Entropy-compressed bitvectors

Nicola Prezza

References and Reading

[1] Sections 4.1-4.3 of: Navarro, Gonzalo. Compact data structures: A practical approach. Cambridge University Press, 2016.

Exercises

1 Sparse arrays Let $A[1..n]$ be a word array (i.e. with entries of w bits each) of length n. Assume that at most n/w entries of A are different than 0.

1.1 Show how to represent A using at most $n + o(n)$ bits of space so that the following operations are supported in constant time:

- Access cell $A[i]$, for any $1 \leq i \leq n$
- Count number of non-zero cells in a range $A[i..j]$, for any $1 \leq i \leq j \leq n$

1.2 Using $n + o(n)$ bits, show how to count efficiently the number of integers starting with bit '1' (i.e. the most significant bit is equal to 1) in any prefix $A[1..j]$

1.3 Try to generalize the solution of point 1.2 so that it uses in total $2n + o(n)$ bits and it supports counting efficiently the number of integers starting with 00, 01, 10, or 11 in any prefix $A[1..j]$. Example: with $w = 3$ and $A = 001, 101, 110, 000, 001, 101, 001$, we have $count_{00}(4) = 2$ and $count_{10}(7) = 2$.

2 Rank on strings Propose a generalization of the `rank` operation to a more general alphabet $\Sigma = \{0, ..., \sigma - 1\}$ (note that bitvectors are the case $\sigma = 2$). Describe a space-and-time efficient data structure supporting this operation on a string from Σ^n.