

RawBench

A Comprehensive Benchmarking Framework
for Raw Nanopore Signal Analysis Techniques

Furkan Eris

Ulysse McConnell Can Firtina Onur Mutlu

ETH zürich

SAFARI

 UNIVERSITY OF
MARYLAND

Outline

Background

Motivation and Goal

RawBench

Evaluation

Conclusion

Genomic Data Analysis

- Study of genomics through the lens of **growing sequencing data** has shaped groundbreaking advances in



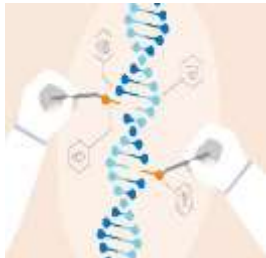
Evolutionary biology



Personalized medicine



Outbreak tracing

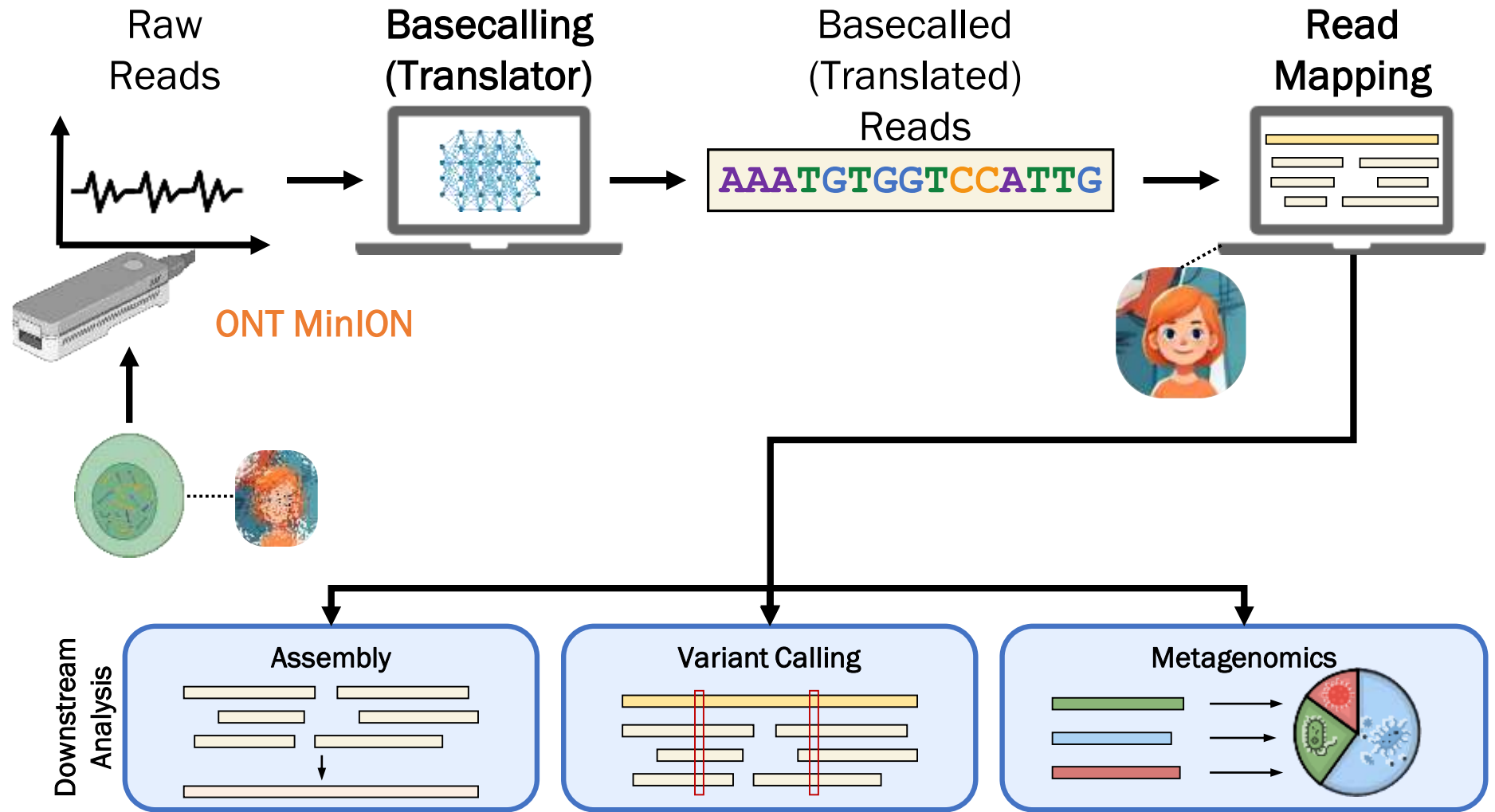


Gene editing



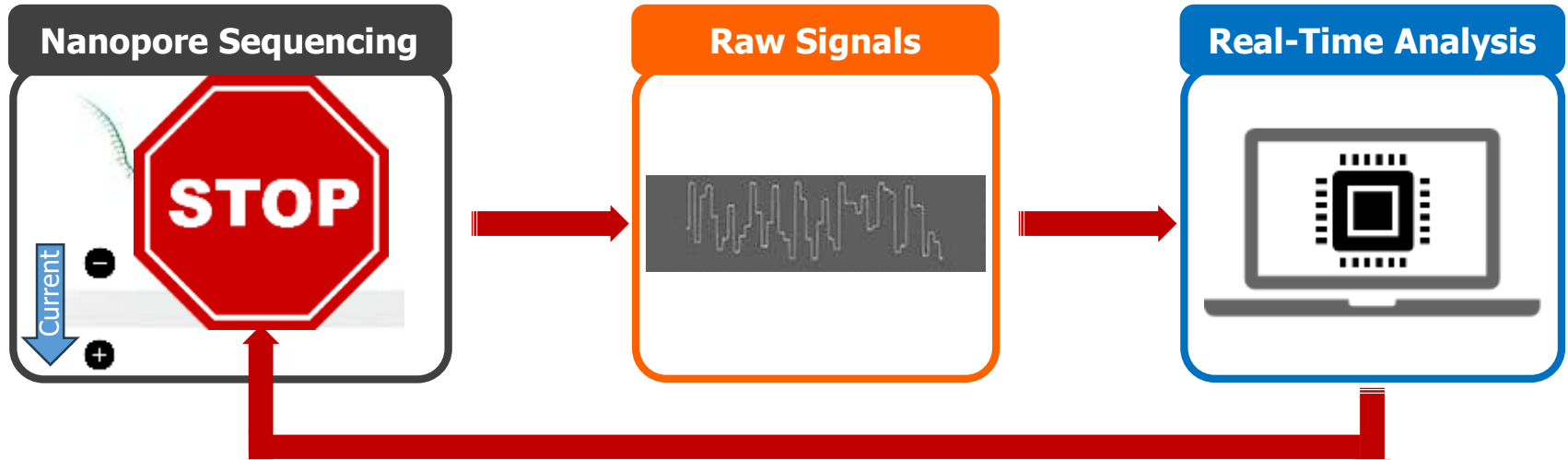
Novel mutation identification

A Common Genome Analysis Pipeline

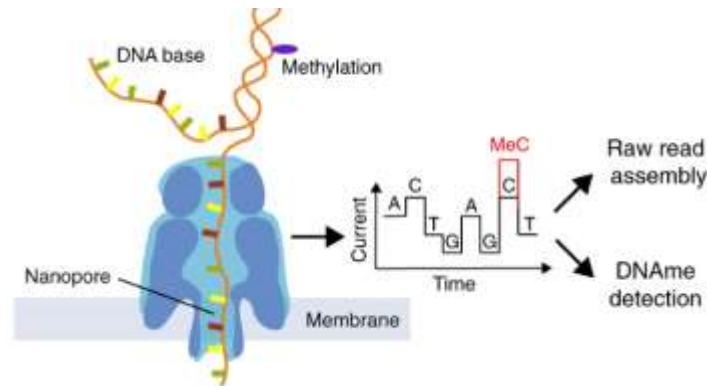


Benefits of Nanopore Sequencing

- Adaptive sampling as a **unique** feature



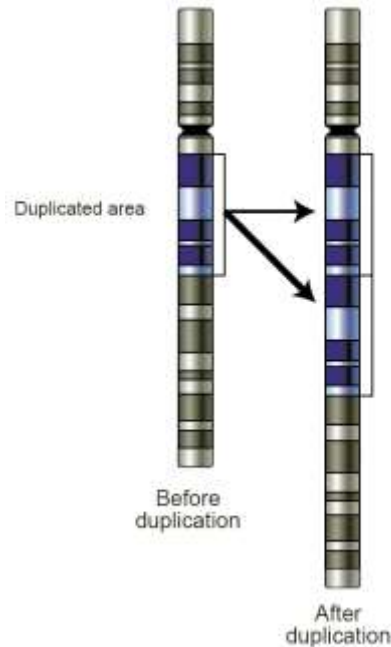
- Raw nanopore signals are **inherently rich**
Methylation detection



Benefits of Nanopore Sequencing

- Ultra-long reads (up to **4 million bases**)

Copy number variant detection

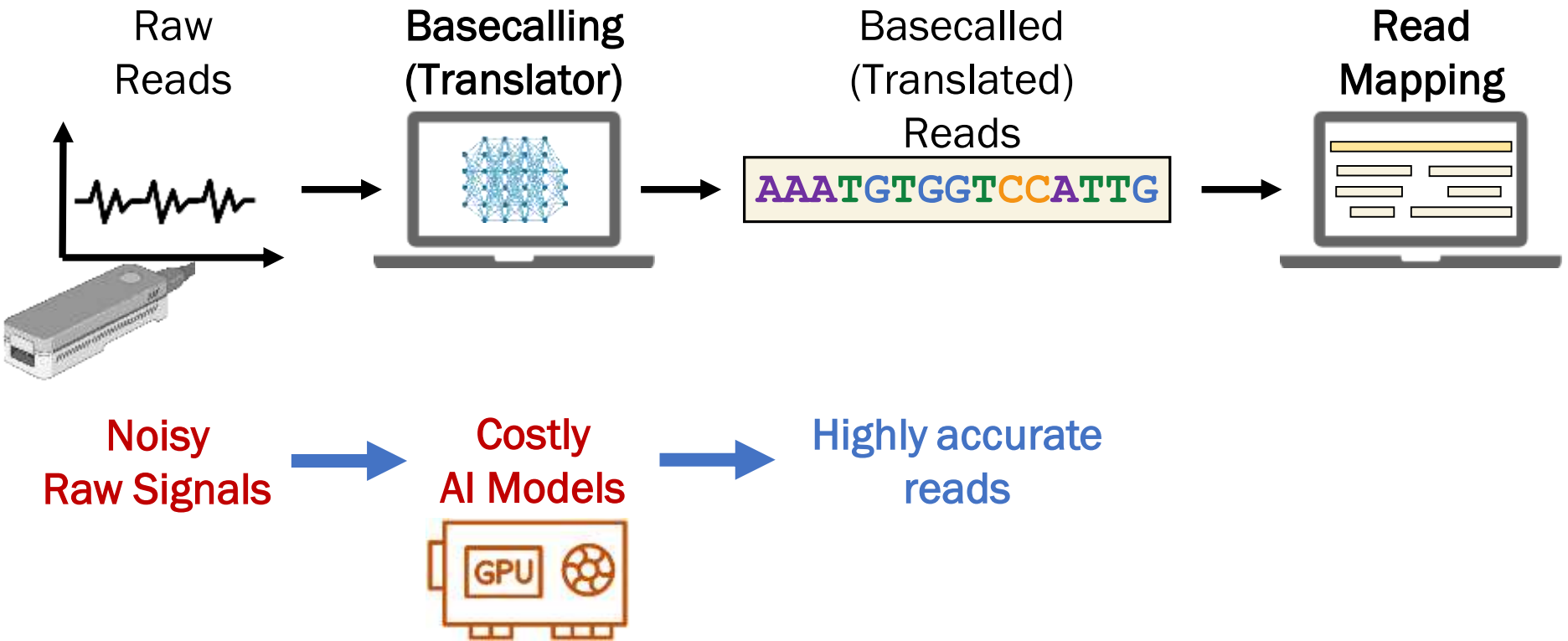


- Portable sequencing (enables **field deployment**)

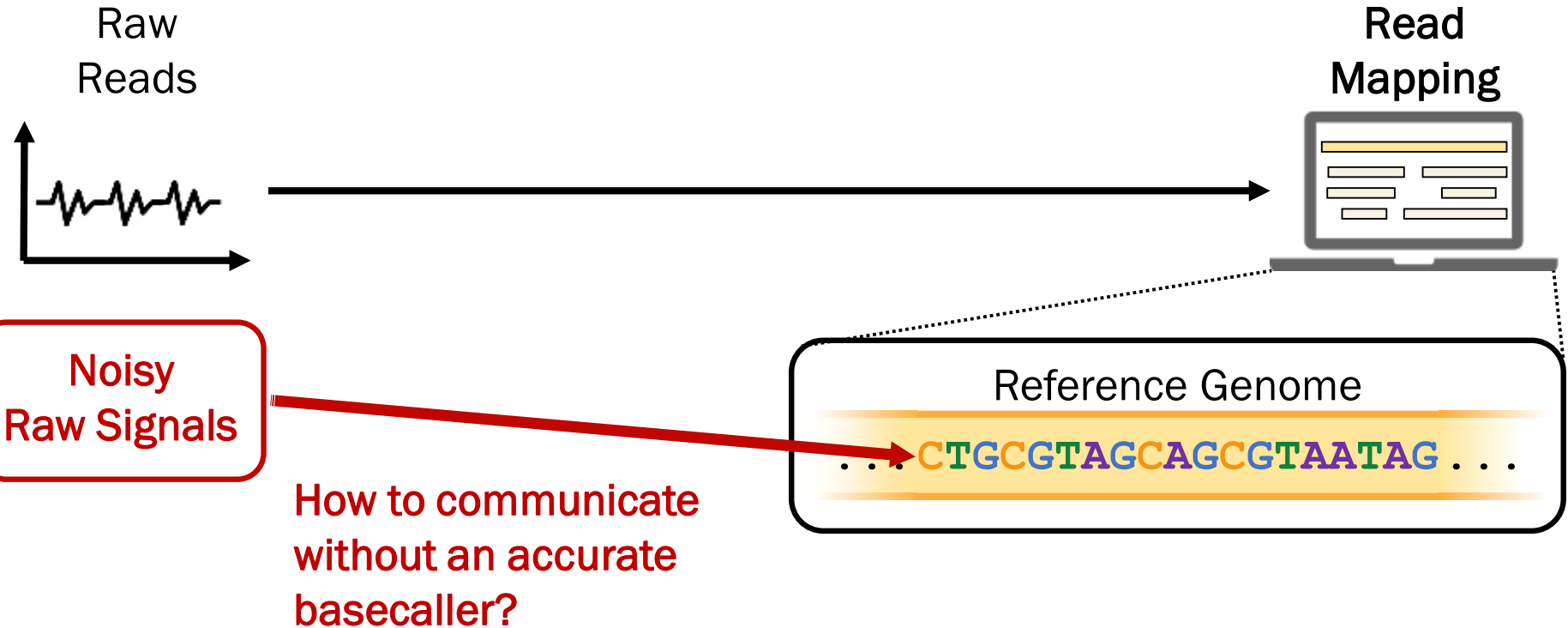
ONT MinION



Basecalling is Accurate yet Costly

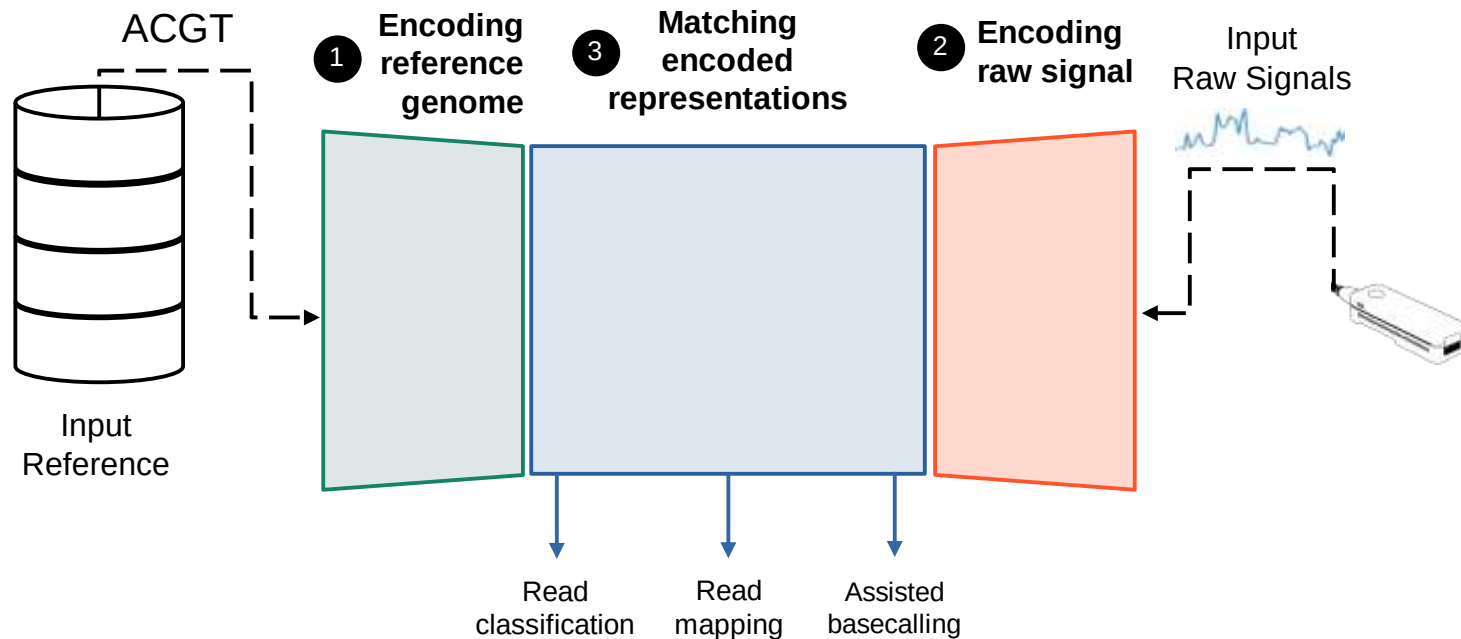


Can We Manage without Basecalling?



Raw Signal Analysis (RSA) Overview

- **First**, the reference genome is encoded into a comparable representation
- **Second**, raw signals are encoded similarly
- **Third**, these representations are matched



Outline

Background

Motivation and Goal

RawBench

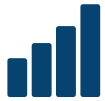
Evaluation

Conclusion

Motivation



Traditional pipelines struggle with increasing real-time requirements
RSA techniques emerged as competitive alternatives



Existing benchmarking frameworks overlook RSA
RSA techniques differ in quality, speed and resource usage



There is a critical need for fair and extensive comparison of emerging RSA techniques



Goal: Compare quality and performance for different RSA techniques
Target different downstream tasks and organism complexities

Problems of Existing Works

Do **not** include raw signal analysis (RSA) tools

Lack the flexibility to incorporate newly developed methods

Lack access to standardized datasets from latest chemistry

Our Goal



*Design a **comprehensive, extensible** and **up-to-date** benchmarking framework for RSA*

Outline

Background

Motivation and Goal

RawBench

Evaluation

Conclusion

RawBench: Comprehensive Benchmarking for RSA

- **First Benchmarking Framework for RSA**
- **Key Idea:** Enable systematic evaluation of existing and future RSA techniques in a flexible, comprehensive and up-to-date framework.
- Holistic design combining **a modular structure for different RSA stages** and **different nanopore sequencing datasets**



Modular RSA stages to increase resolution of RSA benchmarking

- Allows better exploring quality and performance trends



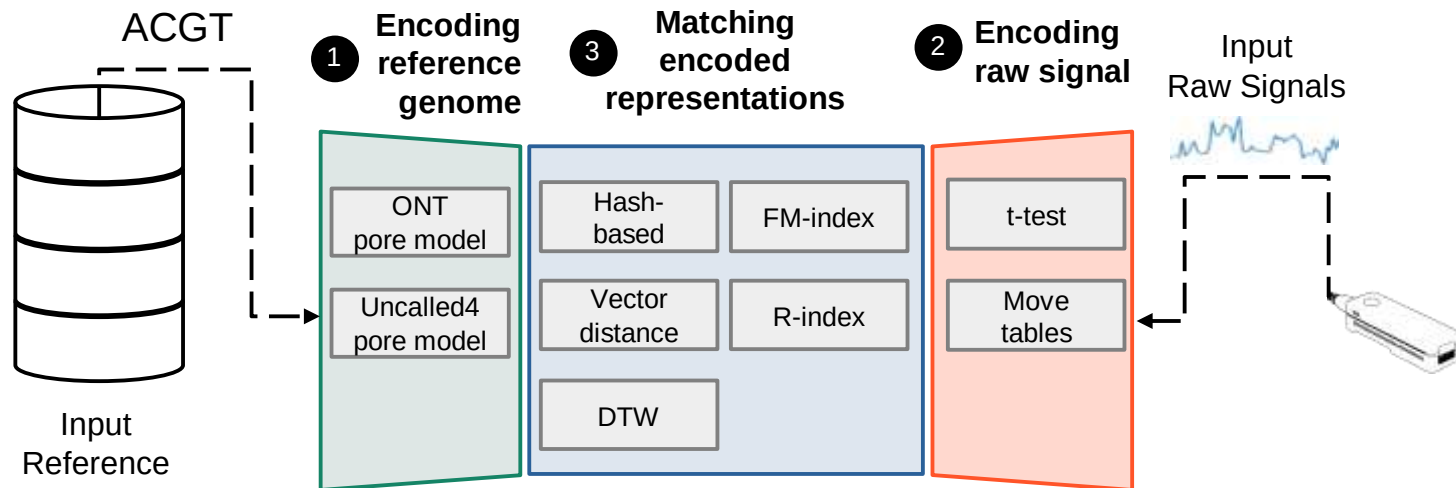
Wide range of datasets for a fair and comprehensive evaluation of RSA techniques on **multiple downstream tasks**

- Covers a spectrum of datasets from **different genome complexities**
- Includes datasets from the **latest nanopore chemistry**

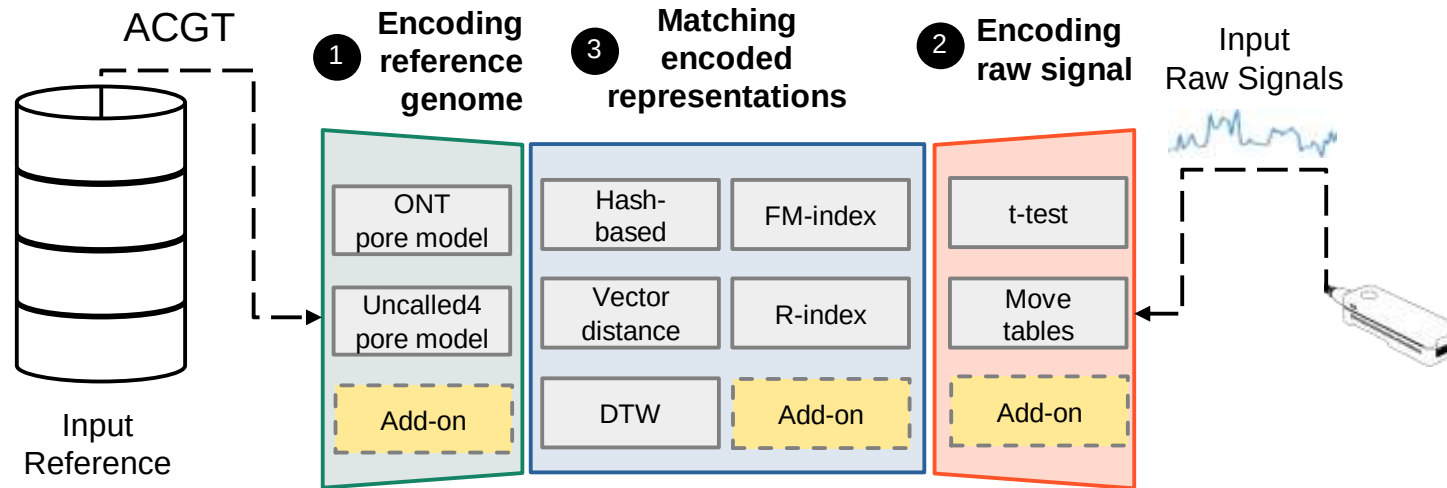
Towards a Modular RawBench

Aim: Break RSA down to different stages for in-depth benchmarking of different RSA techniques

Challenge: Preserve applicability for a wide range of existing (e.g., Sigmap and Uncalled) and future RSA tools



Closer Look at RawBench Pipeline



1 Encoding Reference Genome

- Uses learned pore models by ONT and community

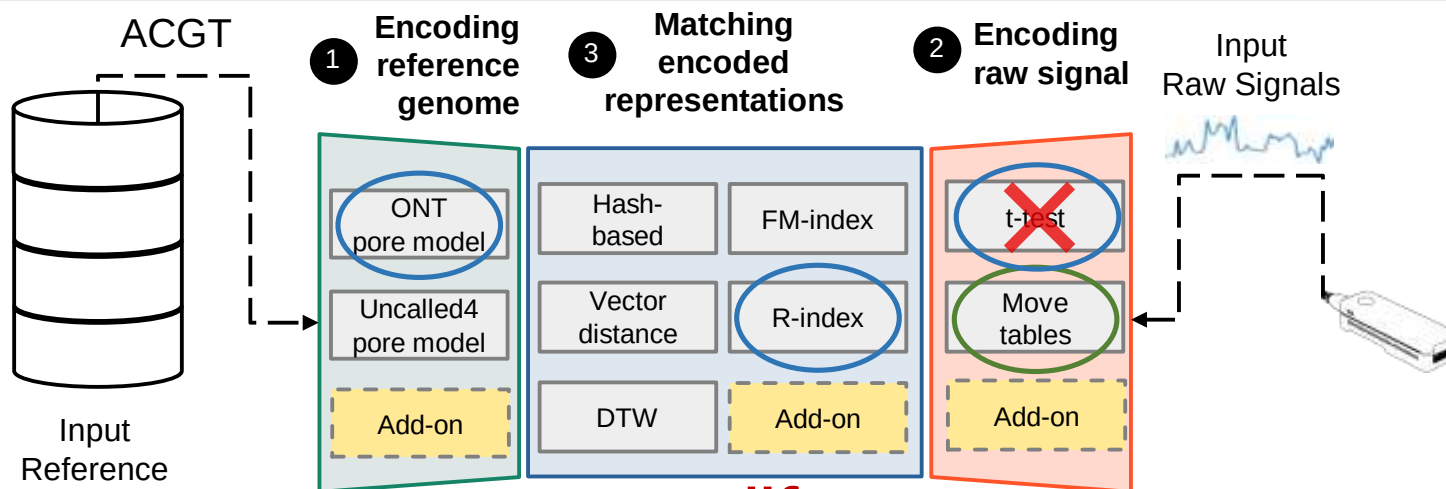
2 Encoding Raw Signal

- Utilizes statistical and ML-based encoding techniques

3 Matching Encoded Representations

- Compares representations that are now encoded into the same space

A RawBench Example



Modify

Sigmoni (Shivakumar et al.)

Performs read classification using raw signals

- 1 ONT pore model
- 2 ~~t-test~~
- 3 R-index



Sigmovie



- 1 ONT pore model
- 2 Move tables
- 3 R-index

RawBench Datasets

Insight: Datasets from different chemistries and genomic complexities is a prerequisite for comprehensive and future-proof analysis.

Dataset	Genome Size (Mbp)	Downstream Task	Nanopore Chemistry
<i>E. coli</i>	4.6	Read Mapping	R10.4
<i>D. melanogaster</i>	143.7	Read Mapping	R10.4
<i>H. sapiens</i>	3,200	Read Mapping	R10.4
<i>Zymo mock</i>	65.4	Read Classification	R9.4



These are all real datasets. We intend to release simulated data for new downstream tasks.

Mapping any RSA Workflow to RawBench

- 1 Break down RSA tool into **three RSA stages**
 - Encoding of both reference genome and raw signal and matching these encoded representations
- 2 Implement RSA stages as C++ modules
 - Each stage mapped to a dedicated **Nextflow** process

**More details on the implementation
can be found in the paper**

- and downstream tasks
- New datasets and tasks can be integrated
 - e.g., simulated data and structural variant calling

Outline

Background

Motivation and Goal

RawBench

Evaluation

Conclusion

Evaluation Methodology

- **Experimental Setup**

- **CPU baseline:** Intel Xeon Gold 6226R @2.90GHz
 - 64 threads for each analysis
- **GPU baseline:** NVIDIA A6000

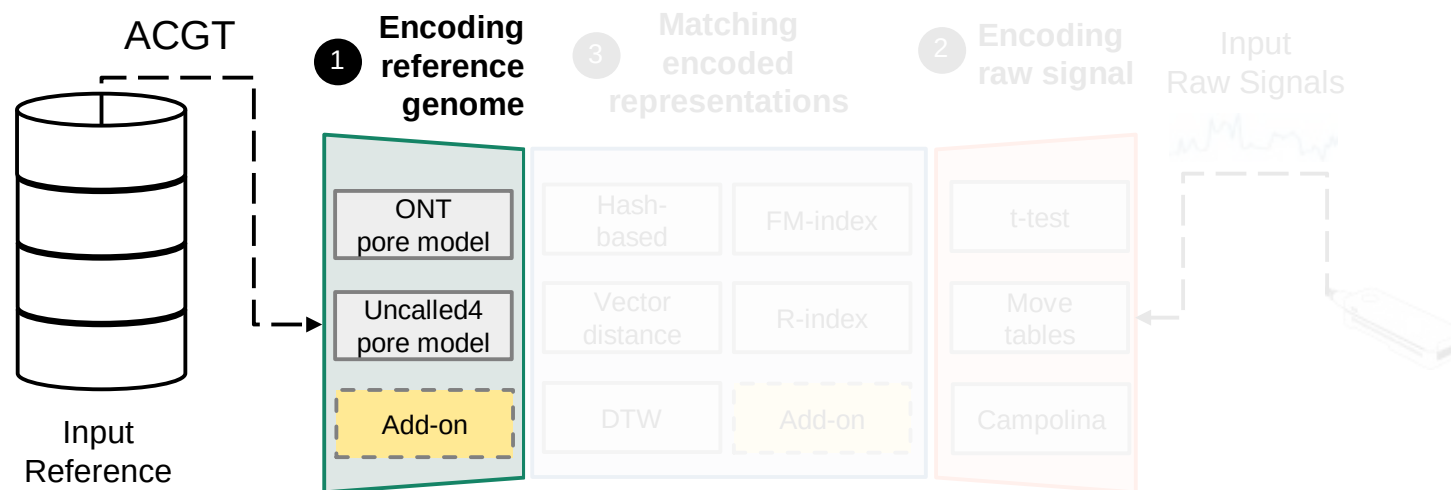
- Currently available **downstream analysis tasks**

- Read mapping
- Read classification
- RSA-assisted basecalling

Evaluation Methodology

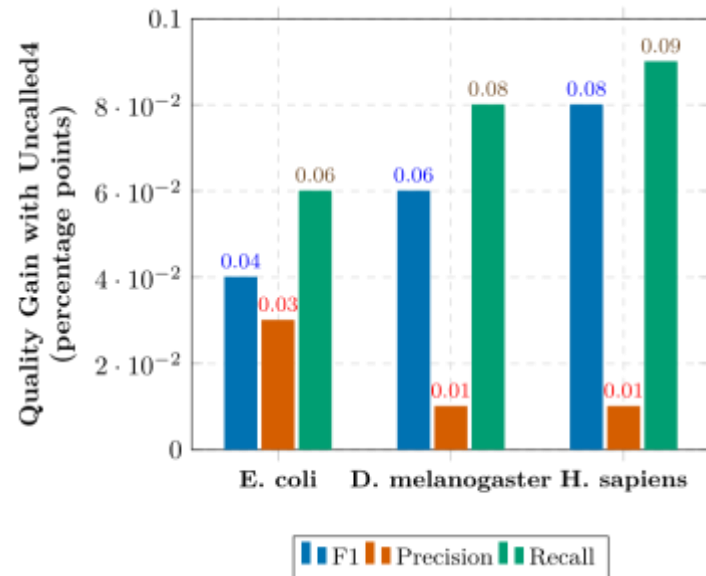
- Evaluation metrics
 - **Performance** (runtime and memory footprint)
 - Coverage statistics
 - **Quality**
 - **Baseline:** Mapping basecalled reads using Dorado SUP + minimap2
 - Precision, recall and F1 scores
- **4 real datasets** with
 - Various **coverage** (0.11x-225x) and
 - **Genome complexities** (bacterial to human genomes)

Encoding Reference Genome



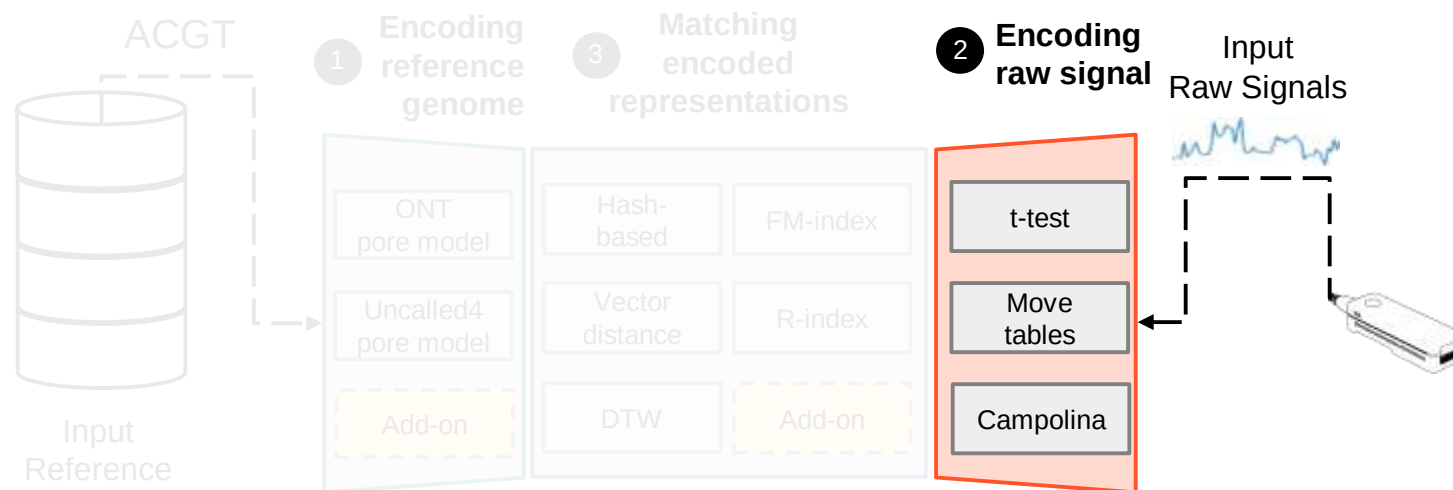
Encoding Reference Genome – Quality

Read Mapping			
Pore Model	F1	Precision	Recall
<i>E. coli</i>			
ONT	0.79	0.88	0.71
Uncalled4	0.83	0.91	0.77
<i>D. melanogaster</i>			
ONT	0.66	0.93	0.51
Uncalled4	0.72	0.94	0.59
<i>H. sapiens</i>			
ONT	0.58	0.86	0.44
Uncalled4	0.66	0.87	0.53



Uncalled4 provides the **best quality** in all metrics, with increasing benefits in recall **for larger genomes**

Encoding Raw Signal



Encoding Raw Signal – Quality

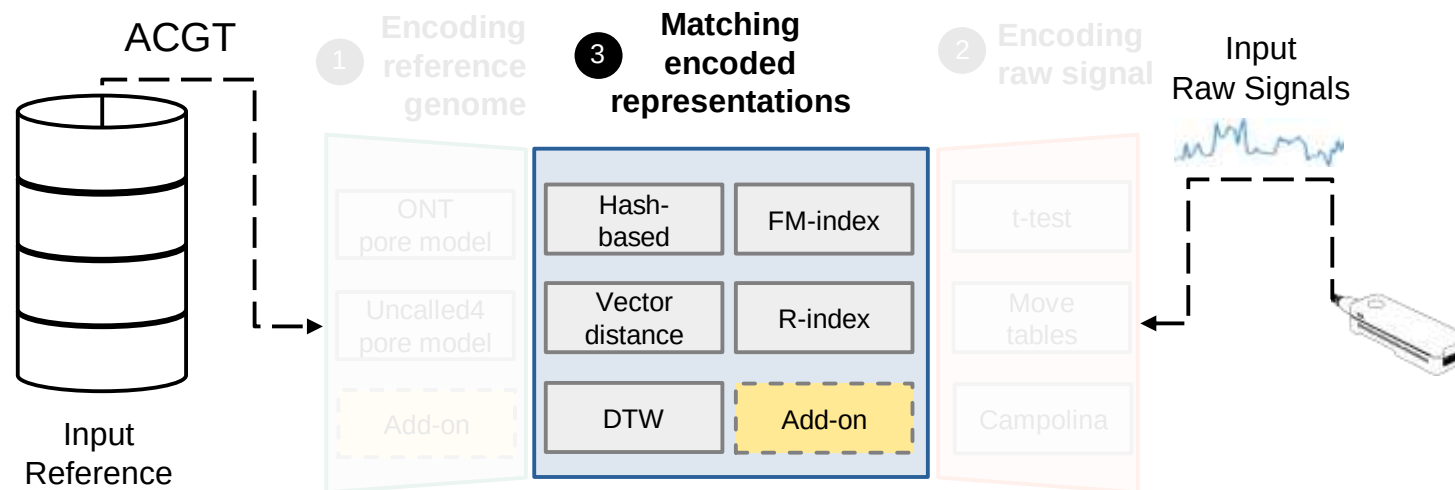
Read Mapping			
Segmentation Method	F1	Precision	Recall
<i>E. coli</i>			
t-test	0.83	0.91	0.77
Move tables	0.07	0.07	0.06
Campolina	0.89	0.94	0.85
<i>D. melanogaster</i>			
t-test	0.72	0.94	0.59
Move tables	0.05	0.23	0.03
Campolina	0.72	0.95	0.57
<i>H. sapiens</i>			
t-test	0.66	0.87	0.53
Move tables	0.01	0.11	0.01
Campolina	0.79	0.96	0.67

Campolina enables **better quality**, resulting in

1.07× - **1.2×** improvement in F_1 score

Move tables perform poorly, pointing out to the need for
more intelligent use of intermediate basecalling output

Matching Encoded Representations



Matching Encoded Representations – Quality

Read Mapping			
Matching Method	F1	Precision	Recall
<i>E. coli</i>			
Hash-based	0.83	0.91	0.77
FM-index	0.23	0.13	0.80
Vector distances	0.83	0.84	0.82
R-index	0.67	0.79	0.58
DTW	0.86	0.99	0.75
<i>D. melanogaster</i>			
Hash-based	0.72	0.94	0.59
FM-index	0.02	0.17	0.01
Vector distances	0.80	0.94	0.69
R-index	0.59	0.96	0.42
DTW	0.75	0.94	0.62
<i>H. sapiens</i>			
Hash-based	0.66	0.87	0.53
FM-index	0.01	0.05	0.01
Vector distances	0.26	0.57	0.16
R-index	0.66	0.85	0.54
DTW	0.75	0.94	0.62

Read Classification			
<i>Zymo</i>			
Matching Method	F1	Precision	Recall
Hash-based	0.95	0.92	0.97
FM-index	0.62	0.45	0.99
Vector distances	0.96	0.97	0.95
R-index	0.96	1.0	0.93
DTW	0.98	0.99	0.97

DTW-based matching shows a **consistently strong F1 score** while **vector distances** remains a competitive approach **for smaller genomes**

R-index and hash-based methods catch up in read classification, trends indicate that matching should be designed on a **case-by-case basis**

Matching Representations – Performance

Read Mapping			
Matching Method	Elapsed time (hh:mm:ss)	CPU time (sec)	Peak Mem. (GB)
<i>E. coli</i>			
Hash-based	0:05:51	5,730	4.36
FM-index	6:57:45	1,603,653	1.09
Vector distances	0:20:10	54,310	54.32
R-index	0:04:25	4,224	1.4
DTW	0:09:23	6,128	4.43
<i>D. melanogaster</i>			
Hash-based	2:07:02	462,608	9.6
FM-index	3:56:13	892,824	1.49
Vector distances	3:22:15	823,117	255.97
R-index	1:22:30	310,695	3.1
DTW	0:24:02	88,044	10.46
<i>H. sapiens</i>			
Hash-based	0:53:03	186,301	91.96
FM-index	0:08:44	32,808	7.52
Vector distances	5:59:29	1,238,190	265.16
R-index	0:35:02	131,095	29.43
DTW	0:46:35	158,289	116.2

FM-index enables read mapping in **resource-constrained** settings
despite its **existing headroom for quality**

RSA-assisted Basecalling

- Running RSA as a pre-filter to basecalling
 - Discard reads unmapped by a RSA pipeline
 - **Reduce** the expensive basecalling load

Basecalled Read Mapping				
Dataset	RSA Pre-filter	Average Depth of Cov. (×)	Breadth of Coverage (%)	Aligned Reads (#)
<i>E. coli</i>	✓	164.39	82.42	182,871

**More details on the results
can be found in the paper**

Basecalling load is reduced by **17-39%** using a **lightweight RSA pre-filter**
with only **0.07-0.09%** drop in genome completeness

Outline

Background

Motivation and Goal

RawBench

Evaluation

Conclusion

Conclusion

RawBench

The *first* benchmarking framework for **raw signal analysis (RSA)** enabling *end-to-end* fair and systematic evaluation of different raw signal analysis techniques



Currently supports

30 different RSA combinations

4 different raw nanopore signal datasets from **2** different nanopore chemistries

2 different downstream tasks and **2** RSA-assisted basecalling tasks



High modularity

enables

combination of existing and new RSA techniques from across the RSA literature

integration of **new datasets and downstream tasks**

fair comparison of **newly developed methods**

RawBench

A Comprehensive Benchmarking Framework
for Raw Nanopore Signal Analysis Techniques

Furkan Eris

Ulysse McConnell Can Firtina Onur Mutlu

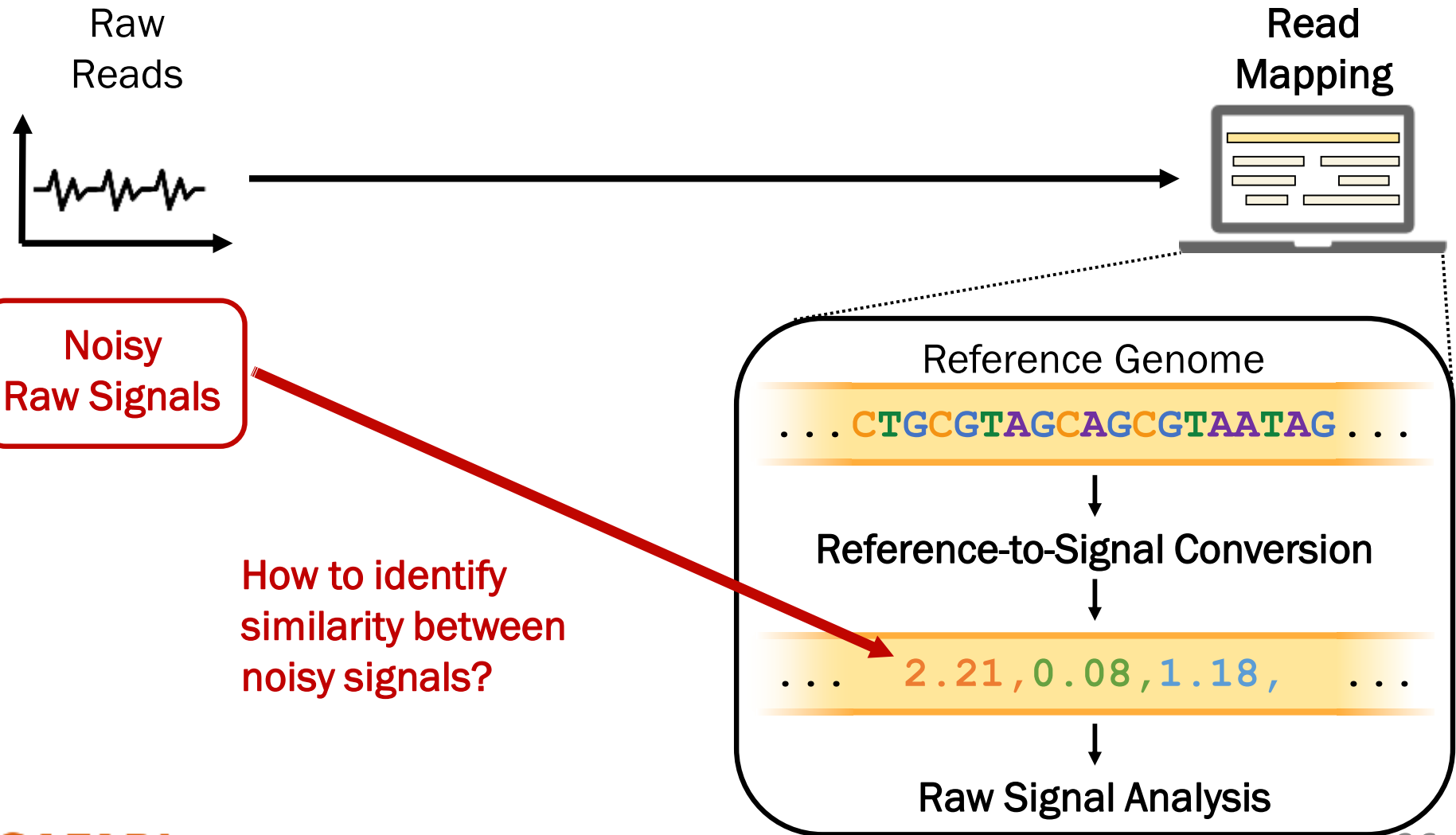
ETH zürich

SAFARI



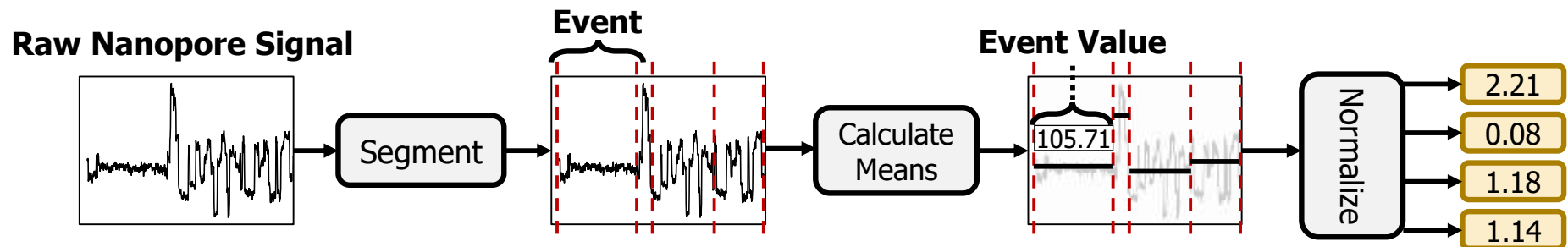
Backup Slides

Encoding the Reference Genome



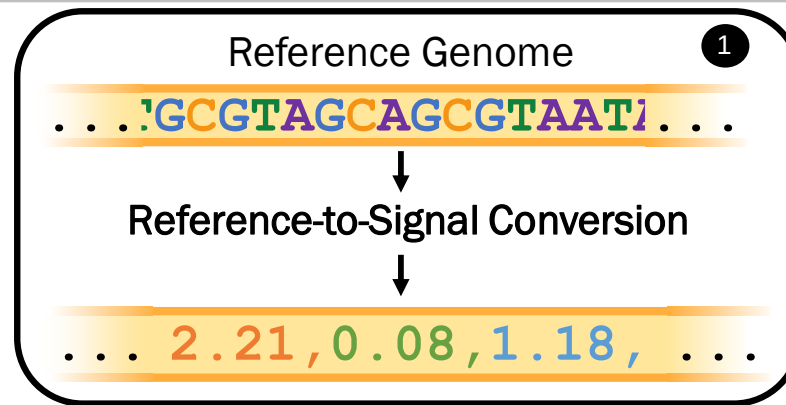
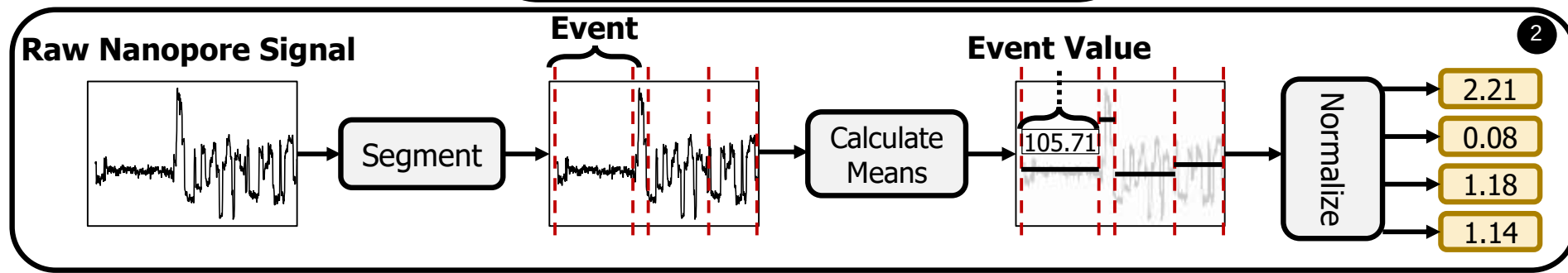
Dealing with Noisy Signals

- Signal regions corresponding to specific k-mers are identified
- Consecutive events → consecutive k-mers

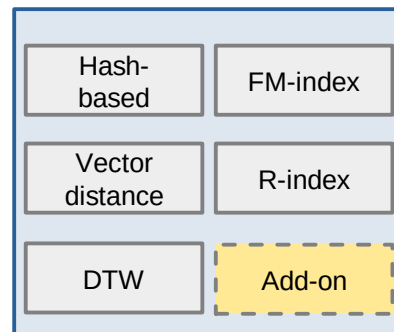


Can we match events (k-mers) between reference genome and raw signals?

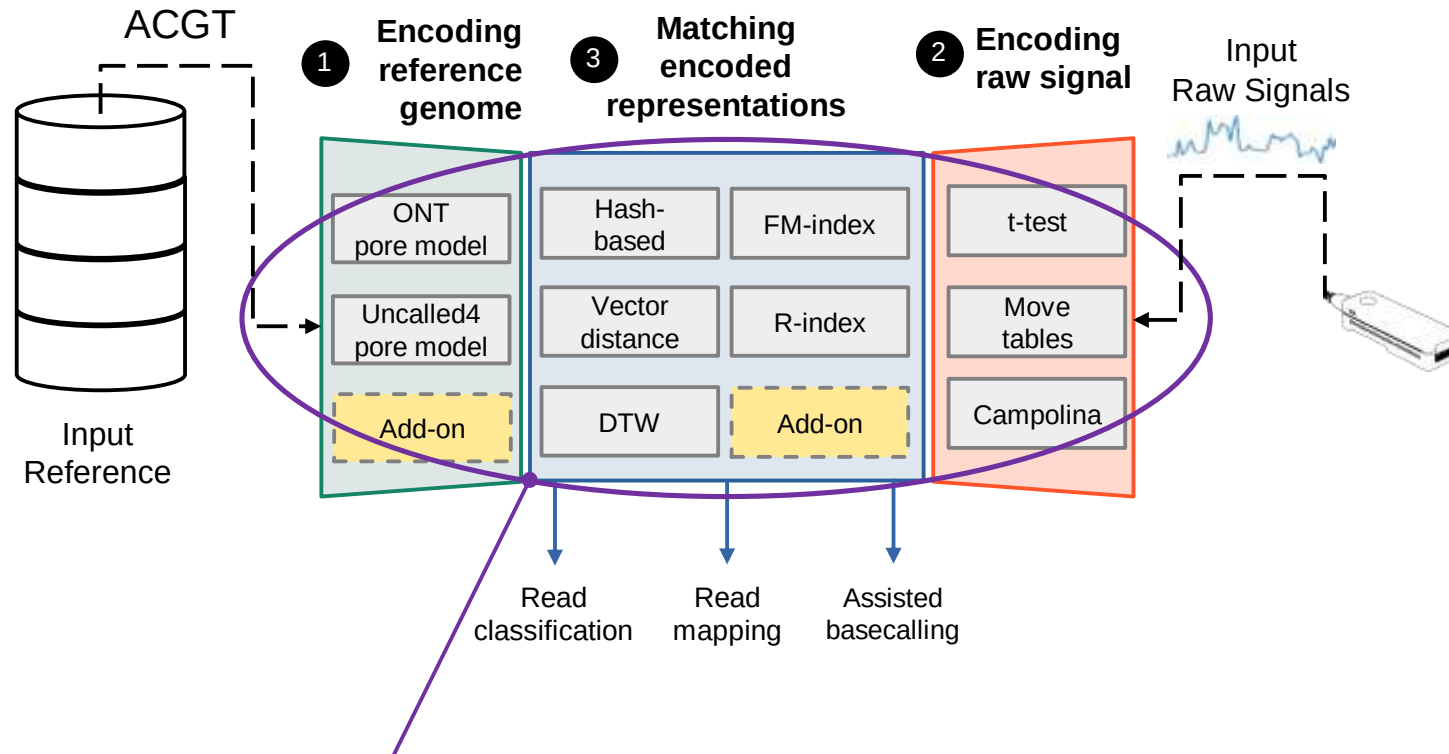
Matching Encoded Representations



3 Matching encoded representations



Existing Benchmarking Frameworks



Few existing works do not benchmark RSA techniques **at all** – let alone in an extensive and extensible manner