Master Thesis

LCP Construction with Prefix Doubling and DC3/7 in Thrill

Overview

Suffix arrays are among the most popular data structures for text indexing and the basis for many string and compression algorithms. In combination with additional auxiliary arrays like the LCP array or the Φ array, suffix arrays can emulate powerful text data structures like the suffix tree and be used to solve a myriad of combinatorial string problems but also real-world bioinformatics applications.

In previous work we presented Thrill [1], a distributed next-generation high-performance algorithmic big data processing framework in C++. One of the premiere applications of Thrill is distributed suffix sorting, for which we developed and implemented five algorithms [2].

The objective of this master thesis is to extend the fastest suffix sorters, prefix doubling with discarding, DC3, and DC7, with LCP array construction in Thrill. In prior work, Feist showed how to extend DC3 with LCP array construction in external memory using batched range minimum queries [3]. A similar technique can be used for distributed machines.

Distributed suffix and LCP array construction with Thrill will enable processing of large inputs using a cluster of parallel machines. These two arrays can then be used to construct more complex data structures such as compressed suffix arrays or String B-Trees. By implementing these subsequent steps also using Thrill and thus combining the algorithms efficiently, text indices for very large instances will become practical.

Requirements

- Interest in stringology algorithms and distributed big data processing
- Excellent programming skills in C++
- Ability to think for yourself

Info: Other related string and text processing topics are available.

