**MANISH ANAND PAWAR**

**ID NO : 16301023**

**TE COMPS**

**mail : i\_am\_manish@outlook.com**

**AI that builds AI (autonomous Deep learning)**



While building ConvNets for image classification,we’ve to go through troublesome handcrafted feature extraction with layers suiting every dataset.

**Instead why not build a CNN model that generates other CNN models according to need?** A recent paper published at School of Computer Science, NPU has emereged with an idea to do this.

**Proposed genetic DCNN designer involves feeding it randomly initialized population(each encoded).Based on current generation,new generation is produced by performing a combination of operations called selection, crossover and mutation and iter through them till accomplishment point.**

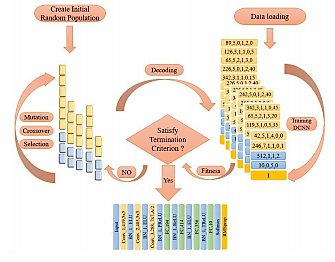
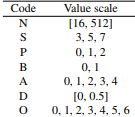
Figure-1:Typical DCNN designer

Figure 1 gives an insight to what DCNN generator looks like.

They have evaluated on six image classification tasks using the **MNIST, EMNISTDigits, EMNISTLetters,FashionMNIST,CIFAR10,CIFAR100** datasets.

Lets go step by step.

STEP 1: ***ENCODING SCHEME AND INITIALIZATION***

Value range of each parameters for DCNN (#O=optimizers)

Here,convolutional blocks compose a **convolutional arm** and fully connected blocks compose a **fully connected arm**. Convolutional block contains 6 loci in sequence and can be encoded as-**NSPBAD (N=filters,S=filter\_size,P=Pooling,B=Batch\_normalization, A=activation,D=dropout),**whereas fully connected block consist of **NBAD**(N=no. of neurons).For eg,NSPBAD=[64,3,0,1,4,0] respectively

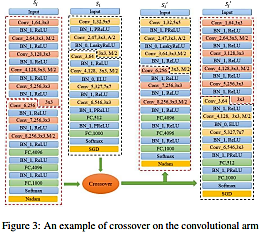
https://cdn-images-1.medium.com/max/660/1*m7dy8X0VD569XQNiHP2UMA.png figure 2: representation

A DCNN with N^C(n)convolutional blocks and N^F(n) fully connected blocks is presented as figure 2.

STEP 2: ***SELECTION***

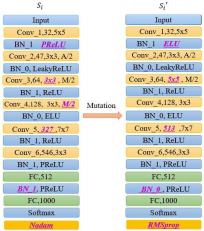
Before producing the next generation,we evaluate individuals fitness. Based on the fitness ranking, we use the **elitism roulette wheel selection** scheme(<https://pdfs.semanticscholar.org/feee/c4229f71c6ed155e2f2b732464dbc8c5b93c.pdf>) to select **0.1%** of top ranking elites from current generation to carry over to next and select **0.9%** for subsequent genetic operations.

STEP 3: ***CROSSOVER***

Si & Sj producing Si’ & Sj’

For a pair of selected DCNNs,S(i)&S(j),we randomly locate a cross point(k) on each of them,which breaks the architecture into two segments. By swapping the segments of those 2 DCNNs,2 new DCNNs are generated,whose depths may be different from the depths of their parents.If u wanna go deeper into code-lengths of each,click [here](https://arxiv.org/pdf/1807.00284.pdf).

STEP 3: ***MUTATION***

An example of Mutation

To maintain genetic diversity from one generation to the next,mutation operation is applied to each individual, altering an DCNN architecture.Its just changing some parameters of N,P,B,A,S,D.

**RESULTS :**

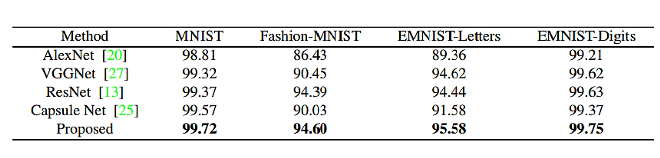
figure 4 : Classification accuracy of four DCNNs (%)

Figure 4 shows that the model which our DCNN designer made **outperformed** famous models like AlexNet,ResNet,etc. on MNIST,Fashion-MNIST,EMNIST- digits/letters datasets.

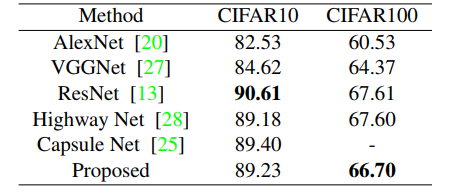
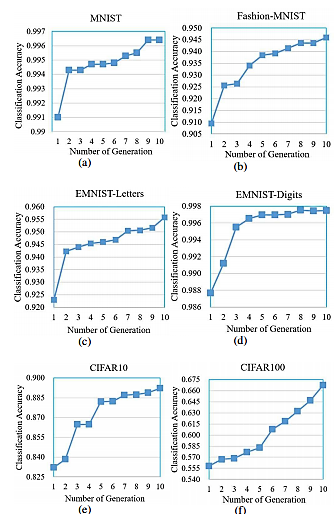
figure 5 : Classification accuracy on CIFAR10 and CIFAR100 (%)

figure 5 similarly shows on CIFAR10/100 datasets and proposed did quite good there.

Thus, highest classification accuracy achieved in each generation(1 generation=100 epochs)on each dataset is shown below.



The experiments were conducted using server with *2-Intel Xeon & 8-NVIDIA Titan GPUs*. It would perform better for more epochs and computational power.

link to paper : <https://arxiv.org/pdf/1807.00284.pdf>