ELEN-0060: Information and Coding Theory

Project 2 - Source coding, data compression and channel coding

Maxime Goffart
180521

Olivier Joris
182113

Academic year 2021 - 2022

1 Source coding and reversible data compression

1.1 Question 5

First, we can compute the marginal probability distribution of all the symbols based on the given Morse text. The distribution is:

Symbol	.	-	_	/
Probability	0.43378	0.28706	0.21452	0.06464

Table 1: Marginal probability distribution of all the symbols

Based on the distribution of probabilities, we can compute the binary Huffman code. We get the following code:

Symbol	.	-	_	/
Huffman code	0	11	101	100

Table 2: Binary Huffman code

By applying the obtained Huffman code to the Morse text, we get the encoded Morse text whose size is 2213141 bits. The original Morse text has a size of 4797160 bits. Thus, we have a compression rate of 2.16758.

1.2 Question 6

We can compute the expected average length of the Huffman code by computing the sum for the 4 symbols of the probability of each symbol times the length of the code associated with the symbol. We get that the expected average length is equal to 1.84538.
By comparing to the empirical average length of the Huffman code, we get:

Length of encoded / length of initial text = 1.84538

This value is the same as the one for the expected average length which is logical since the expected average length was computed based on the probabilities of occurrence of each symbol based on the given Morse text.
By comparing to the theoretical bounds, we get that:

$$\frac{H(S)}{log_2(q)} \leq \overline{n} \leq \frac{H(S)}{log_2(q)} + 1 \text{ because } 1.77138 \leq 1.84538 \leq 2.77138 \tag{1}$$

Thus, our code is optimal, because the inequation is satisfied, which was expected because Huffman codes are optimal. But, the obtained code is not absolutely optimal because $\frac{H(S)}{log_2(q)} \neq \overline{n}$.