INSTITUTO SUPERIOR DE ENGENHARIA DE LISBOA
Engenharia Informática e de Computadores

Compressão e Codificação de Dados

Inverno 2020/2021

Trabalho prático I

1 – Entendendo o conteúdo de um ficheiro como uma sequência de símbolos (bytes) produzida por uma fonte discreta com alfabeto {0,1,…,255}, desenvolva uma aplicação que recebendo o nome do ficheiro:

a) Estime a entropia da fonte assumindo que esta não tem memória (estimativa de primeira ordem) e determine o número de símbolos diferentes e os cinco mais frequentes;
b) Determine o menor número de símbolos que faz com a frequência de ocorrência de símbolos desse subconjunto seja igual ou superior à dos restantes símbolos;
c) Admitindo codificação de cada símbolo com $\lceil \log_2(1/P(a_i)) \rceil$ bit, determine a compressão obtida;
d) Com a aplicação desenvolvida analise o conjunto de ficheiros do corpus e outros.

2 – Para cada ficheiro, admitindo modelo de Markov de 1.ª ordem, estime a entropia. Analise os ficheiros do corpus e compare com os valores da entropia estimada no pressuposto de que a fonte não tem memória.

3 – Considere o método de geração de sequências proposto por Claude Shannon:

To construct "second-order approximation, i. e., first-order Markov chain," for example, one opens a book at random and selects a letter at random on the page. This letter is recorded. The book is then opened to another page and one reads until this letter is encountered. The succeeding letter is then recorded. Turning to another page this second letter is searched for and the succeeding letter recorded, etc. A similar process was used for "high-order approximations".

a) Adaptando a ideia substituindo o livro por um ficheiro, implemente gerador de sequências para, pelo menos, aproximações de primeira e de segunda ordem;
b) Para cada ficheiro, compare a estimativa da entropia do ficheiro com a estimativa da entropia da sequência gerada com base nesse ficheiro;
c) Usando uma aplicação de compressão, para cada ficheiro, compare a compressão do ficheiro com as de sequências geradas com o modelo com diferentes ordens e comente os resultados obtidos.

4 – Considere sequências de números obtidas com lançamentos de dois dados. O primeiro lançamento estabelece o símbolo do alfabeto {1, 2, 3, 4, 5, 6} e o segundo lançamento o número de vezes que esse símbolo se repete. Implemente gerador de sequências com esta estrutura e analise a compressão obtida usando uma aplicação de compressão. Comente os resultados de compressão e as estimativas da entropia.

5 – Considere uma fonte com memória (modelo de Markov de 1.ª ordem), alfabeto {0,1,…,255} e probabilidades de transição P((n-1) mod 256|n)=P(n|n)=P((n+1) mod 256|n)=p/3 e P((n-2) mod 256|n)=P((n+2) mod 256|n)=(1-p)/2. Determine a entropia da fonte. Gere sequências de acordo com o modelo e analise, para valores de p entre 0 e 1, a compressão obtida com uma aplicação de compressão.

6 – Considerando alfabeto com seis símbolos:

a) Caracterize as árvores binárias que estão na origem de códigos de Huffman canónicos para esse alfabeto;
b) Para cada árvore, admitindo que o código é ideal, determine o comprimento médio do código.

7 – Considere apostas do euro milhões codificadas em grupos de 7 bytes (5 com valores de 1 a 50, sem repetição, e os 2 últimos com valores de 1 a 12 sem repetição).

a) Implemente um gerador de apostas (simulador de fonte) e analise a compressão de sequências de apostas usando uma aplicação de compressão;
b) Proponha uma técnica de codificação mais eficiente e determine a sua redundância. Compare essa técnica com a inicial e com os resultados obtidos com a aplicação de compressão.

8 – Sem prejuízo da definição de outras, defina as medidas de compressão bit/byte, fator de compressão e razão de compressão. Usando essas medidas, avalie a compressão dos ficheiros do corpus com os algoritmos DEFLATE (por exemplo, https://zlib.net/) e zstandard (https://github.com/facebook/zstd/releases).

a) Comente os resultados e compare com as estimativas da entropia obtidas para cada ficheiro;
b) Analise a evolução da compressão à medida que vai sendo codificada a sequência (ficheiro) e compare com o que ocorre com sequências obtidas com o gerador implementado em (3) e esse ficheiro.

ISEL, 3 de novembro de 2020