Suffix Arrays. Application and Algorithms with Respect to Text/Digital Media

Aleksandar Gyorev
Jacobs University Bremen
a.gyorev@jacobs-university.de

January 16, 2014

Abstract

The suffix array is one of those underlooked data structures in modern computer science, that bear more power than you would expect at first glance. Invented by Manber and Myers [1] it proposed a completely different way of searching in the web. Unlike suffix trees [2], suffix arrays are much more efficient with respect to memory management and saving, but from speed perspective the difference seems to be negligible. In today's world we seek efficiency in both directions. When we search for something in a website it is most of the times just a single keyword. Which is what the data structure being discussed takes advantage of.

Speaking more technically, suffix array is a list, in sorted order, of all suffixes of a given character string. Once we have the sorted order, in order to identify whether the substring is contained within the web page we are going to use probably the most widespread algorithm known for searching sorted sequences - binary search [3], with the help of an auxiliary sparse table for finding the longest common prefix (often shortened to *lcp*).

Complexity of known algorithms for finding the array varies from $\mathcal{O}(n^2 \, log \, n)$ down to linear time precomputation. As quadratic and more complex algorithms are often impractical and not used in industry, we will focus on the faster ones. Namely the $\mathcal{O}(n \, log^2 \, n)$ and the $\mathcal{O}(n)$ ones. The main idea behind the first approach is that if we break two substrings (with their length being a power of 2) of the original one into two equal parts and we already know the result of their comparison we can determine the outcome in constant time. One of the logarithmic factor comes from the fact that we explore substring with length that is only power of 2, and the other from a quick sort [4] that is used for sorting the indices of different siffixes. The linear algorithm's (known as Skew Algorithm [5]) main idea is to divide the suffixes into three groups, depending on their parity in modulo 3, and separately handle the first two groups, and then merging them with the last one. The linearity comes from the radix sort [6] being used to sort and merge the buckets.

References

[1] Manber, Udi; Myers, Gene. *"Suffix arrays: a new method for on-line string searches"*. University of Arizona, 1990

[2] Weiner, P. *"Linear pattern matching algorithms"*. 1973

[3] Knuth, Donald. *The Art of Computer Programming.* Volume 3. Page 412, Algorithm C, 1968

[4] Hoare, C. A. R. *"Algorithm 64: Quicksort"*. 1961

[5] Kärkkäinen, Juha; Sanders, Peter. *"Simple Linear Work Suffix Array Construction"*. ISBN 978-3-540-40493-4, 2003

[6] Knuth, Donald. *The Art of Computer Programming.* Volume 3. pp. 168–179, 1997