

Synopsis

Title

Data compression using Huffman Coding

Team members

1. Pranshu Srivastava - 9917103250 - F8
2. Subhradip Mukherjee - 9917103234 - F7
3. Prashant Dixit - 9917103209 - F7
4. Karan Dev Tyagi - 9917103081 - F3

Brief Introduction

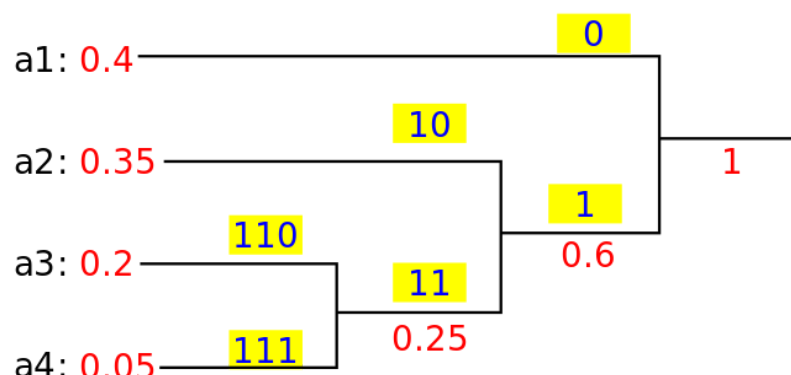
Huffman Coding is a way to assign binary codes to symbols that reduce the overall number of bits used to encode a typical string of those symbols. It is a technique of compressing data so as to reduce its size without losing any of the details and was first developed by David Huffman. Huffman Coding is generally useful to compress the data in which there are frequently occurring characters. Huffman Coding is a famous Greedy Algorithm. It provides a lossless compression method for the data and uses a variable-length encoding.

Main Algorithm

The Huffman coding scheme takes each symbol and its weight (or frequency of occurrence) and generates proper encodings for each symbol taking account of the weights of each symbol so that higher weighted symbols have fewer bits in their encoding.

Encoding

1. Create a leaf node for each symbol and add it to the priority queue.
2. While there is more than one node in the queue:
 - A. Remove the node of the highest priority (lowest probability) twice to get two nodes.
 - B. Create a new internal node with these two nodes as children and with probability equal to the sum of the two nodes' probabilities.
 - C. Add the new node to the queue.
3. The remaining node is the root node and the tree is complete.



Decoding

1. We start from the root and do the following until a leaf is found.
2. If the current bit is 0, we move to the left node of the tree.
3. If the bit is 1, we move to the right node of the tree.
4. If during traversal, we encounter a leaf node, we print the character of that particular leaf node and then again continue the iteration of the encoded data starting from step 1.