ETH zürich

Scalable Annotated Genome Graphs for Representing Sequence Data

Mikhail Karasikov

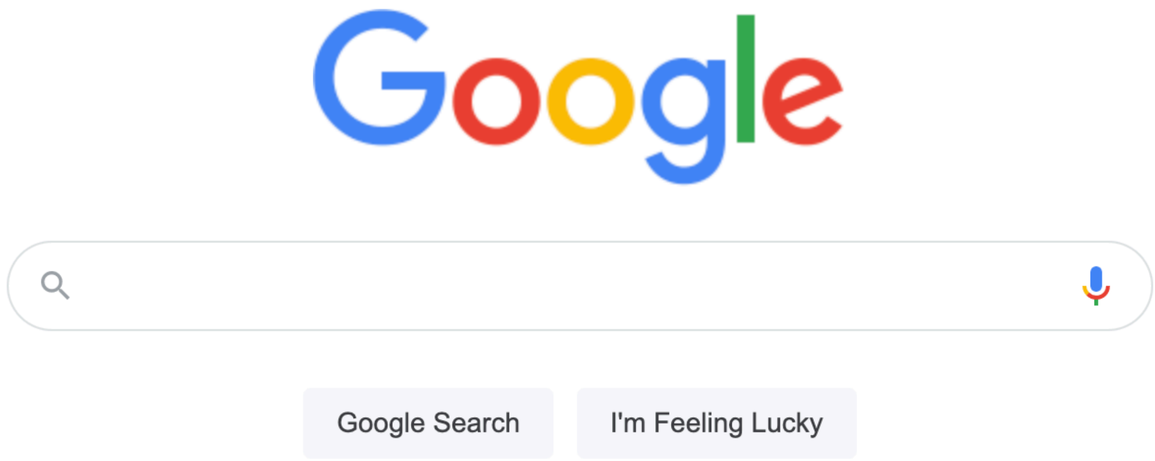
Doctoral examination

10 July 2023 (Zurich, Switzerland)

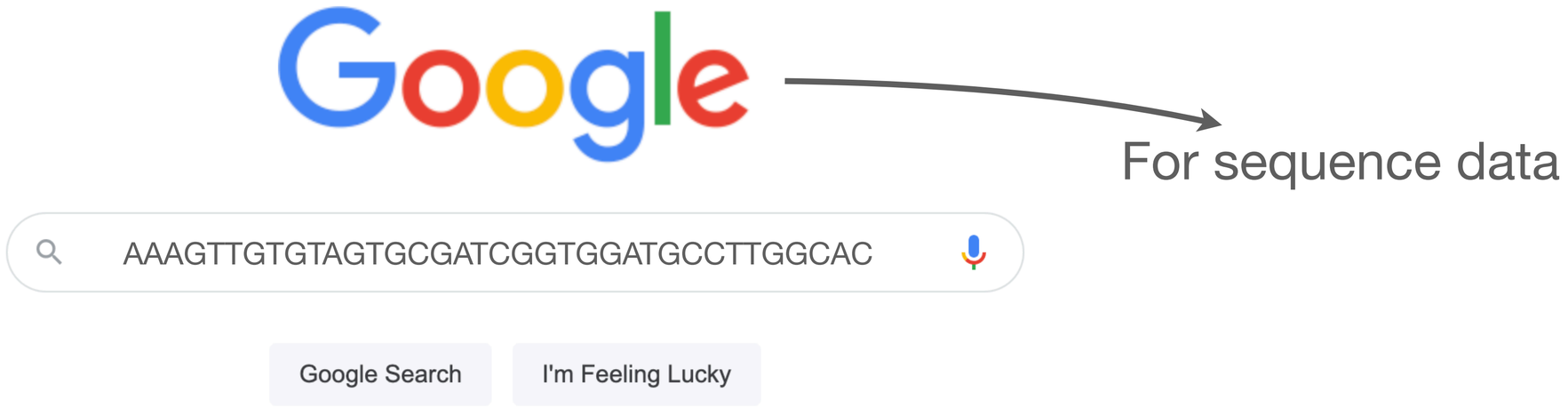
Committee:

Prof. Dr. Gunnar Rätsch
Prof. Dr. Jean-Philippe Vert
Prof. Dr. Ewan Birney
Dr. Rayan Chikhi

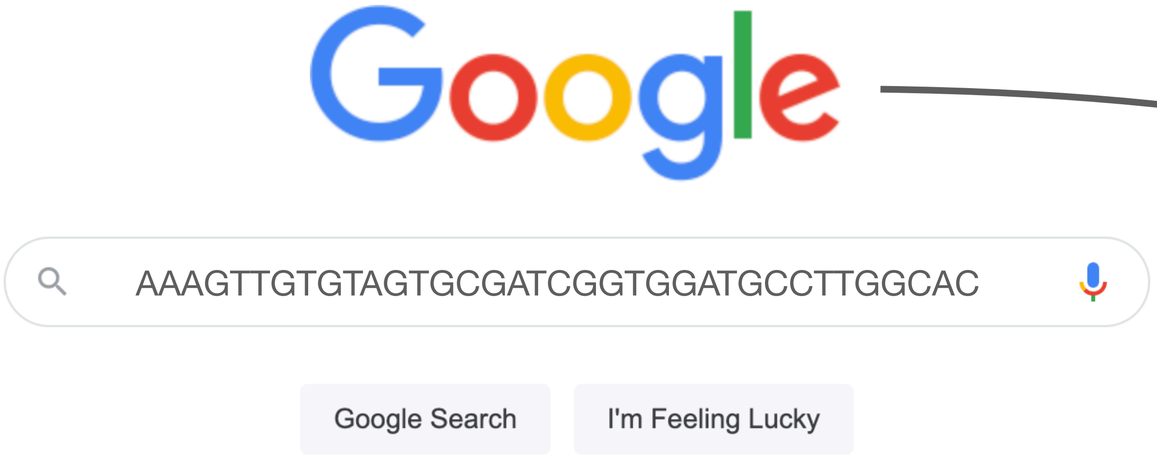
What we want



What we want


Google
For sequence data
AAAGTTGTGTAGTGCGATCGGTGGATGCCTTGGCAC
Google Search
I'm Feeling Lucky

What we want



For sequence data

ENA European Nucleotide Archive

UK 10K

HMP

GENOME 10K

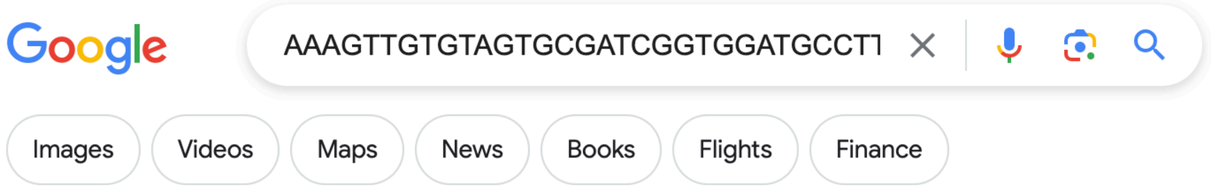
european genome-phenome archive

NCBI SRA

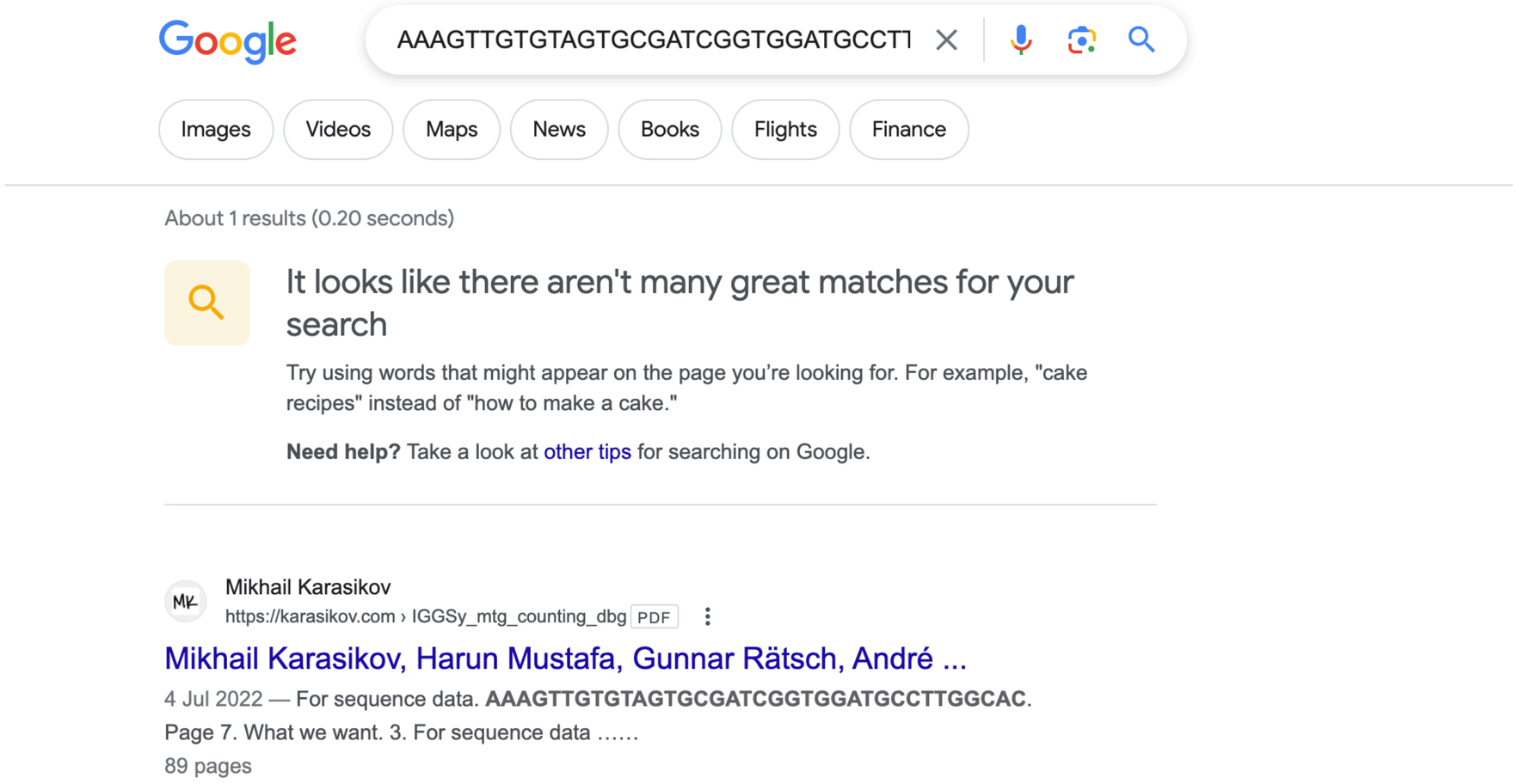
MetaSUB

• • •

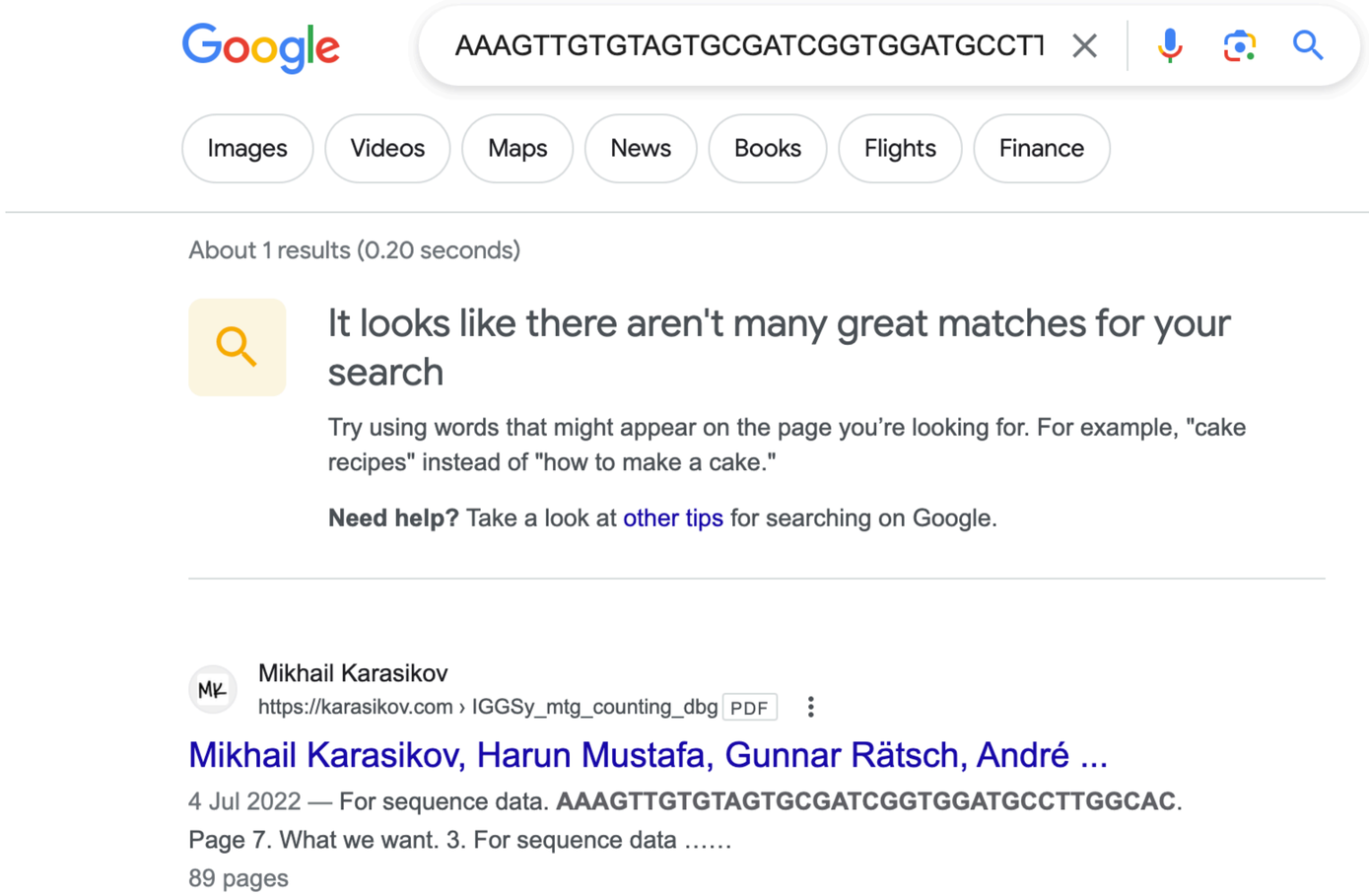
Existing search engines do not work (yet)



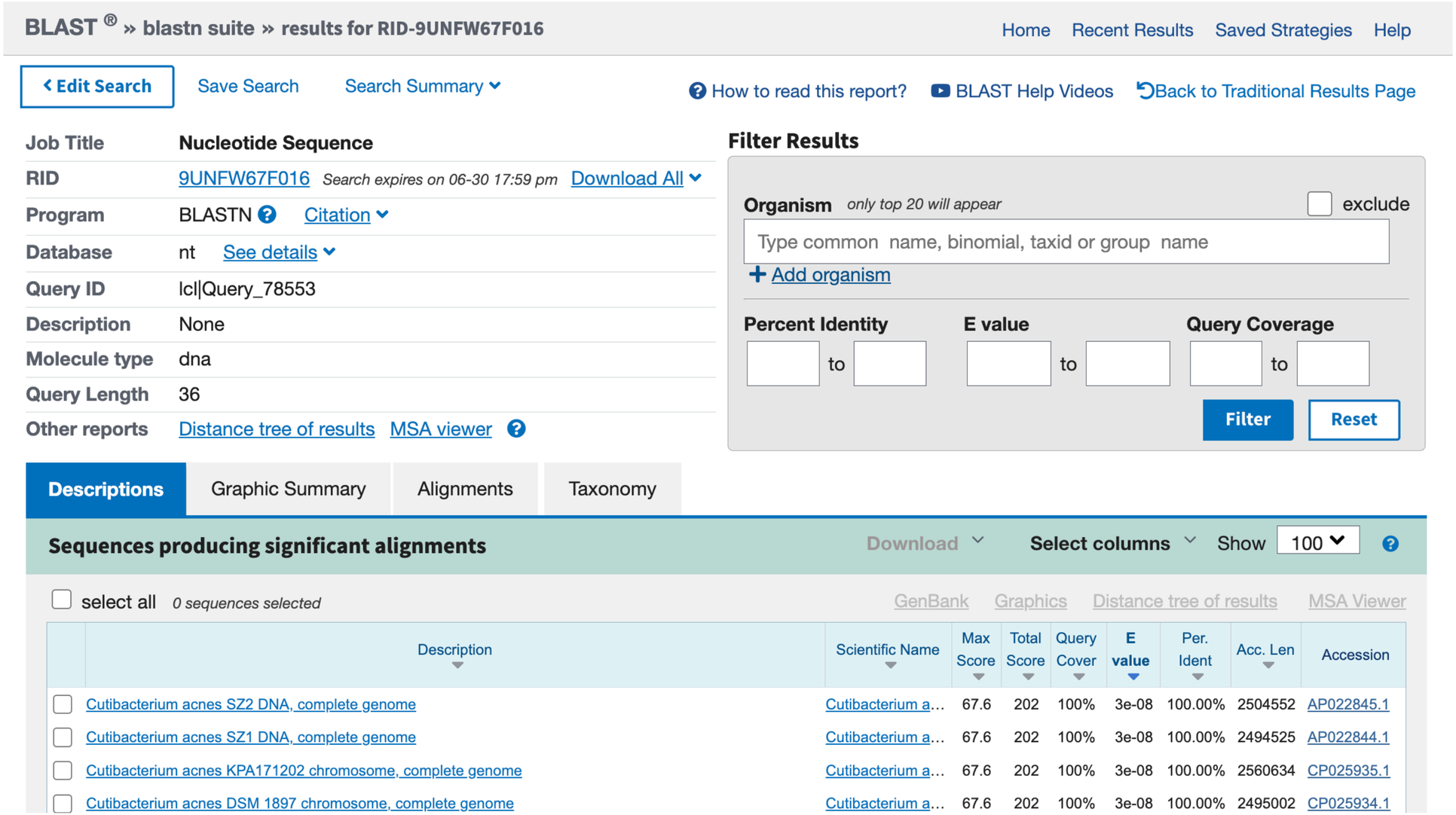
Existing search engines do not work (yet)



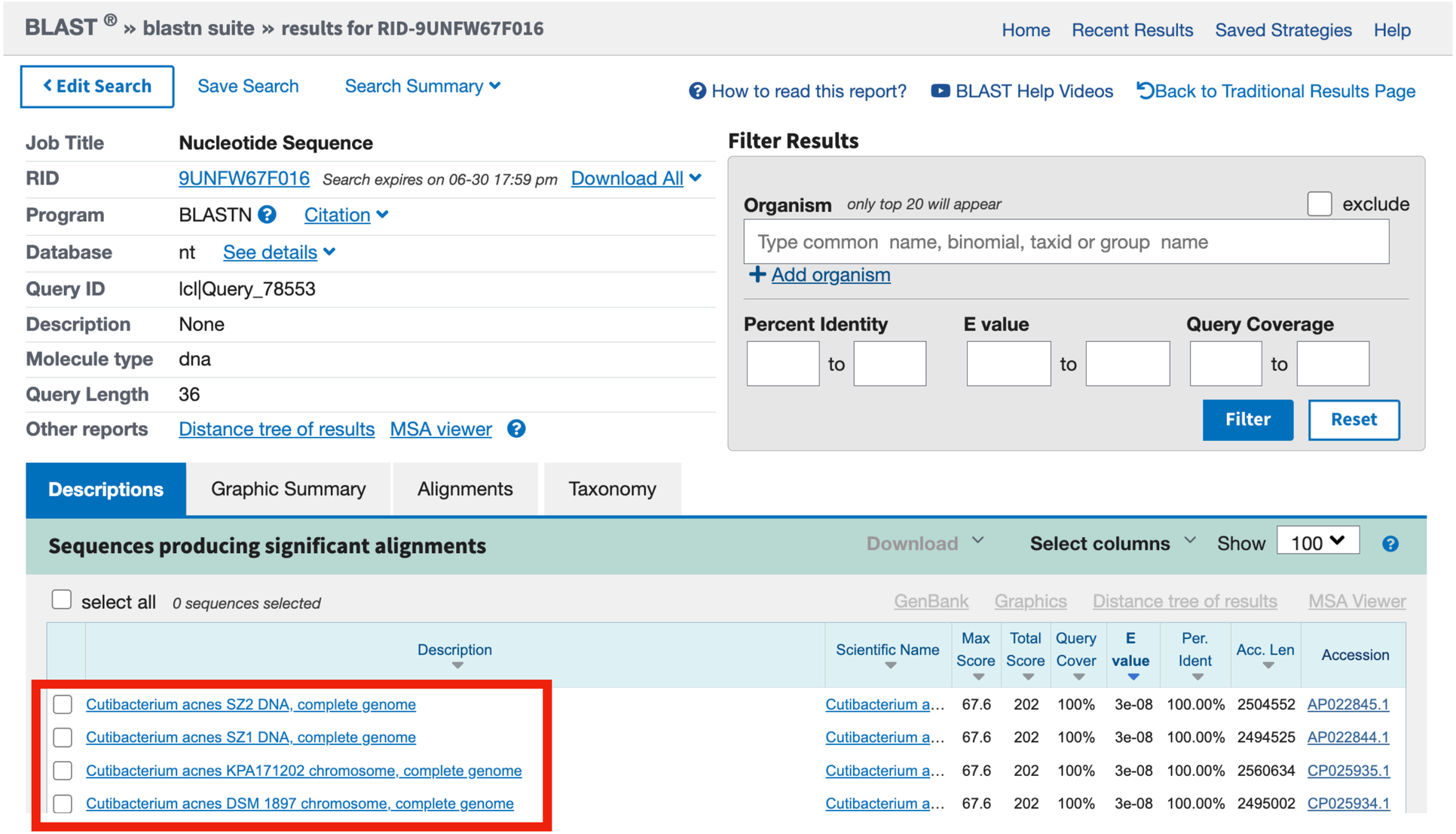
Existing search engines do not work (yet)



BLAST finds reference sequences

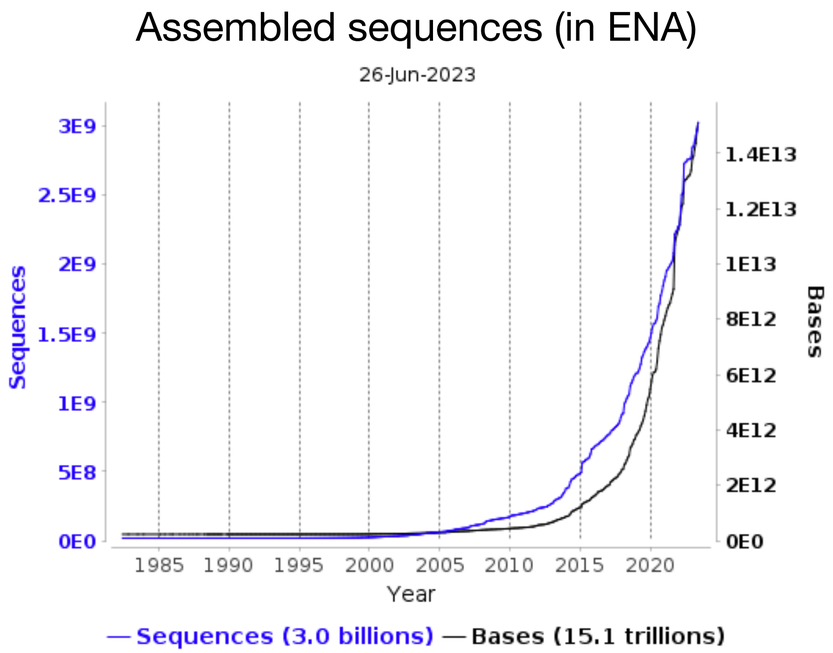


BLAST finds reference sequences



assembled sequences

Growth of sequence archives

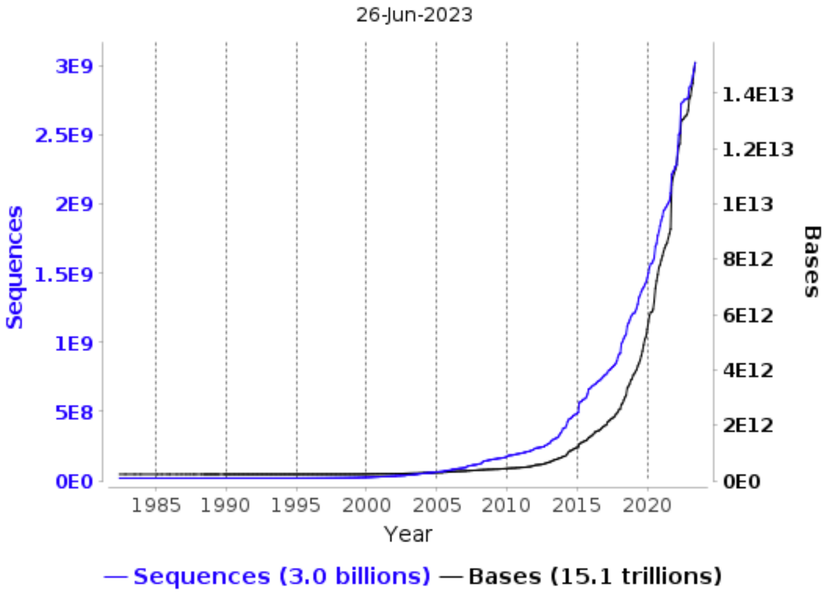


ENA
European Nucleotide Archive

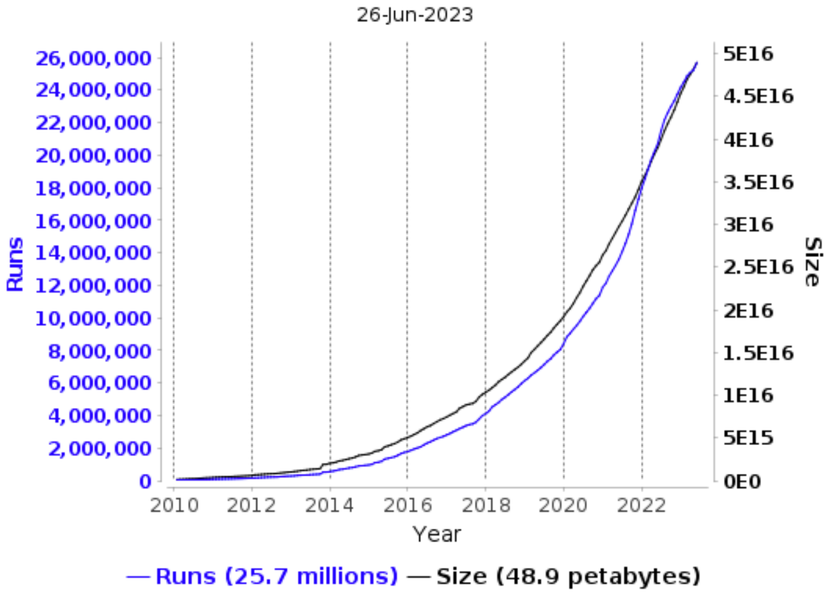
Data source: https://www.ebi.ac.uk/ena/browser/about/statistics

Growth of sequence archives

Assembled sequences (in ENA)



Raw reads (in ENA)

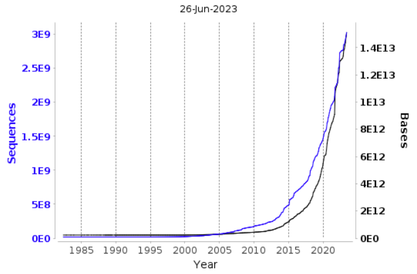


ENA
European Nucleotide Archive

Data source: https://www.ebi.ac.uk/ena/browser/about/statistics

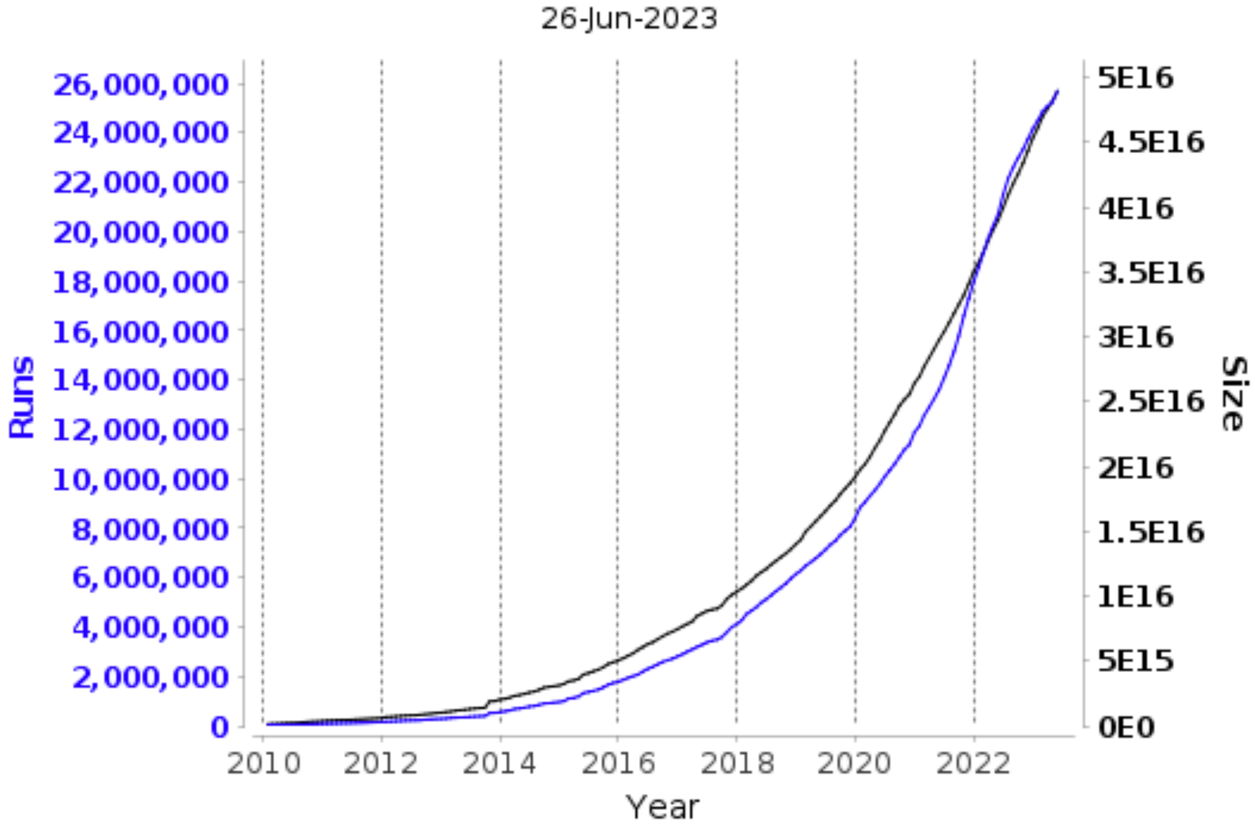
Growth of sequence archives

Assembled sequences (in ENA)



15 Tbp

Raw reads (in ENA)



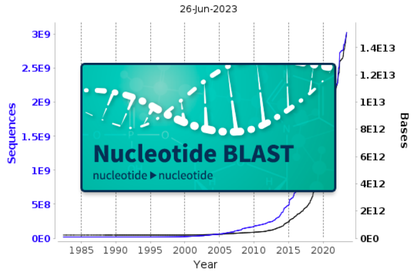
> 40,000 Tbp

ENA
European Nucleotide Archive

Data source: https://www.ebi.ac.uk/ena/browser/about/statistics

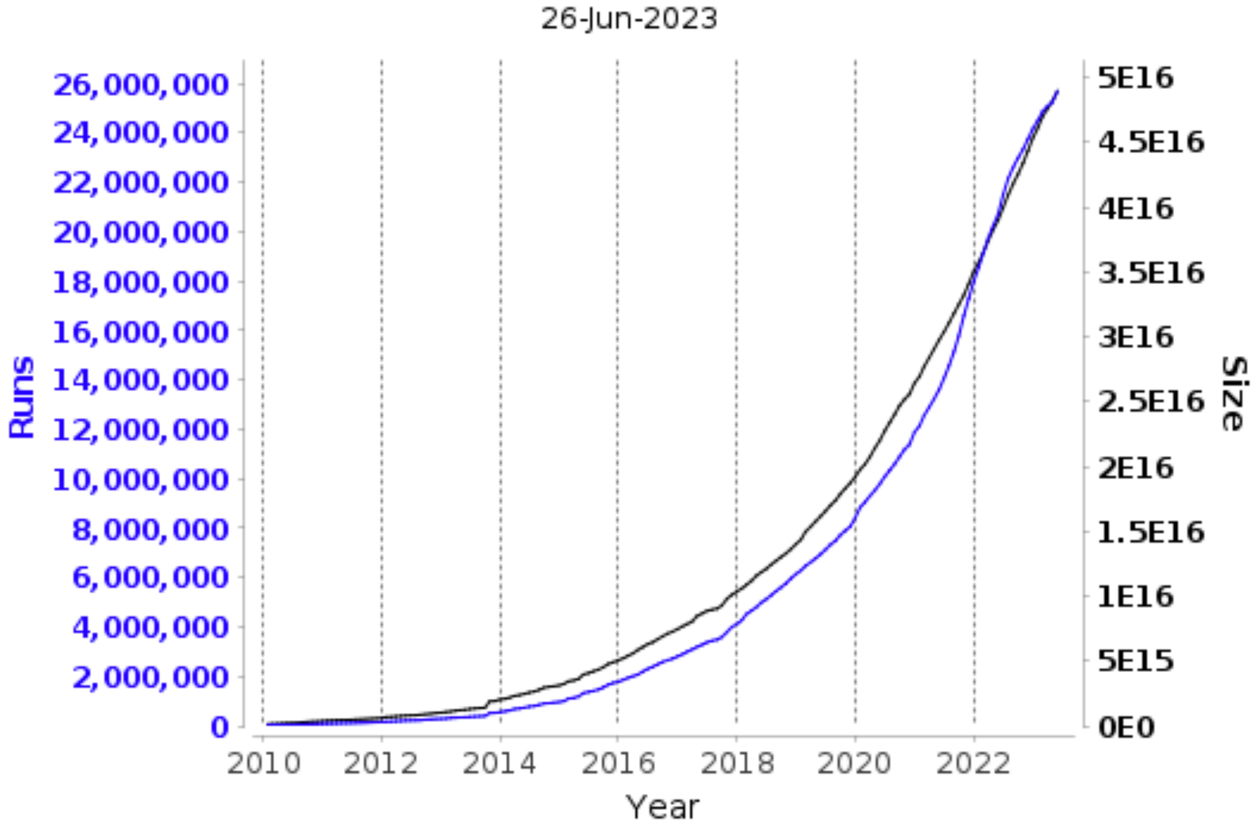
Growth of sequence archives

Assembled sequences (in ENA)



15 Tbp

Raw reads (in ENA)



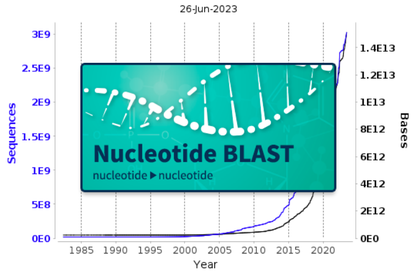
> 40,000 Tbp

ENA
European Nucleotide Archive

Data source: https://www.ebi.ac.uk/ena/browser/about/statistics

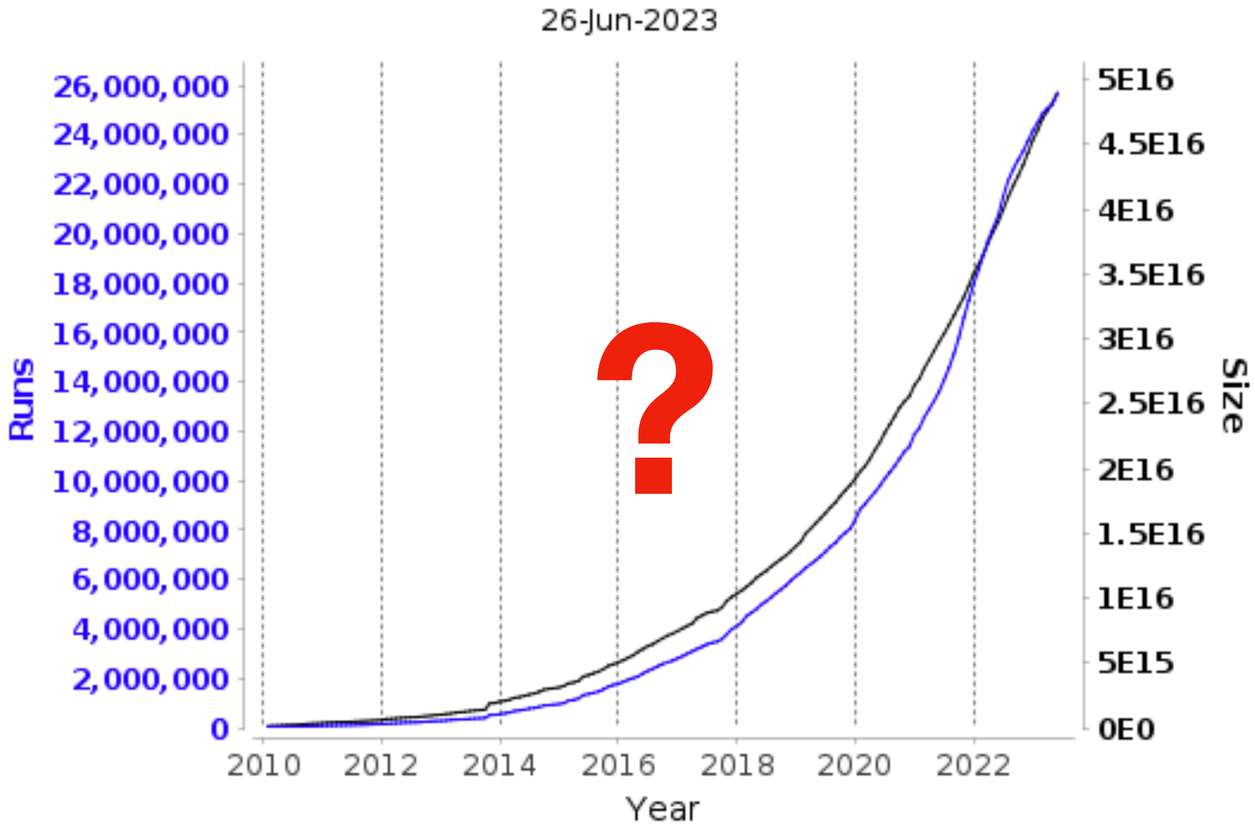
Growth of sequence archives

Assembled sequences (in ENA)



15 Tbp

Raw reads (in ENA)



> 40,000 Tbp

ENA
European Nucleotide Archive

Data source: https://www.ebi.ac.uk/ena/browser/about/statistics

Indexing sequences

- Why not use methods from typical search engines?

Indexing sequences

- Why not use methods from typical search engines?
  - sequences lack structure of natural language (“AAAGTTGTGT" vs. “cake recipes”)
  - enormous amount of data (> 50 PB)
  - special metrics for inexact search (sequence alignment)

Indexing sequences

- Why not use methods from typical search engines?
  - sequences lack structure of natural language (“AAAGTTGTGT" vs. “cake recipes”)
  - enormous amount of data (> 50 PB)
  - special metrics for inexact search (sequence alignment)
- Burrows-Wheeler transform and FM-index do not scale

Indexing sequences

- Why not use methods from typical search engines?
  - sequences lack structure of natural language (“AAAGTTGTGT" vs. “cake recipes”)
  - enormous amount of data (> 50 PB)
  - special metrics for inexact search (sequence alignment)
- Burrows-Wheeler transform and FM-index do not scale
- Need something simpler

Indexing sequences

- Why not use methods from typical search engines?
  - sequences lack structure of natural language (“AAAGTTGTGT" vs. “cake recipes”)
  - enormous amount of data (> 50 PB)
  - special metrics for inexact search (sequence alignment)
- Burrows-Wheeler transform and FM-index do not scale
- Need something simpler

  De Bruijn graph

Background

De Bruijn graphs have been introduced to the field by Pevzner in 1989

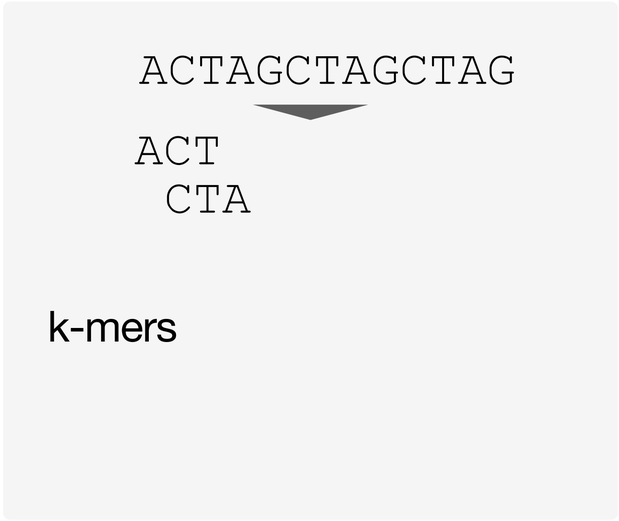
Background

De Bruijn graphs have been introduced to the field by Pevzner in 1989

```
ACTAGCTAGCTAG
```

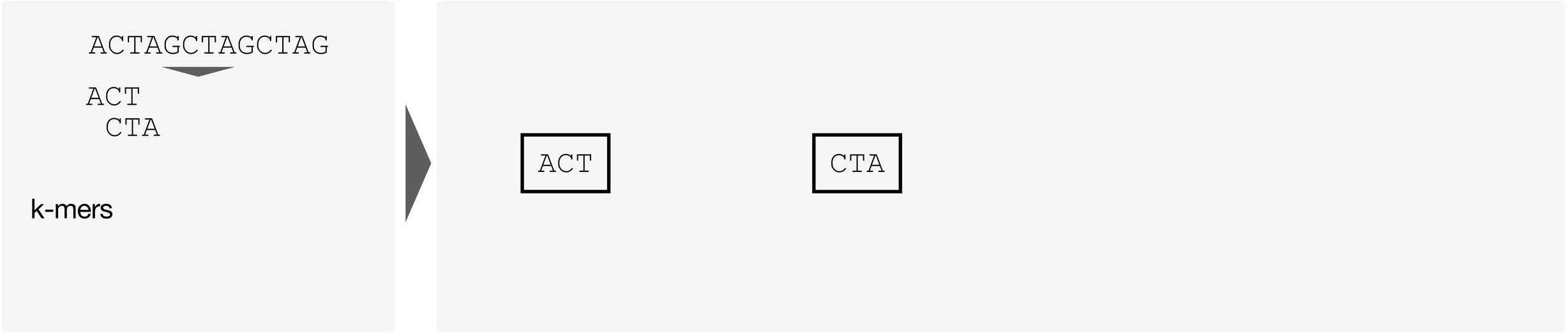
Background

De Bruijn graphs have been introduced to the field by Pevzner in 1989



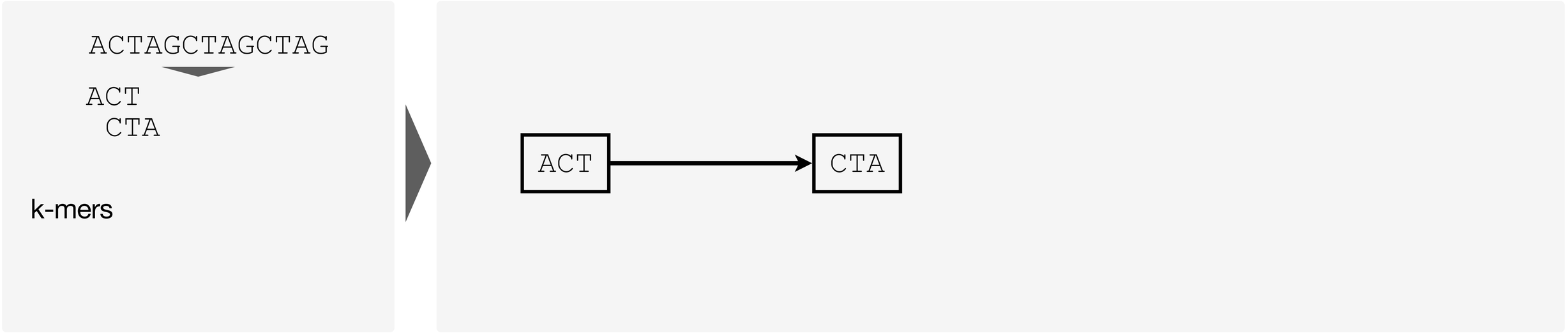
Background

De Bruijn graphs have been introduced to the field by Pevzner in 1989



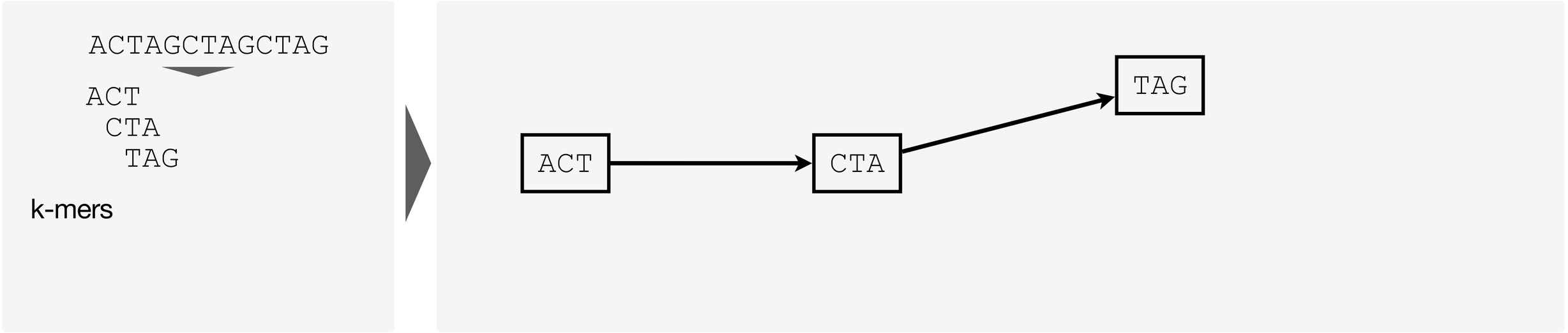
Background

De Bruijn graphs have been introduced to the field by Pevzner in 1989



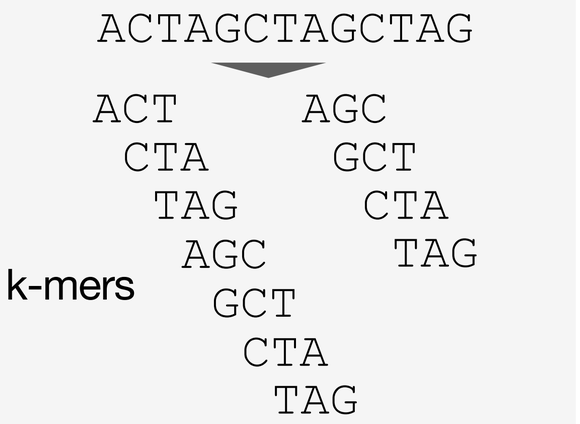
Background

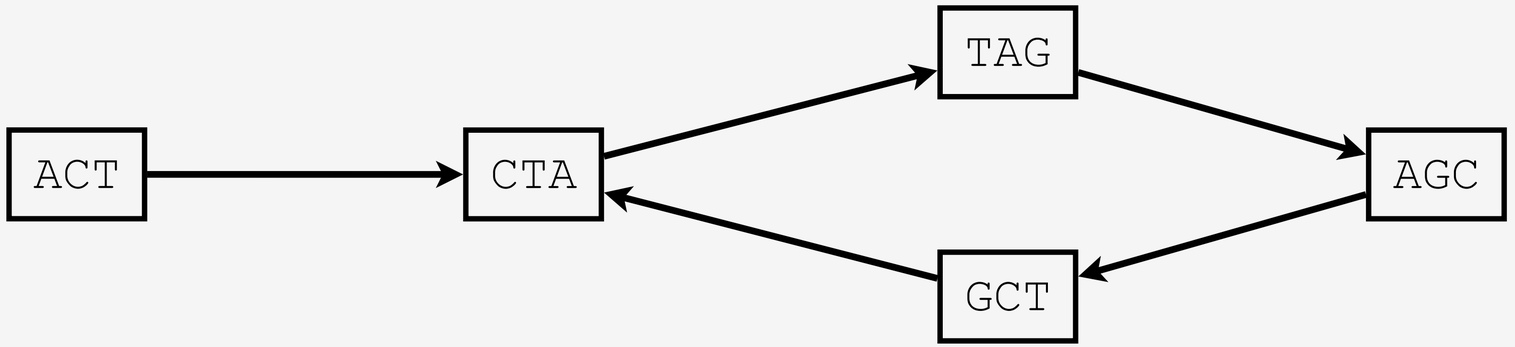
De Bruijn graphs have been introduced to the field by Pevzner in 1989



Background

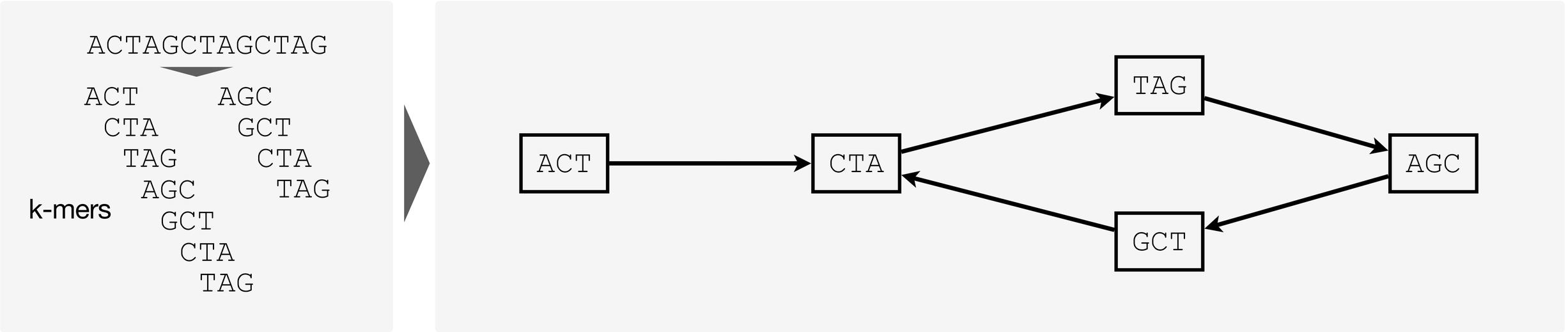
De Bruijn graphs have been introduced to the field by Pevzner in 1989





Background

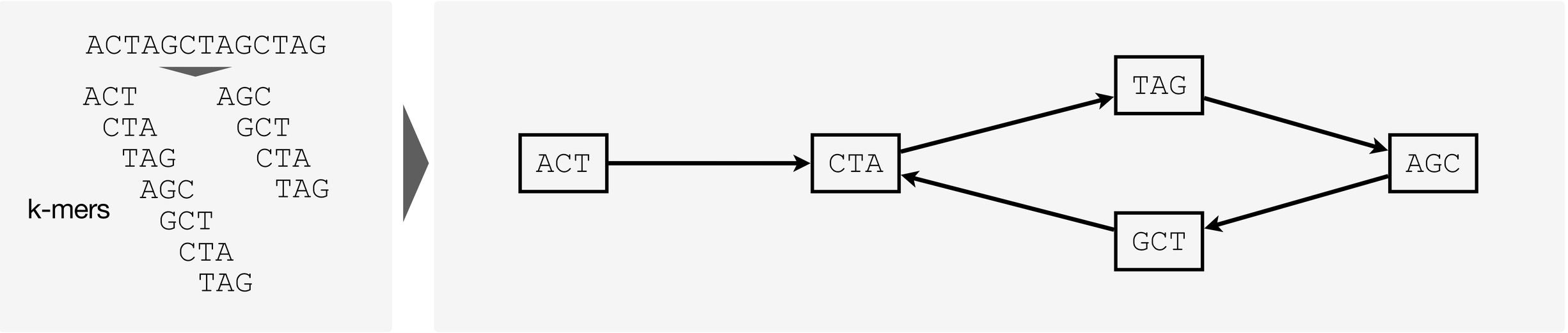
De Bruijn graphs have been introduced to the field by Pevzner in 1989



- Originally employed for ***de novo* assembly**

Background

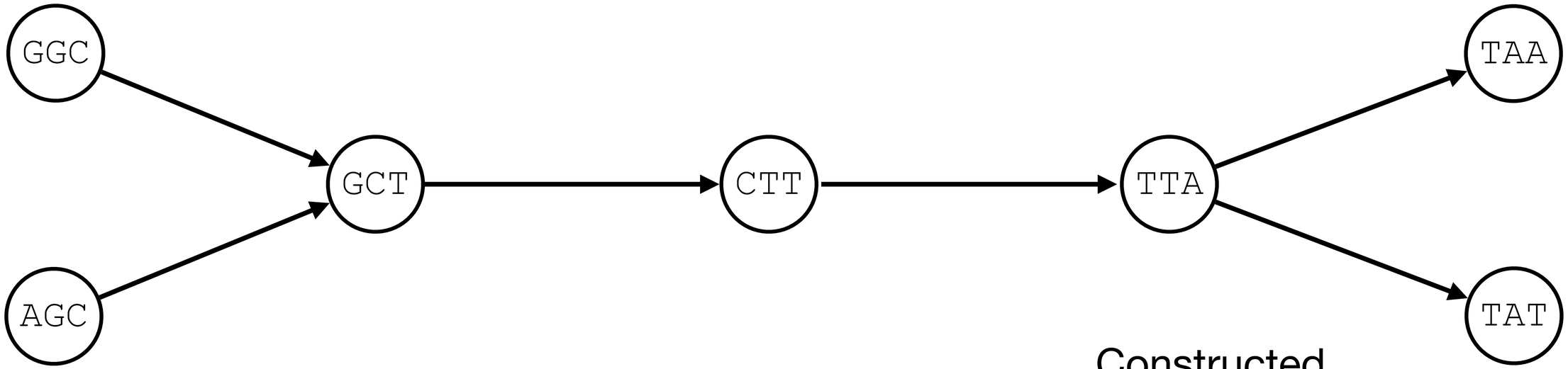
De Bruijn graphs have been introduced to the field by Pevzner in 1989



- Originally employed for ***de novo* assembly**
  - [Pevzner, 1989]
  - [Idury, Waterman, 1995]
  - [Pevzner, Tang, Waterman, 2001]
  - [Zerbino, Birney, 2008]

Background

Annotated De Bruijn graphs



Constructed
from sequences

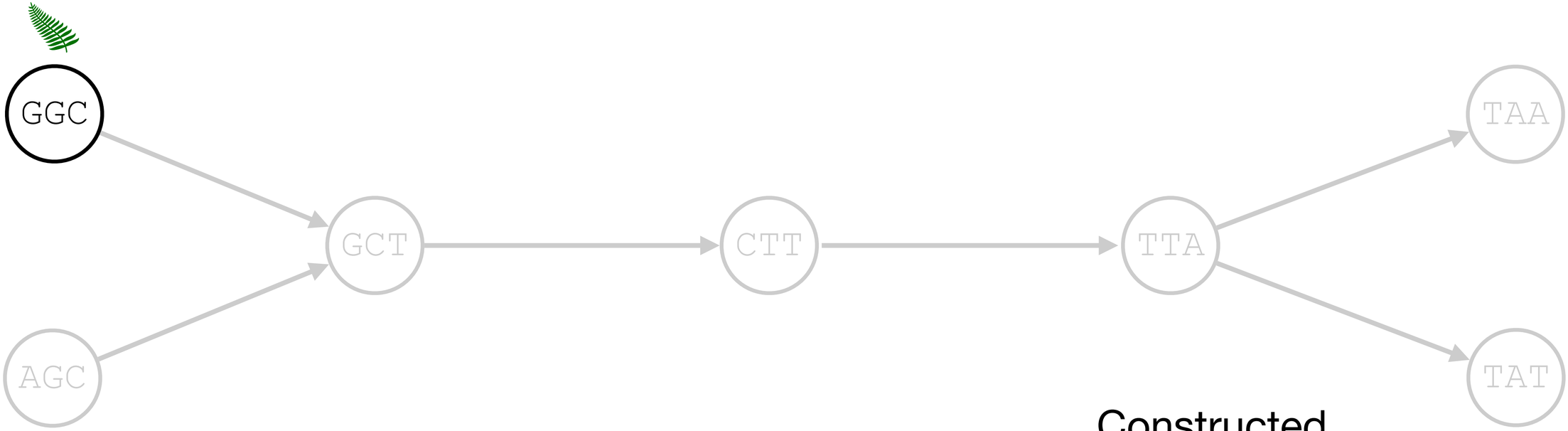
L1: GGCTTAT

L2: AGCTTAA

L3: TTAA

Background

Annotated De Bruijn graphs



Constructed
from sequences

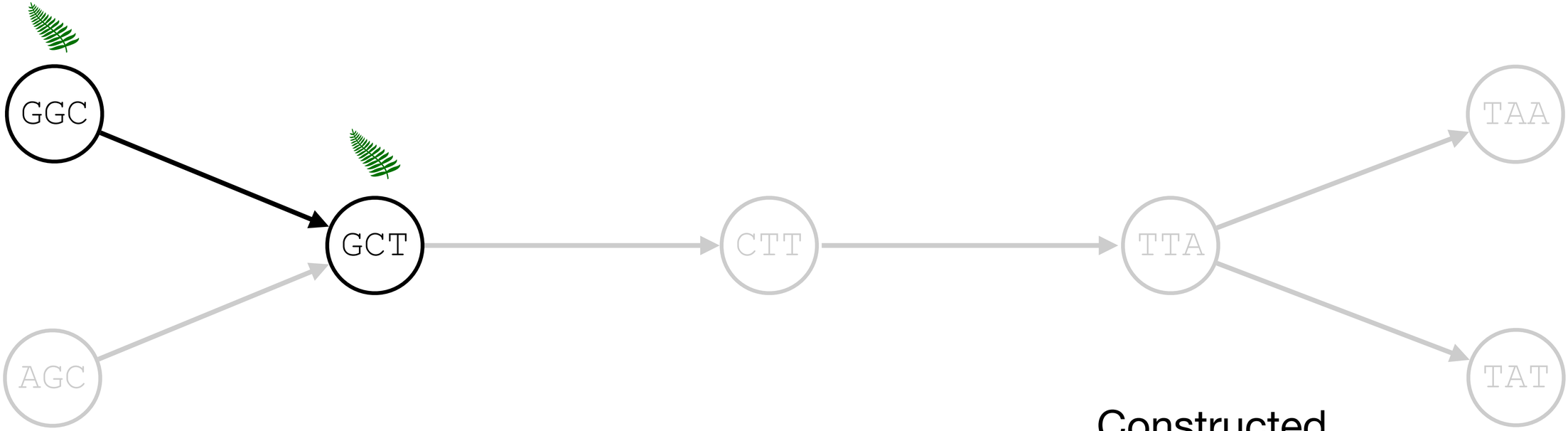
L1: GGCTTAT

L2: AGCTTAA

L3: TTAA

Background

Annotated De Bruijn graphs



Constructed
from sequences

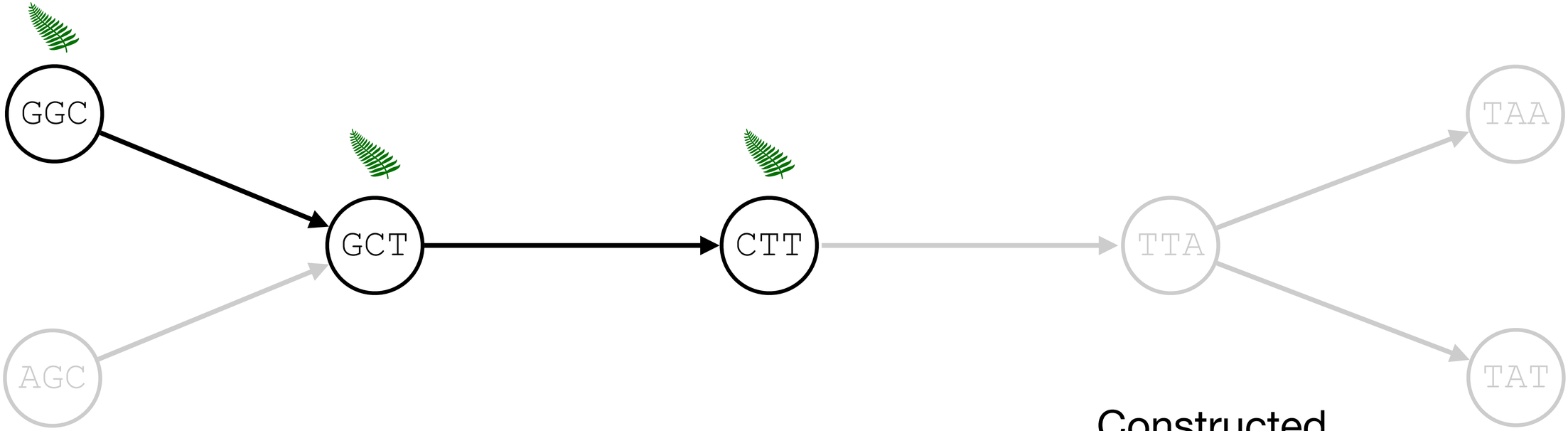
L1: GGCTTAT

L2: AGCTTAA

L3: TTAA

Background

Annotated De Bruijn graphs



Constructed
from sequences

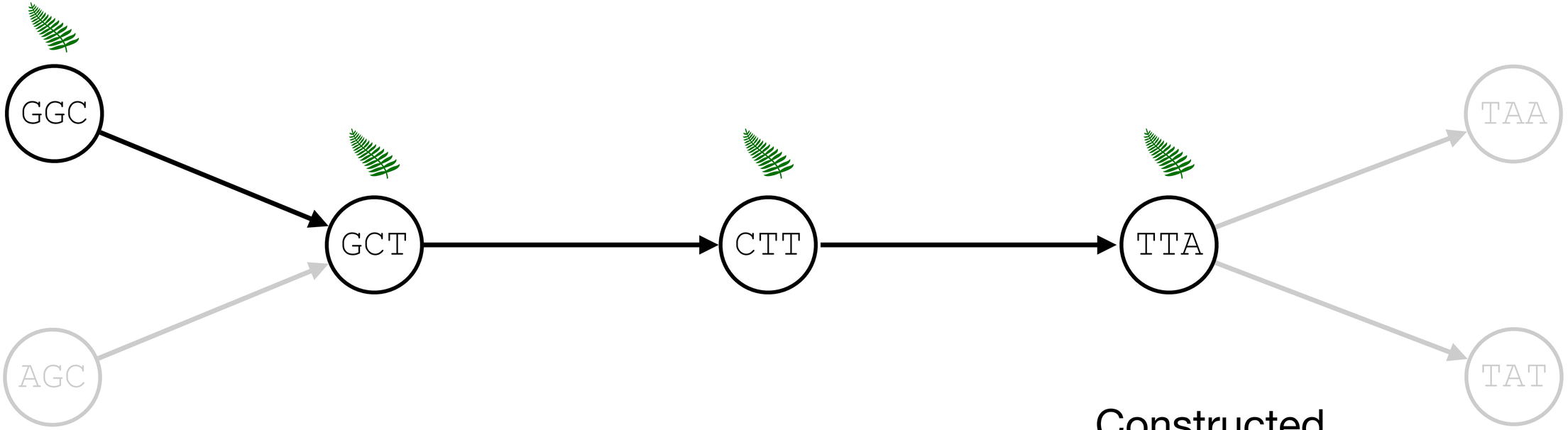
L1: GGCTTAT

L2: AGCTTAA

L3: TTAA

Background

Annotated De Bruijn graphs



Constructed
from sequences

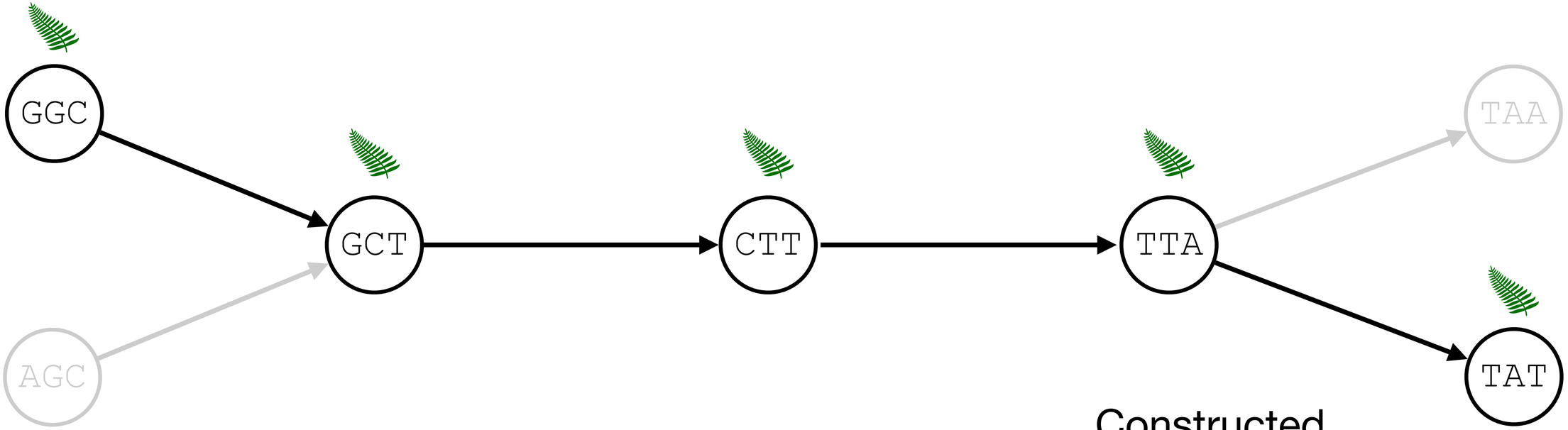
L1: GGCTTAT

L2: AGCTTAA

L3: TTAA

Background

Annotated De Bruijn graphs



Constructed
from sequences

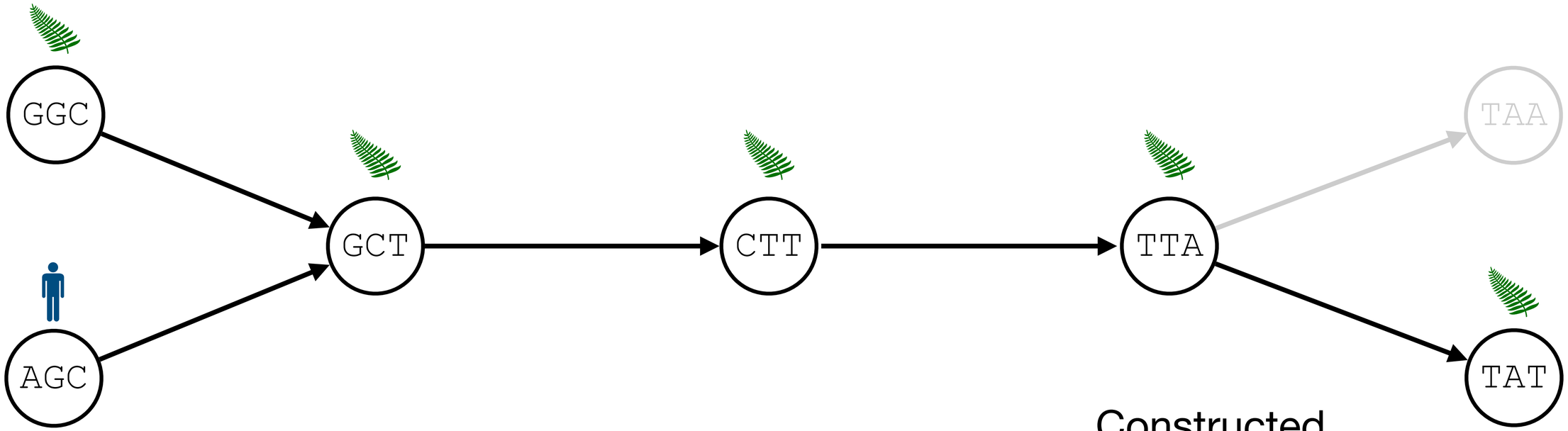
L1: GGCTTAT

L2: AGCTTAA

L3: TTAA

Background

Annotated De Bruijn graphs



Constructed
from sequences

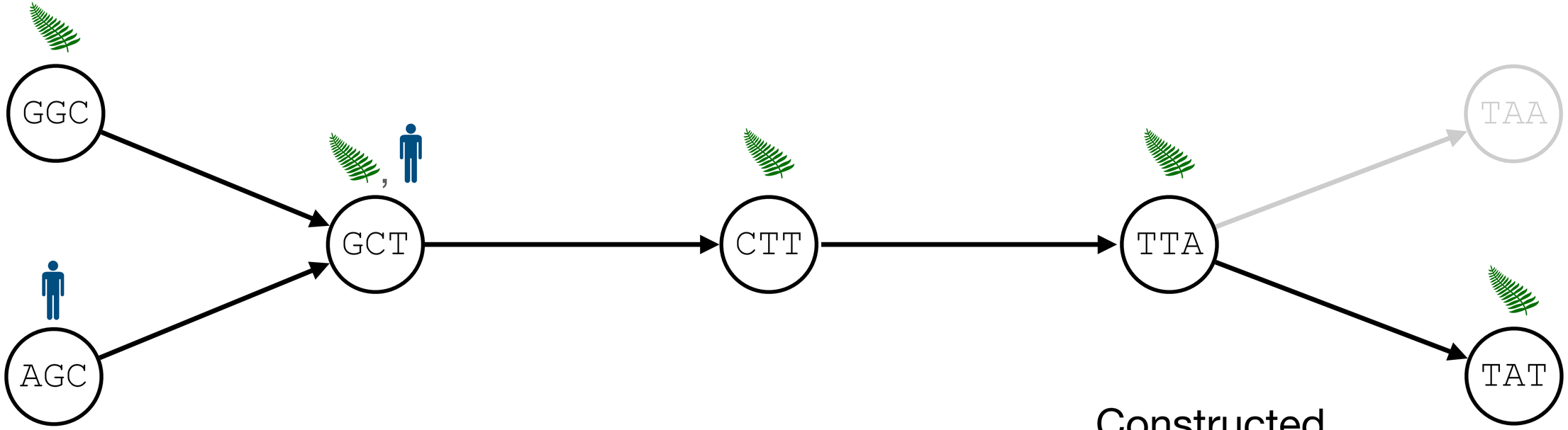
L1: GGCTTAT

L2: AGCTTAA

L3: TTAA

Background

Annotated De Bruijn graphs



Constructed
from sequences

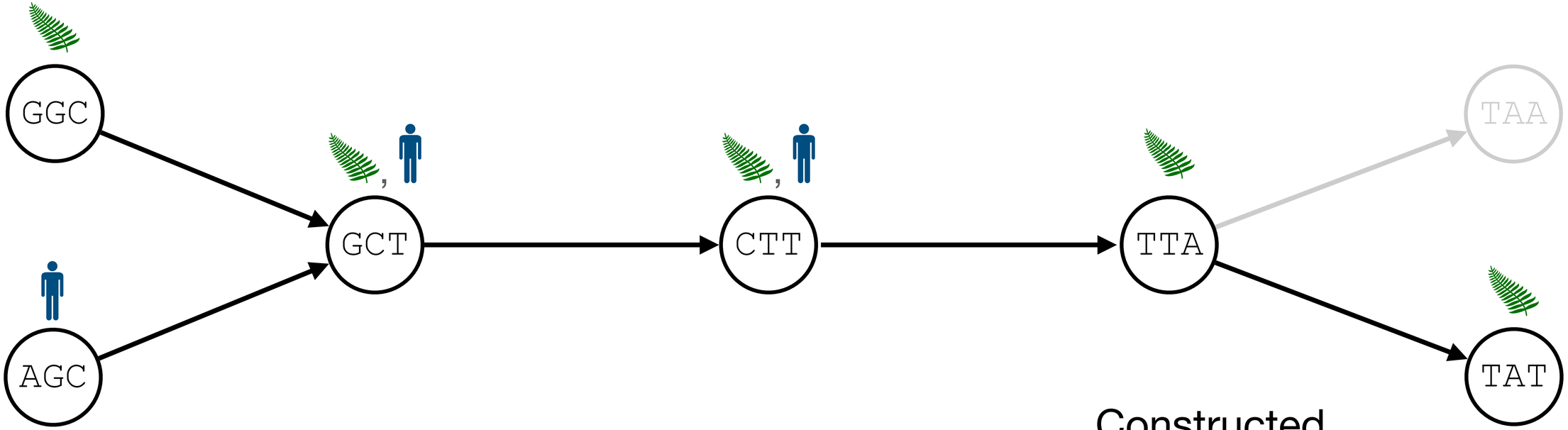
L1: GGCTTAT

L2: AGCTTAA

L3: TTAA

Background

Annotated De Bruijn graphs



Constructed
from sequences

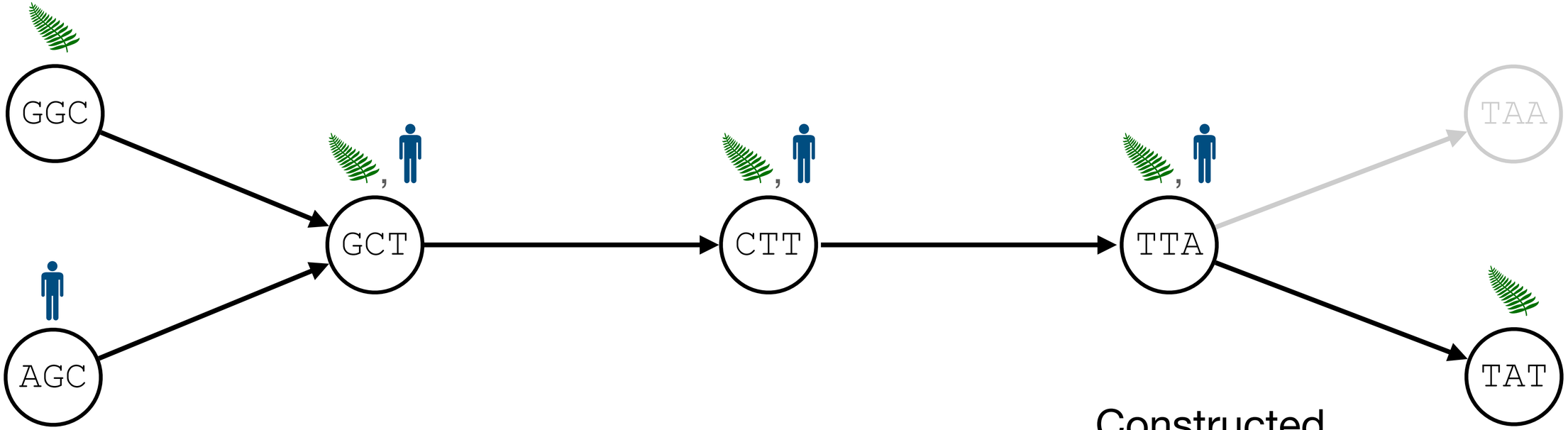
L1: GGCTTAT

L2: AGCTTAA

L3: TTAA

Background

Annotated De Bruijn graphs



Constructed
from sequences

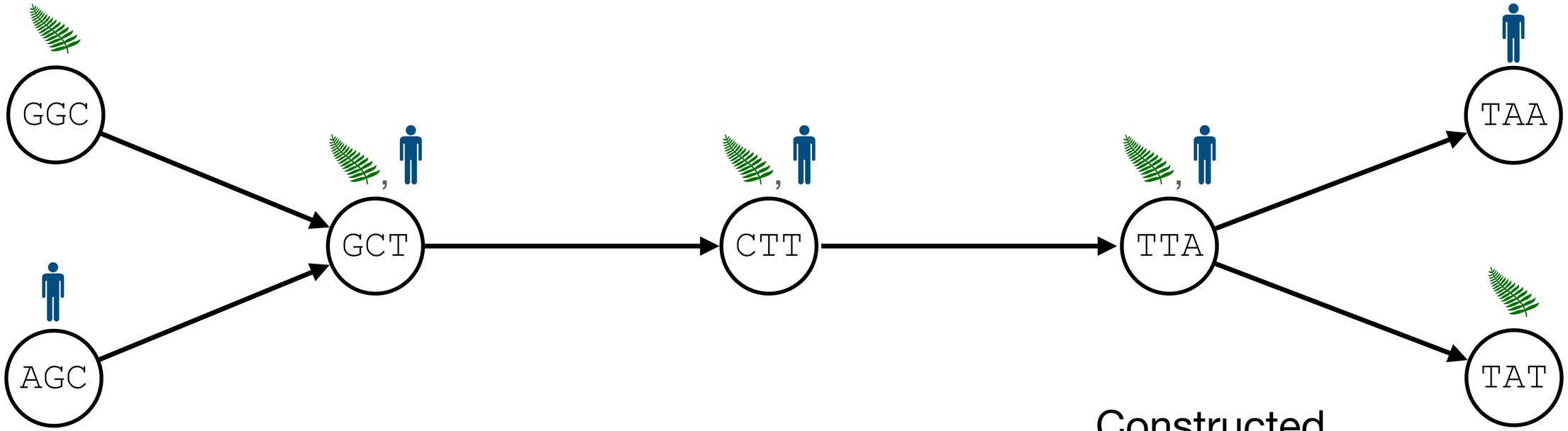
L1: GGCTTAT

L2: AGCTTAA

L3: TTAA

Background

Annotated De Bruijn graphs



Constructed
from sequences

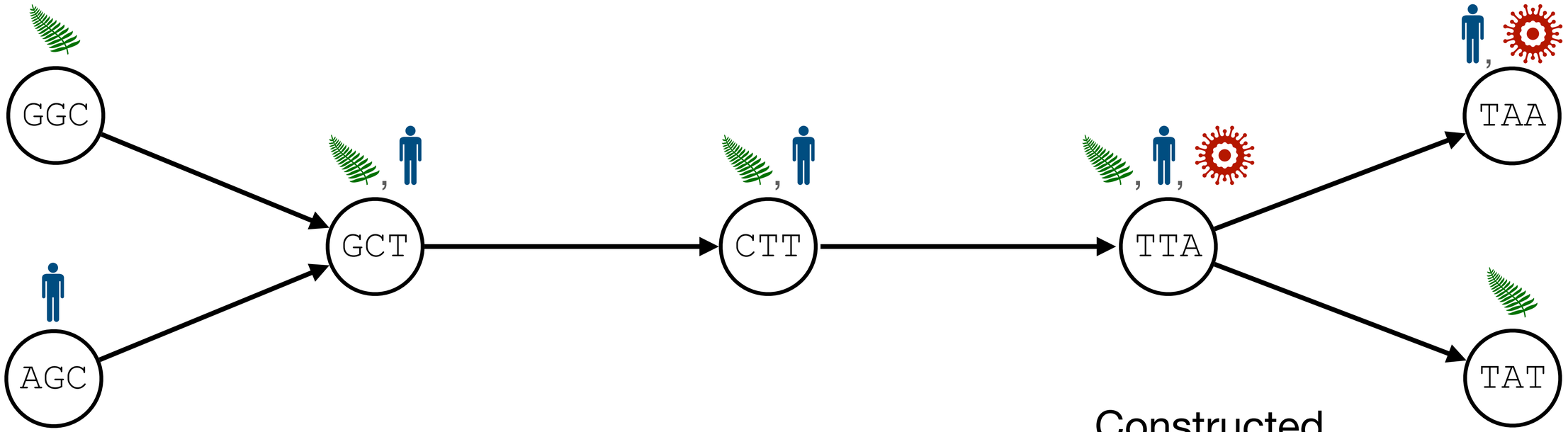
L1: GGCTTAT

L2: AGCTTAA

L3: TTAA

Background

Annotated De Bruijn graphs



Constructed
from sequences

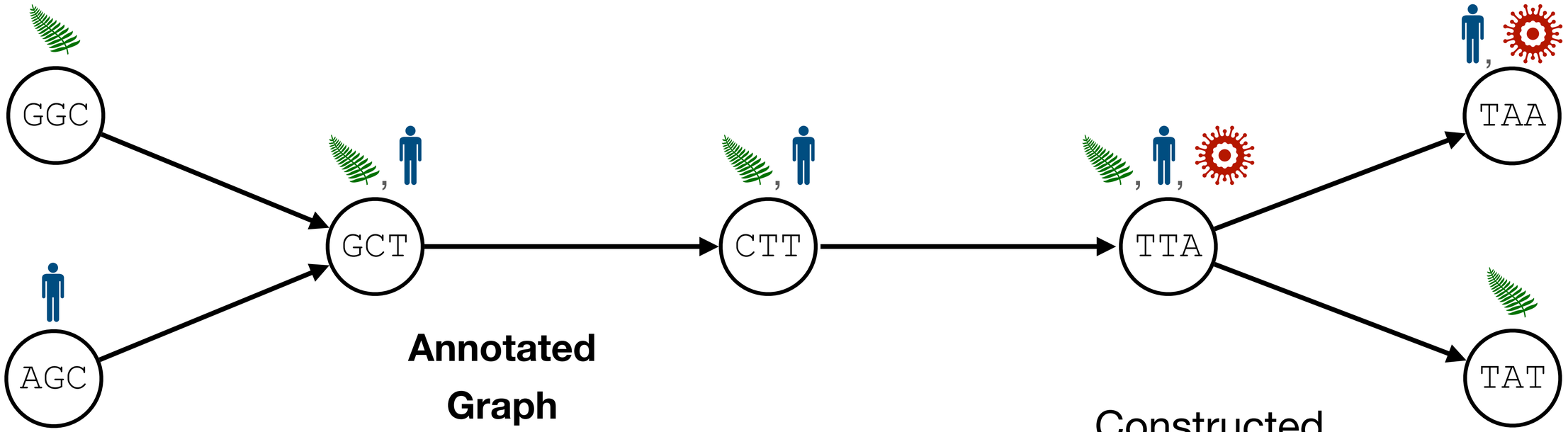
L1: GGCTTAT

L2: AGCTTAA

L3: TTAA

Background

Annotated De Bruijn graphs



Annotated Graph

Constructed from sequences

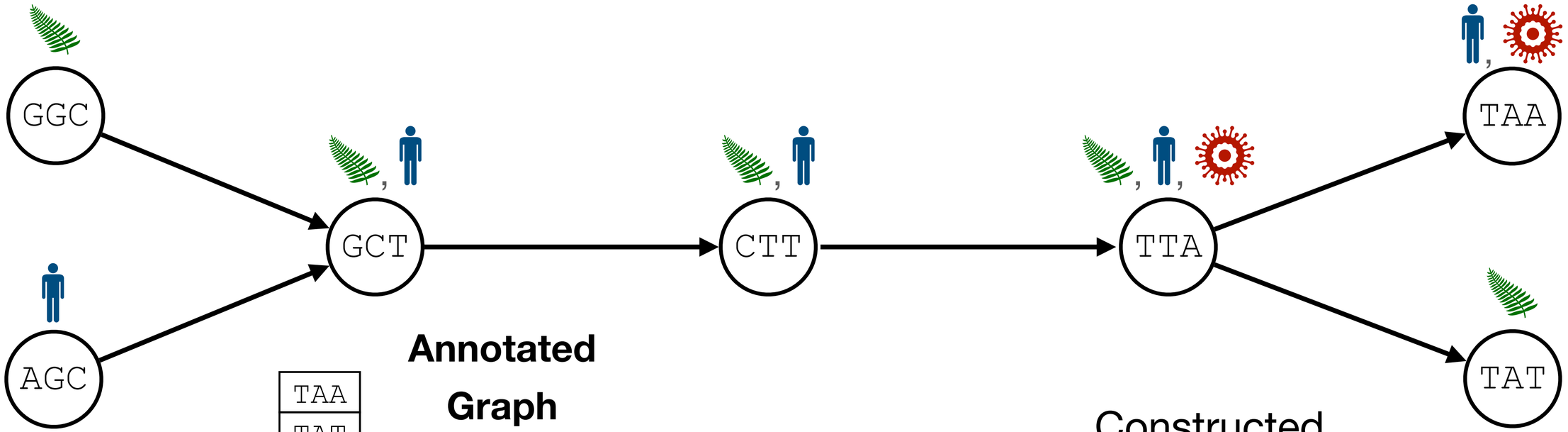
L1: GGCTTAT

L2: AGCTTAA

L3: TTAA

Background

Annotated De Bruijn graphs



Annotated Graph

TAA
TAT
GCT
AGC
GGC
CTT
TTA

k-mer dictionary

Constructed from sequences

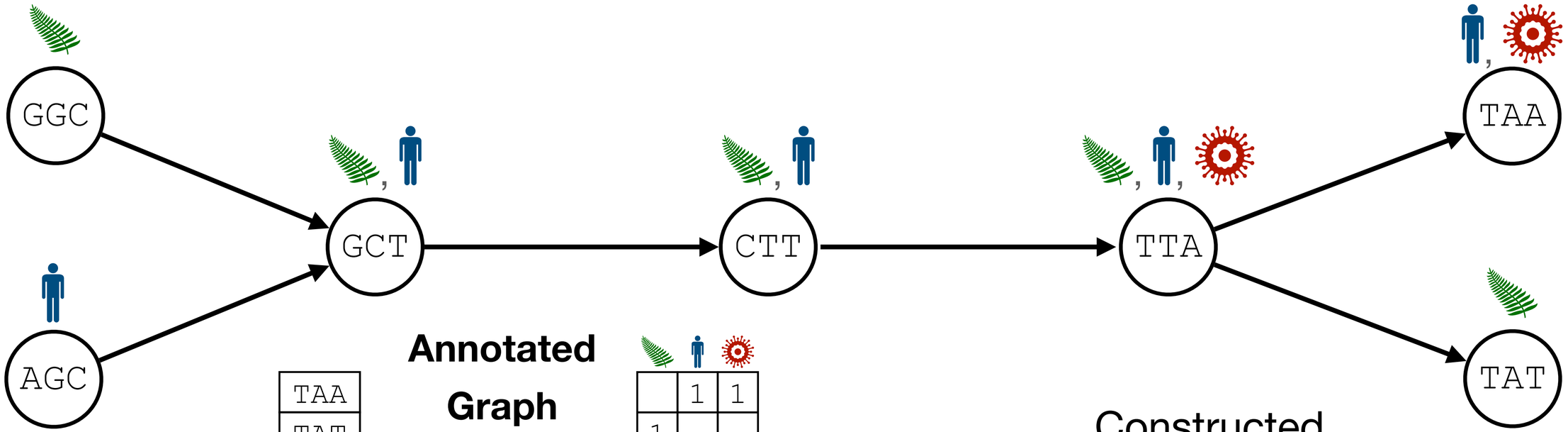
L1: GGCTTAT

L2: AGCTTAA

L3: TTAA

Background

Annotated De Bruijn graphs



TAA
TAT
GCT
AGC
GGC
CTT
TTA

k-mer dictionary

Annotated Graph

	1	1
1		
1	1	
	1	
1		
1	1	
1	1	1

graph annotation

Constructed from sequences

L1: GGCTTAT

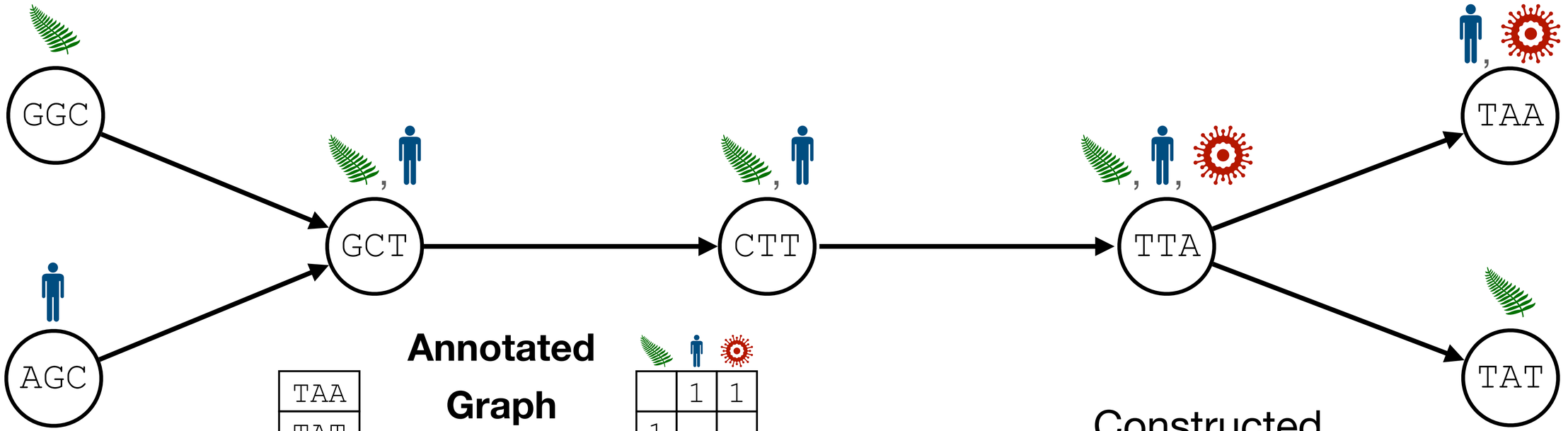
L2: AGCTTAA

L3: TTAA

Background

Annotated De Bruijn graphs

- [Iqbal, Caccamo, Turner, Flicek, McVean, 2012]
- [Muggli *et al.*, 2017]



Annotated Graph

TAA
TAT
GCT
AGC
GGC
CTT
TTA

k-mer dictionary

	1	1
1		
1	1	
	1	
1		
1	1	
1	1	1

graph annotation

Constructed from sequences

L1: GGCTTAT

L2: AGCTTAA

L3: TTAA

Indexing workflow

```
>smp_1
ACGTAC
ACGC
CGTAC
```

```
>smp_2
ACGAA
ACGTAC
ACG
```

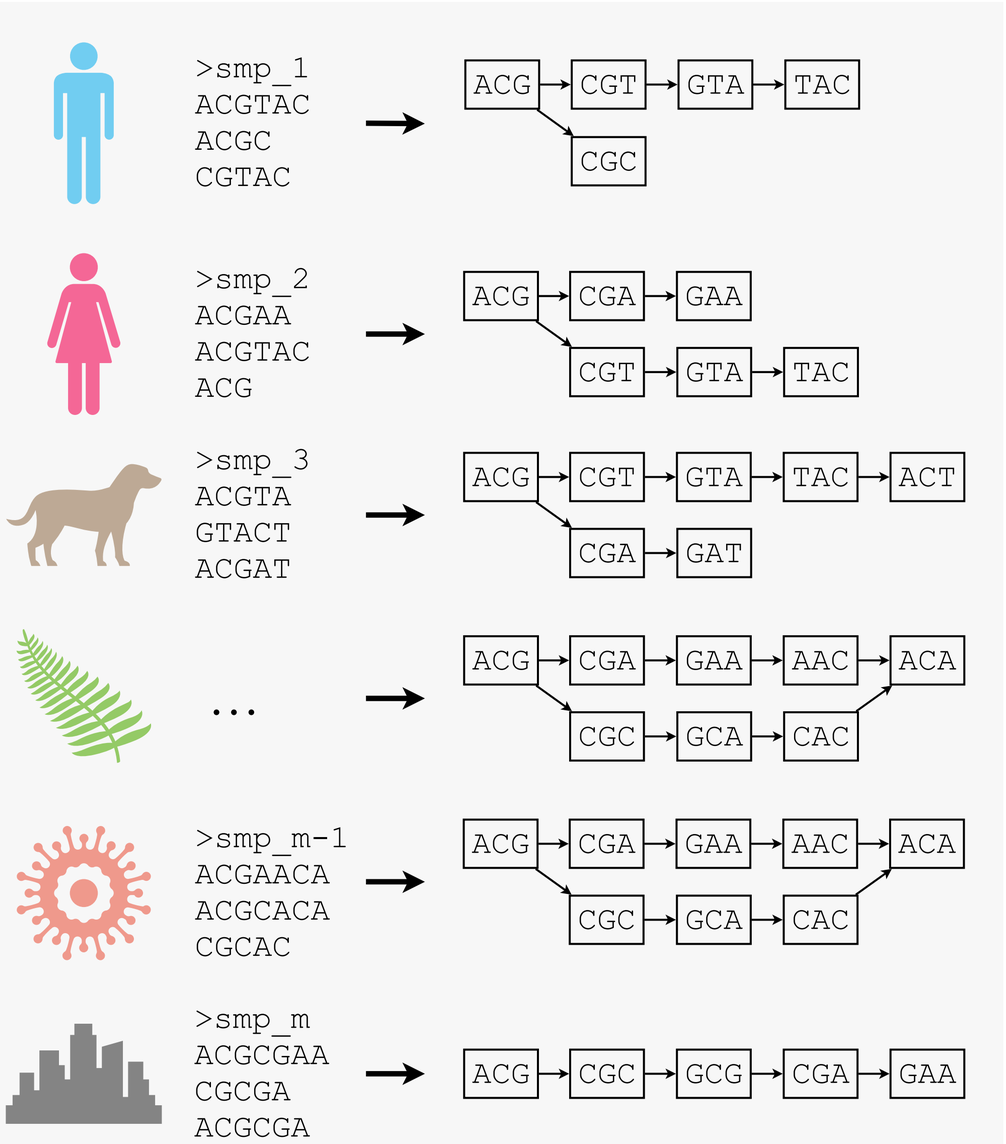
```
>smp_3
ACGTA
GTACT
ACGAT
```

...

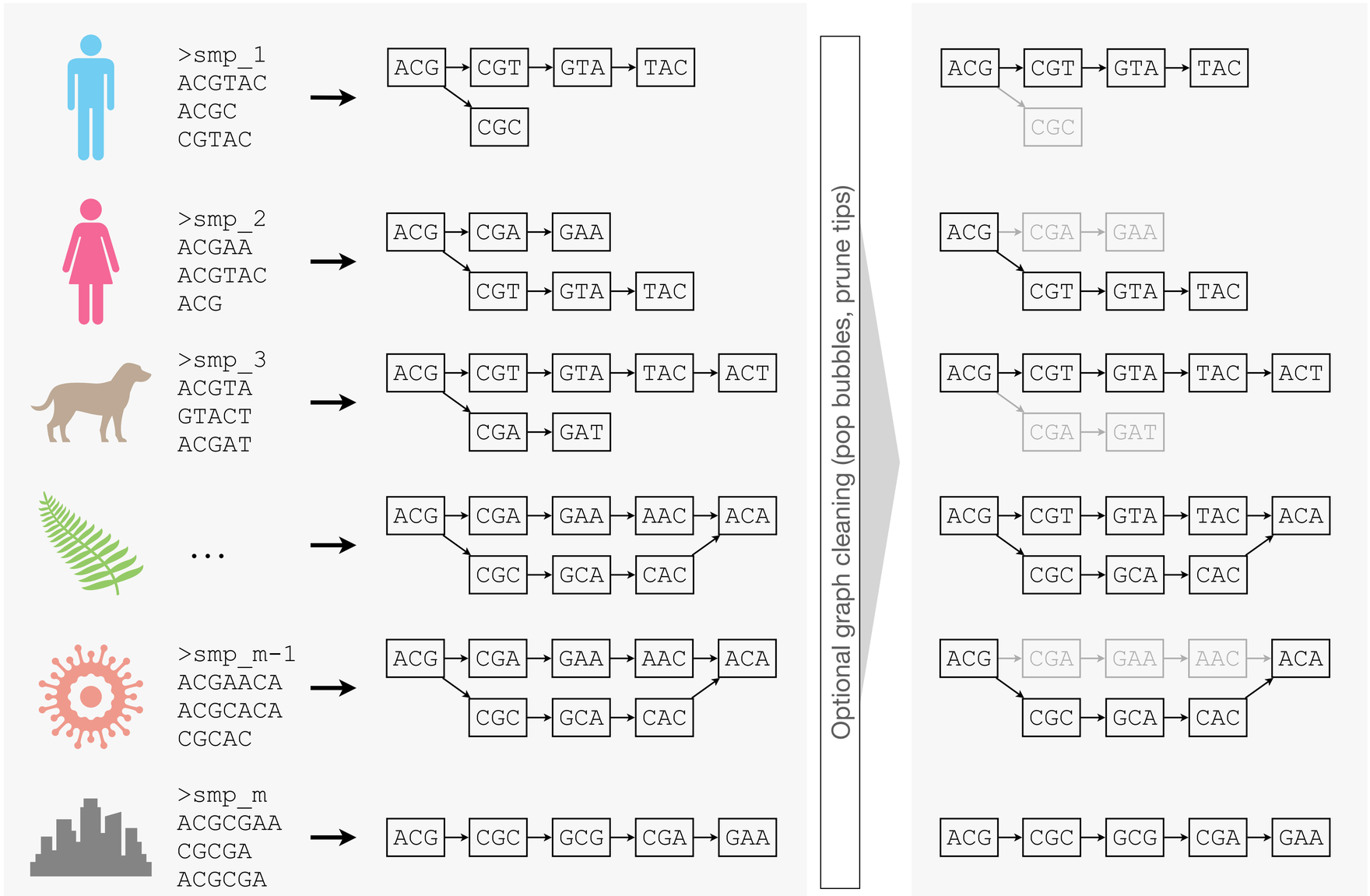
```
>smp_m-1
ACGAACA
ACGCACA
CGCAC
```

```
>smp_m
ACGCGAA
CGCGA
ACGCGA
```

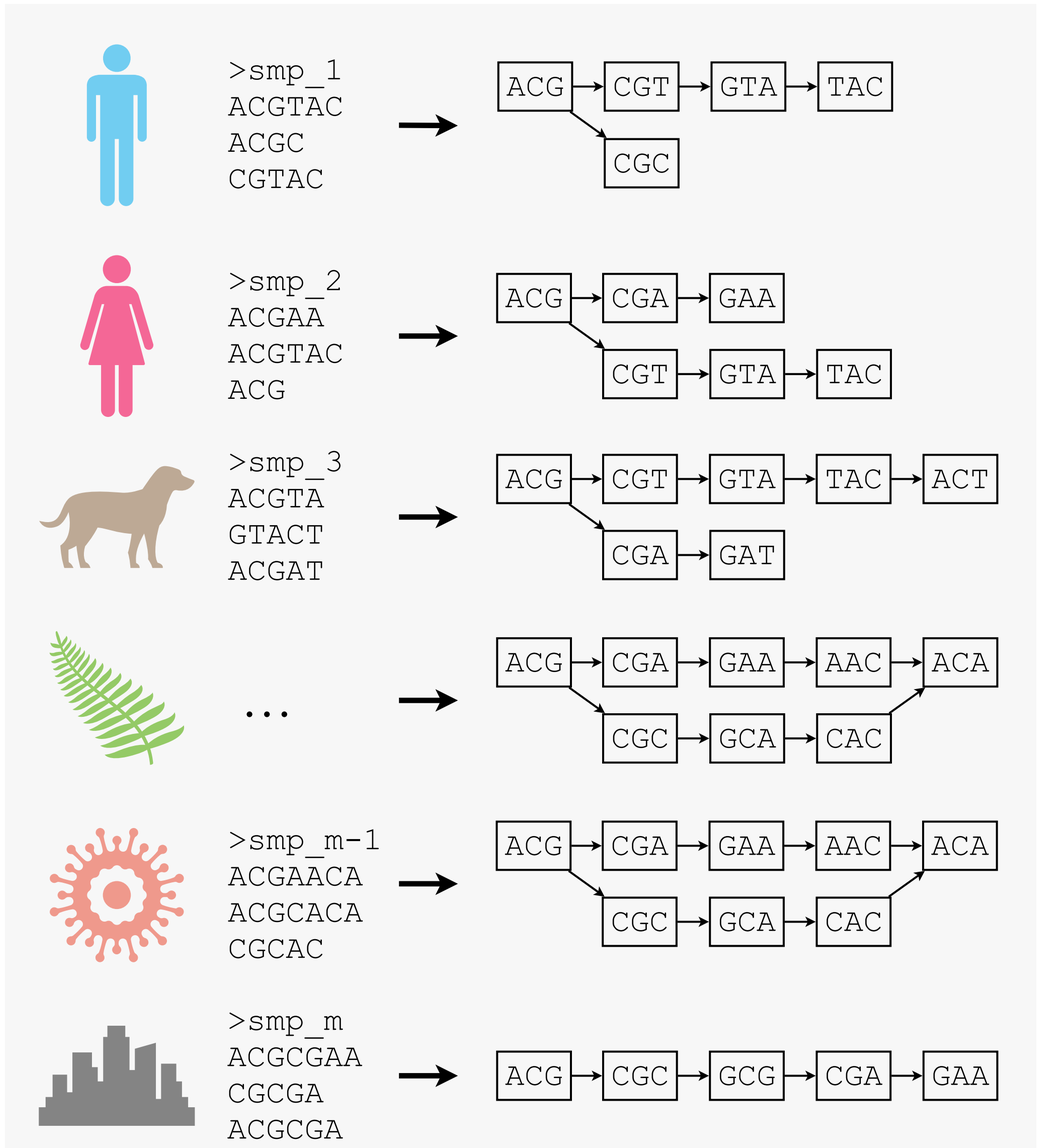
Indexing workflow



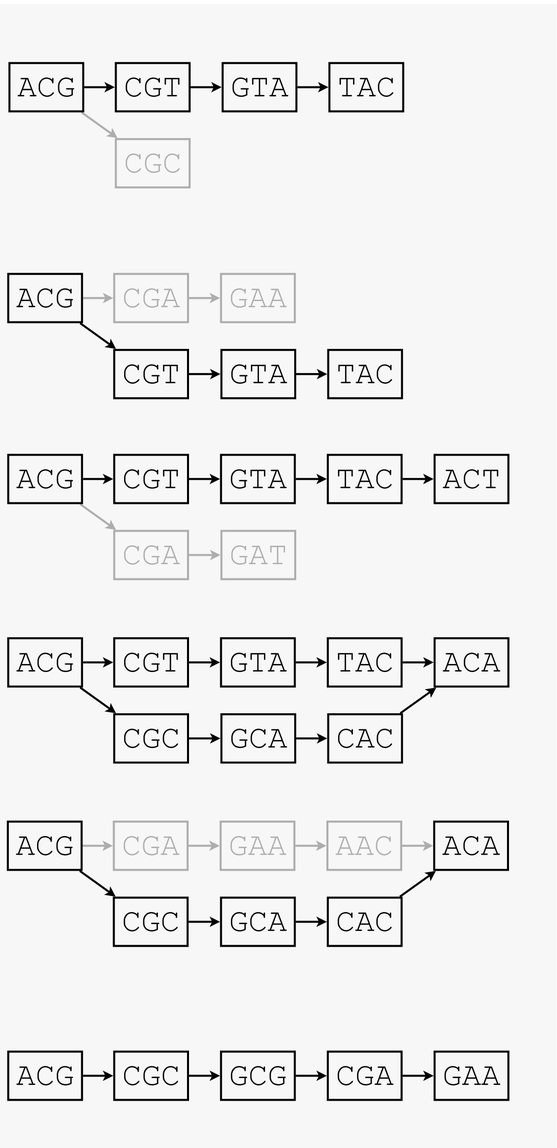
Indexing workflow



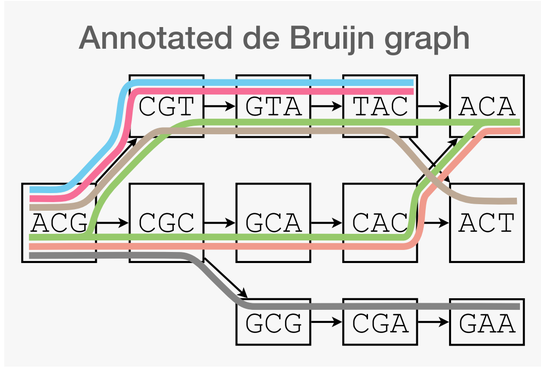
Indexing workflow



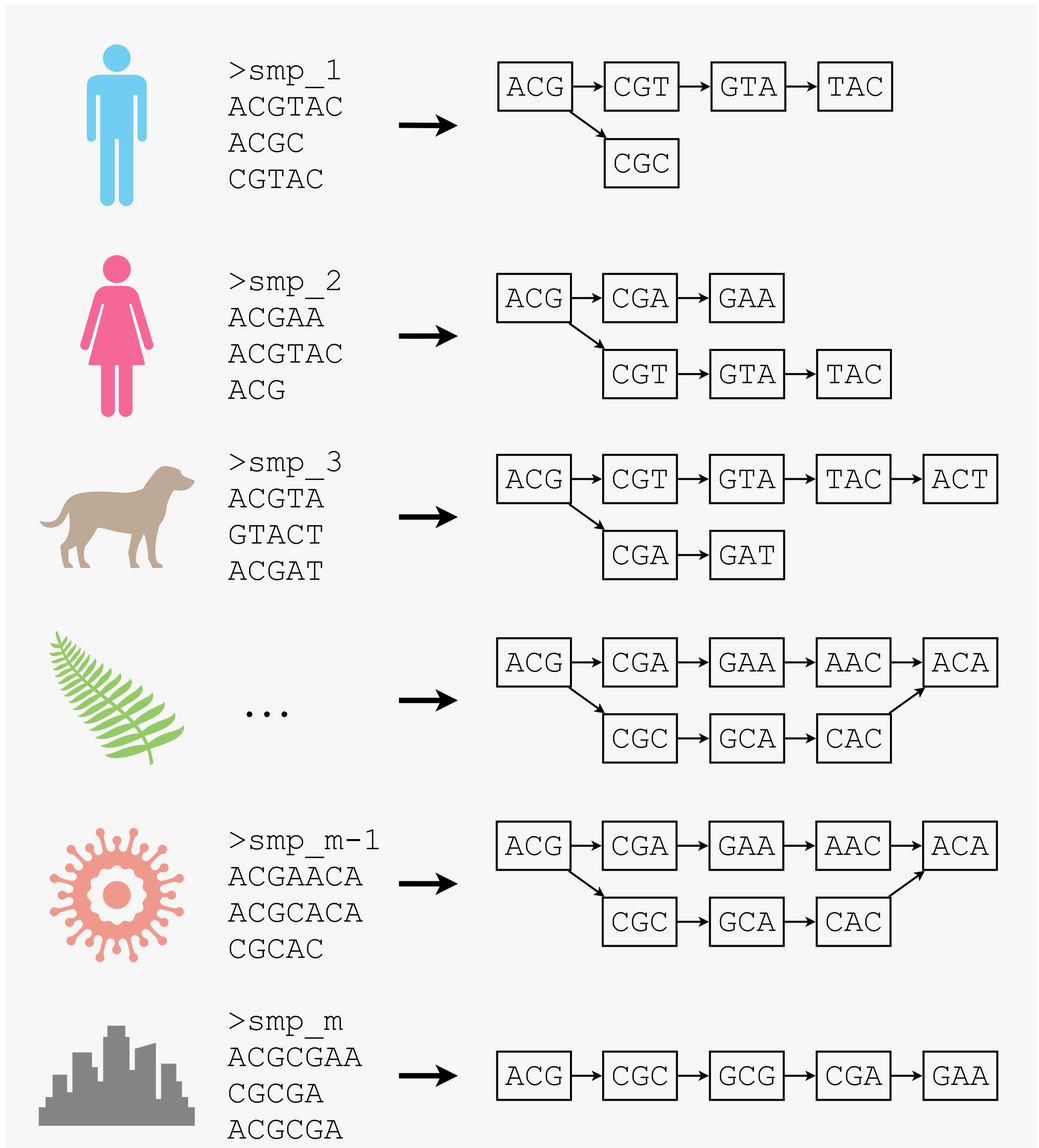
Optional graph cleaning (pop bubbles, prune tips)



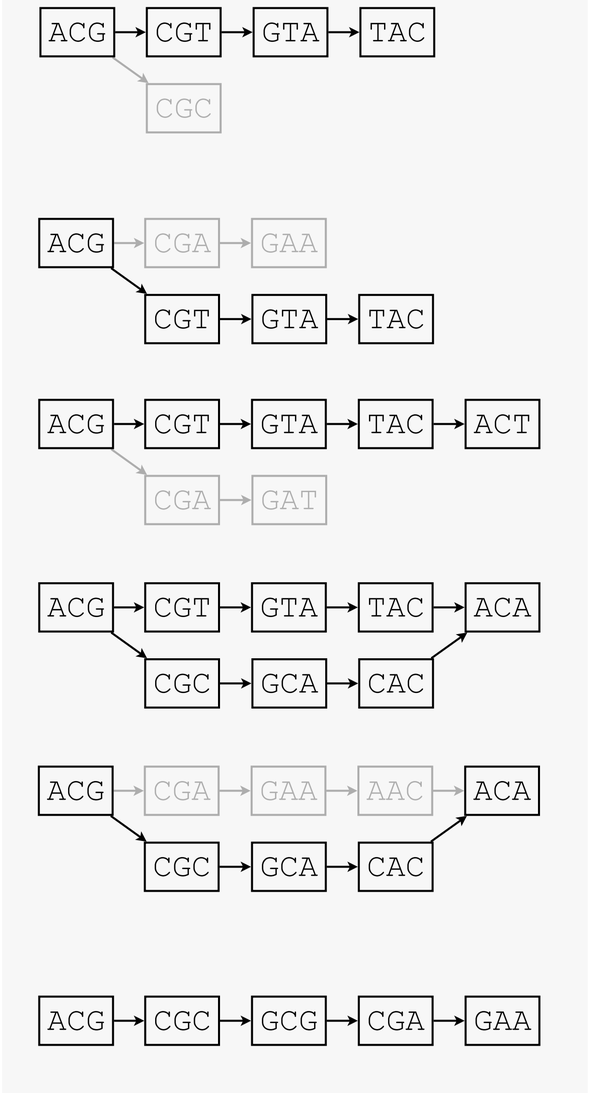
Merge into a joint graph



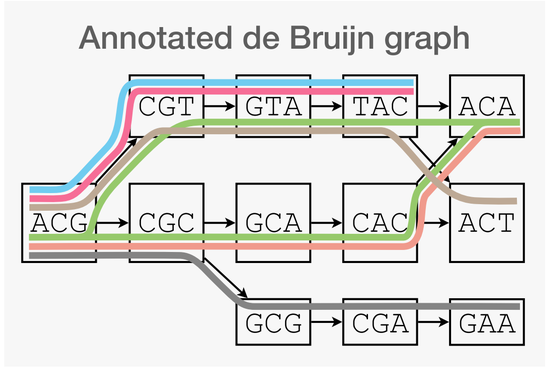
Indexing workflow

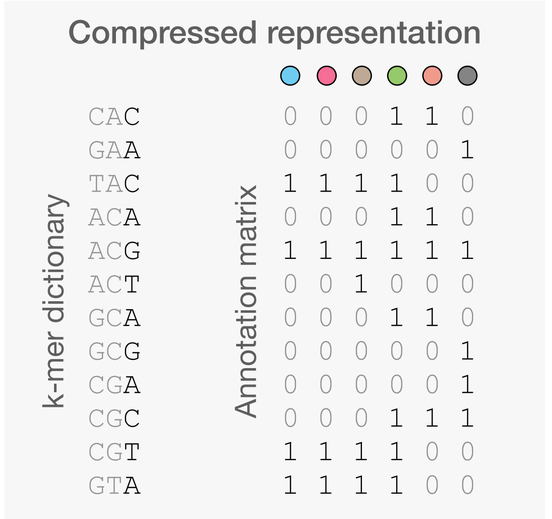


Optional graph cleaning (pop bubbles, prune tips)



Merge into a joint graph





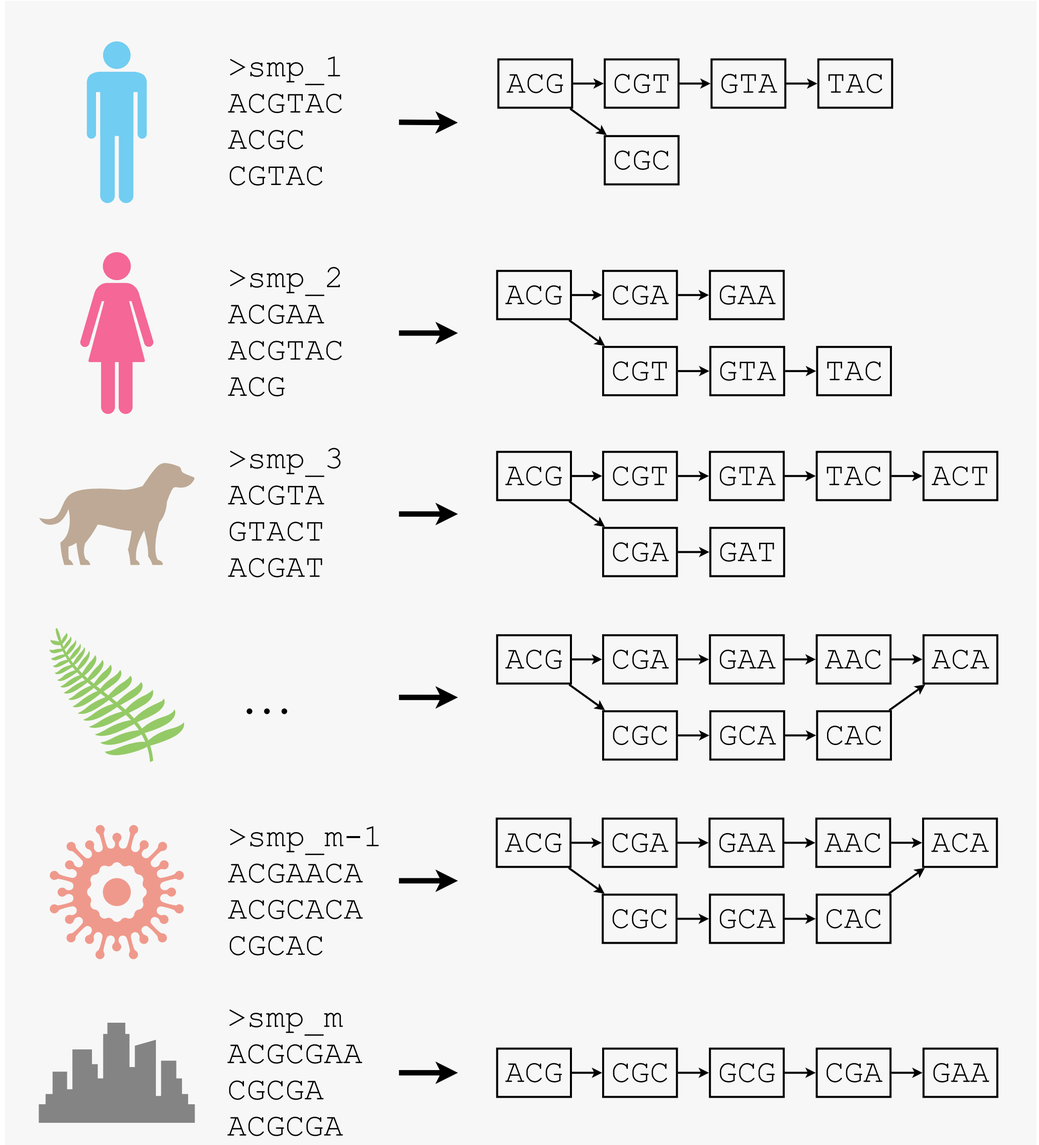
Indexing workflow

MetaGraph

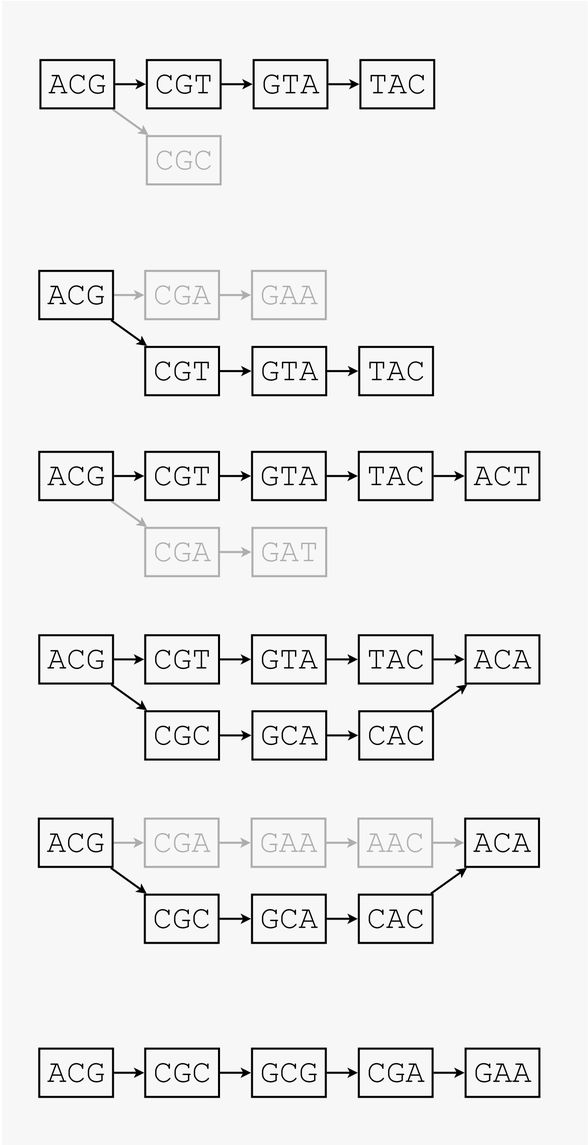
MetaGraph: Indexing and Analysing Nucleotide Archives at Petabase-scale

Mikhail Karasikov, Harun Mustafa, Daniel Danciu, Marc Zimmermann, Christopher Barber, Gunnar Rätsch, André Kahles

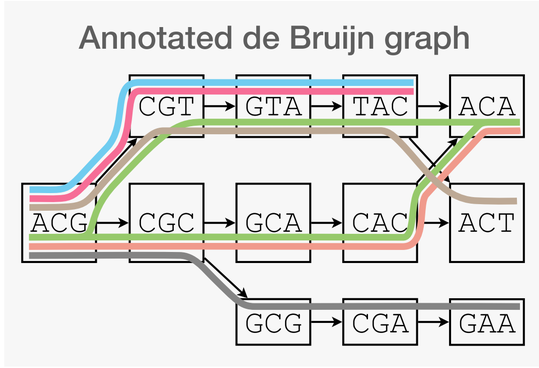
doi: https://doi.org/10.1101/2020.10.01.322164

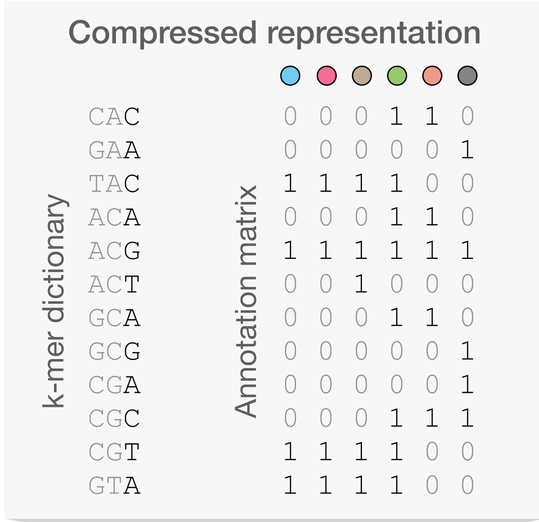


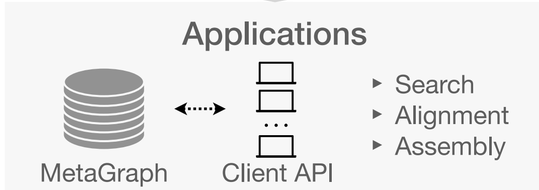
Optional graph cleaning (pop bubbles, prune tips)



Merge into a joint graph







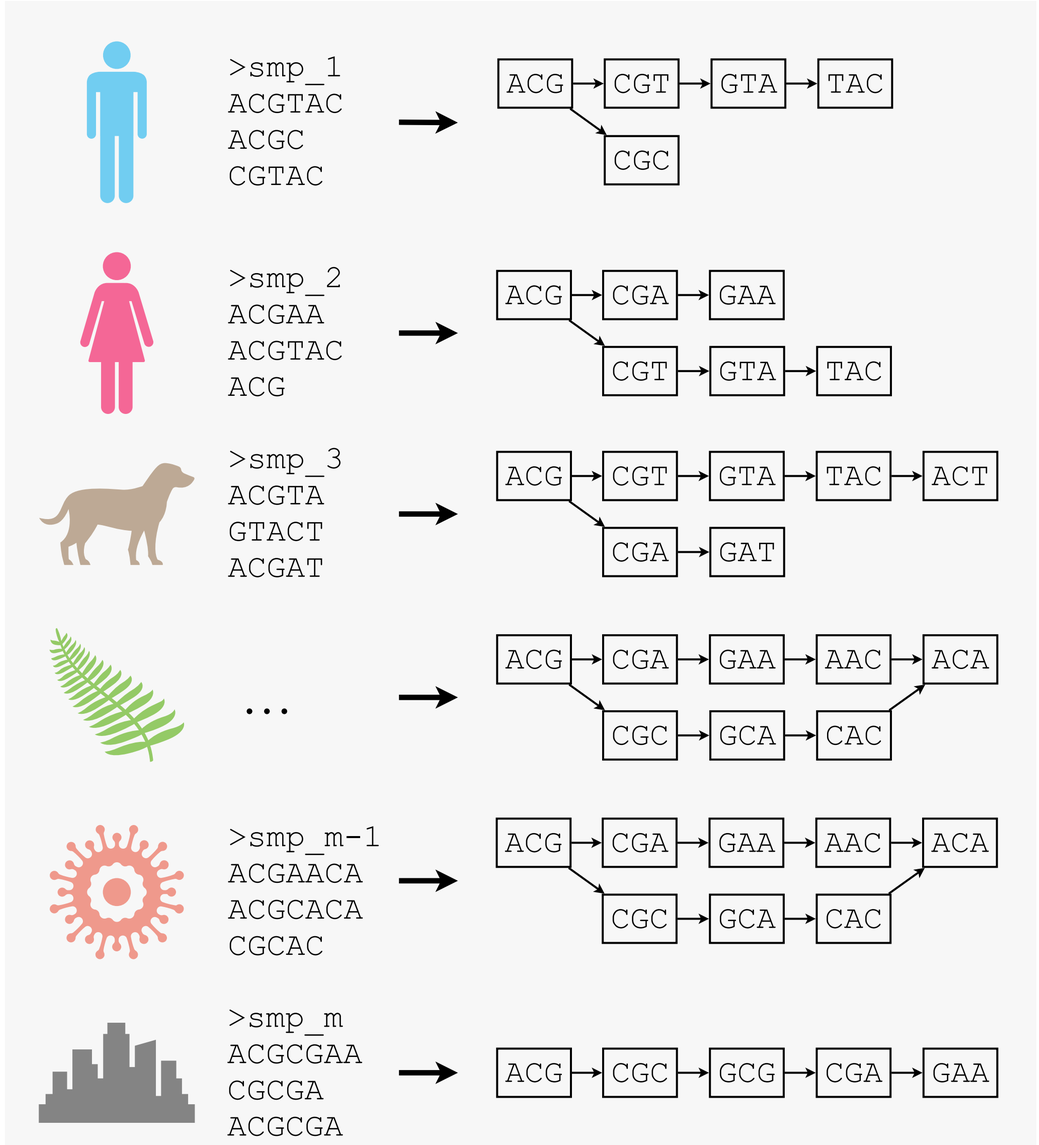
Indexing workflow

MetaGraph

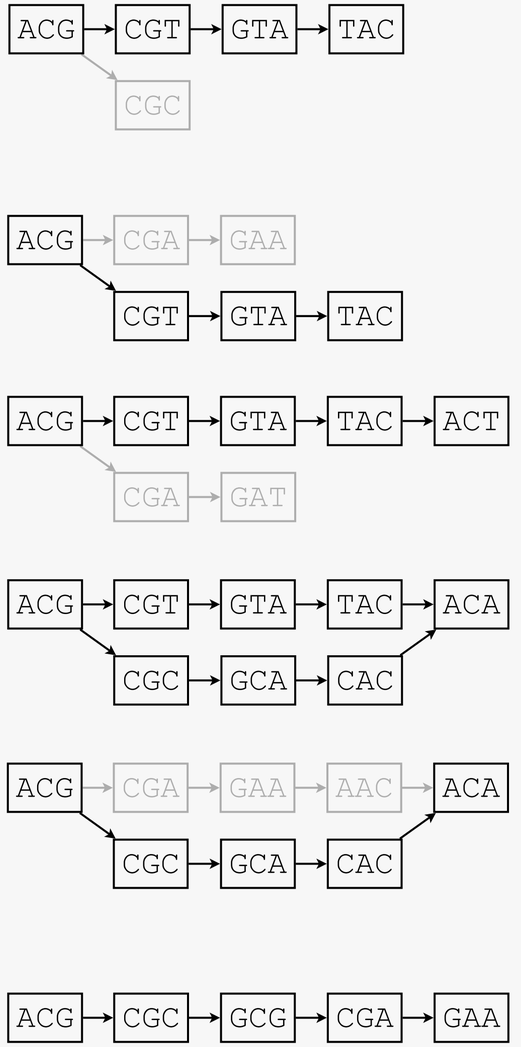
MetaGraph: Indexing and Analysing Nucleotide Archives at Petabase-scale

Mikhail Karasikov, Harun Mustafa, Daniel Danciu, Marc Zimmermann, Christopher Barber, Gunnar Rätsch, André Kahles

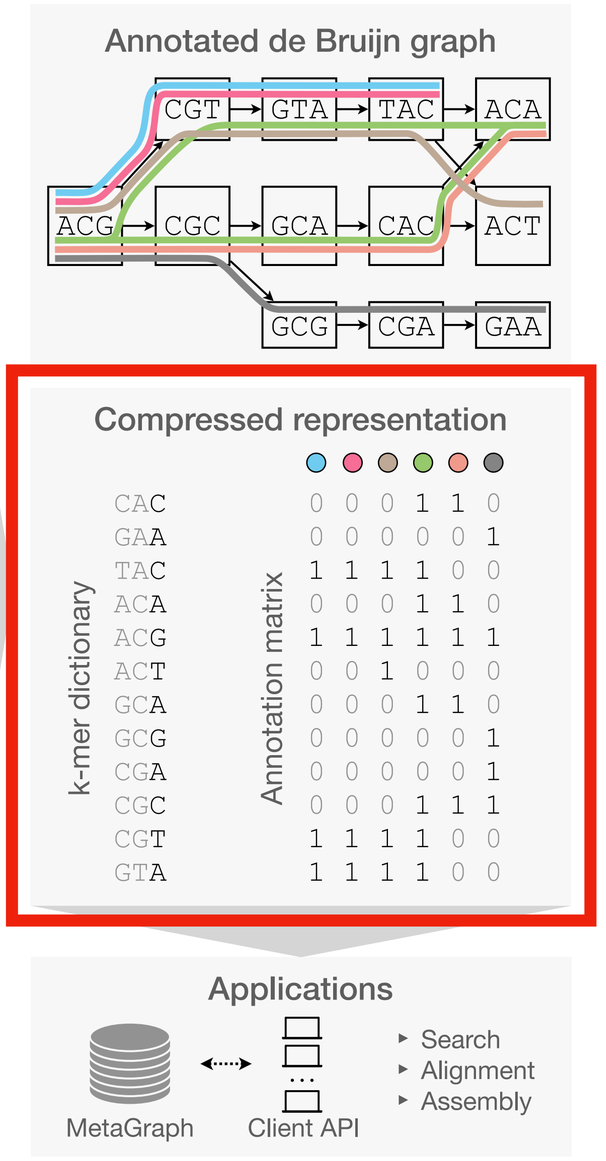
doi: https://doi.org/10.1101/2020.10.01.322164



Optional graph cleaning (pop bubbles, prune tips)



Merge into a joint graph



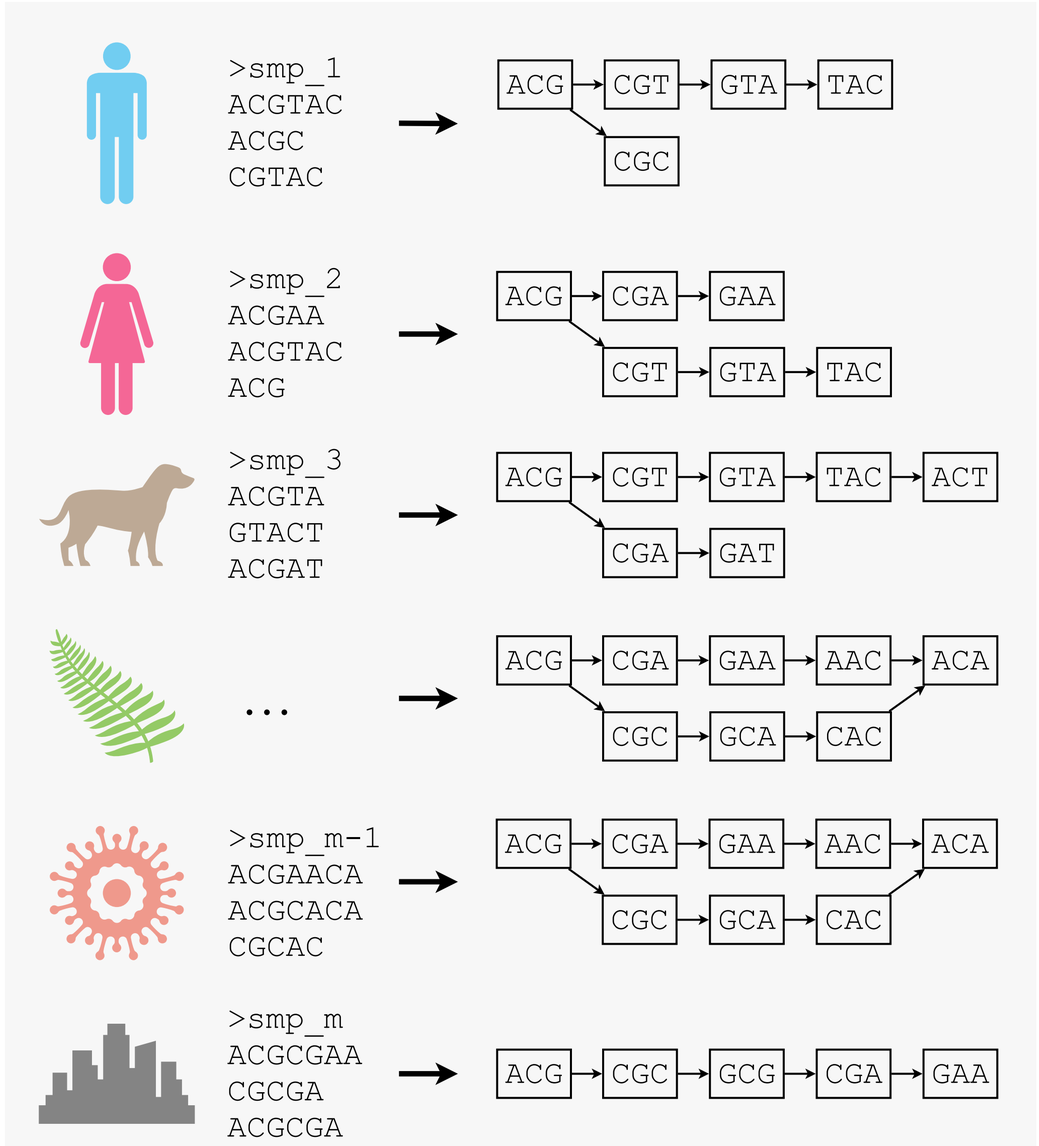
Indexing workflow

MetaGraph

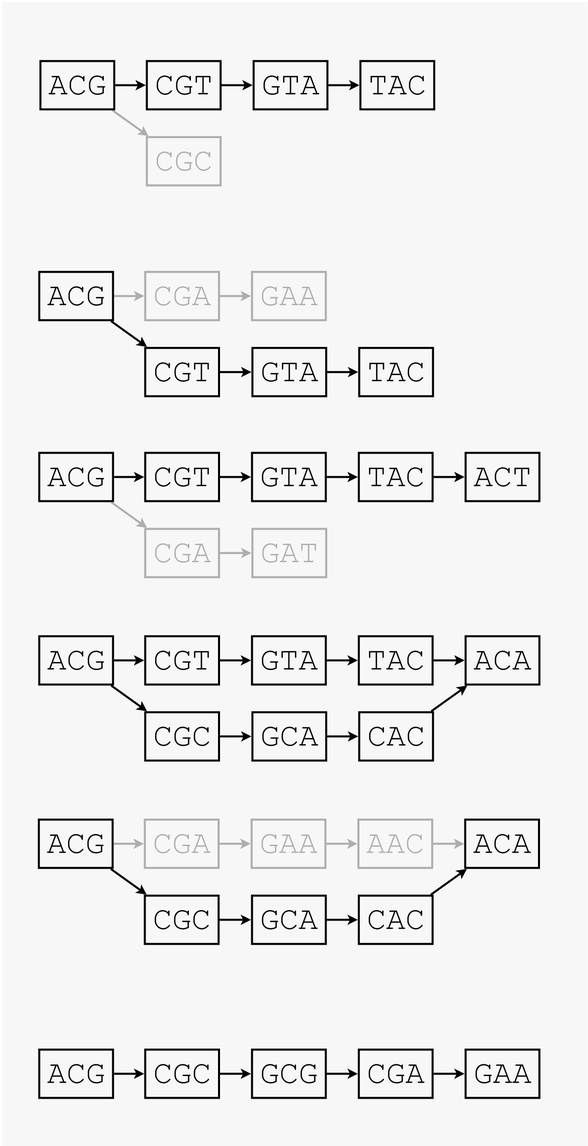
MetaGraph: Indexing and Analysing Nucleotide Archives at Petabase-scale

Mikhail Karasikov, Harun Mustafa, Daniel Danciu, Marc Zimmermann, Christopher Barber, Gunnar Rätsch, André Kahles

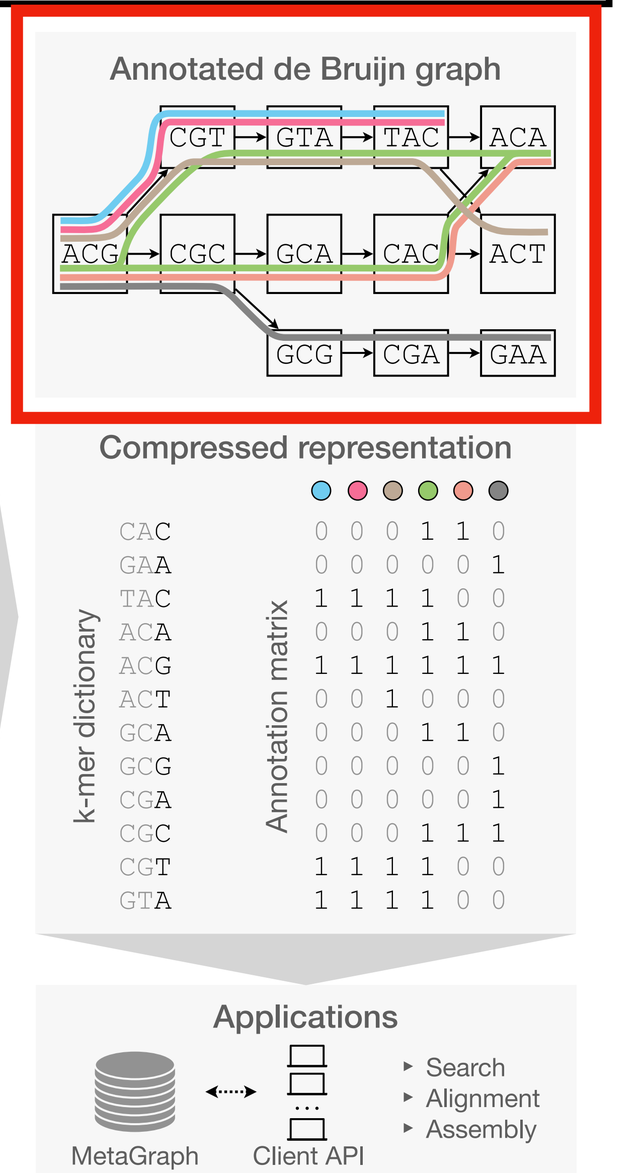
doi: https://doi.org/10.1101/2020.10.01.322164



Optional graph cleaning (pop bubbles, prune tips)



Merge into a joint graph



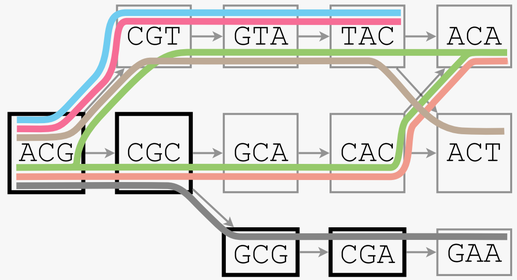
Sequence search

MetaGraph: Indexing and Analysing Nucleotide Archives at Petabase-scale

Mikhail Karasikov, Harun Mustafa, Daniel Danciu, Marc Zimmermann, Christopher Barber, Gunnar Rätsch, André Kahles

doi: https://doi.org/10.1101/2020.10.01.322164

```
>query
ACGCGAT
```



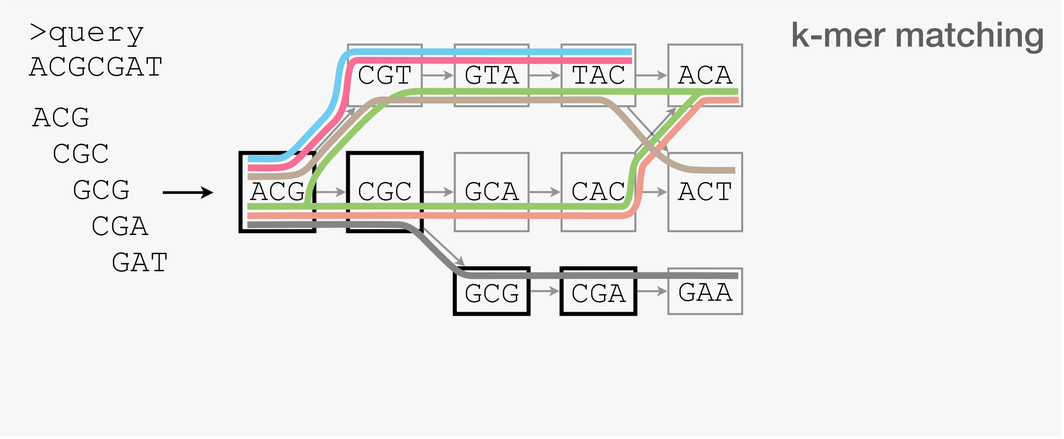
k-mer matching

Sequence search

MetaGraph: Indexing and Analysing Nucleotide Archives at Petabase-scale

Mikhail Karasikov, Harun Mustafa, Daniel Danciu, Marc Zimmermann, Christopher Barber, Gunnar Rätsch, André Kahles

doi: https://doi.org/10.1101/2020.10.01.322164

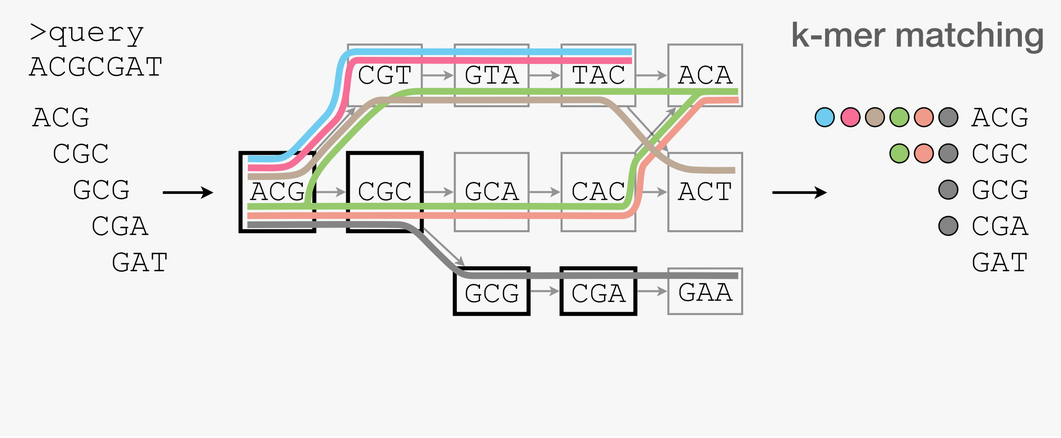


Sequence search

MetaGraph: Indexing and Analysing Nucleotide Archives at Petabase-scale

Mikhail Karasikov, Harun Mustafa, Daniel Danciu, Marc Zimmermann, Christopher Barber, Gunnar Rätsch, André Kahles

doi: https://doi.org/10.1101/2020.10.01.322164

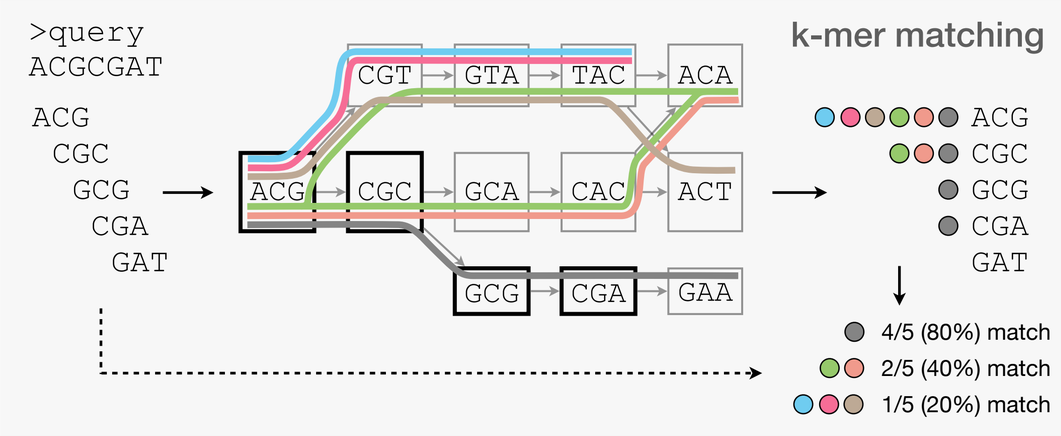


Sequence search

MetaGraph: Indexing and Analysing Nucleotide Archives at Petabase-scale

Mikhail Karasikov, Harun Mustafa, Daniel Danciu, Marc Zimmermann, Christopher Barber, Gunnar Rätsch, André Kahles

doi: https://doi.org/10.1101/2020.10.01.322164

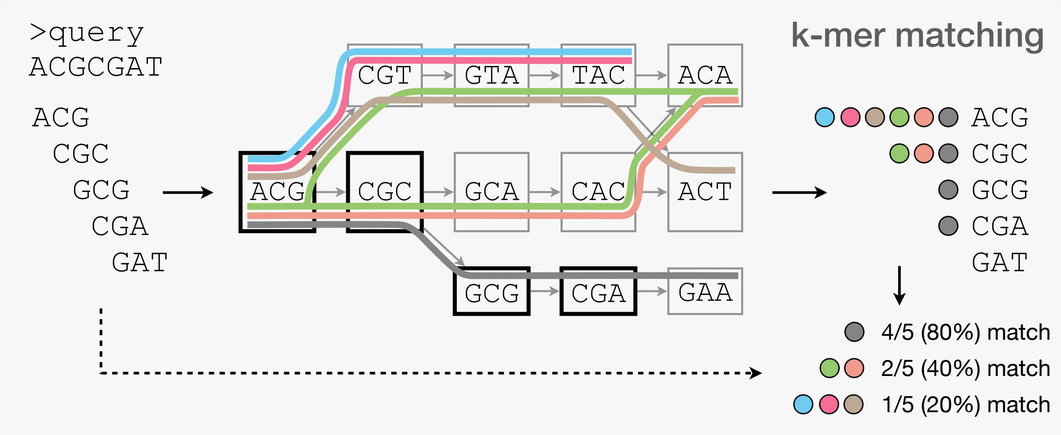


Sequence search

MetaGraph: Indexing and Analysing Nucleotide Archives at Petabase-scale

Mikhail Karasikov, Harun Mustafa, Daniel Danciu, Marc Zimmermann, Christopher Barber, Gunnar Rätsch, André Kahles

doi: https://doi.org/10.1101/2020.10.01.322164



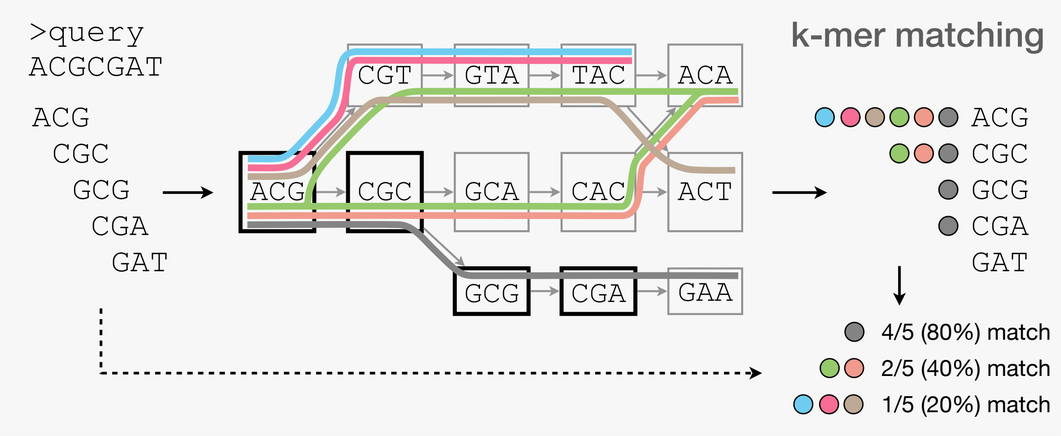
- SBT [Solomon, Kingsford, 2016]
- VARI [Muggli *et al.*, 2017]
- Mantis [Pandey *et al.*, 2018]

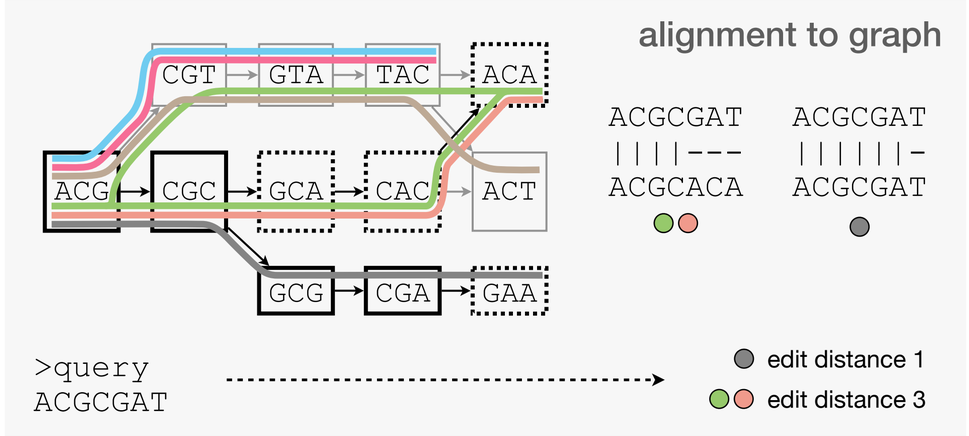
Sequence search

MetaGraph: Indexing and Analysing Nucleotide Archives at Petabase-scale

Mikhail Karasikov, Harun Mustafa, Daniel Danciu, Marc Zimmermann, Christopher Barber, Gunnar Rätsch, André Kahles

doi: https://doi.org/10.1101/2020.10.01.322164

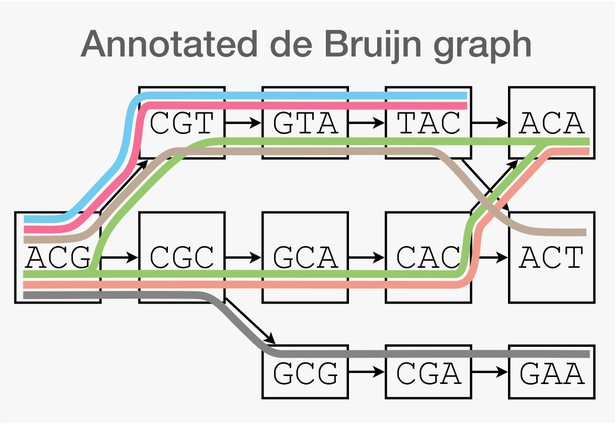


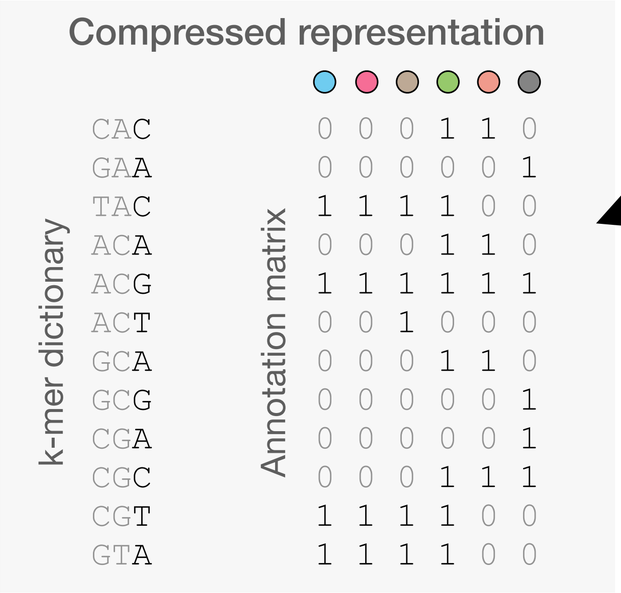


- SBT [Solomon, Kingsford, 2016]
- VARI [Muggli *et al.*, 2017]
- Mantis [Pandey *et al.*, 2018]

- [Lee, Grasso, Sharlow, 2002]
- deBGA [Liu, Guo, Brudno, Wang, 2016]
- SPAligner [Dvorkina *et al.*, 2020]
- AStarix [Ivanov *et al.*, 2020]
- [Schulz *et al.*, 2021]
- MetaGraph-MLA [Mustafa *et al.*, 2022]

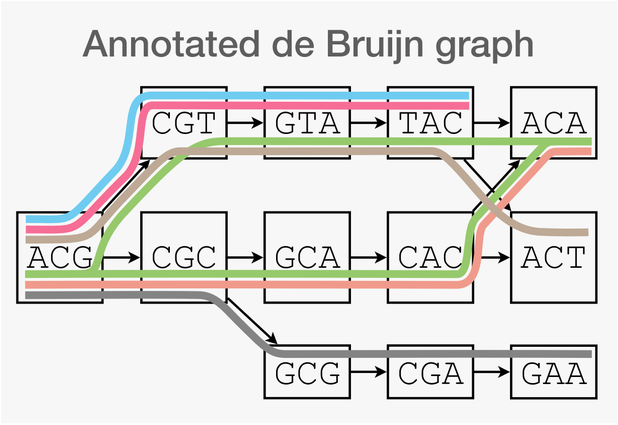
Representing annotated De Bruijn graphs

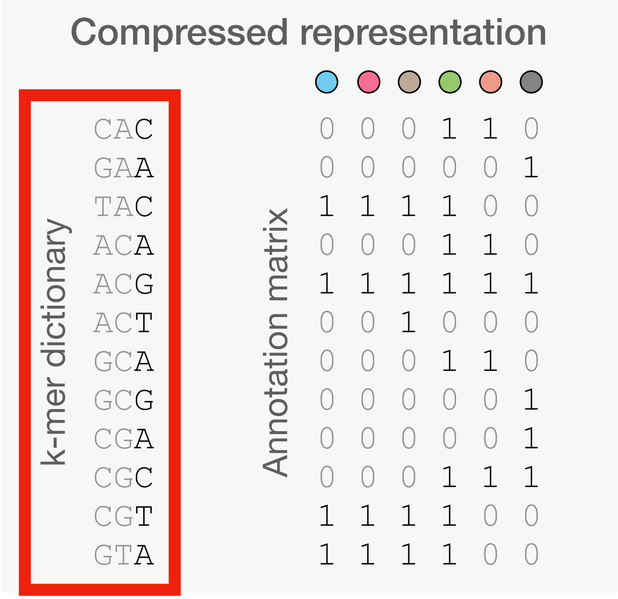




Must be efficiently **queryable**

Representing annotated De Bruijn graphs

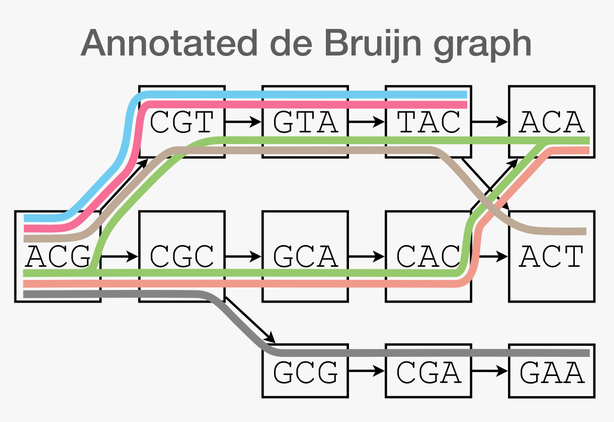


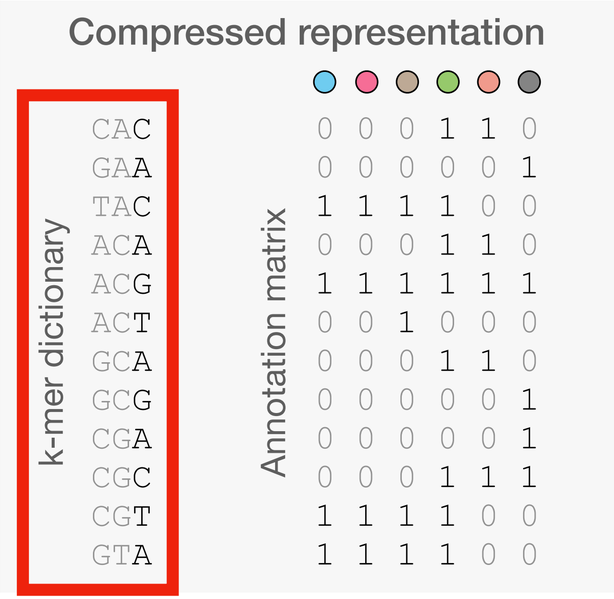


Representing graph

- Hash table
- Indicator set bit vector [Conway *et al.*, 2011]
- BOSS table [Bowe *et al.*, 2012]

Representing annotated De Bruijn graphs

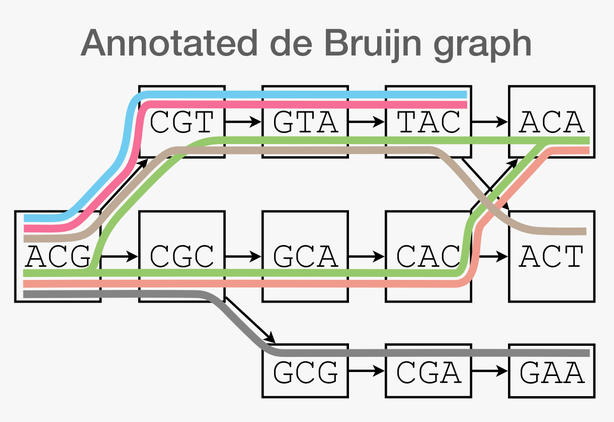


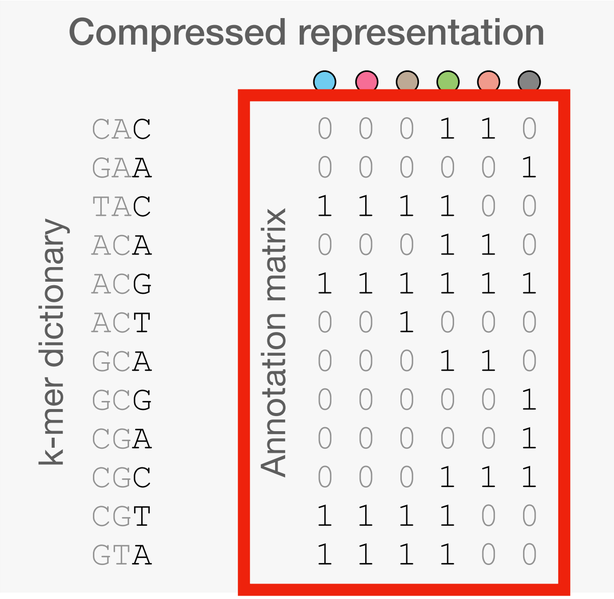


Representing graph

- Hash table
- Indicator set bit vector [Conway *et al.*, 2011]
- BOSS table [Bowe *et al.*, 2012]
  - ✓ ~2–3 bits per k-mer for DNA
  - ✓ supports search of sub-k-mers

Representing annotated De Bruijn graphs





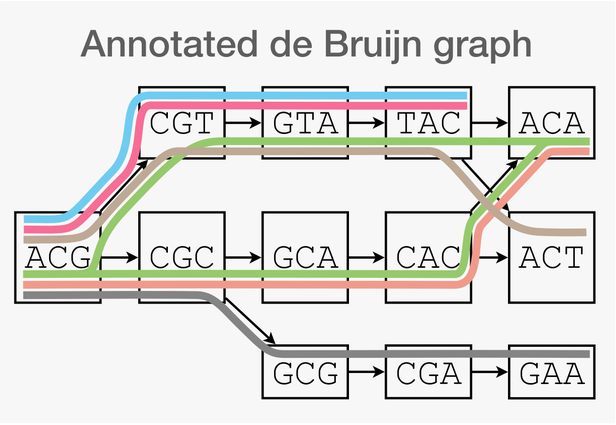
Representing graph

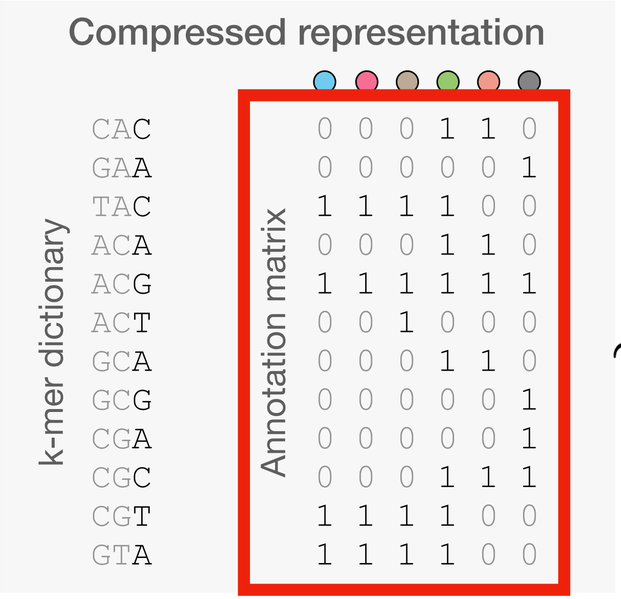
- Hash table
- Indicator set bit vector [Conway *et al.*, 2011]
- BOSS table [Bowe *et al.*, 2012]
  - ✓ ~2–3 bits per k-mer for DNA
  - ✓ supports search of sub-k-mers

Representing annotation

- Column-major sparse representation
- RowFlat (employed in VARI [Muggli *et al.*, 2017])
- Rainbowfish [Almodaresi *et al.*, 2017]
- Rainbow-MST [Almodaresi *et al.*, 2019]

Representing annotated De Bruijn graphs





$\sim 10^{11}$

$\sim 10^{6}$

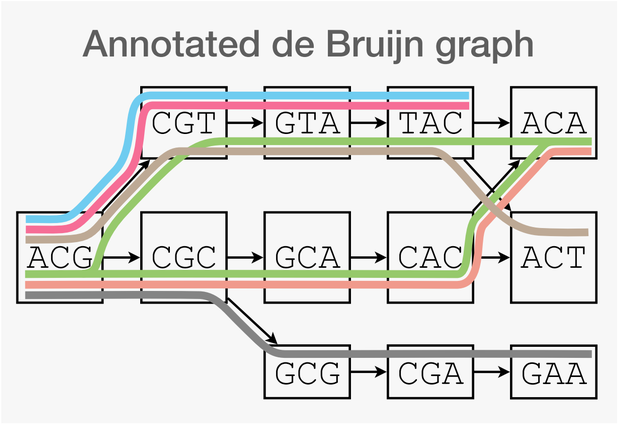
Representing graph

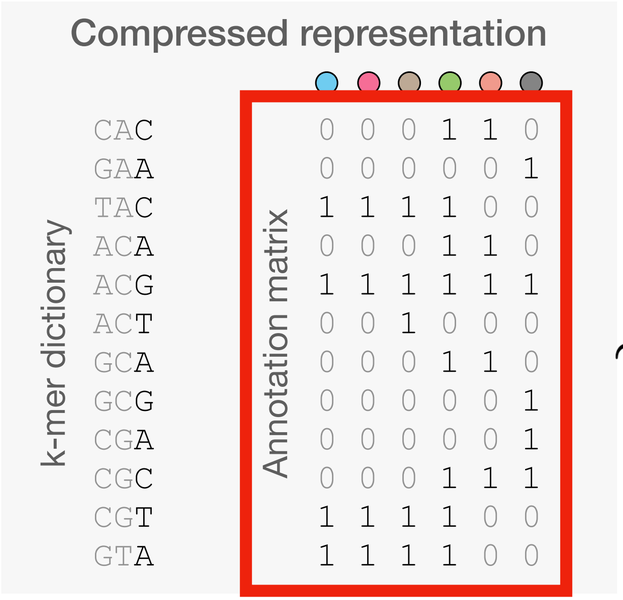
- Hash table
- Indicator set bit vector [Conway *et al.*, 2011]
- BOSS table [Bowe *et al.*, 2012]
  - ✓ ~2–3 bits per k-mer for DNA
  - ✓ supports search of sub-k-mers

Representing annotation

- Column-major sparse representation
- RowFlat (employed in VARI [Muggli *et al.*, 2017])
- Rainbowfish [Almodaresi *et al.*, 2017]
- Rainbow-MST [Almodaresi *et al.*, 2019]

Representing annotated De Bruijn graphs





$\sim 10^{11}$

$\sim 10^{6}$

Representing graph

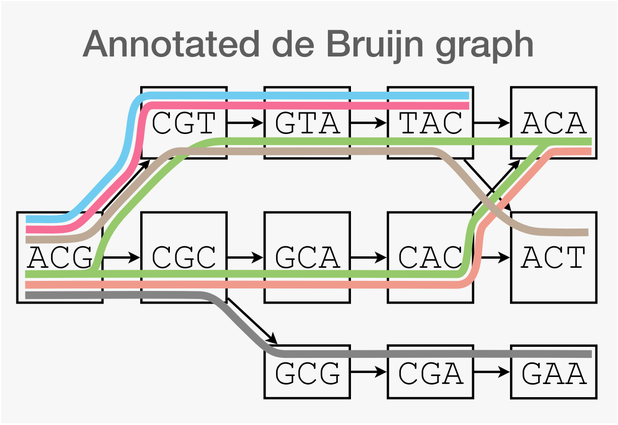
- Hash table
- Indicator set bit vector [Conway *et al.*, 2011]
- BOSS table [Bowe *et al.*, 2012]
  - ✓ ~2–3 bits per k-mer for DNA
  - ✓ supports search of sub-k-mers

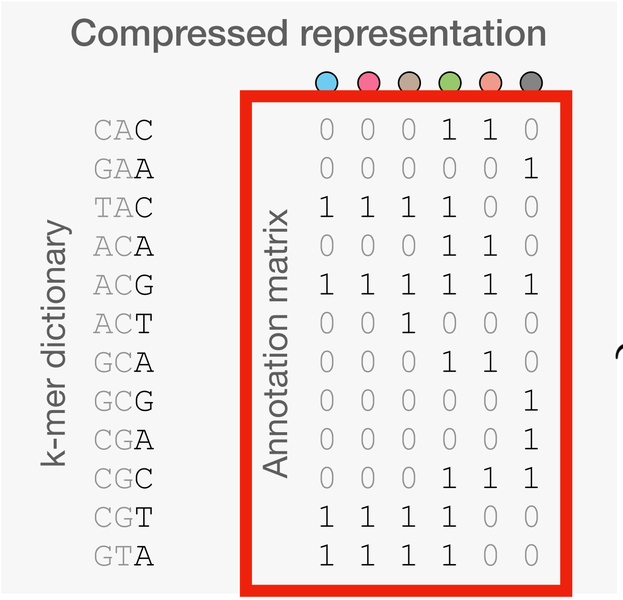
Representing annotation

- Column-major sparse representation
- RowFlat (employed in VARI [Muggli *et al.*, 2017])
- Rainbowfish [Almodaresi *et al.*, 2017]
- Rainbow-MST [Almodaresi *et al.*, 2019]

Challenge:
Represent huge-scale annotations

Representing graph annotations





$\sim 10^{11}$

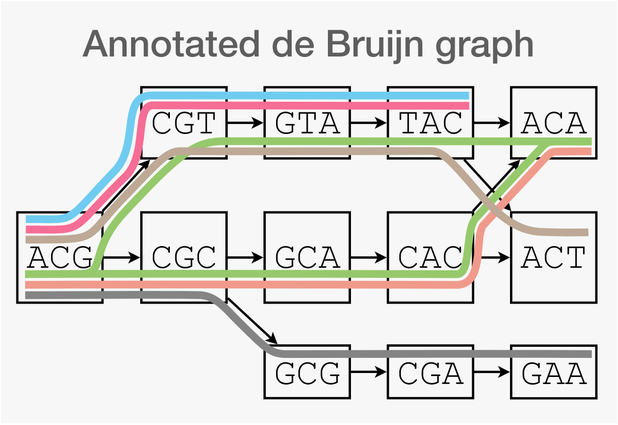
$\sim 10^{6}$

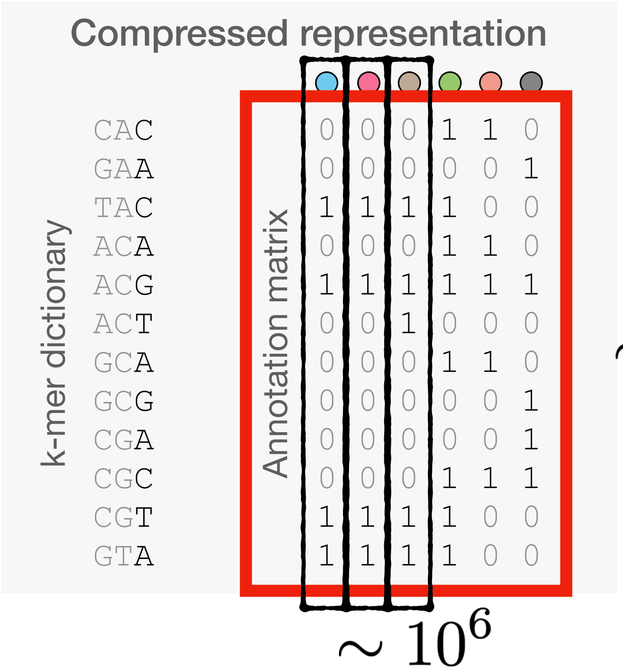
Challenge:
Represent huge-scale annotations

Typical properties

1. high sparsity

Representing graph annotations



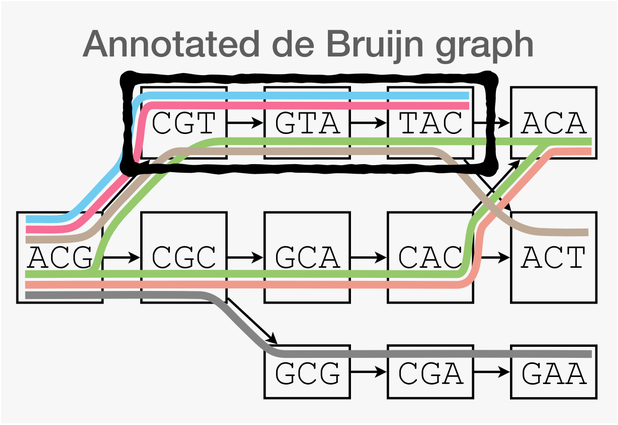


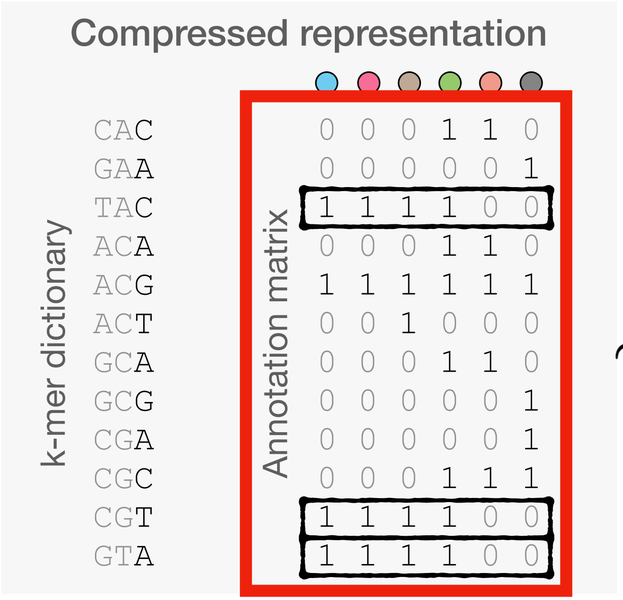
Challenge:
Represent huge-scale annotations

Typical properties

1. high sparsity
2. similarity of **columns**

Representing graph annotations





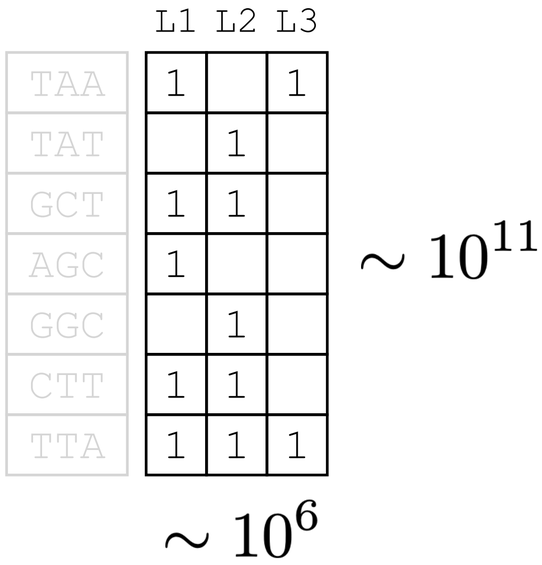
Challenge:
Represent huge-scale annotations

Typical properties

1. high sparsity
2. similarity of **columns**
3. similarity of **rows**

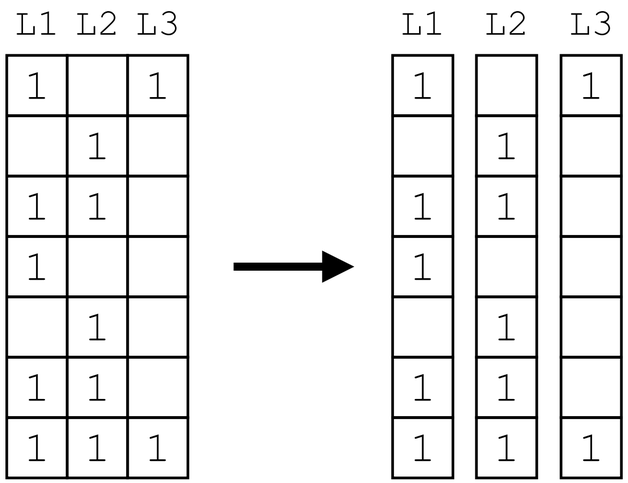
Annotation representations

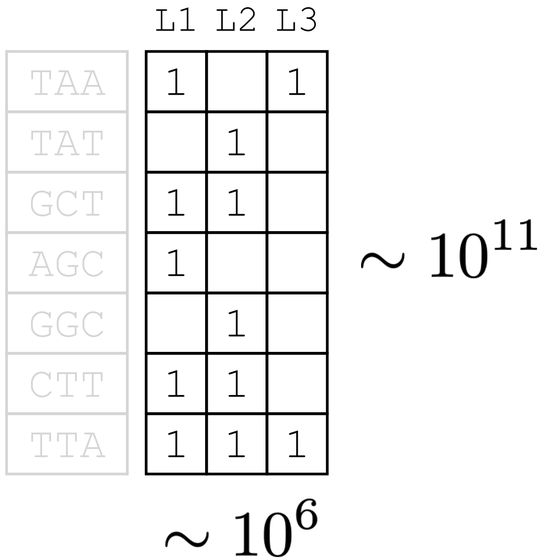
1. Column-major sparse representation



Annotation representations

1. Column-major sparse representation

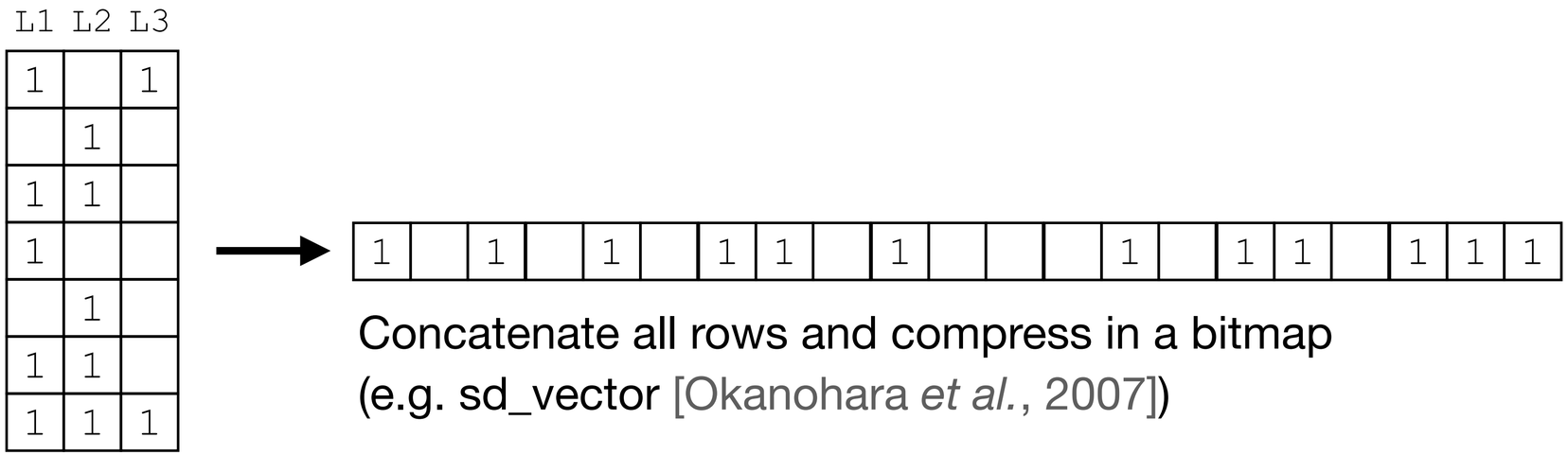




Columns are stored as **compressed bitmaps**
(e.g. sd_vector [Okanohara *et al.*, 2007])

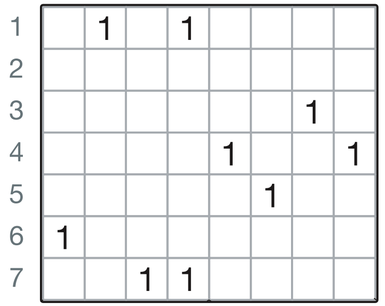
Annotation representations

1. Column-major sparse representation
2. RowFlat (employed in VARI [Muggli *et al.*, 2017])



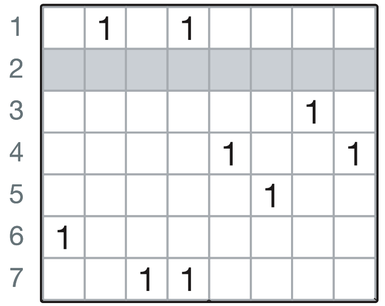
Annotation representations

1. Column-major sparse representation
2. RowFlat (employed in VARI [Muggli *et al.*, 2017])
3. BRWT [Barbay *et al.*, 2013], [Karasikov *et al.*, 2019]



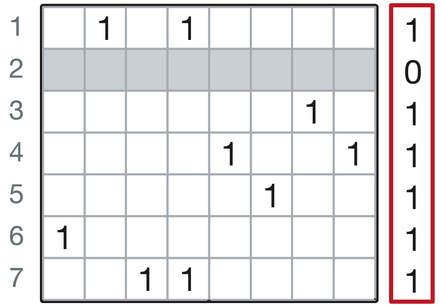
Annotation representations

1. Column-major sparse representation
2. RowFlat (employed in VARI [Muggli *et al.*, 2017])
3. BRWT [Barbay *et al.*, 2013], [Karasikov *et al.*, 2019]



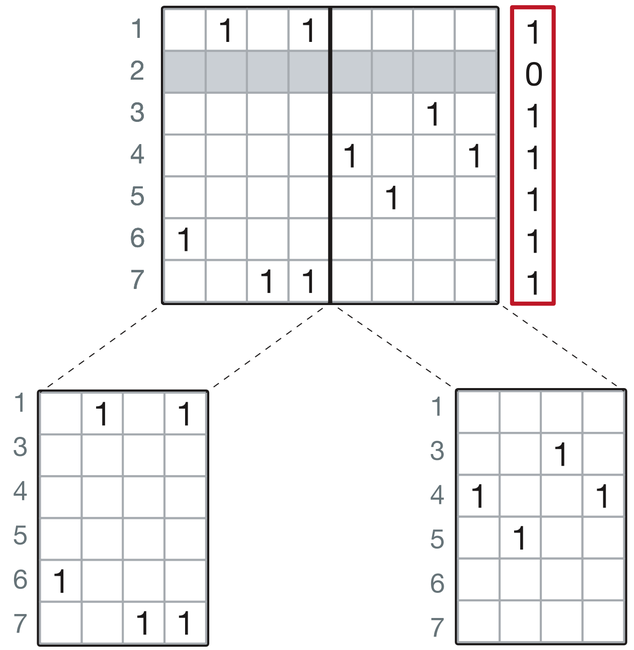
Annotation representations

1. Column-major sparse representation
2. RowFlat (employed in VARI [Muggli *et al.*, 2017])
3. BRWT [Barbay *et al.*, 2013], [Karasikov *et al.*, 2019]



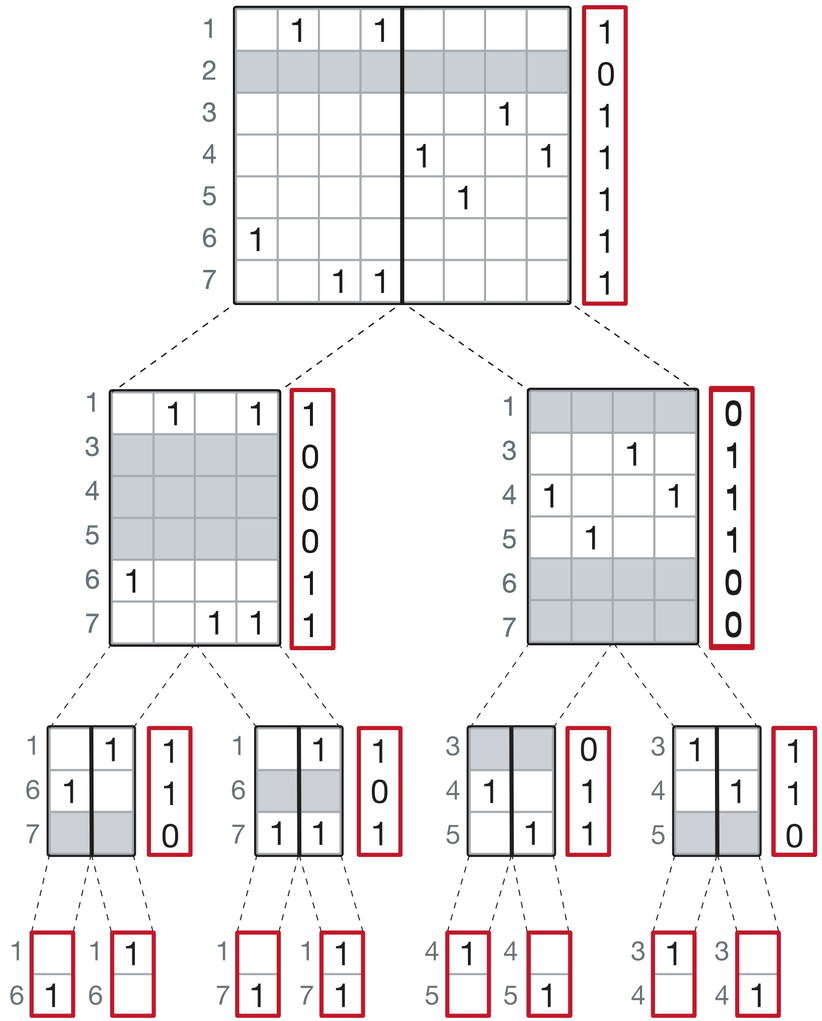
Annotation representations

1. Column-major sparse representation
2. RowFlat (employed in VARI [Muggli *et al.*, 2017])
3. BRWT [Barbay *et al.*, 2013], [Karasikov *et al.*, 2019]



Annotation representations

1. Column-major sparse representation
2. RowFlat (employed in VARI [Muggli *et al.*, 2017])
3. BRWT [Barbay *et al.*, 2013], [Karasikov *et al.*, 2019]



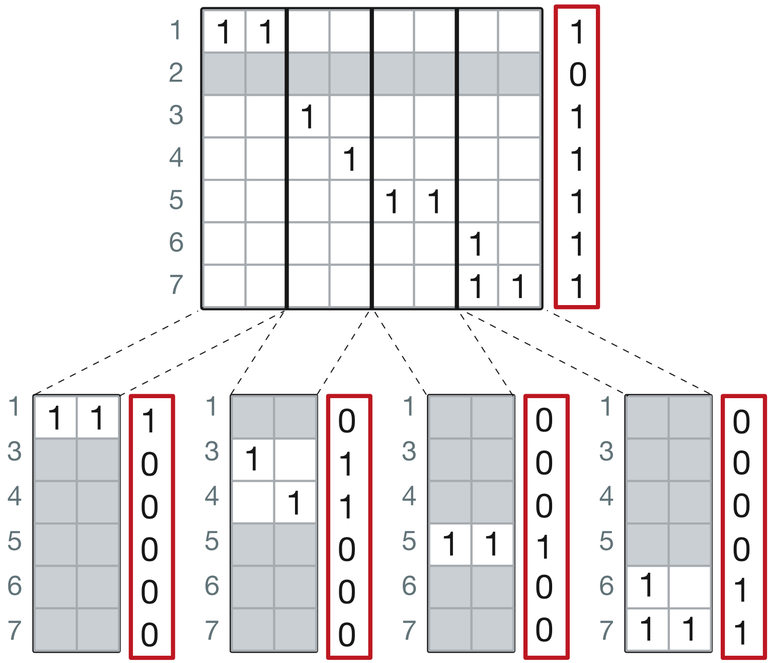
Annotation representations

Sparse Binary Relation Representations for Genome Graph Annotation

Mikhail Karasikov, Harun Mustafa, Amir Joudaki, Sara Javadzadeh-No, Gunnar Rätsch & André Kahles

Conference paper | First Online: 02 April 2019

1. Column-major sparse representation
2. RowFlat (employed in VARI [Muggli *et al.*, 2017])
3. BRWT [Barbay *et al.*, 2013], [Karasikov *et al.*, 2019]
4. Multi-BRWT [Karasikov *et al.*, 2019]
   - Optimize column arrangement
   - Use multi-ary trees



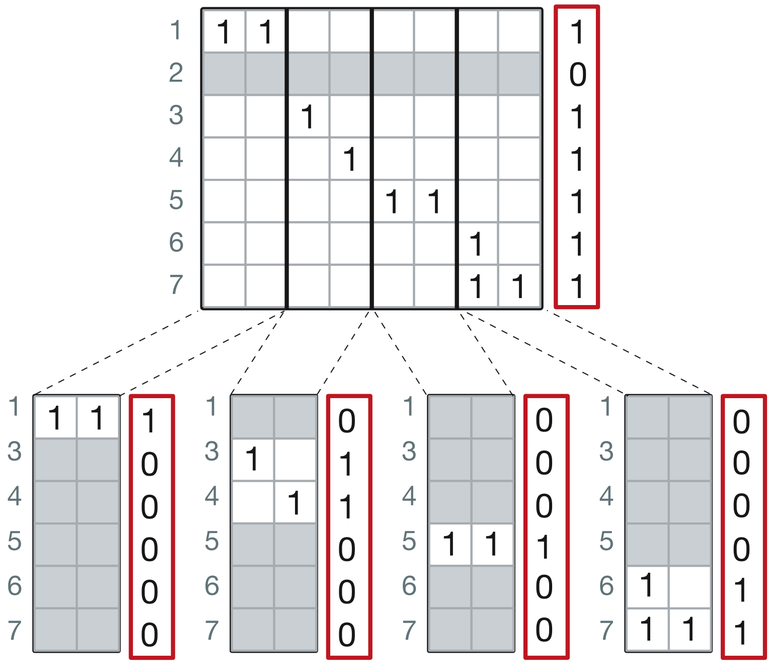
Annotation representations

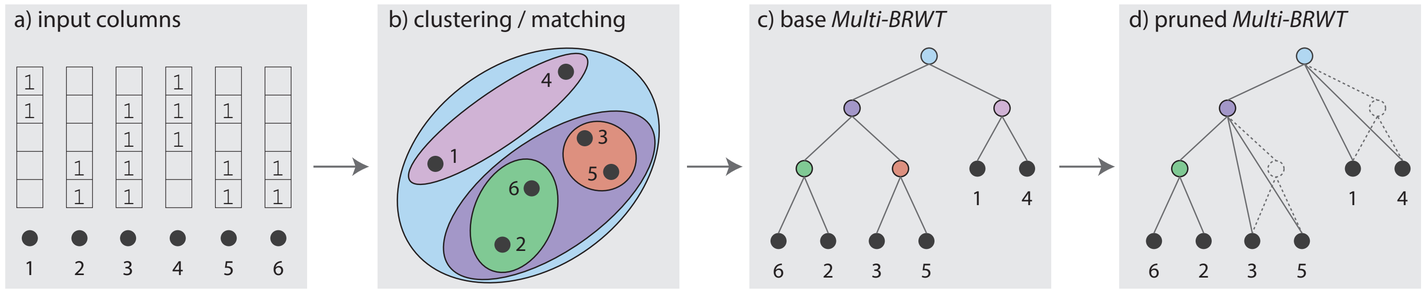
Sparse Binary Relation Representations for Genome Graph Annotation

Mikhail Karasikov, Harun Mustafa, Amir Joudaki, Sara Javadzadeh-No, Gunnar Rätsch & André Kahles

Conference paper | First Online: 02 April 2019

1. Column-major sparse representation
2. RowFlat (employed in VARI [Muggli *et al.*, 2017])
3. BRWT [Barbay *et al.*, 2013], [Karasikov *et al.*, 2019]
4. Multi-BRWT [Karasikov *et al.*, 2019]
   - Optimize column arrangement
   - Use multi-ary trees





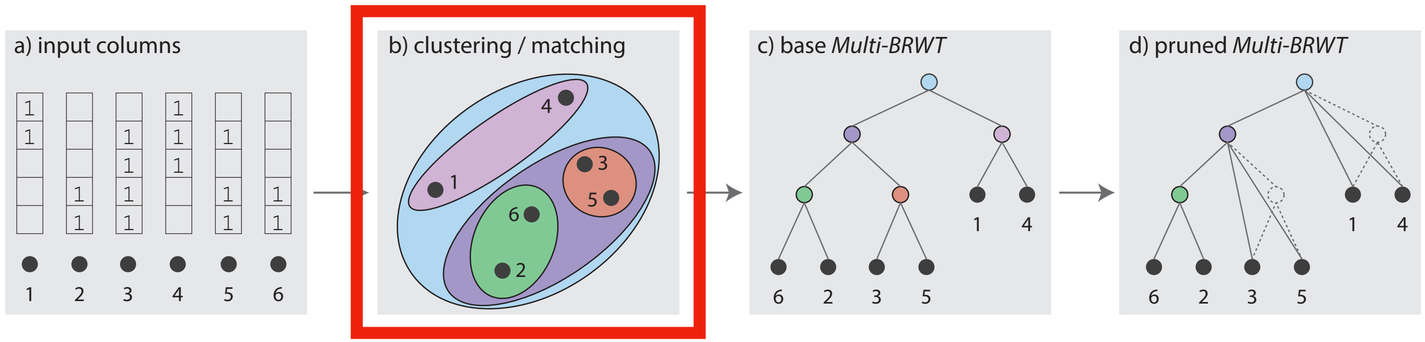
Annotation representations

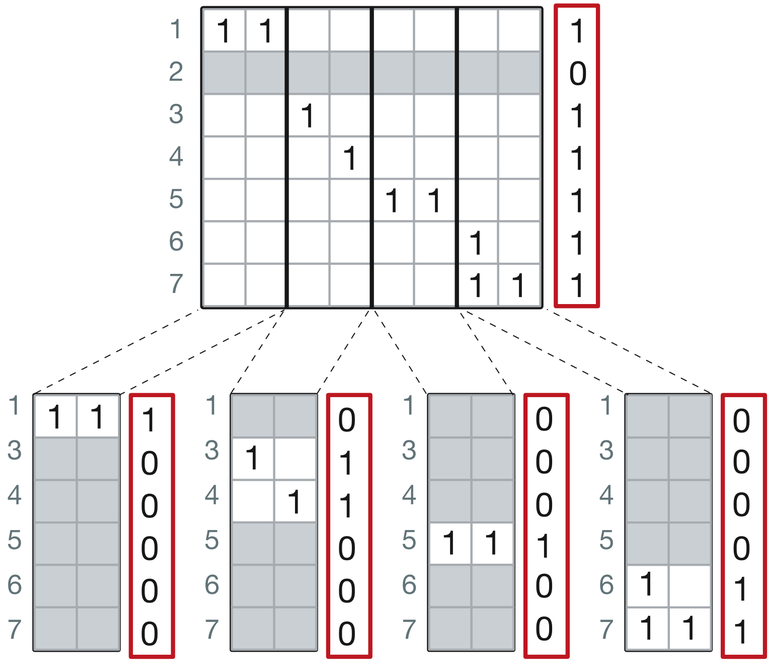
Sparse Binary Relation Representations for Genome Graph Annotation

Mikhail Karasikov, Harun Mustafa, Amir Joudaki, Sara Javadzadeh-No, Gunnar Rätsch & André Kahles

Conference paper | First Online: 02 April 2019

1. Column-major sparse representation
2. RowFlat (employed in VARI [Muggli *et al.*, 2017])
3. BRWT [Barbay *et al.*, 2013], [Karasikov *et al.*, 2019]
4. Multi-BRWT [Karasikov *et al.*, 2019]
   - Optimize column arrangement
   - Use multi-ary trees





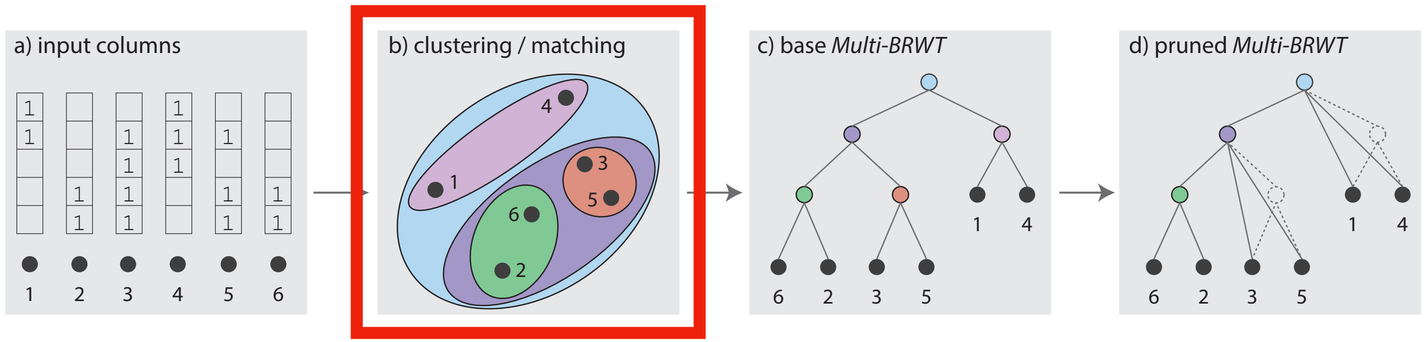
Annotation representations

Sparse Binary Relation Representations for Genome Graph Annotation

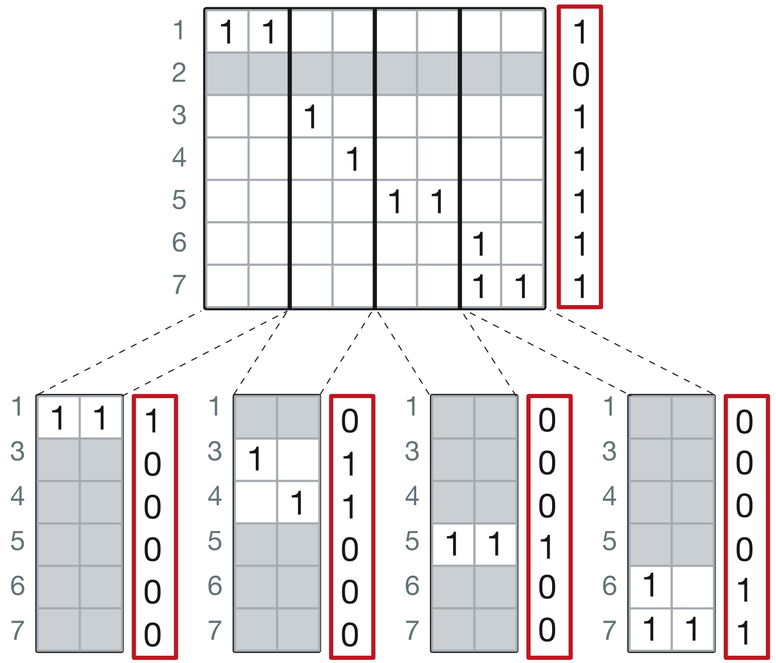
Mikhail Karasikov, Harun Mustafa, Amir Joudaki, Sara Javadzadeh-No, Gunnar Rätsch & André Kahles

Conference paper | First Online: 02 April 2019

1. Column-major sparse representation
2. RowFlat (employed in VARI [Muggli *et al.*, 2017])
3. BRWT [Barbay *et al.*, 2013], [Karasikov *et al.*, 2019]
4. Multi-BRWT [Karasikov *et al.*, 2019]
   - Optimize column arrangement
   - Use multi-ary trees



Non-trivial for >500,000 columns



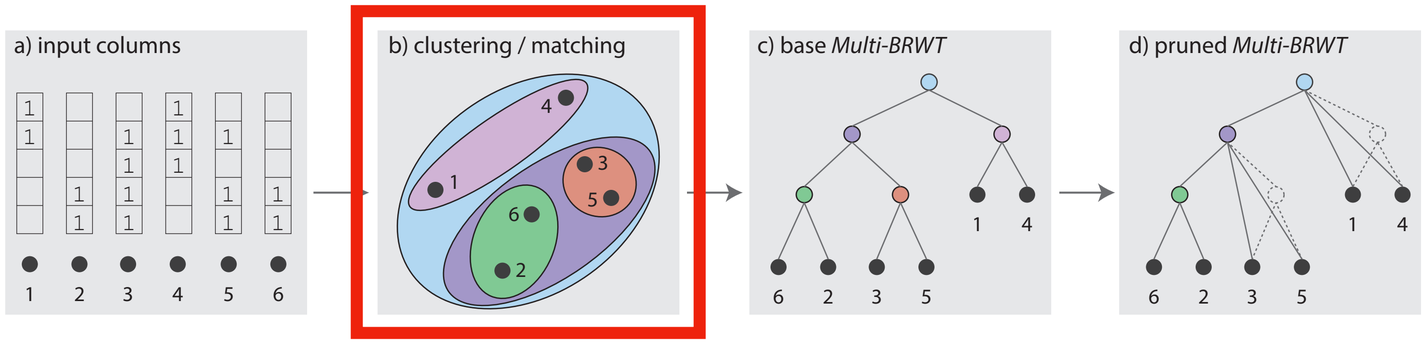
Annotation representations

Sparse Binary Relation Representations for Genome Graph Annotation

Mikhail Karasikov, Harun Mustafa, Amir Joudaki, Sara Javadzadeh-No, Gunnar Rätsch & André Kahles

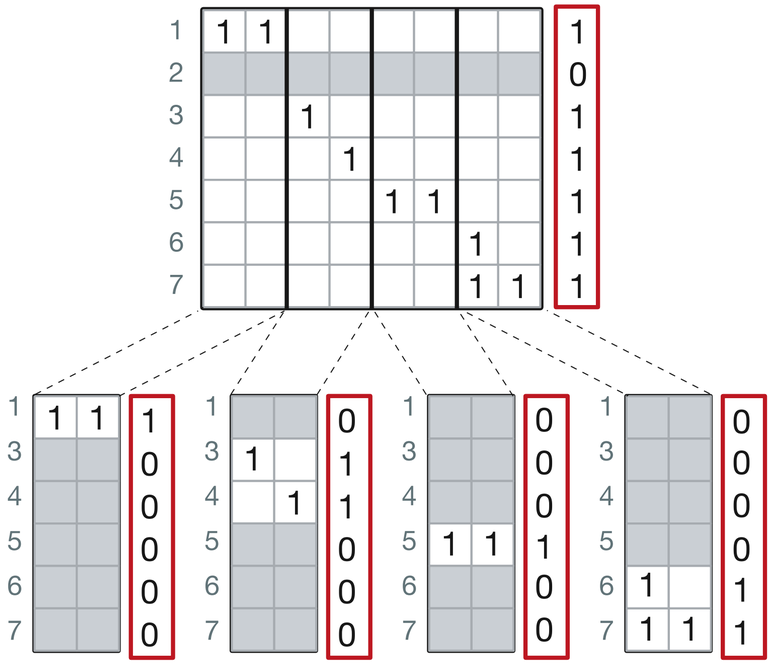
Conference paper | First Online: 02 April 2019

1. Column-major sparse representation
2. RowFlat (employed in VARI [Muggli *et al.*, 2017])
3. BRWT [Barbay *et al.*, 2013], [Karasikov *et al.*, 2019]
4. Multi-BRWT [Karasikov *et al.*, 2019]
   - Optimize column arrangement
   - Use multi-ary trees



Non-trivial for >500,000 columns

Use taxonomy



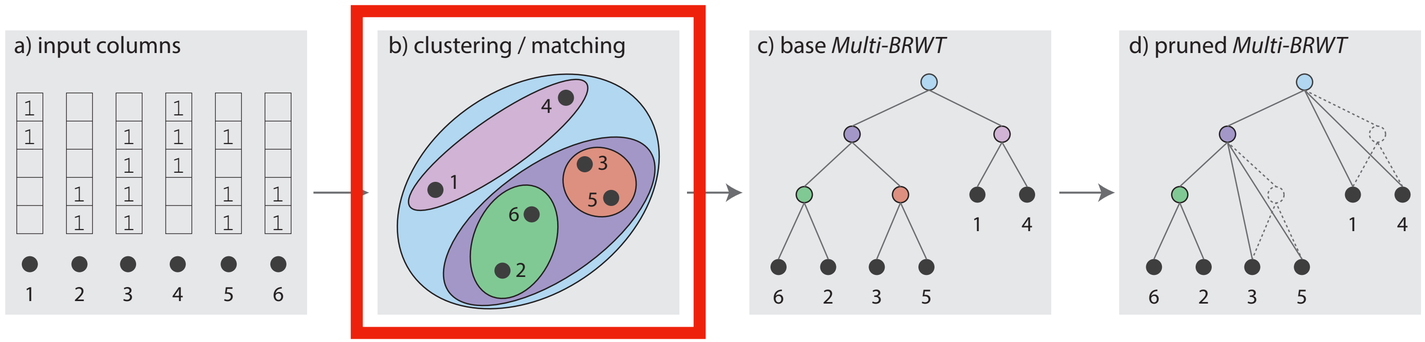
Annotation representations

Sparse Binary Relation Representations for Genome Graph Annotation

Mikhail Karasikov, Harun Mustafa, Amir Joudaki, Sara Javadzadeh-No, Gunnar Rätsch & André Kahles

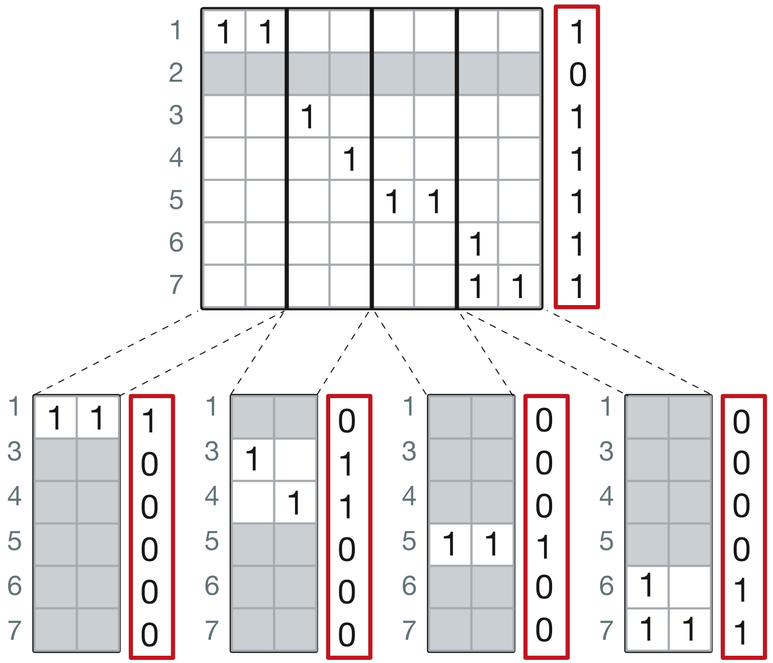
Conference paper | First Online: 02 April 2019

1. Column-major sparse representation
2. RowFlat (employed in VARI [Muggli *et al.*, 2017])
3. BRWT [Barbay *et al.*, 2013], [Karasikov *et al.*, 2019]
4. Multi-BRWT [Karasikov *et al.*, 2019]
   - Optimize column arrangement
   - Use multi-ary trees



Non-trivial for >500,000 columns

Use taxonomy

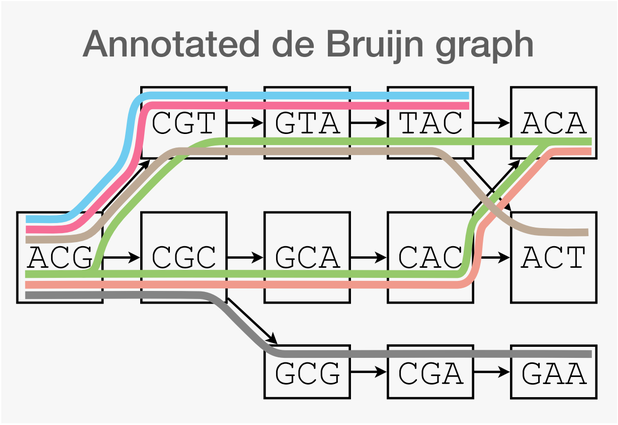


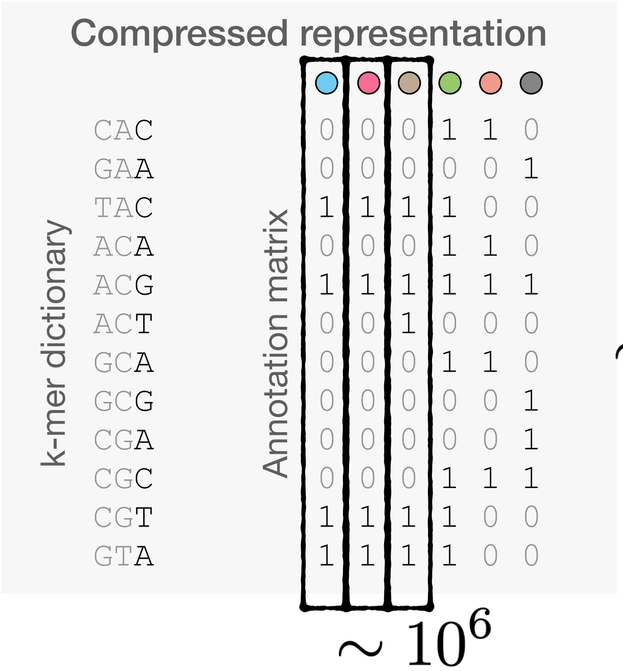
Efficient and Robust Search of Microbial Genomes via Phylogenetic Compression

Karel Břinda, Leandro Lima, Simone Pignotti, Natalia Quinones-Olvera, Kamil Salikhov, Rayan Chikhi, Gregory Kucherov, Zamin Iqbal, Michael Baym

doi: https://doi.org/10.1101/2023.04.15.536996 Posted April 18, 2023.

Representing graph annotations



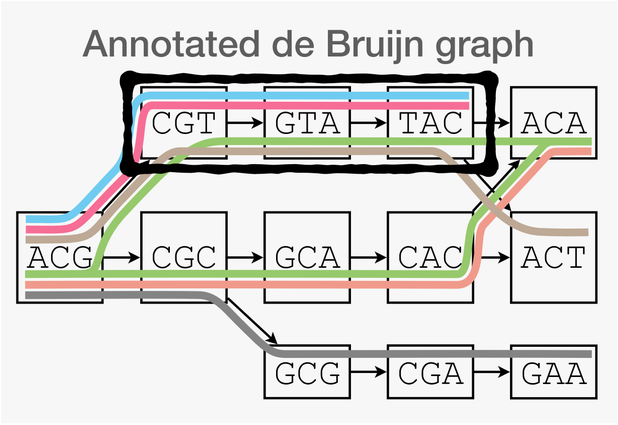


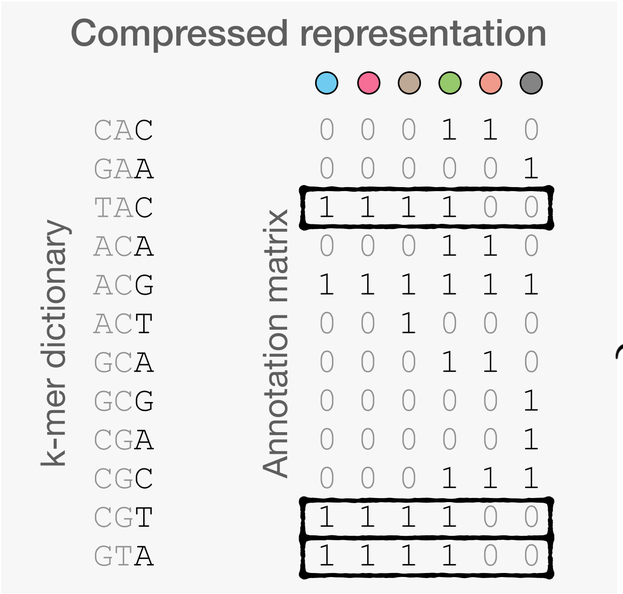
Challenge:
Represent huge-scale annotations

Typical properties

1. high sparsity
2. similarity of **columns** $\Rightarrow$ **Multi-BRWT**

Representing graph annotations





$\sim 10^{11}$

$\sim 10^{6}$

Challenge:
Represent huge-scale annotations

Typical properties

1. high sparsity
2. similarity of **columns** $\Rightarrow$ **Multi-BRWT**
3. similarity of **rows**
   - **adjacent nodes** have similar annotations

RowDiff

Delta compression

Bioinformatics, 37, 2021, i169–i176
doi: 10.1093/bioinformatics/btab330
ISMB/ECCB 2021
OXFORD

Topology-based sparsification of graph annotations

Daniel Danciu[1,2,†], Mikhail Karasikov[1,2,3,†], Harun Mustafa[1,2,3], André Kahles[1,2,3,*] and Gunnar Rätsch[1,2,3,4,*]

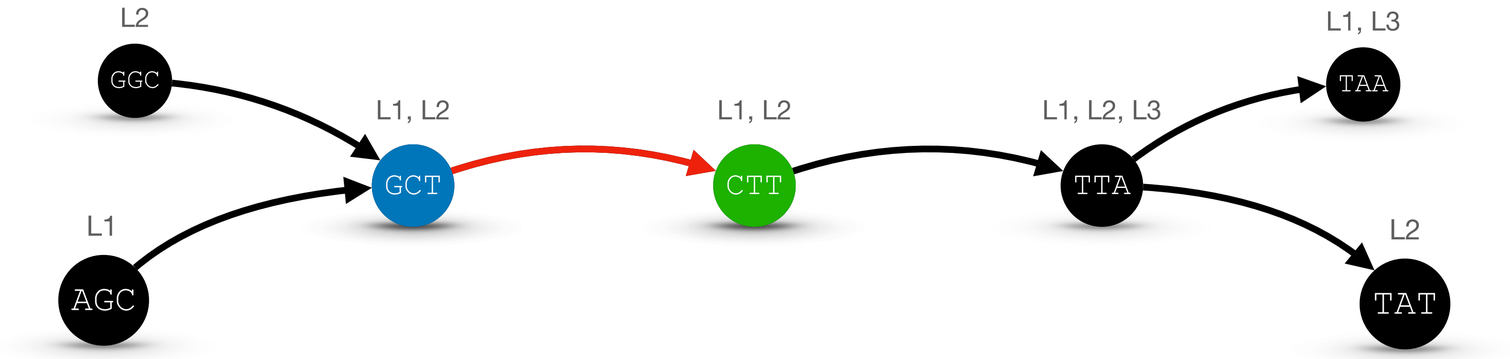
Idea:

- Store only **diffs**

$$L^{\delta}(v) := L(v) \oplus L(v_{\text{succ}})$$

XOR

	L1	L2	L3	
TAA	1		1	
TAT		1		
GCT	1	1		v
AGC	1			
GGC		1		
CTT	1	1		v_{succ}
TTA	1	1	1	



RowDiff

Delta compression

Bioinformatics, 37, 2021, i169–i176
doi: 10.1093/bioinformatics/btab330
ISMB/ECCB 2021
OXFORD

Topology-based sparsification of graph annotations

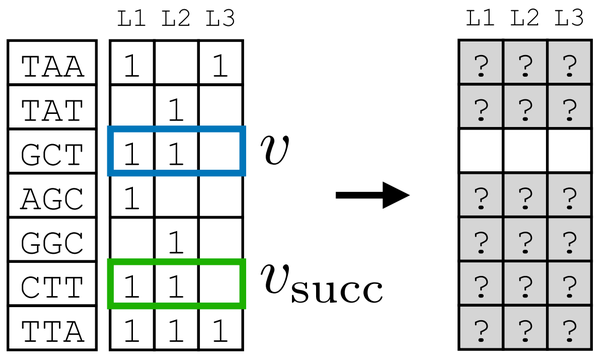
Daniel Danciu[1,2,†], Mikhail Karasikov[1,2,3,†], Harun Mustafa[1,2,3], André Kahles[1,2,3,*] and Gunnar Rätsch[1,2,3,4,*]

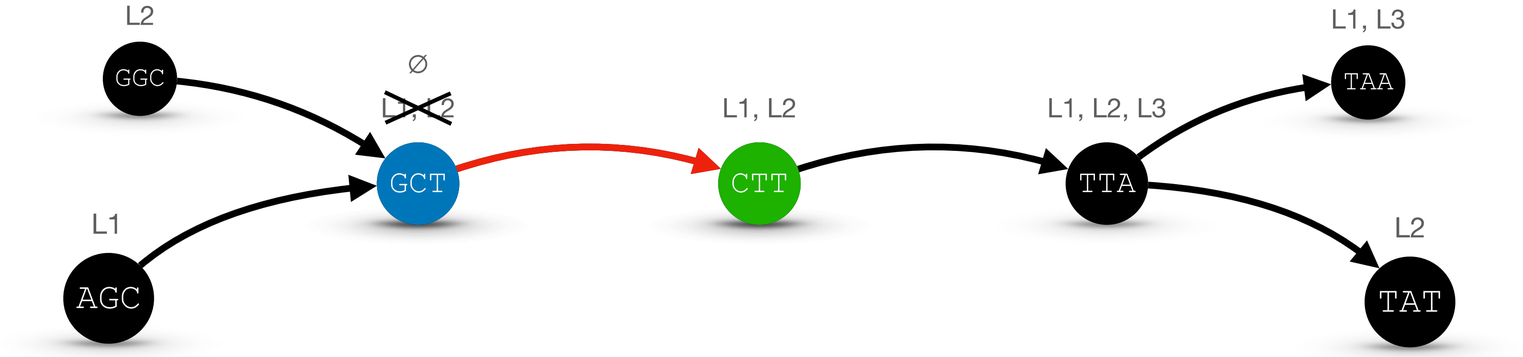
Idea:

- Store only **diffs**

$$L^{\delta}(v) := L(v) \oplus L(v_{\text{succ}})$$

XOR





RowDiff

Delta compression

Bioinformatics, 37, 2021, i169–i176
doi: 10.1093/bioinformatics/btab330
ISMB/ECCB 2021
OXFORD

Topology-based sparsification of graph annotations

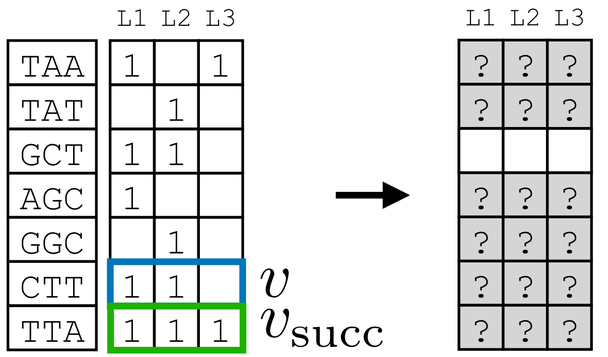
Daniel Danciu[1,2,†], Mikhail Karasikov[1,2,3,†], Harun Mustafa[1,2,3], André Kahles[1,2,3,*] and Gunnar Rätsch[1,2,3,4,*]

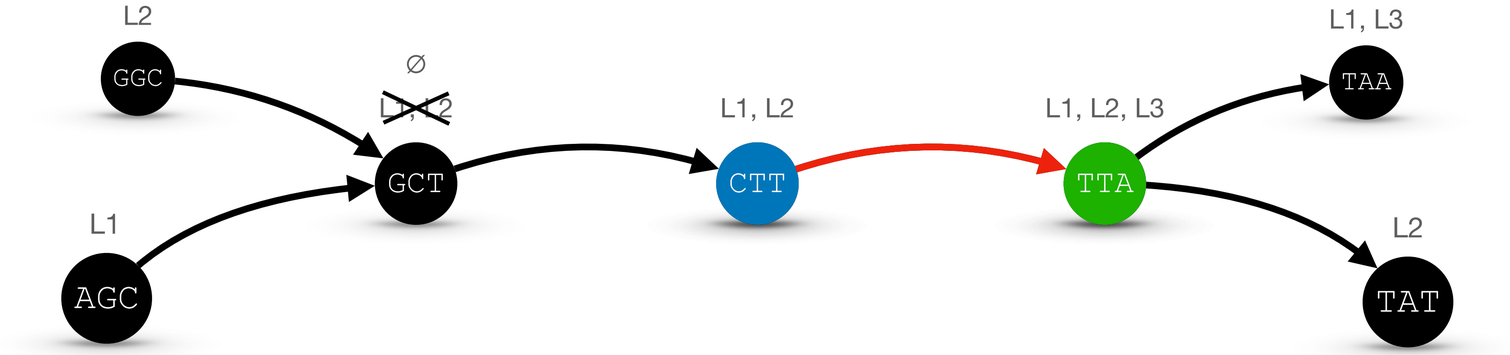
Idea:

- Store only **diffs**

$$L^{\delta}(v) := L(v) \oplus L(v_{\text{succ}})$$

XOR





RowDiff

Delta compression

Bioinformatics, 37, 2021, i169–i176
doi: 10.1093/bioinformatics/btab330
ISMB/ECCB 2021
OXFORD

Topology-based sparsification of graph annotations

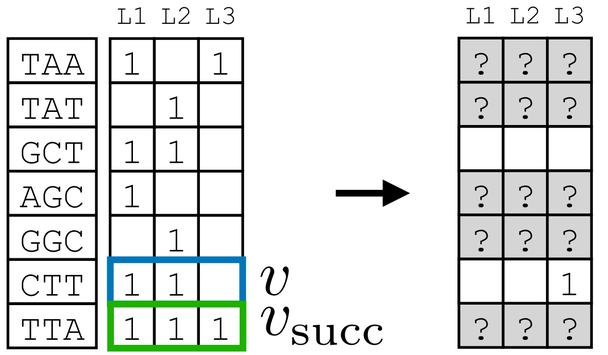
Daniel Danciu[1,2,†], Mikhail Karasikov[1,2,3,†], Harun Mustafa[1,2,3], André Kahles[1,2,3,*] and Gunnar Rätsch[1,2,3,4,*]

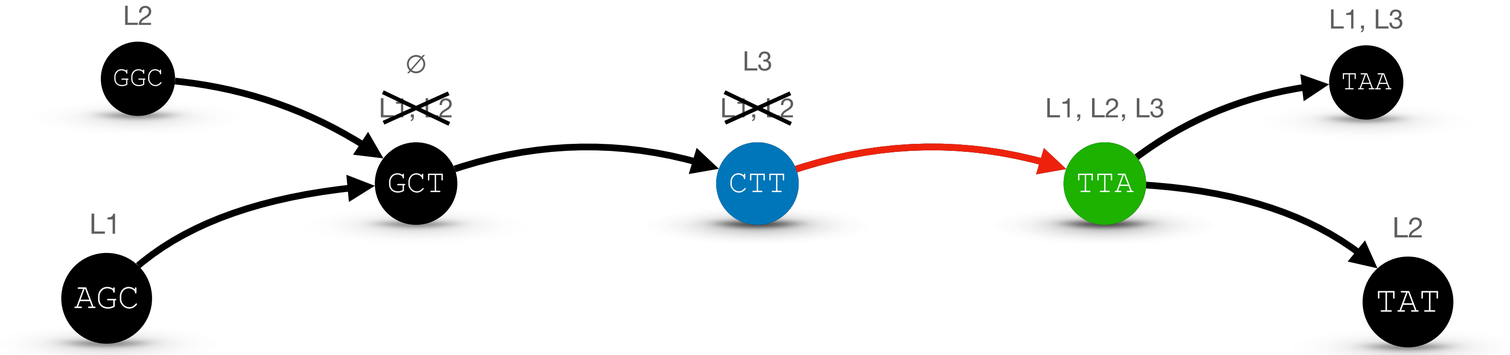
Idea:

- Store only **diffs**

$$L^{\delta}(v) := L(v) \oplus L(v_{\text{succ}})$$

XOR





RowDiff

Delta compression

Bioinformatics, 37, 2021, i169–i176
doi: 10.1093/bioinformatics/btab330
ISMB/ECCB 2021
OXFORD

Topology-based sparsification of graph annotations

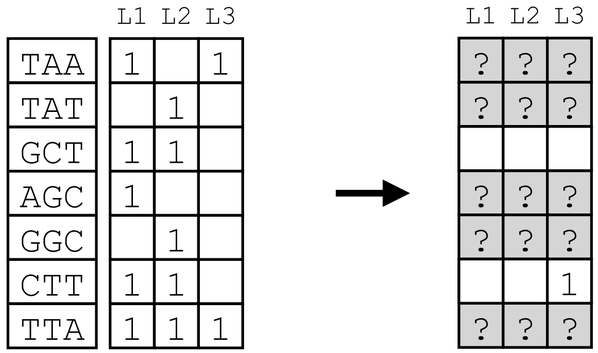
Daniel Danciu[1,2,†], Mikhail Karasikov[1,2,3,†], Harun Mustafa[1,2,3], André Kahles[1,2,3,*] and Gunnar Rätsch[1,2,3,4,*]

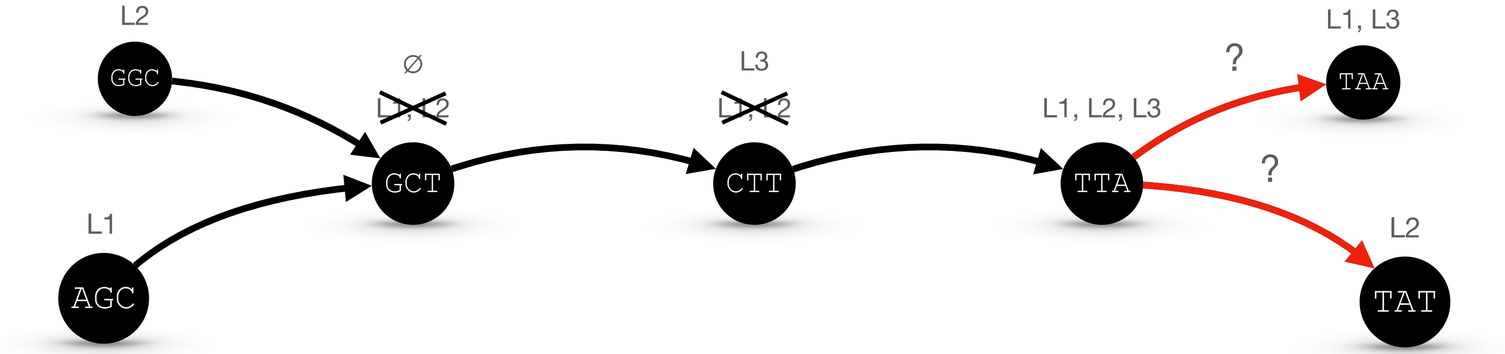
Idea:

- Store only **diffs**

$$L^{\delta}(v) := L(v) \oplus L(v_{\text{succ}})$$

XOR





RowDiff

Delta compression

Bioinformatics, 37, 2021, i169–i176
doi: 10.1093/bioinformatics/btab330
ISMB/ECCB 2021
OXFORD

Topology-based sparsification of graph annotations

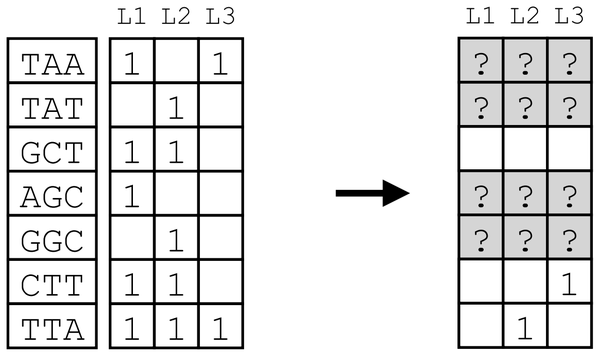
Daniel Danciu[1,2,†], Mikhail Karasikov[1,2,3,†], Harun Mustafa[1,2,3], André Kahles[1,2,3,*] and Gunnar Rätsch[1,2,3,4,*]

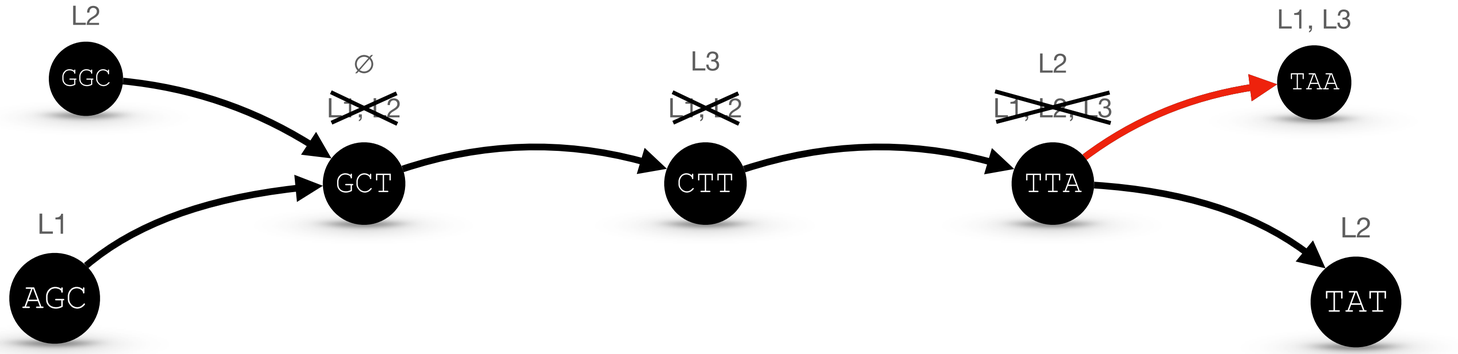
Idea:

- Store only **diffs**

$$L^{\delta}(v) := L(v) \oplus L(v_{\text{succ}})$$

XOR





RowDiff

Delta compression

Bioinformatics, 37, 2021, i169–i176
doi: 10.1093/bioinformatics/btab330
ISMB/ECCB 2021
OXFORD

Topology-based sparsification of graph annotations

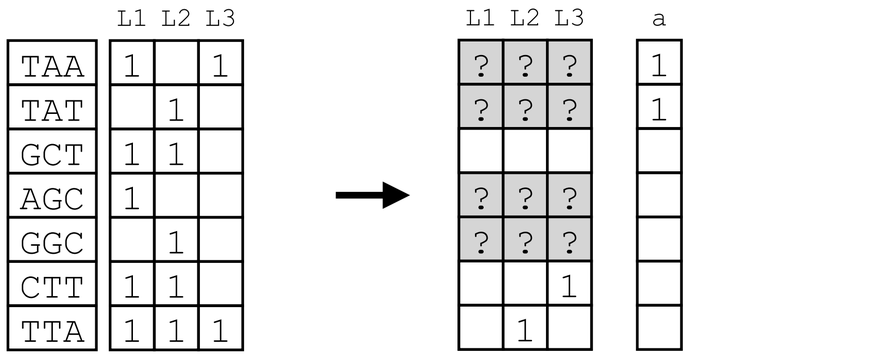
Daniel Danciu[1,2,†], Mikhail Karasikov[1,2,3,†], Harun Mustafa[1,2,3], André Kahles[1,2,3,*] and Gunnar Rätsch[1,2,3,4,*]

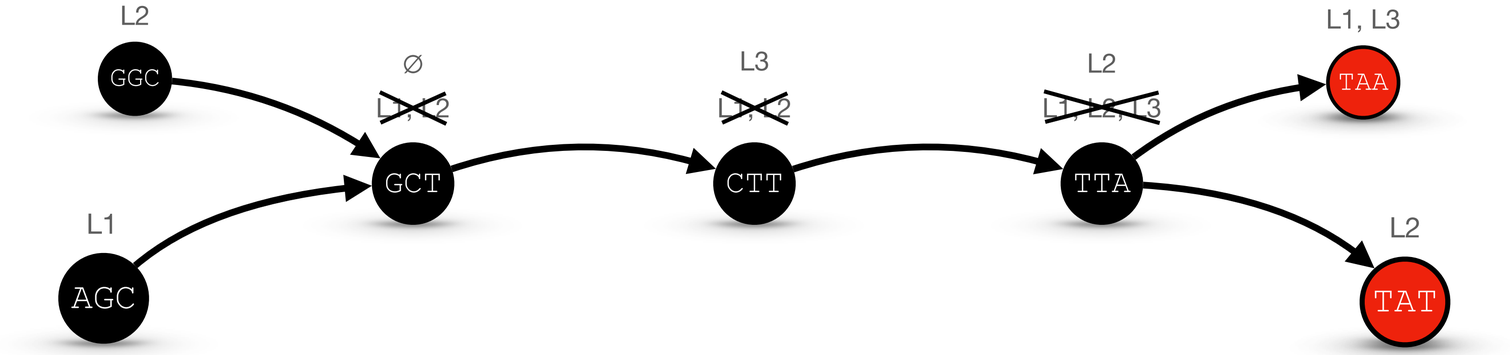
Idea:

- Store only **diffs**

$$L^{\delta}(v) := L(v) \oplus L(v_{\text{succ}})$$

XOR





RowDiff

Delta compression

Bioinformatics, 37, 2021, i169–i176
doi: 10.1093/bioinformatics/btab330
ISMB/ECCB 2021
OXFORD

Topology-based sparsification of graph annotations

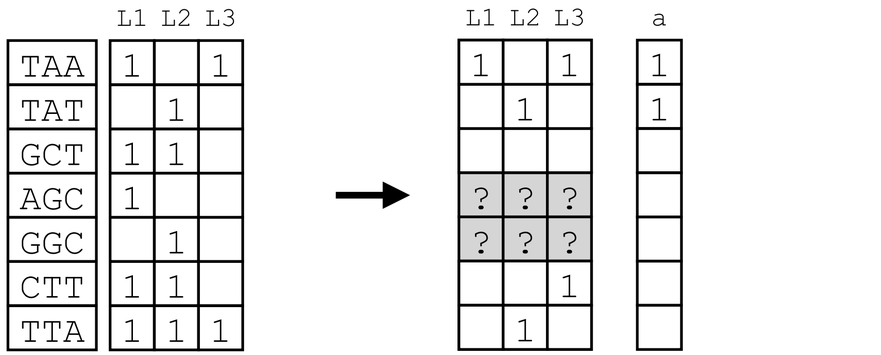
Daniel Danciu[1,2,†], Mikhail Karasikov[1,2,3,†], Harun Mustafa[1,2,3], André Kahles[1,2,3,*] and Gunnar Rätsch[1,2,3,4,*]

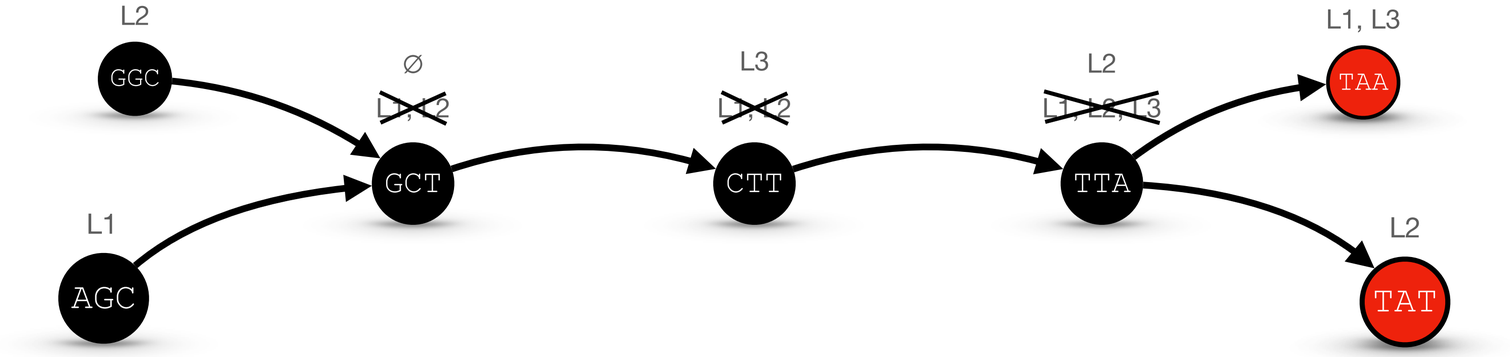
Idea:

- Store only **diffs**

$$L^{\delta}(v) := L(v) \oplus L(v_{\text{succ}})$$

XOR





RowDiff

Delta compression

Bioinformatics, 37, 2021, i169–i176
doi: 10.1093/bioinformatics/btab330
ISMB/ECCB 2021

Topology-based sparsification of graph annotations

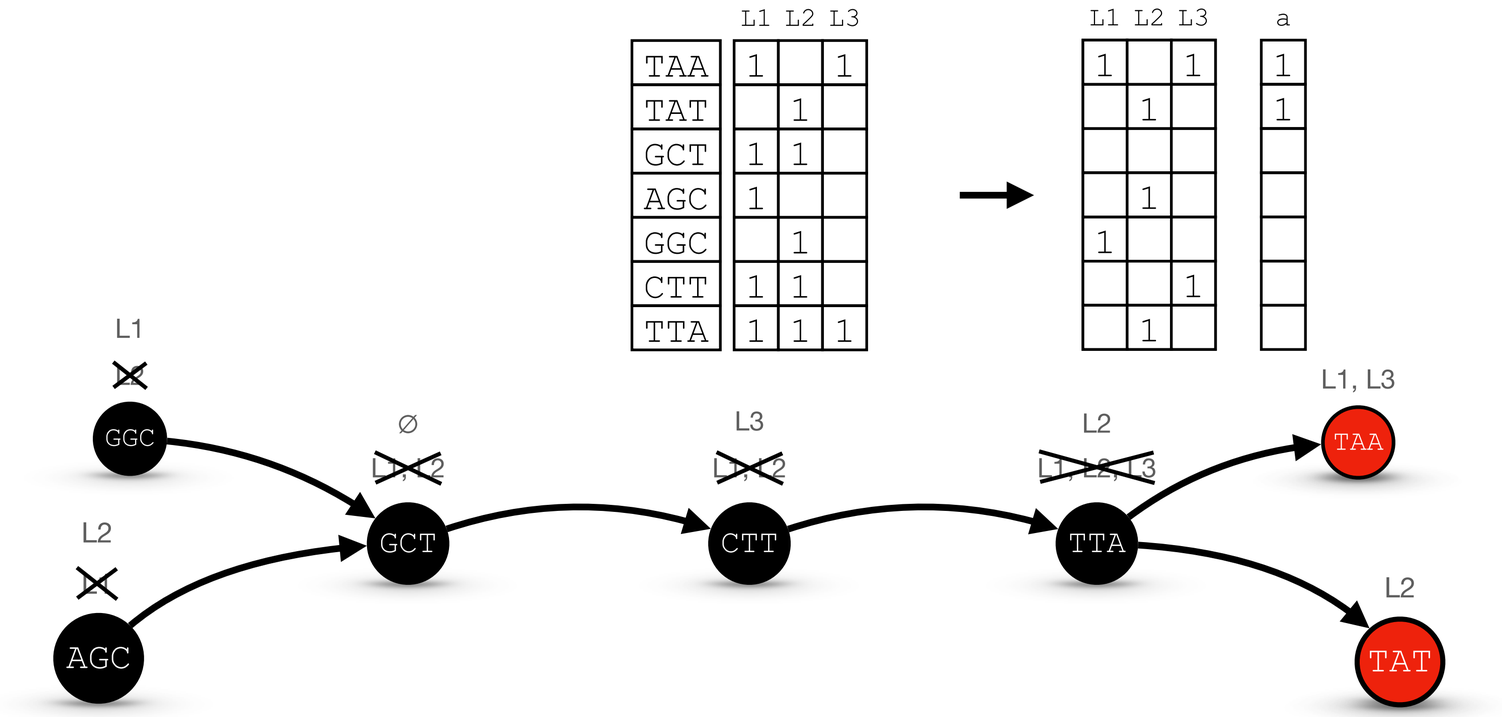
Daniel Danciu[1,2,†], Mikhail Karasikov[1,2,3,†], Harun Mustafa[1,2,3], André Kahles[1,2,3,*] and Gunnar Rätsch[1,2,3,4,*]

Idea:

- Store only **diffs**

$$L^{\delta}(v) := L(v) \oplus L(v_{\text{succ}})$$

XOR



RowDiff effectively **transforms** the matrix:

- **makes it sparser**, and thus, more compressible
- **any representation scheme** can then be applied on top
- the overhead is very small (<1 bit per node)

RowDiff

Delta compression

Bioinformatics, 37, 2021, i169–i176
doi: 10.1093/bioinformatics/btab330
ISMB/ECCB 2021

Topology-based sparsification of graph annotations

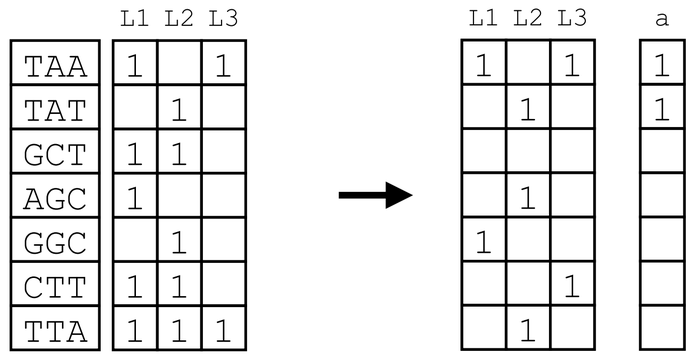
Daniel Danciu[1,2,†], Mikhail Karasikov[1,2,3,†], Harun Mustafa[1,2,3], André Kahles[1,2,3,*] and Gunnar Rätsch[1,2,3,4,*]

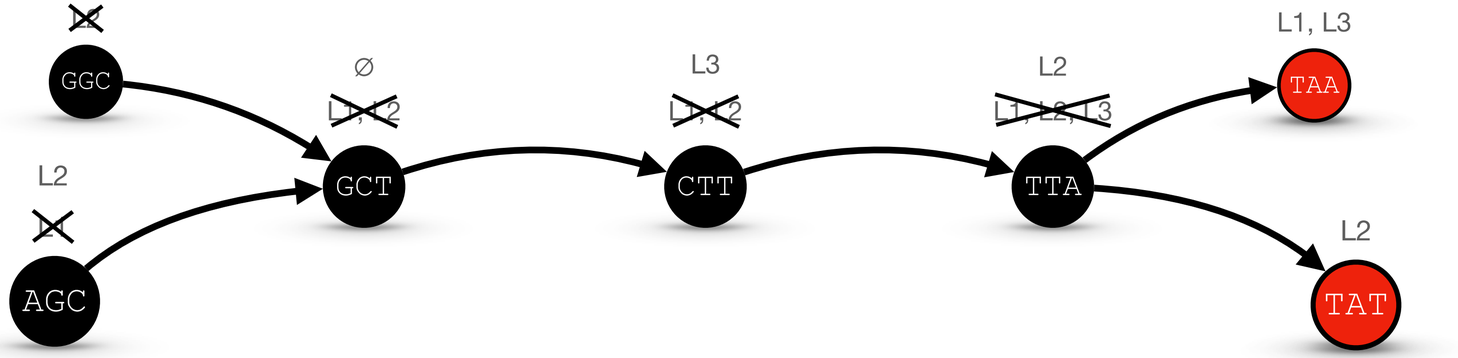
Idea:

- Store only **diffs**
  $L^{\delta}(v) := L(v) \oplus L(v_{\text{succ}})$

- Reconstruct
  $L(v) = L(v_{\text{succ}}) \oplus L^{\delta}(v)$

reconstruct recursively





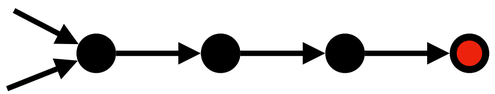
RowDiff effectively **transforms** the matrix:

- **makes it sparser**, and thus, more compressible
- **any representation scheme** can then be applied on top
- the overhead is very small (<1 bit per node)

RowDiff

Anchor nodes

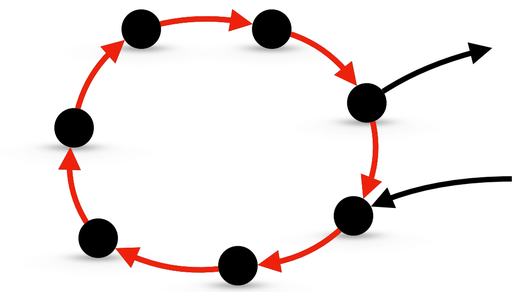
1. **Each *sink* *node*** (with no outgoing edges) **must be anchored**



RowDiff

Anchor nodes

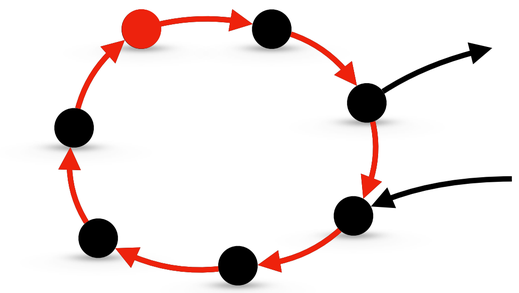
1. **Each *sink* *node*** (with no outgoing edges) **must be anchored**
2. **Each *row-diff cycle*** must have **at least one anchor** node in it



RowDiff

Anchor nodes

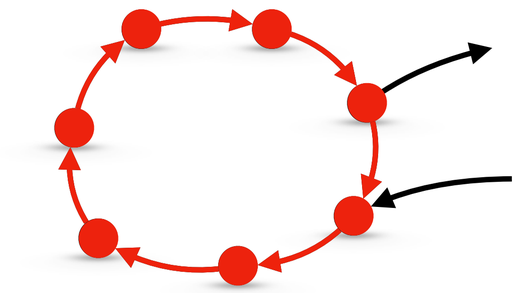
1. **Each *sink* *node*** (with no outgoing edges) **must be anchored**
2. **Each *row-diff cycle*** must have **at least one anchor** node in it



RowDiff

Anchor nodes

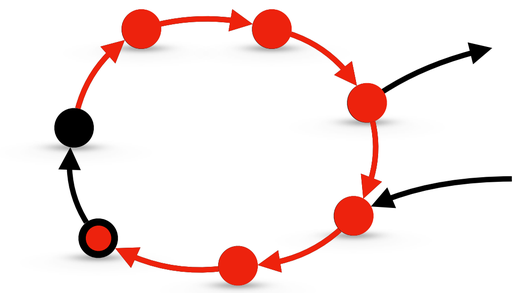
1. **Each *sink* *node*** (with no outgoing edges) **must be anchored**
2. **Each *row-diff cycle*** must have **at least one anchor** node in it



RowDiff

Anchor nodes

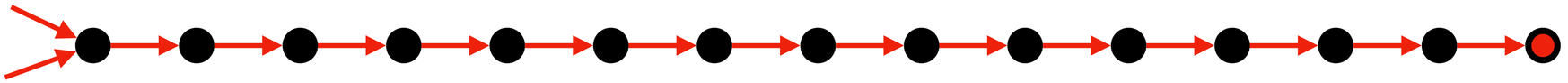
1. **Each *sink* *node*** (with no outgoing edges) **must be anchored**
2. **Each *row-diff cycle*** must have **at least one anchor** node in it



RowDiff

Anchor nodes

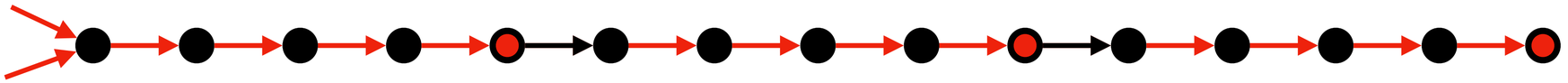
1. **Each *sink* *node*** (with no outgoing edges) **must be anchored**
2. **Each *row-diff cycle*** must have **at least one anchor** node in it
3. **Each *row-diff path* is bounded** (to keep a constant query time)



RowDiff

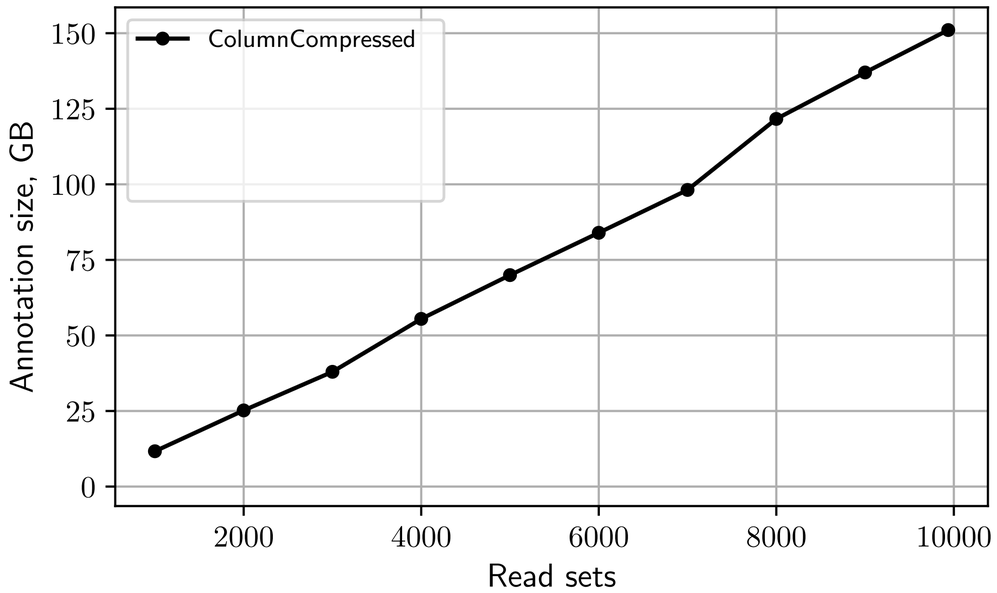
Anchor nodes

1. **Each *sink* *node*** (with no outgoing edges) **must be anchored**
2. **Each *row-diff cycle*** must have **at least one anchor** node in it
3. **Each *row-diff path* is bounded** (to keep a constant query time)



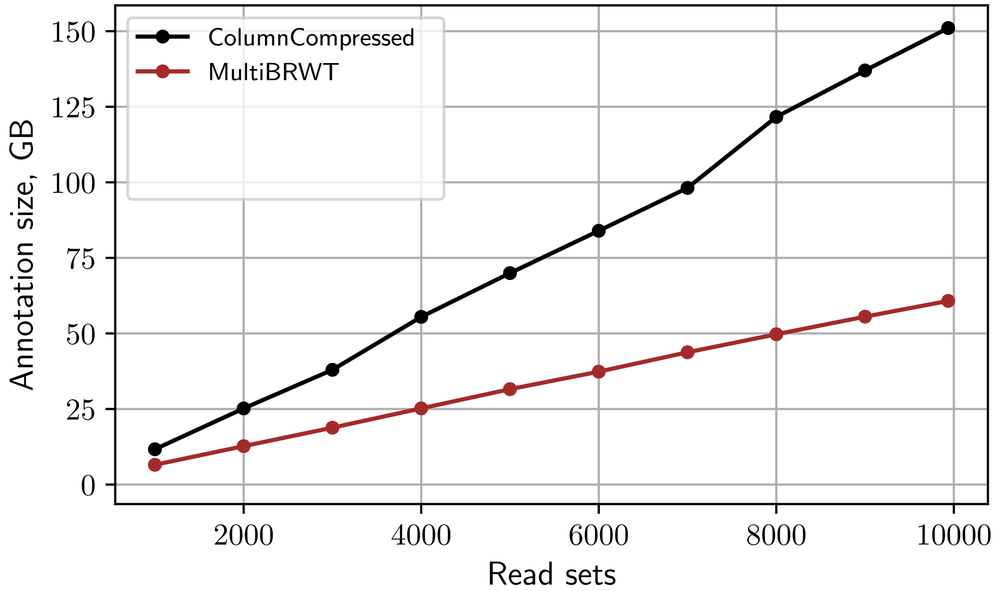
RowDiff: results

Indexing 10,000 (25 TB) human RNA-seq experiments from ENA [Almodaresi et al., 2019]



RowDiff: results

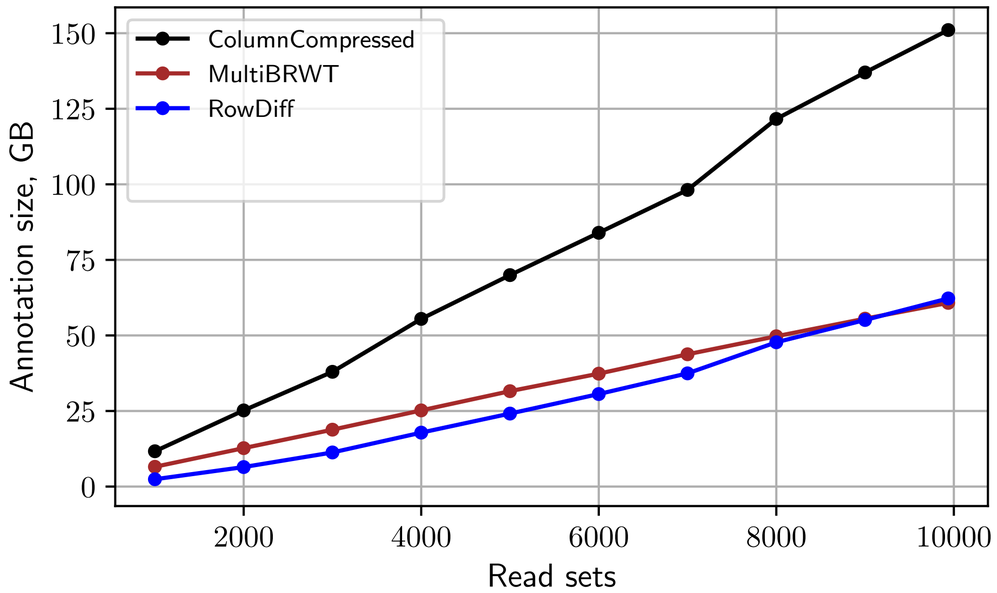
Indexing 10,000 (25 TB) human RNA-seq experiments from ENA [Almodaresi et al., 2019]



- **Multi-BRWT** exploits similarity of **columns**

RowDiff: results

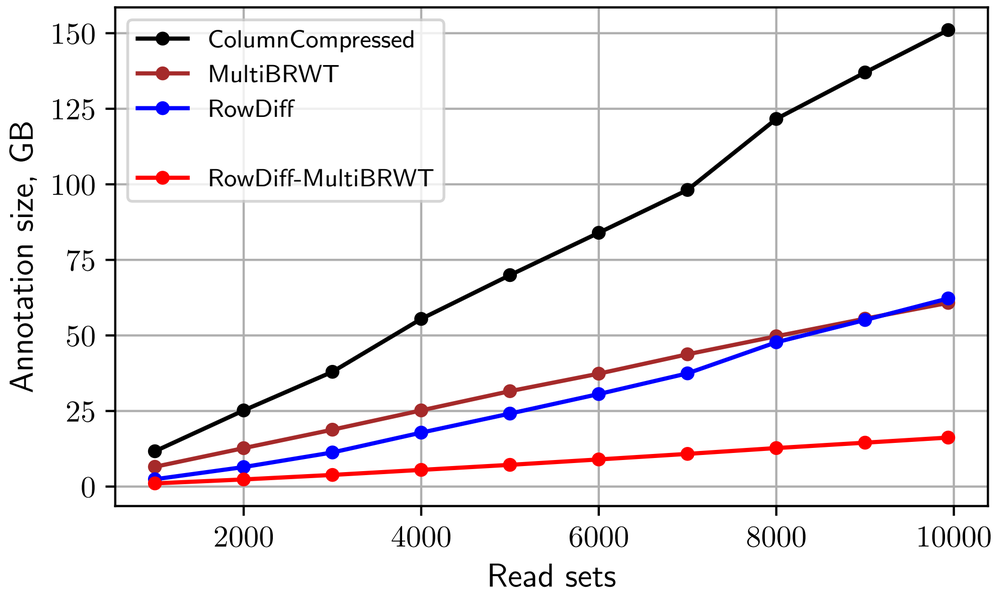
Indexing 10,000 (25 TB) human RNA-seq experiments from ENA [Almodaresi et al., 2019]



- **Multi-BRWT** exploits similarity of **columns**
- **RowDiff** exploits similarity of **rows**
  - effectively transforms the matrix
  - can be combined with any representation scheme

RowDiff: results

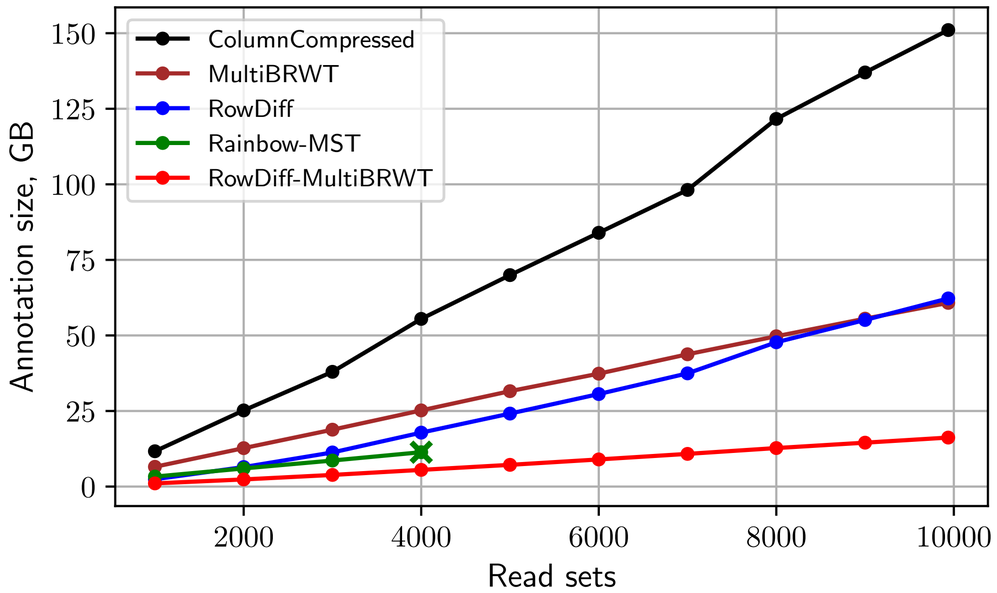
Indexing 10,000 (25 TB) human RNA-seq experiments from ENA [Almodaresi et al., 2019]



- **Multi-BRWT** exploits similarity of **columns**
- **RowDiff** exploits similarity of **rows**
  - effectively transforms the matrix
  - can be combined with any representation scheme

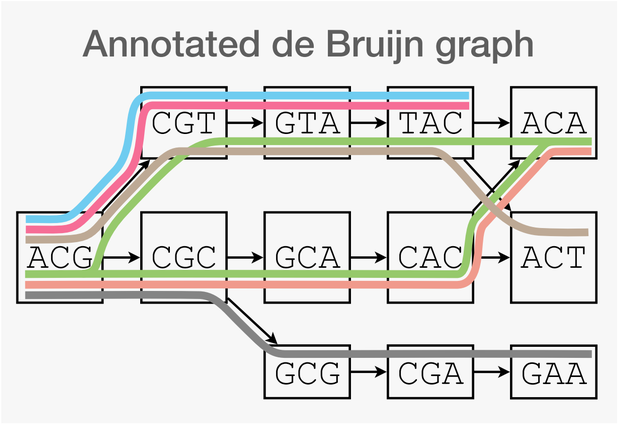
RowDiff: results

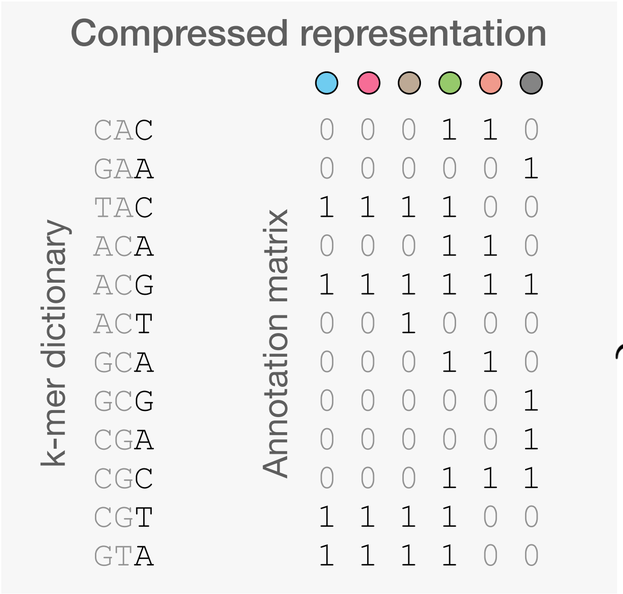
Indexing 10,000 (25 TB) human RNA-seq experiments from ENA [Almodaresi et al., 2019]



- **Multi-BRWT** exploits similarity of **columns**
- **RowDiff** exploits similarity of **rows**
  - effectively transforms the matrix
  - can be combined with any representation scheme
- When combined together, **outperform** the **state-of-the-art**

Representing graph annotations





$\sim 10^{11}$

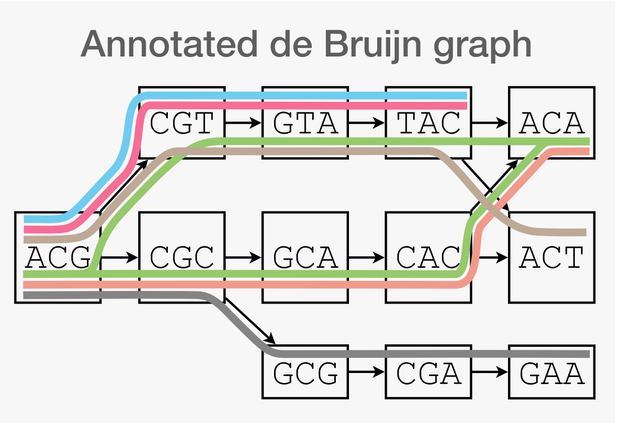
$\sim 10^{6}$

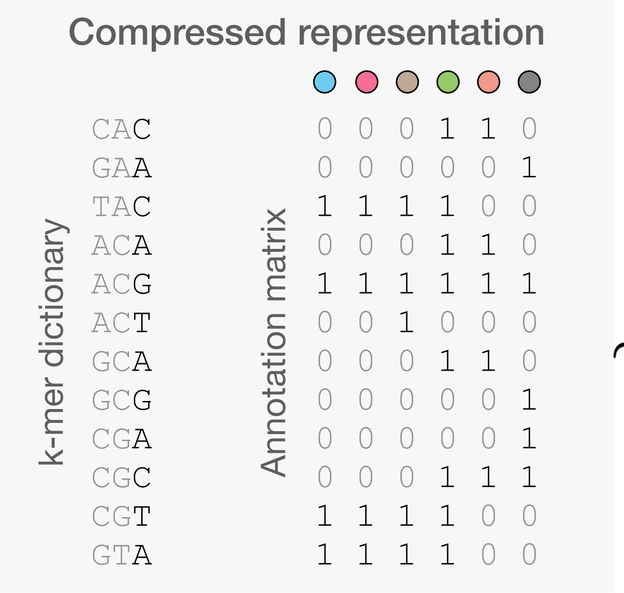
Challenge:
Represent huge-scale annotations

Typical properties

1. high sparsity
2. similarity of **columns** $\Rightarrow$ **Multi-BRWT**
3. similarity of **rows** $\Rightarrow$ **RowDiff**

Representing graph annotations





$\sim 10^{11}$

$\sim 10^{6}$

Challenge:
Represent huge-scale annotations

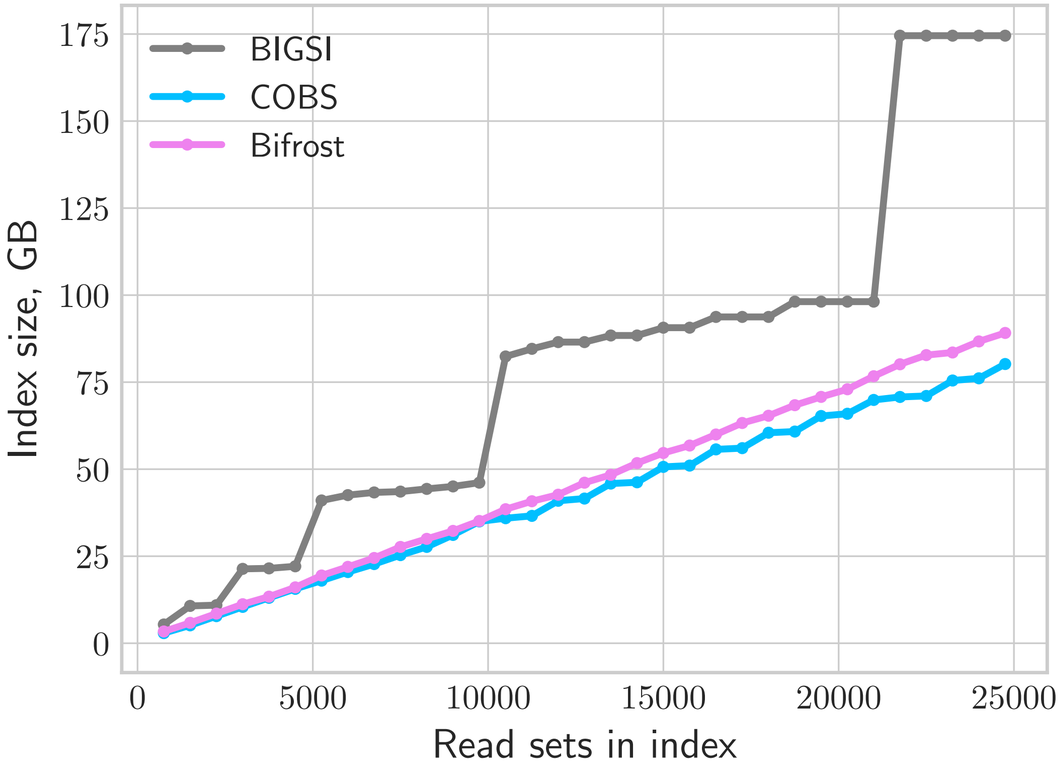
✔

Typical properties

1. high sparsity
2. similarity of **columns** ⇒ **Multi-BRWT**
3. similarity of **rows** ⇒ **RowDiff**

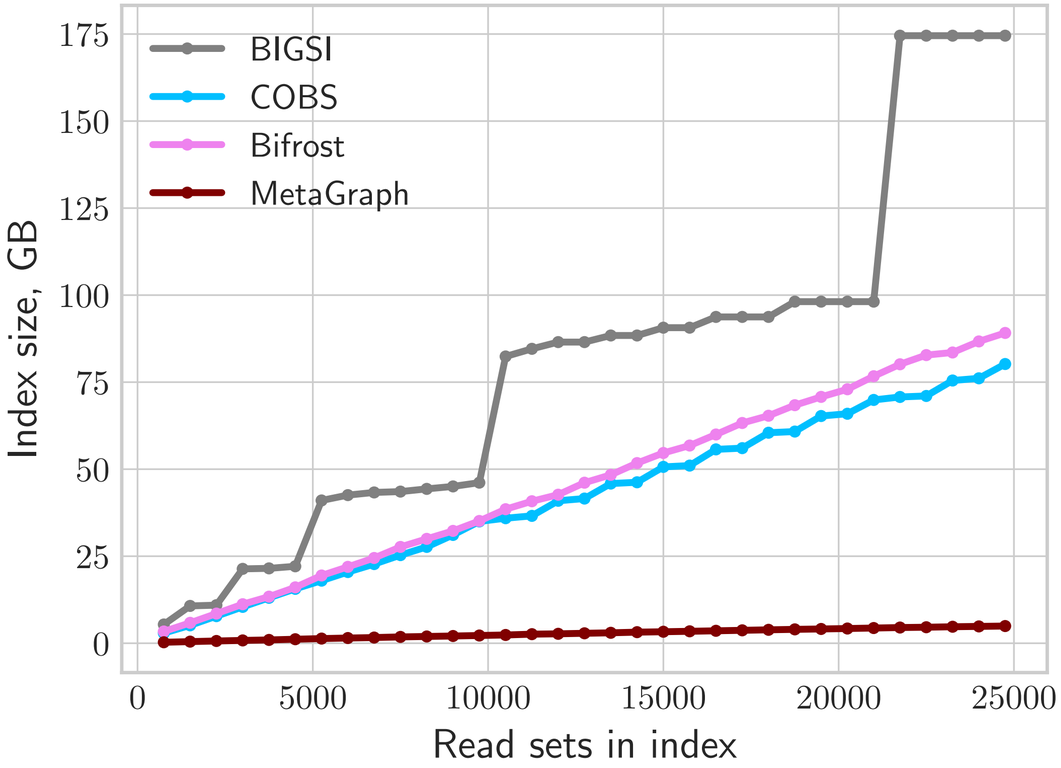
Scalability of MetaGraph

Indexing microbial samples (BIGSI [Bradley *et al.*, 2019] subsets)



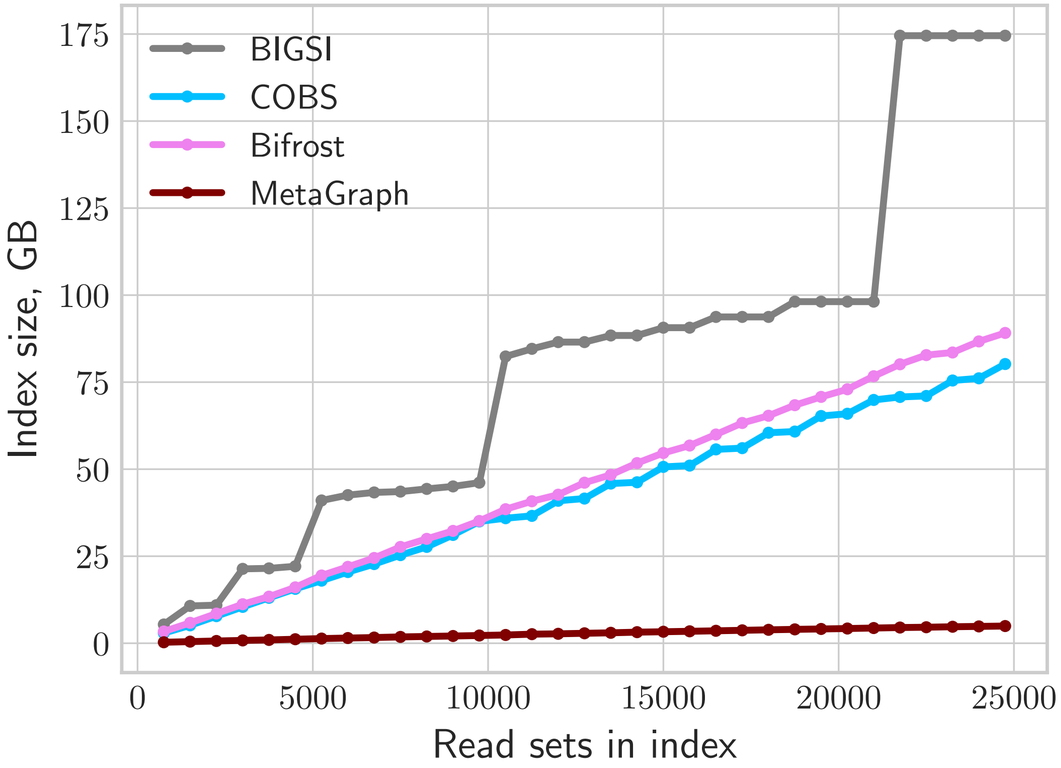
Scalability of MetaGraph

Indexing microbial samples (BIGSI [Bradley *et al.*, 2019] subsets)

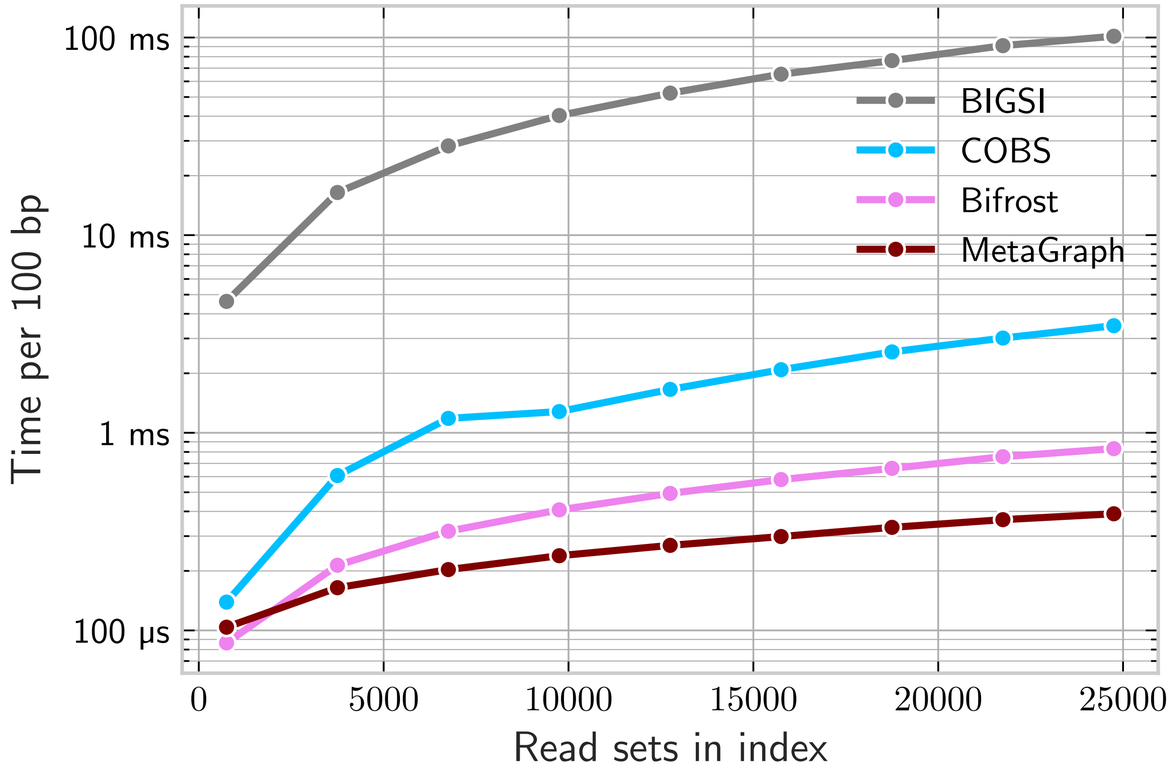


Scalability of MetaGraph

Indexing microbial samples (BIGSI [Bradley *et al.*, 2019] subsets)

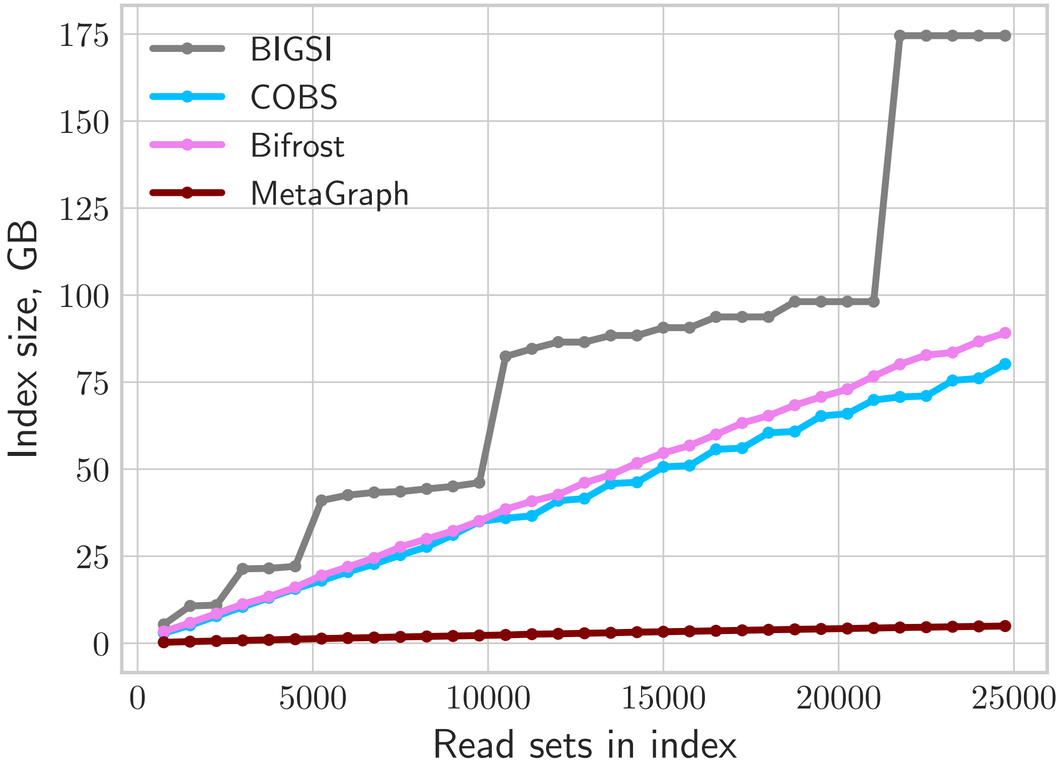


Querying DRR067889

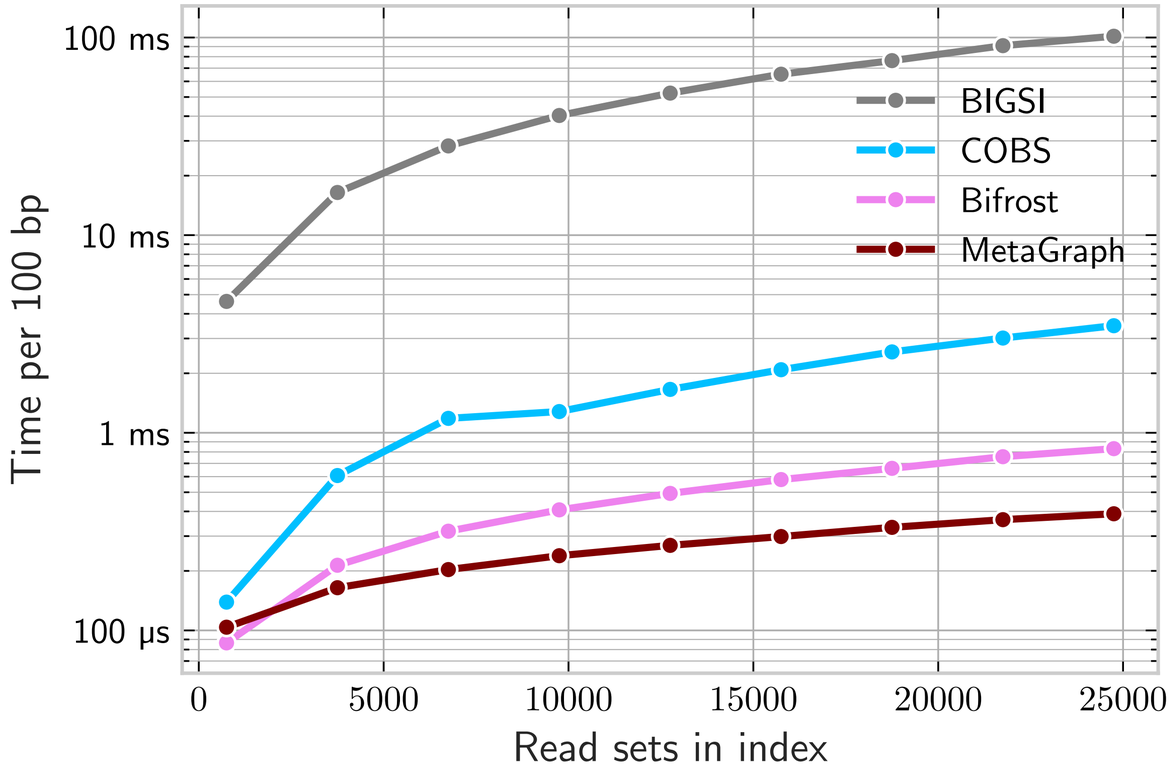


Scalability of MetaGraph

Indexing microbial samples (BIGSI [Bradley *et al.*, 2019] subsets)



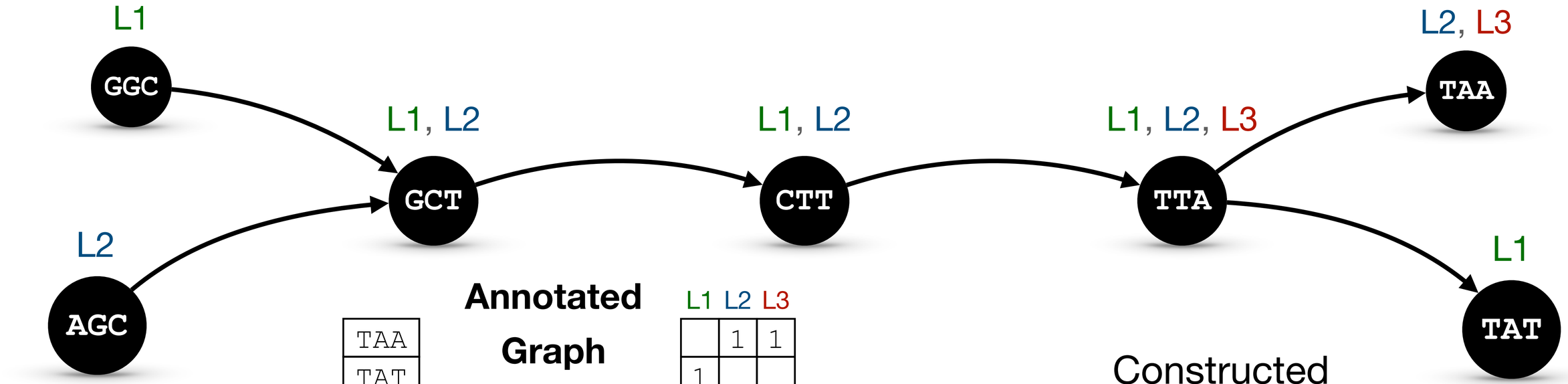
Querying DRR067889



- Use of **succinct data structures** and efficient representation schemes
- Choice of **efficient algorithms** (e.g., batch operations)

From qualitative to quantitative

0. Representing **presence/absence**



Annotated Graph

TAA
TAT
GCT
AGC
GGC
CTT
TTA

k-mer dictionary

L1	L2	L3
	1	1
1		
1	1	
	1	
1		
1	1	
1	1	1

graph annotation

Constructed from sequences

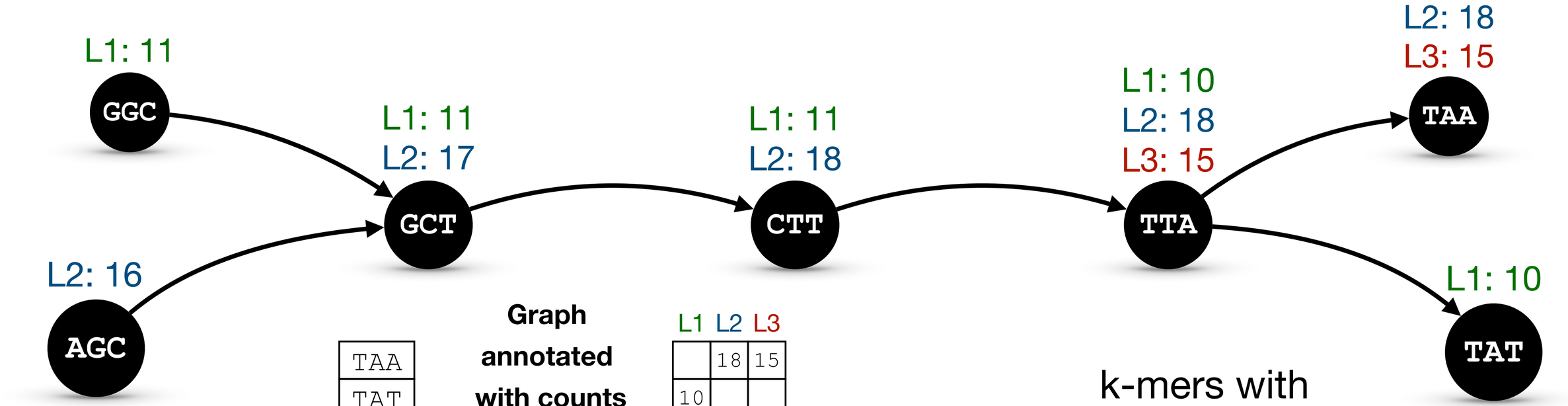
L1: GGCTTAT

L2: AGCTTAA

L3: TTAA

From qualitative to quantitative

1. Representing **k-mer abundances**



TAA
TAT
GCT
AGC
GGC
CTT
TTA

k-mer dictionary

Graph annotated with counts

L1	L2	L3
	18	15
10		
11	17	
	16	
11		
11	18	
10	18	15

graph annotation

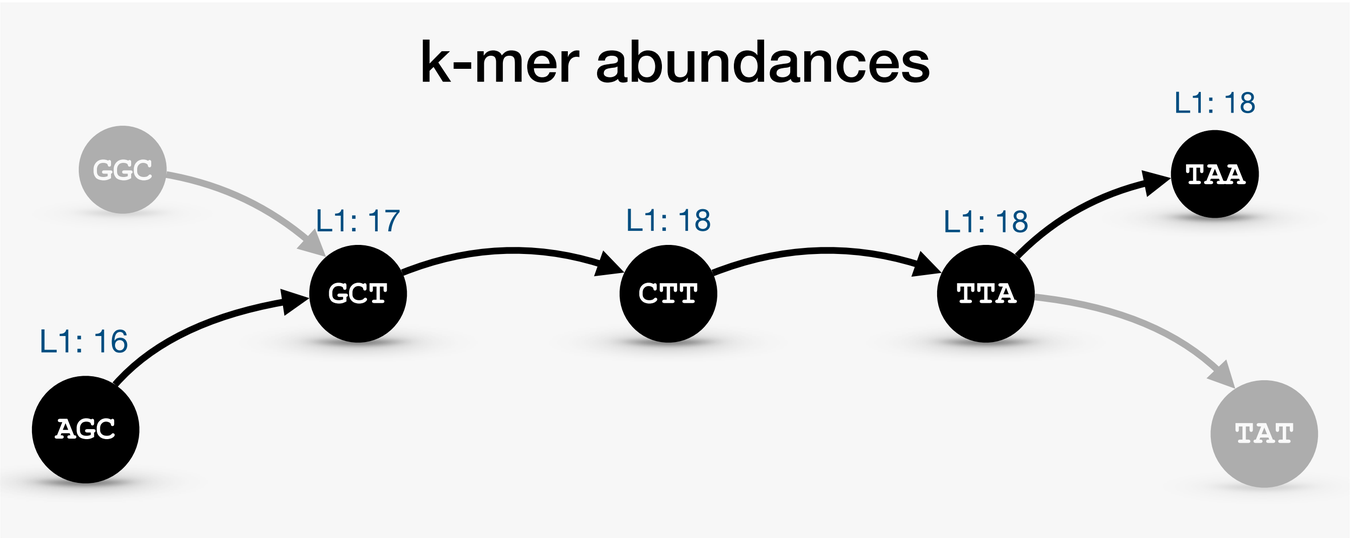
k-mers with multiplicities

L1: `GGC: 11, …`

L2: `AGC: 16, …`

L3: `TTA: 15, …`

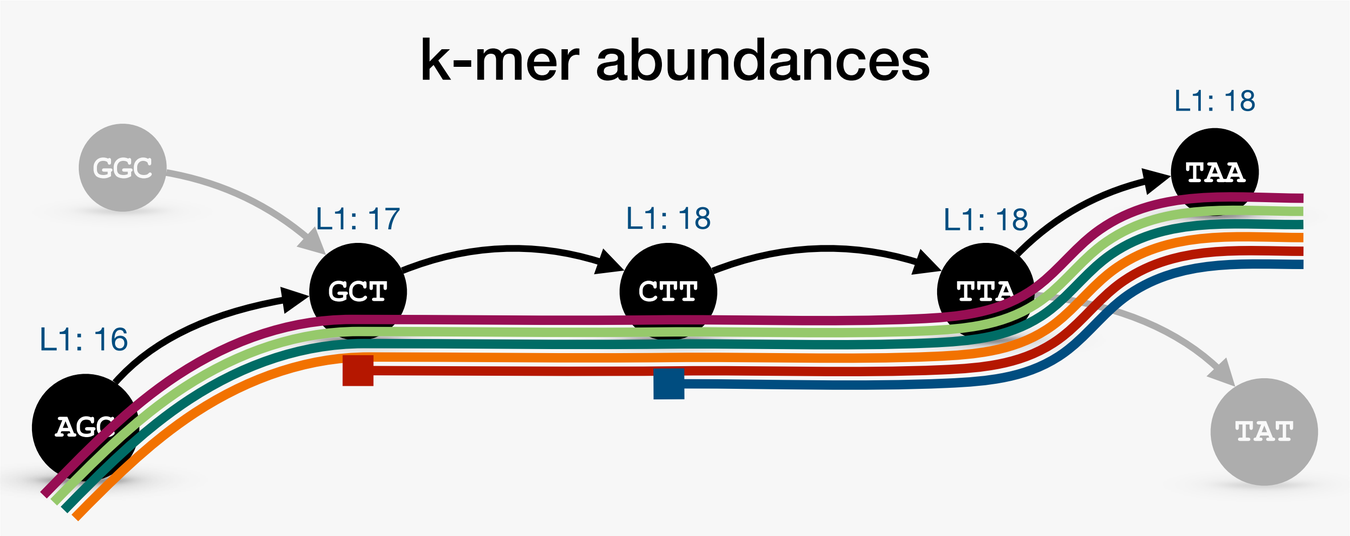
Exploiting regularities



Observe regularities:

Abundances of adjacent k-mers are often similar

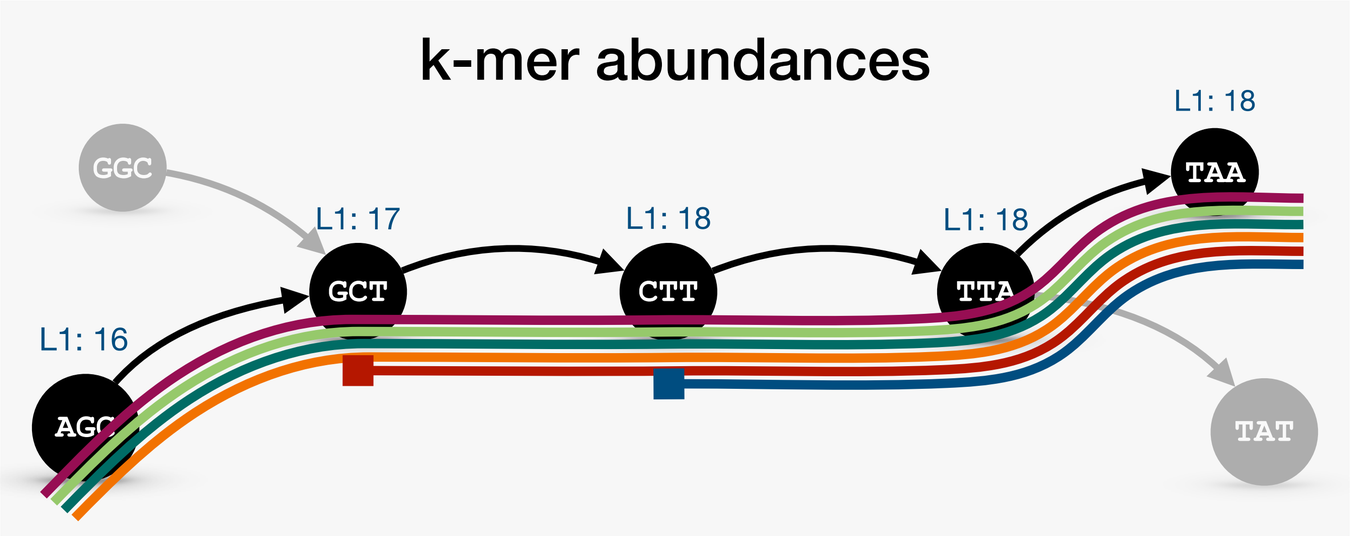
Exploiting regularities



Observe regularities:

Abundances of adjacent k-mers are often similar

Exploiting regularities



Observe regularities:

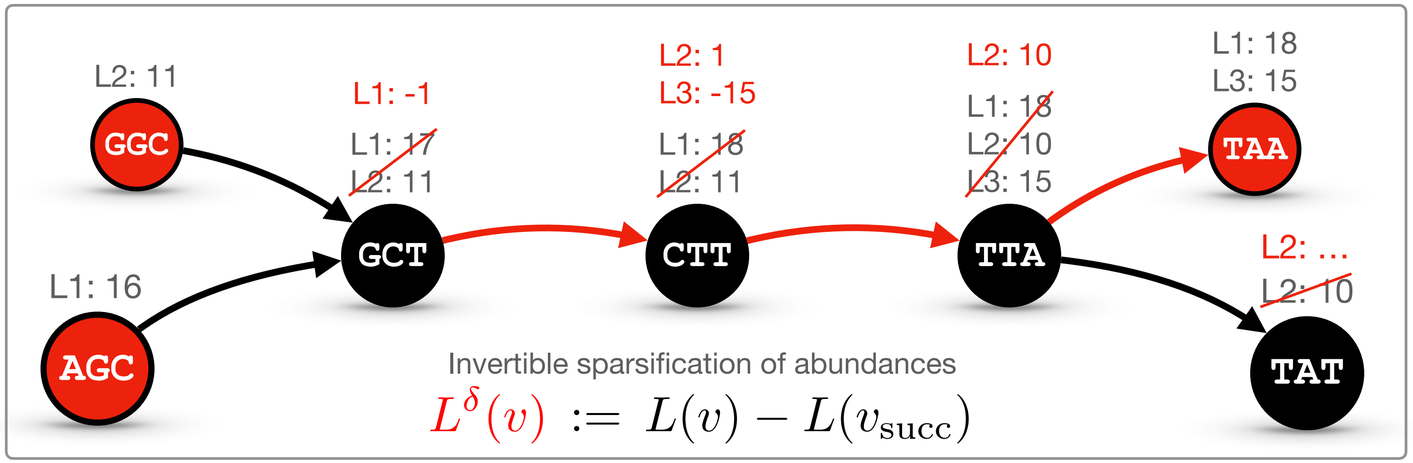
Abundances of adjacent k-mers are often similar

Idea:

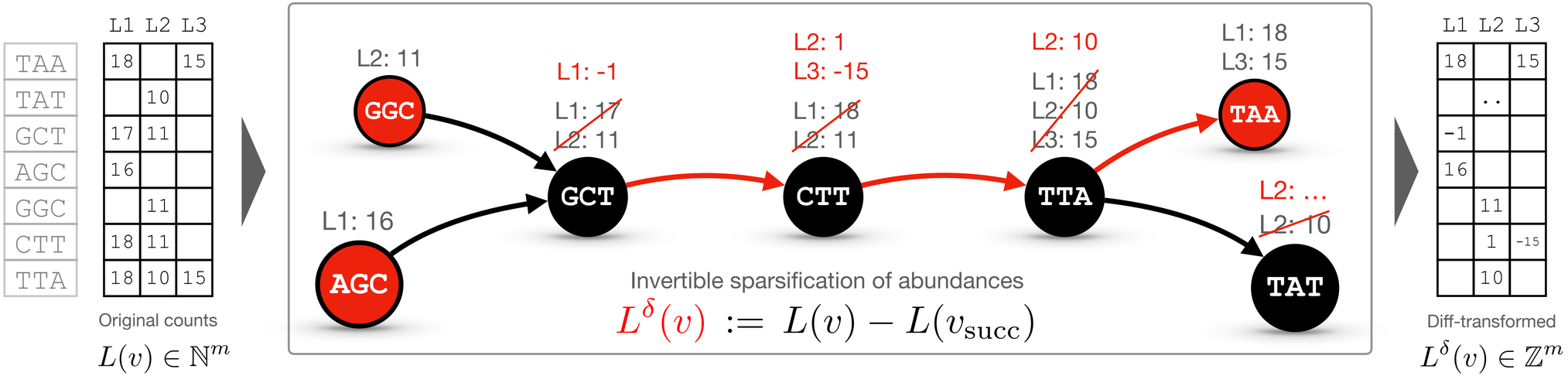
Generalize the RowDiff scheme

Lossless Indexing with Counting de Bruijn Graphs
Mikhail Karasikov, Harun Mustafa, Gunnar Rätsch, André Kahles
Genome Res. 2022; doi: https://doi.org/10.1101/gr.276607.122

RowDiff for k-mer abundances



RowDiff for k-mer abundances



Indexing k-mer counts

Lossless Indexing with Counting de Bruijn Graphs
Mikhail Karasikov, Harun Mustafa, Gunnar Rätsch, André Kahles
Genome Res. 2022; doi: https://doi.org/10.1101/gr.276607.122

Indexing 2,586 human RNA-Seq read sets [Solomon, Kingsford, 2016].
Querying 100 random human transcripts (≈ 90 kbp).

Method	Index size	Peak RAM	Query time
REINDEER	59 GB	91 GB	56.5 sec
MetaGraph	**11 GB**	**11 GB**	**17.6 sec**

Indexing k-mer counts

Lossless Indexing with Counting de Bruijn Graphs
Mikhail Karasikov, Harun Mustafa, Gunnar Rätsch, André Kahles
Genome Res. 2022; doi: https://doi.org/10.1101/gr.276607.122

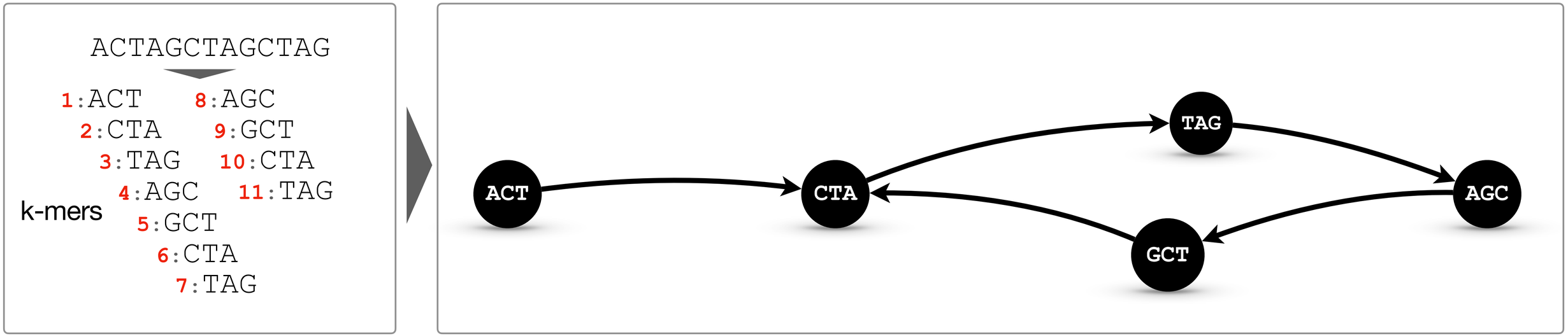
Indexing 2,586 human RNA-Seq read sets [Solomon, Kingsford, 2016].
Querying 100 random human transcripts (≈ 90 kbp).

Method	Index size	Peak RAM	Query time
REINDEER	59 GB	91 GB	56.5 sec
MetaGraph	**11 GB**	**11 GB**	**17.6 sec**

- **5x smaller** representations, uses **8x less RAM**
- **3x faster** to query

From qualitative to quantitative

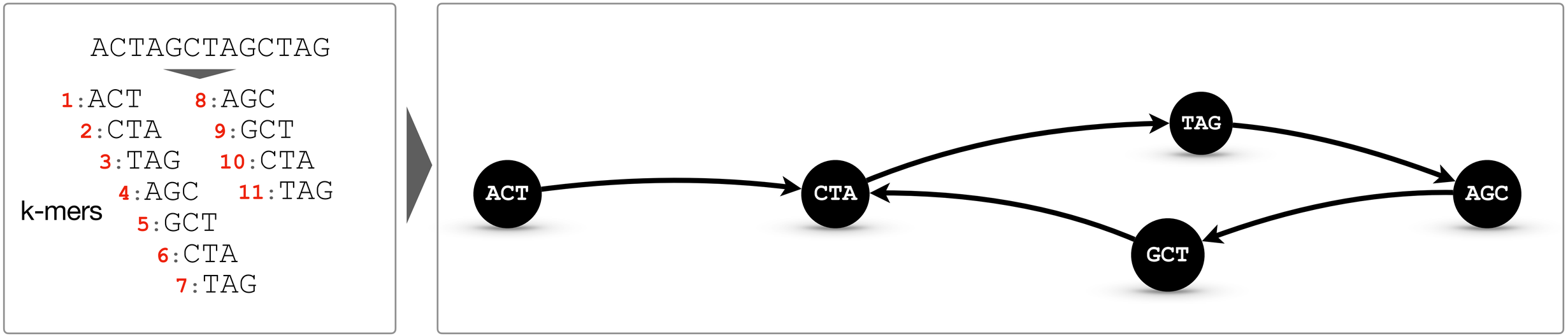
2. Representing **k-mer coordinates**



De Bruijn graph

From qualitative to quantitative

2. Representing **k-mer coordinates**

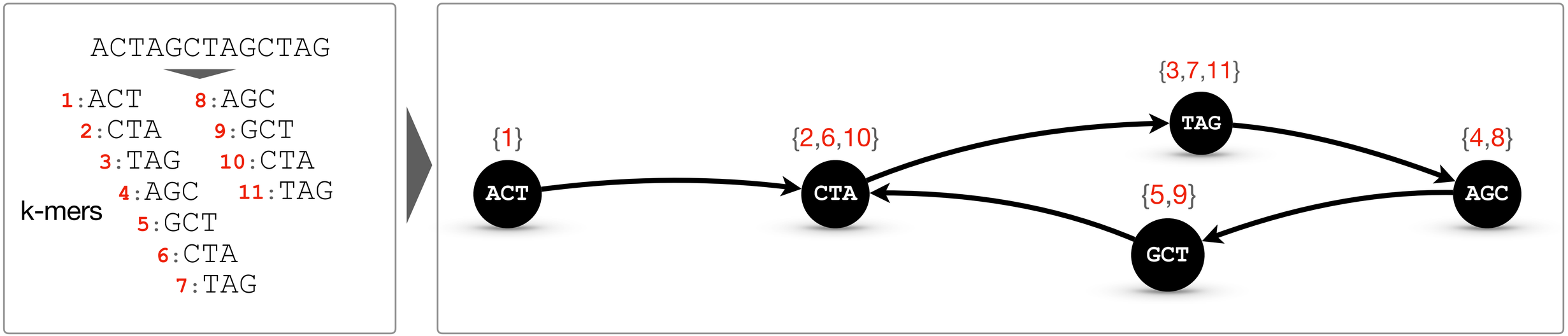


Not invertible representation

De Bruijn graph

From qualitative to quantitative

2. Representing **k-mer coordinates**



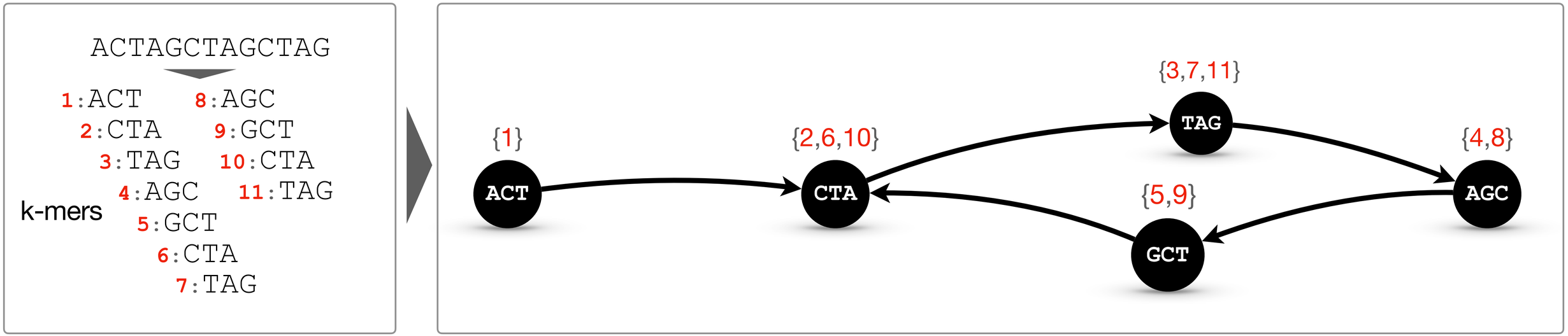
Invertible 👍

De Bruijn graph

with k-mer coordinates

From qualitative to quantitative

2. Representing **k-mer coordinates**



Encoding ***k-mer coordinates*** allows

- reconstructing original sequences

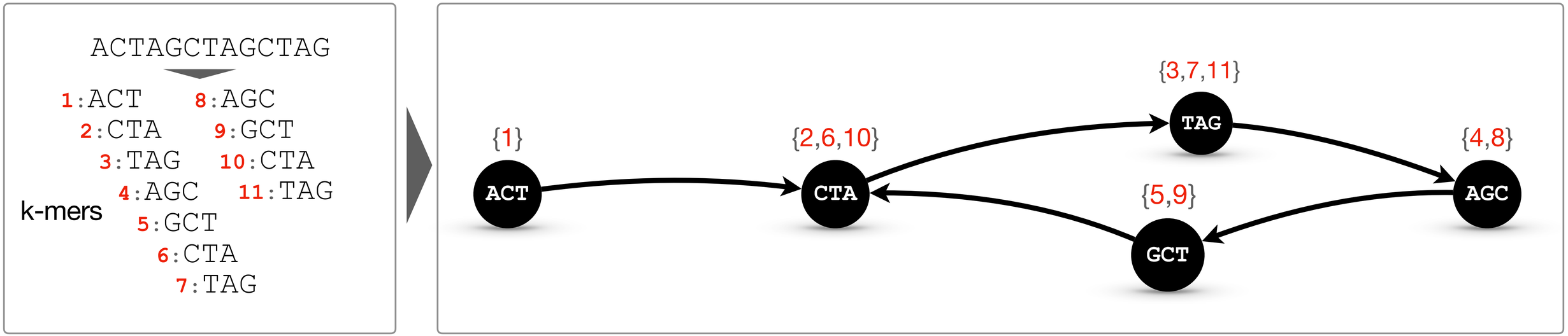
  (hence, **lossless sequence representation)**

De Bruijn graph

with k-mer coordinates

From qualitative to quantitative

2. Representing **k-mer coordinates**



Encoding ***k-mer coordinates*** allows

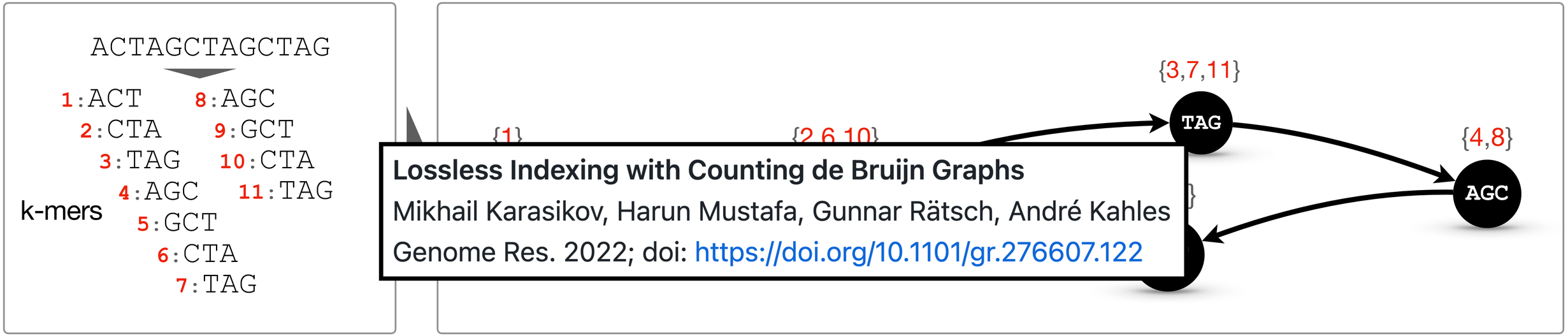
- reconstructing original sequences
  (hence, **lossless sequence representation)**
- performing exact sequence alignment

De Bruijn graph

with k-mer coordinates

From qualitative to quantitative

2. Representing **k-mer coordinates**



Encoding ***k-mer coordinates*** allows

- reconstructing original sequences
  (hence, **lossless sequence representation)**
- performing exact sequence alignment

De Bruijn graph

with k-mer coordinates

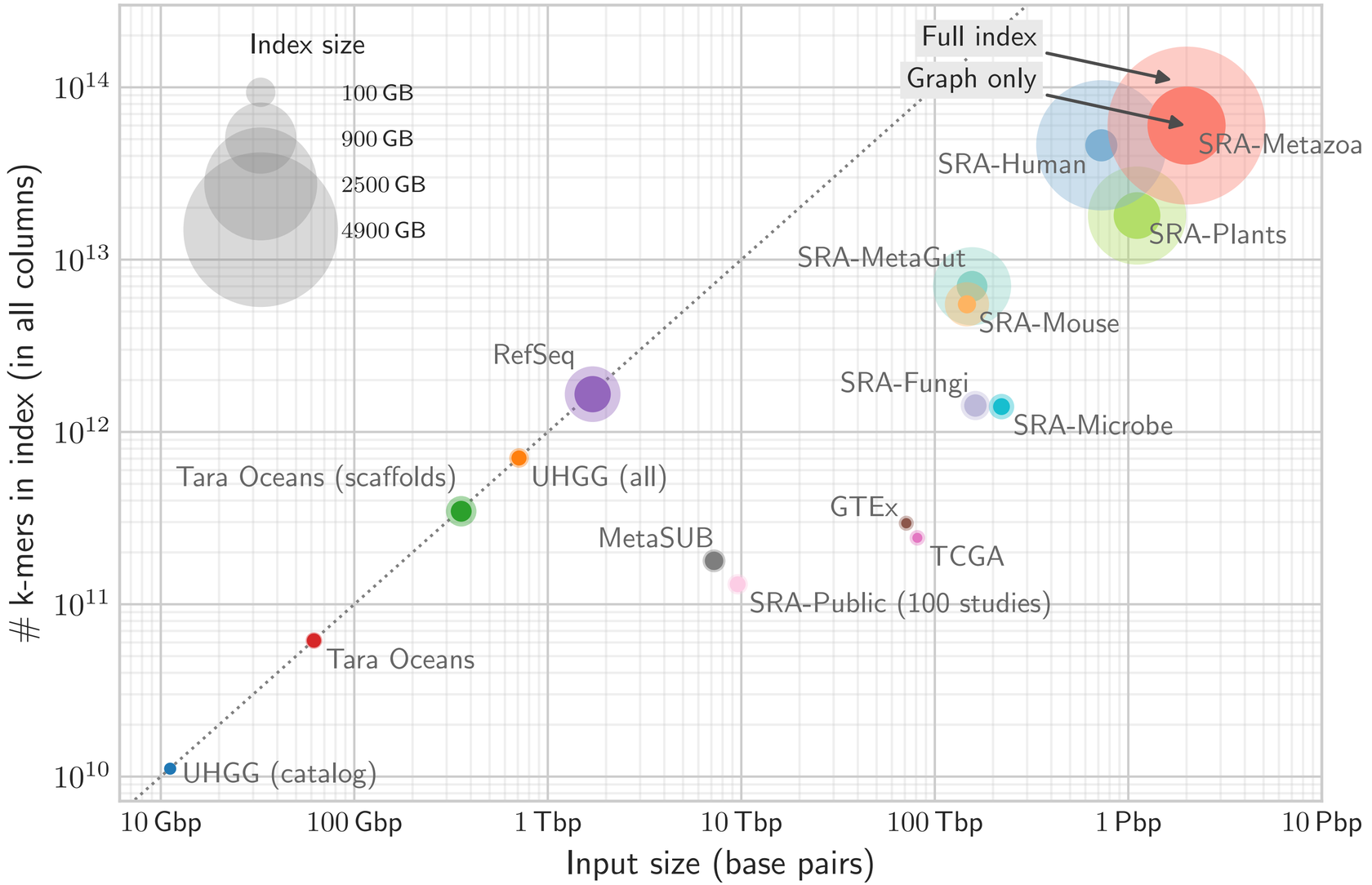
Indexing large sequence archives

Indexing at Petabase-scale

MetaGraph: Indexing and Analysing Nucleotide Archives at Petabase-scale

Mikhail Karasikov, Harun Mustafa, Daniel Danciu, Marc Zimmermann, Christopher Barber, Gunnar Rätsch, André Kahles

doi: https://doi.org/10.1101/2020.10.01.322164



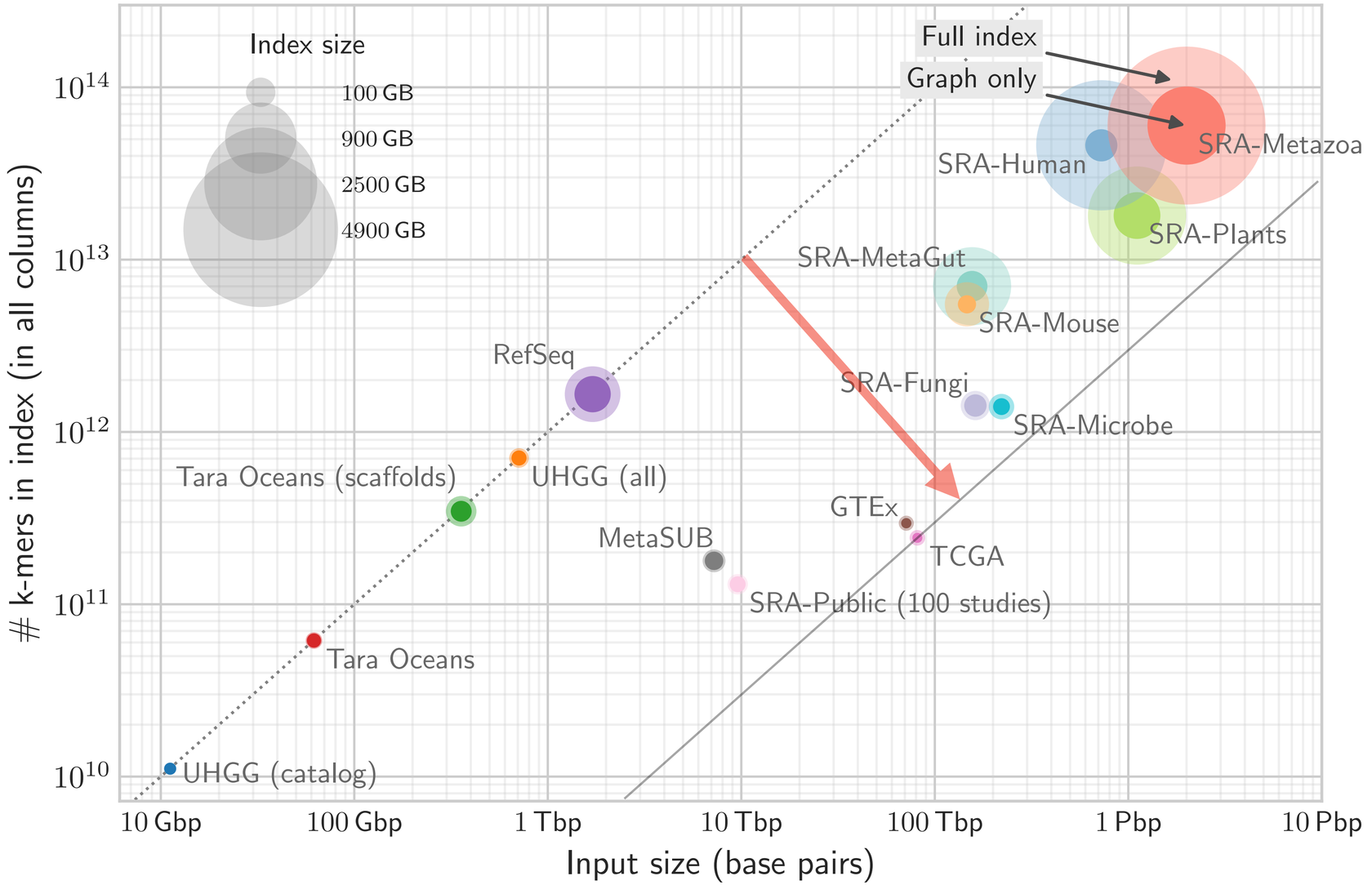
Datasets: UHGG (Almeida *et al.*, 2020), Tara Oceans (Paoli, 2022), GTEx (Lonsdale *et al.*, 2013), TCGA (Kaiser, 2005), MetaSUB (Danko *et al.*, 2019), SRA-Microbe (Bradley *et al.*, 2019)

Indexing at Petabase-scale

MetaGraph: Indexing and Analysing Nucleotide Archives at Petabase-scale

Mikhail Karasikov, Harun Mustafa, Daniel Danciu, Marc Zimmermann, Christopher Barber, Gunnar Rätsch, André Kahles

doi: https://doi.org/10.1101/2020.10.01.322164



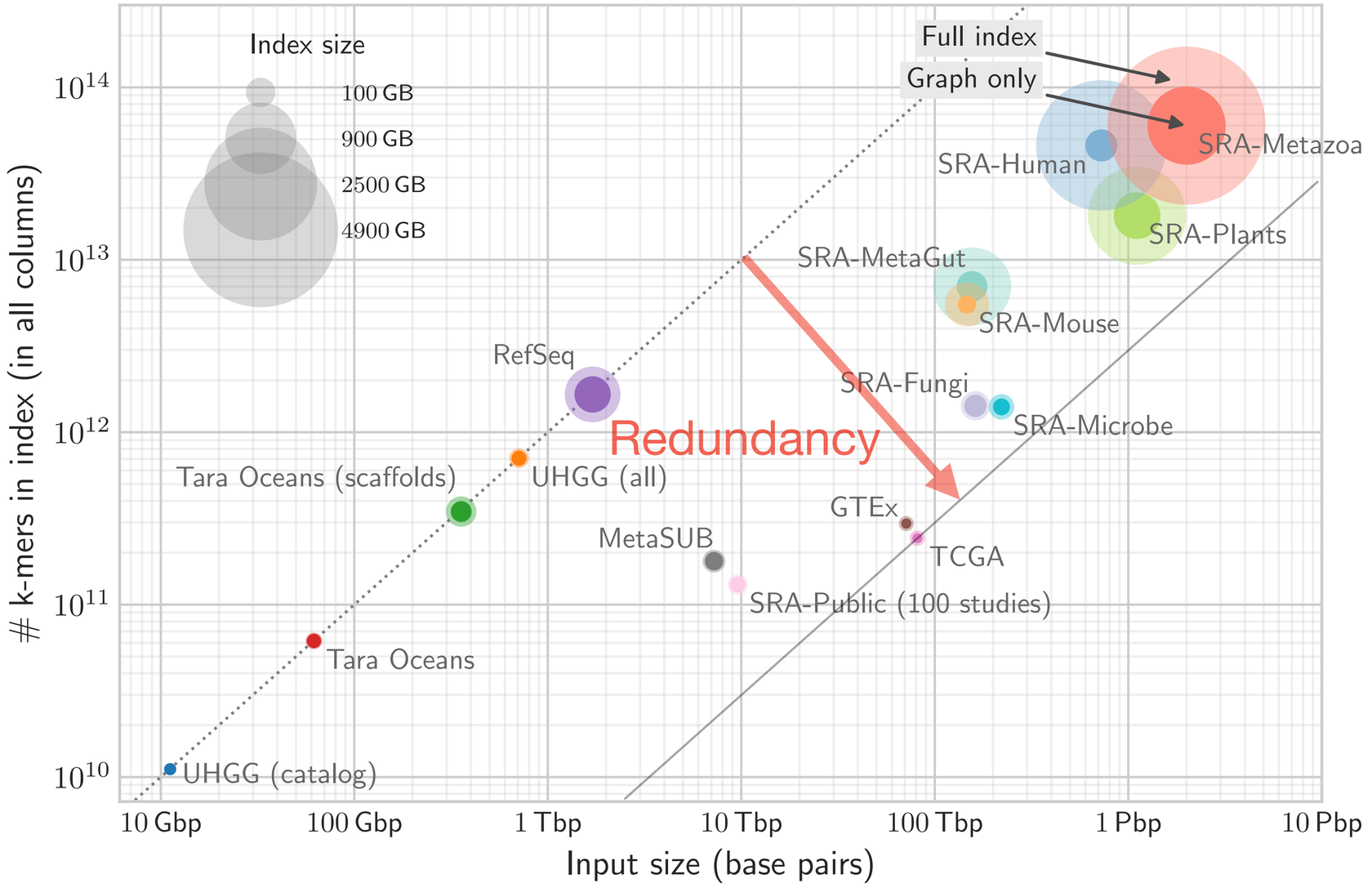
Datasets: UHGG (Almeida *et al.*, 2020), Tara Oceans (Paoli, 2022), GTEx (Lonsdale *et al.*, 2013), TCGA (Kaiser, 2005), MetaSUB (Danko *et al.*, 2019), SRA-Microbe (Bradley *et al.*, 2019)

Indexing at Petabase-scale

MetaGraph: Indexing and Analysing Nucleotide Archives at Petabase-scale

Mikhail Karasikov, Harun Mustafa, Daniel Danciu, Marc Zimmermann, Christopher Barber, Gunnar Rätsch, André Kahles

doi: https://doi.org/10.1101/2020.10.01.322164



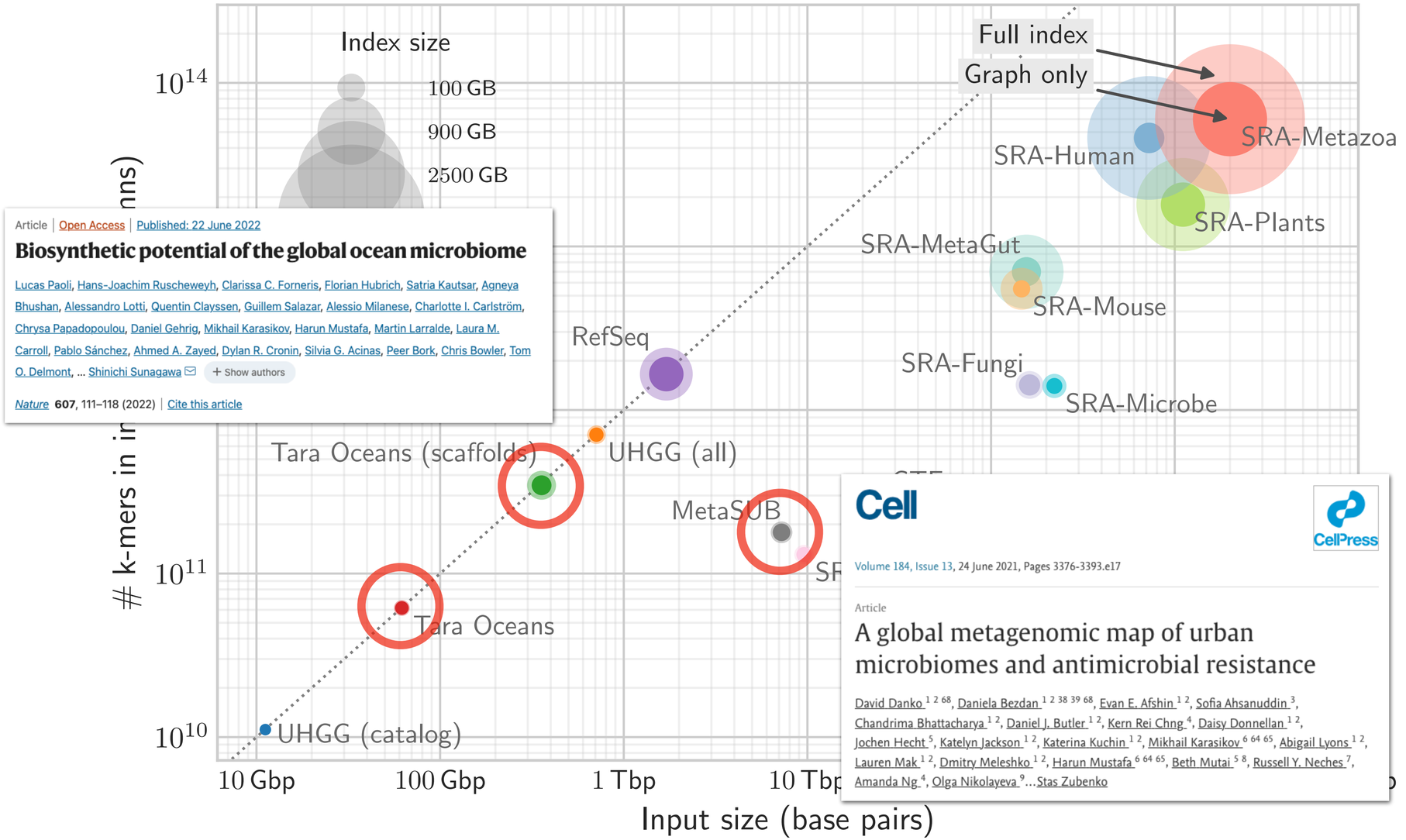
Datasets: UHGG (Almeida *et al.*, 2020), Tara Oceans (Paoli, 2022), GTEx (Lonsdale *et al.*, 2013), TCGA (Kaiser, 2005), MetaSUB (Danko *et al.*, 2019), SRA-Microbe (Bradley *et al.*, 2019)

Indexing at Petabase-scale

MetaGraph: Indexing and Analysing Nucleotide Archives at Petabase-scale

Mikhail Karasikov, Harun Mustafa, Daniel Danciu, Marc Zimmermann, Christopher Barber, Gunnar Rätsch, André Kahles

doi: https://doi.org/10.1101/2020.10.01.322164



Article | Open Access | Published: 22 June 2022

Biosynthetic potential of the global ocean microbiome

Lucas Paoli, Hans-Joachim Ruscheweyh, Clarissa C. Forneris, Florian Hubrich, Satria Kautsar, Agneya Bhushan, Alessandro Lotti, Quentin Clayssen, Guillem Salazar, Alessio Milanese, Charlotte I. Carlström, Chrysa Papadopoulou, Daniel Gehrig, Mikhail Karasikov, Harun Mustafa, Martin Larralde, Laura M. Carroll, Pablo Sánchez, Ahmed A. Zayed, Dylan R. Cronin, Silvia G. Acinas, Peer Bork, Chris Bowler, Tom O. Delmont, ... Shinichi Sunagawa + Show authors

Nature **607**, 111–118 (2022) | Cite this article

Cell

CellPress

Volume 184, Issue 13, 24 June 2021, Pages 3376-3393.e17

Article

A global metagenomic map of urban microbiomes and antimicrobial resistance

David Danko [1 2 68], Daniela Bezdan [1 2 38 39 68], Evan E. Afshin [1 2], Sofia Ahsanuddin [3], Chandrima Bhattacharya [1 2], Daniel J. Butler [1 2], Kern Rei Chng [4], Daisy Donnellan [1 2], Jochen Hecht [5], Katelyn Jackson [1 2], Katerina Kuchin [1 2], Mikhail Karasikov [6 64 65], Abigail Lyons [1 2], Lauren Mak [1 2], Dmitry Meleshko [1 2], Harun Mustafa [6 64 65], Beth Mutai [5 8], Russell Y. Neches [7], Amanda Ng [4], Olga Nikolayeva [9]...Stas Zubenko

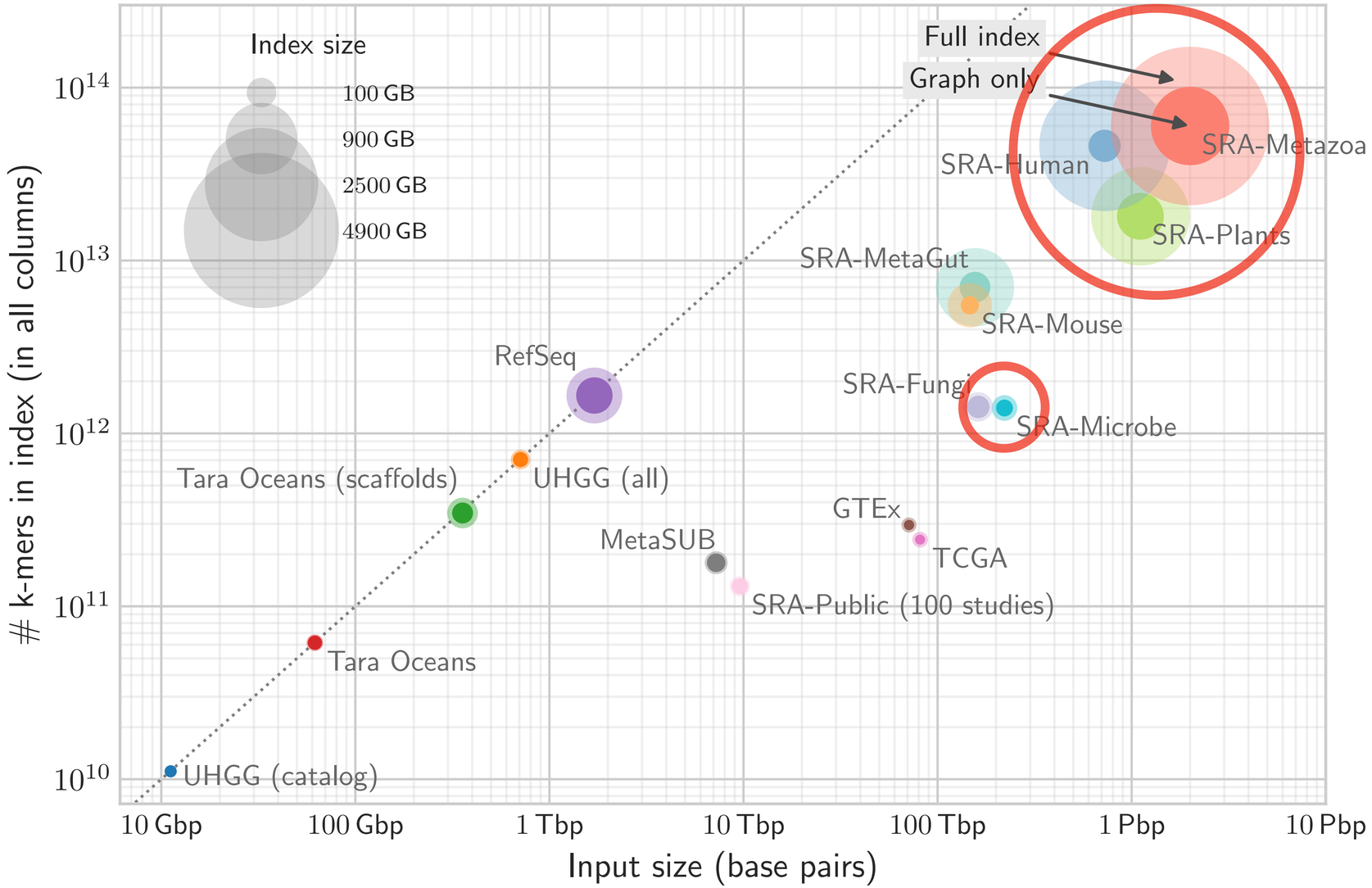
Datasets: UHGG (Almeida *et al.*, 2020), Tara Oceans (Paoli, 2022), GTEx (Lonsdale *et al.*, 2013), TCGA (Kaiser, 2005), MetaSUB (Danko *et al.*, 2019), SRA-Microbe (Bradley *et al.*, 2019)

Indexing at Petabase-scale

MetaGraph: Indexing and Analysing Nucleotide Archives at Petabase-scale

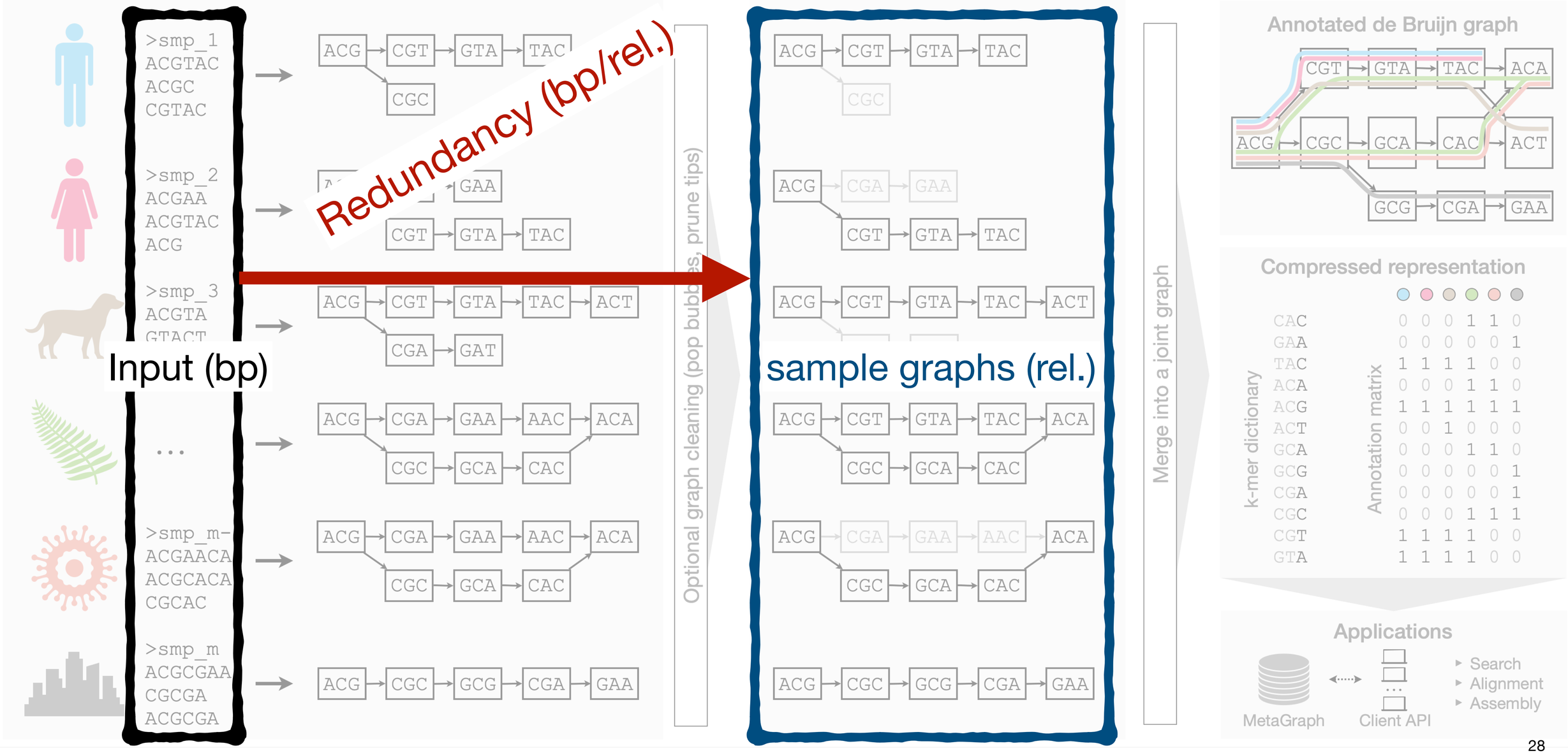
Mikhail Karasikov, Harun Mustafa, Daniel Danciu, Marc Zimmermann, Christopher Barber, Gunnar Rätsch, André Kahles

doi: https://doi.org/10.1101/2020.10.01.322164

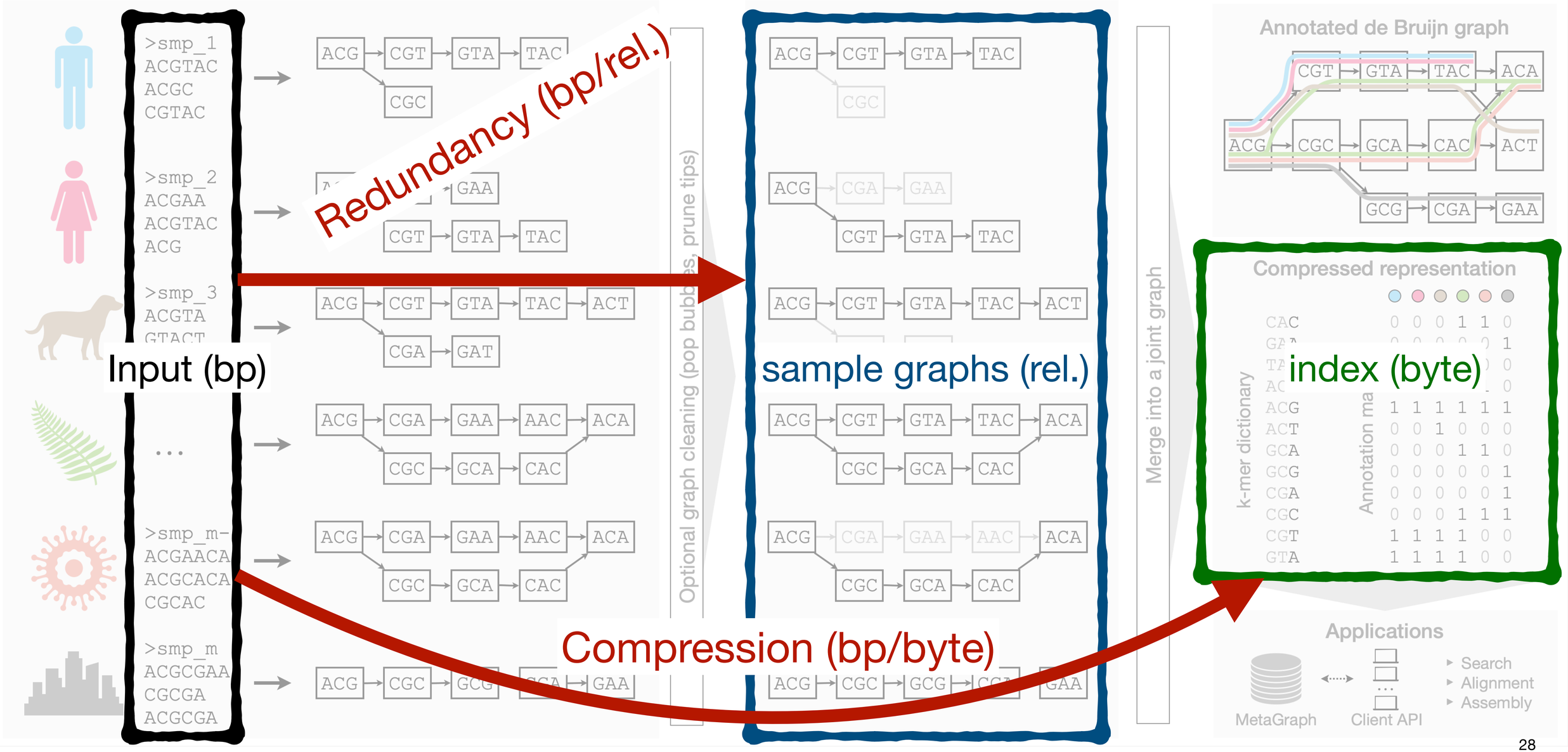


Datasets: UHGG (Almeida *et al.*, 2020), Tara Oceans (Paoli, 2022), GTEx (Lonsdale *et al.*, 2013), TCGA (Kaiser, 2005), MetaSUB (Danko *et al.*, 2019), SRA-Microbe (Bradley *et al.*, 2019)

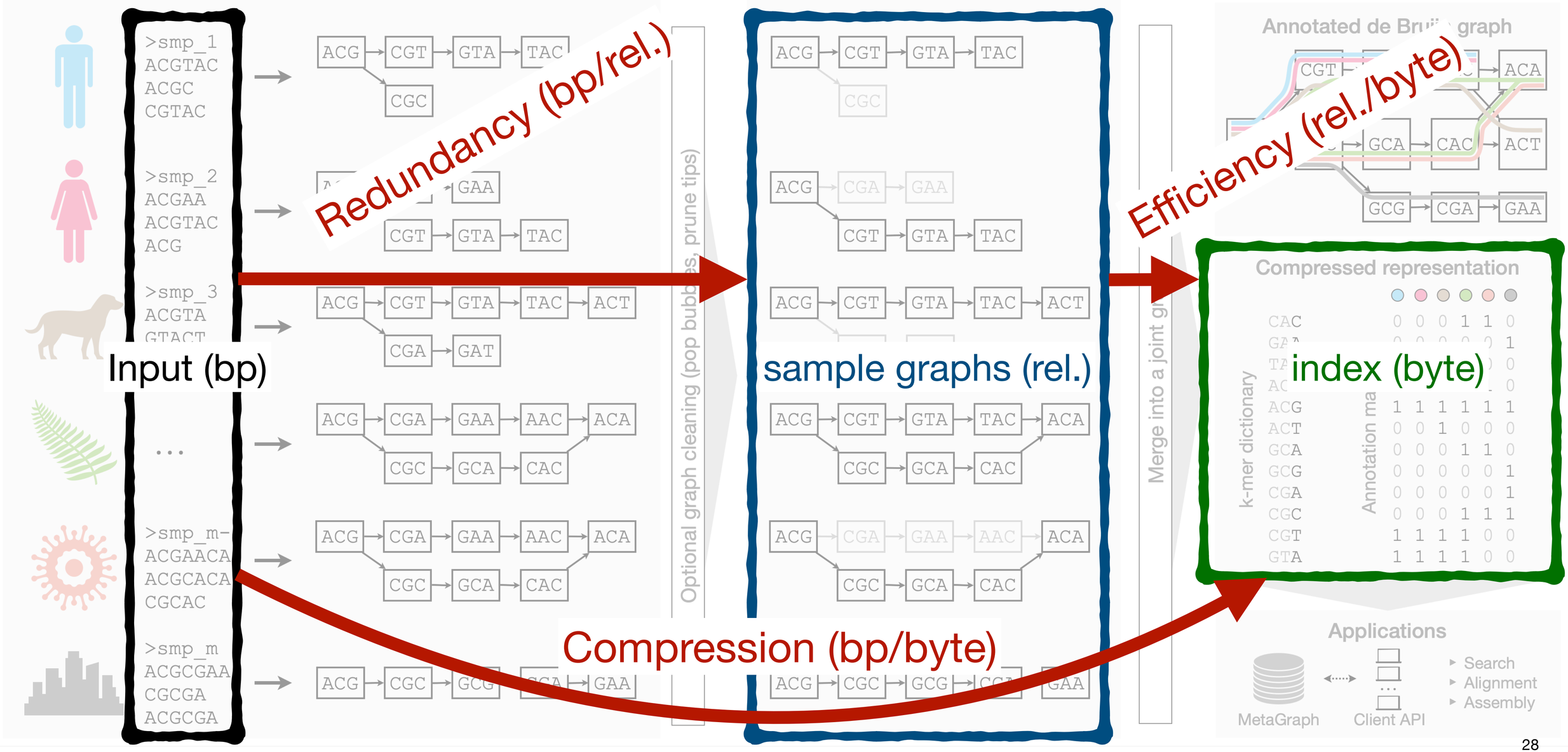
Compression, redundancy, efficiency



Compression, redundancy, efficiency



Compression, redundancy, efficiency



Indexing petabase-scale inputs

Type	Data set	Tbp	# labels	Index size	Compression (bp/byte)	Redund. (bp/rel.)	Efficiency (rel./byte)
Assembled seq.	UHGG (catalog)	0.01	4,644	3.2 GB	3.5	1.0	3.5
	UHGG (all)	0.71	286,997	27.3 GB	26.0	1.0	25.9
	Tara Oceans (scaffolds)	0.36	318,205,057	110.2 GB	3.2	1.0	3.1
	Tara Oceans with coord.†	0.06	34,815	14.6 GB	4.2	1.0	4.2
	RefSeq with coordinates†	1.70	85,375	508.9 GB	3.3	1.0	3.3

Datasets: UHGG (Almeida *et al.*, 2020), Tara Oceans (Paoli, 2022), GTEx (Lonsdale *et al.*, 2013), TCGA (Kaiser, 2005), MetaSUB (Danko *et al.*, 2019), SRA-Microbe (Bradley *et al.*, 2019)

Indexing petabase-scale inputs

Type	Data set	Tbp	# labels	Index size	Compression (bp/byte)	Redund. (bp/rel.)	Efficiency (rel./byte)
Assembled seq.	UHGG (catalog)	0.01	4,644	3.2 GB	3.5	1.0	3.5
	UHGG (all)	0.71	286,997	27.3 GB	26.0	1.0	25.9
	Tara Oceans (scaffolds)	0.36	318,205,057	110.2 GB	3.2	1.0	3.1
	Tara Oceans with coord.†	0.06	34,815	14.6 GB	4.2	1.0	4.2
	RefSeq with coordinates†	1.70	85,375	508.9 GB	3.3	1.0	3.3
RNA-Seq	GTEx	71.2	9,759	9.6 GB	7,416	241.2	30.7
	GTEx with counts‡	71.2	9,759	76.3 GB	934	241.2	3.9
	TCGA	81.2	11,095	11.1 GB	7,288	334.4	21.8
	TCGA with counts‡	81.2	11,095	81.2 GB	1,001	320.4	3.1

Datasets: UHGG (Almeida *et al.*, 2020), Tara Oceans (Paoli, 2022), GTEx (Lonsdale *et al.*, 2013), TCGA (Kaiser, 2005), MetaSUB (Danko *et al.*, 2019), SRA-Microbe (Bradley *et al.*, 2019)

Indexing petabase-scale inputs

Type	Data set	Tbp	# labels	Index size	Compression (bp/byte)	Redund. (bp/rel.)	Efficiency (rel./byte)
Assembled seq.	UHGG (catalog)	0.01	4,644	3.2 GB	3.5	1.0	3.5
	UHGG (all)	0.71	286,997	27.3 GB	26.0	1.0	25.9
	Tara Oceans (scaffolds)	0.36	318,205,057	110.2 GB	3.2	1.0	3.1
	Tara Oceans with coord.†	0.06	34,815	14.6 GB	4.2	1.0	4.2
	RefSeq with coordinates†	1.70	85,375	508.9 GB	3.3	1.0	3.3
RNA-Seq	GTEx	71.2	9,759	9.6 GB	7,416	241.2	30.7
	GTEx with counts‡	71.2	9,759	76.3 GB	934	241.2	3.9
	TCGA	81.2	11,095	11.1 GB	7,288	334.4	21.8
	TCGA with counts‡	81.2	11,095	81.2 GB	1,001	320.4	3.1
MGS	MetaSUB	7.2	4,220	46.7 GB	155	40.5	3.8
	SRA-MetaGut	155.8	241,384	1,111.3 GB	140	22.2	6.3

Datasets: UHGG (Almeida *et al.*, 2020), Tara Oceans (Paoli, 2022), GTEx (Lonsdale *et al.*, 2013), TCGA (Kaiser, 2005), MetaSUB (Danko *et al.*, 2019), SRA-Microbe (Bradley *et al.*, 2019)

Indexing petabase-scale inputs

Type	Data set	Tbp	# labels	Index size	Compression (bp/byte)	Redund. (bp/rel.)	Efficiency (rel./byte)
Assembled seq.	UHGG (catalog)	0.01	4,644	3.2 GB	3.5	1.0	3.5
	UHGG (all)	0.71	286,997	27.3 GB	26.0	1.0	25.9
	Tara Oceans (scaffolds)	0.36	318,205,057	110.2 GB	3.2	1.0	3.1
	Tara Oceans with coord.†	0.06	34,815	14.6 GB	4.2	1.0	4.2
	RefSeq with coordinates†	1.70	85,375	508.9 GB	3.3	1.0	3.3
RNA-Seq	GTEx	71.2	9,759	9.6 GB	7,416	241.2	30.7
	GTEx with counts‡	71.2	9,759	76.3 GB	934	241.2	3.9
	TCGA	81.2	11,095	11.1 GB	7,288	334.4	21.8
	TCGA with counts‡	81.2	11,095	81.2 GB	1,001	320.4	3.1
MGS	MetaSUB	7.2	4,220	46.7 GB	155	40.5	3.8
	SRA-MetaGut	155.8	241,384	1,111.3 GB	140	22.2	6.3
SRA subsets	SRA-Microbe	221.1	446,506	65.5 GB	3,376	157.6	21.4
	SRA-Fungi	162.1	121,900	98.3 GB	1,649	113.9	14.5
	SRA-Plants	1,109.2	531,714	1,844.1 GB	602	61.5	9.8
	SRA-Human	725.4	436,494	3,402.1 GB	213	15.7	13.5
	SRA-Metazoa (Mouse)	146.6	57,938	291.6 GB	503	26.6	18.9
	SRA-Metazoa (1k studies)	118.8	67,391	302.7 GB	393	33.5	11.7
	SRA-Metazoa	1,999.5	805,239	5.1 TB*	390*	33.3	11.7*

Datasets: UHGG (Almeida *et al.*, 2020), Tara Oceans (Paoli, 2022), GTEx (Lonsdale *et al.*, 2013), TCGA (Kaiser, 2005), MetaSUB (Danko *et al.*, 2019), SRA-Microbe (Bradley *et al.*, 2019)

Indexing petabase-scale inputs

Type	Data set	Tbp	# labels	Index size	Compression (bp/byte)	Redund. (bp/rel.)	Efficiency (rel./byte)
Assembled seq.	UHGG (catalog)	0.01	4,644	3.2 GB	3.5	1.0	3.5
	UHGG (all)	0.71	286,997	27.3 GB	26.0	1.0	25.9
	Tara Oceans (scaffolds)	0.36	318,205,057	110.2 GB	3.2	1.0	3.1
	Tara Oceans with coord.†	0.06	34,815	14.6 GB	4.2	1.0	4.2
	RefSeq with coordinates†	1.70	85,375	508.9 GB	3.3	1.0	3.3
RNA-Seq	GTEx	71.2	9,759	9.6 GB	7,416	241.2	30.7
	GTEx with counts‡	71.2	9,759	76.3 GB	934	241.2	3.9
	TCGA	81.2	11,095	11.1 GB	7,288	334.4	21.8
	TCGA with counts‡	81.2	11,095	81.2 GB	1,001	320.4	3.1
MGS	MetaSUB	7.2	4,220	46.7 GB	155	40.5	3.8
	SRA-MetaGut	155.8	241,384	1,111.3 GB	140	22.2	6.3
SRA subsets	SRA-Microbe	221.1	446,506	65.5 GB	3,376	157.6	21.4
	SRA-Fungi	162.1	121,900	98.3 GB	1,649	113.9	14.5
	SRA-Plants	1,109.2	531,714	1,844.1 GB	602	61.5	9.8
	SRA-Human	725.4	436,494	3,402.1 GB	213	15.7	13.5
	SRA-Metazoa (Mouse)	146.6	57,938	291.6 GB	503	26.6	18.9
	SRA-Metazoa (1k studies)	118.8	67,391	302.7 GB	393	33.5	11.7
	SRA-Metazoa	1,999.5	805,239	5.1 TB*	390*	33.3	11.7*
SRA	SRA-Public to 01.01.2023	38,949.9	23,010,648				

Datasets: UHGG (Almeida *et al.*, 2020), Tara Oceans (Paoli, 2022), GTEx (Lonsdale *et al.*, 2013), TCGA (Kaiser, 2005), MetaSUB (Danko *et al.*, 2019), SRA-Microbe (Bradley *et al.*, 2019)

Indexing petabase-scale inputs

Type	Data set	Tbp	# labels	Index size	Compression (bp/byte)	Redund. (bp/rel.)	Efficiency (rel./byte)
Assembled seq.	UHGG (catalog)	0.01	4,644	3.2 GB	3.5	1.0	3.5
	UHGG (all)	0.71	286,997	27.3 GB	26.0	1.0	25.9
	Tara Oceans (scaffolds)	0.36	318,205,057	110.2 GB	3.2	1.0	3.1
	Tara Oceans with coord.†	0.06	34,815	14.6 GB	4.2	1.0	4.2
	RefSeq with coordinates†	1.70	85,375	508.9 GB	3.3	1.0	3.3
RNA-Seq	GTEx	71.2	9,759	9.6 GB	7,416	241.2	30.7
	GTEx with counts‡	71.2	9,759	76.3 GB	934	241.2	3.9
	TCGA	81.2	11,095	11.1 GB	7,288	334.4	21.8
	TCGA with counts‡	81.2	11,095	81.2 GB	1,001	320.4	3.1
MGS	MetaSUB	7.2	4,220	46.7 GB	155	40.5	3.8
	SRA-MetaGut	155.8	241,384	1,111.3 GB	140	22.2	6.3
SRA subsets	SRA-Microbe	221.1	446,506	65.5 GB	3,376	157.6	21.4
	SRA-Fungi	162.1	121,900	98.3 GB	1,649	113.9	14.5
	SRA-Plants	1,109.2	531,714	1,844.1 GB	602	61.5	9.8
	SRA-Human	725.4	436,494	3,402.1 GB	213	15.7	13.5
	SRA-Metazoa (Mouse)	146.6	57,938	291.6 GB	503	26.6	18.9
	SRA-Metazoa (1k studies)	118.8	67,391	302.7 GB	393	33.5	11.7
	SRA-Metazoa	1,999.5	805,239	5.1 TB*	390*	33.3	11.7*
SRA	SRA-Public (100 studies)	9.6	5,184	32.0 GB	300	73.3	4.1
	SRA-Public to 01.01.2023	38,949.9	23,010,648				

Datasets: UHGG (Almeida *et al.*, 2020), Tara Oceans (Paoli, 2022), GTEx (Lonsdale *et al.*, 2013), TCGA (Kaiser, 2005), MetaSUB (Danko *et al.*, 2019), SRA-Microbe (Bradley *et al.*, 2019)

Indexing petabase-scale inputs

Type	Data set	Tbp	# labels	Index size	Compression (bp/byte)	Redund. (bp/rel.)	Efficiency (rel./byte)
Assembled seq.	UHGG (catalog)	0.01	4,644	3.2 GB	3.5	1.0	3.5
	UHGG (all)	0.71	286,997	27.3 GB	26.0	1.0	25.9
	Tara Oceans (scaffolds)	0.36	318,205,057	110.2 GB	3.2	1.0	3.1
	Tara Oceans with coord.†	0.06	34,815	14.6 GB	4.2	1.0	4.2
	RefSeq with coordinates†	1.70	85,375	508.9 GB	3.3	1.0	3.3
RNA-Seq	GTEx	71.2	9,759	9.6 GB	7,416	241.2	30.7
	GTEx with counts‡	71.2	9,759	76.3 GB	934	241.2	3.9
	TCGA	81.2	11,095	11.1 GB	7,288	334.4	21.8
	TCGA with counts‡	81.2	11,095	81.2 GB	1,001	320.4	3.1
MGS	MetaSUB	7.2	4,220	46.7 GB	155	40.5	3.8
	SRA-MetaGut	155.8	241,384	1,111.3 GB	140	22.2	6.3
SRA subsets	SRA-Microbe	221.1	446,506	65.5 GB	3,376	157.6	21.4
	SRA-Fungi	162.1	121,900	98.3 GB	1,649	113.9	14.5
	SRA-Plants	1,109.2	531,714	1,844.1 GB	602	61.5	9.8
	SRA-Human	725.4	436,494	3,402.1 GB	213	15.7	13.5
	SRA-Metazoa (Mouse)	146.6	57,938	291.6 GB	503	26.6	18.9
	SRA-Metazoa (1k studies)	118.8	67,391	302.7 GB	393	33.5	11.7
	SRA-Metazoa	1,999.5	805,239	5.1 TB*	390*	33.3	11.7*
SRA	SRA-Public (100 studies)	9.6	5,184	32.0 GB	300	73.3	4.1
	SRA-Public to 01.01.2023	38,949.9	23,010,648	130 TB*	300*	73.3*	4.1*

Datasets: UHGG (Almeida *et al.*, 2020), Tara Oceans (Paoli, 2022), GTEx (Lonsdale *et al.*, 2013), TCGA (Kaiser, 2005), MetaSUB (Danko *et al.*, 2019), SRA-Microbe (Bradley *et al.*, 2019)

Search in the Sequence Read Archive

- Search with Serratus [Edgar *et al.*, 2022]
  - (viral pangenome search of 82 Mbp against 4M samples at 0.62 cents per sample)

Search in the Sequence Read Archive

- Search with Serratus [Edgar *et al.*, 2022]
  - (viral pangenome search of 82 Mbp against 4M samples at 0.62 cents per sample)
  - effective query cost for SRA-Public (23M samples): **$1,700** per **1 Mbp** of query

Search in the Sequence Read Archive

- Search with Serratus [Edgar *et al.*, 2022]
  - (viral pangenome search of 82 Mbp against 4M samples at 0.62 cents per sample)
  - effective query cost for SRA-Public (23M samples): **$1,700** per **1 Mbp** of query

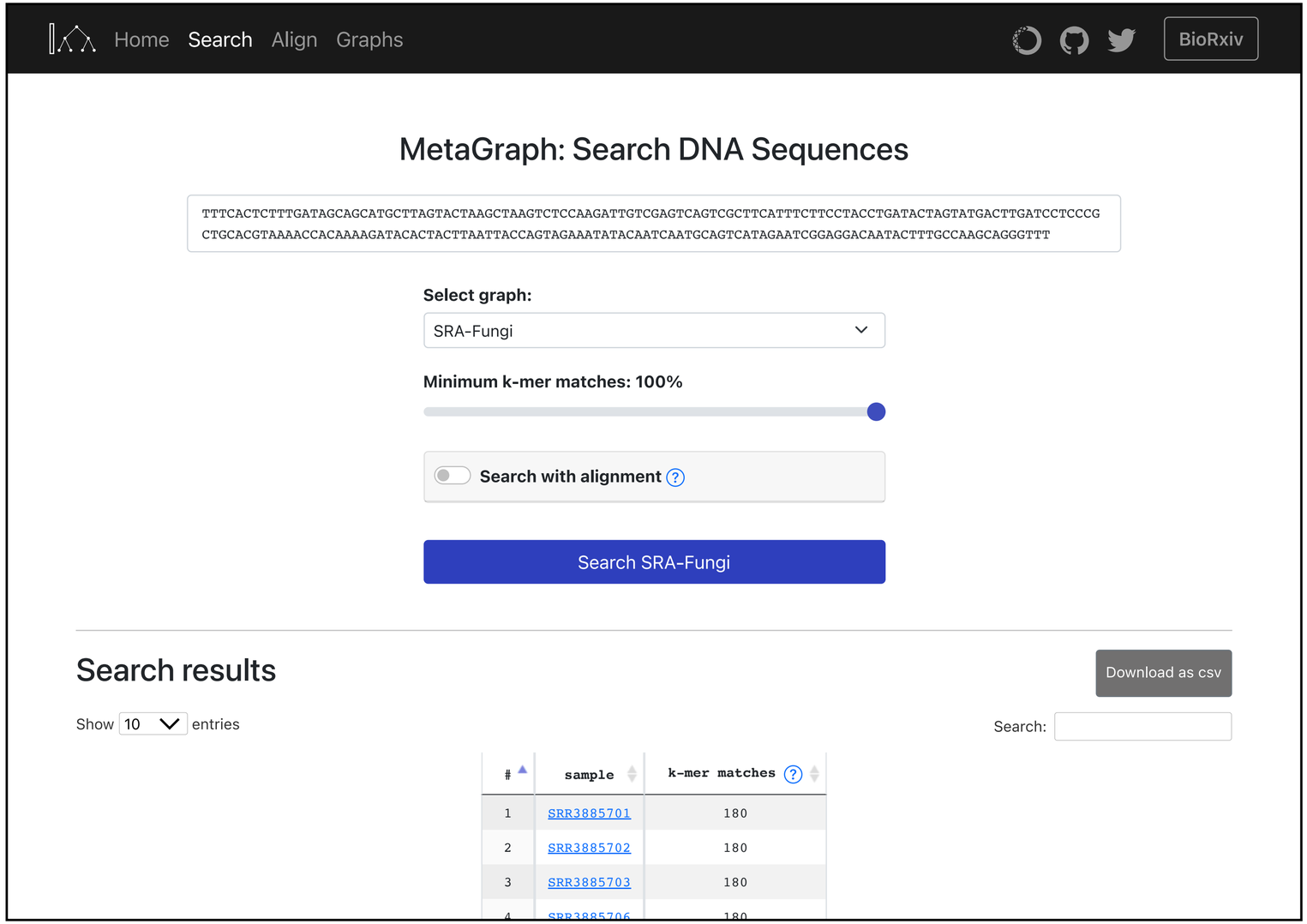
- Search with MetaGraph
  - (extrapolated from ‘SRA-Public (100 studies)’ with prices for Google Cloud)

Search in the Sequence Read Archive

- Search with Serratus [Edgar *et al.*, 2022]
  - (viral pangenome search of 82 Mbp against 4M samples at 0.62 cents per sample)
  - effective query cost for SRA-Public (23M samples): **$1,700** per **1 Mbp** of query

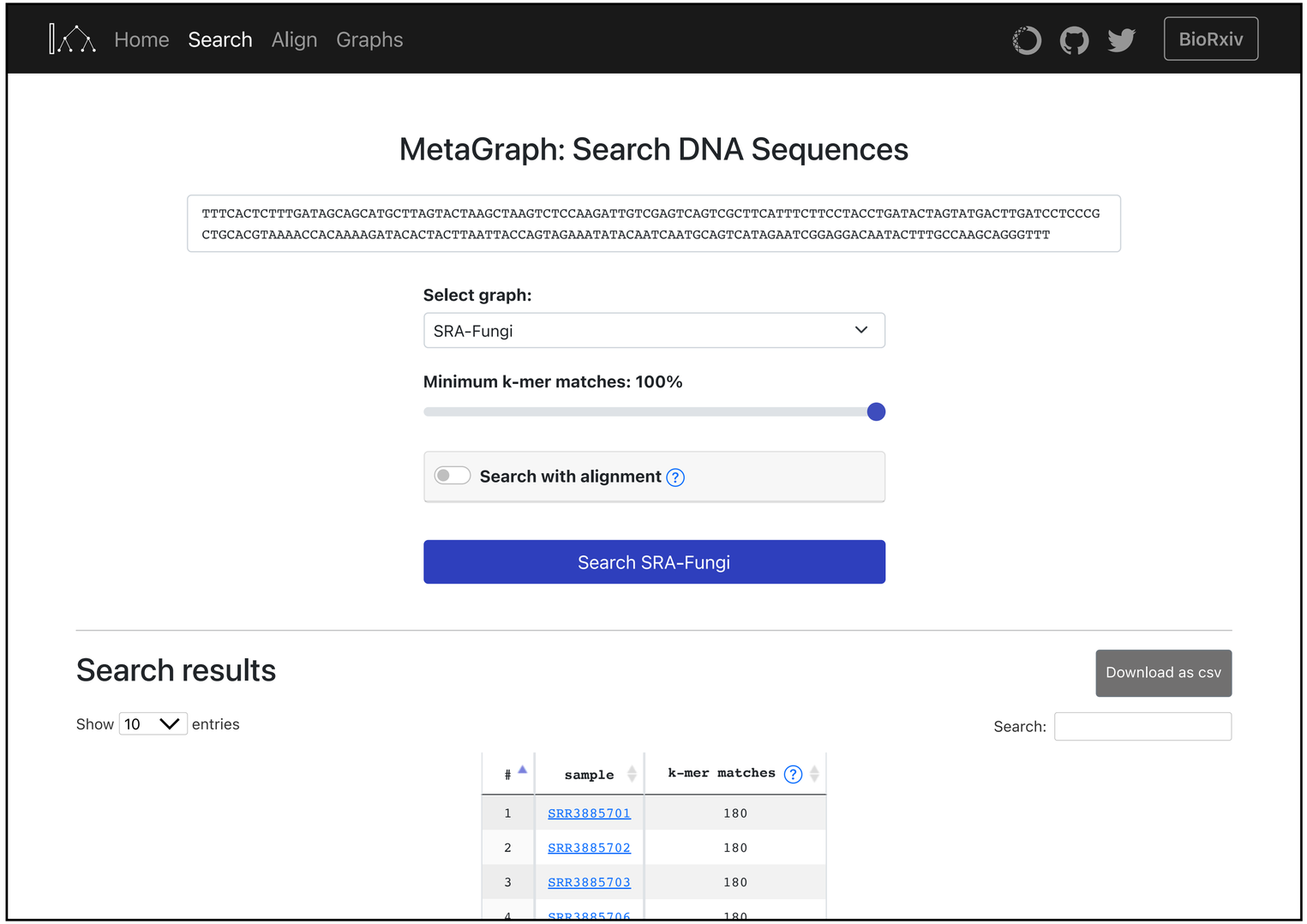
- Search with MetaGraph
  - (extrapolated from ‘SRA-Public (100 studies)’ with prices for Google Cloud)
  - effective query cost for SRA-Public (23M samples): **$2.8** per **1 Mbp** of query
  - **600x cheaper** than Serratus
  - indexing cost: only 2.2 cents per sample

MetaGraph Online



metagraph.ethz.ch/search

MetaGraph Online



metagraph.ethz.ch/search

Python Client API

```
In [1]: from metagraph.client import GraphClient

        SRV = "metagraph.ethz.ch"
        PORT = 12345
        g1 = GraphClient(SRV, PORT, api_path="/metasub")
        g2 = GraphClient(SRV, PORT, api_path="/refseq")
```

```
In [2]: query = "GGCTAACTACGTGCCAGCAGCCGCGGTAATAC"
        g1.search(query, align=True)
```

Out [2]:

	sample	sequence	score
0	SRR2201245	GGCTAACTACGTGCCAGCAGCCGCGGTAATAC	64
1	ERR1732568	GGCTAACTACGTGCCAGCAGCCGCGGTAATAC	64
2	ERR847096	GGCTAACTACGTGCCAGCAGCCGCGGTAATAC	64
...	...	...	...

Conclusion

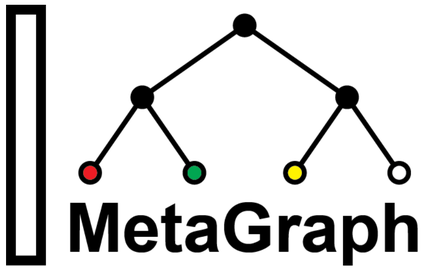
Main results of the thesis

- Showed the feasibility of indexing **all existing sequence archives**

Main results of the thesis

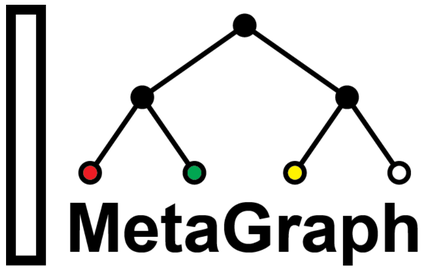
- Showed the feasibility of indexing **all existing sequence archives**
- Developed efficient methods for scalable **indexing** and **retrieval**
  - k-mer presence/absence
  - k-mer abundances
  - k-mer positions

Main results of the thesis



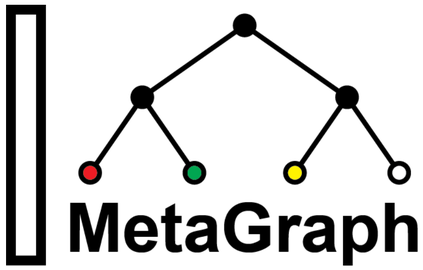
- Showed the feasibility of indexing **all existing sequence archives**
- Developed efficient methods for scalable **indexing** and **retrieval**
  - k-mer presence/absence
  - k-mer abundances
  - k-mer positions
- Efficient and scalable tool **MetaGraph**

Main results of the thesis



- Showed the feasibility of indexing **all existing sequence archives**
- Developed efficient methods for scalable **indexing** and **retrieval**
  - k-mer presence/absence
  - k-mer abundances
  - k-mer positions
- Efficient and scalable tool **MetaGraph**
- **Community resource indexes** of very large sequence data sets

Main results of the thesis



- Showed the feasibility of indexing **all existing sequence archives**
- Developed efficient methods for scalable **indexing** and **retrieval**
  - k-mer presence/absence
  - k-mer abundances
  - k-mer positions
- Efficient and scalable tool **MetaGraph**
- **Community resource indexes** of very large sequence data sets
- **MetaGraph Online** https://metagraph.ethz.ch

Publications

Primary works

- Mikhail Karasikov, Harun Mustafa, Daniel Danciu, Marc Zimmermann, Christopher Barber, Gunnar Rätsch, and André Kahles. "MetaGraph: Indexing and Analysing Nucleotide Archives at Petabase-scale". In: *bioRxiv* (2020). DOI: 10.1101/2020.10.01.322164
- Mikhail Karasikov, Harun Mustafa, Gunnar Rätsch, and André Kahles. "Lossless indexing with counting de Bruijn graphs". In: *Genome Research* 32.9 (2022), pp. 1754–1764. DOI: 10.1101/gr.276607.122
- Mikhail Karasikov, Harun Mustafa, Amir Joudaki, Sara Javadzadeh-No, Gunnar Rätsch, and André Kahles. "Sparse binary relation representations for genome graph annotation". In: *Journal of Computational Biology* 27.4 (2020), pp. 626–639. DOI: 10.1089/cmb.2019.0324
- Daniel Danciu, Mikhail Karasikov, Harun Mustafa, André Kahles, and Gunnar Rätsch. "Topology-based sparsification of graph annotations". In: *Bioinformatics* 37.Supplement_1 (July 2021), pp. i169–i176. DOI: 10.1093/bioinformatics/btab330

Significant contributions

- David Danko et al. "A global metagenomic map of urban microbiomes and antimicrobial resistance". In: *Cell* 184.13 (2021), 3376–3393.e17. DOI: 10.1016/j.cell.2021.05.002
- Lucas Paoli, Hans-Joachim Ruscheweyh, Clarissa C. Forneris, Florian Hubrich, Satria Kautsar, Agneya Bhushan, Alessandro Lotti, Quentin Clayssen, Guillem Salazar, Alessio Milanese, Charlotte I. Carlström, Chrysa Papadopoulou, Daniel Gehrig, Mikhail Karasikov, Harun Mustafa, Martin Larralde, Laura M. Carroll, Pablo Sánchez, Ahmed A. Zayed, Dylan R. Cronin, Silvia G. Acinas, Peer Bork, Chris Bowler, Tom O. Delmont, Josep M. Gasol, Alvar D. Gossert, André Kahles, Matthew B. Sullivan, Patrick Wincker, Georg Zeller, Serina L. Robinson, Jörn Piel, and Shinichi Sunagawa. "Biosynthetic potential of the global ocean microbiome". In: *Nature* 607.7917 (2022), pp. 111–118. DOI: 10.1038/s41586-022-04862-3
- Harun Mustafa, Mikhail Karasikov, Gunnar Rätsch, and André Kahles. "MetaGraph-MLA: Label-guided alignment to variable-order De Bruijn graphs". In: *bioRxiv* (2022). DOI: 10.1101/2022.11.04.514718
- Harun Mustafa, Ingo Schilken, Mikhail Karasikov, Carsten Eickhoff, Gunnar Rätsch, André Kahles, and John Hancock. "Dynamic compression schemes for graph coloring". In: *Bioinformatics* (2018). DOI: 10.1093/bioinformatics/bty632
- Maciej Besta, Raghavendra Kanakagiri, Harun Mustafa, Mikhail Karasikov, Gunnar Rätsch, Torsten Hoefler, and Edgar Solomonik. "Communication-Efficient Jaccard similarity for High-Performance Distributed Genome Comparisons". In: *2020 IEEE International Parallel and Distributed Processing Symposium (IPDPS)*. Los Alamitos, CA, USA: IEEE Computer Society, 2020, pp. 1122–1132. DOI: 10.1109/IPDPS47924.2020.00118

Outreach

Conferences

Talks:

- RECOMB 2019 (Washington, DC, USA) — presented by HM
- ISMB/ECCB 2021, HiTSeq COSI (online)
- RECOMB 2022 (La Jolla, California, USA)
- IGGSy 2022 (Ascona, Switzerland)
- JOBIM 2022 (Rennes, France) — **invited talk**

Poster:

- Biological Data Science - CSHL Meeting 2022 (Cold Spring Harbor, New York, USA)

Other talks

- MLSS 2020 (online)
- Zurich Seminars in Bioinformatics 2022 (Zurich, Switzerland)

Collaborations

- Tara Oceans (Sunagawa lab)
- **MetaSUB consortia** (C. Mason)
- Differential assembly (M. Huber)

Possible future applications

- Detection/confirmation of
  - novel splice junctions
  - **trans-splice junctions**
  - genomic variants (incl. structural variants)
  - novel sequence elements
  - gene fusions
  - phages in metagenomes
  - novel functional elements in metagenomes
  - pathogen pervasiveness (e.g. SARS CoV-II)
- Differential assembly
- Joint variant calling
- Large association studies on sequence elements

Thanks to the team

@m_karasikov


Mikhail Karasikov


Marc Zimmermann


Andre Kahles


Harun Mustafa


Daniel Danciu


Gunnar Rätsch

NRP75 Team


Torsten Hoefler


Mario Stanke


Matthis Ebel


Giovanna Migliorelli

Further contributors

- **Marek Kokot**
- **Thomas Zhou**
- Chris Barber
- Radu Muntean
- Jan Studený
- Sara Javadzadeh-No
- Predrag Krnetic

75 NRP Big Data National Research Programme

Swiss National Science Foundation

ETH zürich