

Genetic Convolutional Neural Network

Ashish Sethi (MT18024)

Kshitij Srivastava (MT18099)

Vishal (MT18019)

I. INTRODUCTION

In recent times, CNNs have emerged as the go-to technique for image classification problems. However, they rely highly on their architecture, which needs to be hand crafted by a personnel having expertise in both, CNN and the respective problem. This makes it difficult for researchers who have no expertise in CNNs, to explore their full potential. We will try to implement an already proposed algorithm, which is capable of discovering a potentially good architecture for a CNN best suitable for image classification. The proposed algorithm uses GA as its core to explore networks.

For a large scale network usually higher depth networks along with vvariable innner structures are used. These structures are designed manually but still most of these structures are fundamentally similar. In order to introduce flexibility we are using automated process for finding networks.

structure. Automating this process with the help of GA helps in finding better network structure.

III. OUR APPROACH

For both encoding we add a default convolution layer as top layer and two fully connected layer at the back of the network.

A. Encoding 1

- Initialize a random population of 10 chromosomes, each chromosome has four genes.
- If value of gene is 1 then it represents a convolution layer and if it is 0 then it represents a pooling layer.
- Generate a CNN model using this chromosome.
- Evaluate this CNN model based on its fitness value.
- Fitness function for GA is classification accuracy of the network.
- Select best chromosomes and repeat until convergence.

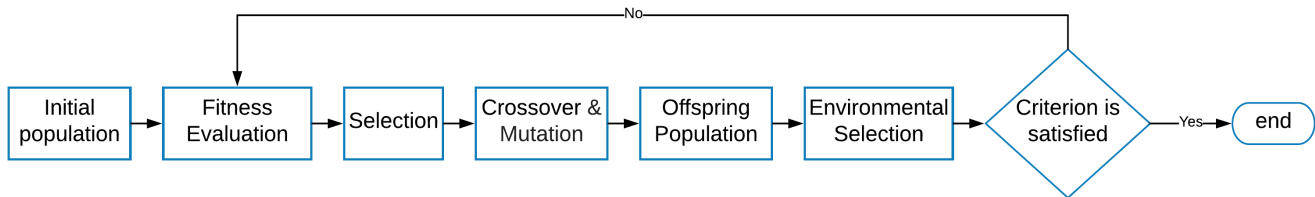


Fig. 1: Block Diagram of Algorithm

II. LITERATURE REVIEW

L.Xie and A. Yuille proposed a fixed length encoding method to represent neural networks. By using genetic operations they automated the process of exploring better networks. Fitness of an individual is calculated by training the decoded network on MNIST data set. Their network consists of S stages and any stage s has Ks internal nodes. All nodes in a stage are ordered and is a convolution operation. [2]

A stage with 4 internal nodes, needs $0.5 * K_s * (K_s - 1)$ bits to represent string. A possible encoding can be 1-00-111. 1 represents that node 1 is connected to node 2, 00 means node 1 and node 2 are not connected to node 3. 111 represents, node 1, node 2 and node 3 are connected to node 4. Connections are only allowed from a low numbered to a high numbered node. They used 2 default nodes, default input which receives input from previous stage and a default output node which gives output to next stage. [1].

Even for a simple network we can have very large number of network structures. It is practically impossible to find best

B. Encoding 2

- Initialize a random population of 10 chromosomes, each chromosome has four genes of size 2.
- First value in gene defines the type of layer, 1 for convolution and 0 for max pool layer.
- For convolution layer, if second value in gene is 1 then we use a stride of 2 and padding of size 3. For max pool layer we ignore second value.
- Generate a CNN model using this chromosome.
- Evaluate this CNN model based on its fitness value.
- Fitness function for GA is classification accuracy of the network.
- Select best chromosomes and repeat until convergence.

IV. DATA SET

- MNIST: A database of handwritten digits, has a training set of 60,000 examples, and a test set of 10,000 examples. The digits have been size-normalized and centered in a fixed-size image.

- CIFAR-10: The CIFAR-10 data set consists of 60000 32x32 colour images in 10 classes, with 6000 images per class. There are 50000 training images and 10000 test images.

V. PROJECT STATUS

A. Task Completed

- Implemented LeNet like structure encoding for chromosome.
- Proposed two encoding for chromosomes.

B. Future Work

- We are using only convolution and pooling layers for networks. This limits the complexity of networks that can be generated by our approach.
- Complex encoding techniques can be developed by adding more convolution operations and can handle large visualization problems.
- A fitness function can be defined which can also incorporate training time and not just accuracy of the network.

VI. RESULTS

- Encoding 1:
Best chromosome = [1, 1, 1, 0]
Accuracy on test MNIST Dataset-98.01 %
Accuracy on test CIFAR-10 Dataset-98.39%

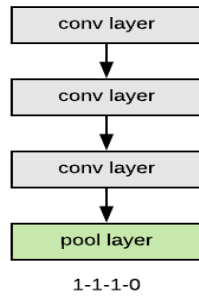


Fig. 1: Structure of best chromosome

Layer (type)	Output Shape	No.of Params
Conv2d-1	[4, 20, 26, 26]	200
Conv2d-2	[4, 40, 24, 24]	7,240
Conv2d-3	[4, 60, 22, 22]	21,660
Conv2d-4	[4, 80, 20, 20]	43,280
Pool2d	[4, 80, 10, 10]	-
Linear-5	[4, 500]	4,000,500
Linear-6	[50, 10]	5,010

- Encoding 2:
Best chromosome = [1, 0, 1, 1, 1, 0, 0, 1]
Accuracy on test MNIST data set - 98.75 %
Accuracy on test CIFAR-10 data set - 57.17%

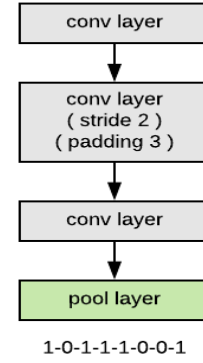


Fig. 2: Structure of best chromosome

Layer (type)	Output Shape	No.of Params
Conv2d-1	[4, 20, 26, 26]	200
Conv2d-2	[4, 40, 24, 24]	7,240
Conv2d-3	[4, 60, 14, 14]	21,660
Conv2d-4	[4, 80, 12, 12]	43,280
Pool2d	[4, 80, 6,]	-
Linear-5	[4, 500]	4,000,500
Linear-6	[50, 10]	5,010

VII. OBSERVATION

- Convolution layers are preferred over pooling layers.
- Network with all pooling layers is discarded.
- Networks with large depth suffer from over fitting.

VIII. CONCLUSION

- Convolution layers find features and affects the overall accuracy. This explains why conv layers are preferred.
- Proposed encoding are able to classify images with great accuracy.

REFERENCES

- [1] Lingxi Xie and Alan L. Yuille. Genetic CNN
- [2] Yanan Sun, Bing Xue, Mengjie Zhang, and Gary G Yen. Automatically designing CNN architectures using genetic algorithm for image classification.