UNIVERSIDADE FEDERAL DE JUIZ DE FORA

Estrutura de Dados II

Relatório Referente ao trabalho 3:
Compressão de Tweet
- Huffman
- LZ77
- LZ78
- LZW

Bruno Carvalho
Diogo Destefano
Pedro Bellotti
Rafael Terra

04 de Dezembro de 2017

Índice

1 – Introdução

Desenvolvido em linguagem C++, o projeto tem como objetivo analisar e comparar os diferentes algoritmos de compressão de dados utilizando-se métodos distintos, a fim de se concluir quais métodos se apresentam mais eficientes em determinados contextos.

Variáveis como número de entradas, tamanho em disco, tempo de execução e taxa de compressão foram utilizadas para definir e comparar os resultados dentre os algoritmos testados.

O projeto foi desenvolvido visando consumir uma quantidade reduzida de espaço em disco, visto que por vezes, os algoritmos foram testados com até 1.000.000 entradas. A cada iteração de testes, o número de entrada foi aumentando, de forma que houveram 5 iterações para cada conjunto de dados de tamanho variando entre 1000 e 1.000.000. Os resultados foram salvos, as estruturas foram desalocadas e os dados compressos foram salvos em um arquivo de texto, para que o gasto de memória seja controlado. A cada iteração, o conjunto de dados foi *randomizado*, a fim de se obter um resultado mais confiável (complexidade de randomização: O(n)).

Para facilitação no uso dos métodos de compressão, todos os tweets tiveram seus caracteres maiúsculos trocados por minúsculos e tiveram também seus caracteres especiais removidos. Após isso, todos os tweets foram colocados em uma única string e esta foi utilizada na compressão.

O software utiliza o console para controle de execução. Ao executar o código, o menu é impresso em tela, e ao escolher uma opção de compressão, o software realiza os testes necessários do método escolhido, salvando os resultados nos arquivos TXT respectivos a cada tipo de teste.

Infelizmente, o software possui a implementação de apenas os métodos de compressão Huffman e LZ77, pois os integrantes responsáveis pela implementação dos métodos LZ78 e LZW não conseguiram terminar os mesmos a tempo para entrega do trabalho.

2 – Dados Brutos

-Análise dos Algoritmos

As tabelas a seguir contém os dados obtidos na compressão dos tweets usando dois diferentes métodos (Huffman e LZ77). As métricas tempo gasto, tamanho do arquivo e taxa de compressão foram utilizadas. Lembrando que tais valores foram obtidos através da média entre 5 execuções para cada N (onde N é o tamanho do conjunto de dados).

Os arquivos de saída com os resultados dos testes estão anexados ao trabalho.

Algoritmo de Huffman	Tamanho do arquivo antes da compressao	Tamanho do arquivo apos a compressao	Tempo gasto	Taxa de compressao
N = 1000	69,187	31,349	0.147	~45%
N = 5000	350,247	158,903	0.749	~45%
N = 10000	696,590	315,921	1.677	~45%
N = 50000	3,530,750	1,598,910	7.435	~45%
N = 100000	7,025,000	3,182,490	14.354	~45%
N = 500000	35,833,200	16,244,700	74.357	~45%
N = 1000000	71,285,900	32,314,200	144.259	~45%

Algoritmo LZ77	Tamanho do arquivo antes da compressao	Tamanho do arquivo apos a compressao	Taxa de compressao
N = 1000	69,187	57,394	~17%
N = 5000	350,247	294,205	~16%
N = 10000	696,590	585,130	~16%
N = 50000	3,530,750	3,001,133	~15%
N = 100000	7,025,000	6,041,500	~14%
N = 500000	35,833,200	31,533,210	~12%
N = 1000000	71,285,900	62,731,596	~12%

3 – Análises dos Resultados

3.1. Huffman

O método Huffman apresentou-se estável e rápido. Observou-se uma taxa de compressão constante de aproximadamente 45% em todos os casos e o tempo gasto, mesmo no pior caso, não foi tão grande. Talvez isso se deva ao fato do método Huffman estático codificar apenas uma string (e, aproveitando o fato de que todos os tweets foram salvos em uma única string e então compressos, o método estático foi escolhido), podendo construir uma tabela e códigos precisa para cada entrada. A desvantagem deste método é a limitação de apenas uma string, não sendo possível fazer novas inserções na tabela de códigos após ela já ter sido criada.

3.2. LZ77

O método LZ77 foi um pouco menos estável do que o método Huffman. A taxa de compressão variou durante os casos, de forma que a taxa acabou sendo inversamente proporcional ao tamanho do arquivo de entrada (ou seja, quanto maior o arquivo, menor a compressão). Houve um problema na execução do cálculo da média do tempo gasto e, portanto, esta métrica não pode ser usada como comparação nos resultados.

3.3. LZ78

O método LZ78 não foi implementado.

3.3. LZW

O método LZW não foi implementado.

4 – Dados Sobre o Desenvolvimento

4.1 – Hardware e Software Utilizado

O projeto foi executado em um computador com Core i5 de 3.2GHz, com 16GB de memória RAM, em um sistema operacional Windows 10 de 64bits utilizando as IDEs Visual Studio 2017 e Code Blocks (em linguagem C++). Para a gestão e controle do projeto entre o grupo, a plataforma GitHub foi utilizada.

4.2 – Divisão de tarefas entre o Grupo

Rafael Terra: Método LZ77 e auxílio na produção do relatório.

Pedro Bellotti: Adaptações de funções no main, método Huffman, criação de tabelas de resultados, auxílio na produção o relatório, aplicação dos testes e extração dos dados de saída.

Bruno Carvalho: Método LZ78.

Diogo Destefano: Método LZW.