

Design and development of learning model for compression and processing of deoxyribonucleic acid genome sequence

Raveendra Gudodagi¹, Rayapur Venkata Siva Reddy¹, Mohammed Riyaz Ahmed²

¹School of Electronics and Communication Engineering, REVA University, Bengaluru, India

²School Multidisciplinary Studies, REVA University, Bengaluru, India

Article Info

Article history:

Received Mar 11, 2021

Revised Sep 14, 2021

Accepted Oct 10, 2021

Keywords:

Data compression

Gene quality score

Gene-interaction-networks

Heterogeneous-data-processing

Network-representation-learning

Quality score distribution

ABSTRACT

Owing to the substantial volume of human genome sequence data files (from 30-200 GB exposed) Genomic data compression has received considerable traction and storage costs are one of the major problems faced by genomics laboratories. This involves a modern technology of data compression that reduces not only the storage but also the reliability of the operation. There were few attempts to solve this problem independently of both hardware and software. A systematic analysis of associations between genes provides techniques for the recognition of operative connections among genes and their respective yields, as well as understandings into essential biological events that are most important for knowing health and disease phenotypes. This research proposes a reliable and efficient deep learning system for learning embedded projections to combine gene interactions and gene expression in prediction comparison of deep embeddings to strong baselines. In this paper we perform data processing operations and predict gene function, along with gene ontology reconstruction and predict the gene interaction. The three major steps of genomic data compression are extraction of data, storage of data, and retrieval of the data. Hence, we propose a deep learning based on computational optimization techniques which will be efficient in all the three stages of data compression.

This is an open access article under the [CC BY-SA](https://creativecommons.org/licenses/by-sa/4.0/) license.



Corresponding Author:

Raveendra Gudodagi

Department of Electronics and Communication Engineering, REVA University

Kattigenahalli, Yelahanka, Bengaluru-64, India

Email: rsgudodagi@gmail.com

1. INTRODUCTION

The genomics is an interdisciplinary field that deals with a full collection of deoxyribonucleic acid (DNA) species called genomes [1]. The DNA of the organism (deoxyribonucleic acid) consists of a sequence of four nucleotides in a particular pattern that encodes specifics according to their order [2]. Take the figure, for instance. 1, adenine (A) is accompanied by guanine (G) in this piece of DNA, followed by thymine (T), cytosine (C), cytosine (C), and so on. In simple words, gene sequences are used to evaluate the sequence of certain nucleotides in the gene that forms the DNA of the organism [2]. Of those genetic letters, more than 3 billion make up the human genome [3]. Machine learning on graphs is a crux of the matter and omnipresent role in social networks, with implementations ranging from product design to suggestions for friendship [4], [5]. Over the past few years, however, have seen a rise in autonomous approaches that learn to encode graph structure into low-dimensional embedding, use of deep learning (DL) and Nonlinear techniques, dimensional saving along with key advances in graphic representation learning, including matrix factorization-based methods, random-walk-based algorithms, and graphic neural networks [6], [7]. Machine learning approaches have historically relied on user-defined heuristics to extract features that represent structural graph

information (e.g., kernel functions or degree of statistics) [8], [9]. Figure 1 describes the available binary encoding methods used in DNA-based data compression schemes. Figure 1(a) describes one binary digit is mapped to 2 optional bases. Two binary digits are mapped to 1 fixed base. Figure 1(b) illustrates 1 byte are encoded through Huffman coding and then encoded to 5 or 6 bases. Figure 1(c) presents Two bytes are mapped to 9 bases. In Figure 1(d) eight binary bits are mapped to 5 bases. In Figure 1(e) the crop residues and animal manure composting (CRAM) which is a standard file format for storing compressed DNA sequencing data and with a high correlation between values, CRAM allows numerous data series to be stored in the same block, and a combination would result in superior compression ratios as compared to bi-directional associative memory (BAM) file formats. Although the CRAM format outlines the file layout and decoding mechanism, the encoder has full control over how the data is broken up.

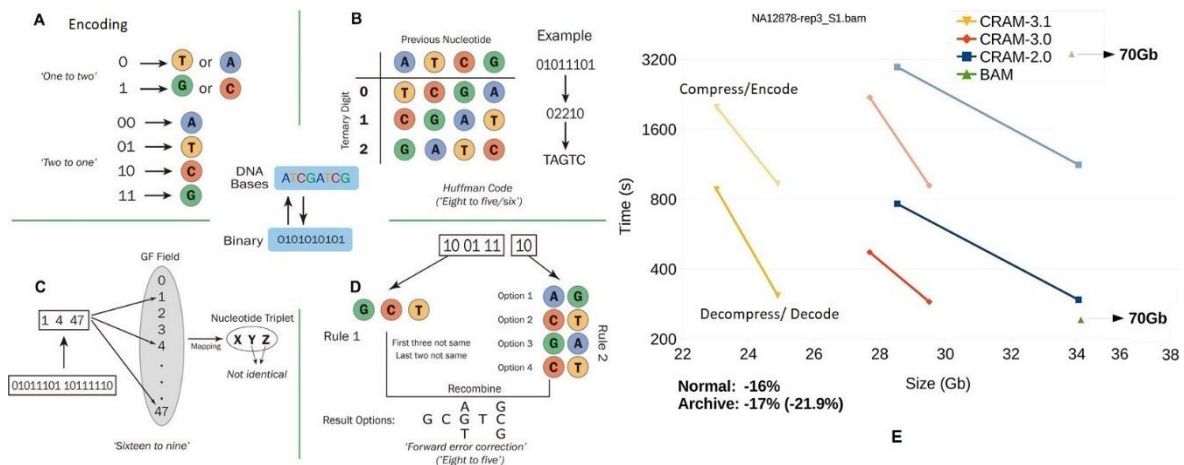


Figure 1. Binary encoding methods used in DNA-based data compression schemes: (a) encoding, (b) Huffman encoding, (c) 2:9 base mapping, (d) 8:5 base mapping, and (e) bases represented with crop residues and animal manure composting CRAM

The principal challenge in this area is to find a way to characterize, or encrypt, the structure of graphs so that machine learning models can effortlessly exploit [10], [11]. For example, one might want to classify a protein's position in a biological interaction graph, find contribution of a sample under consideration in a collaboration network, recommends new mates to a social network consumer, or predict novel therapeutic formulations of present receptors for the drug, the composition of which can be described as a graph [12], [13]. The key contributions of this research manuscript are: i) gene functional prediction; ii) gene ontology (GO) reconstruction; and iii) genetic interaction (GI) prediction. Novelty of this research is: we propose to generate copy number of variants (CNV), copy number state (CNS) and quality score distribution of deletions and duplications in gene population [14]. Following are the key contributions of this research manuscript: i) we propose an efficient DL technique for learning embedded projections to combine gene interaction and expression in prediction comparison of deep embeddings to strong baselines and ii) we perform data processing operations and predict gene function, with ontology reconstruction and predict gene interaction.

2. BACKGROUND

Previously, general-purpose compression tools have been used to compress genomic data, and recently several advanced compression tools have been developed to serve this process [15], [16]. Genes and protein properties in a human organism are important molecular units [8], [17], [18]. Awareness of their roles is essential in understanding processes (genetic, functional, and clinical), as well as developing new medicines and therapies [17]. A gene or protein's connection with its functions, defined by regulated terms of biomolecular terminologies or ontologies, is called functional gene annotation [17]. There are many useful annotations of DNA, represented by terminology and ontology [19]. Nevertheless, they may contain many misspelled details, since curators are reviewing only a part of the annotations [20]. However, despite the rapidly increasing speed of biomolecular information, they are incomplete by definition [17], [21]. In this scenario, computational methods which can accelerate the process of curing the annotation and reliably recommend new annotations are very relevant [15], [22]. The structural patterns of higher order are important for the structure and operation of

complex networks, and the development of decoding algorithms capable of decoding complex patterns is a main course for future research [20], [23]. Typical applications for the grouping of semi-supervised nodes include labeling of proteins according to their biological function and classification of documents, images, web pages or individuals into different categories/communities [19], [24]. The inductive node classification task, where the objective is to classify nodes not seen during training [25].

The GI is a form of communication, where one gene's effect is mutated by one or more other genes [26], [27]. These communications are essential in order to define functional relationships between genes and their respective proteins, and for revealing multifaceted biological and disease processes [28]. An significant form of GI synthetic illness or synthetic lethality includes two or more genes where no lack of any single gene affects the sustainability of the cells, however the blended loss of both genes results in an extreme illness or cell fatality [29], [18], [21]. Identifying GIs is a big challenge as it can help to describe the corridors, complex proteins and supervisory dependencies [15]. Although near-systematic high-content examination for GIs is feasible in single cell organisms such as yeast, the systematic discovery of GIs in mammalian cells is exceptionally tricky [20], [21]. “Therefore, computational methods are greatly required in order to accurately predict GIs in these species, including synthetic lethal interactions. Here, we examine state-of-the-art techniques, tactics, and systematic methods for studying and predicting GIs, both under general conditions (healthy/standard laboratory) and in particular circumstances, such as diseases” [13], [30].

Figure 2 depicts the functional and deep learning models for processing of genomic/DNA data. In computational biology, for example in the functional modelling of DNA and protein sequences in the Figure 2(a), describes predicting which regions of biological sequences are functional and what that function could be using the sequential modelling strategies where sequential data pops up absolutely everywhere. Figure 2(b) illustrates the block diagram representation of the deep learning model for genomic data processing and the network interface. Gene network embedding (GNE) system for predicting gene interaction. On the left, the one-hot encrypted depiction of the gene is mapped to the opaque vector $v(s)_i$ of dimension $[d, 1]$ which captures topological properties and transforms the gene expression vector into $v(a)_i$ of dimension $d=1$ which accumulates the information of the attributes. First, combining the power of modeling of all network structure and attributes with 2 embedded vectors (creates a vector embedded in 2d dimension). Then, nonlinear aggregated vector transformation enables GNE to capture and assign information to thorny statistical liaisons between network structures and learn better descriptions. Ultimately, this examined depiction of dimensions $[d, 1]$ is converted into a probability longitude vector $[M, 1]$ in the output layer containing the foretelling gene v_i likelihood of all genes in the model. Restricted probability on output layer $p[v_j|v_i]$ implies the possibility that gene v_j is related to gene v_i .

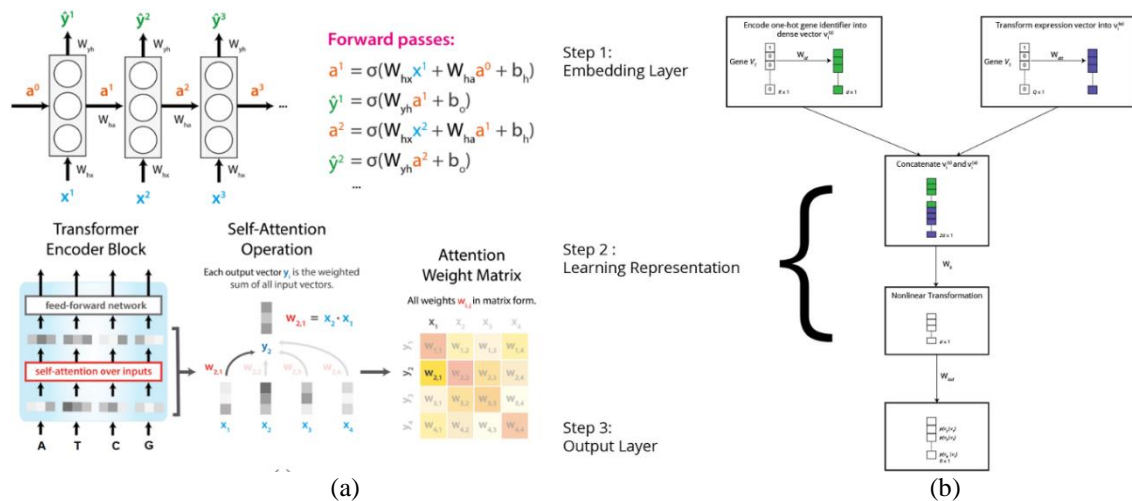


Figure 2. Functional and deep learning models for processing of genomic/DNA data for (a) functional modelling of DNA and protein sequences and (b) block diagram illustration of deep learning model for genomic data processing and network inference

3. DATASETS AND VALIDATION PROCEDURE

In order to learn a lower dimensional depiction, GNE combines gene interaction network and gene expression data. Now, first we go with recommending a data pipeline using various semantic and machine

learning (ML) methods to predict novel ontology dependent annotations of activated genes; then we implement a new semantic priority instruction to compartmentalize expected annotations by their probability of being accurate. Our experiments and validations proved the efficacy and importance of the expected annotations in our pipeline, by choosing as most likely several forecasted annotations that were later validated. The nodes are genes, so the same-colored genes have identical vocal profiles. GNE groups genes with identical network topology that are related in the graph or have a common neighborhood and allocate similarities (common profiles of articulation) in the implanted space which is illustrated using Figure 3.

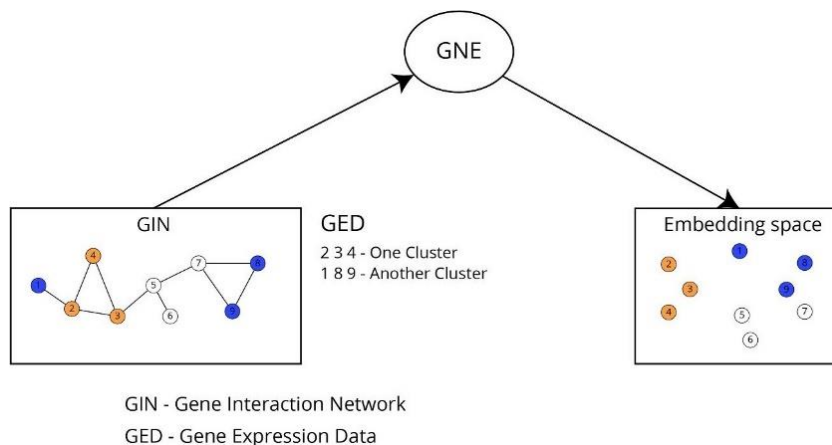


Figure 3. Gene network embedding

Algorithm 1. Inference algorithm for gene interaction network (GIN) reconstruction

```

Start;
Gene expression data;
Obtain range ( $\mu$ ) and step threshold coefficient ( $\mu_{min}$ -  $\mu_{max}$ ) variation;
if : $\mu = \mu_{min}$ 
Gene regulatory network reconstruction;
Calculate parameters of network topology;
else if:
 $\mu < \mu_{max}$ 
    Analyze the obtained results;
    Setup the new range;
else if:
If  $\mu > \mu_{max}$ 
    Calculate complex topological criteria;
    Determine the optimal value of coefficient threshold;
    Formation of Optimal network topology;
end of if
Function to obtain symmetric adjacency matrix on an undirected graph;
Function to reduce incoherent linked genes;
Function to reduce the diffusion kernels;
Stop;
    
```

The weights of the validated neural network training are as depicted in Figure 2(b) with the position sequences of a three-layered neural network. The weights are obtained after training the neural network considering the parameters of pre-training model of Table 1 and the trained neural network is viewed. There are two types of applications for DNA-based data storage as depicted in Figure 4. In vivo DNA-based data storage is demonstrated in Figure 4(a) and Figure 4(b); in vitro DNA-based data storage is demonstrated in Figure 4(c) and Figure 4(d). In Figure 4(a) there is high-throughput DNA oligo analysis using an array. The digital information carried by DNA oligos is stored in the form of an oligo pool. In Figure 4(b) the information to be processed will be carried by DNA fragments synthesized by polymerase cycling assembly. The Figure 4(c) presents the plasmids implanted with digital information and then moved into bacterial cells. The Figure 4(d) using the clustered regularly interspaced short palindromic repeats (CRISPR) method and the Cas1-Cas2 integrate, DNA fragments containing digital information are incorporated into the bacterial genome. Finally Figure 4(e) the mechanism provides for the massive partitioning and barcoding of single DNA cells using >1,000,000 unique barcodes.

Table 1. Neural network pre-training model

Parameter	
Total genes	5950
Training interactions (positive)	395875
Training interactions (negative)	395875
Validation interactions (positive)	43987
Validation interactions (negative)	43987
Test interactions (positive)	48874
Test interactions (negative)	48874
Dimension of attributes	536
Number of genes	5950

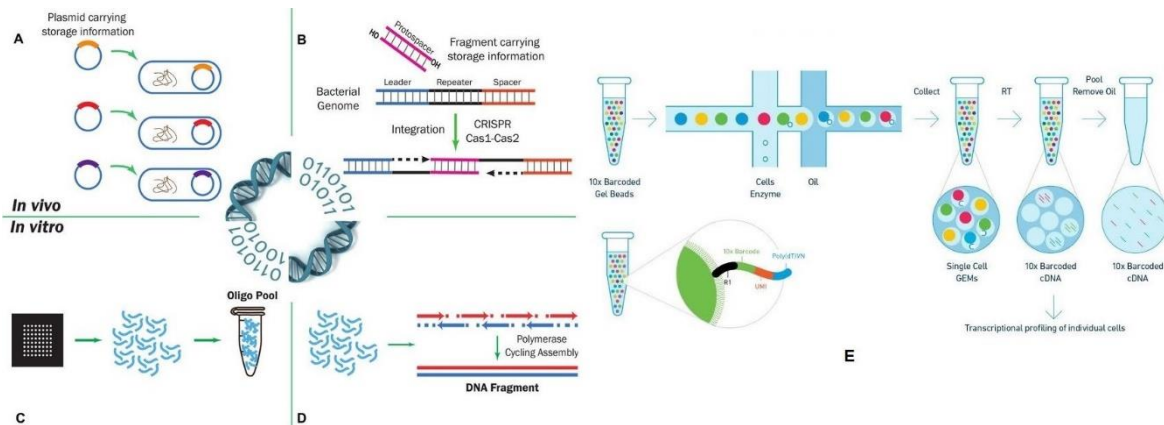


Figure 4. In vivo DNA-based data storage (a) plasmid carrying, (b) fragment carrying, (c) oligo pool, (d) DNA fragment, and (e) massive partitioning and barcoding

4. RESULTS AND DISCUSSION

Ontology can be characterized as the representation of observed objects, and relations among those objects. GO is a fine tuned, biologically dependent structural terminology. It is de facto norm for prediction of gene functionality [19], [31]. This is widely used for gene-functional annotation and enrichment analysis which explains species-neutrally molecular structure, biological cycle, and gene product cellular components [32], [33]. The GO is used to explain gene function, analyze pathways, and model networks, to name but a few. The benefit of GO is that it enables exchange of knowledge between the different biological communities. The complete alignment score was summed up by bringing the geometric mean of three forms of GO ontology alignment ratings.

Table 2 depicts the neural network training parameters, whereas Table 3 illustrates the neural network training runtime values. With * indicates the true positives obtained during neural network training runtime. The remaining parameters without * indicates the false positive values obtained at run time. The GIN reconstruction algorithm accepts gene expression data as input and calculates the threshold coefficient range μ . The minimum value ' μ_{min} ' and maximum value ' μ_{max} ' are taken from the data set. After getting these values the threshold coefficient variation is obtained by taking the difference between maximum and minimum values ($\mu_{min}-\mu_{max}$). Now in the first step, the range μ is compared with minimum and maximum values, if the range is equal to minimum value ($\mu=\mu_{min}$), gene regulatory network reconstruction is done in accordance with the obtained parameters of the network topology. In another way, if the range is having value lesser than the maximum value ($\mu < \mu_{max}$) the obtained results are analyzed, and accordingly, the new range is updated in the data set.

In the later part of the algorithm, if the maximum value is lower than the range ($\mu > \mu_{max}$), corresponding complex topological criteria is obtained, and optimal value of coefficient threshold is determined. On this basis the new network topology is generated. If the maximum value is larger than the range, the functions are executed to: i) obtain symmetric adjacency matrix on an undirected graph; ii) to reduce incoherent linked genes, and iii) to reduce the diffusion kernels. The output values of neural network are as follows: "Average precision score->GNE test AP score: 0.820479941, region of convergence->GNE test ROC score: 0.823534986 and alpha value obtained is 1".

The efficiency of the various tools was measured by the following similar metrics: ratio of compression, the copy number of variants (CNV) of the ratio of compression, copy number state (CNS) of

the ratio of compression, the overall compression quality score. We have evaluated different parameters with respect to compression of copy CNV and CNS data of the gene population. Here we propose CNV in the gene population with considering top 25 genes that overlap with the given CNV data set, and in later part we have to obtain CNV length distribution in each gene population by dropping to 1% of gene population. This is shown in Figure 5, the first graph gene count versus gene name of top 25 genes. CNV and quality score distribution of deletions and duplications are shown in Figure 6. First part of graph gives CNV counts of deletions and duplications in each gene population, other part depicts the quality score distributions of deletions and duplications. The duplications (dup) are indicated with blue colored bar/line and deletions (del) are indicated in red color. Gene population considered in this model are ExAC-FIN, ExAC-NFE, ExAC-AMR, ExAC-SAS, ExAC-OTH, ExAC-AFR, ExAC-EAS. In Figure 7, deletions and duplications are shown in different graphs with respect to CNV length distribution in each gene population after excluding the top 1% of population by considering the CNV length in kb versus the cumulative frequency. Figure 8 illustrates the performance of our cell compression model on Markov-k sources. Our cell compression model is able to compress Markov-50 sources, which is better than the performance and essentially it is able to capture dependence up to 50 timesteps.

Table 2. Neural network training parameters

Parameter	
id_embedding_size	128
attr_embedding_size	128
representation_size	128
Alpha	1
n_neg_samples	10
Epoch	20
batch_size	256
learning_rate	0.01

Table 3. Neural network training-runtime values

Epoch No.	Train-Batch Loss	Validation ACC
1	7.653298878	0.651800936 *
2	5.138429771	0.691491825 *
3	2.894665637	0.739070140 *
4	1.74604629	0.752840842 *
5	1.628749394	0.764563969 *
6	1.520544519	0.776617788 *
7	1.503320801	0.785356915 *
8	1.468874492	0.789824807 *
9	1.497068605	0.791630406 *

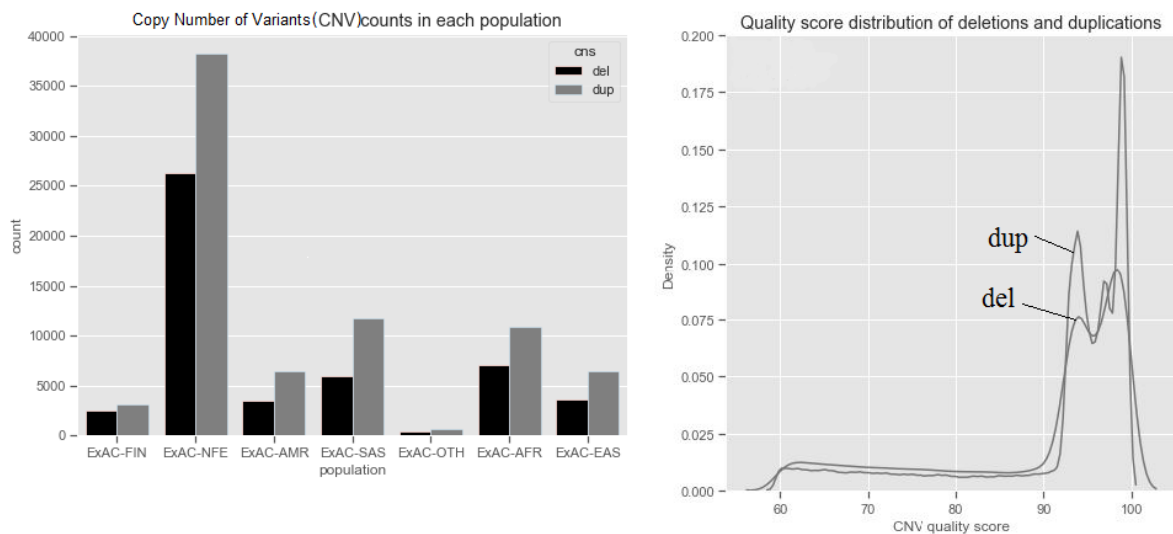


Figure 5. Copy number of variants and quality score distribution of deletions and duplications

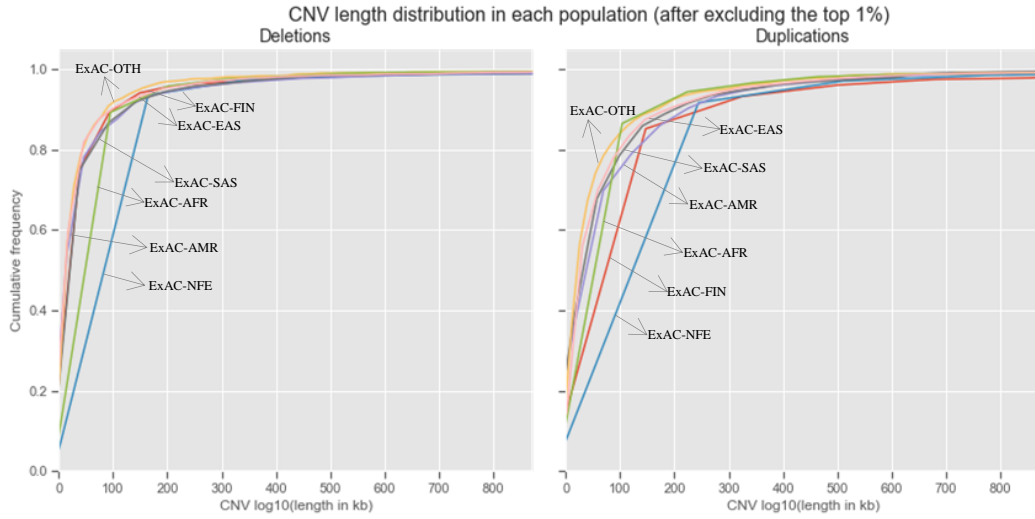


Figure 6. Copy number of variants distribution excluding top 1% of population

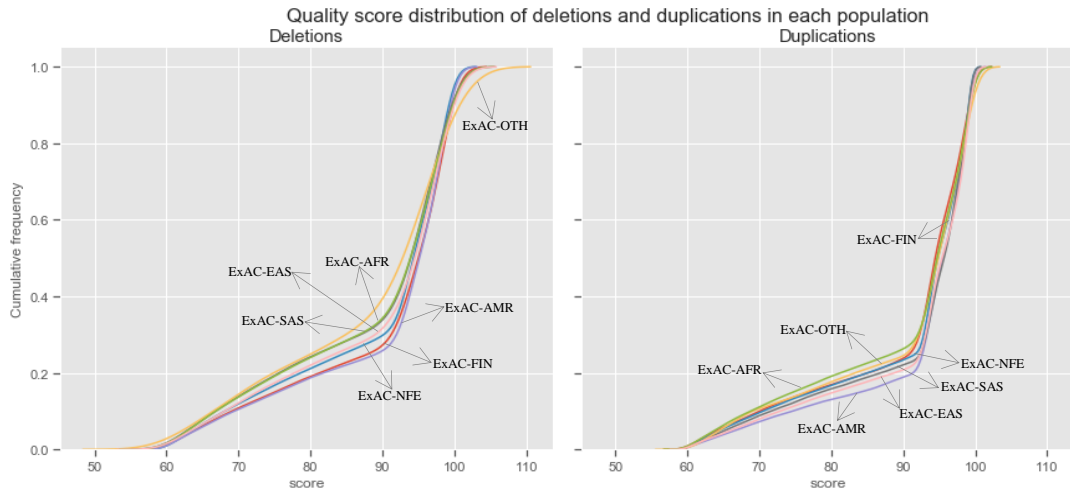


Figure 7. Copy number of variants distribution excluding top 1% of population

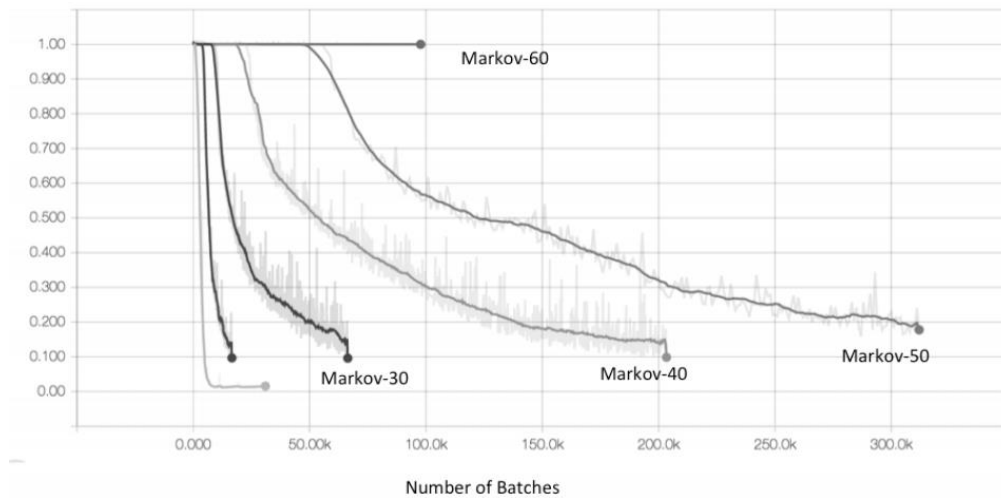


Figure 8. Performance of our cell compression model on markov-k sources

5. CONCLUSION

In this paper we have proposed a deep learning model, which is able to carry a systematic analysis of associations between genes. Which provides techniques for the recognition of operative connections among genes and their individual products. We performed detailed insights into primary biological events which are crucial in understanding health and disease phenotypes. This research paves a way for reliable and efficient deep learning system for learning embedded projections to integrate gene interactions. Further an inference algorithm for gene interaction network is presented which illuminates decrease in number of incoherent genes and the diffusion kernel. Further this inference algorithm forms the core of this research paper. Only while sophisticated tech is operating on state-of-the-art hardware can a huge storage problem be overcome. Computational methods for different forms of data compression have been proposed. The length of each gene population is reduced by doing duplications and deletions.




REFERENCES

- [1] R. Wang, T. Zang, and Y. Wang, "Human mitochondrial genome compression using machine learning techniques," *Human Genomics*, vol. 13, no. 1, Oct. 2019, doi: 10.1186/s40246-019-0225-3.
- [2] S. T. Park and J. Kim, "Trends in next-generation sequencing and a new era for whole genome sequencing," *International Neurourology Journal*, vol. 20, pp. 76–83, 2016, doi: 10.5213/inj.1632742.371.
- [3] C. Ting, R. Gooding, R. Field, and J. Caswell, "Reordering genomic sequences for enhanced classification via compression analytics," *Proceedings - 18th IEEE International Conference on Machine Learning and Applications, ICMLA 2019*, pp. 252–258, 2019, doi: 10.1109/ICMLA.2019.00047.
- [4] M. S. Rao *et al.*, "Novel computational approach to predict off-target interactions for small molecules," *Frontiers in Big Data*, vol. 2, pp. 1–17, Jul. 2019, doi: 10.3389/fdata.2019.00025.
- [5] R. S. Gudodagi and M. R. Ahmed, "Customized computational environment for investigations and compression of genomic data," *International Journal of Pharmaceutical Research*, vol. 12, Nov. 2020, doi: 10.31838/ijpr/2020.SP2.423.
- [6] M. Goyal, K. Tatwawadi, S. Chandak, and I. Ochoa, "DeepZip: lossless data compression using recurrent neural networks," in *2019 Data Compression Conference (DCC)*, Mar. 2019, pp. 575–575, doi: 10.1109/DCC.2019.00087.
- [7] W. L. Hamilton, R. Ying, and J. Leskovec, "Representation learning on graphs: methods and applications," pp. 1–24, Sep. 2017, [Online]. Available: <http://arxiv.org/abs/1709.05584>.
- [8] V. Cecchini, T. P. Nguyen, T. Pfau, S. De Landtsheer, and T. Sauter, "An efficient machine learning method to solve imbalanced data in metabolic disease prediction," *Proceedings of 2019 11th International Conference on Knowledge and Systems Engineering, KSE 2019*, 2019, doi: 10.1109/KSE.2019.8919337.
- [9] M. R. Kounte, P. K. Tripathy, P. P., and H. Bajpai, "Implementation of brain machine interface using mind wave sensor," *Procedia Computer Science*, vol. 171, no. 2019, pp. 244–252, 2020, doi: 10.1016/j.procs.2020.04.026.
- [10] J. S. Sousa *et al.*, "Efficient and secure outsourcing of genomic data storage," *BMC Medical Genomics*, vol. 10, Jul. 2017, Art. no. 46, doi: 10.1186/s12920-017-0275-0.
- [11] M. Hosseini, D. Pratas, and A. J. Pinho, "Cryfa: a secure encryption tool for genomic data," *Bioinformatics*, vol. 35, no. 1, pp. 146–148, 2019, doi: 10.1093/bioinformatics/bty645.
- [12] J. P. Hou, A. Emad, G. J. Puleo, J. Ma, and O. Milenkovic, "A new correlation clustering method for cancer mutation analysis," *Bioinformatics*, vol. 32, no. 24, pp. 3717–3728, 2016, doi: 10.1093/bioinformatics/btw546.
- [13] C. Kockan *et al.*, "Sketching algorithms for genomic data analysis and querying in a secure enclave," *Nature Methods*, vol. 17, no. 3, pp. 295–301, 2020, doi: 10.1038/s41592-020-0761-8.
- [14] A. Vinayagam *et al.*, "Controllability analysis of the directed human protein interaction network identifies disease genes and drug targets," *Proceedings of the National Academy of Sciences of the United States of America*, vol. 113, no. 18, pp. 4976–4981, 2016, doi: 10.1073/pnas.1603992113.
- [15] L. Mertzanis, A. Panotonoulou, M. Skoularidou, and I. Kontoyiannis, "Deep tree models for 'big' biological data," in *2018 IEEE 19th International Workshop on Signal Processing Advances in Wireless Communications (SPAWC)*, Jun. 2018, pp. 1–5, doi: 10.1109/SPAWC.2018.8445994.
- [16] R. Jumar, H. Maaß, and V. Hagenmeyer, "Comparison of lossless compression schemes for high rate electrical grid time series for smart grid monitoring and analysis," *Computers and Electrical Engineering*, vol. 71, pp. 465–476, Oct. 2018, doi: 10.1016/j.compeleceng.2018.07.008.
- [17] D. Chicco and M. Masseroli, "Ontology-based prediction and prioritization of gene functional annotations," *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, vol. 13, no. 2, pp. 248–260, 2016, doi: 10.1109/TCBB.2015.2459694.
- [18] N. S. Madhukar, O. Elemento, and G. Pandey, "Prediction of genetic interactions using machine learning and network properties," *Frontiers in Bioengineering and Biotechnology*, vol. 3, pp. 1–12, Oct. 2015, doi: 10.3389/fbioe.2015.00172.
- [19] M. Asif, H. F. M. C. M. Martiniano, A. M. Vicente, and F. M. Couto, "Identifying disease genes using machine learning and gene functional similarities, assessed through gene ontology," *PLOS ONE*, vol. 13, no. 12, Dec. 2018, doi: 10.1371/journal.pone.0208626.
- [20] Y. Li, W. Shi, and W. W. Wasserman, "Genome-wide prediction of cis-regulatory regions using supervised deep learning methods," *BMC Bioinformatics*, vol. 19, no. 1, pp. 1–14, 2018, doi: 10.1186/s12859-018-2187-1.
- [21] T. Wang *et al.*, "Gene essentiality profiling reveals gene networks and synthetic lethal interactions with oncogenic ras," *Cell*, vol. 168, no. 5, pp. 890–903, Feb. 2017, doi: 10.1016/j.cell.2017.01.013.
- [22] R. Challa, G. P. Devi, K. Arava, and K. S. Rao, "A novel compression technique for DNA sequence compaction," in *2016 International Conference on Signal Processing, Communication, Power and Embedded System (SCOPEs)*, Oct. 2016, pp. 1351–1354, doi: 10.1109/SCOPEs.2016.7955660.
- [23] M. Aledhari, M. Di Pierro, M. Hefeida, and F. Saeed, "A deep learning-based data minimization algorithm for fast and secure transfer of big genomic datasets," *IEEE Transactions on Big Data*, vol. 7, no. 2, pp. 271–284, Jun. 2021, doi: 10.1109/TBDATA.2018.2805687.
- [24] M. Z. Hasan, M. S. R. Mahdi, M. N. Sadat, and N. Mohammed, "Secure count query on encrypted genomic data," *Journal of Biomedical Informatics*, vol. 81, pp. 41–52, May 2018, doi: 10.1016/j.jbi.2018.03.003.




- [25] V. W. Zheng, S. Cavallari, H. Cai, K. C.-C. Chang, and E. Cambria, "From node embedding to community embedding," Oct. 2016, [Online]. Available: <http://arxiv.org/abs/1610.09950>.
- [26] M. Blatt, A. Gusev, Y. Polyakov, and S. Goldwasser, "Secure large-scale genome-wide association studies using homomorphic encryption," *Proceedings of the National Academy of Sciences*, vol. 117, no. 21, pp. 11608–11613, May 2020, doi: 10.1073/pnas.1918257117.
- [27] S. Mostafavi, A. Goldenberg, and Q. Morris, "Labeling nodes using three degrees of propagation," *PLoS ONE*, vol. 7, no. 12, Dec. 2012, doi: 10.1371/journal.pone.0051947.
- [28] M. Hosseini, D. Pratas, and A. J. Pinho, "A survey on data compression methods for biological sequences," *Information*, vol. 7, no. 4, 2016, doi: 10.3390/info7040056.
- [29] E. Helgason *et al.*, "Bacillus anthracis, bacillus cereus, and bacillus thuringiensis-one species on the basis of genetic evidence," *Applied and Environmental Microbiology*, vol. 66, no. 6, pp. 2627–2630, Jun. 2000, doi: 10.1128/AEM.66.6.2627-2630.2000.
- [30] W. Li, "Optimize genomics data compression with hardware accelerator," in *2017 Data Compression Conference (DCC)*, Apr. 2017, pp. 446–446, doi: 10.1109/DCC.2017.49.
- [31] J. J. Ward, J. S. Sodhi, B. F. Buxton, and D. T. Jones, "Predicting gene ontology annotations from sequence data using kernel-based machine learning algorithms," in *Proceedings of the 2004 IEEE Computational Systems Bioinformatics Conference (CSB 2004)*, 2004, pp. 507–508, doi: 10.1109/CSB.2004.1332485.
- [32] P. Roncaglia *et al.*, "The gene ontology (GO) cellular component ontology: integration with SAO (subcellular anatomy ontology) and other recent developments," *Journal of Biomedical Semantics*, vol. 4, no. 1, 2013, Art. no. 20, doi: 10.1186/2041-1480-4-20.
- [33] L. M. O. Matos, A. J. R. Neves, D. Pratas, and A. J. Pinho, "MAFCO: a compression tool for MAF files," *PLOS ONE*, vol. 10, no. 3, Mar. 2015, doi: 10.1371/journal.pone.0116082.

BIOGRAPHIES OF AUTHORS






Raveendra Gudodagi    received undergraduate degree in Electronics and Communication Engineering and Postgraduate degree in Signal Processing from Visveswaraya Technical University, Belgaum, India, in 2009 and 2013 respectively. Currently he is pursuing Ph.D. degree in Electronics and Communication Engineering from REVA university. His research interests include Artificial Intelligence, Cognitive Sciences, Machine Learning, Genomic data processing and compression. He is a recognized mentor from Texas Instruments. He is a Member of IEEE, IAENG and IETE. He can be contacted at email: rsgudodagi@gmail.com.



Rayapur Venkata Siva Reddy    received Undergraduate and Postgraduate degree in ECE FROM Gulbarga University, India and Ph.D. degree in VLSI from Sri Krishnadevaraya University, AP, India, in 2013. He is Currently working as Professor in school of Electronics and Communication Engineering REVA University. His research interests include FPGA, VLSI RFID, Smart devices, IOT, Artificial Intelligence, and Embedded systems. He is a recognized mentor from Texas Instruments and recipient of grants from IEEE standards committee. He is a Senior Member of IEEE. He can be contacted at email: venkatasivareddy@reva.edu.in.



Mohammed Riyaz Ahmed    received Undergraduate degree in ECE and Postgraduate degree in CNE from Visveswaraya Technical University, Belgaum, India, in 2007 and 2010 respectively, and Ph.D. degree in Electronics and Communication Engineering from Jain university, India, in 2016. He is Currently working as associate professor in school of Multidisciplinary Studies in REVA University. His research interests include RFID, Smart devices, IOT, Artificial Intelligence, Cognitive Sciences, Machine Learning, Technology Intervention for Elderly, Memristors and Neuromorphic Engineering. He is a recognized mentor from Texas Instruments and recipient of grants from IEEE standards committee. He is a Senior Member of IEEE. He can be contacted at email: riyaz@reva.edu.in.