Nanopore sequencing: *from basecalling to structural variant detection*

Yao-zhong Zhang
yaozhong@hgc.jp
Human Genome Center
University of Tokyo
2019-10-08

Materials here:
https://github.com/yaozhong/mexico_workshop_nanopore_hands_on

https://github.com/yaozhong/mexico_workshop_nanopore_hands_on

Before we started, please do …

1.Copy data and related folders

cp -r /share/lect/nanopore/ ~/

2.Prepare singularity environment

singularity build --sandbox singularity_img/nanopore docker://yaozhong/nanopore_analysis

(*) If you want to use the same environment in you local computer

(1). Install docker in your local machine
(2). https://hub.docker.com/r/yaozhong/nanopore_analysis

2nd Generation

Short-read sequencing

3rd Generation

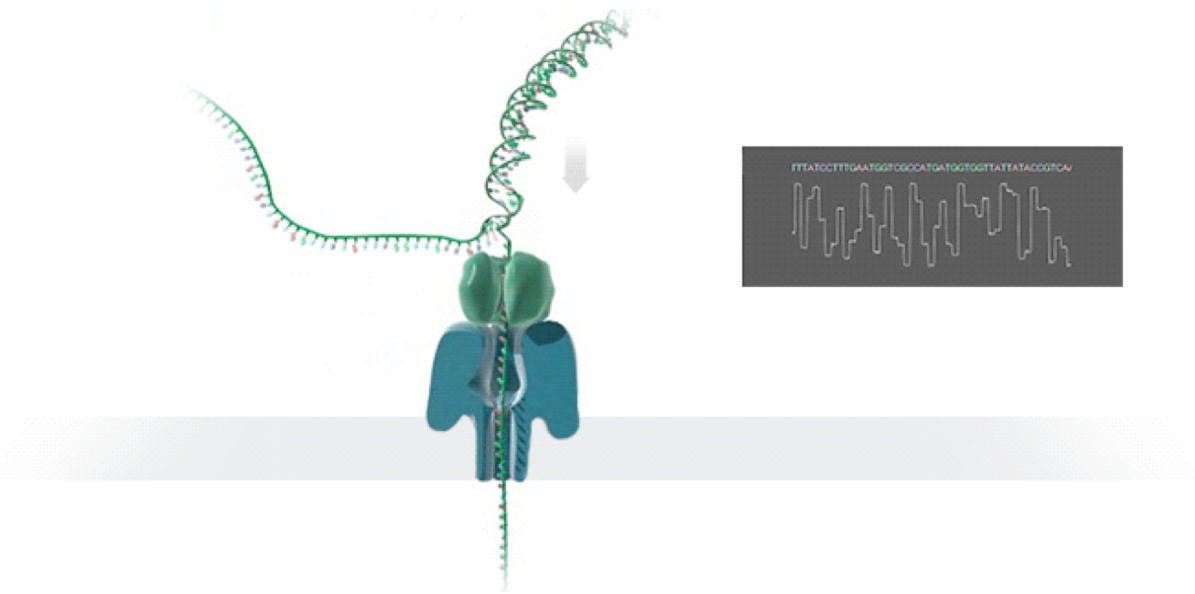
Long-read sequencing

Outline

1. *Nanopore sequencing basics*
2. *Basecalling*
3. *Assembly*
4. *Nanopore Structural variant detection*
5. *Hands-on section*

1.Nanopore sequencing basics

Nanopore sequencing (a third generation)



Advantage:

- ***Long reads*** (*up to 2Mb, avg. 10k)
- No PCR amplification
- Low-cost genotyping
- Mobility and real time

Disadvantage:

- High error rate of base-calling (85% ~ 95%)

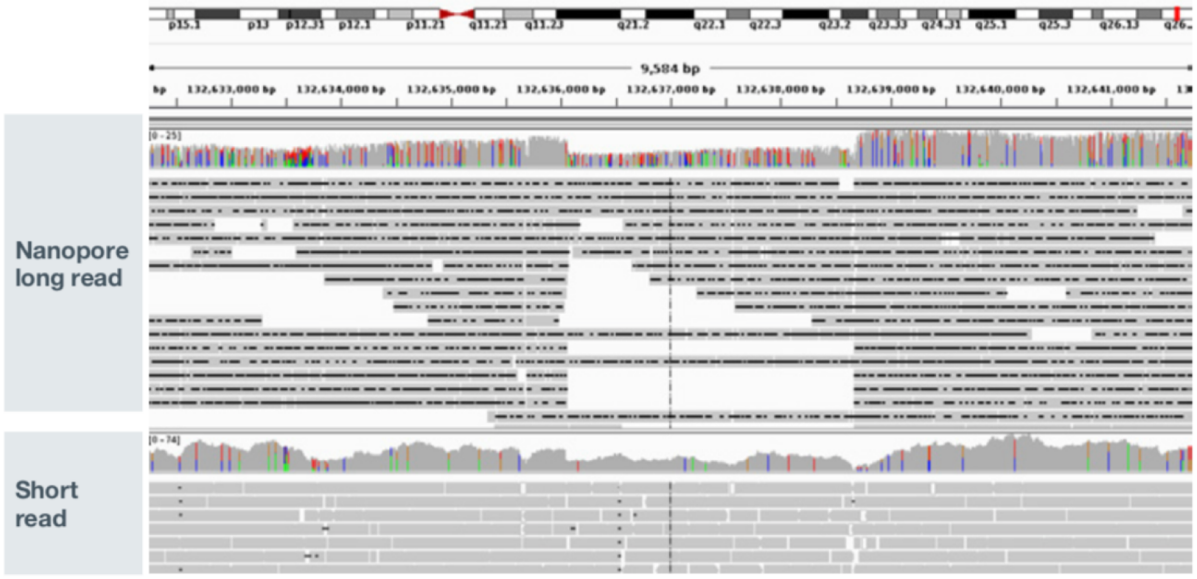
Advantage of long reads

- Easier assembly
- Ability to span repetitive genomic regions
- Identification of large structural variation

~ 50 base read
~ 92,000 “pieces”

~ 500,000 base read
~ 9 “pieces”

(ONT white papers)



Disadvantage

- Higher error rate
  - Read accuracy is in a range of 5% ~ 15%
  - Vs. Illumina 0.1% ~ 1%

- Current solutions:
  - Improving base-calling accuracy
  - Develop new error

For MinION / GridION

Flongle

Adapter to enable small, rapid nanopore sequencing tests, for mobile or desktop sequencers

MinION Mk1B

Your personal nanopore sequencer, putting you in control

MinION Mk1C

Your personal nanopore sequencer including compute and screen, putting you in control

GridION Mk1

Higher-throughput, on demand nanopore sequencing at the desktop, for you or as a service

PromethION 24/48

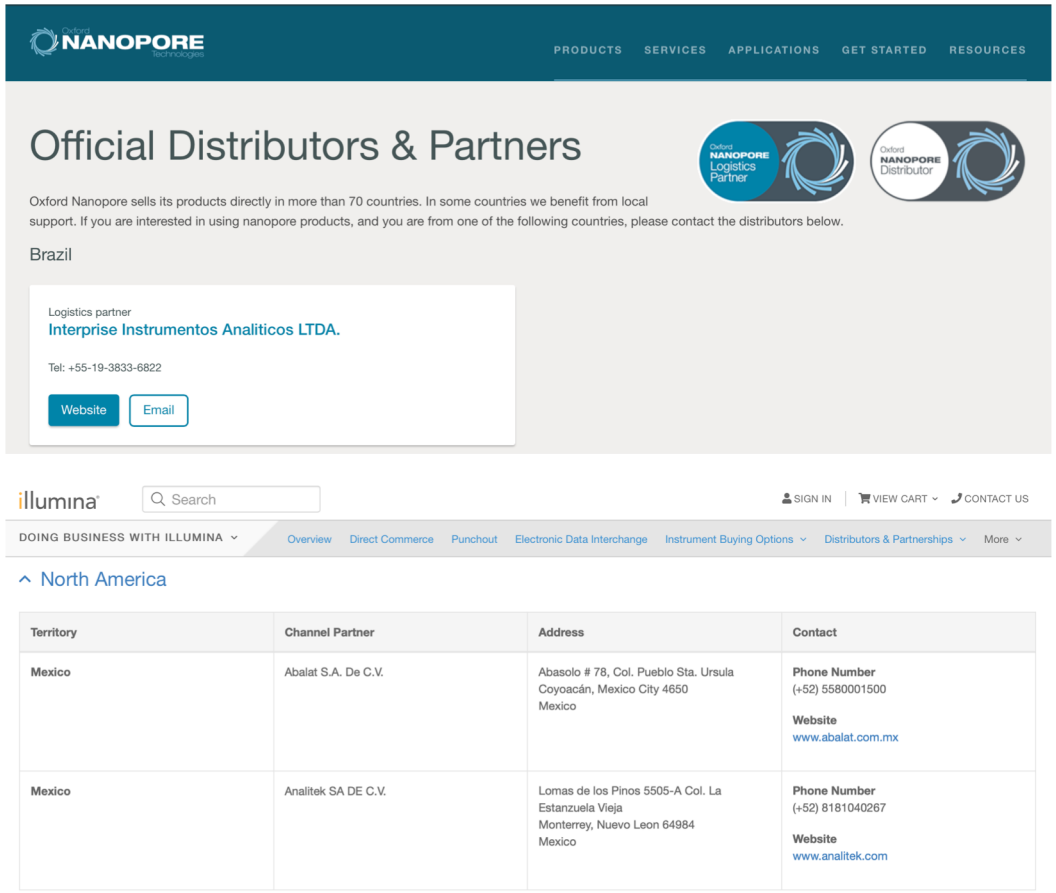
Ultra-high throughput, on-demand nanopore sequencing, for you or as a service

	Flongle	MinION Mk1B	MinION Mk1C	GridION Mk1	PromethION 24	PromethION 48
Read length	Nanopores read the length of DNA presented to them. Longest read so far: > 2Mb.					
Yield per flow cell, DNA/cDNA (Best Internal)	2 Gb	50 Gb	50 Gb	50 Gb	220 Gb	220 Gb
Number of flow cells per device	1	1	1	5	24	48
Yield per device		50 Gb	50 Gb	250 Gb	5.2 Tb	10.5 Tb
System access	From $1,760	From $1,000	From $4,900	From $49,995	From $165,000	From $285,000
Suitable applications include	Amplicons Panels/targeted sequencing Quality testing Small sequencing tests	Whole genomes/exomes Metagenomics Targeted sequencing Whole transcriptome (cDNA) Smaller transcriptomes (direct RNA) Multiplexing for smaller samples	Whole genomes/exomes Metagenomics Targeted sequencing Whole transcriptome (cDNA) Smaller transcriptomes (direct RNA) Multiplexing for smaller samples Particularly suitable for field use	Larger genomes or projects Whole transcriptomes (direct RNA or cDNA) Large numbers of samples	Very large genomes or projects Population-scale human Whole transcriptomes Very large numbers of samples	

Oxford NANOPORE Technologies

https://nanoporetech.com/products

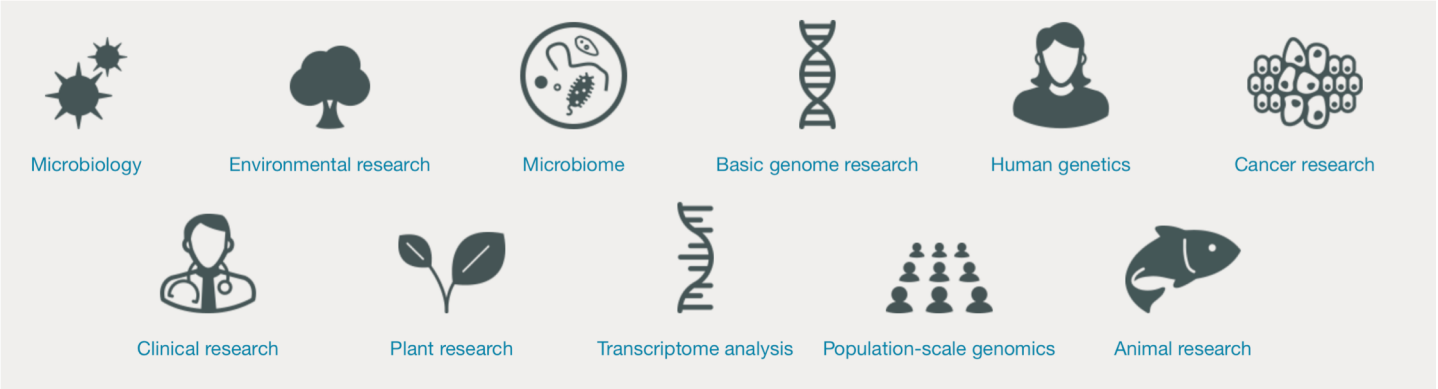
Oxford Nanopore in Mexico



Seems not official available right now.

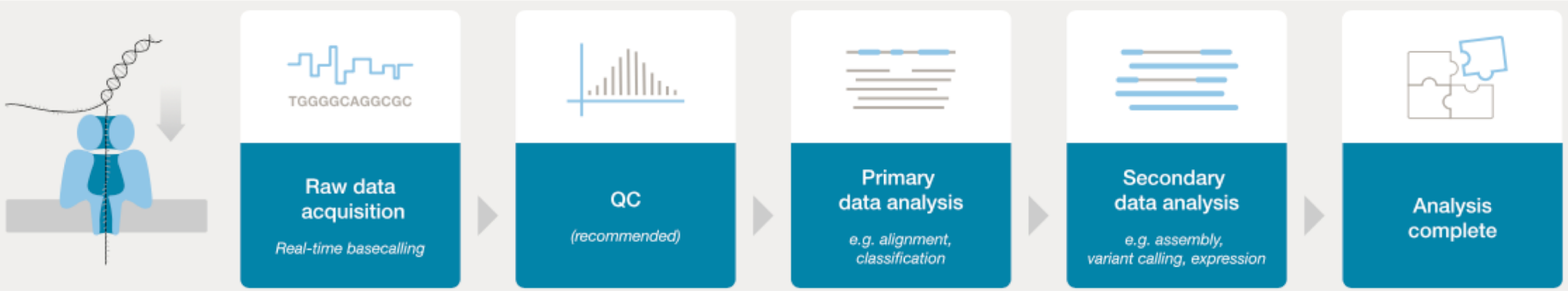
The geographical closest country in sale is Brazil.

Applications



Suitable for applied uses...

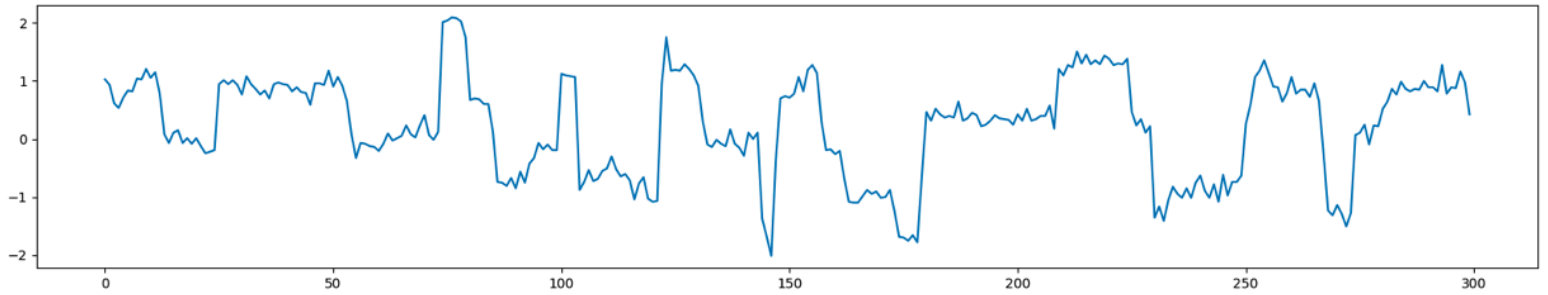
Data Analysis



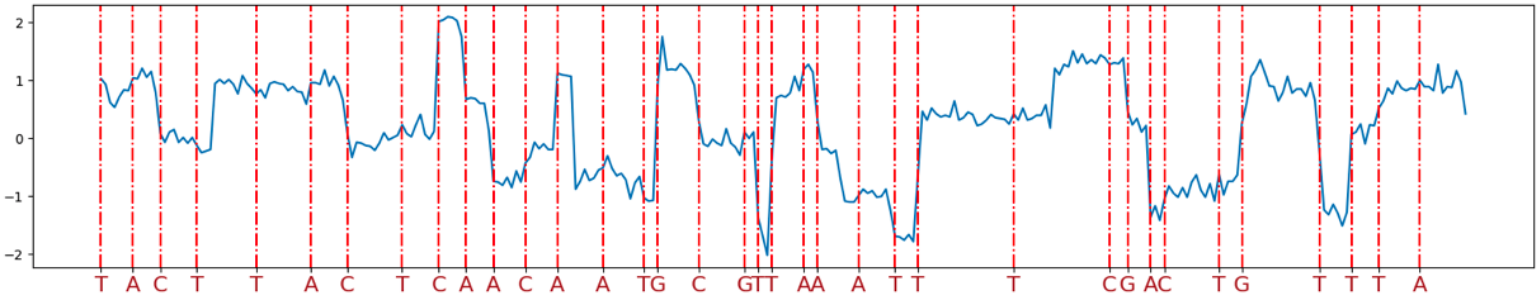
	EPI2ME	Protocol & analysis tutorials	Community developed tools	Custom analysis pipelines
Bioinformatic capability needed	● ○ ○ ○	● ● ○ ○	● ● ● ○	● ● ● ●
How	Use the cloud-based EPI2ME platform for real-time analysis workflows.	Get analysis recommendations and clear tutorials on the use of open-source tools.	Run open-source tools written and developed by the Nanopore Community.	All the data, raw or basecalled, can be used in custom analysis pipelines written by the user for specific applications.

2. Basecalling methods

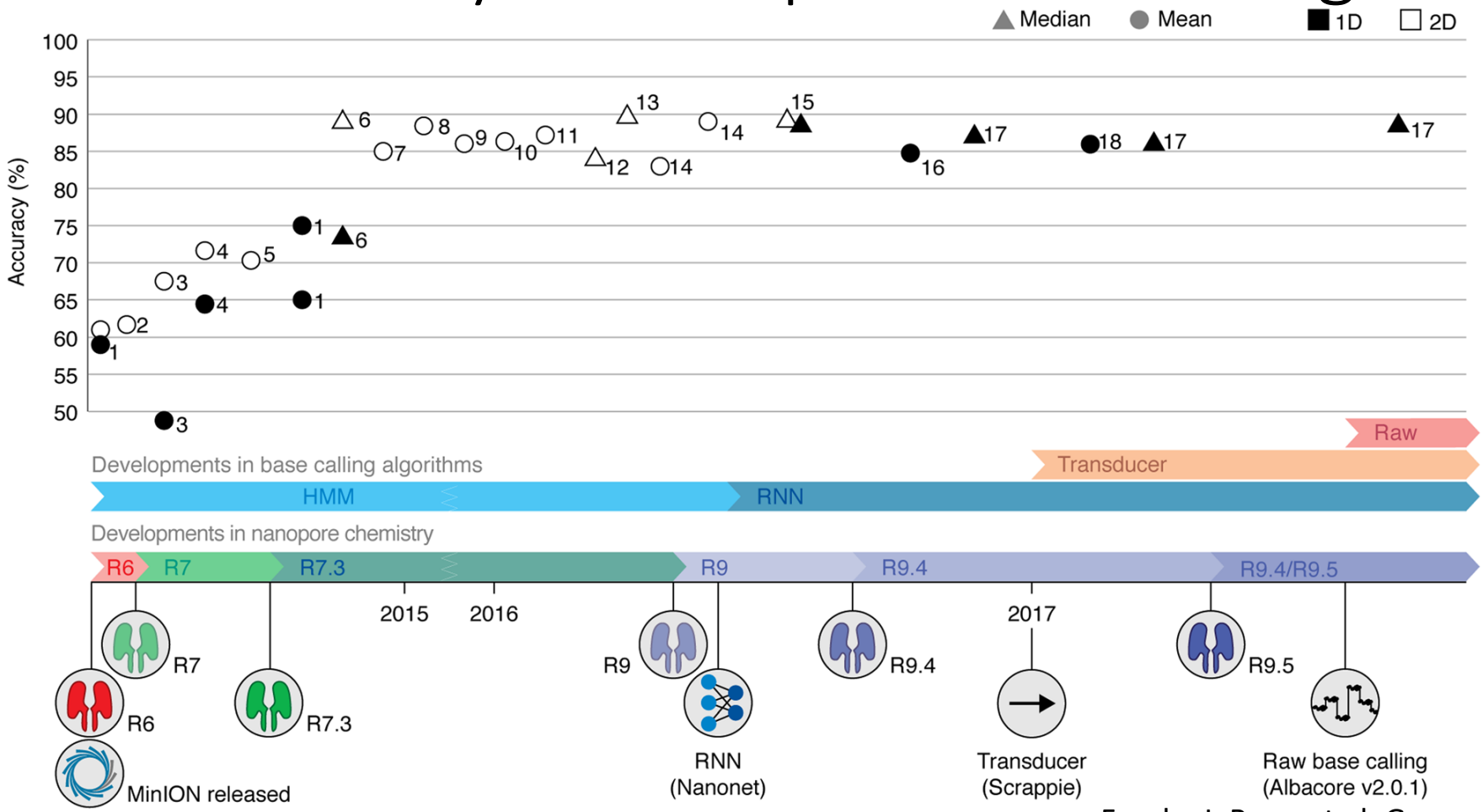
Basecalling



TACTTACTCAACAATGCGTTAAATTTCGACTGTTTA



A brief history of nanopore base-calling:



Franka J. Rang et al. Genome biology 2018

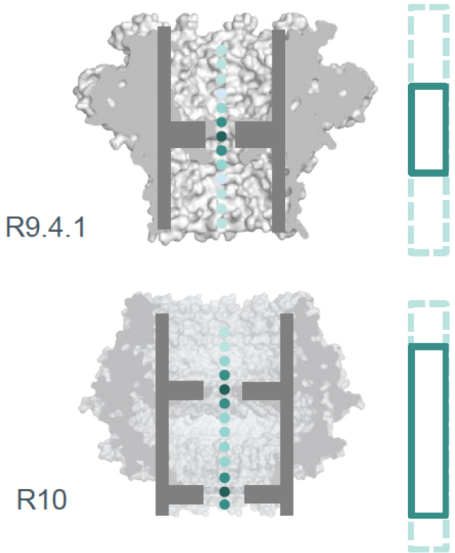
R10 chemistry

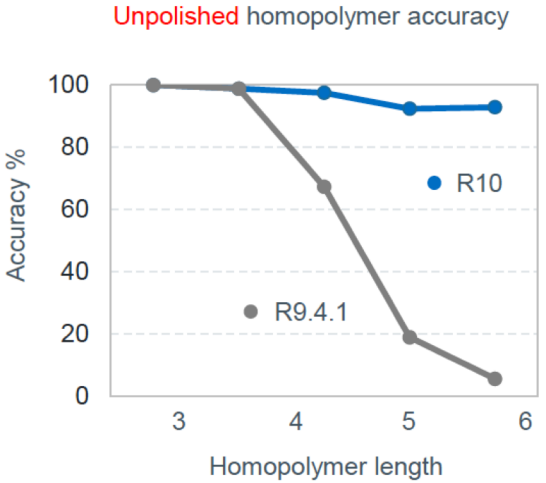
CONSENSUS SEQUENCING ACCURACY

New chemistry for improved accuracy – R10 **Just start shipping, July 8th**

New pore accurately calls homopolymers

- A pore with a longer or multiple “readers” has more bases dominating the signal
- Longer homopolymers are “seen” by the pore and can be decoded with high accuracy

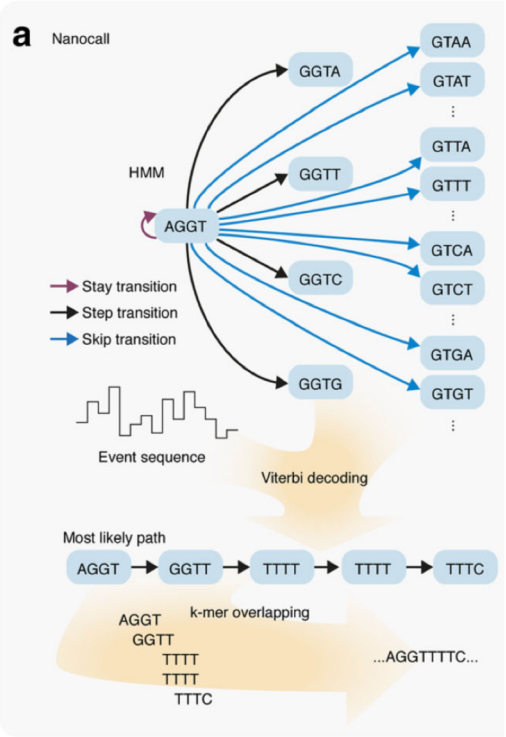


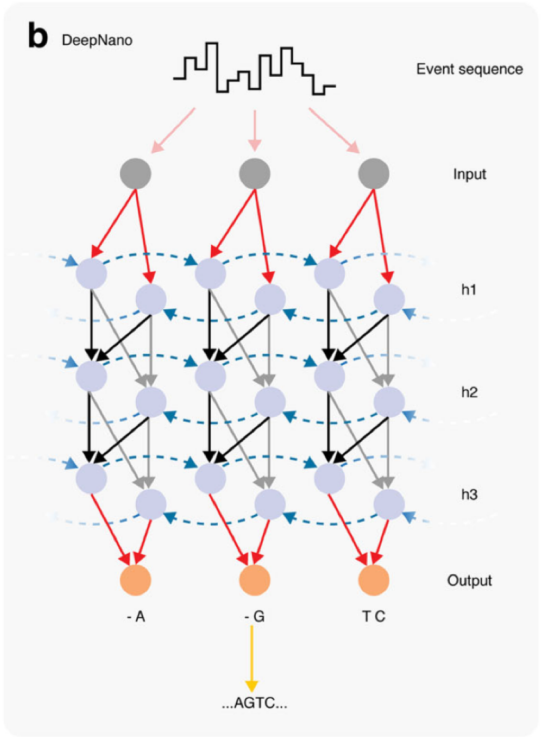


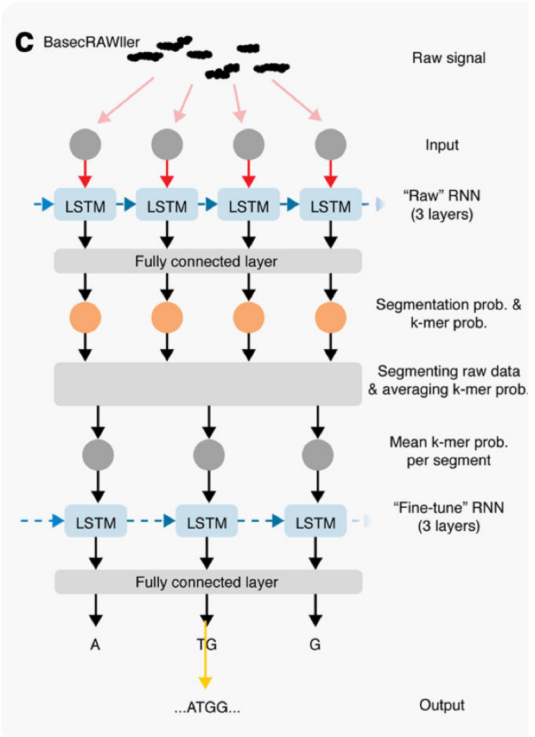
Oxford NANOPORE Technologies

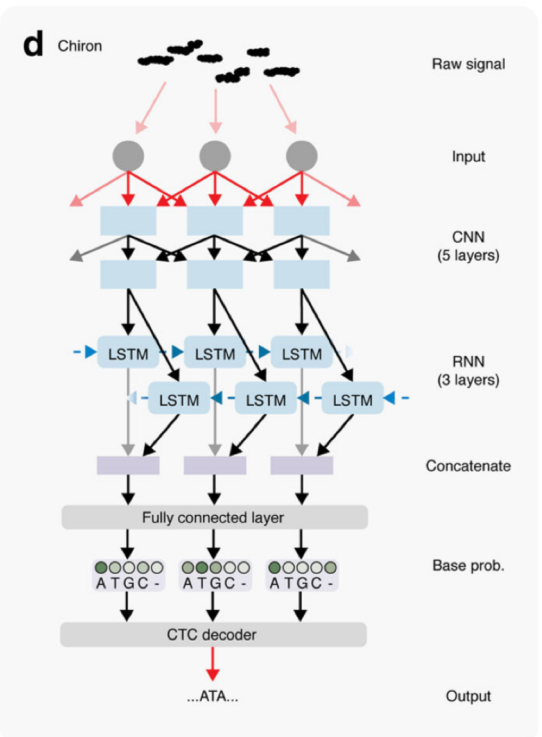
Existed methods of base-calling

Segmentation first	Segmentation later/ implicit segmentation



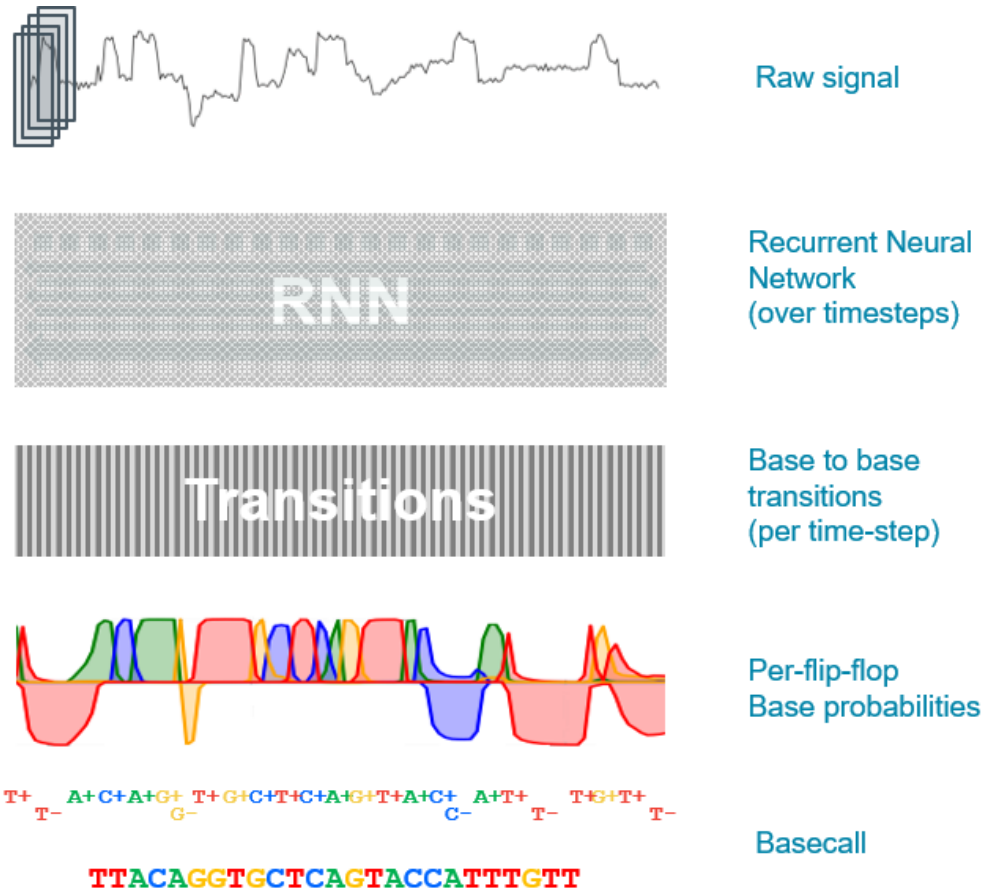






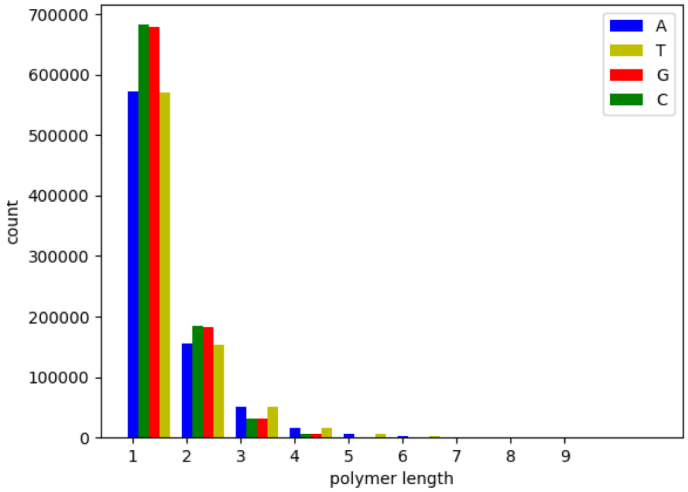
Franka J. Rang et al. Genome biology 2018

ONT Guppy (V3.2.2)

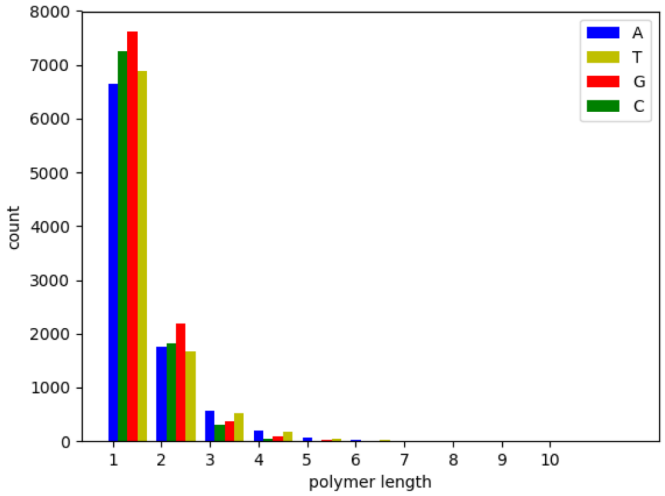


Homo-polymer
(Distributions from real genome)

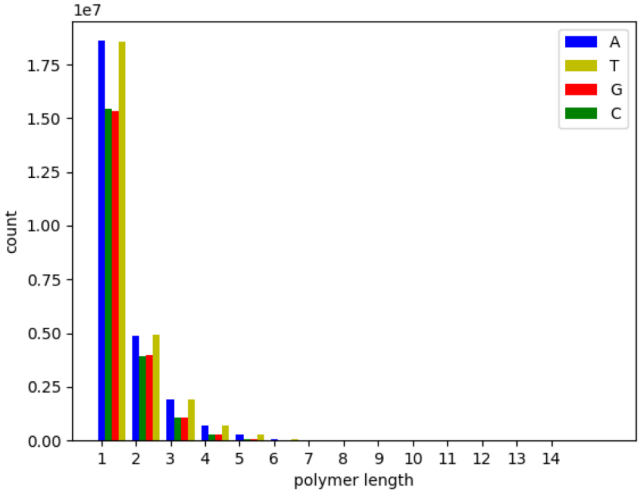
- Var different from species



E.coli

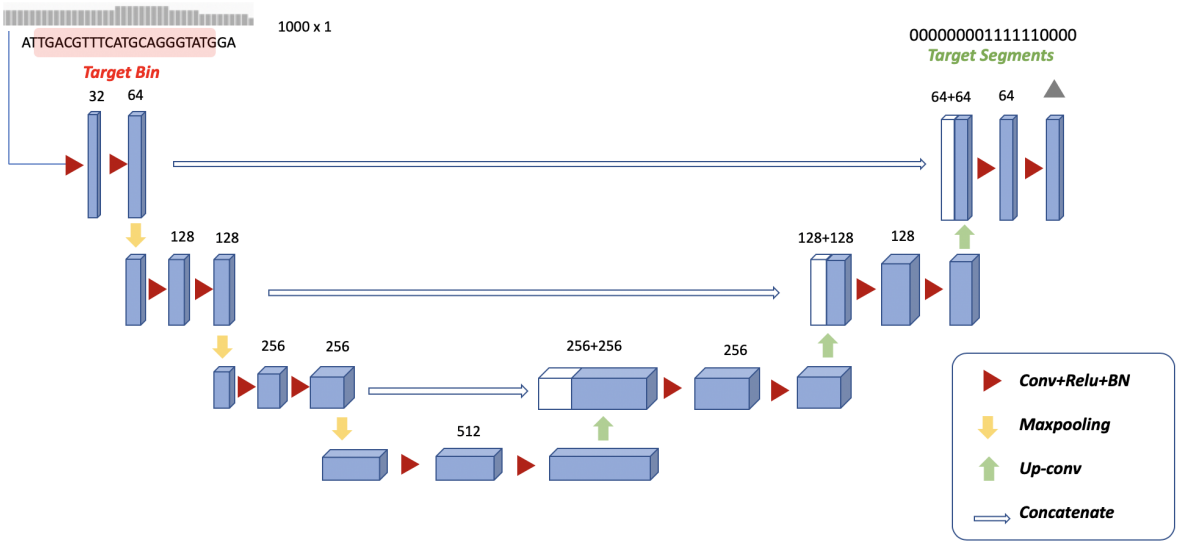


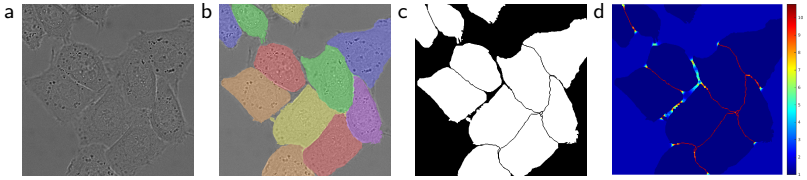
Phage



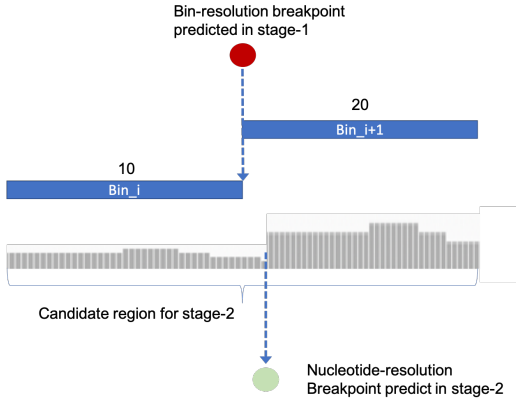
Human, Chr11

U-net model



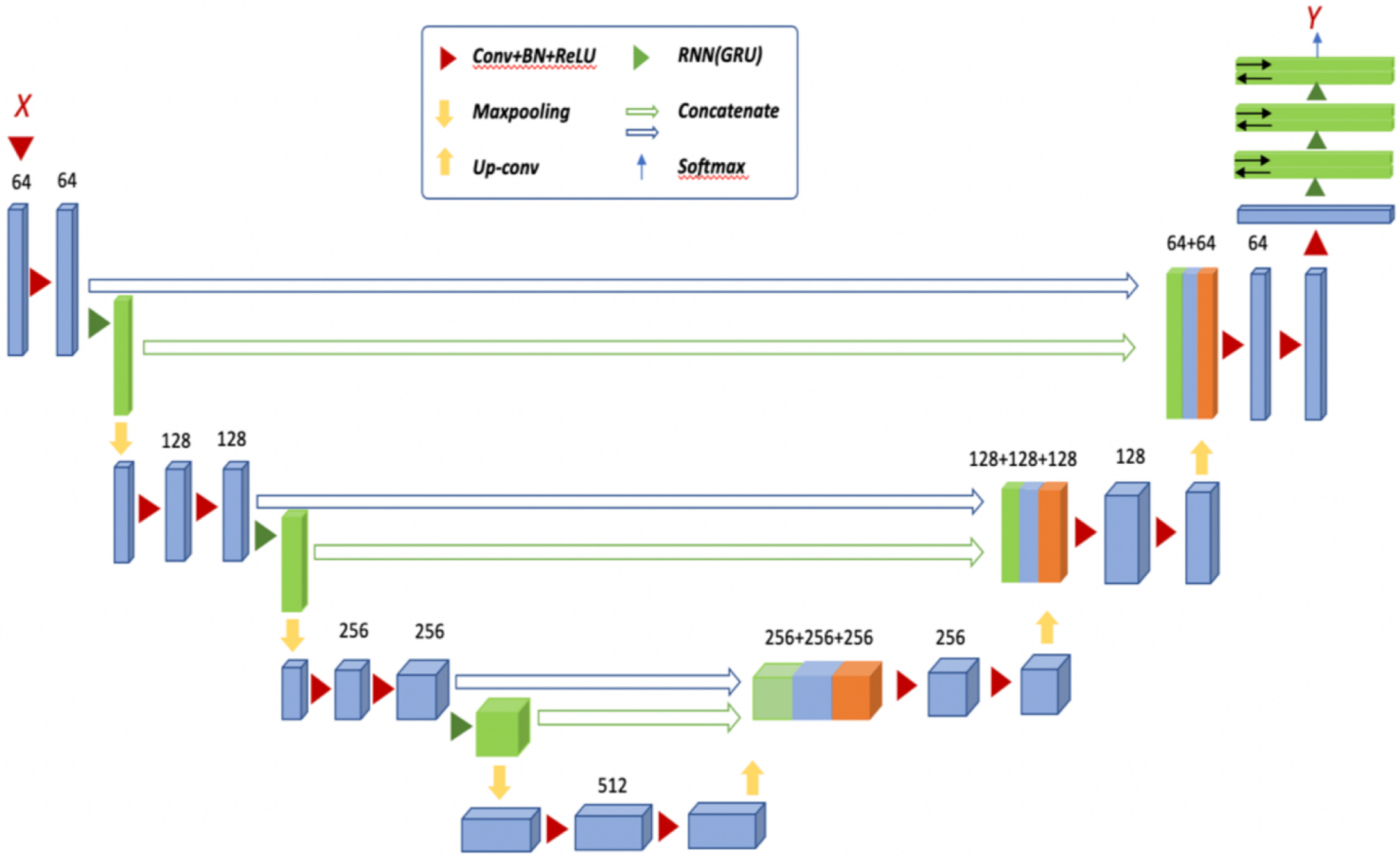


(Olaf Ronneberger et. al. arXiv 2015)

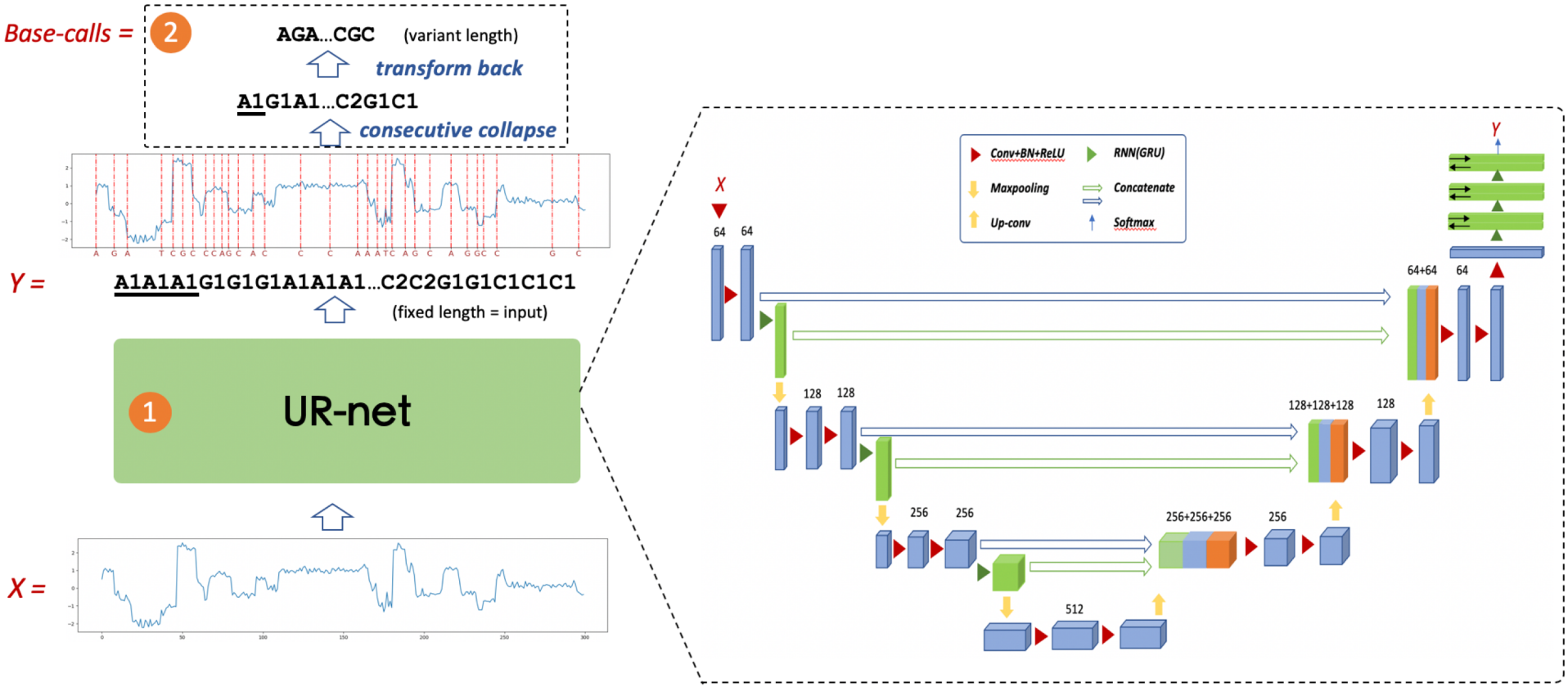


(Zhang et. al. BioArxiv 2019)

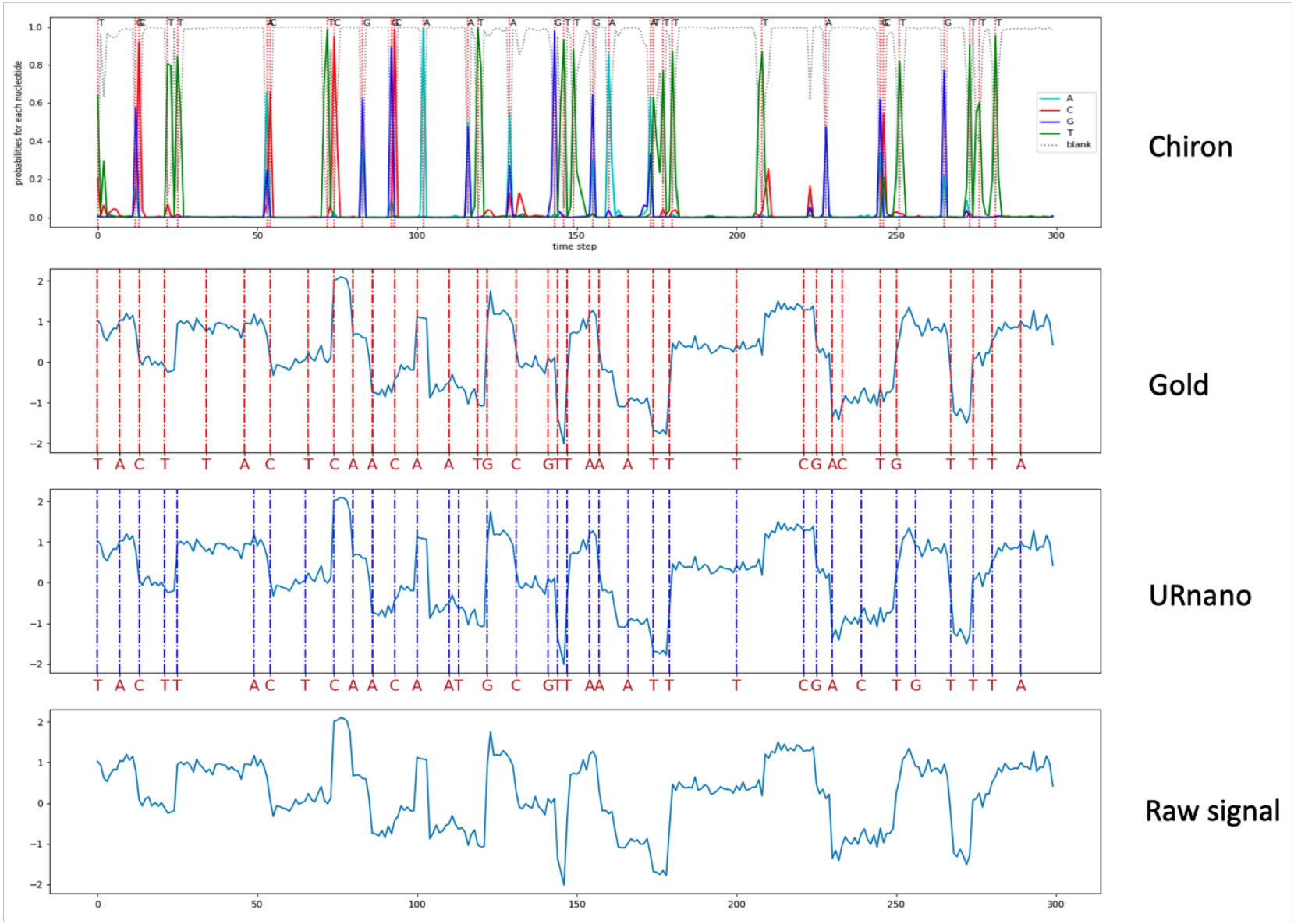
Enhancement with sequential context modeling: UR-net



URnano



Zhang et al., 2019 BMC Bioinformatics, accepted


probabilities for each nucleotide
time step
A
C
G
T
blank
Chiron
Gold
URnano
Raw signal

Incremental evaluation of different network structures

Network structure	**Mean**	**Std**
Unet	0.3528	0.2448
3GRU	0.2808	0.1631
Unet+3GRU	0.1800	0.1296
UR-net	**0.1665**	0.1329

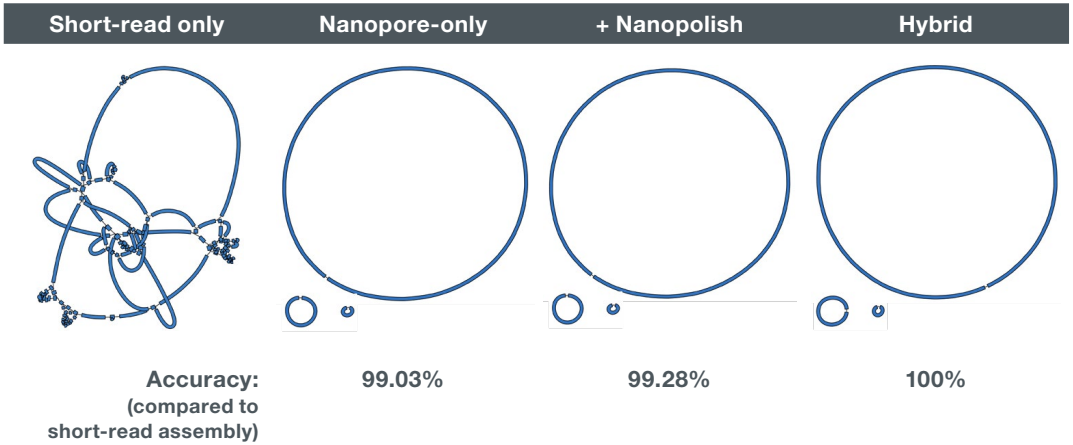
Read accuracy

Species	basecaller	Deletion	Insertion	Mismatch	Read Identity	unaligned	Read Accuracy
E. coli	Chiron	0.0692	0.0465	0.0600	0.8709	7/2000	0.8243
	URnano	0.0584	0.0533	0.0407	0.9010	8/2000	0.8476
	Guppy_taiyaki	0.0585	0.0343	0.0436	0.8978	4/1998	0.8636
λ-phage	Chiron	0.0799	0.0467	0.0641	0.8559	9/2000	0.8093
	URnano	0.0662	0.0455	0.0363	0.8975	10/2000	0.852
	Guppy_taiyaki	0.0655	0.0397	0.0481	0.8864	6/2000	0.8467

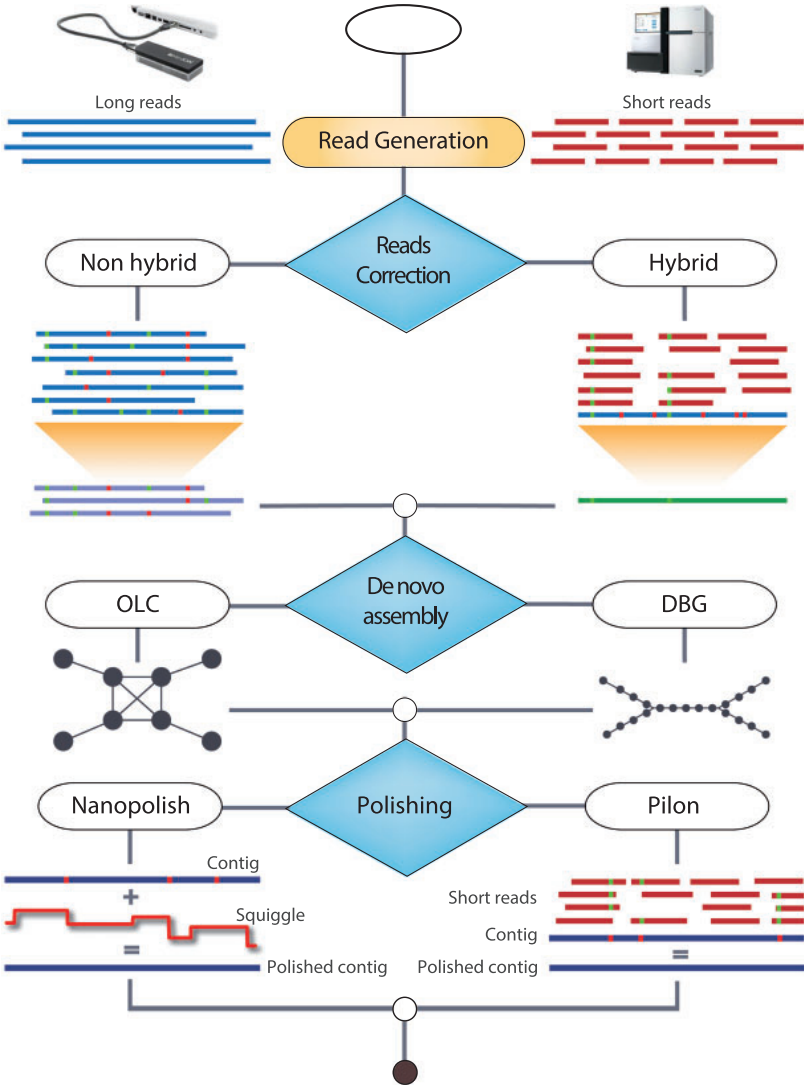
Species	basecaller	Deletion	Insertion	Mismatch	Read Identity	unaligned	Read Accuracy
Human	Chiron	0.0983	0.0687	0.0866	0.8151	385/1000	0.7464
	URnano	0.0954	0.0796	0.0734	0.8312	390/1000	0.7516
	Guppy_taiyaki	0.0822	0.0748	0.0756	0.8422	284/932	0.7674

3.Assembly

De novo assembly



Nanopore white paper

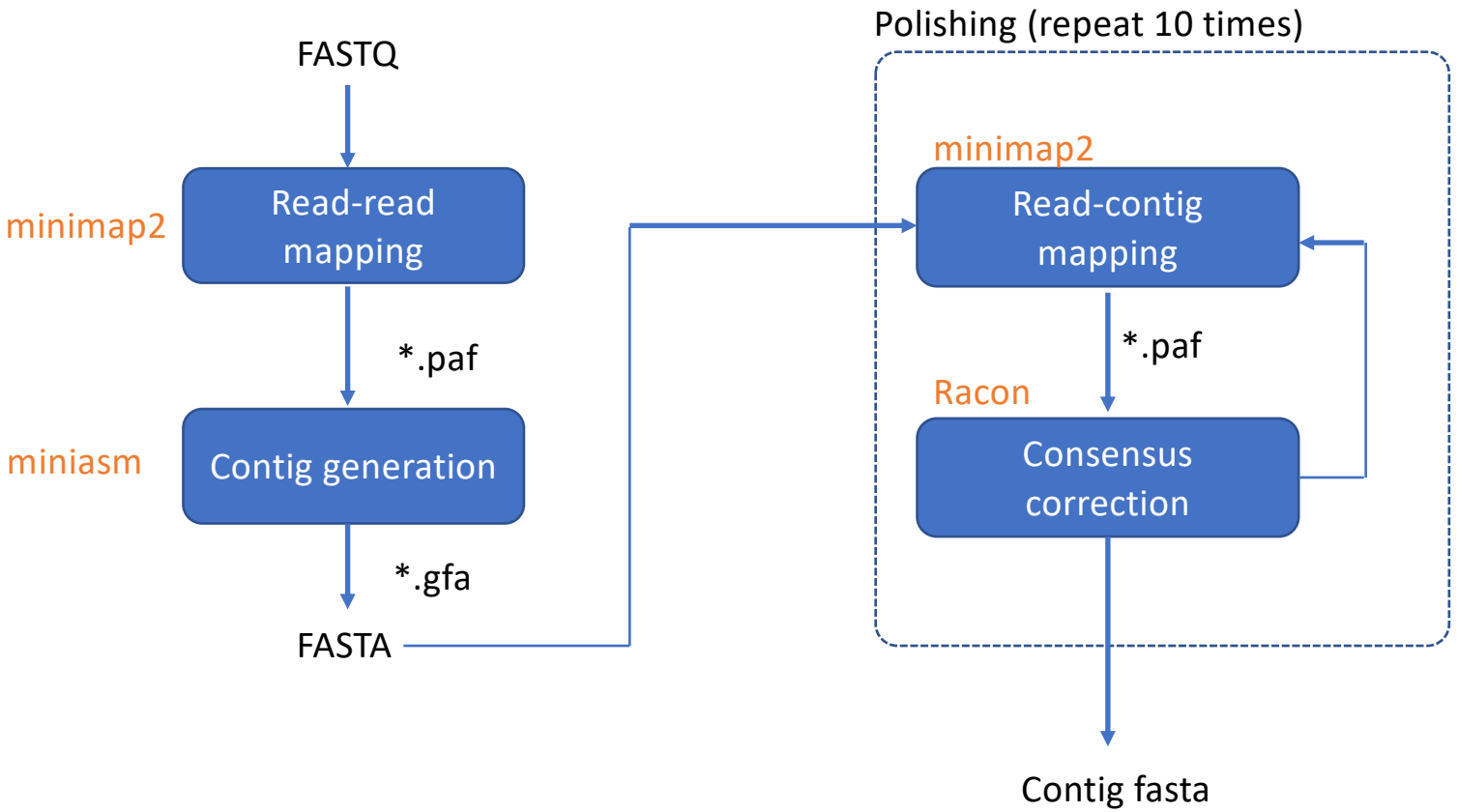


Magi et al. Briefings in Bioinformatics 2017

Overall Process of de-novel Assembling

1. Read-Read Mapping
   - Mapping reads to each other to find overlaps between them
2. Contig generation
   - Using consensus of multiple overlaps to generate contigs
3. Read-Contig Mapping
   - Mapping the initial reads back to contigs
4. Consensus correction (Polishing)
   - taking the consensus of all mapped reads to a specific contig region to remove possible errors in contigs

Workflow of read assembly



1.Read - Read Mapping

- Main drawback of ONT sequencing : High error rates (5-15%)
- Read-read mapping before assembly is necessary to have meaningful overlaps
- **Minimap2** offers read-read mapping for finding overlaps between long reads
  - Uses k-mer minimizers (e.g. k=12)

Minimap2 -x ava-ont -k12 -w5 reads.fastq reads.fastq > read-overlaps.paf

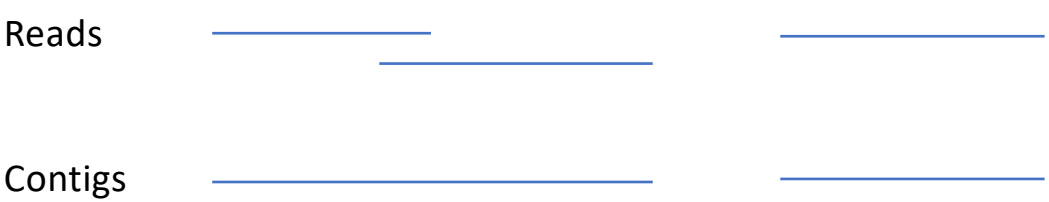
PAF format

- Each line contains the start - end indexes for overlaps between 2 reads

```
Singularity nanopore:~/output/assembly> cat lambda.paf | head -10
f79f7821-2a95-4368-9ef7-4e813b1f41d2    6739    53      6644    +       fe559c88-7255-4173-8f6f-9e578071601c    17884   10400   17011   3392    6741    0
A:S     cm:i:608        s1:i:3362       dv:f:0.0823     rl:i:1184
f79f7821-2a95-4368-9ef7-4e813b1f41d2    6739    53      6644    -       f96b452f-119c-4a59-b321-5ee4ee87c201    10446   882     7517    3099    6748    0
A:S     cm:i:543        s1:i:3068       dv:f:0.0899     rl:i:1184
f79f7821-2a95-4368-9ef7-4e813b1f41d2    6739    53      6644    +       fe2dab60-20e0-4b98-9954-dc287e810453    8444    941     7572    3070    6751    0
A:S     cm:i:544        s1:i:3035       dv:f:0.0897     rl:i:1184
f79f7821-2a95-4368-9ef7-4e813b1f41d2    6739    28      6644    +       fc2be91e-073e-4d60-a920-5202f87e4c7e    10399   2875    9518    3030    6767    0
A:S     cm:i:537        s1:i:3002       dv:f:0.0909     rl:i:1184
f79f7821-2a95-4368-9ef7-4e813b1f41d2    6739    28      6644    +       fc804f36-5a3f-43e2-9cf8-d4ee2e940818    12718   5208    11837   2998    6771    0
A:S     cm:i:504        s1:i:2964       dv:f:0.0951     rl:i:1184
f79f7821-2a95-4368-9ef7-4e813b1f41d2    6739    53      6644    +       ff05b1b9-7aaa-4512-b05f-c568c9dc9dd2    9791    2343    8925    2958    6720    0
A:S     cm:i:493        s1:i:2927       dv:f:0.0963     rl:i:1184
f79f7821-2a95-4368-9ef7-4e813b1f41d2    6739    53      6644    -       fcf9d1b4-882b-4adc-ba6b-f388f77a76b8    7608    867     7470    2949    6744    0
A:S     cm:i:490        s1:i:2915       dv:f:0.0967     rl:i:1184
f79f7821-2a95-4368-9ef7-4e813b1f41d2    6739    28      6644    +       f7bed5f4-87c0-46cd-8504-7444ec5e09c7    9826    2375    8985    2923    6743    0
A:S     cm:i:518        s1:i:2893       dv:f:0.0933     rl:i:1184
f79f7821-2a95-4368-9ef7-4e813b1f41d2    6739    28      6623    -       faa0e2f3-1dba-4846-b910-500566e3480e    8958    926     7589    2861    6785    0
A:S     cm:i:471        s1:i:2820       dv:f:0.0996     rl:i:1184
f79f7821-2a95-4368-9ef7-4e813b1f41d2    6739    28      6644    +       fc3c44f0-4de1-4b0e-ae24-d1755db7c2f8    10519   2988    9635    2843    6777    0
A:S     cm:i:479        s1:i:2807       dv:f:0.0985     rl:i:1184
```

2. Contig generation

- Contig : A contiguous sequence of nucleotides formed by taking the consensus of overlapping reads
- Contain more errors and does not overlap nicely




```
miniasm -f  reads   reads_overlap.pfa  >  contig.gfa

# extract contig fastq file
awk '$1 ~/S/ {print ">"$2"\n"$3}’  contig.gfa >  contigs.fasta
```

Read-Contig Mapping

- Similar to read-read mapping, but change the target to the contigs

```
minimap2   contigs.fasta   reads.fastq >  read_contigs.paf
```

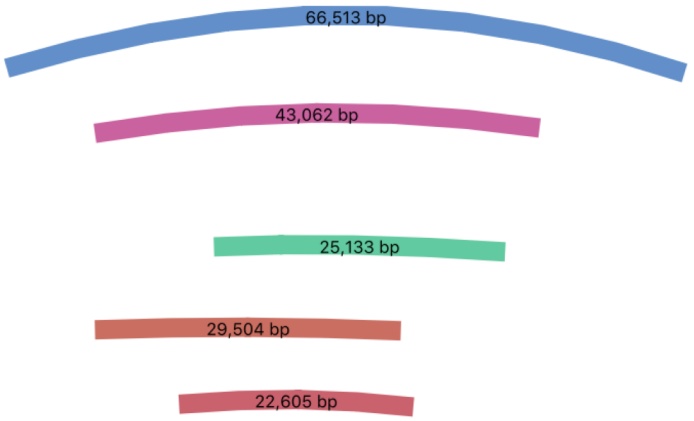
Polishing

- Repeat read-contig alignment and consensus correction for contigs
- Each round returns better contigs which in turn results in better overlap with initial reads

Visualization Using Bandage

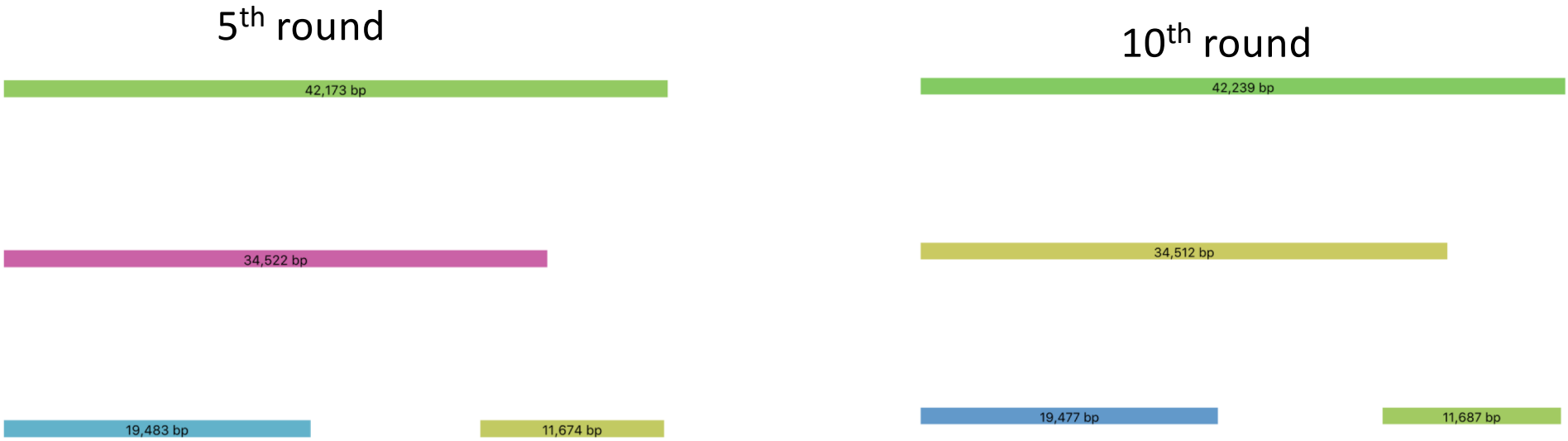
Only high-quality contigs are taken

Contigs formed



Visualization Bandage 2

Polishing reduces error rate and generates longer contigs



Assembly evaluation metrics

- Assembly identity and relative length

Assembly identity (AI) and relative length (RL). We assembled genomes using the results of each basecaller. Assembly identity and relative length are calculated by taking the mean of individual accuracy rates and relative lengths for each shredded contig, respectively.

$$AI = \frac{1}{N}\sum_{i=1}^{N} RA_i \qquad RL = \frac{1}{N}\sum_{i=1}^{N} \frac{L_{pred_i}}{L_{ref_i}}$$

where N is the total number of aligned parts, L_{pred_i} is the length of the assembled i^{th} basecall and L_{ref} is the length of the reference genome.

Assembly evaluation with different basecallers

Table 3: Evaluation with assembly on the test set. URnano with the new assembly method outperforms both models in all species when trained on the same dataset in assembly identity rate.

Dataset	basecaller	Assembly Identity	Relative Length
E. coli	Chiron	97.1318	99.0607
	URnano	**98.3256**	99.8947
	Guppy_taiyaki	98.2989	99.4885
λ-phage	Chiron	92.2302	98.9411
	URnano	**99.6308**	99.8902
	Guppy_taiyaki	99.3900	99.5190
Human	Chiron	92.0913	99.9763
	URnano	**93.9488**	100.8250
	Guppy_taiyaki	93.4319	100.9401

4. Structural variant detection

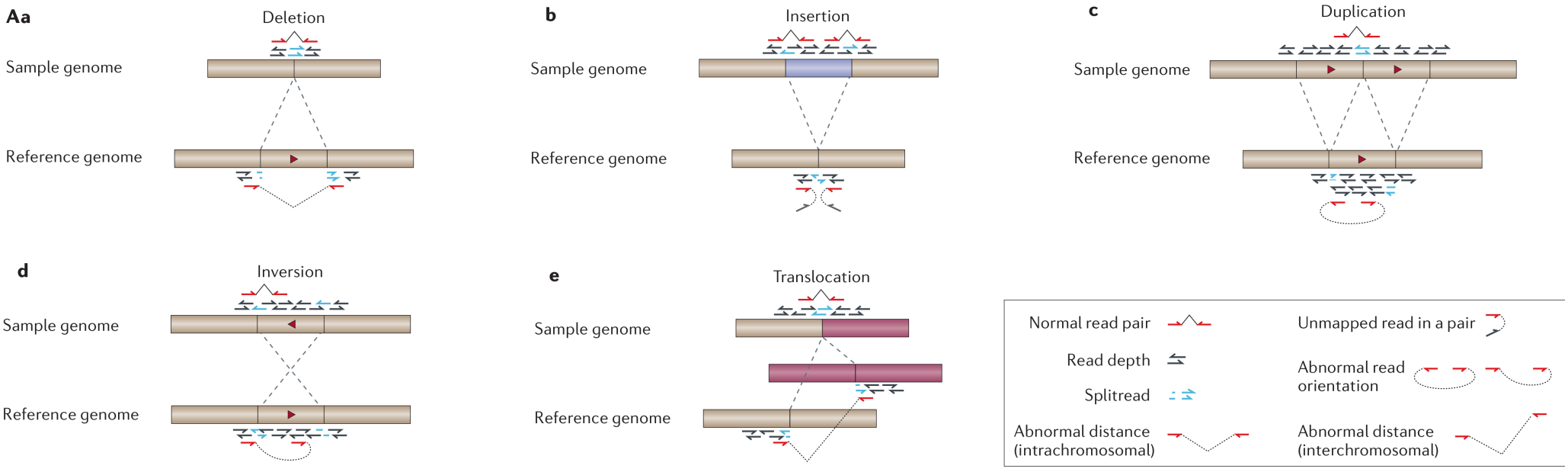
Genomic variation

- Single-nucleotide variants (SNP)
- Small indels (< 50bp)
- Structural variant (> =50 bp)
  - Insertion
  - Deletions
  - Duplications
  - Inversions
  - …
- Copy number variant
  - Deletion
  - Amplification

Advantages and challenge of Long reads

- Advantage:
  - Repetitive regions
  - Nested SVs
- Challenge
  - Higher error rate => alignment

Different types of Structural Variations



(J Weischenfeldt et. al. Nature Review 2013)

nature methods
ARTICLES
https://doi.org/10.1038/s41592-018-0001-7

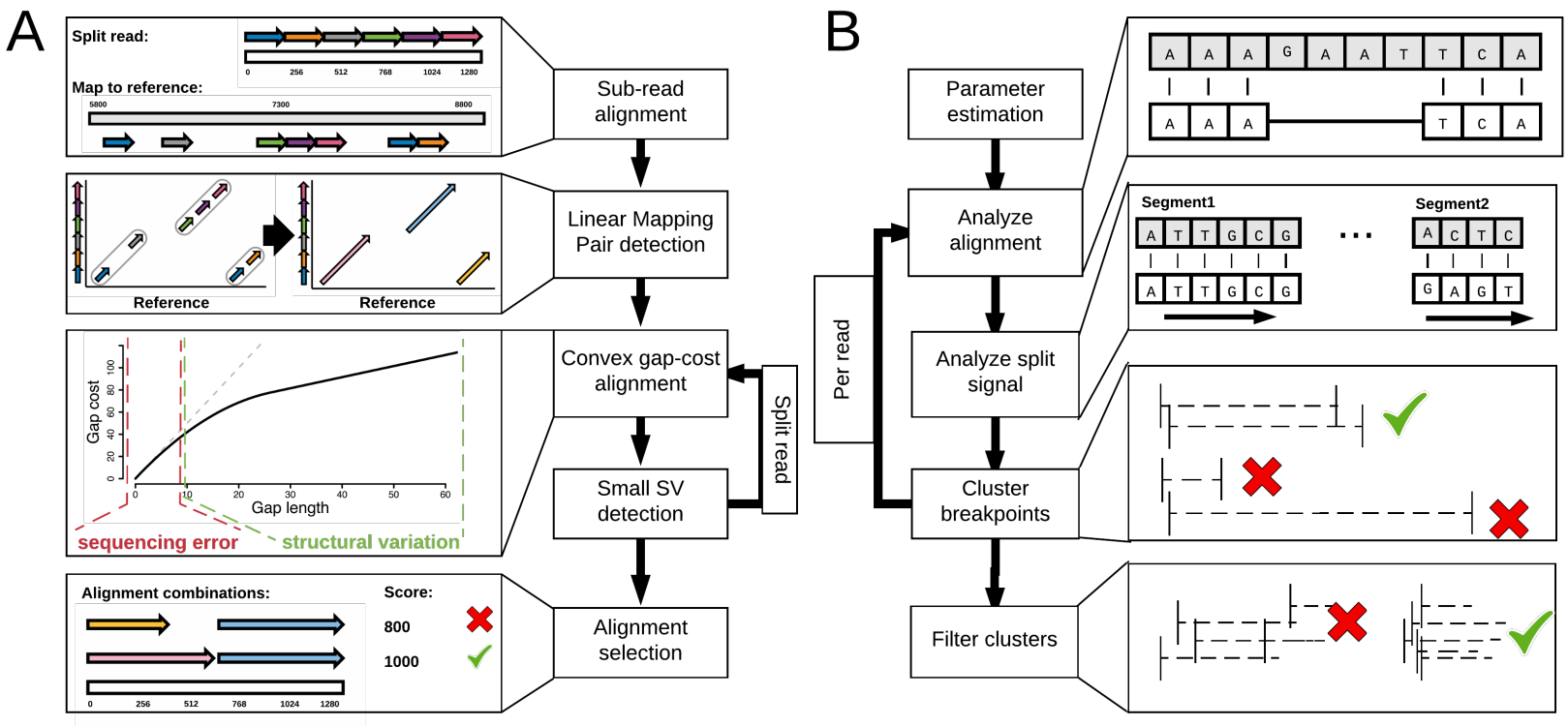


Figure 1: Overview of the main steps implemented in NGMLR (A) and Sniffles (B). For details see Supplementary Sections 1 and 2 for NGMLR and Sniffles, respectively.

Sedlazeck et al. 2018

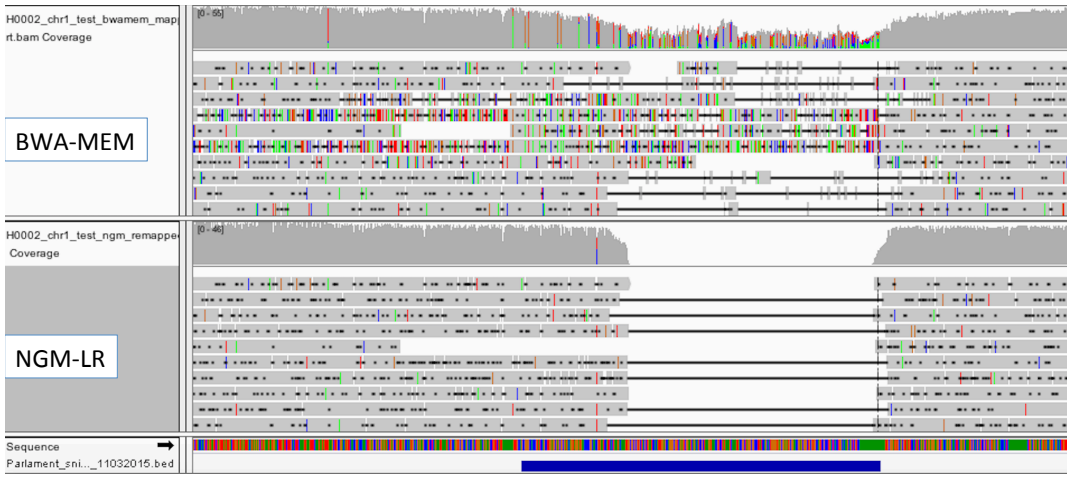
Convex Pairwise Alignment

```
AAAGAATTCA
A-A-A-T-CA
```

vs.

```
AAAGAATTCA
AAA----TCA
```

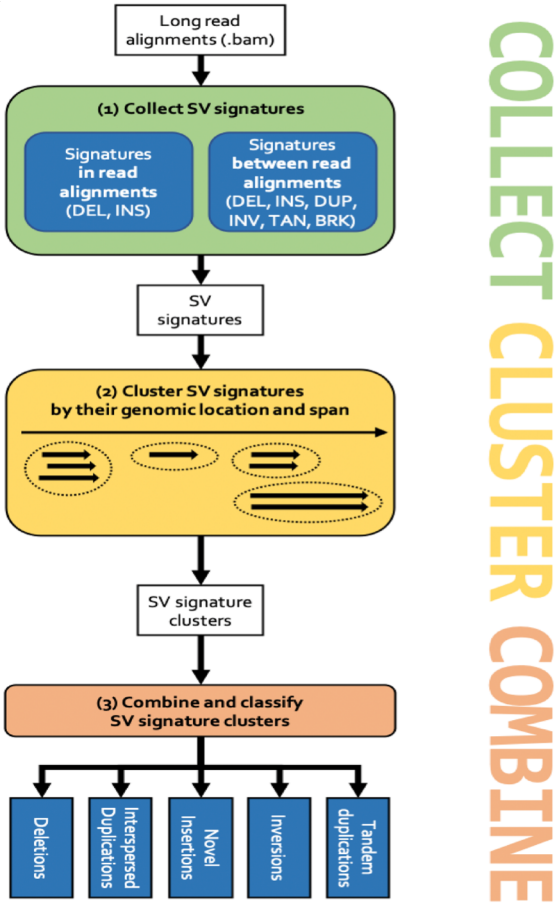
- Gap cost is the same, but not good for downstream SV analysis.
- Considering the SV in the early alignment stage for long reads.



Sniffles for detecting SV

1. Estimate parameters for the underlying data set.
   E.g. distribution and distance between indels and mismatches on the read
2. Search for putative read alignment and segments.
3. Clustering putative SVs
4. Genotyping SV for homozygous or heterozygous SVs
5. Provide clustering SV based on the overlap with the same reads

The SVIM workflow. (1) Signatures for SVs are collected from the input read alignments. SVIM collects them



SVIM does not filter its output but writes out all SV calls and their respective scores

Bioinformatics, Volume 35, Issue 17, 1 September 2019, Pages 2907–2915, https://doi.org/10.1093/bioinformatics/btz041

5. Hands-on section

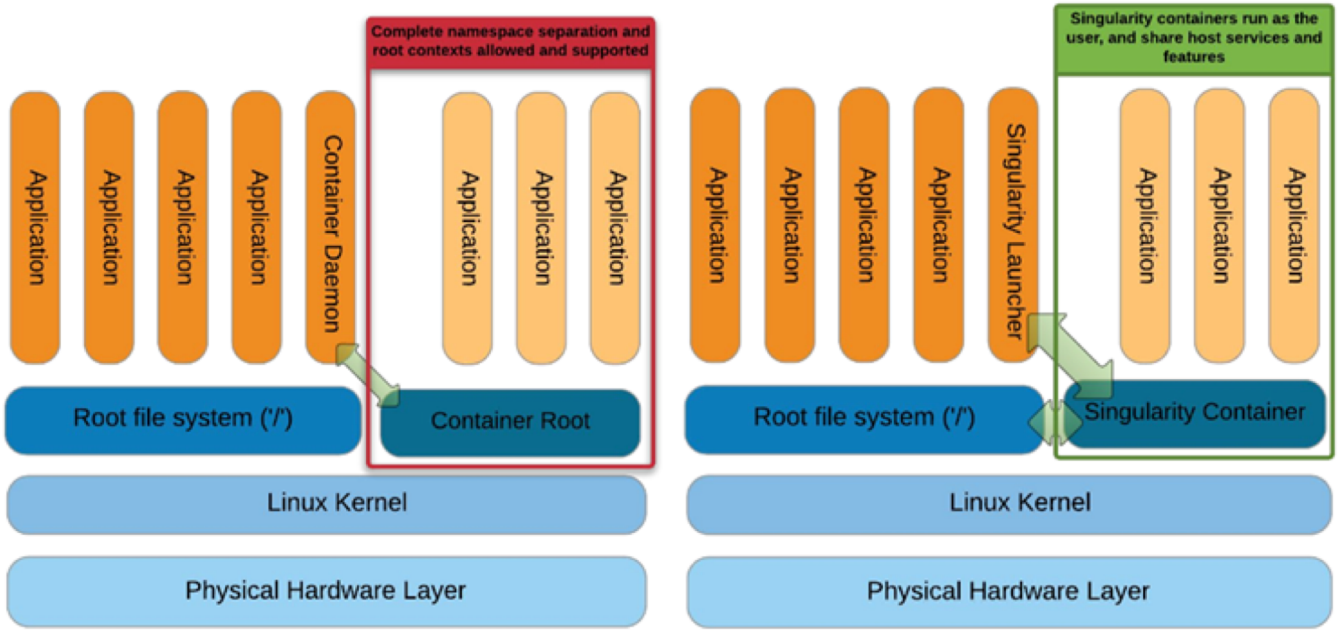
Materials are available on Github.

https://github.com/yaozhong/mexico_workshop_nanopore_hands_on/commits/master

Things to do in general:

1. Set up computing environment with docker/singularity
2. Get input data ready
3. Get output and analysis

Containers (Docker and Singularity)



General Container
eg Docker

HPC Container
Singularity

Optional GPUs available in Shirokane-5

- 80 V100 GPU cards
- Commonly used for:
  - Training deep learning models
  - Acceleration of genome analysis pipeline

Nanopore Data resources

- https://www.plabipd.de/portal/solanum-pennellii
- https://www.ebi.ac.uk/ena/data/view/PRJEB26791
- https://github.com/nanopore-wgs-consortium/CHM13
- https://www.nature.com/articles/nmeth.3444
- https://github.com/nanopore-wgs-consortium/NA12878