Streaming Pattern Matching

Tatiana Starikovskaya
DIENS, École normale supérieure, PSL Research University, Paris, France

Abstract

Many classical algorithms for string processing assume that the input can be accessed in full via constant-time random access, which poses a serious limitation in the modern era of data deluge. In this talk, we will focus on the streaming model of computation that allows to overcome this issue. In this model of computation, we assume that the input arrives as a stream, one character at a time, which captures a situation when the data are sequential measurements or an output of an algorithm. The space complexity is defined as all the space used, including the space used to store any information about the input, which allows to develop ultra-efficient algorithms.

The first streaming algorithm for pattern matching was presented in the seminal paper of Porat and Porat in FOCS 2009. For a pattern of length m, the algorithm uses only $O(\log m)$ space, while any classical algorithm requires $\Omega(m)$ space. This result served as a foundation of the area of streaming algorithms for pattern matching. After a brief survey of the area, we will discuss two questions in more details: the k-mismatch problem and the pattern matching with k-edits problem. In the k-mismatch problem, one is given a pattern and a text, and the task is to find all substrings of the text that have at most k mismatches with the pattern. The current best algorithm for this problem was given by Clifford, Kociumaka, and Porat in SODA 2019, and for a pattern of length m it uses $O(k \log m)$ space and $\tilde{O}(\sqrt{k})$ time per character of the text. In the pattern matching with k-edits problem, the task is similar, but one must find substrings that can be transformed into the pattern by at most k edits, i.e. substitutions, insertions, and deletions of a character. For this problem, the first streaming algorithm was presented by Kociumaka, Porat, and Starikovskaya in FOCS 2021. The algorithm takes $\tilde{O}(\text{poly}(k))$ space and $\tilde{O}(\text{poly}(k))$ time per character of the text.

2012 ACM Subject Classification Theory of computation → Pattern matching; Theory of computation → Sketching and sampling

Keywords and phrases Streaming algorithms, Pattern matching, Hamming distance, Edit distance

Digital Object Identifier 10.4230/LIPIcs.ISAAC.2021.1

Category Invited Talk

32nd International Symposium on Algorithms and Computation (ISAAC 2021).
Editors: Hee-Kap Ahn and Kunihiko Sadakane; Article No. 1; pp. 1:1–1:1
Leibniz International Proceedings in Informatics
Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany