Minimal perfect hash functions in large scale bioinformatics Problem
Antoine Limasset, Camille Marchet, Pierre Peterlongo, Lucie Bittner

To cite this version:
Antoine Limasset, Camille Marchet, Pierre Peterlongo, Lucie Bittner. Minimal perfect hash functions in large scale bioinformatics Problem. JOBIM 2016, Jun 2016, Lyon, France. <hal-01341718>

HAL Id: hal-01341718
https://hal.archives-ouvertes.fr/hal-01341718
Submitted on 4 Jul 2016

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers. L’archive ouverte pluridisciplinaire HAL, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d’enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.
(Meta)Genomic Data

Billions of short sequences of hundreds of base pairs, from one or multiple genomes

Questions

Dataset comparison: Detection of similar reads inter or intra datasets

Problem

Indexing huge set of elements

Problem

False positive rate:

\[ \frac{1}{2^{\phi / 2}} \]

Memory consumption:

Bit/elements: 10 FP rate: 1/10^6
Bit/elements: 20 FP rate: 1/10^5
Bit/elements: 30 FP rate: 1/10^8

... BUT

- No membership operation
- A ‘stranger’ key can be associated to a value

Hash functions

- Classical hashing
- Perfect hashing (no collision)
- Minimal perfect hashing (no collision, image buckets)

BBhash library

- Memory efficient (less than 3bits per key)
- Fast query (200ns)
- Fast to construct (even for billions elements)

Quasi-dictionary

Put a fingerprint in the value and check it at the query

Short Read Connector tools

Given A and B sets of reads:

SRC Linker:
Output reads in A that has T Kmers that appear in set B.

SRC Counter:
Output reads in A that has T Kmers in common with the reads of B and estimate their coverage.

Results

Time for indexing and querying 1M reads with SRC Linker

Time for indexing and querying 100M reads with SRC Counter

Memory used for indexing 100M reads with SRC Counter

Less pressure on your machine!

Still have to assess the qualitative aspects of our methods...

References

Bowtie2

BWA

Starcode

BLAST

SRC

Links

BBhash library:
github.com/rizkg/BBHash

Quasi-dictionary:
github.com/pierrepeterlongo/quasi_dictionary

Short Read Connector:
github.com/GATB/rconnector

QR code

Team