Lab for Data Intensive Biology

Decentralized indexes for public genomic data

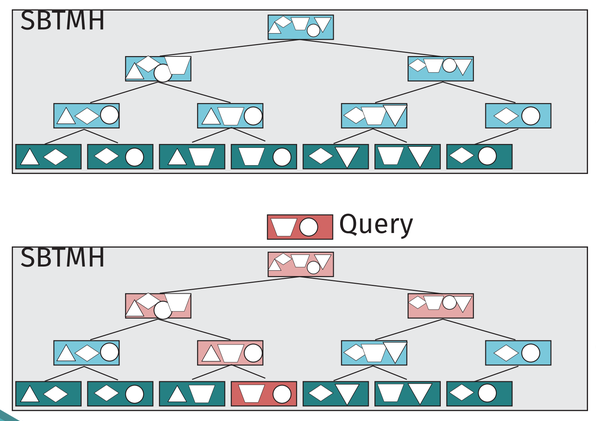
Luiz Carlos Irber Júnior[1], C. Titus Brown[1], Tim Head[2]
lcirberjr@ucdavis.edu, ctbrown@ucdavis.edu, tim@wildtreetech.com
[1]Department of Population Health and Reproduction, University of California, Davis, USA
[2]Head's Wild Tree Tech, Switzerland

@luizirber @ctitusbrown @betatim

luizirber/2017-recomb

Introduction

MinHash [Broder, 1997] is a technique for **estimating the similarity** of two or more datasets. Expanding on the work pioneered by **Mash** [Ondov et al, 2016] and extended in our library **sourmash** [Brown and Irber, 2016], we calculated signatures for **412 thousand microbial reads datasets** on the **Sequence Read Archive**. To be able to **efficiently search for matches** of these signatures in the **RefSeq microbial genomes database** we developed a new data structure based on **Sequence Bloom Trees** [Solomon and Kingsford, 2016] adapted for **searching MinHash signatures** (named **SBTMH**) to **index signatures** and made it available publicly.



The SBTMH is a **binary tree** where **leaf nodes** are **MinHash signatures** and **internal nodes** are **Bloom Filters**. Each Bloom Filter can be queried for approximate membership of **all the values from its children**, and so the **root node** roughly represents **all the values from all signatures** in the tree.

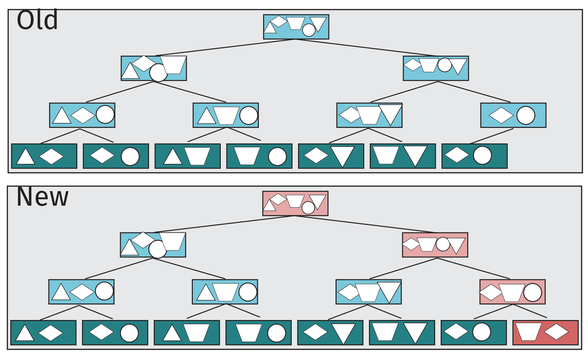
Searching for **similarity to a query signature** involves checking for **query elements** present in **each internal node**, and if it **doesn't reach the threshold** the subtree is **pruned**. If a **leaf is reached**, it is **returned as a match** to the query signature.

Saving a SBTMH in IPFS

The SBTMH structure can be encoded as **nodes in a MerkleDAG** and stored in **IPFS** (InterPlanetary File System) [Benet, 2014]. Typical **data archive systems**, like **Amazon S3** or the **NCBI SRA**, stop working when the **central service is down**. **IPFS nodes** can communicate and synchronize data **without requiring a central source**, and they can also **serve data requests among them**, which benefits from local networks and **increases the bandwidth** available for **data transfers**.

The SBTMH behaves like a **persistent data structure** [Driscoll et al, 1989], where new versions of a SBTMH (after new nodes are added or removed) **can share parts of the structure of previous versions**. While this is usually used to avoid duplicating data on pure functional programming languages, for our use case it is important because it allows **remixing of indexes and signatures**: users can **expand an index with their own signatures**, and share the new index with other users.

Signatures calculated from **public datasets** can also be **shared**: by indexing **RefSeq** and **GenBank** and sharing the signatures on IPFS, users can become **curators** by **selecting organisms** of interest and creating **SBTMH indexes that fit their needs** or the needs of a specific area.



Adding a new signature to SBTMH causes **parent nodes to be updated**, but other nodes are not affected. This means users from both SBTMH can benefit from **increased availability** of the data for the nodes that didn't change (and are shared among trees).

Sharing datasets in IPFS

IPFS is a **content-addressable storage**, meaning that a file will always have the **same multihash** value (and so, the same **address in the network**), as long as the content is the same. This can change how databases and archives (like the SRA) are offered and implemented, since users can collaborate by choosing to **share subsets of the archive** and **spread the network bandwidth**. More importantly, it **avoids the central point of failure**, while still allowing for **curation** and **quality assurance** of the data.

A common way of interacting with the SRA is using the **SRA Toolkit**, which **generates a local cache**. By indexing their local caches and sharing on IPFS, other users can **download data directly from peers**, **decreasing the load** on the central SRA servers. If the **central repository is not available** [GB Editorial Team, 2011], the data can still be found by **connecting to peers instead**. SRA submission guidelines can also be updated to require that each submission is **available for a period of time** on the **submitter IPFS node** after it is accepted, to **increase redundancy** and help **seed the content** to other users.

Future Work

Inserting signatures in SBTMH can be optimized: The current implementation finds the next available leaf position and puts the signature in it. This can lead to **longer searches**, since similar signatures might end up very far apart in the tree. By **inserting signatures based on similarity**, the search can potentially be **pruned earlier**, **reducing runtime**.

IPFS continues to evolve, and a promising technology is **IPLD**, a **data description format** that maps well to the way SBTMH are saved: a JSON file with a representation of the tree, where each node points to an IPFS objects.

Another interesting IPFS technology for the signature calculation service is a **PubSub implementation** that can replace Amazon SQS as the event notification system. On top of **decentralized storage** aspects of this project, this also makes possible a **loosely-coupled distributed computation system**.

References

Benet, Juan. 2014. "**IPFS - Content Addressed, Versioned, P2P File System.**" arXiv:1407.3561 [Cs], July. http://arxiv.org/abs/1407.3561.

Broder, Andrei Z. 1997. "**On the Resemblance and Containment of Documents.**" In Compression and Complexity of Sequences 1997. Proceedings, 21–29. IEEE. http://ieeexplore.ieee.org/abstract/document/666900/.

GB Editorial Team. 2011. "**Closure of the NCBI SRA and Implications for the Long-Term Future of Genomics Data Storage.**" Genome Biology 12: 402. doi:10.1186/gb-2011-12-3-402.

Driscoll, James R., Neil Sarnak, Daniel D. Sleator, and Robert E. Tarjan. 1989. "**Making Data Structures Persistent.**" Journal of Computer and System Sciences 38 (1): 86–124. doi:10.1016/0022-0000(89)90034-2.

Ondov, Brian D., Todd J. Treangen, Páll Melsted, Adam B. Mallonee, Nicholas H. Bergman, Sergey Koren, and Adam M. Phillippy. 2016. "**Mash: Fast Genome and Metagenome Distance Estimation Using MinHash.**" Genome Biology 17: 132. doi:10.1186/s13059-016-0997-x.

Solomon, Brad, and Carl Kingsford. 2016. "**Fast Search of Thousands of Short-Read Sequencing Experiments.**" Nature Biotechnology 34 (3): 300–302. doi:10.1038/nbt.3442.

Titus Brown, C., and Luiz Irber. 2016. "**sourmash: A Library for MinHash Sketching of DNA.**" The Journal of Open Source Software 1 (5). doi:10.21105/joss.00027.