Tutorial 7 Exercises

COMP526: Efficient Algorithms

18–19 November, 2024

Exercise 1. Compress the text T = `HANNAHBANSBANANASMAN` using a Huffman code; give

1. the character frequencies,
2. a step-by-step construction of the Huffman tree,
3. the Huffman code, and
4. the encoded text.
5. Finally, compute the compression ratio of the result (ignoring space needed to store the Huffman code).

Recall that we use the following conventions for building a Huffman tree:

- When merging two characters/nodes in the tree, the lower weight node becomes the left (0) child of the parent.
- Whenever there is a tie between weights, the node containing the alphabetically first child becomes the the left child of the parent.

Exercise 2. Prove the following *no-free-lunch* theorems for lossless compression.

1. *Weak version:* For every compression algorithm A and $n \in \mathbb{N}$ there is an input $w \in \Sigma^n$ for which $|A(w)| \geq |w|$, i. e. the "compression" result is no shorter than the input.

   Hint: Try a proof by contradiction. There are different ways to prove this.

2. *Strong version:* For every compression algorithm A and $n \in \mathbb{N}$ it holds that

$$\left|\{w \in \Sigma^{\leq n} : |A(w)| < |w|\}\right| < \tfrac{1}{2} \cdot \left|\Sigma^{\leq n}\right| .$$

   In words, less than half of all inputs of length at most n can be compressed below their original size.

   Hint: Start by determining $\left|\Sigma^{\leq n}\right| = \left|\Sigma^{0}\right| + \left|\Sigma^{1}\right| + \cdots + |\Sigma^{n}|$.

The theorems hold for every non-unary alphabet, but you can restrict yourself to the binary case, i.e., $\Sigma = \{0, 1\}$.

We denote by $\Sigma^\star$ the set of all (finite) strings over alphabet Σ and by $\Sigma^{\leq n}$ the set of all strings with size $\leq n$. As the domain of (all) compression algorithms, we consider the set of (all) *injective* functions in $\Sigma^\star \to \Sigma^\star$, i.e., functions that map any input string to some output string (encoding), where no two strings map to the same output.