

Evolving hash table algorithm conception for innovating data compression algorithms

Deepanker Shukla

Independent Researcher

incdsg@yahoo.com

ORCID 0000-0002-3870-0761

Abstract

The author presents a theoretical method based on an assumption to use an evolving hash table for data compression.

Keywords

Hash table, Data compression, Algorithm Conception, Flow Chart.

Theory

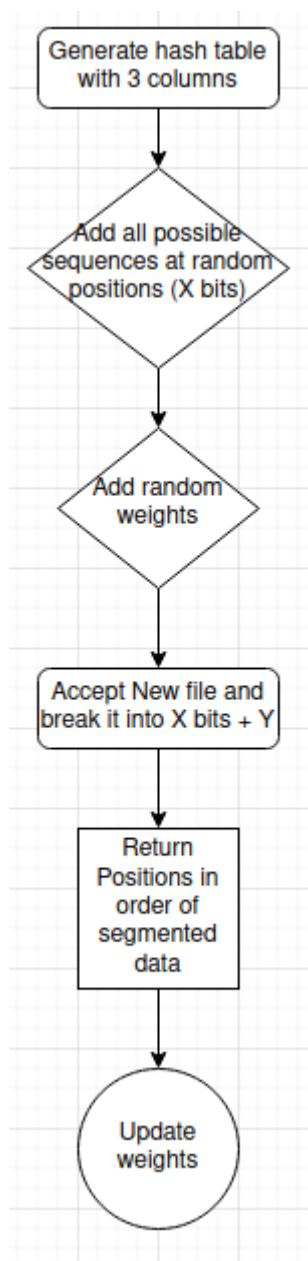
A hash table with 3 coloumns consists of all the combinations of x bits (consisting of 2 to the power of x bits).

	A	B
1	0b1000100011001011010110101001010100100110	0.4908
2	0b1010000011100100001100011011100000001110	0.55326
3	0b1111110010110001111100010101111001001111	1.144776
4	0b1110100000100000010110111001101000010001	1.806654
5	0b1010000110011100111011100101110010000010	1.937119
6	0b1011001101011000100000111101100011010100	1.876058
7	0b1111100110100011000000110101111101001010	0.085043
8	0b1100011101111001010010000000001011110001	1.80636
9	0b1110010110101010011100101111101101100100	0.594777
10	0b1110100011100010110011010000110111001001	1.390849
11	0b1110000010000111100111011010011110010111	1.644609
12	0b1111000000100001111100110100001011000101	0.94201
13	0b1001001110000111101001111010111011101111	0.636841
14	0b1101001101000011111110111101101001010011	0.624778
15		

Figure 1

As depicted in the figure 1 above, possible sequences have been initialised at random position with random weights in the hash table. The algorithm takes files from users, breaks it into sequences of x bits and gives back the position of the bit in the hash table and adds weight to the sequences according to their occurrence in the given file. Thus creating a base social experiment to innovate data compression algorithms upon using heighest weighed sequences bits.

Algorithm Flow Diagram



Remarks

* X depends upon the capabilities of the server storage and relation between indexed sequences and their base 10 counterparts and should be kept as high as possible for indexing as much bits as possible for delivering “range” of the files in use.

* This algorithm has not been implemented in any coding language and has never been tested through confusion tests.

Conclusion

This algorithm although being completely theoretical at the moment does signifies the vast amount of digital storages possible and in use, and can levy neural network architectures (similar to deep convolution and/or autoencoders) to innovate new data compression algorithms.