# Deep Learning with Darwin: Evolutionary Synthesis of Deep Neural Networks

## Background

A generic algorithm is a metaheuristic inspired by evolution. These algorithms are typically used to generate high-quality solutions for optimization. These algorithms leverage operations including mutations, crossover, and selection. Applied to machine learning hyperparameter optimization.

A typical generic algorithm requires:

1. A genetic representation of the solution domain – this can encode the appearance, behavior, or physical qualities of individuals to represent the solution that is evolving
2. A fitness function to evaluate the solution domain – an objective function that is used to evaluation the solution. The fitness function is used to drive to optimal solutions

Diagram

Description automatically generated

Psuedo code:

* Create a population of NNs
* Assign random hyper-parameters to NNs
* Run GA algorithms
  + Train
  + Evaluation Fitness (training cost or error)
  + Find maximum fitness in population
  + Pick 2 NNs
  + Cross-over the genes
  + Mutate the genes of the child
  + Repeat

## Motivation and background

Shafiee, Mishrea, and Wong take inspiration from genetic algorithms and explore the idea of “Can neural networks evolve naturally over successive generation into highly efficient deep learning networks?”.

The authors encode the neural networks using synaptic probability model to encode the genetic representation of the network. The authors believe that they can reach state of the art performance and develop networks that are more efficient.

Deep neural networks as the authors note “require high performance computing systems due to the tremendous quantity of computational layers they possess, leading to a massive quantity of parameters to learn and compute.” The continued complexity is also increasing driven the demand as we see in larger more complex models to boost accuracy.

## Research objective/question(s)

“Can neural networks evolve naturally over successive generation into highly efficient deep learning networks?”.

Specifically, the authors explore environment factors to force the optimization over number of synapses in a neural network making that the primary lever of fitness in addition to the accuracy measures.

The authors believe with a genetic algorithm then can create “state-of-the-art” performance while having efficient network architectures.

## Prior relevant work/literature gap

Prior literature has focused in improving the accuracy of training of deep neural networks the authors note this is the first application against the notion of focusing on evolutionary synthesis (focus on the synapse).

The earliest article from GMS and Pollack (1994) applied evolutionary algorithms which looked at the structure and weights.

The other inspiration came from Stanley, Bryant, and Miikkulainen (2005) applying evolution to evolving neural networks in real time (as a game was being played). This also focused on the weights (the gene was encoded as in-node, out-node and weight of the connection), Mutation in this approach also afforded weight and network structure changes.

## Theory, conceptual framing

Diagram

Description automatically generated

Evolutionary synthesis process to derive highly efficient deep neural networks.

1. The architectural traits of the ancestor network are modeled with probabilistic in the form of synaptic probability model
2. At each generation a new offspring of deep neural networks is synthesized in stochastic manner (mimic random mutation) based on environmental factors (mimic natural selection) and probabilistic DNA of ancestors (mimicking heredity)

## Methods

The authors propose using an evolutionary synthesis of deep neural networks. This is inspired by genetic algorithms based on evolution.

The following mechanisms are leveraged as part of the research:

1. Heredity
2. Natural selection
3. Random mutation

To drive diverse and enhanced traits in later generation the authors realize computational constructs to mimic the system of passing traits down from generation through generation leveraging these methods.

1. Heredity – here this idea of heredity was modeled by encoding the architectural traits of deep neural networks in the form of synaptic probability models.
   1. H = (N, S) denotes the possible architecture (N – Neurons, S – set of synapses) with sk ∈ S denoting a synapse between two neurons (ni , n j ) ∈ N.
   2. Treating areas of strong synapses in an ancestor network in generation g as desirable traits - Synaptic probability where Wg-1 encodes the synaptic strength of each synapse in Sg-1.

A picture containing text, clock, watch

Description automatically generated

1. Natural selection and random mutation
   1. The authors mimic environmental condition to drive natural selection through random models.
   2. Synapse is synthesized randomly between two possible neurons in a descendent network based on P(Sg|Wg−1) and an environmental factors F(E)

A picture containing text

Description automatically generated

* 1. The environmental factor is imposed on the descendant network

1. Efficiency-driven Evolutionary synthesis
   1. One of the environment factors in encouraging efficiency is restricting resources available (to encourage highly efficient deep neural networks)
   2. To survive the NN must therefore factor more efficient energy consumption than its original ancestor network to survive

A picture containing text, watch, clock, gauge

Description automatically generated

* 1. C is the highest percentage of synapses desired

## Results

The environmental constraint imposed during synthesis in this study introduced that no descendent network should have more than 40% of the total number of synapses in its direct ancestor (C=.4).

The evolutionary algorithm was tested against two dataset MSRA-B (5000 images used in salient object detection) and HKU-IS (4447 challenging images used in salient object detection).

The first generation was based on VGG16 (a very deep convolutional neural network).

Table

Description automatically generated

* Simonyan Zisserman (2014) A Very Deep convolutional network

Table

Description automatically generated A picture containing text, skating, sky, outdoor

Description automatically generated

Performance was evaluated using Mean absolute errors (MAE) measuring the saliency result to the ground truth (represented by the ground truth map). The Fβ score is the evaluation metric for binary classification (both positive and negative misclassification at the same time for wrong pixel annotation).

The detailed experimental results (success measured in the reduction of synapses) provide the following insights:

1. The performance differences (MSRA-B) from one generation to the next is small (<3%)
2. The performance difference (HKU-IS) from first two (<.5%) and (<8%) by fourth
3. The second and third generation produce great results while having ~18 fold decrease in synapses

## Discussion

An interesting observation was the importance of traits (ability to identify salient objects versus ability to distinguish fine-grained saliency) as was noted in the HKU-IS dataset. To drive efficient architectures, it is hypothesized that the larger networks are needed for the fine-grained saliency.

In many problems GAs have a tendency to converge towards local optima and really dependent on the fitness landscape. Emphasizing the need to ensure the definition of fitness is leading to the optimization. For example, the authors optimized the network at the cost of fine-grained saliency. Furthermore, definition of the fitness function is not straightforward and often require that they be iteratively developed when the genetic algorithms are not performing

## Thoughts

The authors submit a novel idea of leveraging a synapse probability model and leveraging generic algorithms through an environmental factor to drive efficient networks up to almost a ~48-fold decrease that still performs well. The results were comparable and performed well and met the research objective. It would be interesting to understand other approaches to the genetic representation of the network. The authors imply the synapse strength is what maps to the traits of the network (e.g. saliency traits) versus the network architecture (layers), it would be interesting to look at inherent traits of the network architecture in addition to the traits of the neurons, possibly combining some of the real-time approach in the prior literature applied to game programing and synapse strength.