|  |  |
| --- | --- |
| nature inspired cnn optimization | gROUP members  14140044 Rida Yaqoob  15140094 Mohsin Ashraf  14140062 Mushood Hanif  15140107 Mehvish Hameed  Supervisor  Sayed Qamar Askari |

Introduction

A **C**onvolutional **N**eural **N**etwork is a deep neural network whose architecture most commonly contains several convolutions, pooling and fully connected layers. Several recent studies focus on developing a novel CNN architecture that achieves higher classification accuracy e.g. Google Net, ResNet and DesNet. Despite their success, designing CNN architectures is still a difficult task because many design parameters exist such as the depth of a network, the type and parameters of each layer and their connectivity.

CNN architectures have become more complex suggestive of the fact that a significant number of design parameters have to be tuned to realize best performance for the given dataset. In light of this situation, automatic design methods for CNNs are highly beneficial.

Algorithms inspired by evolution and other natural phenomenon have been traditionally applied to designing CNN architectures. Evolution based algorithms basically mimic Darwinian evolution by implementing concepts like “survival-of-the-fittest”, crossover and mutation over a significant number of generations. Metaheuristics are algorithms that mimic basic and instinctual actions performed by various living entities for survival e.g. PSO (Particle-swarm optimization). These algorithms have the ability to not depend on a known goal state for progression and also due to their dynamic nature, encoding CNN architectures into their population elements is relatively easy.

CNNs have become an area of highly concentrated research in terms of finding an optimal architecture for a given dataset and much progress has been made in terms of this field. In this paper, we propose a novel method to optimize CNNs using nature-inspired metaheuristics as specified in section 4. The next section defines related work previously done towards this research field that we’ve referred to. Section 5 will shed light on our testing methods and results and section 6 will be the conclusion and description of the future work we plan to do.

**IMAGENET**

Deep convolutional neural networks were trained to classify 1.2 million images in the ImageNet LSVRC-2010 contest into 1000 different classes. The CNN, which consisted of 60 mill. Parameters and 650,000 neurons and 5 convolutional layers and a final 1000-way softmax layer performed exceptionally and gave results ranked top-1 and top-5 in error rates.

To improve their performance, they have collected larger datasets, used more powerful models, and used better techniques for preventing overﬁtting. For example, the best error rate at the time on the MNIST digit-recognition task (<0.3%) approached human performance. But objects in realistic settings exhibit considerable variability, so to learn to recognize them it is necessary to use much larger training sets.

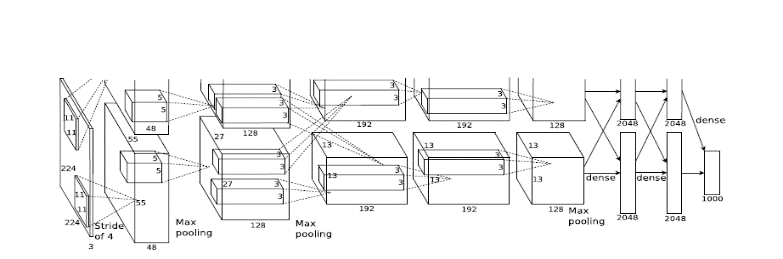
They trained one of the largest convolutional neural networks to date on the subsets of ImageNet used in the ILSVRC-2010 and ILSVRC-2012 competitions and achieved by far the best results ever reported on these datasets.

The images were collected from the web and labeled by human labelers using Amazon’s Mechanical Turk crowd-sourcing tool. Starting in 2010, as part of the Pascal Visual Object Challenge, an annual competition called the ImageNet Large-Scale Visual Recognition Challenge (ILSVRC) has been held. ILSVRC-2010 is the only version of ILSVRC for which the test set labels are available, so this is the version on which we performed most of our experiments. ImageNet consists of variable-resolution images, while our system requires a constant input dimensionality. Therefore, down-sampled the images to a ﬁxed resolution of 256 × 256.

Deep convolutional neural networks with ReLUs train several times faster than their equivalents with tanh units. This is demonstrated in, which shows the number of iterations required to reach 25% training error on the CIFAR-10 dataset for a particular four-layer convolutional network

If at least some training examples produce a positive in putto a ReLU, learning will happen in that neuron. However, they still ﬁnd that the following local normalization scheme aids generalization.

Now they are ready to describe the overall architecture of our CNN. As depicted in Figure 2, the net contains eight layers with weights; the ﬁrst ﬁve are convolutional and the remaining three are fully connected. Our network maximizes the multinomial logistic regression objective, which is equivalent to maximizing the average a cross training cases of the log-probability of the correct label under the prediction distribution.



The easiest and most common method to reduce overﬁtting on image data is to artiﬁcially enlarge the dataset using label-preserving transformations. They employ two distinct forms of data augmentation. The ﬁrst form of data augmentation consists of generating image translations and horizontal reﬂections. We do this by extracting random 224×224 patches and their horizontal reﬂections from the 256×256 images and training our network on these extracted patches. The second form of data augmentation consists of altering the intensities of the RGB channels in training images.

They trained their models using stochastic gradient descent with a batch size of 128 examples, momentum of 0.9, and weight decay of 0.0005

They initialized the weights in each layer from a zero-mean Gaussian distribution with standard deviation 0.01. They initialized the neuron biases in the second, fourth, and ﬁfth convolutional layers, as well as in the fully-connected hidden layers, with the constant 1

