Indexing Compressed Text: A Tale of Time and Space

Nicola Prezza
LUISS Guido Carli, Rome, Italy
https://nicolaprezza.github.io/
nprezza@luiss.it

Abstract

Text indexing is a classical algorithmic problem that has been studied for over four decades. The earliest optimal-time solution to the problem, the suffix tree [11], dates back to 1973 and requires up to two orders of magnitude more space than the text to be stored. In the year 2000, two breakthrough works [6, 3] showed that this space overhead is not necessary: both the index and the text can be stored in a space proportional to the text’s entropy. These contributions had an enormous impact in bioinformatics: nowadays, the two most widely-used DNA aligners employ compressed indexes [9, 8]. In recent years, it became apparent that entropy had reached its limits: modern datasets (for example, collections of thousands of human genomes) are extremely large but very repetitive and, by its very definition, entropy cannot compress repetitive texts [7]. To overcome this problem, a new generation of indexes based on dictionary compressors (for example, LZ77 and run-length BWT) emerged [7, 5, 1], together with generalizations of the indexing problem to labeled graphs [2, 10, 4]. This talk is a short and friendly survey of the landmarks of this fascinating path that took us from suffix trees to the most modern compressed indexes on labeled graphs.

2012 ACM Subject Classification Theory of computation → Data compression; Theory of computation → Sorting and searching; Theory of computation → Pattern matching

Keywords and phrases Compressed Text Indexing

Digital Object Identifier 10.4230/LIPIcs.SEA.2020.3

Category Invited Talk

References

Indexing Compressed Text: A Tale of Time and Space


