Regularization Techniques for Image Classification Tasks

Aashka Trivedi (aht323), Natalia Zubkova (nz2094)

1 Project Description

This project aims to empirically understand the effects of introducing the following four regularization techniques- Batch Normalization [1, 2], Dropout [3], Data Augmentation [4] and Adaptive Gradient Clipping [5]- to a Resnet-50 Model. We evaluate the performance of the model on an Image Classification task using the CIFAR-10 Dataset.

The major goal of this project is to determine a way to use the above mentioned techniques to reduce overfitting and improve the accuracy of a self-implemented Resnet. A second goal of our project is to study whether Adaptive Gradient Clipping [5] can be used as an effective replacement of Batch Normalization.

2 Approach

There are two phases of this project- the Research phase, and the Implementation Phase. The approach to both these phases are discussed below.

Research Phase

This phase involves exploring different combinations of regularization parameters to obtain the best configuration. Here we study the effects of the following regularization techniques and their respective parameters on a Resnet-50 model implemented from scratch:

1. Batch Normalization: We mainly observe how the different metrics are effected when we introduce a batch normalization layer to different Resnet blocks.

2. Dropout: We introduce dropout layer after different Resnet blocks, and test the effect that different dropout probabilities have on metrics. Specifically, we define two types of models- those with *Symmetric Dropout*, which have the same dropout probability across all layers, and those containing *Asymmetric Dropout* and have different dropout probabilities across the different layers.

3. Adaptive Gradient Clipping: We seek to test Brock et. al.'s [5] use of adaptive gradient descent as a regularization technique in Normalization-Free Resnets. For the same, we conduct a thorough ablation study on the effects of the clip value threshold and batch size on performance.

The performance of different models is measured by their accuracy on the test set, training time for 200 epochs and time to reach 87% training accuracy. We choose the best model configurations from each training pool to analyze in the implementation phase.

Implementation Phase

Here, we implement a Resnet-50 model from scratch, train it till convergence using the best configuration from the research phase, and evaluate its performance by analyzing the training accuracy, testing accuracy and time to train. In this phase, we seek to empirically determine the following key points:

1. Adaptive Gradient Clipping as an alternative to Batch Normalization
2. Varying the dropout probability across layers
3. Data Augmentation Techniques (specifically, image transformations and cutout regularization) to effectively increase the dataset size and provide regularization
4. Comparison of overfitting in a baseline model and a model with regularization techniques applied

The best models are evaluated on the CIFAR-10 Dataset.

3 Implementation Details

A brief overview of the implementation details are as follows.

Resnet 50 Implementation: Our Resnet 50 implementation consists of Identity blocks and Convolution blocks which forms the basis of out Resnet Blocks. To maintain the modular approach of the original Resnet Models, we have maintained that once we set the parameters (e.g., number of dropout and batchnorm layers, dropout probability,etc) for the Identity and Convolution blocks, these parameters will not change with different instanitations in the Resnet blocks.

Regularization Techniques: The different parameters tested to find the best candidates for Batch Normalization, Dropout and Adaptive Gradient Clipping is given in Table 1.

Regularization Technique	Parameter	Candidates
Batch Normalization	Number of Layers	0,1,2,3
Dropout	Number of Layers Symmetric Dropout Probability Asymmetric Dropout Probability	0,1,2,3 0.2,0.5,0.8 [0.1,0.8] (continuous range)
Adaptive Gradient Clipping	Clip Value Batch Size	0.01,0.02,0.04 64, 128, 256
Cutout Regularization	Cutout Images per Batch	2,8,6

Table 1: Parameters tested to analyze different Regularization Techniques

Framework and Hardware: The entirety of this project has been coded using Tensorflow. Every model has been trained on a Nvidia Tesla V100 GPU, on the Deep Learning VM solution of Google Cloud.

4 Experimental Results

The experiments performed to understand the effects of each regularization technique, followed by a brief discussion on the observations are given in this section.

Dropout

Batch Normalization

Adaptive Gradient Clipping

Data Augmentation

5 Observations

A comparison of the Resnet50 Models with the best parameter candidates are shown in Figure 1. Here, we compare 6 models-

1. The baseline model with no regularization

2. A model with only batchnormalization and symmetric dropout. This contains 2 batch normalization layers and 3 dropout layers with a dropout probability of 0.2 across all layers.

3. A model with symmetric dropout and data augmentation. Image transformations are used to augment the dataset, which contains 2 batchnorm layers and 3 dropout layers with a 0.2 dropout probability across layers.

4. A model with asymmetric dropout and data augmentation. Image transformations are used to augment the dataset, which contains 2 batchnorm layers and 3 dropout layers with a 0.1, 0.2 and 0.3 dropout probability across each layer.

5. A model with symmetric dropout and cutout regularization. The model contains 2 batchnorm layers and 3 dropout layers with a 0.2 dropout probability across layers. The number of cutput images per batch is 2.

6. A model with asymmetric dropout and cutout regularization. The model contains 2 batchnorm layers and 3 dropout layers with a 0.1, 0.2 and 0.3 dropout probability across each layer. The number of cutout images per batch is 2.

Here, we can see that the baseline model gives the worst performance, while we achieve the best accuracy with the model with asymmetric dropout and data augmentation. It can also be seen that it has the lowerst overfitting, as it has the elast difference between the training and testing accuracy.

6 Conclusion

We are able to increase our test accuracy form 69% to 85% using only batch normalization, dropout and data augmentation.

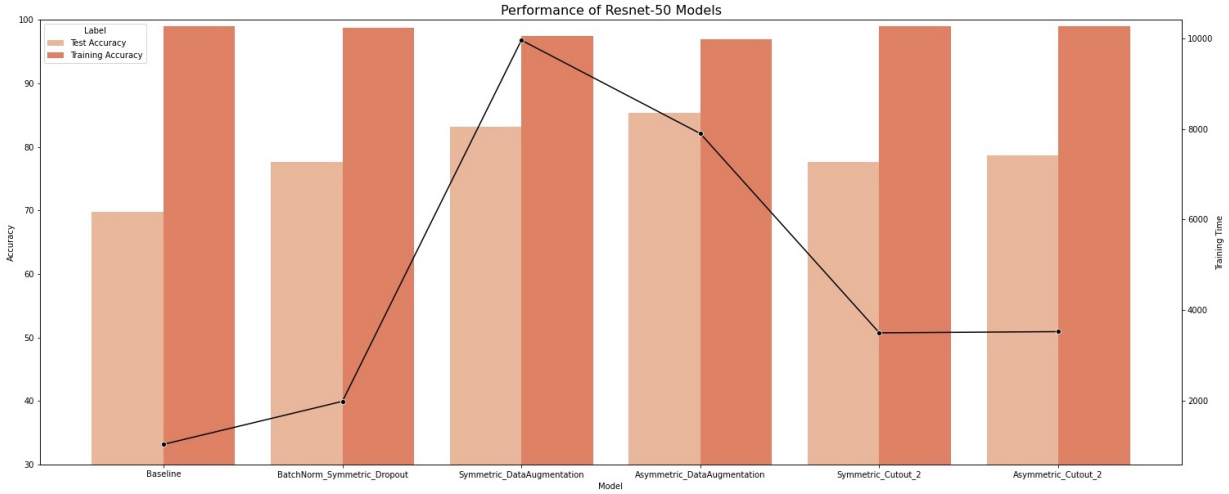

Figure 1: Comparison of Final Models

References

[1] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," 2015.

[2] J. Bjorck, C. P. Gomes, and B. Selman, "Understanding batch normalization," *CoRR*, vol. abs/1806.02375, 2018. [Online]. Available: http://arxiv.org/abs/1806.02375

[3] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: A simple way to prevent neural networks from overfitting," *Journal of Machine Learning Research*, vol. 15, no. 56, pp. 1929–1958, 2014. [Online]. Available: http://jmlr.org/papers/v15/srivastava14a.html

[4] T. DeVries and G. W. Taylor, "Improved regularization of convolutional neural networks with cutout," 2017.

[5] A. Brock, S. De, S. L. Smith, and K. Simonyan, "High-performance large-scale image recognition without normalization," 2021.