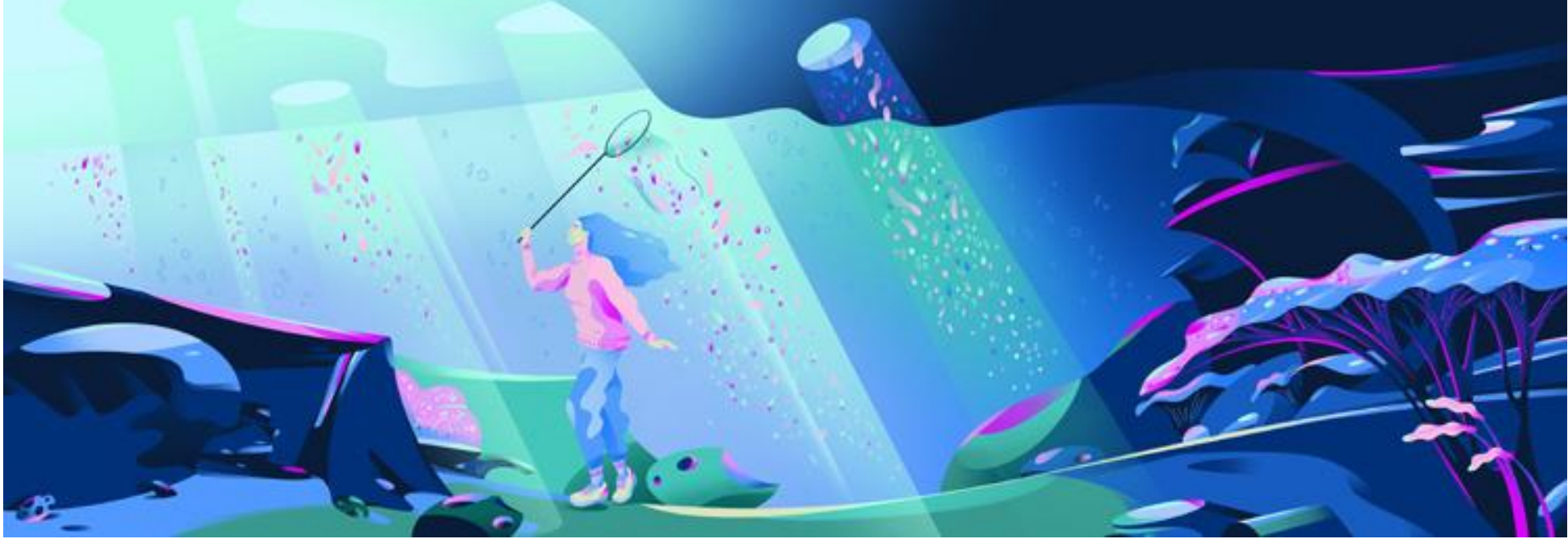


@Manon Sauzara



Kmers

- Representing datasets
- Indexing datasets
- Finding matches



Pierre Peterlongo

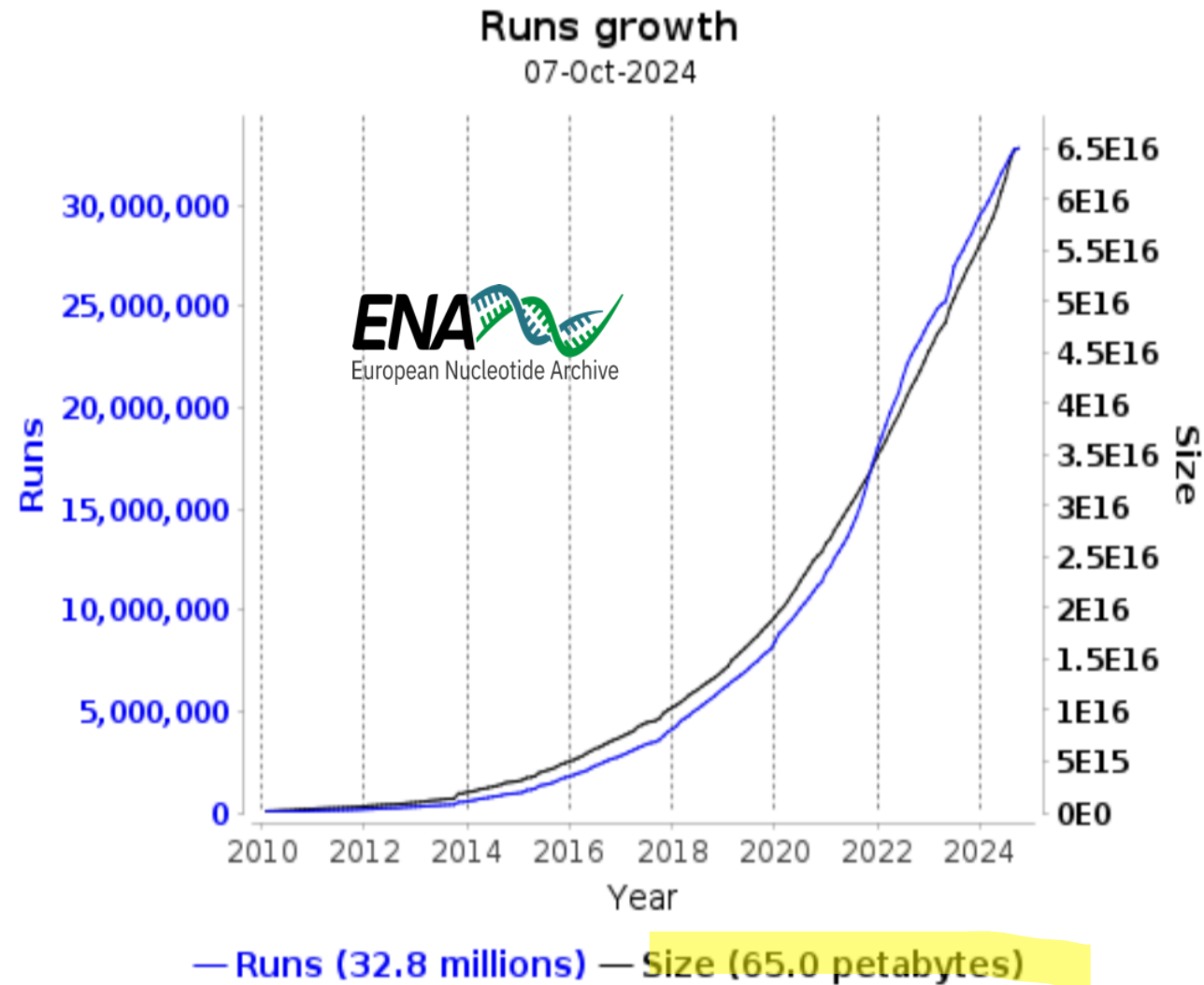


Ebame-9, oct. 2024



Context

Evolution



We have genomes?

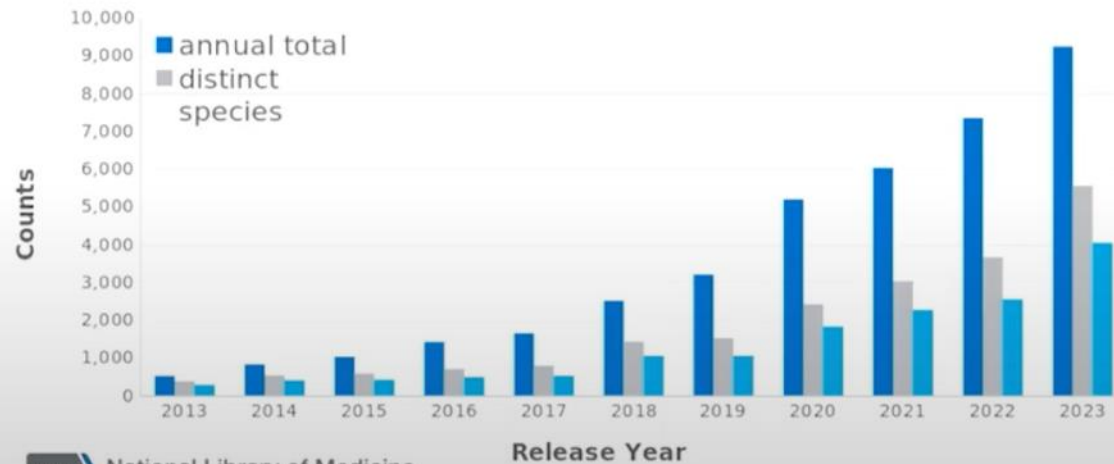
ALL EUKARYOTIC GENOMES (Cumulative: Dec 2023):

GenBank genomes (all): 36,593 (15,453 species)

GenBank (with annotation): 6,817 (3,801 species)

(Out of 8 million known species..)

Annual Growth in Sequenced Species and Genomes



NIH National Library of Medicine
National Center for Biotechnology Information

Slide credit: Terence Murphy, NCBI

How to

- represent and manipulate datasets
- index & query them

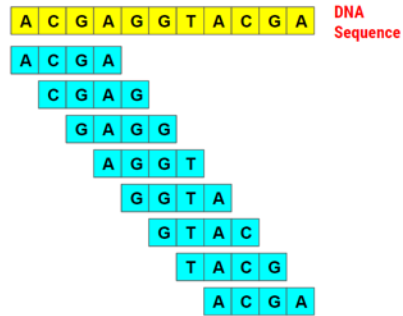
Kmers

Words

- No word in DNA
- Split to subsequences of fixed length k (called **kmers**)

$(20 < k < 40)$

4-mers



- Thousand billions distinct kmers
 - *(google indexes millions)*

$$k = 31$$

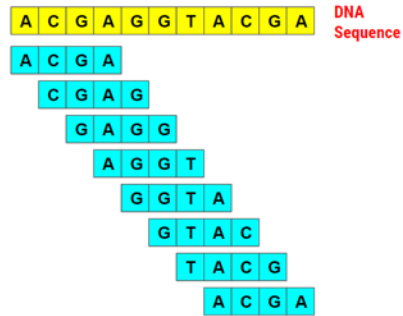
Kmers

Words

- **No word in DNA**
- Split to subsequences of fixed length k (called **kmers**)

$(20 < k < 40)$

4-mers

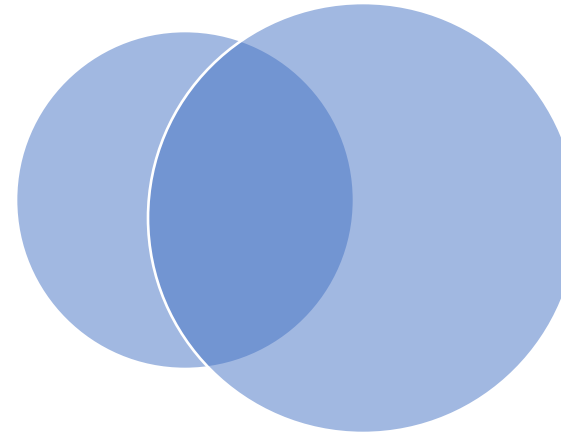


- Thousand billions distinct kmers
 - *(google indexes millions)*

Sequence similarity ~ shared kmers count

Bank1

Bank2



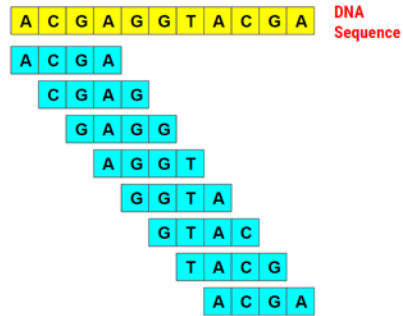
Kmers

Words

- **No word in DNA**
- Split to subsequences of fixed length k (called **kmers**)

$(20 < k < 40)$

4-mers

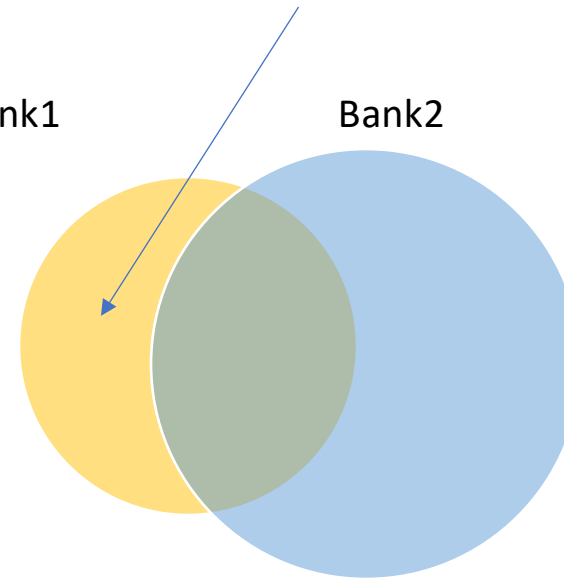


- Thousand billions distinct kmers
 - *(google indexes millions)*

Sequence specificity

Bank1

Bank2



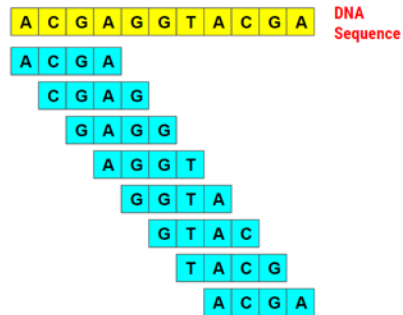
Kmers

Words

- No word in DNA
- Split to subsequences of fixed length k (called **kmers**)

$(20 < k < 40)$

4-mers



- Thousand billions distinct kmers
 - (*google indexes millions*)

Query sequence

ACGAGGTACGA In bank

ACGA Yes

CGAG Yes

GAGG Yes

AGGT Yes

GGTA Yes

GTAC No

TACG Yes

ACGA Yes

- 7 over 8 kmers shared with a dataset
 - 7/8 of the query in the bank



Swim in data tsunami: float on kmers

All kmers from a read set

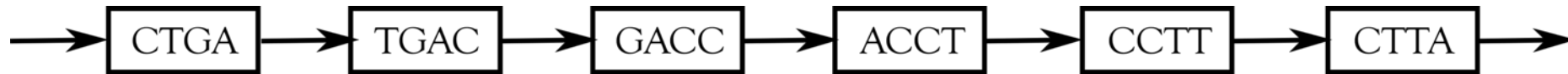
- ~Easy to manipulate/represent
- Represent the sequences
 - diversity
 - Similarity
 - Proxy for sequence alignments
 - variants
- Absorb redundancies (eg. coverage)
 - Eg 100x coverage of a 1GB genome:
 - reads: ~100 GB
 - filtered kmers: ~1GB



kmers are also used for assembly:

The de Bruijn graph

- Instead of comparing all pairs of reads:
 - Kmerize the reads
 - go from one kmer to its child



- Assembled sequence: **CTGACCTTA**

Kmtricks: Big data and k-mer representation

<https://github.com/tlemanek/kmtricks>

T. Lemane, P. Medvedev, R. Chikhi and P. Peterlongo, "kmtricks: Efficient and flexible construction of Bloom filters for large sequencing data collections." Bioinformatics Advances, 2022, doi:10.1093/bioadv/vbac029.

The k -mer matrix representation

- An abstract data type representing k -mer abundances across samples

Abundance matrix

	S1	S2	...	Sn
k1	3	2	...	0
k2	0	0	...	4
...
kn	8	0	...	12

Presence/Absence matrix

	S1	S2	...	Sn
k1	1	1	...	0
k2	0	0	...	1
...
kn	1	0	...	1

Main issue?

- Billions of rows, hundreds/thousands of columns → **terabytes of data**

Sequential vs random access

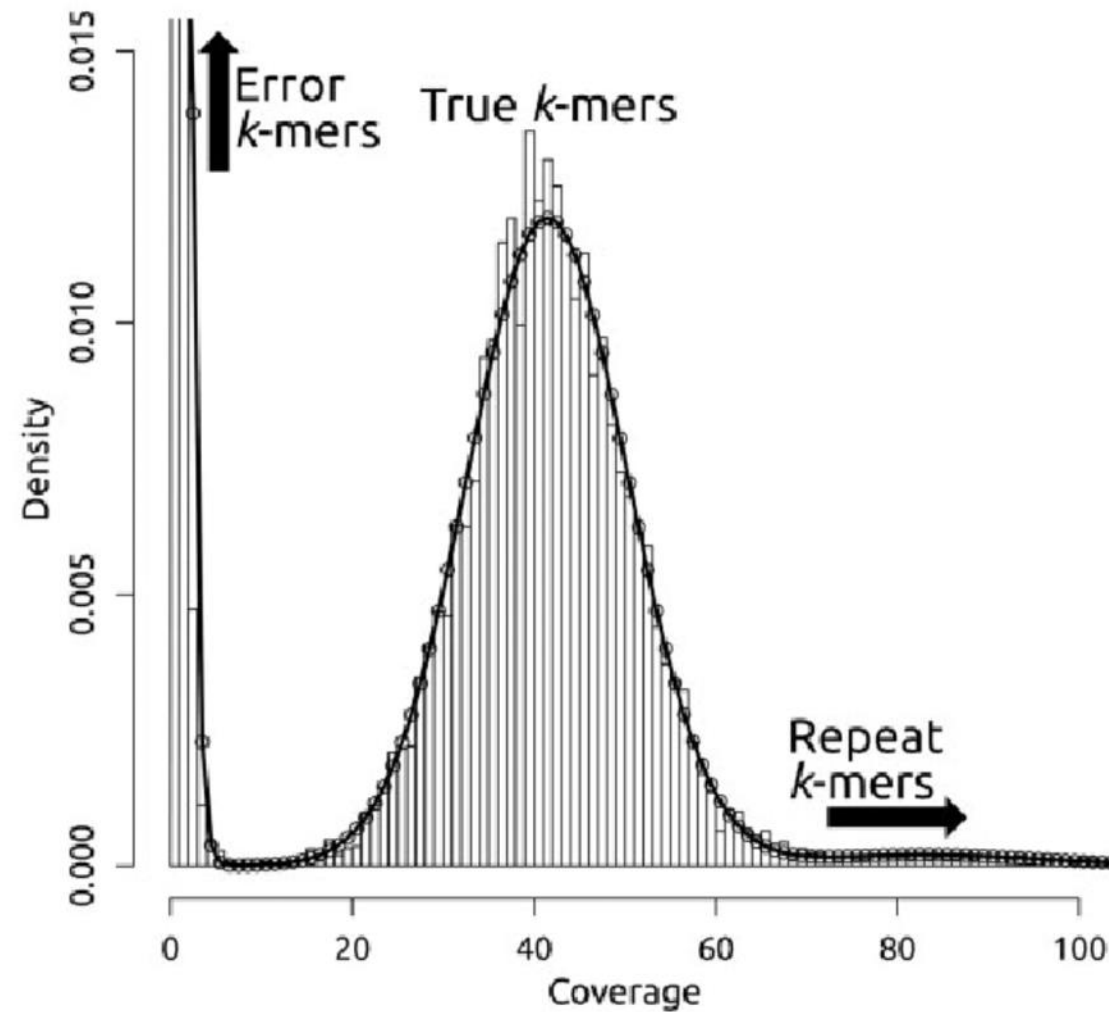
- Focusing on **matrix streaming**, i.e. enumerating each row

Sequencing errors lead to erroneous k -mers

	S_0	S_1	S_2	S_3	S_4	S_5	S_6	S_7
AATC	0	5	0	1	6	5	2	2
ACTA	5	8	5	9	1	6	4	7
ATAA	0	0	1	0	1	1	0	0
ATCA	6	1	1	5	1	6	5	7
CCAC	6	0	0	4	4	4	8	7
CCGC	1	5	8	4	9	7	5	2
CTCG	8	1	0	1	4	8	4	5
GCTC	2	8	9	6	4	7	1	9
GCTG	0	0	0	0	1	0	0	0
GTCG	1	1	9	2	0	5	0	1
TGTG	9	5	9	3	2	6	7	2
TTCA	1	6	7	9	9	3	5	0

Classical k -mer filtering

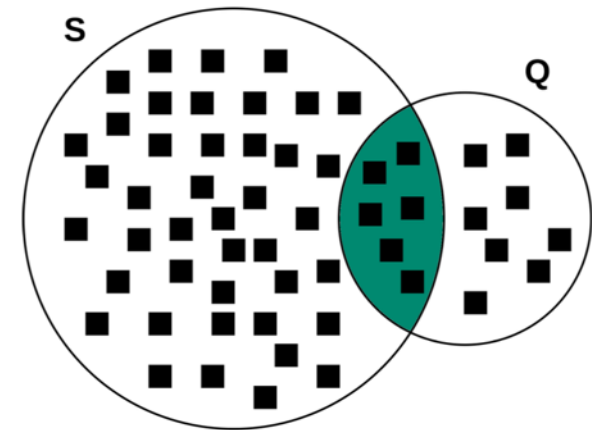
- Discarding low-abundant k -mers



Kmtricks features

- Fast matrices construction
- Count kmers & filter low abundant kmers
 - Also rescue low abundant ubiquitous kmers
- Output **matrices** (0/1 or counts)
- Output **bloom filters** (wait a few slides)
- Enables filtering (intersection, unions)
- Enables streaming
- Enables user-defined plugins

	S_0	S_1	S_2	S_3	S_4	S_5	S_6	S_7
AATC	0	5	0	1	6	5	2	2
ACTA	5	8	5	9	1	6	4	7
ATAA	0	0	1	0	1	1	0	0
ATCA	6	1	1	5	1	6	5	7
CCAC	6	0	0	4	4	4	8	7
CCGC	1	5	8	4	9	7	5	2
CTCG	8	1	0	1	4	8	4	5
GCTC	2	8	9	6	4	7	1	9
GCTG	0	0	0	0	1	0	0	0
GTCG	1	1	9	2	0	5	0	1
TGTG	9	5	9	3	2	6	7	2
TTCA	1	6	7	9	9	3	5	0



Kmindex: querying PB of k-mer-indexed datasets

<https://github.com/tleman/kmindex>

Lemane, Téo, et al. "Indexing and real-time user-friendly queries in terabyte-sized complex genomic datasets with kmindex and ORA" Nature Computational Science 4.2 (2024): 104-109.

Google

AGGGGCTGAGCGGCGGGCAGGCAGCTTTCAGGGACTCAGTTCT



All



Images



Shopping



Videos



Maps



More

Tools

About 0 results (0.18 seconds)

Your search - **AGGGGCTGAGCGGCGGGCAGGCAGCTTTCAGGGACTCAGTTCTACA** -
did not match any documents.

Use blast?

Basic local alignment search tool

[SF Altschul](#), [W Gish](#), [W Miller](#), [EW Myers](#)... - Journal of molecular ..., 1990 - Elsevier

... A new approach to rapid **sequence** comparison, **basic local alignment search tool** (BLAST), directly approximates **alignments** that optimize a measure of **local** similarity, the maximal ...

☆ Save  Cite Cited by 111045 Related articles All 43 versions

« Historical search engine »

Use blast?

Basic local alignment search tool

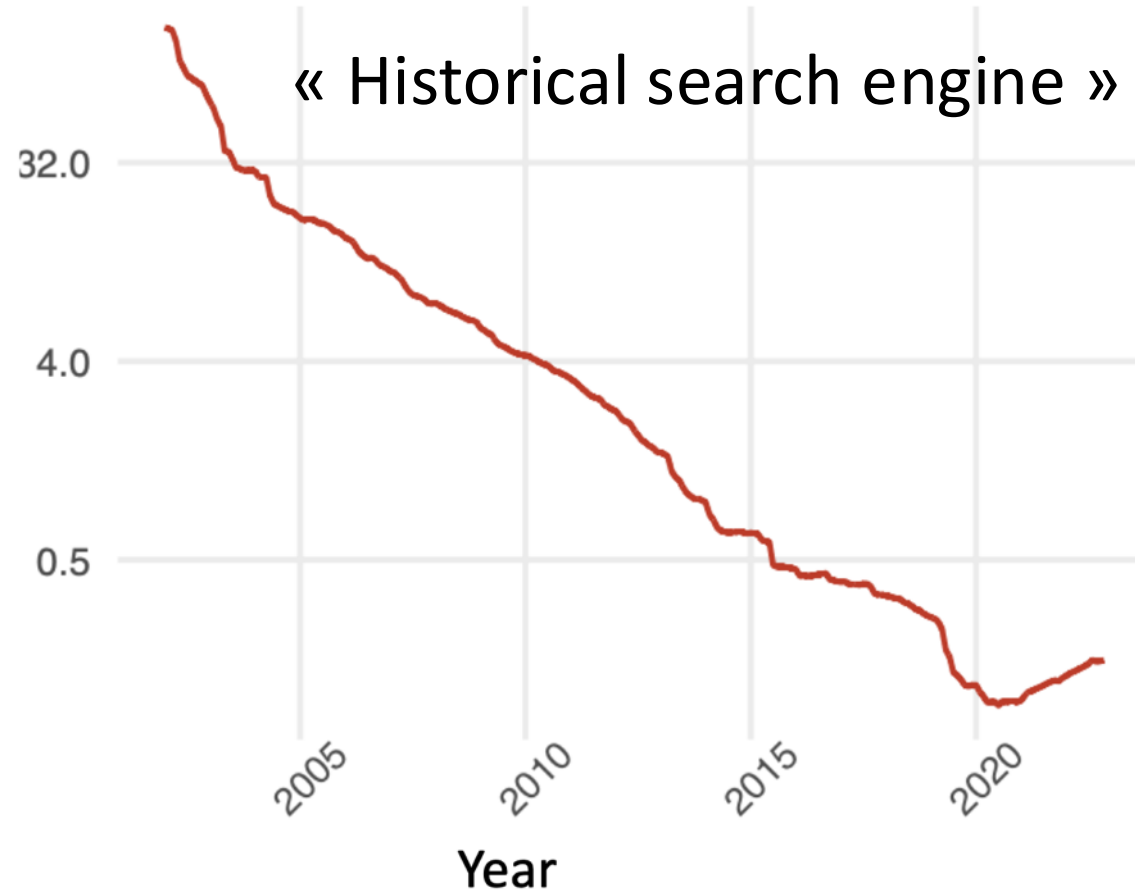
[SF Altschul](#), [W Gish](#), [W Miller](#), [EW Myers](#)... - Journal of molecular ..., 1990 - Elsevier

... A new approach to rapid **sequence** comparison, **basic local alignment search tool** (BLAST), directly approximates **alignments** that optimize a measure of **local** similarity, the maximal ...

☆ Save [Cite](#) Cited by 111045 [Related articles](#) [All 43 versions](#)

$\text{Log} \left(\frac{|\text{BLAST NT}|}{|\text{NCBI Bacteria}|} \right)$

« Historical search engine »



If we could search

- **Agronomy**
- Environment
- Health

Agronomy

- Limit the inputs
- Plant breeding and protection
- Varietal selection

Molecular Plant
Perspective

 **CellPress**
Partner Journal

Creation and judicious application of a wheat resistance gene atlas

Amber N. Hafeez¹, Sanu Arora¹, Sreya Ghosh¹, David Gilbert¹, Robert L. Bowden²
and Brande B.H. Wulff^{1,*}

¹John Innes Centre, Norwich Research Park, Norwich, UK

²USDA-ARS, Hard Winter Wheat Genetics Research Unit, Manhattan, KS 66506, USA

*Correspondence: Brande B.H. Wulff (brande.wulff@jic.ac.uk)

<https://doi.org/10.1016/j.molp.2021.05.014>

If we could search

- Agronomy
- **Environment**
- Health

Environment

- Understand the evolution
- Biodiversity inventory
- Understand interspecies interactions

nature reviews microbiology

[Explore content](#) ▾ [About the journal](#) ▾ [Publish with us](#) ▾

[nature](#) > [nature reviews microbiology](#) > [review articles](#) > [article](#)

Review Article | Published: 12 May 2020

Tara Oceans: towards global ocean ecosystems biology

[Shinichi Sunagawa](#) ✉, [Silvia G. Acinas](#), [Peer Bork](#), [Chris Bowler](#), [Tara Oceans Coordinators](#), [Damien Eveillard](#), [Gabriel Gorsky](#), [Lionel Guidi](#), [Daniele Iudicone](#), [Eric Karsenti](#), [Fabien Lombard](#), [Hiroyuki Ogata](#), [Stephane Pesant](#), [Matthew B. Sullivan](#), [Patrick Wincker](#) & [Colomban de Vargas](#) ✉

[Nature Reviews Microbiology](#) **18**, 428–445 (2020) | [Cite this article](#)

18k Accesses | **172** Citations | **170** Altmetric | [Metrics](#)

If we could search

- Agronomy
- Environment
- **Health**

Health

- Diseases (cancers, neurodegenerative)
- Microbiome characterization
- Rapid Antimicrobial resistance


nature

[Explore content](#) ▾ [About the journal](#) ▾ [Publish with us](#) ▾

[nature](#) > [articles](#) > article

Article | Published: 26 January 2022

Petabase-scale sequence alignment catalyses viral discovery

[Robert C. Edgar](#), [Brie Taylor](#), [Victor Lin](#), [Tomer Altman](#), [Pierre Barbera](#), [Dmitry Meleshko](#), [Dan Lohr](#), [Gherman Novakovsky](#), [Benjamin Buchfink](#), [Basem Al-Shayeb](#), [Jillian F. Banfield](#), [Marcos de la Peña](#), [Anton Korobeynikov](#), [Rayan Chikhi](#) & [Artem Babaian](#) 

[Nature](#) **602**, 142–147 (2022) | [Cite this article](#)

Genomic research engine: conceptual view index

Set representation

- A bank (genome, reads, ...) represented by its kmer content

Atomic question

- Given a queried kmer, does it exist in the indexed set?



Genomic research engine: conceptual view index

Set representation

- A bank (genome, reads, ...) represented by its kmer content

Atomic question

- Given a queried kmer, in which sets does it exist?



Genomic research engine: conceptual view

index

Set representation

- A bank (genome, reads, ...) represented by its kmer content

Atomic question

- Given a queried kmer, in which sets, with which abundance?



: A bloom filter

Bloom Filter

A bit vector B of fixed size

Add one element:

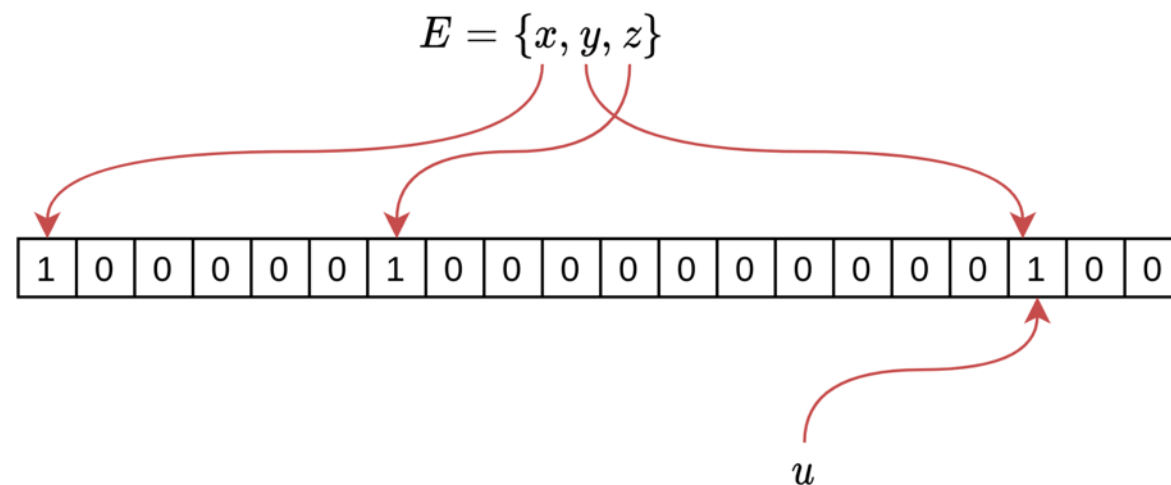
$B[\text{hash}(\text{element})] = 1$

Query one element: $B[\text{hash}(\text{element})]$

0: absent

1: present (possibly a False Positive)

1
0
0
1
0
0
1
1
1



Note: we use a unique hash function

Indexing: conceptual view (uses kmtricks)

One read set:

- Extract & count **kmers**
- Filter kmers
- Generate a bloom filter

Reads

```
>read1
ACGAG...ACGTA
>read2
ACGGC...GGACT
...
>read1000000
GGCGA...AGATA
```

Counted kmers

```
AAAAAC 12
ACCATA  4
AGGTAT  1
...
TCGGAT  5
```

Bloom Filter

```
0
1
1
...
0
```

Indexing: conceptual view (uses kmtricks)

One read set:

- Extract & count **kmers**
- Filter kmers
- Generate a bloom filter

N read sets:

- Create N bloom filters
- This is the index

Reads

```
>read1
ACGAG...ACGTA
>read2
ACGGC...GGACT
...
>read1000000
GGCGA...AGATA
```

Counted kmers

```
AAAAAC 12
ACCATA  4
AGGTAT  1
...
TCGGAT  5
```

Bloom Filter

```
0
1
1
...
0
```

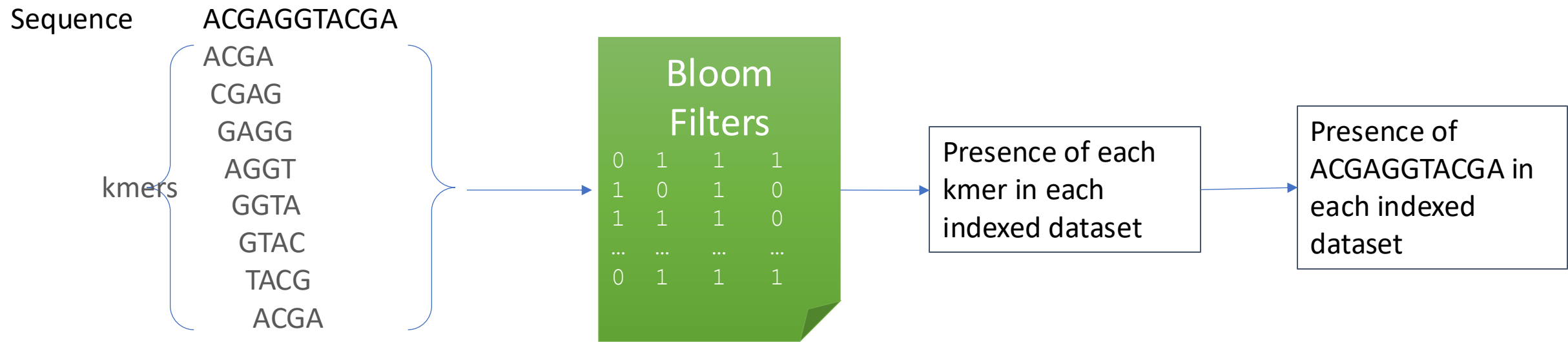
Reads

```
>read1
ACGAG...ACGT
...
>read1000000
GGCGA...AGAT
```

Bloom Filters

0	1	1	1
1	0	1	0
1	1	1	0
...
0	1	1	1

Indexing: conceptual view (uses kmtricks)



A parenthesis about False Positives: findere

« Certainly the simplest and most effective trick I've ever reviewed »



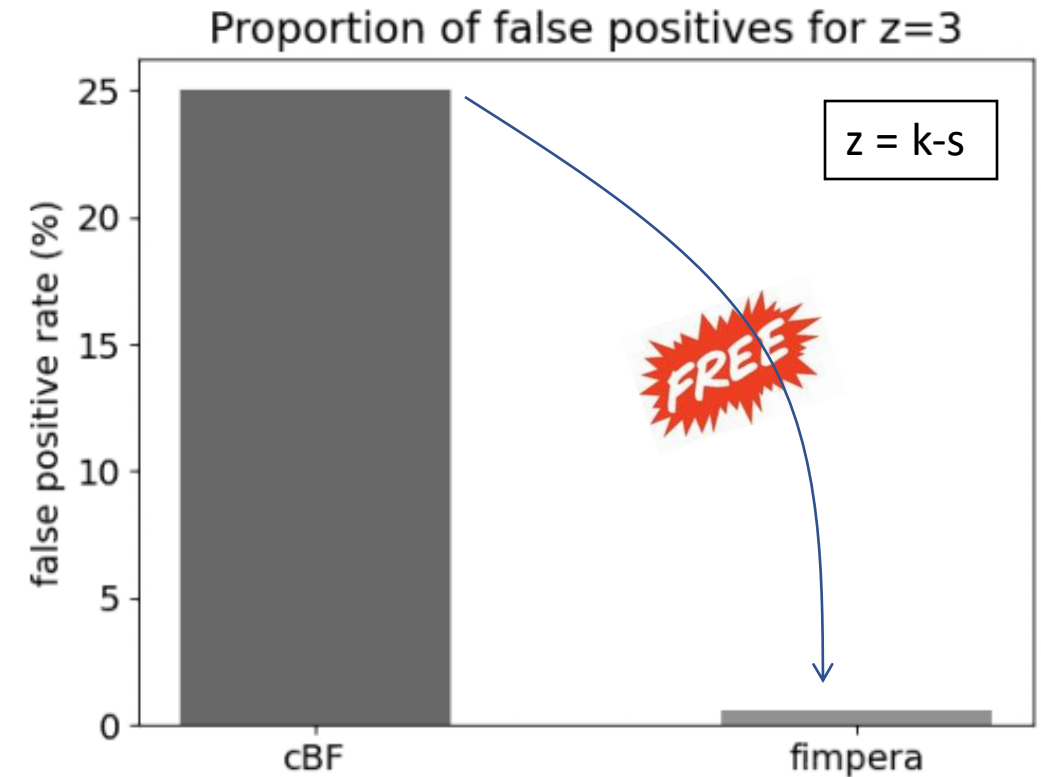
Findere: BF with low disk, low FP, no drawback

key idea for presence absence:

- If a kmer exists all words inside this kmer (smers) exist
- ↔
- If a smer of a kmer does not exist, the kmer does not exist

In practice:

- Index smers
- When querying a kmer, report it as present *iff* all its constituent smers are present



Indexed: Tara Ocean ERR1726642
Queried: Tara Ocean ERR4691696



Findere: BF with low disk, low FP, no drawback

key idea for presence absence:

- If a kmer exists all words inside this kmer (smers) exist
- ↔
- If a smer of a kmer does not exist, the kmer does not exist

In practice:

- Index smers
- When querying a kmer, report it as present *iff* all its constituent smers are present



Fast query





Findere: BF with low disk, low FP, no drawback

key idea for presence absence:

- If a kmer exists all words inside this kmer (smers) exist
- ↔
- If a smer of a kmer does not exist, the kmer does not exist

In practice:

- Index smers
- When querying a kmer, report it as present *iff* all its constituent smers are present



Fast query





Findere: BF with low disk, low FP, no drawback

key idea for presence absence:

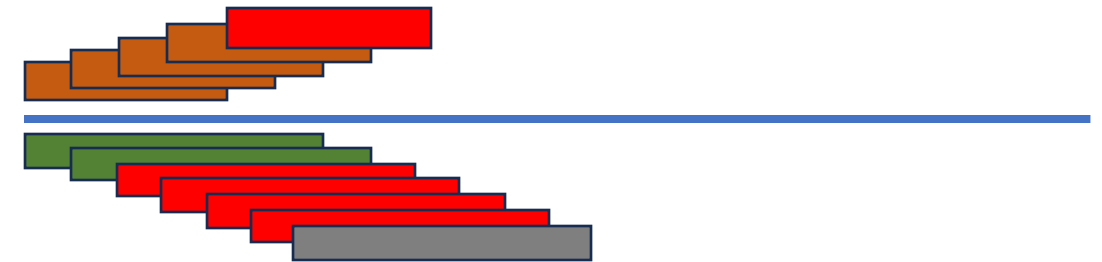
- If a kmer exists all words inside this kmer (smers) exist
- ↔
- If a smer of a kmer does not exist, the kmer does not exist

In practice:

- Index smers
- When querying a kmer, report it as present *iff* all its constituent smers are present



Faster query





Findere: BF with low disk, low FP, no drawback



key idea for presence absence:

- If a kmer exists all words inside it kmer (smers) exist
- If a smer kmer does not exist

Without **Fimperera**, the same precision would require **~35x times** more space

In practice:

- Index smer
- When querying a kmer, report it as present *iff* all its constituent smers are present

A few technical details about kminindex construction and structure

- If we have time...

STORED INDEX

	S_1
hash ₁	0
hash ₂	0
hash ₃	0

Partition 1

STORED INDEX

	S_1	S_2	S_3	S_4	S_5	S_6	$S_{N=7}$	
hash ₁	0	1	0	0	0	1	1	Partition 1
hash ₂	0	1	0	1	0	0	1	
hash ₃	0	1	1	0	1	0	0	

STORED INDEX

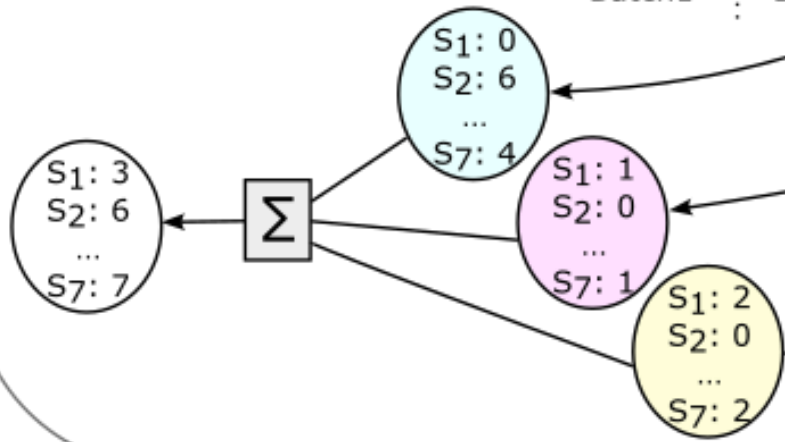
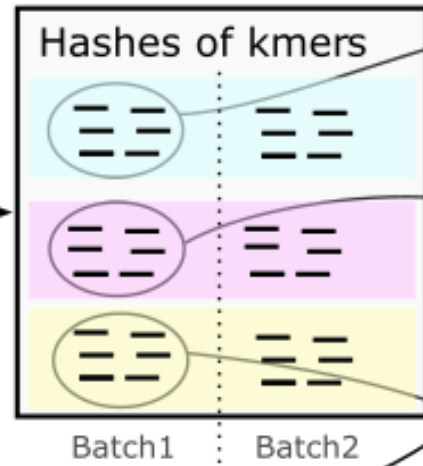
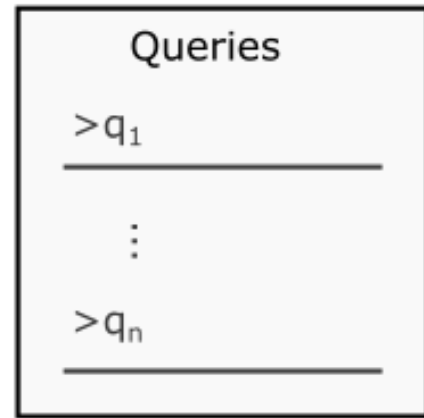
	S_1	S_2	S_3	S_4	S_5	S_6	$S_{N=7}$	
hash ₁	0	1	0	0	0	1	1	Partition 1
hash ₂	0	1	0	1	0	0	1	
hash ₃	0	1	1	0	1	0	0	

STORED INDEX

	S_1	S_2	S_3	S_4	S_5	S_6	$S_{N=7}$	
hash ₁	0	1	0	0	0	1	1	Partition 1
hash ₂	0	1	0	1	0	0	1	
hash ₃	0	1	1	0	1	0	0	
hash ₄	0	0	1	0	0	0	1	Partition 2
hash ₅	0	0	0	0	1	0	0	
hash ₆	1	0	1	0	0	1	0	
hash ₇	0	0	0	1	0	0	0	Partition 3
hash ₈	1	0	0	0	0	0	1	
hash ₉	1	0	1	0	0	1	1	

\longleftrightarrow
 $\min(0, 8 - N \% 8)$

QUERY TIME



STORED INDEX

	S_1	S_2	S_3	S_4	S_5	S_6	$S_{N=7}$	
hash ₁	0	1	0	0	0	1	1	Partition1
hash ₂	0	1	0	1	0	0	1	
hash ₃	0	1	1	0	1	0	0	
hash ₄	0	0	1	0	0	0	1	Partition2
hash ₅	0	0	0	0	1	0	0	
hash ₆	1	0	1	0	0	1	0	
hash ₇	0	0	0	1	0	0	0	Partition3
hash ₈	1	0	0	0	0	0	1	
hash ₉	1	0	1	0	0	1	1	

$8 - (N \% 8)$

A few results

Tara Oceans POC

Databank:

- 50 Tara Ocean samples
- Avg 11 billions distinct kmers per sample
- 1.4TB fastq.gz

Indexing: one command line

```
kmindex files | smer |
```

(23)

```
| bloom |
```

(25% FP)

nature computational science

Explore content ▾ About the journal ▾ Publish with us ▾ Subscribe

[nature](#) > [nature computational science](#) > [brief communications](#) > article

Brief Communication | Published: 26 February 2024

Indexing and real-time user-friendly queries in terabyte-sized complex genomic datasets with kmindex and ORA

[Téo Lemane](#) ✉, [Nolan Lezsoche](#), [Julien Lecubin](#), [Eric Pelletier](#), [Magali Lescot](#), [Rayan Chikhi](#) & [Pierre Peterlongo](#) ✉

[Nature Computational Science](#) 4, 104–109 (2024) | [Cite this article](#)



Tara Schooner - Creative Commons Attribution 3.0

Tara Oceans POC

Comparative results (successful tools)

	Time	Build index			Query time		FP rate (%)	
		RAM	Disk	Index size	Number of queried reads		Average	Maximum
		GB	GB	GB	1	10 million		
MetaProFi	30h 15min	278	5,684	226	12.72s	1h 29min	11.18	21.55
COBS	26h 30min	278	5,684	184	1.51s	15h 56min	13.29	24.60
kindex	2h 56min	107	878	164	0.06s	4min 21s	0.006	0.18

ORA Server

<https://ocean-read-atlas.mio.osupytheas.fr/>

Index: all Tara Ocean Metagenomic samples (no abundance yet)

- Input fastq.gz files
 - 6TB
 - 1,393 samples
- Final index size: 0.6TB
- Abundance: soon

ORA is part of the ELIXIR infrastructure
ORA is an Elixir service - Read more

Back to sequences: Find the origin of k-mers

https://github.com/pierrepeterlongo/back_to_sequences

Baire et al., (2024). Back to sequences: Find the origin of k-mers. Journal of Open Source Software, 9(101), 7066, <https://doi.org/10.21105/joss.07066>

Find similar sequences

Kmindex enables to know to which dataset D my query Q is similar

“Super, but Q is similar to which sequences d_i from D ?”

$Q = \text{ACGGATCGCATCA}$

D

```
>read1
CGGCATCTAGGGGCAT
>read2
TTACGGATGGCATCAC
...
>read100,000,000
GGCATGGCGAGCGGCA
```

Back to sequences (b2s)

$Q = \text{ACGGATCGCATCA}$ similar to
 $d_i = \text{TTACGGATTGCATCACA}$

Back to sequences

- IN:
 - A query Q (seen as a set of kmers)
 - A bank D
- OUT:
 - Sequences d_i from the bank similar to the query
- Optionally:
 - Abundance of kmers from Q in D
 - Mapping positions of kmers from Q in each d_i



@Manon Sauzara

Thanks!!!

Tutorial is here:

<https://github.com/pierrepeterlongo/kmersEbame-9>



Pierre Peterlongo



Ebame-9, oct. 2024

