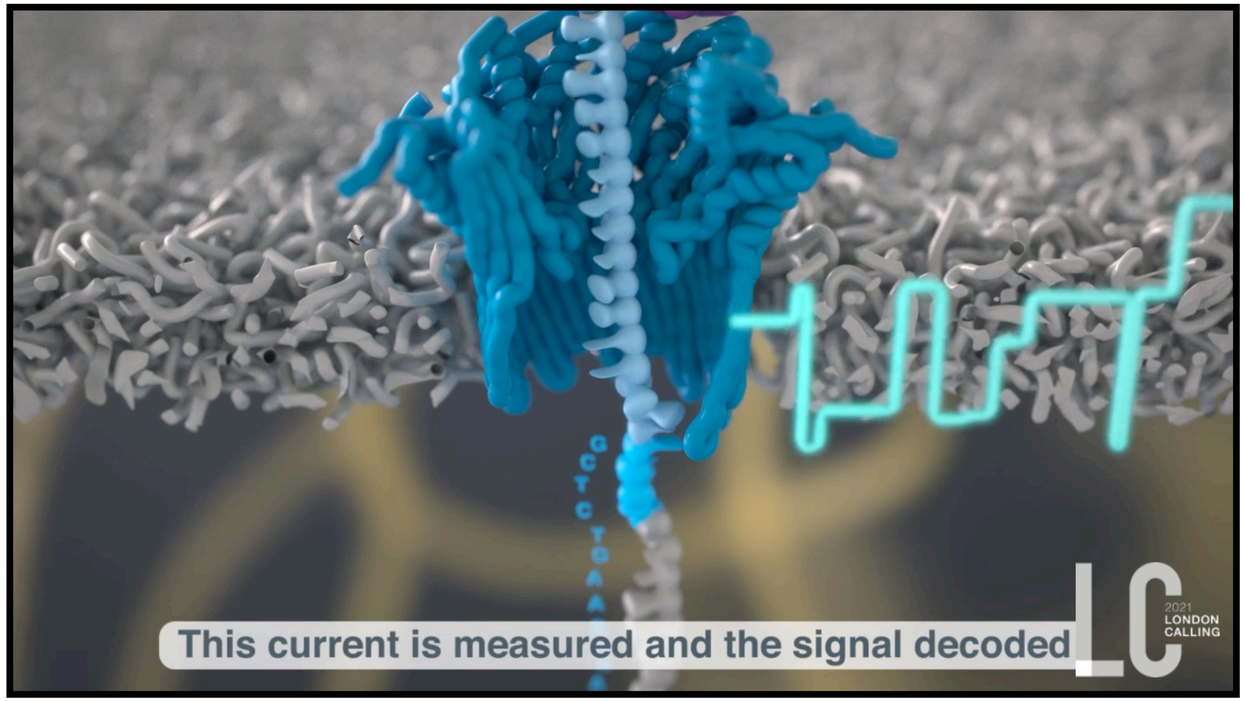
Pan-genomic matching statistics for targeted nanopore sequencing

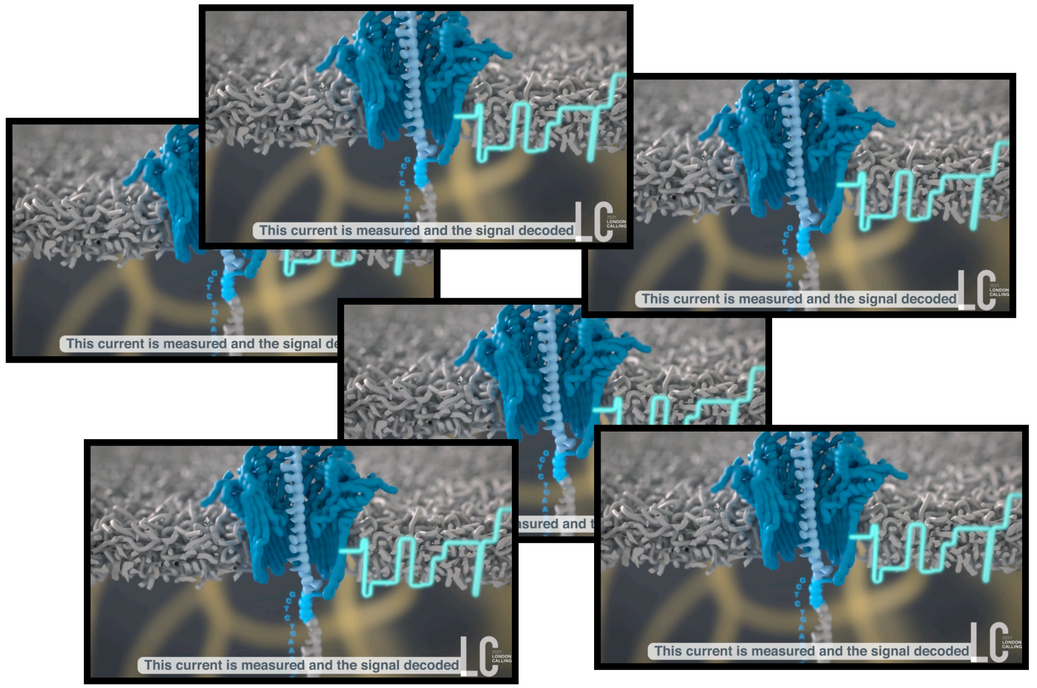
Omar Ahmed[1], Massimiliano Rossi[2], Sam Kovaka[1], Michael C. Schatz[1], Travis Gagie[3], Christina Boucher[2], and Ben Langmead[1]

RECOMB-Seq
August 27-28, 2021

[1] Department of Computer Science, Johns Hopkins University, Baltimore, MD, USA
[2] Department of Computer and Information Science and Engineering, University of Florida, Gainesville, FL, USA
[3] Faculty of Computer Science, Dalhousie University, Halifax, NS, CAN

Nanopore Sequencing





- Allows users to perform targeted sequencing using software
- MinION Flowcells have 512 pores[1]

Key Idea: The software needs to be fast enough to keep up with incoming signal from numerous pores.

[1]https://nanoporetech.com/products/comparison

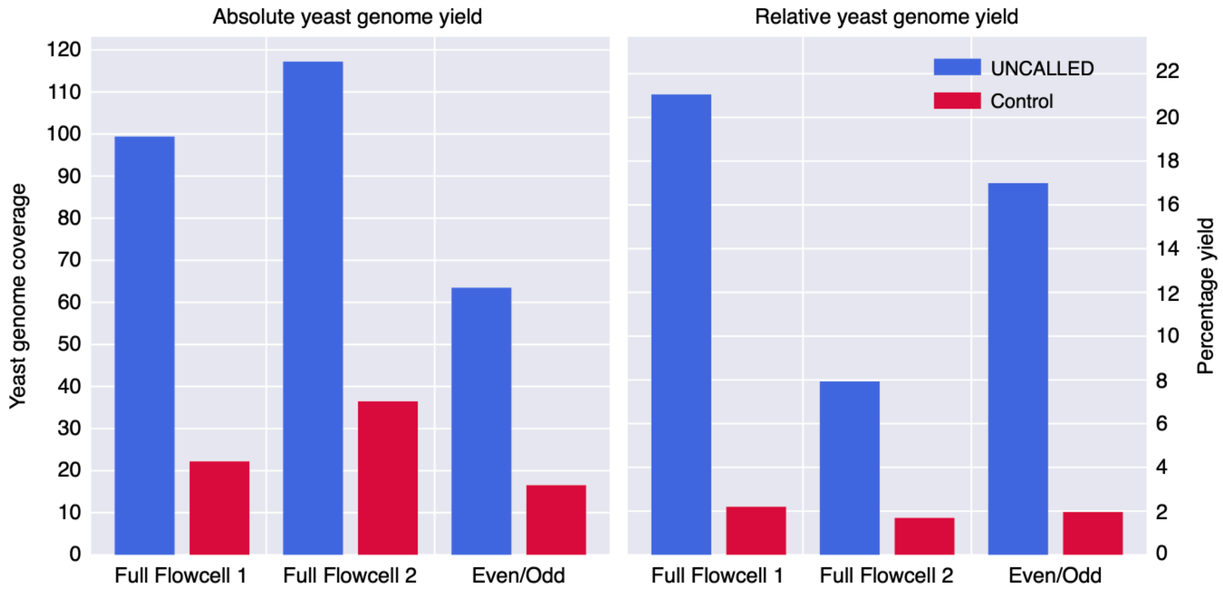
Video From: https://nanoporetech.com/about-us/news/towards-real-time-targeting-enrichment-or-other-sampling-nanopore-sequencing-devices

Targeted Sequencing with Nanopore Sequencing

- Recent methods such as UNCALLED[1] and Readfish[2] have performed targeted sequencing

Results from UNCALLED[1] paper

- Goal: Sequence a mock community and target the yeast reads and eject the microbial reads
- Using targeted sequencing yielded higher coverage of the yeast genome, and higher percent yield



"UNCALLED's performance degrades as references become larger and more repetitive"[1]

Motivation: A need for faster methods to classify reads against large, repetitive databases

Why is it important? ① Improves targeting accuracy ② Target genomes of unassembled strains

[1]Kovaka, S., Fan, Y., Ni, B. *et al.* Targeted nanopore sequencing by real-time mapping of raw electrical signal with UNCALLED. *Nat Biotechnol* **39,** 431–441 (2021).
[2]Payne, A., Holmes, N., Clarke, T. *et al.* Readfish enables targeted nanopore sequencing of gigabase-sized genomes. *Nat Biotechnol* **39,** 442–450 (2021).

Method Overview

SPUMONI - Streaming PseUdo MONI[1]

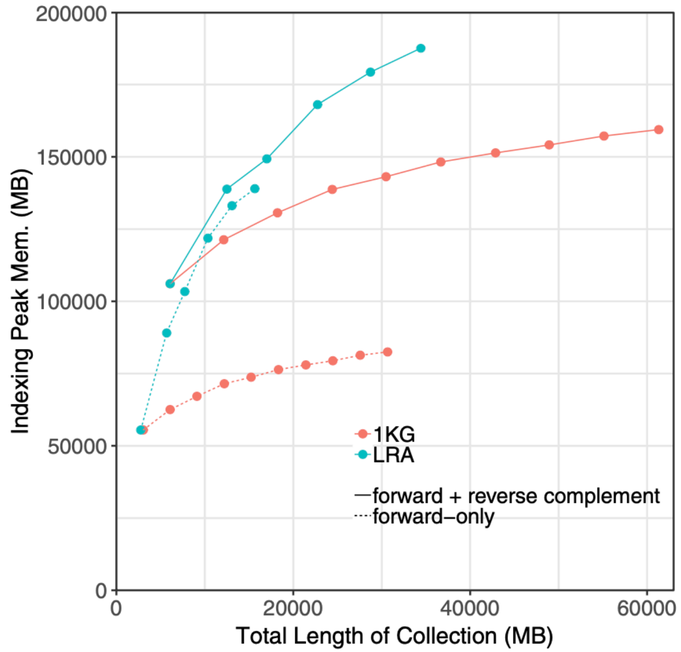
- Makes rapid targeting decisions based on input database
  - Key Intuition: A read's MS/PMLs with respect to a reference can reveal if there appears to be "good" approximate match to reference
- Uses the r-index[2] to enable efficient indexing of large, repetitive collections
  - Number of runs in BWT, r, typically grows sub-linearly w.r.t to length of input sequence, n.
- Extends MONI[1] in two key areas
  - Adds a "null index" and hypothesis testing framework for finding "significant" matches
  - Replaces MONI's "batch" matching statistic (MS) algorithm with a faster, streaming algorithm
    - Calculates new quantity called pseudo-matching lengths (PMLs)

[1]Rossi, M., Oliva, M., Langmead, B., Gagie, T., & Boucher, C. (2021). MONI: A pangenomics index for finding MEMs. *Proc. RECOMB.*

[2]Mun, T., Kuhnle, A., Boucher, C., Gagie, T.,Langmead, B., and Manzini, G. (2020). Matching reads to many genomes with the r-index. J. Comput. Biol.27, 514–518

Strengths of the *r*-index

- Mäkinen and Navarro's[1] O(*r*) rank data-structure over the BWT combined with Gagie et al.'s[2] O(*r*) suffix array sampling make up the *r*-index
  - This combination allows efficient queries to **count** the number of occurrences and **locate** those occurrences
- Peak memory of r-index construction shows a sub-linear trend showing the strength of the r-index for indexing large, repetitive collections (Mun et al.)[3]



[1] V. Makinen and G. Navarro. (2007). Rank and select revisited and extended. Theoretical Computer Science, 387. pp. 332–347.

[2] T. Gagie, G. Navarro, and N. Prezza. (2020). Fully functional suffix trees and optimal text searching in bwt-runs bounded space, J. ACM, 67 (2020), pp. 2:1–2:54.

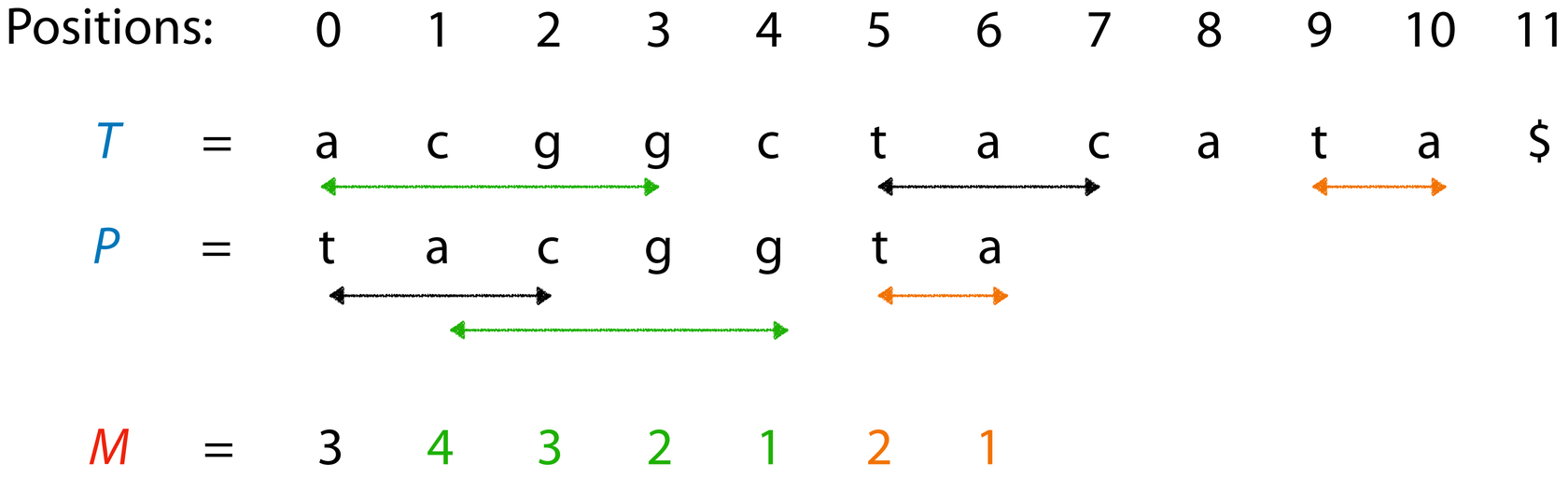
[3] Mun, T., Kuhnle, A., Boucher, C., Gagie, T.,Langmead, B., and Manzini, G. (2020). Matching reads to many genomes with the r-index. J. Comput. Biol.27, 514–518

Matching Statistics

- The matching statistics of *P* w.r.t to *T* is an array *M* of length m where *M*[*i*] is the length of the longest prefix of the pattern *P[i..m]* that occurs in text *T*
  - Let *T* be a text of length n, and *P* be a pattern of length m

- Think of matching statistics like they are half MEMs (Maximally Exact Matches)

  - Example:



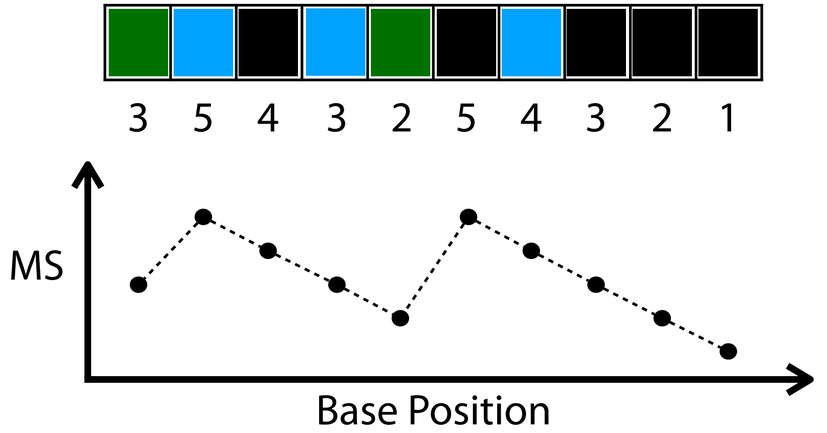
Matching Statistics - Comparing Approaches

- MONI[1] - Two Pass Algorithm → Calculates Matching Statistics

Case 1 $p[i] == BWT[j]$

Case 2a $p[i] != BWT[j]$
but can still extend current match by 1

Case 2b $p[i] != BWT[j]$
but can possibly retain some of current match

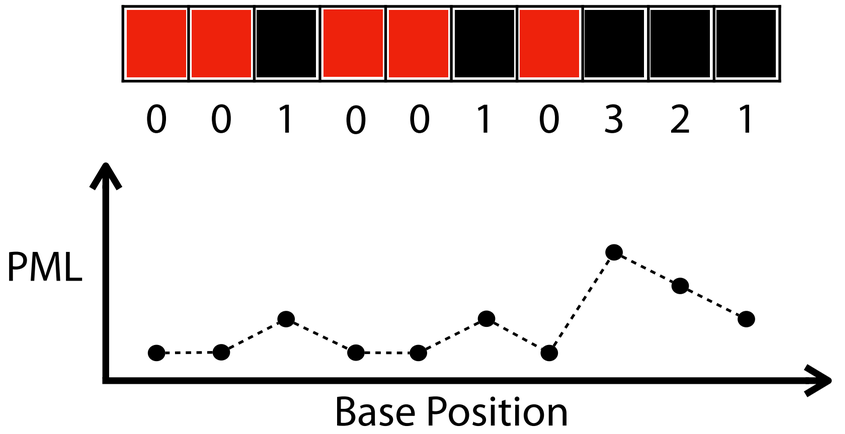


- SPUMONI - Online Algorithm → Calculates Pseudo-Matching Lengths

Case 1 $p[i] == BWT[j]$

Case 2 $p[i] != BWT[j]$
so we set the length down to 0

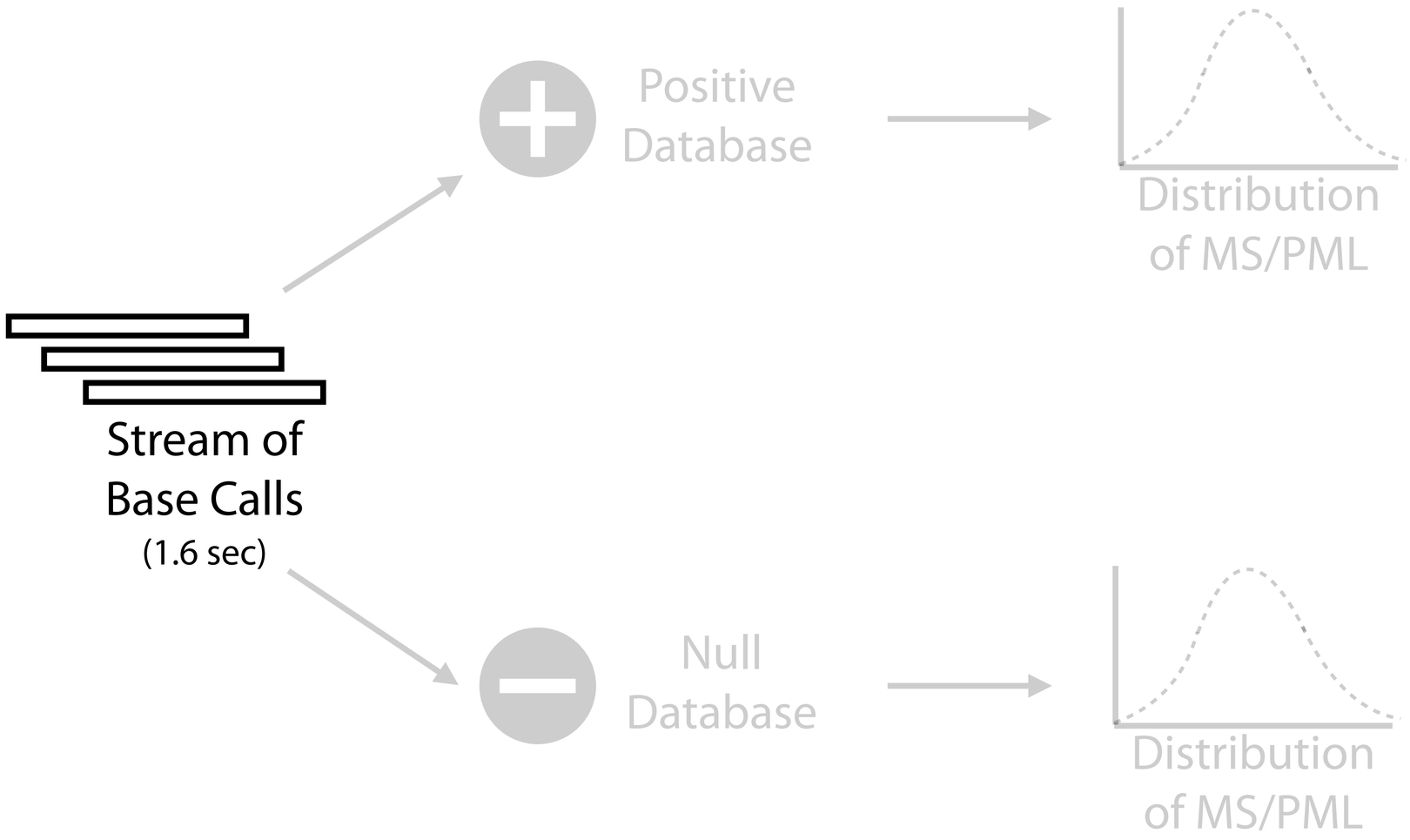
- Simplification leads to smaller indexes, so SPUMONI uses ~3X less memory and is ~3X faster



How will each quantity perform when it comes to classification?

[1]Rossi, M., Oliva, M., Langmead, B., Gagie, T., & Boucher, C. (2021). MONI: A pangenomics index for finding MEMs. *Proc. RECOMB.*

SPUMONI Approach



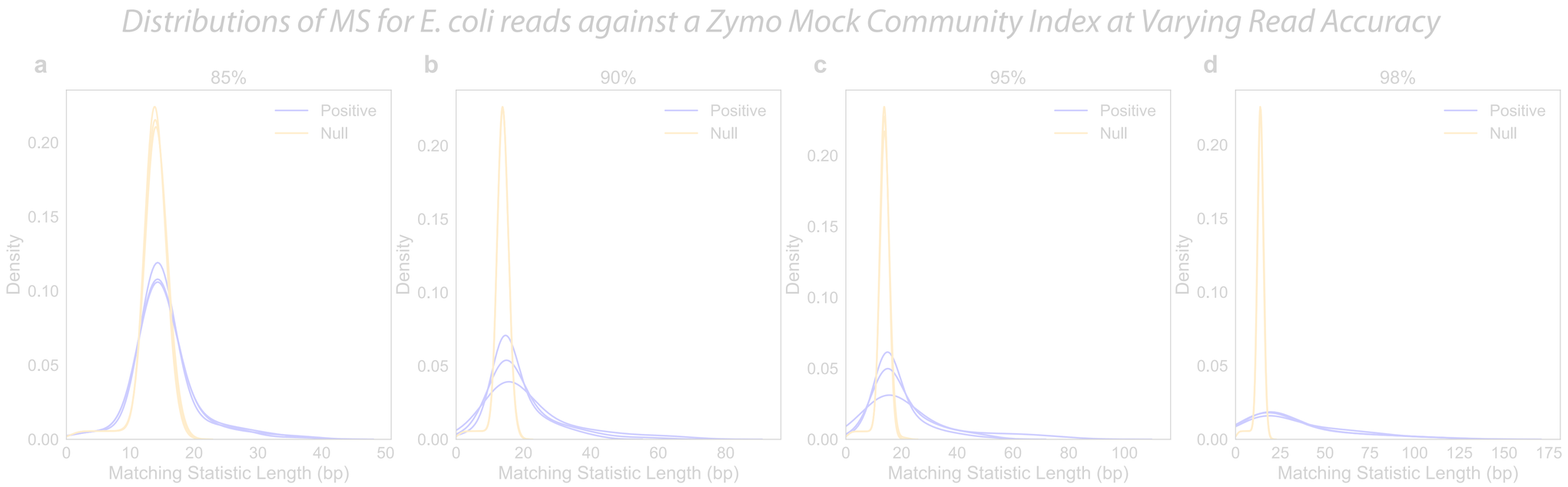
Statistical Tests:

KS-Test

Image: https://en.wikipedia.org/wiki/Kolmogorov%E2%80%93Smirnov_test

Results - Varying Mean Read Accuracy

- **Question:** How does sequencing error affect SPUMONI's classification?



- **Answer:** SPUMONI's accuracy $\geq$ 99.5% for reads with 90% accuracy or more
  - At 85% read accuracy, SPUMONI's accuracy was 95.43 and minimap2's accuracy was 99.16

Results - Real Mock Community Experiment

- **Question:** Can using a pan-genome reference allow us to target a particular strain that is not present in the reference? and how does it compare on time and memory?
  - Using **real** mock community reads[1] where we want to "target" the yeast reads, and eject all the microbial species

Table 3. Comparing SPUMONI and minimap2 across various metrics on Real ZymoMC Reads

Accuracy, throughput and index size on real mock community reads

Reference:	One genome ref			Pan-genome ref		
Reference size:	56 MB	56 MB	28 MB	31 GB	31 GB	16 GB
Approach:	SPUMONI-ms	SPUMONI	minimap2	SPUMONI-ms	SPUMONI	minimap2
Accuracy	81.64	86.72	87.82	94.62	96.02	97.52
Precision	100.00	100.00	100.00	100.00	100.00	99.96
Recall	81.39	86.54	87.66	94.55	95.97	97.53
Specificity	100.00	100.00	100.00	100.00	100.00	96.97
F1-score	89.74	92.79	93.42	97.20	97.94	98.73
Peak RSS (GB)	0.63	0.08	0.17	6.24	1.90	8.07
Index size (GB)[a]	0.68	0.09	0.10	6.20	1.90	31.00
Throughput (bp/s)	252,974	901,609	851,869	64,384	185,618	15,570

[a]The index sizes for SPUMONI-ms and SPUMONI are only for the positive index because the null index can be used offline and removed.

4.2X smaller RSS
16.3X smaller index
11.9X faster

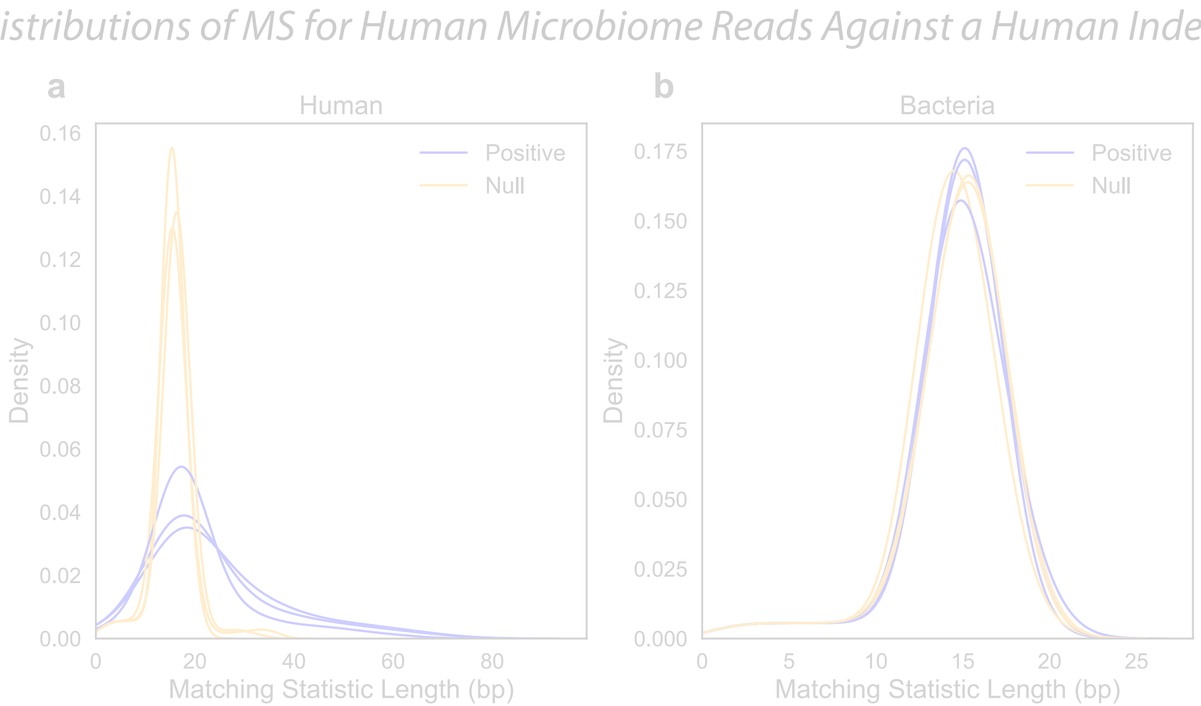
- **Answer:** ① Yes, using a pan-genome reference, allowed us to target the ZymoMC strains

  ② Faster and uses less memory than minimap2 with similar classification metric

[1]Kovaka, S., Fan, Y., Ni, B. *et al.* Targeted nanopore sequencing by real-time mapping of raw electrical signal with UNCALLED. *Nat Biotechnol* **39,** 431–441 (2021).

Results - Human Microbiome Experiment

- **Question:** Does SPUMONI's extend to other scenarios[1], in particular using human genomes?

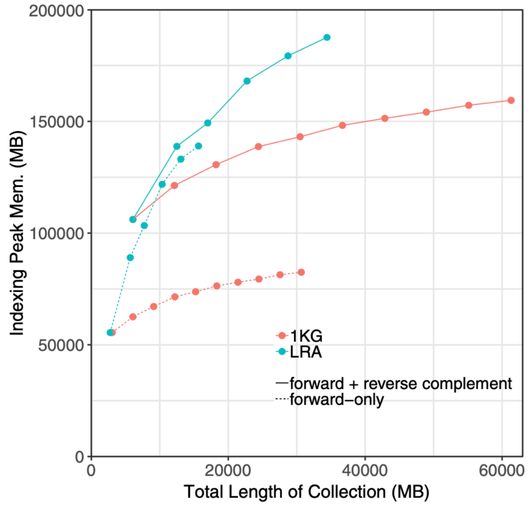


- **Answer:** SPUMONI's accuracy is 99.44, while minimap2's accuracy is 99.84 while SPUMONI is 2X faster.

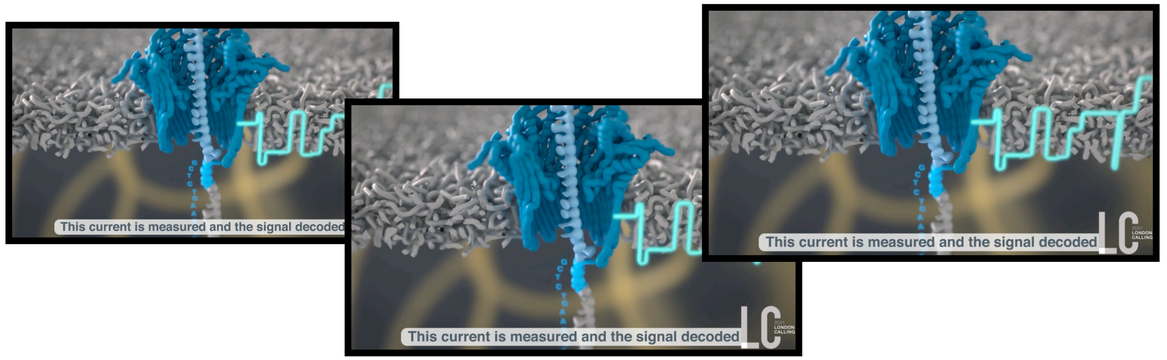
[1]Moss, E.L., Maghini, D.G., and Bhatt, A.S. (2020). Complete, closed bacterial genomes from microbiomes using nanopore sequencing. Nat. Biotech. 38, 701–707.

Conclusion

- SPUMONI is a streaming algorithm for targeted nanopore sequencing that uses a read's MS or PMLs to classify it in real time
  - Use of the r-index and MONI's thresholds allow SPUMONI to index **large, repetitive** references more efficiently than competing approaches



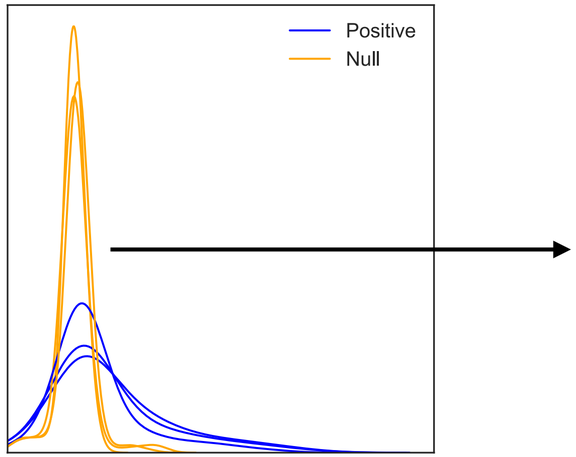
- SPUMONI's computational efficiency makes it well-suited for nanopore's portable sequencers like MinION or Flongle, and there numerous channels in single flow-cell



Conclusion

- SPUMONI's approach opens the door to doing targeted sequencing for strains that have not been assembled previously, or cannot be cultured

- SPUMONI is not limited by the use of a k-mer length setting or other simple threshold



"Null" distribution allows the notion of significance to be a function of database sequences, and the error rate of the query read.

- Future work will consist of implementing the needed software for SPUMONI to interact with the Read Until API

Thank you!

Contact: oahmed6@jhu.edu

GitHub: https://github.com/oma219/spumoni

Acknowledgements:

- Massimiliano Rossi, Sam Kovaka, Michael C. Schatz, Travis Gagie, Christina Boucher & Ben Langmead for help & assistance on the project
- Nae-Chyun Chen, Daniel Baker, Taher Mun, Kathleen Newcomer, Anna Liebhoff & Dominik Kempa from Langmead Lab
- Marco Oliva from Boucher Lab, and Jarno Niklas Alanko from Gagie Lab

Funding:

NSERC
CRSNG