Growth and Evolution of Deep Neural Networks from Gene Regulatory Networks

Colin Flynn
University of Western Australia
Perth
Australia
colin.flynn@research.uwa.edu.au

Mohammed Bennamoun
University of Western Australia
Perth
Australia
mohammed.bennamoun@uwa.edu.au

Farid Boussaid
University of Western Australia
Perth
Australia
farid.boussaid@uwa.edu.au

ABSTRACT

A simple population based Evolutionary Algorithm (EA) was used to evolve convolutional neural networks for solving an image classification problem (CIFAR10). Each member of the population was defined by a genome. This work proposes the construction of a genome based closely on the naturel world. The genes within such a genome regulate each other's expression and hence build a gene regulatory network (GRN). In the proposed approach, the genome contains no information from the problem space and could be applied to any application in principle. The genome behaves as an evolved program that grows multi-cellular organisms through a developmental process from an initial single cell. The cellular structure is an intermediate phenotype which is then mapped to its final form, a convolutional neural network in this case. The proposed GRN approach was able to evolve successful networks whose level of performance is comparable to a LeNet5 implementation.

CCS CONCEPTS

• **Computer methodologies** → *Artificial Intelligence; Machine Learning; Neural Networks; Bio-inspired Approaches;*

KEYWORDS

Deep neural networks, Gene regulatory networks, Evolutionary algorithm

ACM Reference format:

Colin Flynn, Mohammed Bennamoun and Farid Boussaid. 2021. Growth and Evolution of Deep Neural Networks from Gene Regulatory Networks. In *Proceedings of the Genetic and Evolutionary Computation Conference 2021 (GECCO '21). ACM, New York, NY, USA, 2 pages.* https://doi.org/10.1145/3449726.3459519

GECCO '21 Companion, July 10–14, 2021, Lille, France

ACM ISBN 978-1-4503-8351-6/21/07.
https://doi.org/10.1145/3449726.3459519

1 Proposed Genomic Model

The inspiration for this work is drawn from the properties and behavior of genomes in the natural world [1] [2] [3]. A review of GRNs in artificial evolutionary settings is given by Cussat-Blanc et al [4]. The genomic model used contains protein producing genes where the proteins behave as transcription factors, inter-cellular signals, morphogen gradients and terminal proteins. Every gene has its expression controlled by a regulatory region containing an arbitrary number of logic units, which in turn contain an arbitrary number of binding units. Each binding unit is capable of binding a transcription factor protein string, if it is present above a threshold concentration. Bound binding units within a logic unit combine in a Boolean fashion to turn the logic unit on, which in turn either excites or represses the gene. A binding unit is therefore the fundamental unit of a regulatory region. It consists of DNA, a string drawn from a 4-letter alphabet. When a gene is excited to express, it has a protein coding region that is also represented by DNA. This DNA is examined two letters at a time, (a codon), and converted to a string drawn from a 16-letter alphabet. This new string is then referred to as a protein. Proteins have two basic functions: either as a transcription factor for controlling further gene expression, (intra, inter or extra cellular); or as a terminal protein to be interpreted as forming part of the phenotype. Since transcription factor proteins control the expression of other genes, a GRN is formed from these gene-to-gene connections.

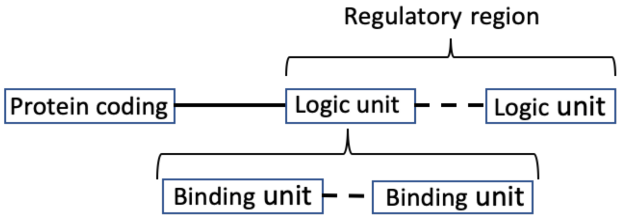

Figure 1 Gene Regulation

A genome is contained within a cell and cells can divide. As gene expression continues in each cell, inter-cellular proteins can be passed between cells, extra-cellular protein concentration

gradients can be formed and cells can collect terminal proteins. As different cells execute different sections of the gene regulatory network, they differentiate from each other. The final collection of cells and their terminal proteins are interpreted to form the final phenotype, which could be anything, but in this case will be a convolutional neural network architecture. Cells are not placed within a coordinate system, but they are ordered. Cell ordering is completely flexible.

2 Experiments and Results

Architectures were evolved to perform an image classification task using the CIFAR10 dataset [5]. Evolution was controlled by a simple GA operating on a population of genomes. Each genome was grown into its final cellular form, mapped onto a phenotype and then trained and graded for its classification accuracy against the validation images provided from the CIFAR10 dataset. Once the phenotypes were graded, pairs of parent genomes were chosen for the production of the next generation and were subjected to point mutations, gene crossover and chromosome segregation. The population was replaced each epoch with the new mutated genomes. The population size was 50 while 40 evolutionary epochs were used.

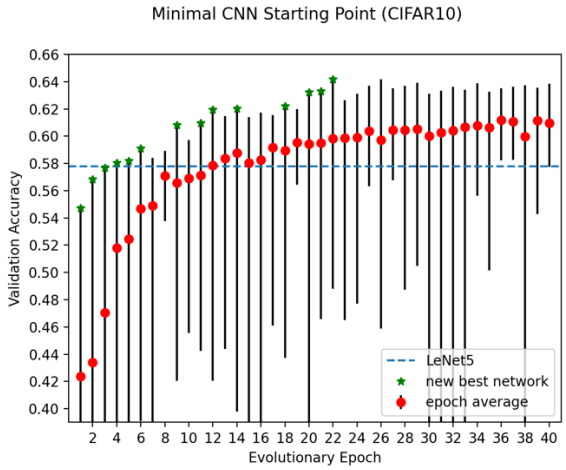

Figure 2 Minimal CNN architecture starting point and the CIFAR10 image classification task. The LeNet5 performance for this task is shown for comparison.

The initial population of genomes was generated from a base genome that was handcrafted to form a minimal CNN. It was then mutated to form the required number of parents for the first population. The minimal CNN had a single convolutional layer (filter size = 2, padding = same, stride = 1, output channels = 2), followed by a max-pooling layer (filter size =2, stride = 1) with a single fully connected layer (output nodes = 20). The convolution layer and the fully connected layer were both followed by a Leaky ReLU activation layer. The output was always a final fully connected layer with 10 output nodes followed by a SoftMax activation layer. There were no evolvable parameters in this final fully connected layer, it merely connected whatever was before it to the 10 classification nodes. Each evolved architecture was trained using 10 training epochs and a batch size of 32. Training was not optimized, as the goal was only to demonstrate the viability of the proposed method.

Results are shown in Figure 2. Validation accuracies started low and continued to improve up to the ninth epoch where progress continued but at a slower rate. New networks within the population that added layers failed to establish themselves. As a comparison, performance for the LeNet5 architecture [6], trained on the same dataset, is shown. The performance of the population as a whole has succeeded in improving on the LeNet5 architecture, with the best performing network achieving 64.19% validation accuracy against 57.84% for LeNet5.

3 Conclusions

During the evolutionary process, it was noticed that networks that added layers could not establish themselves within the population. This was likely due to the simplicity of the EA and is why comparison is restricted to the LeNet5 architecture and not later, more modern architectures.

The results demonstrate two things. **Firstly**, the performance dip when layers are added impedes evolution from moving away from its starting condition. **Secondly**, the proposed GRN approach has proved stable enough to evolve the performance of networks when applied to the CIFAR10 dataset.

The motivation for this work is to apply a biological model of growth and development to the evolution of neural networks.

- The subsequent genome, based closely on nature, will generate a gene regulatory network to control growth.
- The effect of this is to detach the genome from the problem space and requires application specific mappings to connect the two.
- Growth is governed by an abstract cellular structure as an intermediate phenotype before mapping to the final form.

Further progress will depend on replacing the simple EA used here and finding ways to allow network innovations to establish themselves.

REFERENCES

[1] Brown, T. 2011 Introduction to Genetics: A Molecular Approach. Garland Science.
[2] Alon, U. 2019 An Introduction to Systems Biology: Design Principles of Biological Circuits, Second Edition. Chapman and Hall/CRC.
[3] Peter, I. S. and Davidson, E. 2015 Genomic Control Process - Development and Evolution 1st Edition. Academic Press.
[4] Cussat-Blanc, S., Harrington, K., and Banzhaf, W. 2018. Artificial Gene Regulatory Networks-A Review. Artif Life. 24, 4, 296-328. DOI=10.1162/artl_a_00267.
[5] Krizhevsky, A., Nair, V., and Geoffrey, H. CIFAR-10 and CIFAR-100 Datasets. https://www.cs.toronto.edu/~kriz/cifar.html.
[6] LeCun, Y., L., B., Y., B., and P., H. 1998. Gradient-based learning applied to document recognition. Proceedings of the IEEE. 86, 11, 2278-2324. DOI=10.1109/5.726791.