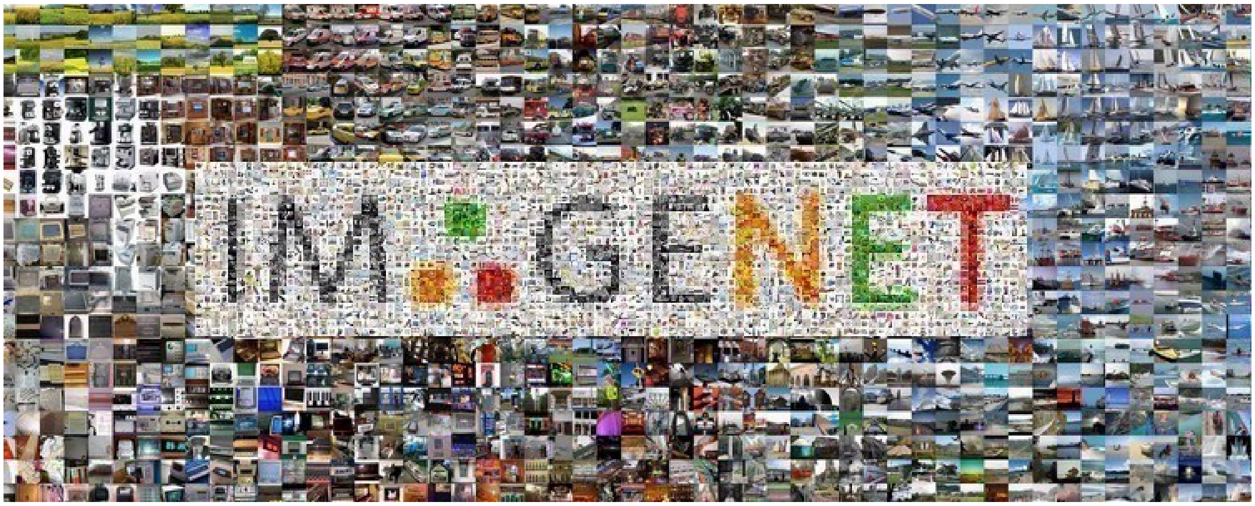

0x00 - CNN Task 6: Journal Response

ImageNet Classification with Deep Convolutional Neural Networks

Aaron Manuel
4919@holbertonstudents.com

Introduction

This journal article covered the implementation and results of a Convolutional Neural Network trained on a subset of images from the ImageNet dataset used in the LSVRC-2010 contest. The training data consisted of 1.2 million images that were classified into 1000 different classes. The architecture of the network consisted of 8 main layers: 5 convolutional layers and 3 fully connected layers with the final being fed into a softmax function. The intention at the time was focused on the belief that a bigger network would yield better results if it was computationally possible. Overall, the architecture and procedures used yielded the highest recorded success at the time of testing. The metrics of top-1 and top-5 error rates were used for ranking in this competition.

Procedures

The procedures used for this paper included several neural network calculations as well as data processing. Formatting of the data was necessary in order to apply any image in the set to the network. Since ImageNet contained variable image sizes, the images needed to be mapped to a standard 256x256 pixel size. This was done by formatting the shorter side of each image to 256, and then taking the centered 256 from the other dimension. The handling of images smaller than 256x256 was not specified.

The architecture of the network used ReLU activations, cross GPU-parallelization, normalization of input activations, and overlapping pooling. ReLU was chosen over tanh due to speed of computation which ultimately yielded a higher result in the timeframe they could manage to run. Cross GPU-parallelization was a bit of a new thing at the time but essentially was the use of two GPUs running simultaneously with different kernels on the same image and then reading from each other to feed the data back into the fully connected layers at the end of the network. Only on convolutional layer 3 do they read from each other before the fully connected layers. Tweaking which layers pulled from which GPU allowed processing speed to be maximized. Normalizing the input data was not considered necessary but aided with generalization and ultimately reduced error by 1.4% and 1.2% with respect to top-1 and top-5 error rates. Overlapping Pooling was a new term for me but I gathered that it effectively was having a stride that was smaller than the kernel so that each pixel contributed to more than one calculation during pooling.

Additionally, to reduce overfitting, they used two different types of data augmentation to create a larger data set from the original data set. Using python on their CPUs with the intent of being computationally free, the original data set had random 224x224 pixel patches extracted from them along with their horizontal reflections. This created a larger set by a factor of 2048 and allowed for their network to be larger. The second form of augmentation was a PCA on the pixel values. Essentially, this was scaling the pixel values by their eigenvalues so the image was the same with different RGB values and effectively created a more generalized data set from the same images so that the patterns were apparent and not just the specific images themselves.

Lastly, Dropout was implemented. At the time, this was very new and contributed a lot to the reduction of error.

Results

The results of this network yielded much higher performance in comparison to the other competitors. Their CNN had Top-1 and Top-5 error rates of 37.5% and 17.0% respectively and were 8.2% and 8.7% lower than the nearest competitor. With this success in mind, the CNN architecture was also trained on other ImageNet data sets from other competition years and they also yielded higher performance than other recorded tests. The architecture in some cases had an additional 6th Convolution layer.

Conclusion

At the time of its writing, this study was state-of-the-art in terms of performance on ImageNet. It serves as clear proof that larger neural networks have an advantage over smaller ones as it was shown that any smaller network yielded worse performance. With a larger data set and a larger network, the overall performance was increased by having better generalization of what the images contained. This was shown in a figure explaining the softmax results of specific pictures as well as an image likeness example.

Personal Notes

I learned a lot with this paper. Reading an academic journal entry was definitely beneficial and seeing the time stamp for this one in particular really puts in perspective the advancements that have been made in such a short amount of time. This was my first time seeing a study that specifically showed the importance of each bit of the network architecture as a percentage of its performance. VERY HELPFUL. I would like to see more of this and understand the scope of computation time with a competition like this.