Exploiting Redundancy for Efficient Processing of RNA-seq Data

Hirak Sarkar

Paradigm of mapping in RNA-seq

- Alignment is heavy, *O(mn)*
- Is it absolutely required ?
- Where we don’t need alignment.
- How mapping can be effective.

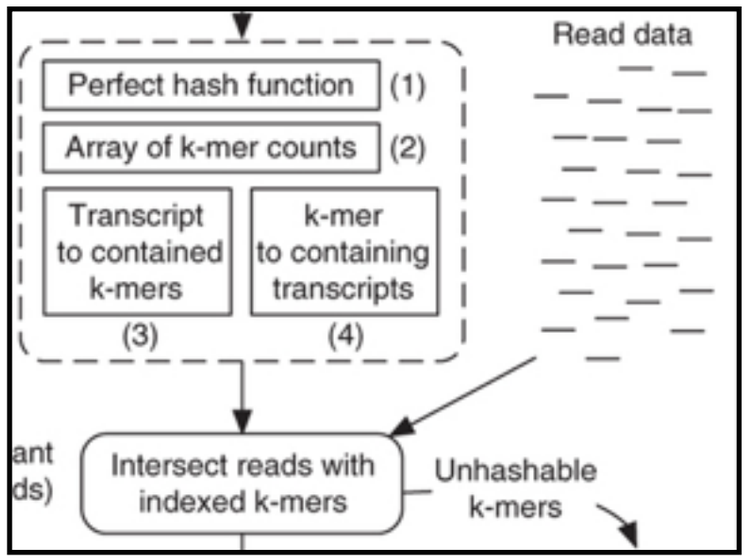
Fast map-based quantifiers

- Sailfish, Salmon and kallisto
- Make use of contiguous matches, and skip sequences of mismatch.
- Reports information about location and target transcript.
- Can simultaneously produce quantification result
- **Substantially** fast, and memory efficient.

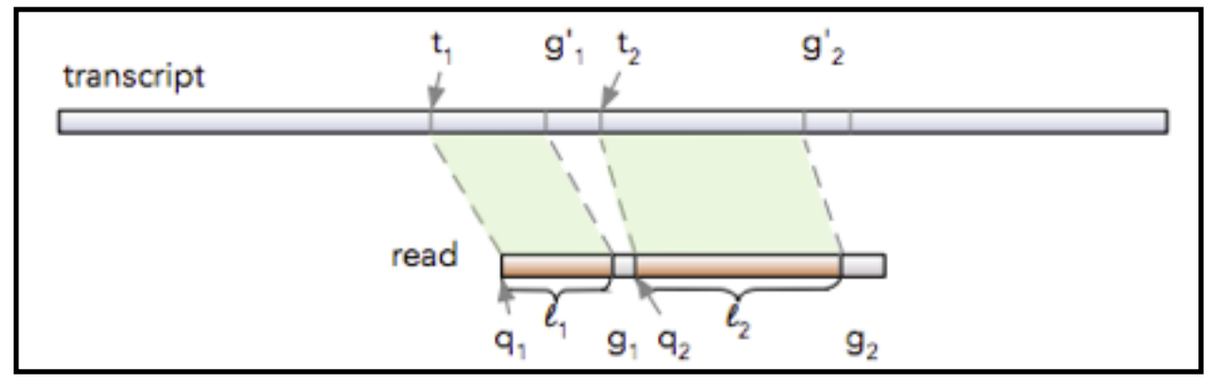
(826 min) → (54 min) → (4 min)

Express Sailfish Salmon/kallisto

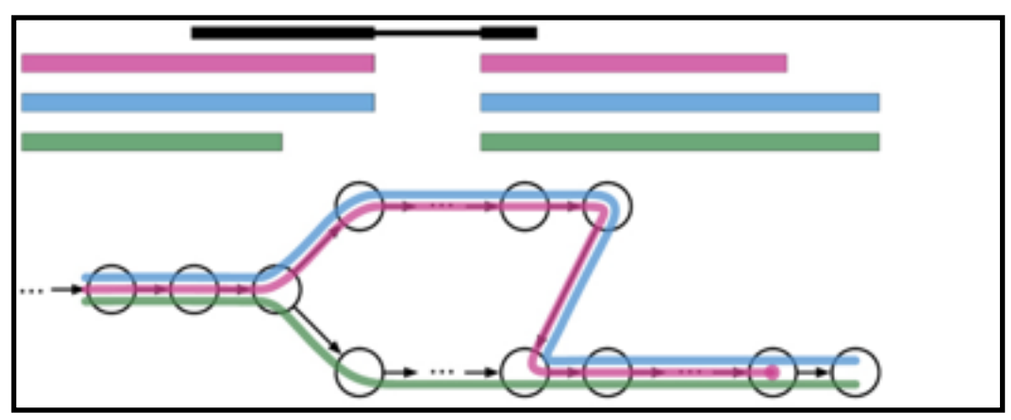
Paradigm of mapping



1



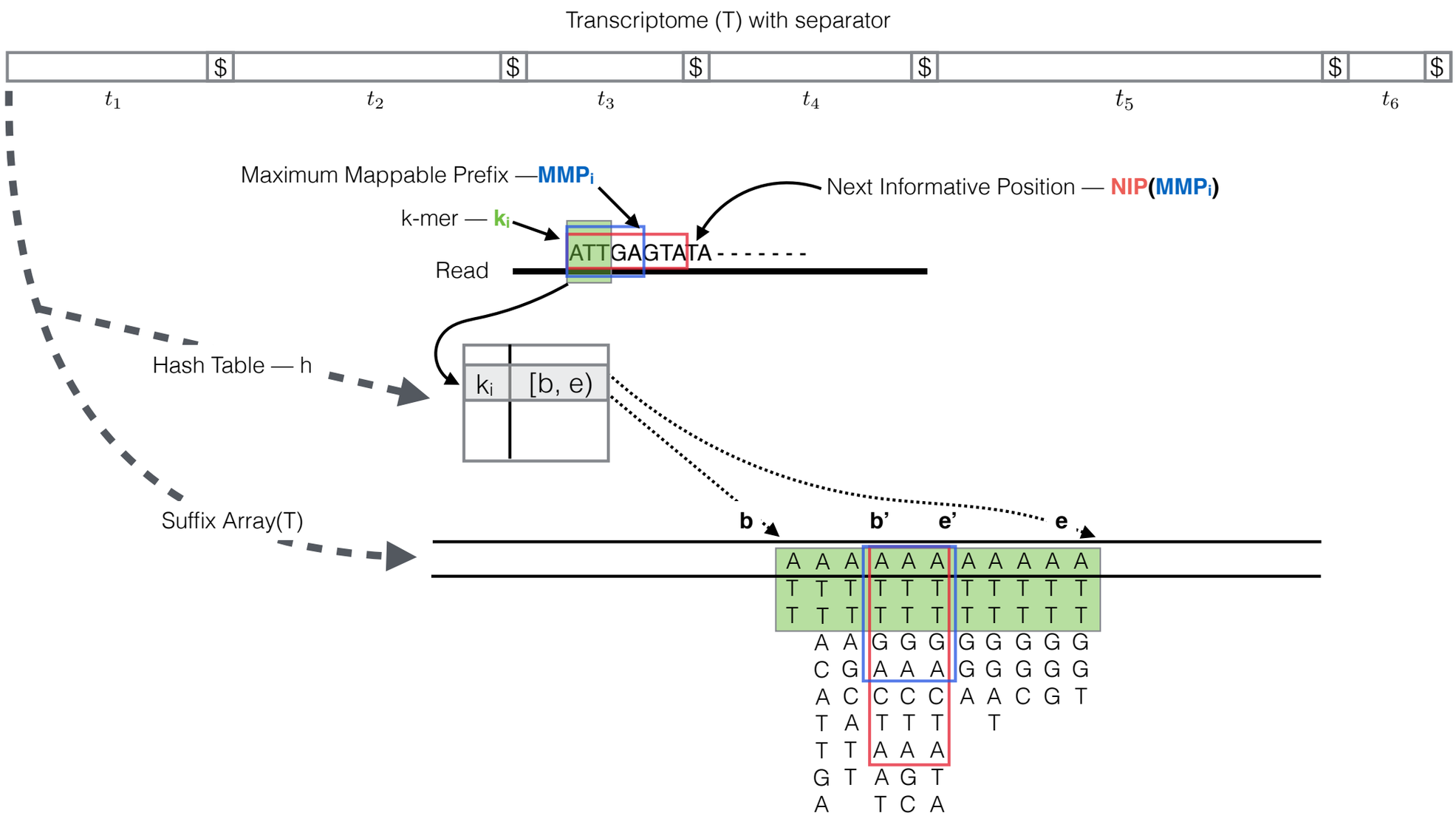
2



3

1. Snippet from **Sailfish** paper
2. Snippet from **Salmon** paper
3. Snippet from **kallisto** paper

Quasi-Mapping[1]



Transcriptome (T) with separator

Maximum Mappable Prefix — **MMP$_i$**

Next Informative Position — **NIP(MMP$_i$)**

k-mer — **k$_i$**

Read

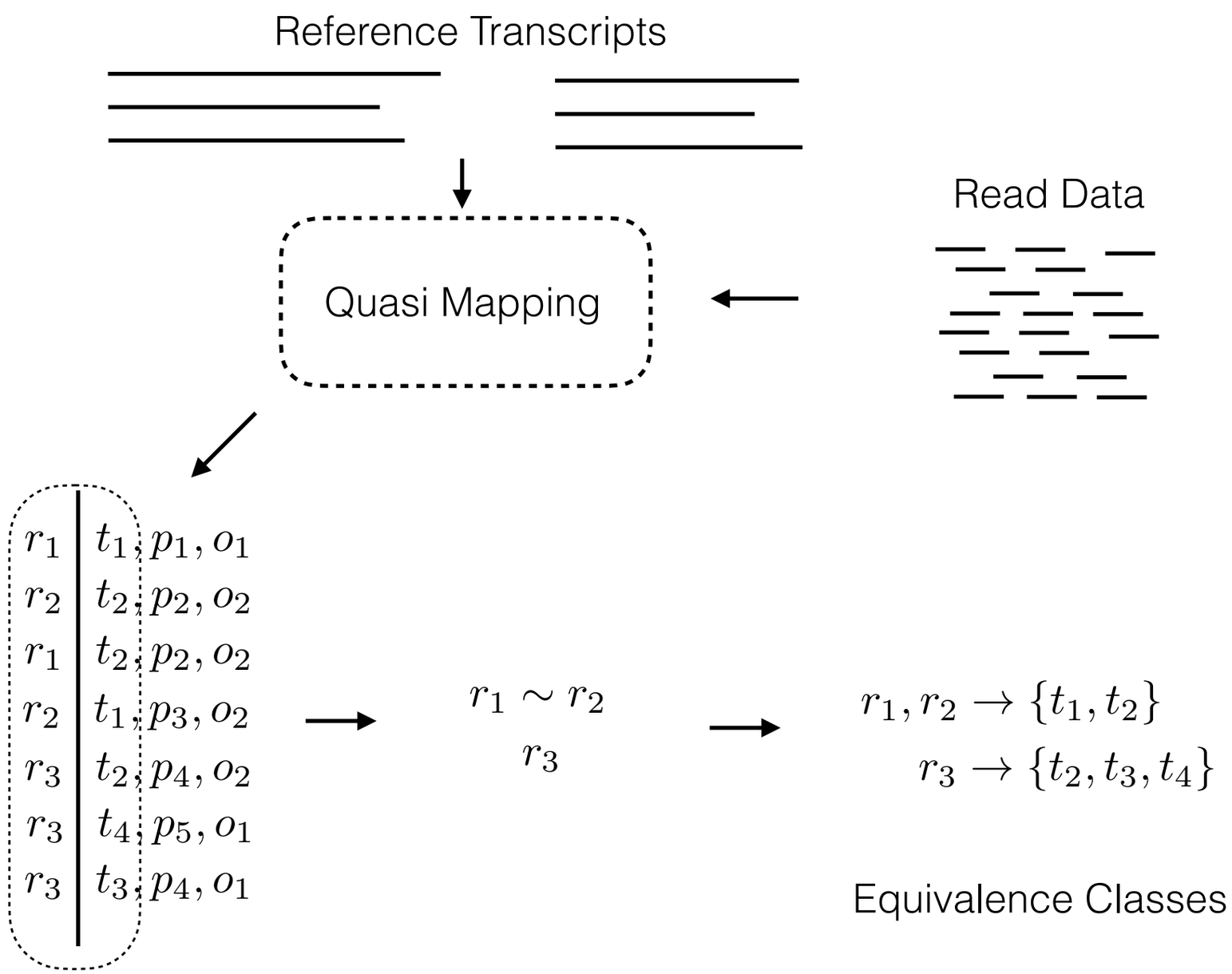
Hash Table — h

Suffix Array(T)

---

1. Adapted from **RapMap** paper

Derivative of Quasi-Mapping



Why clustering is important

- In *de novo* world clustering related contigs together is crucial for **gene-level** analysis

- Can be thought of as a two step process
  1. Grouping contigs into transcript
  2. Grouping transcripts into gene

- Benefits of **gene-level** estimate :
  1. Ease the differential expression step
  2. Robust estimation

- Caveat : Due to absence of reference annotation paralogous genes get co-clustered

Challenges of clustering

- *De novo* transcriptome assembly is inherently difficult problem.
- Despite of rapid improvements assemblers (TRINITY, CD-HIT) often fail to recover *full-length* transcripts.
- *fractured* transcripts can be generated from
  - Insufficient data
  - Erroneous sequence variation
  - Overlap threshold

Challenges of clustering

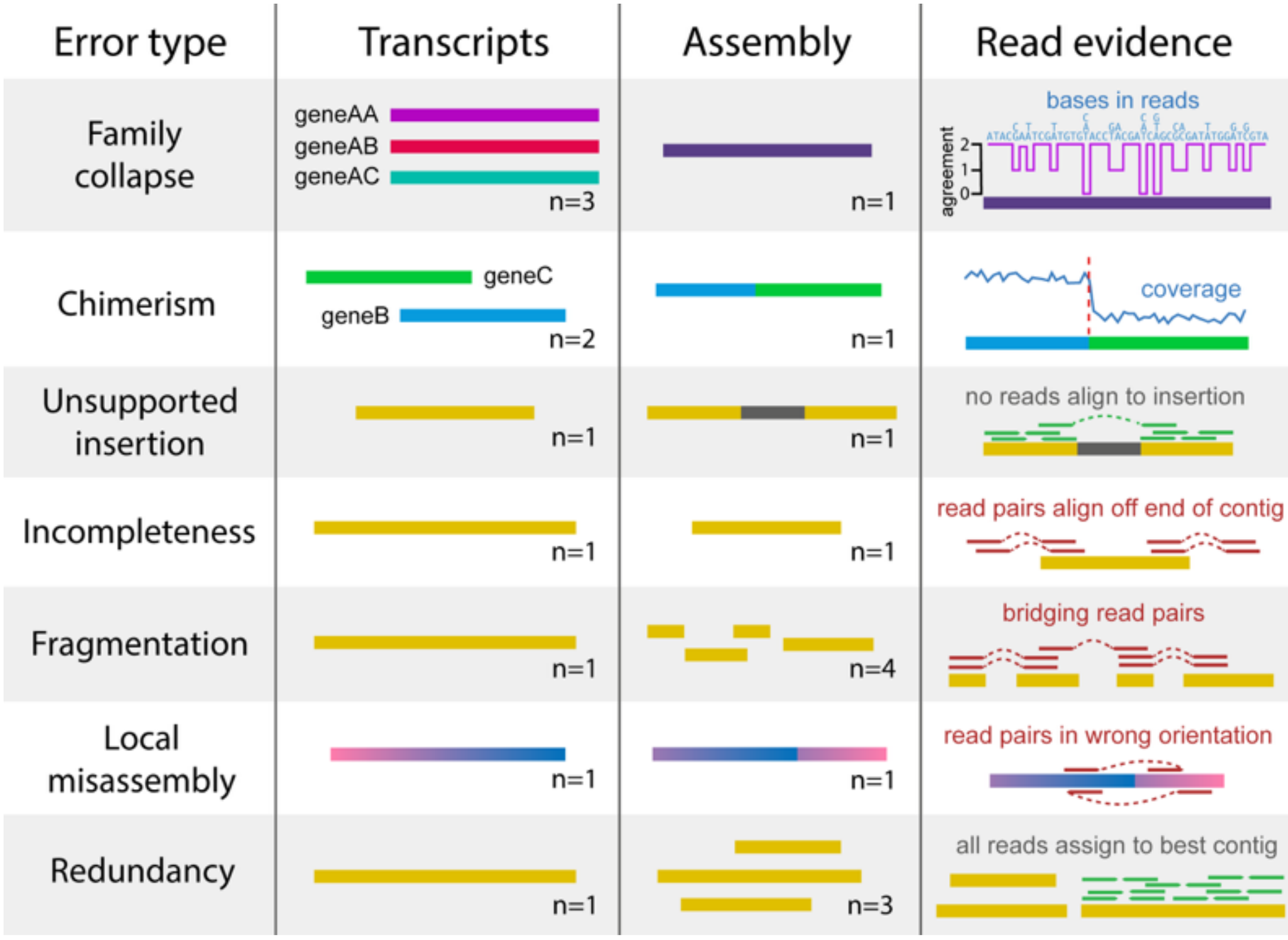


figure from: Richard D Smith-Unna, Chris Boursnell, Rob Patro, Julian M Hibberd, and Steven Kelly. Transrate: reference free quality assessment of de-novo transcriptome assemblies. *BioRxiv*, page 021626, 2015.

Challenges of clustering

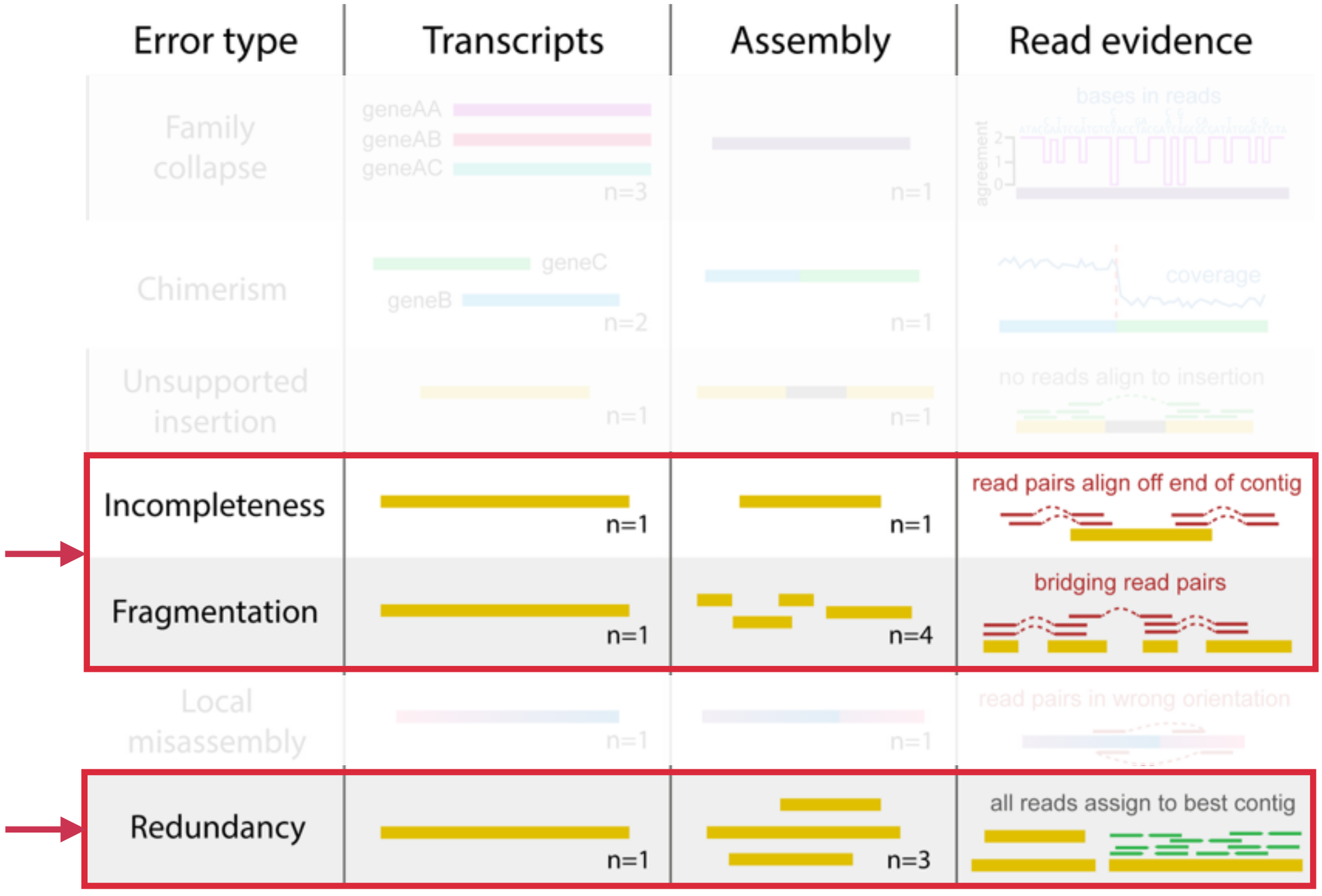


figure from: Richard D Smith-Unna, Chris Boursnell, Rob Patro, Julian M Hibberd, and Steven Kelly. Transrate: reference free quality assessment of de-novo transcriptome assemblies. *BioRxiv*, page 021626, 2015.

Ways of *de novo* clustering

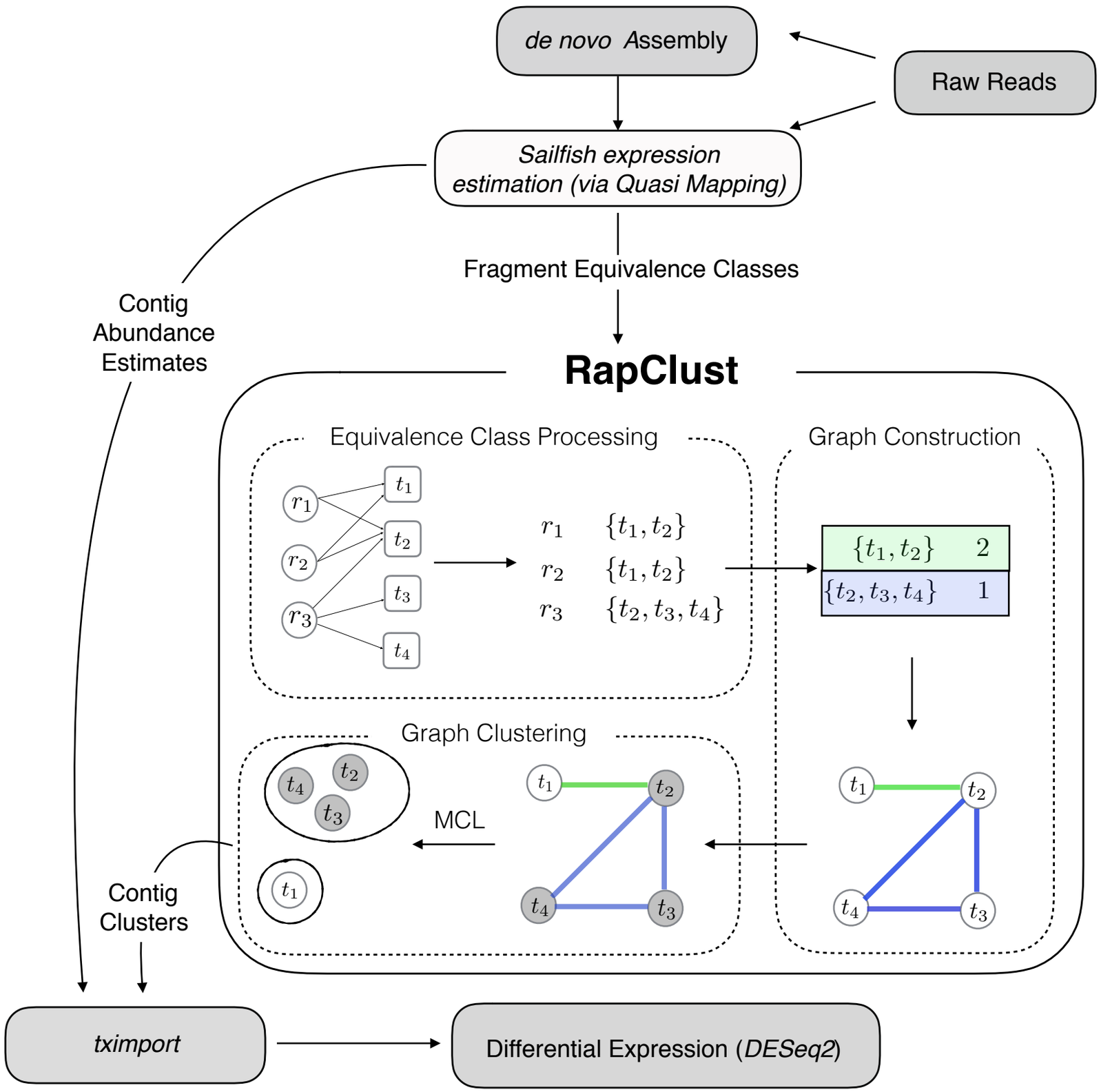
- Search for *similar sequences.* (CD-HIT EST)
- Look for *common subgraphs* during assembly
- Cluster contigs that *share reads* and have *similar expression*

We follow a principle similar to CORSET

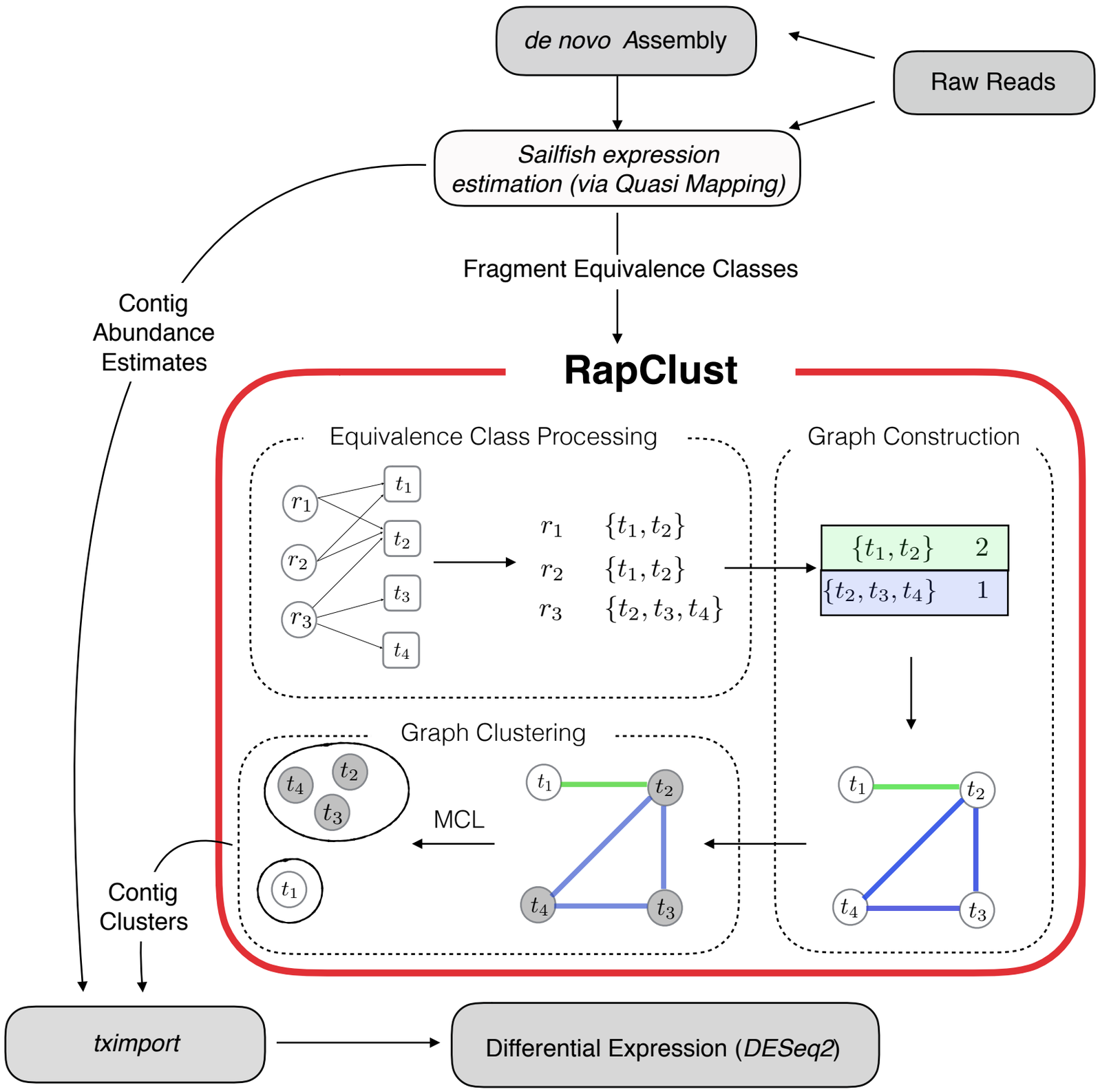
- We re-cast the clustering problem in the context graph (referred as *ambiguity graph*)

- We used contig-level expression value and along with graphs

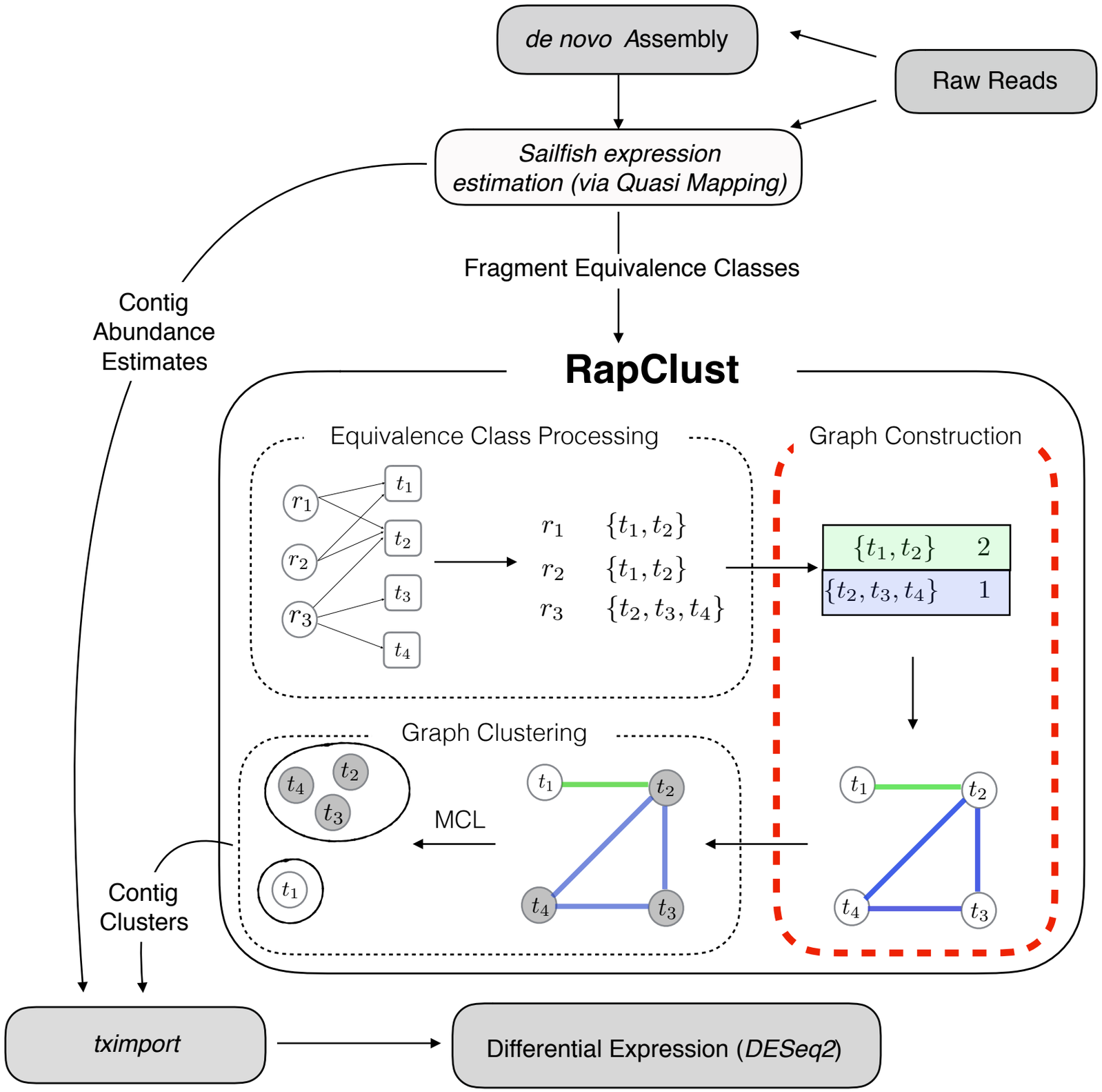
RapClust Pipeline



RapClust Pipeline



RapClust: Graph Construction



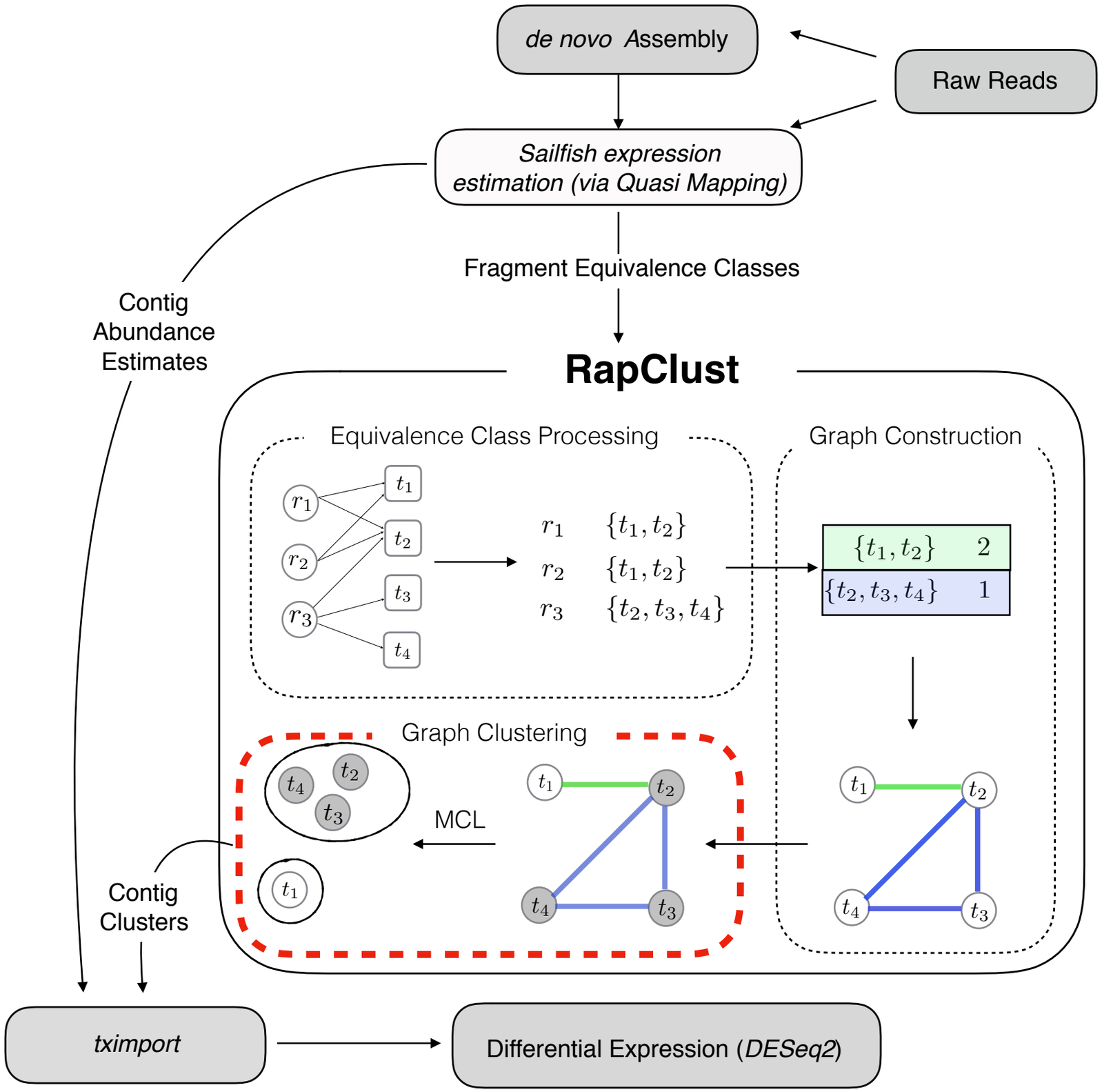
RapClust: Graph Construction

Given: A collection of fragment/read equivalence class, defined on set of reads $\{r_i\}$ and set of contigs $\{t_k\}$.

Compute: A weighted, undirected graph G=(V,E,w) where,

- V: The set of all contigs with more than one read mapped to them, a subset of $\{t_k\}$
- E: There exits an edge (t_i, t_j) between a pair of contig that co-occur in the same equivalence class
- w: The weight of an edge is proportional to the fraction of multi-mapping reads between these contigs.

RapClust: Graph Clustering



RapClust: Graph Clustering

Given: The graph G

Compute: A set of partition on the nodes of G, i.e. clustering of vertices.

- We use off-the-shelf graph clustering tool MCL to achieve the clustering.

- Other techniques like label propagation can also be used in order to get clusters

Speed & accuracy of RapClust

Dataset

	Yeast	Human	Chicken
# contigs	7353	107,389	335,377
# samples	6	6	8
Total (paired-end) reads	~36,000,000	~116,000,000	~181,402,780
Avg # eq. classes (across samples)	5197	100,535	222,216
# edges in mapping ambiguity graph	6195	212,481	2,063,524

We use the same data set from the *Corset* paper, which uses *Trinity** assembly

These assemblies are from organism with available reference genome which eases the evaluation of method.

Speed & accuracy of RapClust

	Yeast		Human		Chicken	
	RapClust	Corset	RapClust	Corset	RapClust	Corset
Time(min)	5.12	37.25	22.67	211.67	64.18	453
Space(Gb)	0.005	5.7	0.092	22	0.49	145
% of reads	88.17	62.32	93.04	77.94	88.80	60.99

- RapClust & Corset both perform clustering **and** yield expression estimates. CD-Hit does not, and so is evaluated separately

  RapClust:
  - Is much faster than Corset
  - The quantifier being used (Sailfish with quasi-mapping) maps substantially more reads
  - Requires much less intermediate space (e.g. no BAM files)

Wall-clock time reported using 4 threads on a 24-core (2.6GHz Xeon) machine with 256GB of RAM.

RapClust: Clustering

	Yeast			Human			Chicken		
	RC	CD	CT	RC	CD	CT	RC	CD	CT
Time(min)	0.04	0.2	2.8	0.82	4.02	16.25	5.29	36.5	87

- If we consider only *clustering* we can include CD-HIT.

- The clustering step of RapClust which consists of graph construction and running MCL, is much faster than other two methods.

Assessing Cluster Quality

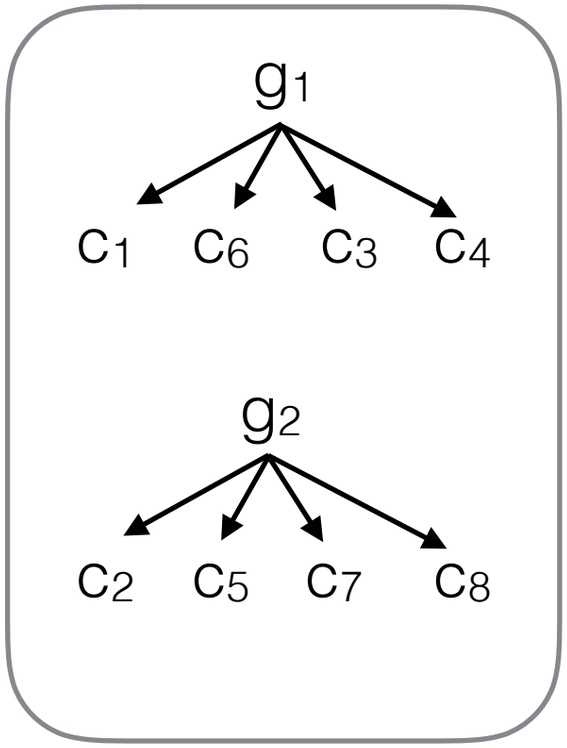
Given two contigs, we label them:

- *TP*: *co-clustered* and labeled with *same* gene
- *FN*: *not* co-clustered and labeled with *same* gene
- *FP*: *co-clustered* and labeled with *different* genes
- *TN*: *not* co-clustered and labeled with *different* gene

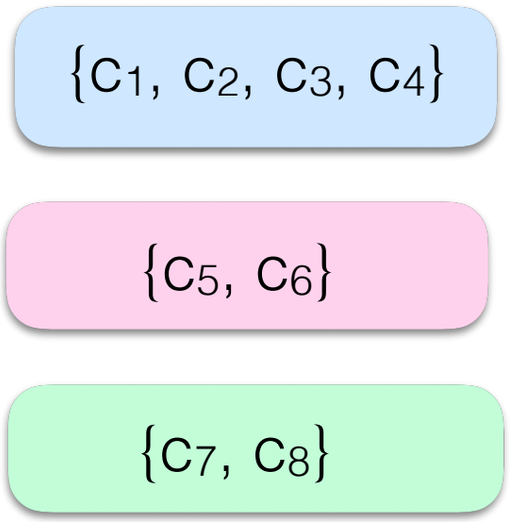
$$Precision = \frac{TP}{TP + FP} \qquad Recall = \frac{TP}{TP + FN}$$

$$F1 = 2\left(\frac{Precision.Recall}{Precision + Recall}\right)$$

Assessing Cluster Quality



True contig labels



Clusters

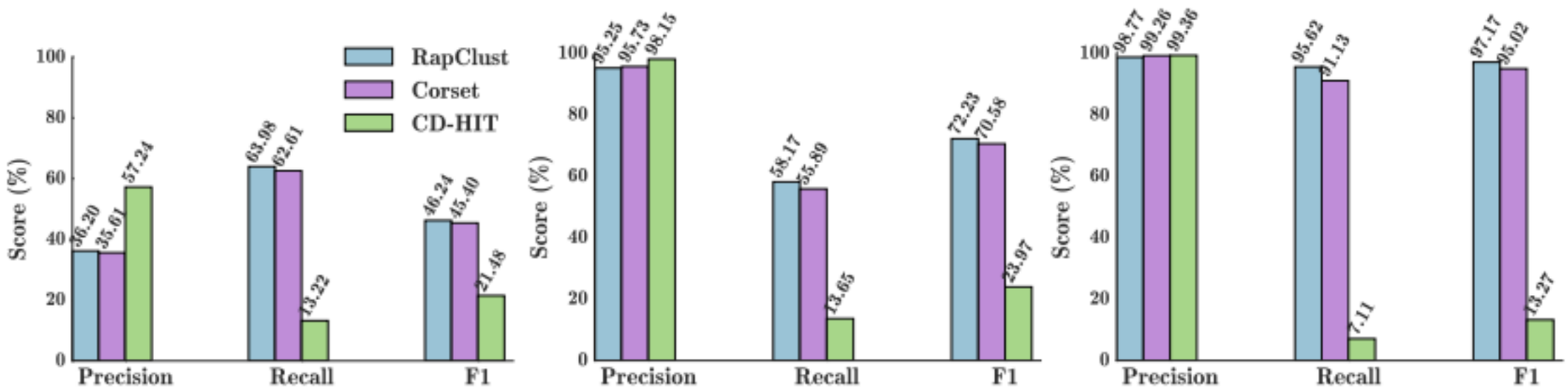
TP - 4
FN - 8
FP - 4
TN -12

Scoring

$$Precision = \frac{4}{4+4} = 0.5 \qquad Recall = \frac{4}{4+8} = 0.33$$

$$F1 = 2\left(\frac{0.5 \times 0.33}{0.5 + 0.33}\right) = 0.398$$

Assessing Cluster Quality



(a) Accuracy for yeast (b) Accuracy for human (c) Accuracy for chicken

RapClust has better accuracy than CD-HIT with other two tools

Assessing Cluster Quality

1,2,3,4,5,6,7,8,9,10,11

Clustering A

C(A,1) = {1,2,4}
C(A,2) = {8,10,11}
C(A,3) = {9}
C(A,4) = {3,5,6,7}

Clustering B

C(B,1) = {1,2,3}
C(B,2) = {8,10,11}
C(B,3) = {4,5,6,7,9}

$$r_{ij} = \frac{|C(A,i) \cap C(B,j)|}{n}$$

r(1,1) = 2/11 ; r(1,3) = 1/11 ; r(2,2) 3/11 etc.

$$VI(A;B) = -\sum_{ij} r_{i,j}[\log(\frac{r_{ij}}{\frac{|C(A,i)|}{n}}) + \log(\frac{r_{ij}}{\frac{|C(B,i)|}{n}})]$$

Assessing Cluster Quality

	RapClust	Corset	CD-HIT
Chicken	**0.127**	0.191	2.01
Human	**0.712**	0.735	1.24
Yeast	**0.176**	0.178	0.216

Variation of information (VI) between predicted and "true" clustering for all methods / datasets.

Transcripts that are not part of any clusters are put in a separate cluster.

Differential gene expression

In the process of contig clustering often we miss out differentially expressed genes, that remain hidden due problems in clustering.

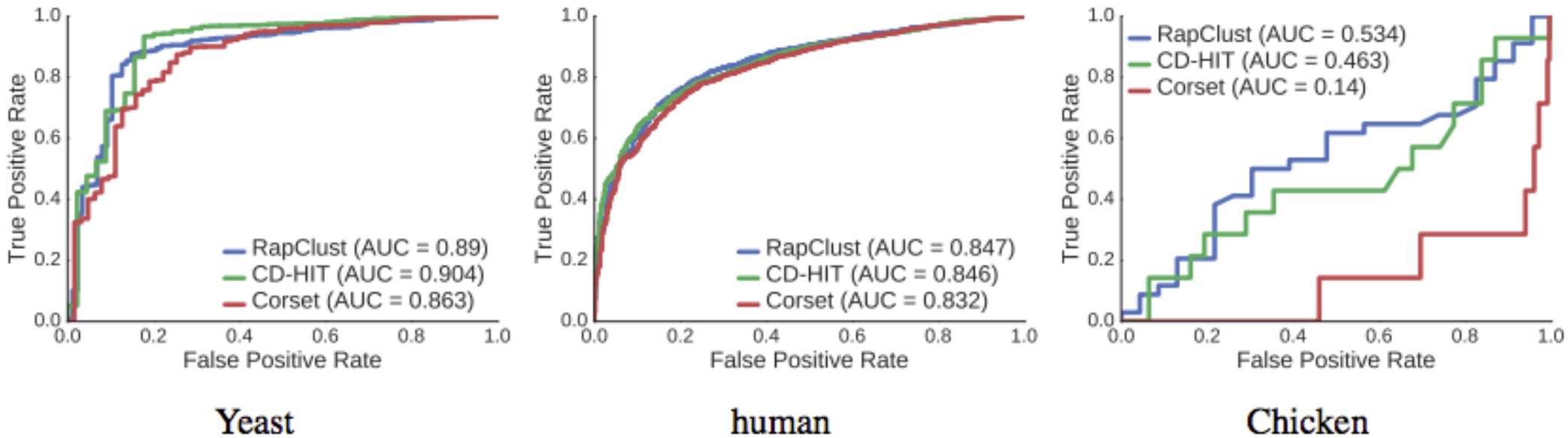
- Over clustering increases false-positives, thereby decreases *precisions.*
  - If true differentially expressed genes are clusters they would never be detected
- Under clustering causes poor *recall.*

Differential gene expression

Data used to test DGE

Clustering Method	Quantification
Truth	Sailfish (contig-level)
RapClust	Sailfish (contig-level)
Corset	Corset (cluster count)
CD-Hit-EST	Sailfish(contig-level)

Differential gene expression



RapClust predictions are concordant with ground truth in terms of gene-level differential expression

Benefits of RapClust

- Fast & accurate clustering
- Works on results of contig-level quantification methods
- The contig-level expressions are not lost in case it is required
- The cluster-level quantification estimates can be used when an approximation of gene-level results are needed, as we saw aggregation worked !

Potential Improvements

Data-driven estimation and usage of clustering parameters (e.g. edge weight cutoffs in MCL):

- We can obtain better results than shown here, but don't yet have a good way to find cutoffs automatically

Richer test for paralogs:

- Instead of adopting the simple count-based paralog test of Corset, we can incorporate the variance information over counts that Sailfish (with quasi-mapping) can produce using posterior Gibbs sampling.

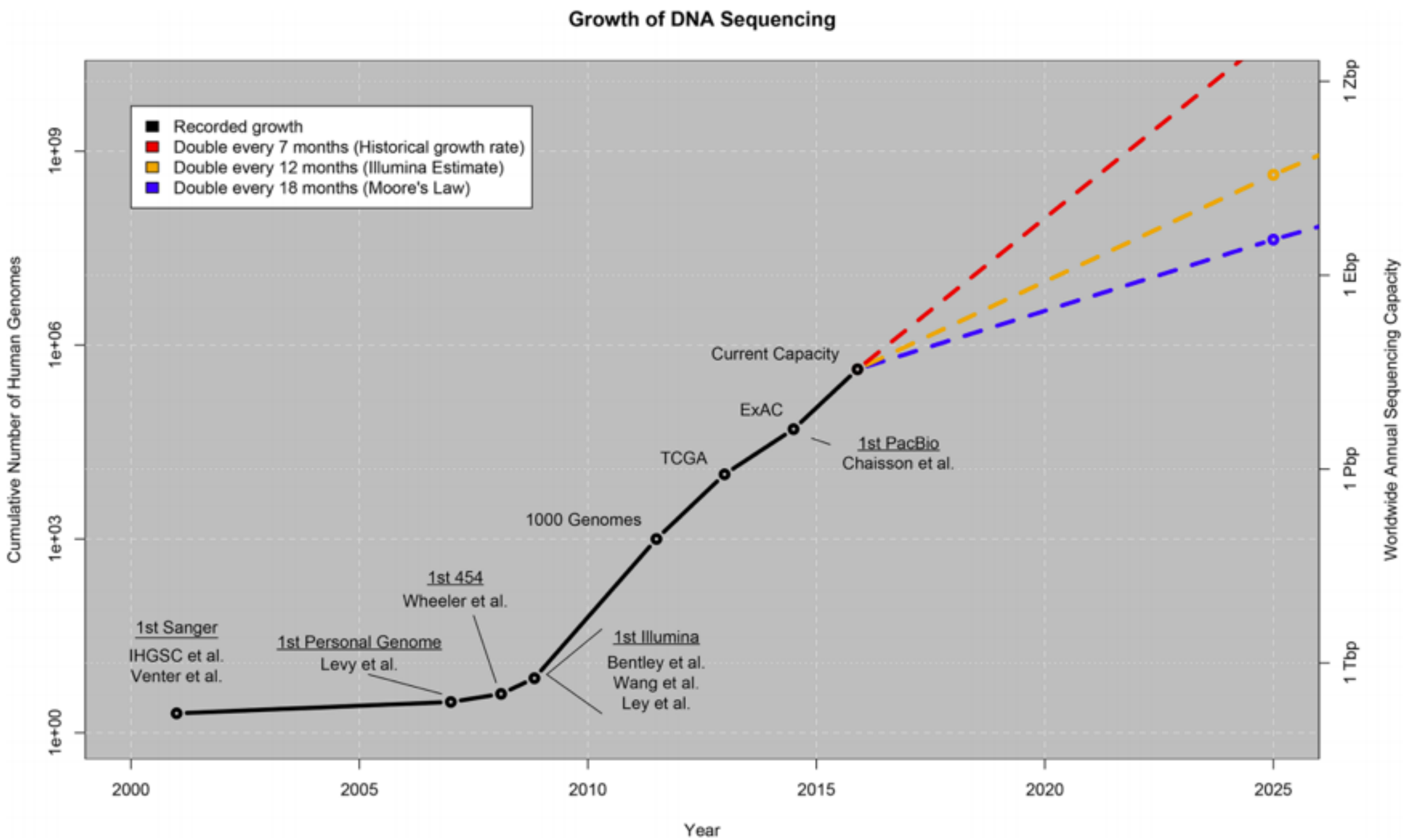
Richer notion of contig similarity:

- For example, align co-clustered contigs & look for evidence of alternative splicing

Why Compress

- Size of sequencing databases, all over the world is increasing at a rapid pace.
- Although memory is getting cheaper, but the communication cost does not.
- Downloading a sequence file
  fastq.gz file ~ t seconds
  compressed file ~ t/k seconds
  You can essentially download k sequence files in same amount of time.

Why Compress ?



Zachary D. Stephens, Skylar Y. Lee, Faraz Faghri, Roy H. Campbell, Chengxiang Zhai, Miles J. Efron, Ravishankar Iyer, Michael C. Schatz, Saurabh Sinha, Gene E. Robinson Big Data: Astronomical or Genomical? PLoS Biol. 2015 Jul; 13(7): e1002195. Published online 2015 Jul 7. doi: 10.1371/journal.pbio.1002195

Quark Compression*

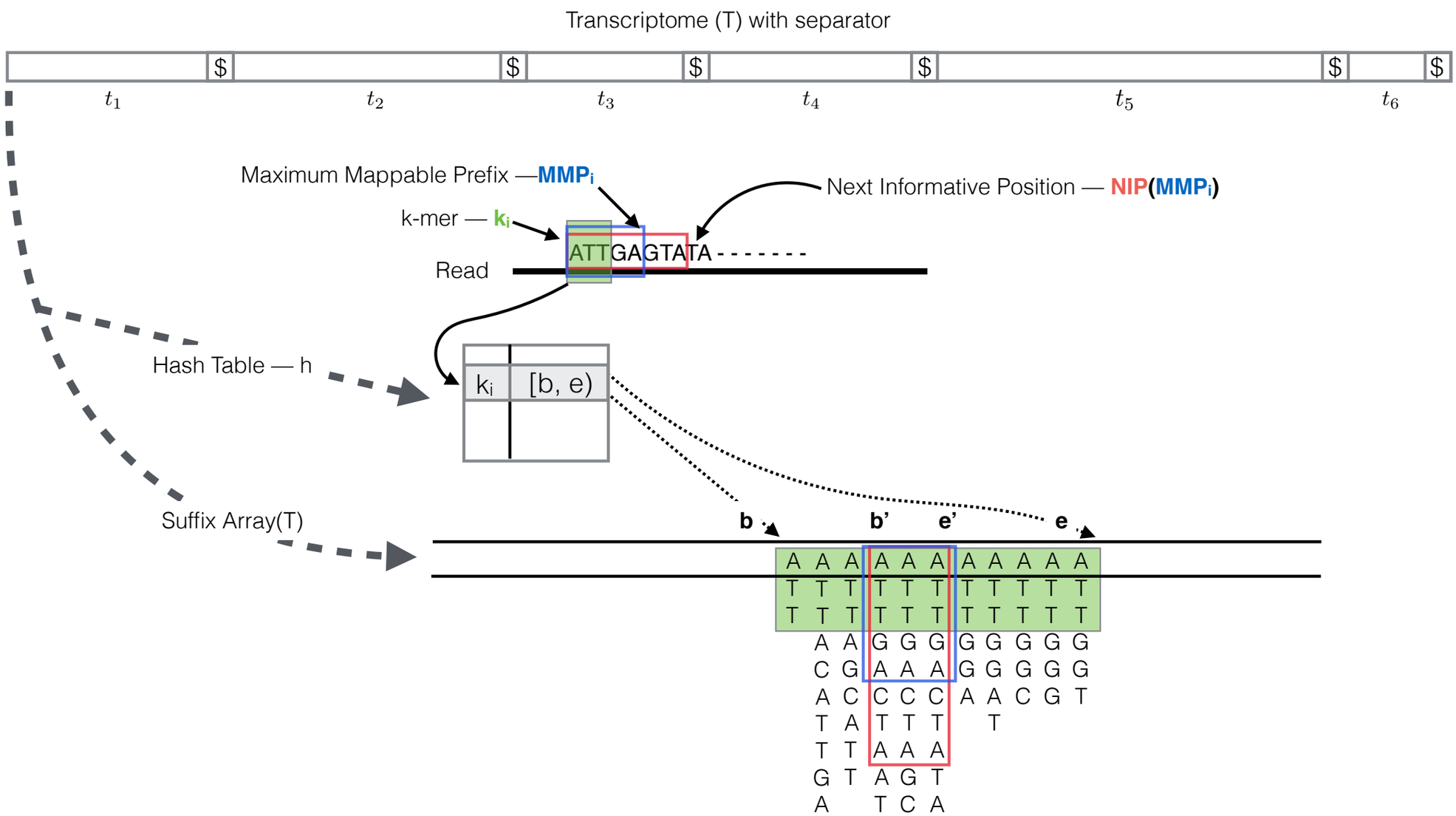
- Reference based scheme where BAM file is compressed. Often specific aligners are also used to navigate meta-data problem.

- Reference-free scheme where similar sequence are put together in order to get.
  Often uses local assembly based approach

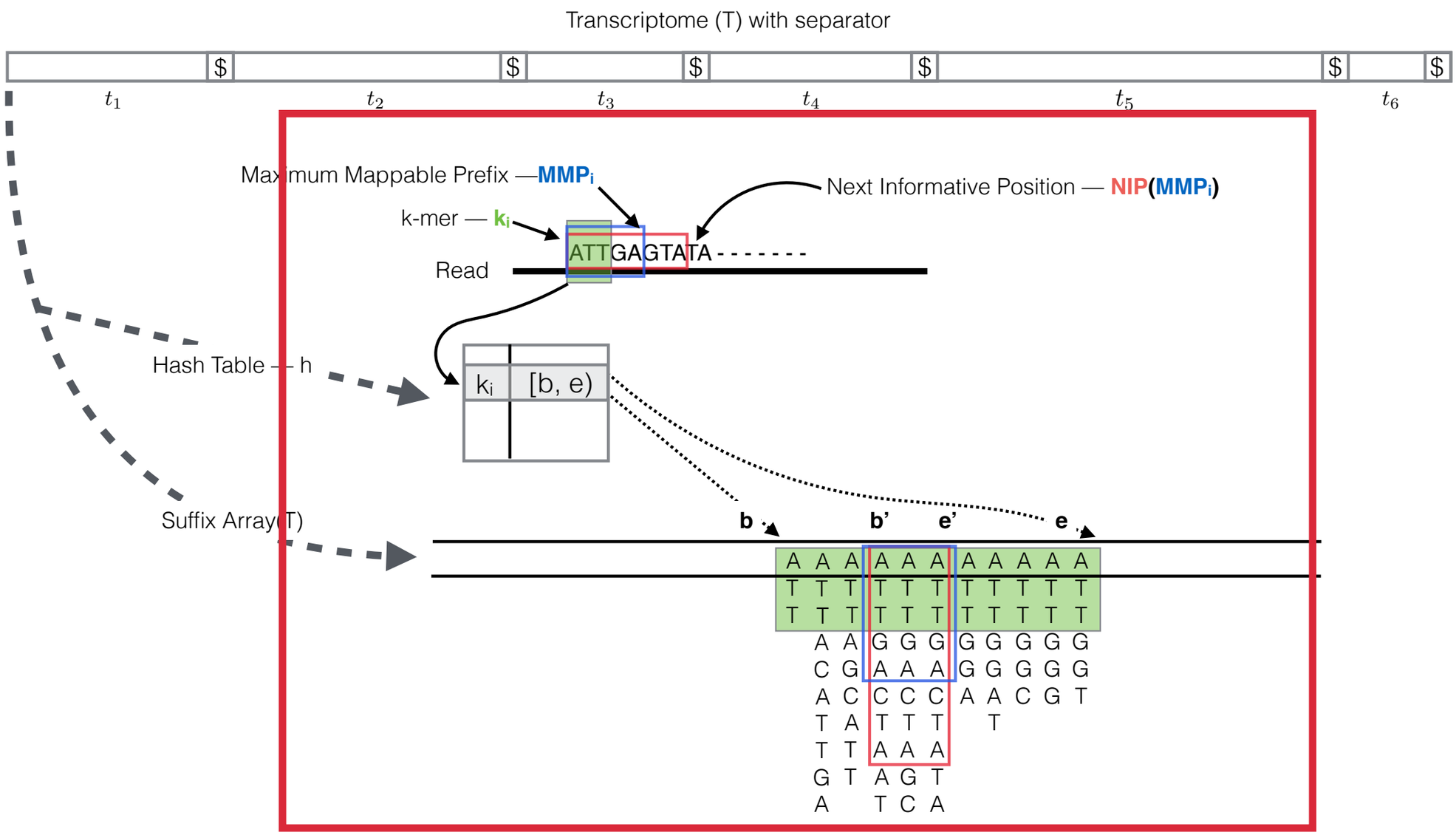
- Quark: *semi-reference* based compression, where, reference is used on compression end, not on the decompression end.

* not released yet

Motivation


Transcriptome (T) with separator
$ $ $ $ $ $
t1 t2 t3 t4 t5 t6
Maximum Mappable Prefix —MMPi
k-mer — ki
Next Informative Position — NIP(MMPi)
ATTGAGTATA - - - - - - -
Read
Hash Table — h
ki [b, e)
Suffix Array(T)
b
b'
e'
e
A A A A A A A A A A A
T T T T T T T T T T T
T T T T T T T T T T T
A A G G G G G G G G
C G A A A G G G G G
A C C C C A A C G T
T A T T T T
T T A A A
G T A G T
A T C A

Motivation


Transcriptome (T) with separator
$ $ $ $ $ $
t_1 t_2 t_3 t_4 t_5 t_6
Maximum Mappable Prefix — **MMP$_i$**
k-mer — **k$_i$**
Next Informative Position — **NIP**(**MMP$_i$**)
ATTGAGTATA - - - - - - -
Read
Hash Table — h
k$_i$
[b, e)
Suffix Array(T)
b
b'
e'
e
A A A A A A A A A A A
T T T T T T T T T T T
T T T T T T T T T T T
A A G G G G G G G G
C G A A A G G G G G
A C C C C A A C G T
T A T T T T
T T A A A
G T A G T
A T C A

Motivation

1. **Condition** for *mapping* a read to a target transcript is finding **at least** one ***k-mer*** that is shared between read and the target transcript. Moreover among all the matches, the set of maximally matched transcripts would be reported.
2. Reads that are *mapped* to same set of transcripts are put together in the same equivalence class.

↓

All reads in one equivalence class share at least one ***k-mer*** with mapped transcript

Motivation

1. **Condition** for Reads to be in the same equivalence class is that they share the same set of transcripts.

↓

All reads in one equivalence share the same transcripts

Intuition of Quark

All reads in one equivalence class share at least one ***k-mer*** with mapped transcript

↓

Encode all reads in that equivalence class with respect to that one transcript

All reads in one equivalence class share the ***same*** transcripts

↓

Save one transcript for entire equivalence class

Challenges

Data specific

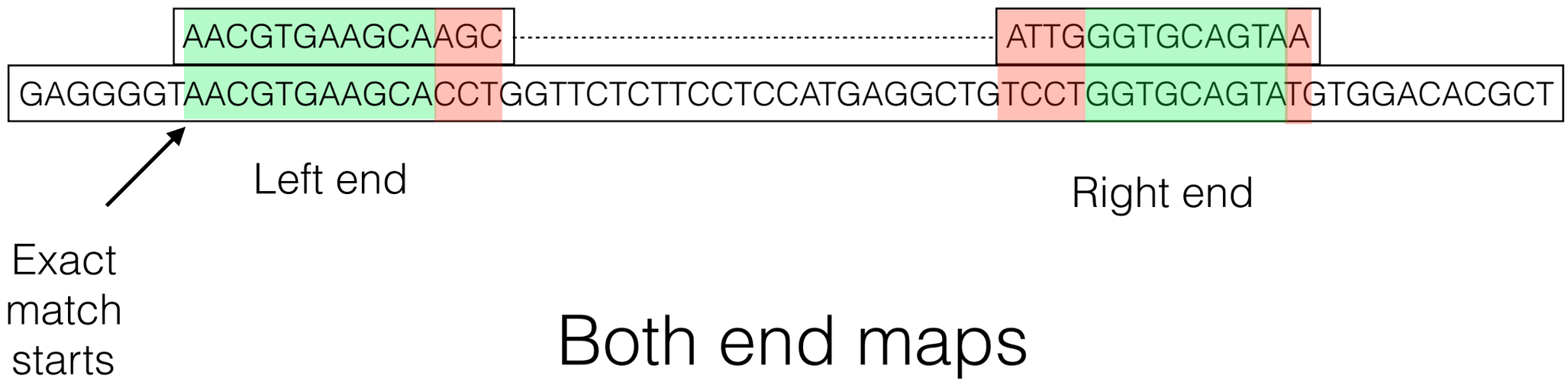
- Wide array of cases to handle. (orphan reads,partially mapped reads etc)

Computational

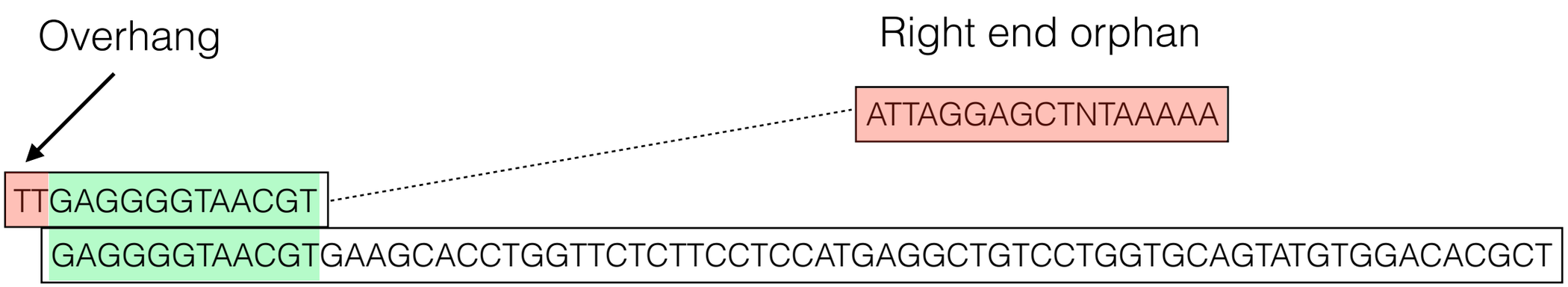
- **Speed:** We are **not** compressing a sam file. Our mapping method is *streaming*.

- **Parallelization:** We simultaneously encode several reads in a parallel fashion.

Looking closely at Quark

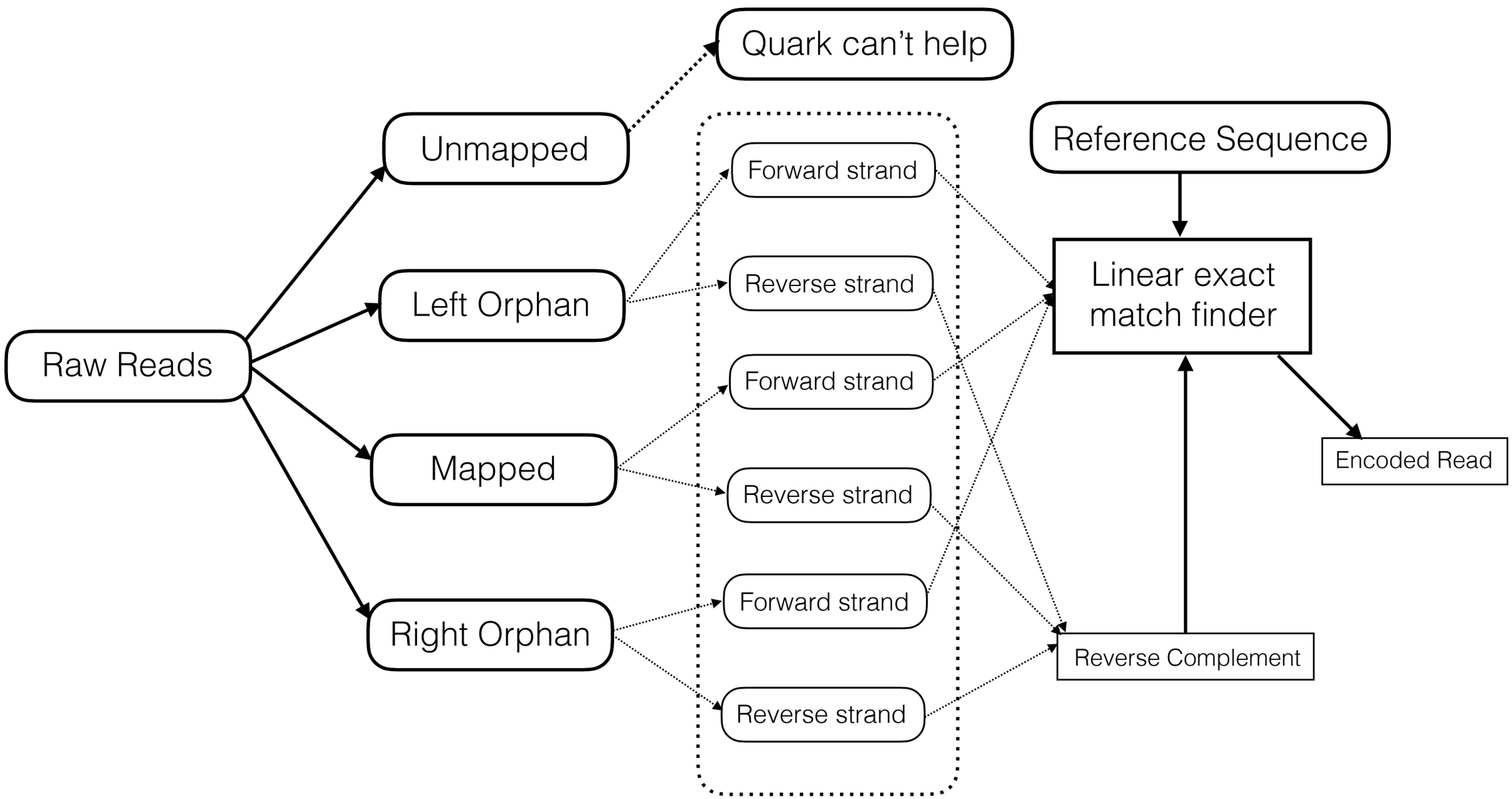


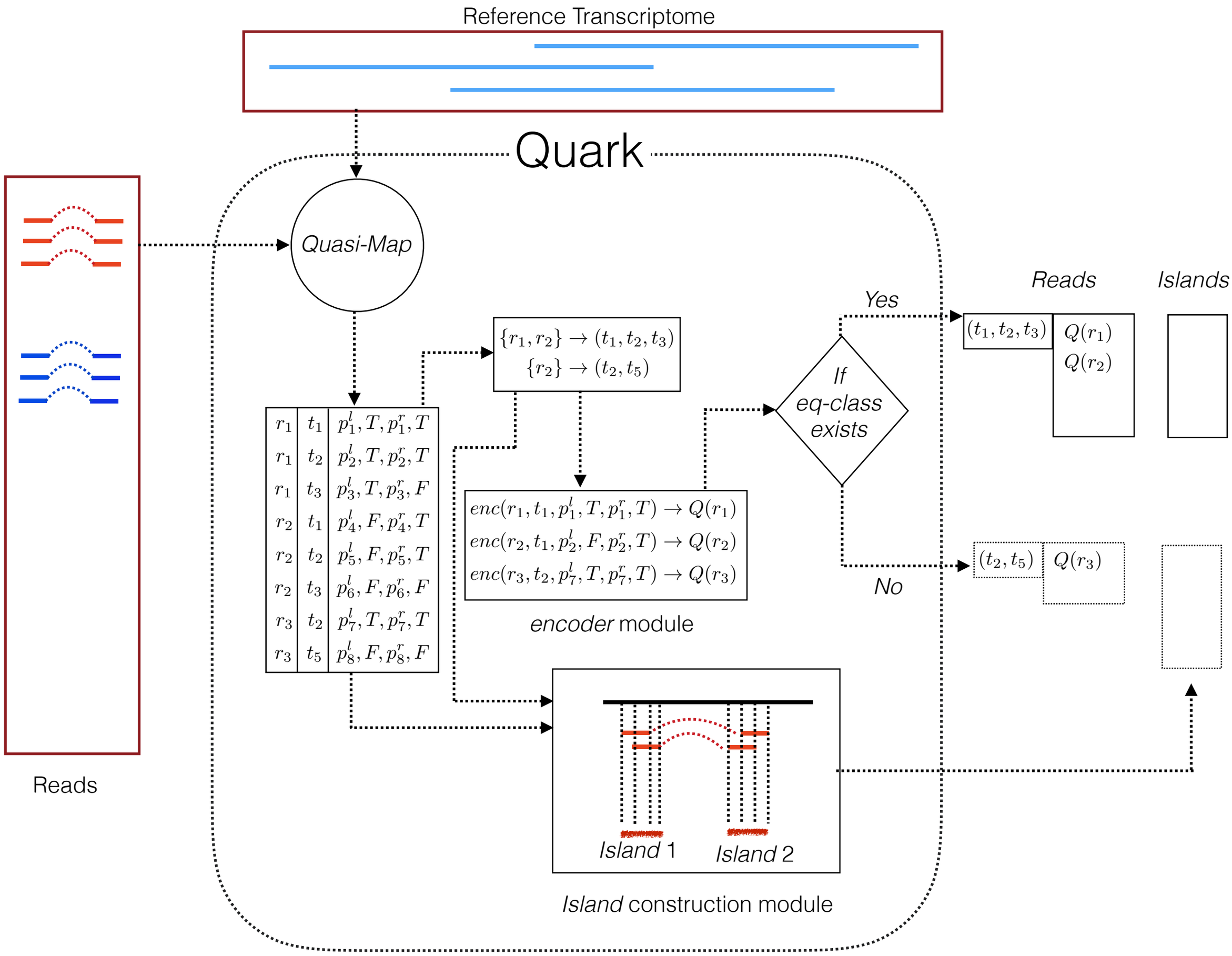
Both end maps



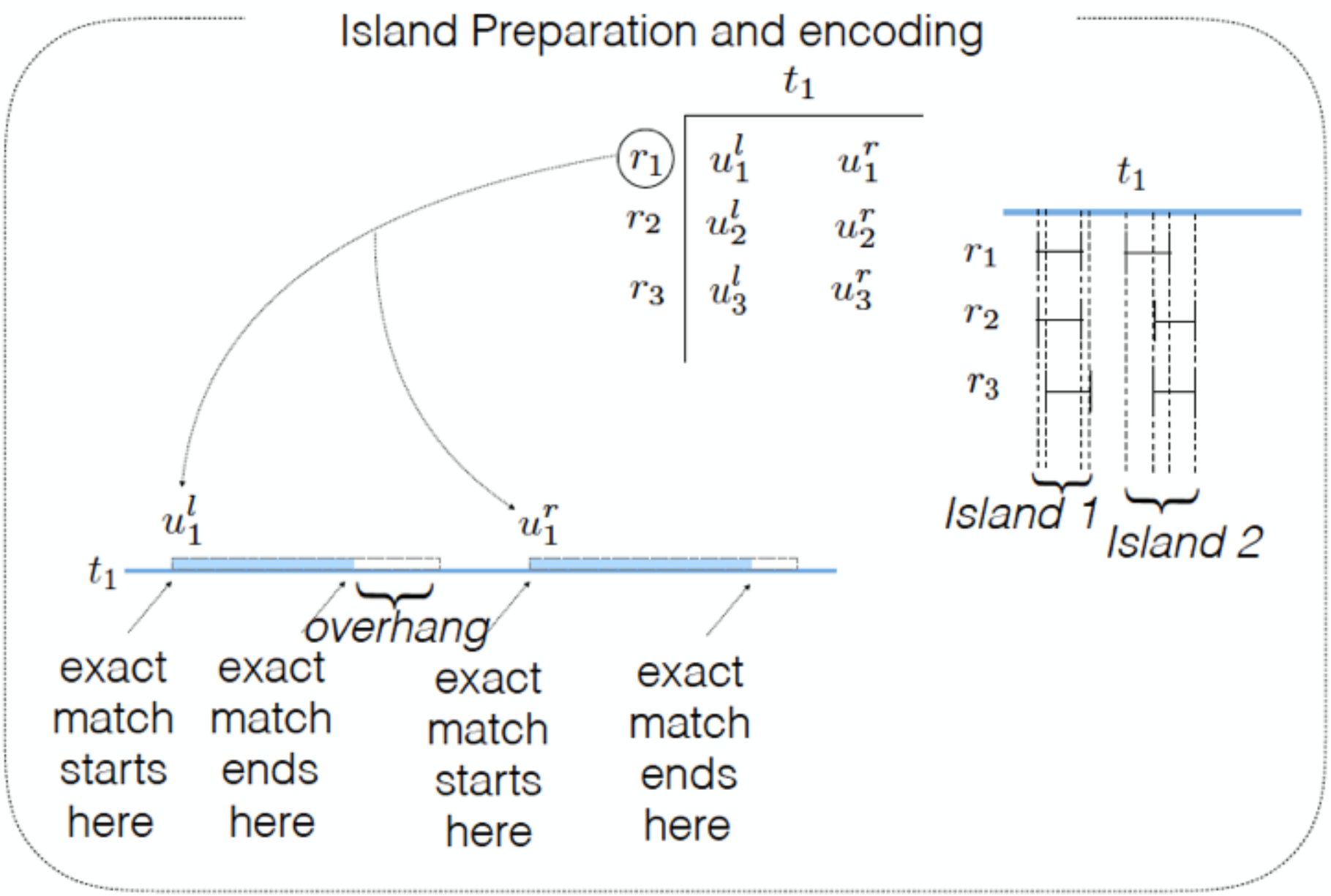
One end is Orphan

Looking closely at Quark




Reference Transcriptome
Quark
Quasi-Map
Reads
r_1 t_1 p_1^l, T, p_1^r, T
r_1 t_2 p_2^l, T, p_2^r, T
r_1 t_3 p_3^l, T, p_3^r, F
r_2 t_1 p_4^l, F, p_4^r, T
r_2 t_2 p_5^l, F, p_5^r, T
r_2 t_3 p_6^l, F, p_6^r, F
r_3 t_2 p_7^l, T, p_7^r, T
r_3 t_5 p_8^l, F, p_8^r, F
$\{r_1, r_2\} \rightarrow (t_1, t_2, t_3)$
$\{r_2\} \rightarrow (t_2, t_5)$
$enc(r_1, t_1, p_1^l, T, p_1^r, T) \rightarrow Q(r_1)$
$enc(r_2, t_1, p_2^l, F, p_2^r, T) \rightarrow Q(r_2)$
$enc(r_3, t_2, p_7^l, T, p_7^r, T) \rightarrow Q(r_3)$
encoder module
If eq-class exists
Yes
No
Reads
Islands
(t_1, t_2, t_3)
$Q(r_1)$
$Q(r_2)$
(t_2, t_5)
$Q(r_3)$
Island 1
Island 2
Island construction module

Closer look to *islands*

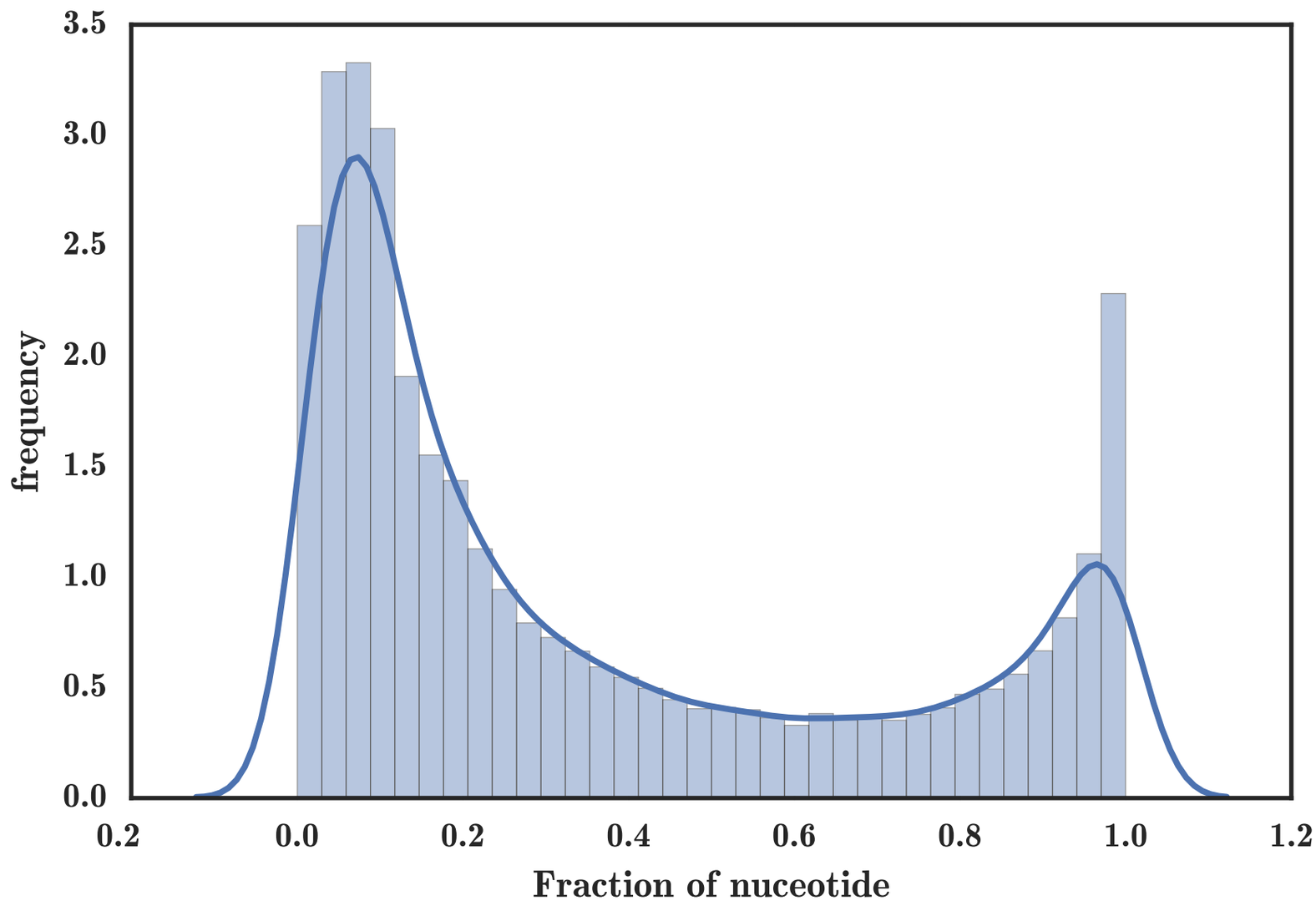


Do we need entire transcript ?

We have observed that only some part of the transcript are responsible for a large proportion of reads.

Moreover we measured what percentage of a transcript is really ever act as a target sequence in an entire experiment

Do we need entire transcript ?

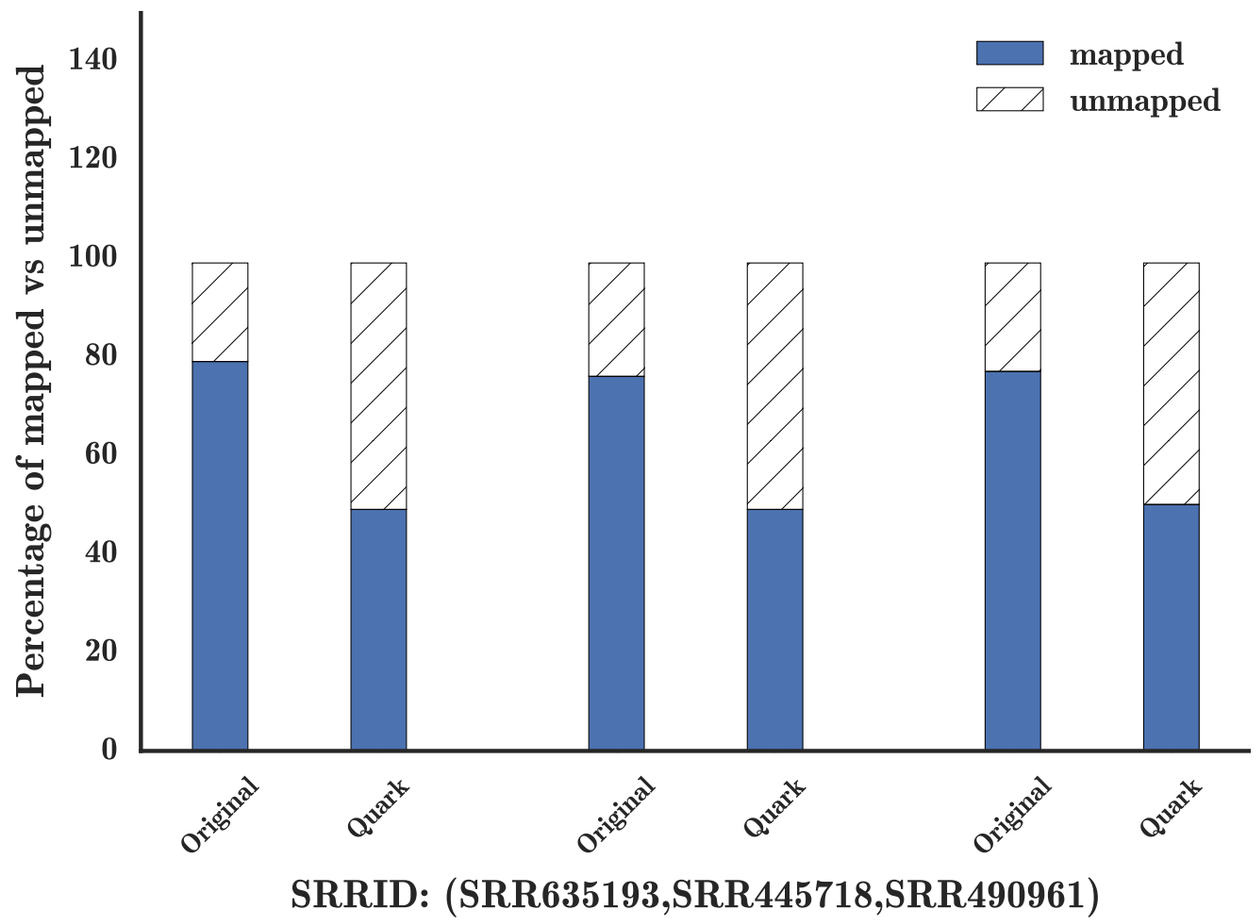


Compression results

	fq.gz	Quark	leon	SCALCE
SRR635193	2,329,761,348	**140,069,396**	384,918,555	294,962,419
SRR445718	2,848,487,774	**171,723,783**	328,915,302	253,352,974
SRR490961	4,115,142,514	**203,458,347**	466,622,492	301,777,076

Quark compressed sequence size is smaller than *leon* and *SCALCE* on both paired end and single end data

Percentage of mapped read matters



The left bar shows the percent of mapped reads and unmapped reads before compression

Potential future work

- The concept of *islands* can be useful in finding out novel splicing sites.
- In ideal case islands should be exons, in *de novo* world.
- So islands from same equivalence class together can infer if a splicing event had happened.
- The ultimate goal is to allow the splicing analysis in *de novo* world.

Reference

(a)Kallisto: Nicolas Bray, Harold Pimentel, Páll Melsted, and Lior Pachter. Near-optimal rna-seq quantification. *arXiv preprint arXiv: 1505.02710*, 2015.,

(b)Salmon: Rob Patro, Geet Duggal, and Carl Kingsford. Accurate, fast, and model-aware transcript expression quantification with salmon. *bioRxiv*, page 021592, 2015.,

(c)RapMap: Avi Srivastava, Hirak Sarkar, Nitish Gupta, and Rob Patro. Rapmap: A rapid, sensitive and accurate tool for mapping rna-seq reads to transcriptomes. *bioRxiv*,

(d) Corset: Davidson NM, Oshlack A. Corset: enabling differential gene expression analysis for de novo assembled transcriptomes. Genome Biol. 2014;15:410.

(e)CD-HIT Li W, Godzik A. Cd-hit: a fast program for clustering and comparing large sets of protein or nucleotide sequences. Bioinformatics 2006;22:1658-1659.

Paralog filter

All boils down to removal of an edge or not. If we remove an edge between two contigs, which are truly paralog then it would be a win. From biological properties we know that paralogs can develop different biological functions

$$X_a^i \sim \lambda_a^i \qquad X_a^i \sim \lambda_a^j$$
$$X_b^i \sim \lambda_a^i \qquad X_b^i \sim \lambda_b^j$$

H_0 : The constant of proportionality between abundance of t_i and t_j is the same

H_1 : The constant of proportionality between abundance of t_i and t_j is different