Comparative study of Shannon-Fanon and Huffman Algorithm

Okunade Temilola & Idris Abiodun Aremu
Computer Science Department
School of Technology
Lagos State Polytechnics, Lagos
E-mail: aremu.i@mylaspotech.edu.ng; taokunade@yahoo.com
Phone: +2348025273062/+2349061357124. +2348054303256

ABSTRACT

Data in a compact form is essential to reduce storage space and redundancies, the science and art of compressing data is called data compression. Data compression is applied to most computer applications areas nowadays. There are lots of data compression algorithms which are available to compress files of different formats in computer system that can be categorized as lossy and lossless. Lossless data compression recreates the exact original data from the compressed data while lossy data compression cannot regenerate the perfect original data from the compressed data. In this paper the survey of different loss less compression algorithms was analyzed using entropy of the source. The statistical coding techniques used are Shannon-Fanon Coding and Huffman coding. Shannon Fanon has the maximum information rate at which steady communication passing through a communication channel with less error probability as compared to Huffman coding

Keywords: Shannon Fano coding, Huffman coding, Lossless compression, Lossy compression, and Data compression

iSTEAMS Conference Proceedings Paper Citation Format
Okunade Temilola & Idris Abiodun Aremu (2018) Comparative Study of Shannon-Fanon and Huffman Algorithm. Proceedings of the 14th iSTEAMS International Multidisciplinary Conference, AlHikmah University, Ilorin, Nigeria, Vol. 14, Pp 67-76

1. INTRODUCTION

In information theory, the noisy-channel coding theorem (sometimes Shannon's theorem), establishes that for any given degree of noise contamination of a communication channel, it is possible to communicate discrete data (digital information) nearly error-free up to a computable maximum rate through the channel. Shannon's theorem is an essential theorem in forward error correction, and depict maximum information rate at which dependable communication is possible over a channel that has a certain error probability or signal-to-noise ratio (SNR). Data compression is the art of representing information in compact form [1]. Data transmission is faster in data communication since the file size has been shrink which in turn reduces the required storage space. Compression techniques solve the problem of redundancies by looking for redundant data in data communication. Data compression can be divided into two broad c lasses: lossless data compression and lossy data compression. In lossless compression, the exact original data can be recovered from compressed data. It is used when the ference between original data and decompressed data cannot be tolerated. Medical images, text needed in legal purposes and computer executable files are compressed using lossless compression techniques. Lossy compression, as the name suggests, involves loss of information. It is used in the applications where the lack of reconstruction is not an issue.

Videos and audios are compressed using lossy compression. The extremely fast growth of data that needs to be stored and transferred has given rise to the demands of better transmission and storage techniques. Various lossless data compression algorithms have been proposed and used. Huffman Coding, Arithmetic Coding, Shannon Fano Algorithm, Run Length Encoding Algorithm are some of the techniques in use. In this paper we would like to discuss Shannon-Fano coding and Huffman coding. Lossless data Compression with appropriate example and compare their performance with Compression Ratio and the entropy of the source.

The rest of this paper is organised as follows. Section II discuss the literature review of data compression Section III discuss error correcting using Shannon Fano theorem, Sections IV data compression methods was discussed, V Comparative analysis between Shannon-Fano and Huffman Coding was discussed, V I concludes the paper.

2. LITERATURE SURVEY

The new compression technique that uses referencing through two-byte numbers (indices) for the purpose of encoding has been presented by U. Khuranaand and A. Koul in 2015. Advanced Bit Reduction Algorithm for lossless text data compression was also presented by A. Kaur and N. S. Sethi [2]. The technique is competent in given high compression ratios and faster search through the text. Also S. Kaur and V.S. Verma implemented LZW data compression algorithm by finite state machine [3], which efficiently compress the text data, the work of R. S. Brar and B. Singh in describing a survey of different basic lossless and lossy data compression techniques. On the basis of these techniques a bit reduction algorithm for compression of text data has been proposed by the authors based on number theory system and file differential technique which is a simple compression and decompression technique free from time complexity.

Future work can be done on coding of special characters which are not specified on key-board to revise better results In 2013, also In 2013, S. Porwal et. al, explained on lossless data compression methodologies and compares their performance. Huffman and arithmetic coding are compared according to their performances. In this paper the author has found that arithmetic encoding methodology is powerful as compared to Huffman encoding methodology. By comparing the two techniques the author has concluded that the compression ratio of arithmetic encoding is better and furthermore arithmetic encoding reduces channel bandwidth and transmission time also. In 2012 Mahmud, Salauddin (March 2012) proposed compression technique can be useful to send a lot of data in wire and wireless network. In 2011, S. Shanmugasundaram and R. Lourdusamy**,** provides a survey of different basic lossless data compression algorithms.

Experimental results and comparisons of the lossless compression algorithms using Statistical compression techniques and Dictionary based compression techniques were performed on text data. Among the statistical coding techniques the algorithms such as Shannon-Fano Coding, Huffman coding, Adaptive Huffman coding, Run Length Encoding and Arithmetic coding are considered. A set of interesting conclusions are derived on their basis. Lossy algorithms achieve better compression effectiveness than lossless algorithms, but lossy compression is limited to audio, images, and video, where some loss is acceptable. The question of the better technique of the two, “lossless” or “lossy” is pointless as each has its own uses with lossless techniques better in some cases and lossy technique better in others. In 2010, Md. Rubaiyat Hasan introduced a method and system for transmitting a digital image (i.e., an array of pixels) from a digital data source to a digital data receiver. More the size of the data be smaller, it provides better transmission speed and saves time. In this communication we always want to transmit data efficiently and noise free.

3. ERROR CORRECTING USING SHANNON THEOREM

The theorem describes the maximum possible efficiency of error-correcting methods with levels of noise interference and data corruption. Shannon's theorem has wide-range of applications in both communications and data storage. This theorem is fundamental importance to the modern field of information communications. Shannon only gave an outline of the proof. The first rigorous proof for the discrete case is due to Amiel Feinstein in 1954. The Shannon theorem states that given a noisy channel with channel capacity *C* and information transmitted at a rate *R*, if there exist codes that allow the probability of error at the receiver to be made arbitrarily small. This means that, theoretically, it is possible to transmit information nearly without error at any rate below a limiting rate, *C*.

The converse is also important. If an arbitrarily small probability of error is not achievable. All codes will have a probability of error greater than a certain positive minimal level, and this level increases as the rate increases. So, information cannot be guaranteed to be transmitted reliably across a channel at rates beyond the channel capacity. The theorem does not support the rare situation in which rate and capacity are equal. The channel capacity C can be calculated from the physical properties of a channel; for a band-limited channel with Gaussian noise, using the Shannon–Hartley theorem. Simple schemes can be adopted by "sending the message 3 times and use a best 2 out of 3 voting scheme if the copies differ" are inefficient error-correction methods, unable to asymptotically guarantee that a block of data can be communicated free of error.

Advanced techniques such as Reed–Solomon codes and, more recently, low-density parity-check (LDPC) codes and turbo codes, come much closer to reaching the theoretical Shannon limit, but at a cost of high computational complexity. Using these highly efficient codes and with the computing power in today's digital signal processors, it is now possible to reach very close to the Shannon limit. In fact, it was shown that LDPC codes can reach within 0.0045 dB of the Shannon limit (for binary AWGN channels, with very long block lengths).

a_i	p(a_i)	1	2	3	4	Code
a_1	0.36	0	00			00
a_2	0.18		01			01
a_3	0.18	1	10			10
a_4	0.12		11	110		110
a_5	0.09			111	1110	1110
a_6	0.07				1111	1111

Fig. 1.0 Shannon fanon error encoding and decoding

4. DATA COMPRESSION METHODS

Compression techniques can be used in a broad range of technologies like software applications storage system, database and operating system. It is useful to reduce the redundancy in data representation thus increasing effective data density [4]. The two basic classes of data compression are applied in different areas. One of these is lossy data compression, which is widely used to compress image data files for communication or archives purposes [5]. The other is lossless data compression which is the scope of this paper, which is commonly used to transmit or archive text or binary files required to keep their information intact at any time. The main purposes of this paper to shows the lossless compression techniques and their comparative study.

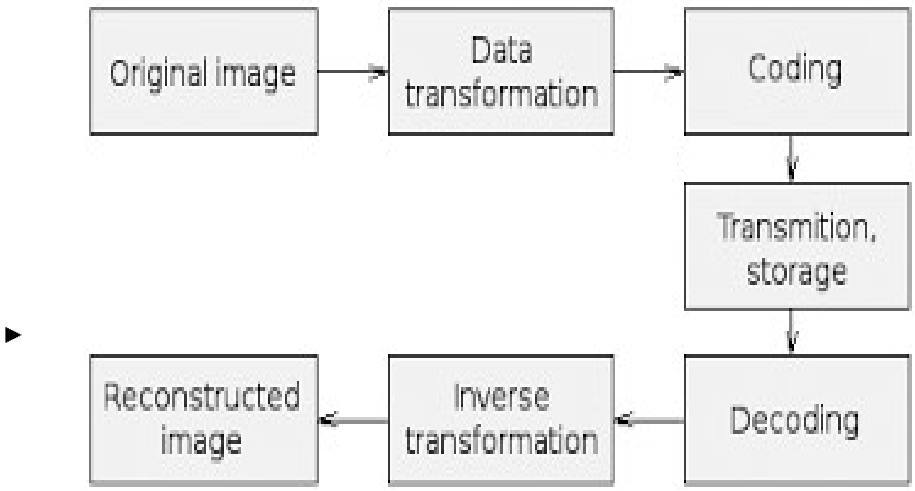


Fig. 2.0 Data Compression and Decompression

4.1 SHANNON-FANO

Shannon – Fano algorithm was consecutively developed by Claude Shannon and R.M. Fano in 1949. It can be apply in information encoding and decoding in information communication depending upon their probabilities [6]. It allocates less number of bits for highly probable messages and more number of bits for unusual occurring messages. The algorithm of Shannon fano is as follows

1. Frequency probability table is developed for a given list of symbols.
2. Keep the most frequently occurring symbols at the top of the table after sorting the symbols
3. The table can be divided into two halves with the total frequency of the upper half being as close to the total frequency of the bottom half as possible.
4. Assign digit '0' for the upper half of the list and the lower half digit '1'.
5. Recursively apply the steps 3 and 4 to each of the two halves, subdividing groups and adding bits to the codes until each symbol has become a corresponding leaf on the tree

Shannon-Fano coding are pleasing to analysed and display, because the code is self separating, Huffman coding are more complicate in creation and they are more not obviously self separated. Shannon – Fano algorithm is more efficient than the Huffman coding when the probabilities are closer to inverses of powers of 2.

4.2 Shannon Fano Tree Representation

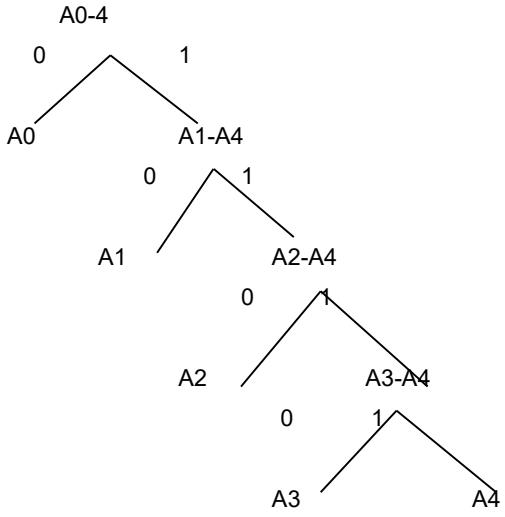


Fig. 3.0 Shannon Fano tree

4.3 Huffman code steps are as follows

1. The symbols or characters are sorted based on their probabilities of descending
2. If the probability is equal, arrange the index of symbols or character descending as well.
3. The symbols with the smallest probability is considered and the upper symbol is given the '1' bit, the symbol under bit '0', merge into new symbol, and sum up the probability
4.The symbols repeated again like the first step
5. If the probability is the same, the latest symbol is under the old symbol
6. Then repeat steps 2 and 3 repeatedly until the probability sum = 1.0
7. Then specify the code words of each symbol with binary

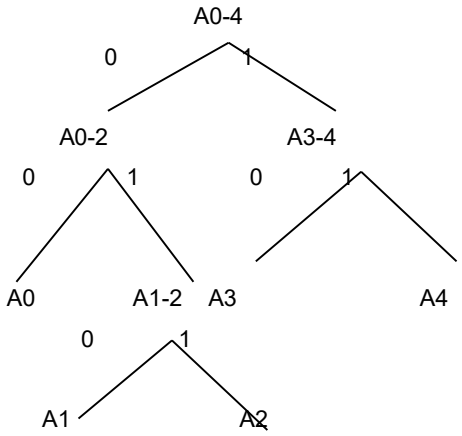


Fig. 4.0 Huffman tree

4.4 LEMPEL ZEV WELCH

Lempel Zev Welch uses the index send to the dictionary. LZW is a Dictionary based compression algorithms and it uses dictionary instead of a statistical model [9]. A dictionary is a lay down of probable words of a language, and is stored in a table like structure and used the indexes of entries to represent larger and repeating dictionary words. The LZW algorithm is one of such algorithms. The technique used a dictionary is to store and index the previously seen string patterns. The index values are used instead of repeating string patterns in the compression routine. The dictionary is created dynamically in the compression process and no need to transfer it with the encoded message for decompressing. In the decompression process, the same dictionary is created dynamically. Therefore, this algorithm is an adaptive compression algorithm

Algorithm

Step 1: The input of first byte is accepted and stored as string

Step 2: Check if the input string is available in the dictionary or not.

Step 3: Then add the string to the dictionary if is not available

Step 4: Add character to the string until the last characters, if it is available

Step 5: Accept the next character and Output the string.

Step 6: Repeat the routine until the character is null.

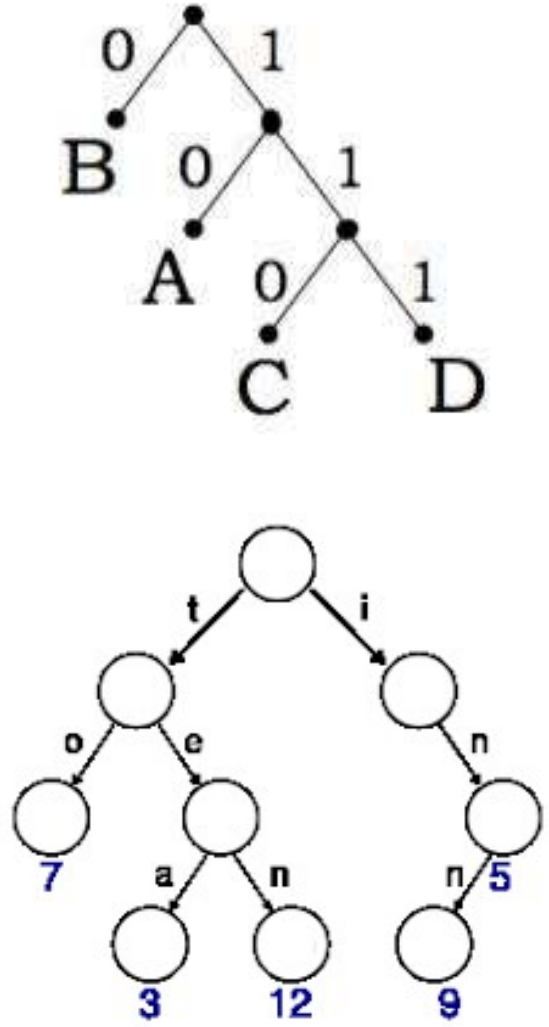


Fig. 5.0 lempel zev Welch diagram

5. **ARITHMETIC CODING**

Arithmetic Coding is functional for small alphabets with highly skewed probabilities. Code word is not used to symbolised text in this method, but it is use to construct a code for the complete message. Arithmetic Coding allocate an interval to each symbol [10]. A decimal number is then allocated to this interval. With an interval of [0, 1] at an initial. A message is represented by a half open interval [x, y] where x and y are real numbers between 0 and 1. The interval is then divided into sub-intervals. The number of sub-intervals is equal to the number of symbols in the existing set of symbols and size is proportional to their probability of appearance. For each symbol a new internal division takes place based on the last sub interval. Consider an example illustrating encoding in Arithmetic Coding.

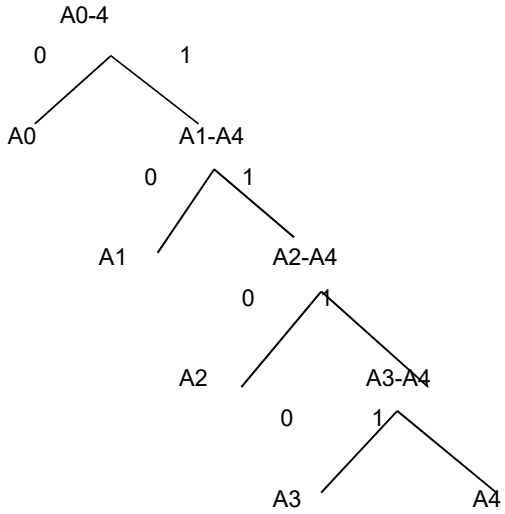
5.1 Comparative Analysis of Huffman and Shannon Fano algorithm

Performance measure is use to find which technique is good according to some criteria. Depending on the nature of application there are various criteria to measure the performance of compression algorithm. When measuring the performance one of the main thing to be considered are space capacity (entropy) and the time efficiency [11]. Since the compression behaviors depend on the redundancy of symbols in the source file, it is difficult to measure performance of compression algorithm in general. The performance of data compression depends on the type and structure of input source. The compression behavior depends on the category of the compression algorithm: lossy or lossless. Following are some measurements use to calculate the performance of lossless algorithms [12].

5.2 Numerical Example

The source of information A's symbols generates the following A0, A1, A2, A3 and A4 with the corresponding probabilities 0.4, 0.3, 0.15, 0.1 and 0.05 respectively. Encoding the symbols using binary encoder for Shannon-Fano and Huffman encoder gives the following.

Shannon Fano Tree representation



Huffman Tree representation

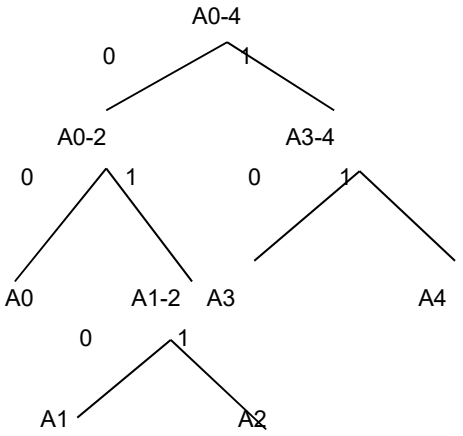


Table 1: Performance evaluation of Shannon Fano and Huffman coding

SYMBOLS	PROBABILITY	BINARY CODE	SHANNON FANO	HUFFMAN
A0	0.4	000	0	00
A1	0.3	001	10	010
A2	0.15	010	110	011
A3	0.1	011	1110	10
A4	0.05	100	1111	11
			2.05	**2.45**

From the above Shannon Fano tree we obtain

A0 = 0

A1 = 10

A2 = 110

A3 = 1110

A4 = 1111

And from Huffman coding we obtain

A0 = 00

A1 = 10

A2 = 110

A3 = 1110

A4 = 1111

The entropy of the above are **Shannon Fano**

E= $\sum_i^n p_i \ l_i \ where \ \ p_i \ is \ the \ probalility \ of \ occurence \ and \ l_i \ is \ the \ binary \ bit$

Hence E = 0.4 *1 + 0.3 * 2 + 0. 15 * 3 + 0.15 * 3 + 0.1 * 4 + 0.05 + 4

= 0.4 + 0.2 + 0.45 + 0.4 + 0.2 = 2.05

The efficiency of the binary code = $\frac{2.05}{3}$ = 0.68333333 = 68.3%

Entropy for Huffman

E = 0.4 *2 + 0.3 * 2 + 0. 15 * 3 + 0.15 * 3 + 0.1 * 4 + 0.05 + 4

= 0.8+ 0.2 + 0.45 + 0.4 + 0.2 = 2.45

The efficiency of the binary code = $\frac{2.45}{3}$ = 0.816666667 = 81.6%

6. CONCLUSION

In this paper, we compare Huffman coding and Shannon-Fano coding techniques of data compression on selected words in terms of compression size and compression ratio. After testing those algorithms, Shannon-Fano coding has efficiency of 68.3% whilst Huffman is 81.6%. Hence, Shannon Fano coding methodologies is very powerful compare to Huffman encoding. From the experiment above Shannon-Fano coding gives better results as compare to Huffman codes in term of space and the size of the text.

REFERENCES

1. Gupta, R., Kumar, M., & Bathla, R. Data Compression-Lossless and Lossy Techniques.
2. A. Kaur and N. S. Sethi (2015). Advanced for lossless text data compression using Advance bit reduction algorithm. International Journal of Advanced research in computer science and software engineering, 1172-1176.
3. Kaur and V.S. Verma (2012) Design and Implementation of data compression and algorithm, International Journal of Information Science and Technology 2(4) 71-81
4. Chen, M., Mao, S., & Liu, Y. (2014). Big data: A survey. Mobile networks and applications, 19(2), 171-209.
5. Lin, M. B., Lee, J. F., & Jan, G. E. (2006). A lossless data compression and decompression algorithm and its hardware architecture. IEEE TRANSACTIONS on very large scale integration (vlsi) systems, 14(9), 925-936.
6. Singh, A., & Singh, V. P. (2013). Enhanced approach of Entropy Coding in Image Compression (Doctoral dissertation).
7. Jalilian, O., Haghighat, A. T., & Rezvanian, A. (2009, October). Evaluation of Persian text based on Huffman data compression. In Information, Communication and Automation Technologies, 2009. ICAT 2009. XXII International Symposium on (pp. 1-5). IEEE.
8. Howard, P. G., & Vitter, J. S. (1994). Arithmetic coding for data compression. Proceedings of the IEEE, 82(6), 857-865.
9. Shanmugasundaram, S., & Lourdusamy, R. (2011). A comparative study of text compression algorithms. International Journal of Wisdom Based Computing, 1(3), 68-76.
10. Kim, H., Wen, J., & Villasenor, J. D. (2007). Secure arithmetic coding. IEEE Transactions on Signal processing, 55(5), 2263-2272.
11. Ranjeet, K., Kumar, A., & Pandey, R. K. (2013, December). An efficient compression system for ECG signal using QRS periods and CAB technique based on 2D DWT and Huffman coding. In Control, Automation, Robotics and Embedded Systems (CARE), 2013 International Conference on (pp. 1-6). IEEE.
12. Kodituwakku, S. R., & Amarasinghe, U. S. (2010). Comparison of lossless data compression algorithms for text data. Indian journal of computer science and engineering, 1(4), 416-425.