
Metrichor	N/A	N/A	Only to ONT customers
Nanocall	N/A	N/A	https://github.com/mateidavid/nanocall
Scrappie	9.3	22.4	https://github.com/nanoporetech/scrappie

[a]based on [8]

Mapping :

Obtaining a sequence is only the beginning of the analysis. Depending on a biological question we ask or why the sequencing was done at the first place, there are several avenues that one may take. However, aligning raw reads to existing sequences is often the first task on the “to do” list. This is especially true if a sequencing project involves organisms for which genomes has been already decoded. Often, the task of aligning raw-sequencing reads to already determined sequences, for instance a genome, is called mapping raw reads to the target. Aligning is such a basic task in the molecular sequence analysis that several algorithms that deal with the problem were developed long before

bioinformatics field existed . Although these early algorithms are very elegant and guarantee optimal solution to the problem, they are computational heavy and not very practical when one has to deal with the vast number of sequences. Consequently, heuristic algorithms have been developed to conquer speed limitation of exhaustive algorithms. Probably, the most successful and the best known is the BLAST algorithm . It is worth to mention that BLAST was developed for database similarity searches and its goal is to find all the instances of similar sequences in a database. In mapping, however, the goal is different, namely to find an exact position on a genome or a transcript from which a given read comes from. In practice, BLAST and other similar tools report many alignments of a query to a set of sequences but mappers, in ideal situation, report just one alignment. In fact, if a sequencing read aligns to multiple positions in a genome, usually it will be discarded from further analysis.

Table 2 Selected aligners suitable for long reads

From: Bioinformatics of nanopore sequencing

Tool	Algorithm	Availability
BWA	Burrows–Wheeler Aligner's Smith–Waterman Alignment	http://bio-bwa.sourceforge.net
GraphMap	Gapped spaced seeds	https://github.com/isovic/graphmap
Kart	Divide and conquer	https://github.com/hsinnan75/Kart
LAMSA	Sparse dynamic programming (SDP)-based split alignment	https://github.com/hitbc/LAMSA
LAST	Adaptive seeds approach	http://last.cbrc.jp/
Minimap2	Hash table approach	https://github.com/lh3/minimap
NanoPipe	A pipeline that includes a consensus sequence calculation based on LAST alignment to a reference sequence	http://bioinformatics.uni-muenster.de/tools/nanopipe2/index.hbi
		https://github.com/IOB-Muenster/nanopipe2
NGMLR	k-mer search followed by a banded Smith–Waterman alignment algorithm	https://github.com/philres/ngmlr

Sequence assembly :

The longest known human transcript is one of the isoforms of titin gene (TTN) with more than 108 kb. One may conclude that with nanopore technology we can sequence any transcript in its entirety. However, even such long reads are not long enough to cover the whole bacterial genome or the whole eukaryotic chromosome by a single read. From the very beginning of sequencing approach, an assembly of raw reads was required in order to obtain a complete, contiguous sequences. Originally, the sequence assemblers used overlaps to merge and order raw sequences . This approach is called Overlap-Layout-Consensus (OLC). However, for the NGS data volume and short length of reads, it proved to be prohibitive to use the OLC algorithms

effectively, although there are some exemptions from this rule . In early 2000s, using de Bruijn graphs to solve assembly problems has been proposed . Algorithms based on de Bruijn graphs turned out to be very useful for short reads assembly and most of modern assemblers developed for the next generation sequencing use this approach. Interestingly, these algorithms are not very well suited for the noisy, long reads and for these sequences the interest has shifted back to the OLC approach. One of these assemblers is Canu developed based on Celera Assembler , which has been used for a successful assembly of the variety of genomes, such as bacteria , fungi , fruit fly ,cotton , or fish. Recently, de Bruijn graph approach was implemented for long reads by Pevzner group but it has not been wildly used so far. The ABruijn assembler includes a sequence polishing module and repeat analysis step, which apparently improves the structural accuracy of the final assembly . Table 3 lists assemblers suitable for long but noisy reads along with other tools useful for assembly

improvement, such as consensus sequence polishing.

Table 3 Selected software for sequence assembly and scaffolding

From: Bioinformatics of nanopore sequencing

Tool	Description	Availability
ABruijn	De novo assembler for long and noisy reads	https://github.com/bioreps/ABruijn
Canu	A hierarchical assembly pipeline based on Celara Assembler	https://github.com/marbl/canu
Cobbler	Gap filling with long sequences	https://github.com/bcgsc/RAILS
Flye	De novo assembler for single-molecule sequencing reads	https://github.com/fenderglass/Flye
HINGE	A long-read assembler based on an idea called hinging	https://github.com/HingeAssembler/HINGE
LINKS	Application for scaffolding genome assemblies with long reads	https://github.com/bcgsc/LINKS
MECAT	An ultra-fast mapping, error correction and de novo assembly tool for long reads	https://github.com/xiaochuanle/MECAT
Medaka	A tool to create a consensus sequence of nanopore-sequencing data using neural networks	https://nanoporetech.github.io/medaka/index.html
Miniasm	OLC-based de novo assembler	https://github.com/lh3/miniasm
NanoPipe	A pipeline that includes a consensus sequence calculation based on LAST alignment to a reference sequence	http://bioinformatics.uni-muenster.de/tools/nanopipe2/index.hbi
		https://github.com/IOB-Muenster/nanopipe2
Nanopolish	Software package for signal-level analysis of Oxford Nanopore-sequencing data, including consensus sequence calculation	https://github.com/jts/nanopolish
npScarf	A program that scaffolds and completes draft genomes assemblies in real time with Oxford Nanopore sequencing	https://github.com/mdcao/npScarf
PBJelly	A pipeline that aligns long sequencing reads to high-confidence draft assembles to fills the gaps	https://sourceforge.net/projects/pb-jelly/
Racon	A standalone consensus module to correct raw contigs generated by rapid assembly methods	https://github.com/isovic/racon

Miscellaneous tools :

Nanopore sequencing is still developing technology and researchers keep finding new applications for it. These usually require new, specialized software. One such an interesting application is studying DNA methylation, which is involved in many biological processes, such as gene regulation and cell differentiation . Nanopore technology enables studying DNA methylation directly, as ionic current signal should be different for methylated and unmethylated nucleotides. Two different software,

Nanopolish and SignalAlign , are using HMMs to identify C5-methylcytosine (5mC) with high accuracy. The newest addition to the methylation toolbox is DeepMod, which is using raw electric signals and a bidirectional recurrent neural network to detect 5mC and N6-methyldeoxyadenosine (6 mA) . Interestingly, the recent ONT's base caller Flappie is able to call 5mC in CpG context for R9.4.1 on PromethION platform.Another promising application of the nanopore sequencing lies within metagenomic studies. Short reads often cannot distinguish between closely related species or microbial strains, as rRNA often used for the bar coding is a very conservative molecule. Several reports showed that long reads might be a solution to that problem and few dedicated software were developed, including MetaG and the ONT's own EMPI2ME.One of the advantages of nanopore sequencing is possibility of direct sequencing of RNA molecules (see ONT's white paper at https://nanoporetech.com/resource-centre/rna-sequencing-white-paper-value-full-length-transcripts-without-bias). This includes defining

complexity of alternative transcripts, unbiased quantification of transcriptome and detection of methylated nucleotides. Several, specialized software appeared recently to specifically deal with noisy long reads. For instance, SQANTI was developed to decipher complexity of alternative transcripts. Although developed and tested with PacBio reads it can be used for any long reads, including nanopore, since it takes FASTA files as an input. FLAIR (Full-Length Alternative Isoform analysis of RNA) enables the correction, isoform definition, and alternative splicing analysis of noisy reads . ONT developed a set of tools called pinfish for long transcriptomics data analyses, which was inspired by Mandalorion pipeline . LoReAn is a tool developed for eukaryotic genome annotation utilizing short- and long-read cDNA sequencing, protein evidence, and ab initio gene prediction .

Table 5 lists few other applications that can be useful for the nanopore sequence analyses. For instance, NanoSim-H is a software to simulate the nanopore

reads . These simulated reads can be used to test performance of other analytical tools developed specifically for the ONT reads. NanoDJ integrates many tools together enabling tasks such as base calling, sequence quality assessment, and sequence assembly in a single environment. Tandem-genotypes is an interesting software that finds changes in length of tandem repeats, from “long” DNA reads aligned to a genome . It simply aligns long reads against a reference genome using last-split and then compares tandem repeats annotated in the reference genome with the aligned long read, one read at a time. Then, it summarizes the results for each tandem repeat locus annotated in the reference genome.

Table 5 Miscellaneous tools useful in the nanopore sequence analyses

From: Bioinformatics of nanopore sequencing

Tool	Description	Availability
DeepMod	Detection of DNA base modifications by deep recurrent neural network	https://github.com/WGLab/DeepMod
EPI2ME	A set of real-time analytical tools, including species identification and reads quality control	Only to ONT customers
FLAIR	Full-length alternative isoform analysis of RNA	https://github.com/BrooksLabUCSC/flair
Flappie	Base calling of 5mC in CpG context	https://github.com/nanoporetech/flappie
IGV	Integrative genomics viewer	https://software.broadinstitute.org/software/igv/home
LoReAn	Annotation pipeline designed for eukaryotic genomes using long and short reads	https://github.com/lfaino/LoReAn
Mandalorion	Analysis Pipeline to analyze Nanopore RNAseq data	https://github.com/rvolden/Mandalorion-Episode-II
MEGAN-LR	Part of MEGAN (metagenomic tool) designed for long reads.	http://ab.inf.uni-tuebingen.de/data/software/megan6/download/
MetaG	A metagenomics pipeline suitable for long-sequencing technologies	http://bioinformatics.uni-muenster.de/tools/metag/index.hbi
NanoDJ	A Jupyter notebook integration of tools for simplified manipulation and assembly of DNA sequences	https://github.com/genomicsITER/NanoDJ
Nanopolish	Software package for signal-level analysis of Oxford Nanopore-sequencing data, including methylation analysis	https://github.com/jts/nanopolish
NanoSim-H	A simulator of Oxford Nanopore reads	https://pypi.org/project/NanoSim-H/
pinfish	A collection of tools helping to make sense of long transcriptomics data	https://github.com/nanoporetech/pinfish

Conclusions :

Nanopore sequencing technology promises to democratize nucleic acid sequencing. However, as a sequence is only a raw material in gaining biological knowledge, to do so, we need analytical tools. Unfortunately, most of the software developed for interpretation of the nanopore sequences require relatively high bioinformatics skills, which most biologists lack. In the growing number of tools available for the nanopore sequence analysis, the NanoPipe seems to be an exception with a simple web interface and a clear, easy to understand output files. Nevertheless, “one swallow does not make a spring” and we need more software for easy data analysis and interpretation to make sequencing fully democratized.

References :

1.Eid J, Fehr A, Gray J, Luong K, Lyle J, Otto G, et al. Real-time DNA sequencing from single polymerase molecules.

2.Kasianowicz JJ, Brandin E, Branton D, Deamer DW. Characterization of individual polynucleotide molecules using a membrane channel. Proc Natl Acad Sci.

3.Leggett RM, Clark MD. A world of opportunities with nanopore sequencing. J Exp Bot.4.Loman NJ, Quinlan AR. Poretools: a toolkit for analyzing nanopore sequence data.

5.Rang FJ, Kloosterman WP, de Ridder J. From squiggle to basepair: computational approaches for improving nanopore sequencing read accuracy. Genome Biol.

6.Boza V, Brejova B, Vinar T. DeepNano: deep recurrent neural networks for base calling in MinION nanopore reads. PLoS ONE.

7.Teng HT, Cao MD, Hall MB, Duarte T, Wang S, Coin LJM. Chiron: translating nanopore raw signal directly into nucleotide sequence using deep learning. Gigascience.