Huffman Algorithm for Data Compression

N. Cortez, A. Martinez, J. Mota, J. Westcott

May 28, 2015

Overview

History

In 1938, Claude Shannon solidified the branch of information theory through his groundbreaking master's thesis, *A Symbolic Analysis of Relay and Switching Circuits*. In his thesis, Shannon demonstrated how to measure information by instituting a relationship between symbolic logic and relay circuits. Moreover, his thesis established the concept of information entropy, measuring the information gained from observing a random variable.

History

One application of information theory is data compression. David Huffman, an MIT graduate, was given the option of taking a final exam or writing a research paper on an efficient binary compression code as a final project in one of his graduate courses. Huffman chose the latter and created what was to be known as the Huffman Algorithm. His algorithm addressed the issue of how to reduce redundancy in an input message by encoding it in as few bits, or pieces of information, as possible.

What is Data Compression?

Definition (Data Compression)

The technique in information theory by which the same amount of data is transmitted via a smaller number of bits. This is useful for truncating information and transporting it near real time.

What is Data Compression?

Definition (Data Compression)

The technique in information theory by which the same amount of data is transmitted via a smaller number of bits. This is useful for truncating information and transporting it near real time.

Lossless	Lossy
PNG	JPEG
FLAC	MP3
QuickTime Animation	MPEG

Probability Theory: The Basics

In order to understand data compression one must have a foundation in elementary probability theory. Note that the probability of the values of a random variable will satisfy a probability distribution.

Probability Theory: The Basics

In order to understand data compression one must have a foundation in elementary probability theory. Note that the probability of the values of a random variable will satisfy a probability distribution.

Definition (Random Variable)

A variable whose value is subject to variations due to chance.

Probability Theory: The Basics

In order to understand data compression one must have a foundation in elementary probability theory. Note that the probability of the values of a random variable will satisfy a probability distribution.

Definition (Random Variable)

A variable whose value is subject to variations due to chance.

Definition (Expected Value)

The weighted sum of the possible values of a random variable with their respective probabilities so that:

$$E[x] = \sum_{i=1}^{n} x_i P(x_i).$$

Entropy

Definition (Entropy)

The expected value of the information content, $-log(P(X))$, of a random variable found by:

$$E\left[-log(P(X))\right] = \sum_{i=1}^{n} p_i(-log(p_i)),$$

where the random variable X takes values $x_1, x_2, \ldots, x_n$ with associated probabilities $p_1, p_2, \ldots, p_n$.

Entropy

Definition (Entropy)

The expected value of the information content, $-log(P(X))$, of a random variable found by:

$$E\left[-log(P(X))\right] = \sum_{i=1}^{n} p_i(-log(p_i)),$$

where the random variable X takes values $x_1, x_2, \ldots, x_n$ with associated probabilities $p_1, p_2, \ldots, p_n$.

Example:

Fair Coin (E=1)

x	p(x)
Heads	1/2
Tails	1/2

Biased Coin (E=0)

x	p(x)
Heads	1
Tails	0

Entropy: Relevancy

Theorem (Shannon's Coding Theorem)

Let X be a random variable with n possible letters and entropy: $H(X) = \sum_{i=1}^{n} p_i(-log(p_i))$. Let L be the average number of bits to encode N symbols selected randomly with X. Then for the optimal coding:

$$H(X) \leq L < H(X) + \frac{1}{N}$$

This theorem establishes the relationship between the entropy and the average length of code. Shannon proved that the entropy is the best average length of code that could possibly be reached. He also demonstrated that one may get arbitrarily close to the entropy and so this establishes a lower bound on how efficient lossless codes may be.

Huffman Algorithm

Definition (The Huffman Algorithm)

A compression algorithm that allows us to give letters with lower probability longer code words and letters with higher probability shorter code words.

David Huffman showed that you can find a code for a given text whose average length can get close to the entropy without losing information.

Huffman Algorithm: Example

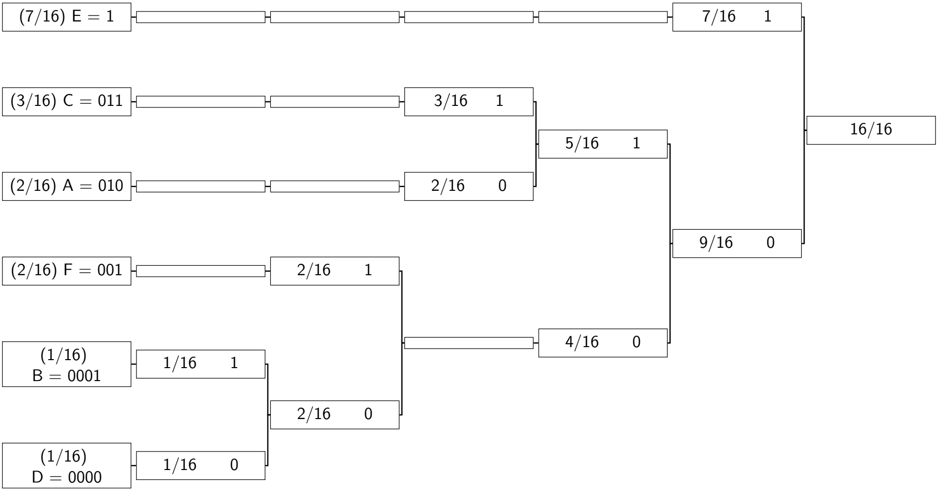
Text: ACCDEEEFAFECEEEB

Huffman Algorithm: Example

Text: ACCDEEEFAFECEEEB

x	p(x)
A	2/16
B	1/16
C	3/16
D	1/16
E	7/16
F	2/16

Huffman Algorithm: Example



What We Did

We made our own implementation of Huffman's Algorithm in a C++ program. We then tested this on different types of files composed of a variety of sizes.

Analysis of Results

	Makefile	Small Text	Circle Image
Original Size (Bytes)	343	20	52958
Compressed Size (Bytes)	291	41	46059
Average Word Size	4.5889	3.85	6.8748
Entropy	4.5666	3.7842	6.8468
Min. Possible Size (Bytes)	195.79	9.4605	45323

	Source Code	Scream Audio	Large Text
Original Size (Bytes)	15249	1004656	316794
Compressed Size (Bytes)	8586	851056	182234
Average Word Size	4.3965	6.7725	4.5966
Entropy	4.3375	6.7304	4.5653
Min. Possible Size (Bytes)	8267.9	845216	180782

References

D. Salomon, *Data Compression: The Complete Reference*, Springer, New York, NY, USA, 4th edition, 2007.

D. A. Huffman, "A Method for the Construction of Minimum Redundancy Codes," *Proceedings of the Institute of Radio Engineers*, vol. 40, no. 9, pp. 1098-1101, 1952.

Lee, Joseph, "Huffman Data Compression," *MIT Undergraduate Journal of Mathematics*, May 23, 2007.

Shannon, C. E. & Weaver, W., *The Mathematical Theory of Communication*. Urbana, IL: University of Illinois Press, 1949.

Thank you!