Review of Krizhevsky et al. (2012)

Matthew Evans

February 20, 2025

Overview

In *ImageNet Classification with Deep Convolutional Neural Networks*, Krizhevsky et al.[1] propose a novel machine learning architecture for object recognition within images. Their approach leverages a combination of techniques which enable the system, though complex (i.e., high parameter count), to generalize well (i.e., low variance). Their work is showcased in the context of the ImageNet LSVRS-2010 contest[1].

Approach

The authors identify four specific design choices, detailed below, which enabled their model to achieve state of the art performance in Top-1 and Top-5[2] accuracy.

ReLU Nonlinearity

Like other learning models, neural networks learn parameters by iteratively minimizing a loss function. The magnitude and direction of a parameter's adjustment at each iteration is primarily dictated by the slope of the gradient of the loss function at the parameter values computed in the preceding iteration. The loss function being minimized is a linear combination of the network's features X and weights W, i.e., $W^T X$. To enable the network to classify *non*-linearly separable data, the linear combination is passed through a non-linear *activation function* such as $\tanh(x)$ 1 or the sigmoid function 2, both of which are continuous and differentiable on $\mathbb{R}$. However, at x values far from the origin, both of these classical activation functions have gradients very close to zero (i.e., they are nearly flat), and as a result, the weight adjustment from iteration to iteration is virtually zero, causing the network to stop learning. This is known as the vanishing gradient problem, and, at these extreme x values, we say that the function is *saturated*.

For deep networks with many parameters, such as AlexNet, the vanishing gradient problem makes learning nearly impossible at deeper layers. To overcome this, the authors leverage the *non*-saturating *Rectified Linear Unit* function, or **ReLU** 3. In addition to avoid saturation and the vanishing gradient problem, which accelerates learning, the ReLU function is computationally simpler than either of the exponential classical functions mentioned, further reducing training time.

Training on Multiple GPUs

Learning models, particularly in object detection, have historically been constrained by the availability of labeled training data. The ImageNet LSVRC-2010 dataset, with its 1.2 million human-labeled images, provided the scale needed for training the AlexNet model. However, the sheer size of this dataset and the model's parameter count demanded parallel processing across two GPUs. While this dual-GPU architecture enabled training a more sophisticated network than possible on a single GPU, it introduced communication overhead between the processors. To optimize performance, the authors selectively limited inter-GPU connections between certain layers, using cross-validation to balance the improved accuracy of a more complex model against the computational costs of GPU communication.

[1] The LSVRS is the Large Scale Visual Recognition Challenge, an annual competition in which teams compete to achieve the highest accuracy on several computer vision tasks.

[2] Top-1 accuracy measures the percentage of cases where the model's highest probability prediction matches the correct label. Top-5 accuracy measures the percentage of cases where the correct label appears among the model's five highest probability predictions.

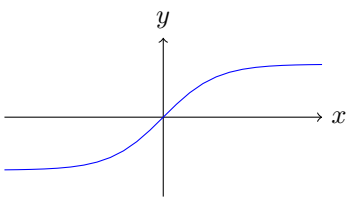

Figure 1: tanh(x)

$$f(x) = \tanh(x)$$

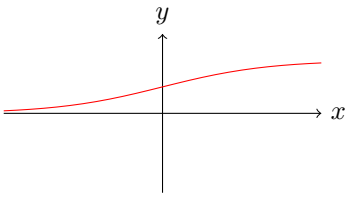

Figure 2: sigmoid(x)

$$f(x) = \frac{1}{1 + e^{-x}}$$

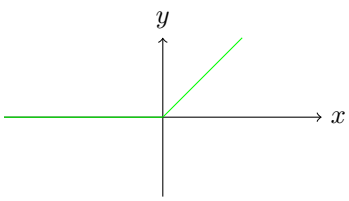

Figure 3: relu(x)

$$f(x) = \max(0, x)$$

Local Response Normalization

While ReLUs do not require input normalization to prevent saturation, the authors found that implementing a local normalization scheme improved the model's ability to generalize. Their approach, called Local Response Normalization (LRN), creates competition between neuron activations at the same spatial position across adjacent feature maps.

Given a neuron activation, the normalized response is calculated using adjacent kernel maps. This normalization scheme implements a form of lateral inhibition, similar to that observed in biological neurons, where strong activations inhibit neighboring neurons' responses.

Overlapping Pooling

Pooling layers in convolutional neural networks reduce dimensionality by summarizing (i.e., down sampling) groups of neighboring neurons within a feature map by returning the maximum value within each group.

For example, consider the following 4x4 feature matrix and its max-pooled representation with a 2x2 window and stride of 2:

$$\begin{bmatrix} 6 & 2 & 3 & 8 \\ 5 & 1 & 7 & 4 \\ 9 & 10 & 11 & 16 \\ 13 & 14 & 15 & 12 \end{bmatrix} \xrightarrow{\text{max-pool}} \begin{bmatrix} 6 & 8 \\ 14 & 16 \end{bmatrix}$$

In traditional approaches, these summaries are computed over non-overlapping regions, such as in the above example. The authors instead propose an overlapping scheme where the stride length s between pooling operations is smaller than the size z of the pooling window (specifically, $s = 2$ and $z = 3$). This creates a situation where each pooling operation shares input values with adjacent pooling operations. In addition to improved prediction accuracy, the overlapping scheme appeared to provide a regularizing effect, making the model more resistant to overfitting.

Reducing Overfitting

The authors' model contained 60 million parameters, making it particularly susceptible to overfitting without specific mitigating techniques, which they addressed through two key strategies.

Dropout

Dropout is a regularization technique which randomly deactivates a proportion of neurons (and their connections) during training, effectively creating an ensemble of thinned networks. During each training iteration, the authors set the output of each hidden neuron to zero with probability 0.5, forcing the network to learn with only half of its neurons active, which prevents complex co-adaptations (i.e., neurons becoming overly dependent on specific neurons in previous layers) and reduces overfitting. At test time, all neurons are used but their outputs are multiplied by 0.5 to compensate for the fact that twice as many of them are active compared to training time.

Data Augmentation

To artificially expand their training dataset, the authors employed label-preserving transformations of the training images. The first approach involved extracting random 224x224 patches from the (already down-sampled) 256x256 images and horizontally flipping them, effectively creating a more diverse set of training examples. Additionally, they altered the RGB pixel intensities using PCA to approximate natural variations in illumination, further augmenting the dataset's variety while maintaining label validity.

Considerations

The authors' approach, while groundbreaking, faced several notable limitations.

- **Computational Cost:** The training process required 5-6 days, making hyperparameter optimization extremely time-consuming.
- **Input Constraints:** The architecture required fixed 256x256 pixel inputs, necessitating image down-sampling and cropping that resulted in loss of potentially valuable information.
- **Overfitting Risk:** With 60 million parameters, the model was highly susceptible to overfitting, requiring additional techniques like data augmentation and dropout for regularization.

Impact

AlexNet's publication marked a pivotal moment in computer vision and deep learning. The architecture demonstrated unprecedented accuracy on the ImageNet dataset, reducing the error rate by nearly half compared to previous methods. This success helped trigger the deep learning revolution, leading to widespread adoption of deep neural networks across various domains. The paper's innovative techniques—particularly ReLU activation, dropout regularization, and GPU implementation—became standard practices in neural network design. As of this writing, the paper has garnered over 170,000 citations, reflecting its foundational influence on modern machine learning architectures and practices.

References

[1] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In F. Pereira, C.J. Burges, L. Bottou, and K.Q. Weinberger, editors, *Advances in Neural Information Processing Systems*, volume 25. Curran Associates, Inc., 2012.