Indian Journal of Natural Sciences www.tnsroindia.org.in ©IJONS

Vol.16 / Issue 89 / April/ 2025 *International Bimonthly (Print) – Open Access* *ISSN: 0976 – 0997*

RESEARCH ARTICLE

A Comparative Analysis of DNA Sequence Compression using Conventional and Emerging Deep Learning Technique

Alam Jahaan[1]* and L. Amudavalli[2]

[1]Assistant Professor in Computer Science, AIMAN College of Arts & Science for Women (Affiliated to Bharathidasan University, Tiruchirappalli), Trichy, Tamil Nadu, India.
[2]Research Scholar, Bharathidasan University, Trichy, Tamil Nadu, India.

Received: 21 Nov 2024 Revised: 18 Dec 2024 Accepted: 17 Mar 2025

***Address for Correspondence**
Alam Jahaan
Assistant Professor in Computer Science,
AIMAN College of Arts & Science for Women
(Affiliated to Bharathidasan University, Tiruchirappalli),
Trichy, Tamil Nadu, India.

This is an Open Access Journal / article distributed under the terms of the **Creative Commons Attribution License** (CC BY-NC-ND 3.0) which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited. All rights reserved.

ABSTRACT

DNA compression techniques aim to reduce size of DNA sequences for efficient storage and transmission. Due to the repetitive and structured nature of DNA sequences, specific compression algorithms are often used instead of general-purpose ones. Since deep learning techniques can learn complicated patterns and representations in data by feeding it with training and test data, they have been effectively used to a variety of data compression problems. Hence Deep Learning can be leveraged for DNA compression too. In this paper, three distinct approaches to DNA sequence compression are thoroughly and methodically examined. These approaches include the conventional 2-bit based method, Bit DNA Squeezer (BDNAS), a single bit-based technique, and Deep DNA: A hybrid convolutional and recurrent neural network for compressing human mitochondrial genomes.

Keywords: DNA Compression, Two-bit Compression, BDNAS, Deep Learning

INTRODUCTION

An organism's genome contains its genetic information. It contains both the genes and the noncoding regions, and it is encoded in DNA (Deoxy-Ribo-Nucleic Acid) [1]. The creation of effective DNA sequence compressors is essential for lowering storage requirements, expediting genetic data transfer, and facilitating analysis. Research in the fields of genetics, biomedical sciences, and forensic sciences all demonstrate its significance. Gene sequences are typically represented by four nucleotides, or bases: adenine (A), thymine (T), guanine (G), and cytosine (C). These sequences are stored in an ASCII-based text file using the FASTA format [2] and can be seen as a one-dimensional array of nucleotides (A, C, G, and T). The DNA sequence of the human mitochondrial genome is input for any DNA

compression method. There are numerous genomic data sets available online, and most DNA compression techniques utilize the standard benchmark datasets. This study compares the contemporary Deep Learning based Deep DNA technology with traditional bit based compression algorithms including general purpose 2-bit based and single bit based (BDNAS) techniques.

Two bit based methods[3]

Before the actual compression process starts, these methods first perform bit pre-processing by assigning the four distinct two bits (A = 00, T = 01, C = 10, and G = 11) to each of the four DNA bases followed by other techniques to achieve a better compression ratio. This method reduces the storage requirement to 2 bits per nucleotide. There are several tools available for compressing DNA, including BioCompress, which combines Lempel-Ziv with Run-Length Encoding; Gen Compress; DNA zip; DNAPack; Genbit; DNAbit[7]; Huffbit; and GEncodex,

One Bit based method (BDNAS)[4]

Here the Bit DNA Squeezer (BDNAS) : A Unique Technique for DNA Compression method is discussed which employs a unique two-step process for compressing DNA sequences as given below :

1. Counting and assigning binary values (0 or 1) to corresponding (1st and 2nd) highest nucleotide frequencies.
2. Location Mapping and assigning the Binary Value (0 or 1): Locating and noting the position of the Next Nucleotide then assigning a Binary Value (0 or 1) respectively.

Deep Learning method (DeepDNA) [5]

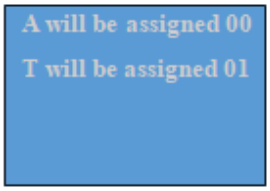
Deep learning has tremendously advanced technology by enabling the automatic and efficient learning of complex patterns in enormous data sets, hence revolutionizing numerous fields. In this work, we examine DeepDNA, A Hybrid Convolutional and Recurrent Neural Network for Compressing Human Mitochondrial Genomes Technique developed by Rongjie Wang and his team. It effectively compresses human mitochondrial genome sequences by combining long short-term memory (LSTM) networks with convolutional neural networks (CNN). Six layers make up the DeepDNA architecture, as illustrated by Figure 3. The remainder of this paper is organized as follows. Section II describes the methodology of all the three types of Compression techniques in detail. Section III presents the Analysis and Discussions. Finally, the Conclusion is reported in Section IV followed by References.

METHODOLOGY

Architecture &Working of Traditional DNA Compression Techniques

Working of 2-bit compression Techniques

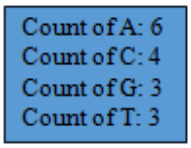
Here each nucleotide in the DNA sequence is encoded using two bits. A 2-bit binary code may uniquely represent each of the four nucleotides (A, T, C, and G), hence only two bits of storage is needed for each nucleotide.
Therefore, "AGTC" would be compressed to 00110110.



Working of 1-Bit Compression (BDNAS)

It employs a unique two-step process for compressing DNA sequences. Here's a breakdown of the process:

Step1: Counting and Assignment of Binary Values: This process effectively categorizes the nucleotides into two groups based on their frequency



- In the first step, each of the four nucleotides (A, C, G, T) in the DNA sequence is counted.
- The nucleotide with the highest count is assigned the binary value "0".
- The nucleotide with the next highest count is assigned the binary value "1".

Step2: Location Mapping and Binary Value Assignment: The positions of the remaining two nucleotides are stored in a position map.

- The position of 3rd most frequent nucleotide is noted down and "0" is assigned to it.
- Position of the fourth most frequent nucleotide is noted down, "1" is assigned to it.

Let's illustrate this process with a hypothetical DNA sequence:
ACGATCACGTACGTAA

Step 1:Counting and Assignment of Binary Values
Nucleotides are arranged according to frequencies, let's assign "A" as "0" and "C" as "1".

Step 2:Location Mapping and Binary Value Assignment:
Positions of G: 3, 9, 13 are noted in Position Map
Positions of T: 5, 10, 14 are noted in Position Map
Then Assign "G" as "0" and Assign "T" as "1"

Compressed Representation
Using the assigned binary values, the compressed representation of the DNA sequence "ACGATCACGTAC GTAA " would be assigned:
A: 0
C: 1
G: 0
T: 1
Resulting in the compressed sequence: "0100110101010100"
Location Map storing the location of G and T Nucleotides.
G: [3, 9, 13]
T: [5, 10, 14]

The Architecture and working of the DeepDNA model
DeepDNA[5] is a hybrid convolutional and recurrent neural network designed to compress human mitochondrial genomes efficiently.

One-Hot Representation
Transformation of input genome sequence into 4-dimensions bit matrix.

- This layer Converts DNA sequences into one-hot encoded matrix, where each nucleotide (A, T, C, G) is represented by a unique binary matrix.
  - Input: DNA sequence (e.g., "ATCG")
  - Output: One-hot encoded matrix

(e.g., A = [1,0,0,0], T = [0,1,0,0], C = [0,0,1,0], G = [0, 0, 0, 1])

Convolution Layer: Acts as a local feature scanner.

- This layer applies convolutional filters to capture local sequence patterns and features, hence identifies motifs and short-range dependencies in the DNA sequence.
- Input: One-hot encoded matrix
- Output: Feature maps highlighting local patterns and motifs within the DNA sequence

Max Pooling: Reduces the size of the output matrix preserving the main features.

- Reduces the dimensionality of the feature maps from the convolutional layer by taking the maximum value over a specified window. This helps in retaining the most significant features while reducing computational complexity.
- Input: Feature maps from the convolution layer
- Output: Reduced dimensionality feature maps, retaining the most significant features

LSTM Layer: Acts the role of capturing sequence global features

- A Long Short-Term Memory (LSTM) layer is used to capture long-range dependencies and sequence patterns over the entire DNA sequence. LSTMs are effective in handling sequential data and can learn relationships that span longer distances than convolutional layers.
- Input: Pooled feature maps
- Output: Contextualized feature representations capturing long-range dependencies in the sequence

Fully Connected Layer: Gathers the LSTM outputs.

- This layer integrates features learned from the previous layers and maps them to the output space. It serves as a bridge between LSTM layer and final output layer.
- Input: Output from the LSTM layer
- Output: Dense representations summarizing features extracted from previous layers

Output Layer: Produces the final compressed representation of the DNA sequence

- performs a sigmoid non-linear transformation to a vector that serves as probability predictions of the sequence base.. This layer typically uses an arithmetic encoder to generate the compressed data based on the learned sequence features and patterns.
- Input: Fully connected layer output
- Output: Compressed DNA data (encoded probabilities for efficient storage)

Decompression

Lossless compression is achieved on the test dataset since both the compression and the decompression estimate the base probability based on the same weight training. The only distinction between the decompression and compression processes is that the former uses an arithmetic encoder to calculate the base probability and output the probability value as a bit-stream, while the latter reconstructs the sequence bases using the bit-stream and the probabilities of the bases.

ANALYSIS & DISCUSSION

Metrics

Compression Performances may be measured by comparing the quality of the original file with the compressed file using quality metrics as discussed below [4] For each technique, the following metrics may be used :

- Compression Ratio: Ratio of size of compressed data to the size of the original data.

$$\text{Compression Ratio} = \frac{\text{Compressed file size}}{\text{Original File Size}}$$

- Compression Time: The time taken by the proposed algorithm to compress each DNA sequence is calculated in milliseconds.
- Decompression Time: The time taken by the proposed algorithm to decompress and retrieve the original DNA sequence. It is calculated in milliseconds too.
- Memory Usage: Amount of memory required during compression and decompression processes.

Performance Analysis

Numerous Datasets are available online for evaluation of compression techniques; however, in this study, we focus on the average compression ratio, which is expressed in bits per base (bpb) and is based on earlier studies . In order to evaluate the BDNAS and general 2-bit DNA compression techniques, a standard benchmark dataset [3] including the four Human Genes of DNA sequences; HUMDYSTROP, HUMHBB, HUMHDABCD, and HUMHPRTB [4] has been examined, and the results have been tabulated. As indicated in Table 1, five randomly chosen sequences from the 100 human mitochondrial genomes dataset [5] are considered for comparison and analysis when using the DeepDNA approach. Metrics based on the average performance of each technique are used in a hypothetical analysis to assess the compression efficiency of the three methods used for this study. Comparison of compression size and compression ratios are considered for evaluation. The original size of a given genome in relation to the chosen datasets has been taken into account for the comparison of original size and compressed size as shown in Table2 and illustrated in Figure 4. The compression ratios are computed using the average compression performance of each technique on the specified dataset as indicated in Table 3 and depicted in Figure 5. A higher compression ratio indicates better compression efficiency, which means that more compression is accomplished compared to the original data size However, actual compression ratios might vary based on factors such as characteristics of the DNA sequences, implementation details of techniques, and exact configurations employed during compression. For accurate compression ratio comparisons, it is imperative to carry out practical experiments or refer to existing literature. The majority of two-bit based coding techniques allocate two bits (00, 01, 10, 11) to each nucleotide (A, T, C, G) in DNA sequences respectively. Here a dataset with 1000 bases needs 2000 bits of storage space. BDNAS Technique reduces the dataset by roughly 50% compared to two-bit based techniques. In this case, 1000 bases in a dataset require 1000 bits of storage space. Further compression may be achieved by implementing existing techniques for both the traditional methods. DeepDNA method has achieved anamazing compression rate in the test dataset with an average compression rate of 0.0336 bpb (bits per base). Compression was conducted for 100 human mitochondrial genomes dataset using DeepDNA method [5].

CONCLUSION

Conventional bit based DNA compression techniques are crucial for advancement in bioinformatics, efficient storage, effective transmission, and analysis of enormous volumes of genomic data. The Bit DNA Squeezer (BDNAS) technique uses a unique binary value assignment process based on the frequency and position of nucleotides in the DNA sequence. This process attempts to effectively compress DNA sequences while maintaining their vital information. Deep learning architecture provides significant advantages for the transmission and storage of genomic data, regardless of higher processing requirements. While traditional methods are still useful because of their simplicity and reduced processing demands. Deep Learning techniques such as DeepDNA are the way of the future for genomic data compression. DeepDNA outperforms traditional DNA compression models in terms of both compression ratio and reconstruction accuracy.

REFERENCES

1. Alam Jahaan, Dr. T.N. Ravi, Dr. S. Panneer Arokiaraj, "A Comparative Study and Survey on Existing DNA Compression Techniques", International Journal of Advanced Research in Computer Science, Volume 8, No. 3, March – April 2017, ISSN No. 0976-5697
2. W. R. Pearson and D. J. Lipman, "Improved tools for biological sequence comparison," Proceedings of the National Academy of Sciences, vol. 85, no. 8, pp. 2444–2448, 1988
3. S. Grumbach and F. Tahi, "Compression of DNA Sequences," in Proc. of the Data Compression Conf., (DEC '93), 1993, 340-350.
4. Alam Jahaan, Dr. T.N. Ravi, Dr. S. Panneer Arokiaraj, "Bit DNA Squeezer (BDNAS): A Unique Technique for DNA Compression," International Journal of Scientific Research in Computer Science, Engineering and Information Technology, Volume 2, Issue 4, 2017.
5. Rongjie Wang, Yang Bai, Chu Yan Shuo and Zhenxing Wang, " DeepDNA: a hybrid convolutional and recurrent neural network for compressing human mitochondrial genomes", Conference: BIBM 2018,Madrid Spain, December 2018, DOI:10.1109/BIBM.2018.8621140, https://www.researchgate.net/publication/329706133,
6. S. R. Kodituwakku Et. Al. "Comparison Of Lossless Data Compression Algorithms For Text Data", Indian Journal Of Computer Science And Engineering, Vol 1 No 4 416-425
7. Rajeswari, P. R., and Apparao, A., 2011, DNABIT Compress - Genome compression algorithm, Bioinformation, 5(8), 350-360
8. Cui, W., Yu, Z., Liu, Z., Wang, G., & Liu, X. (2020). Compressing Genomic Sequences by Using Deep Learning. *ICANN 2020*. Springer. Springer Link

Table 1: Four human Genes Original size

DATASET	ORIGINAL SIZE
HUMDYSTROP	38,770
HUMHBB	73,308
HUMHDABCD	58,864
HUMHPRTB	56,733
Total size : 2,27,676	
DeepDNA **100 human mitochondrial genomes Total Size : 16,56,779**	

Table2: Comparison of Total Original size and Compressed size for Bit based[4]&DeepDNA Techniques[5]

TECHNIQUE	ORIGINAL SIZE	COMPRESSED SIZE
2 Bit Based Technique (for 4 Genomes)	227675	83090
1 Bit Based Technique : BDNAS (for 4 Genomes)	227675	41545
DeepDNA (for 100 Genomes)	1,656,779	55667

Table 3: Comparison of Compression Efficiency.

TECHNIQUE	COMPRESSION RATIO (bpb)
2 Bit Based Technique	0.36
1 Bit Based Technique : BDNAS (avg of 4 Genomes)	0.18
DeepDNA (avg of 100 Genomes)	0.0336

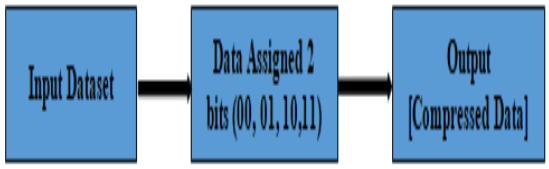


Figure 1: Architecture of 2 bit based conversion

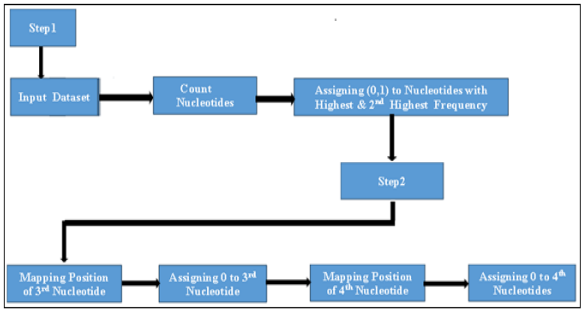


Figure2: Architecture of BDNAS Technique

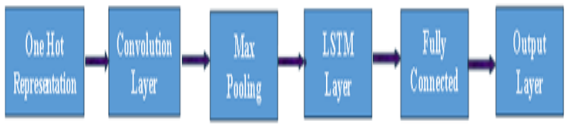


Figure 3: The architecture of DeepDNA which consists of six components

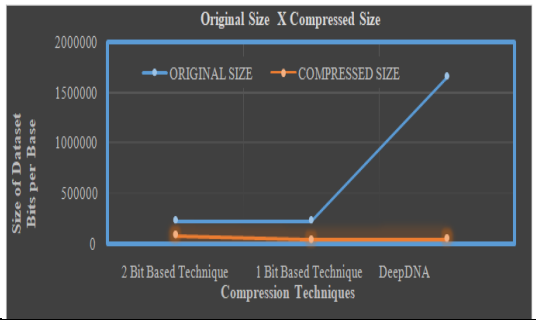


Figure 4: Original size of the three techniques is compared against Compressed size

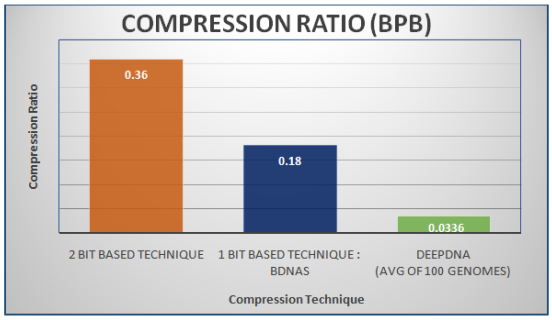


Figure 5: Comparison of Compression Ratio