PRIORITY QUEUE REPORT

Davide Di Mauro

Politecnico di Torino

Student ID: 306089

s306089@studenti.polito.it

Abstract

The main objective of this homework is to devise an antiplagiarism software specialized on poems utilizing hash tables and fingerprint.

Input Parameters

1. S: The number of words in a sentence
2. ε: The probability of False Positives

Data Structures

1. A set to store all the possible n-word sentences
2. A set to store all the fingerprints of the sentences

Results

Stored data (independently from the data structure)

- 516864 B (0.492919921875 MB)

4-words:

- Total number of sentences: 96120
- Experimental amount of stored data: 7516160 B (7.16796875 MB)
- Memory occupancy of the set of sentences: 11710680 B (11.168174743652344 MB)

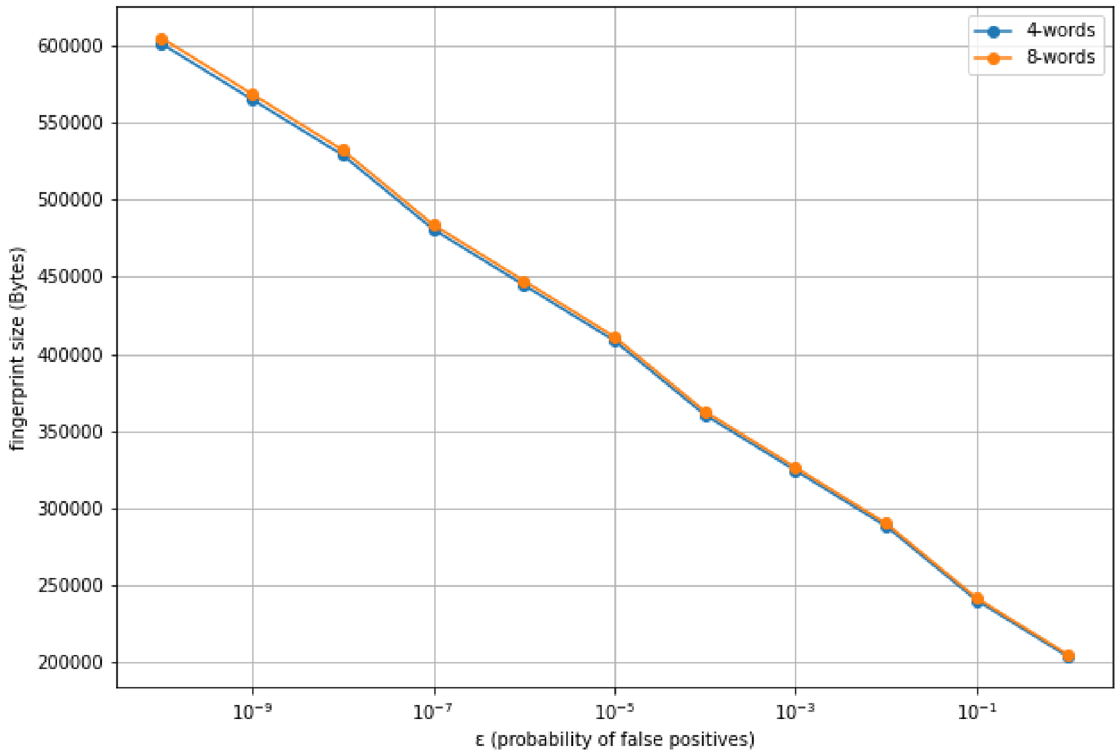
8-words:

- Total number of sentences: 96715
- Experimental amount of stored data: 10001896 B (9.538551330566406 MB)
- Memory occupancy of the set of sentences: 14196416 B (13.53875732421875 MB)

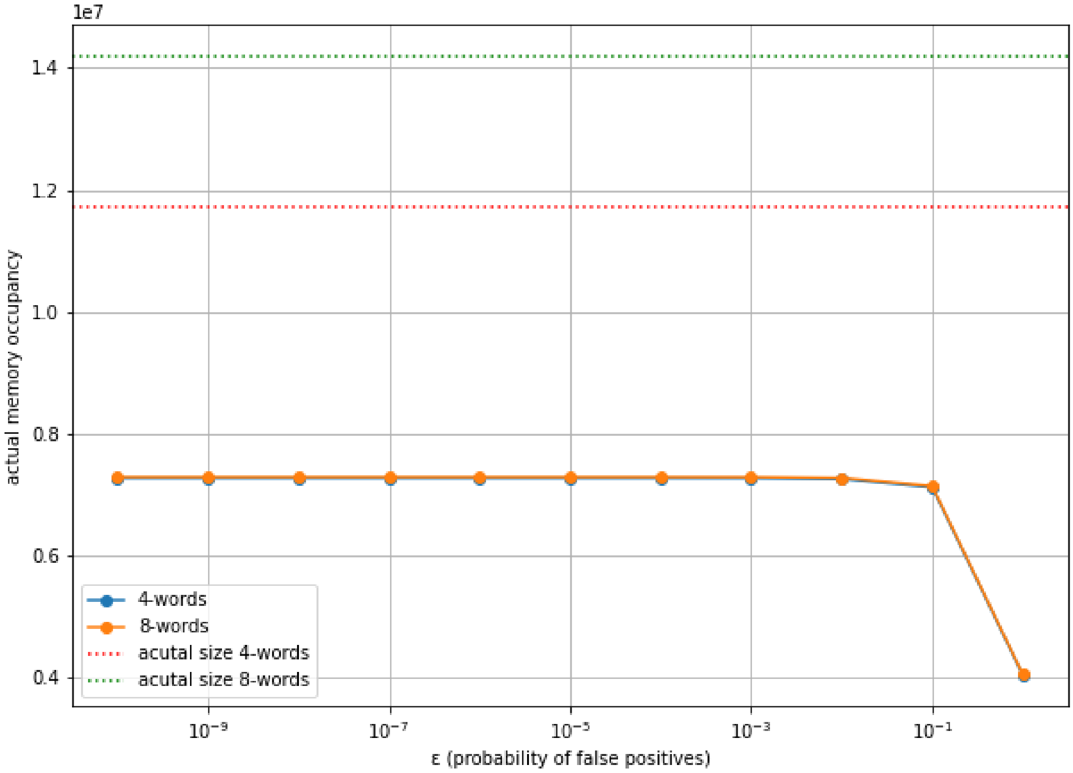
In order to find the number of bits used for each fingerprint the following formula is used:

$$b \geq \log_2 \frac{m}{\varepsilon}$$

Where b is the number of bits, m is the number of entries we want to store and ε is the maximum probability of false positives.



This graph shows the size of the fingerprints in Bytes ($\frac{b \cdot m}{8}$) in function of ε. As expected, increasing the probability of false positives decreases the number of bits used for each fingerprint, thus reducing the memory occupancy.



This graph shows the experimental amount of data used to store the set with the fingerprints in function of the probability of false positives, and compares it to the amount of data used to store the set of 4-words and 8-words sentences. **Since the hashes are converted to an integer value stored in a 32bit variable, the size of the set of fingerprints is always more memory efficient.**

Conclusions

In this specific case the under all conditions fingerprinting allows to reduce the actual amount of memory. **In general this result is independent of S since the hash generated is independent from the size of the value of which we compute the fingerprint.**