Analysis of Semi-Supervised Learning and Data augmentation Techniques

Jayesh Kudase, Prashant Jadhav, Aditya Chavan
Arizona State University, Tempe, AZ, USA
jkudase@asu.edu, prjadhav@asu.edu, apchavan@asu.edu

Abstract

The paper will provide a brief overview of semi-supervised learning experiments along with data augmentation conducted on the CIFAR-10 dataset. The main objective here is to analyze the performance of the semi-supervised classification using data augmentation technique using a framework of supervised classification. The experiments are performed by varying the number of labeled and unlabeled samples in the training data set and testing the performance of the developed machine learning model using the test set. Finally, we provide the analysis from our results in the form of confusion matrices, classification accuracies, and loss function plots. It can be evidently seen from the results that the data augmentation along with entropy-minimization performs very efficiently for a semi-supervised model in a supervised framework even for a high number of unlabeled samples.

1. Introduction

Research has shown that deep convolutional neural networks (CNNs) perform very well in various computer vision problems such as image classification, object detection, and video analysis. The performance of such models is further enhanced by introducing data augmentation and regularization techniques.

The machine learning model in a supervised framework has shown promising performances. In the real-world, the dataset obtained for classification problems consists of a lot of unlabeled samples. Researchers have introduced a regularization technique to incorporate such data into supervised learning. But can the data augmentation techniques along with semi-supervised learning give an improved performance in a supervised learning environment?

We address the above question by implementing a semi-supervised learning technique- entropy minimization along with data augmentation- mixup to classify images of the CIFAR-10 [7] dataset using CNN. Mixup [2] also works as a regularization technique which has been proven to improve the performance of neural networks by increasing the generalization to adversarial examples and increasing the robustness when faced with corrupt labels. The main idea is to augment the data used for the training set, capture the maximum information out of unlabeled samples and augment the generalization capacity of the model.

2. Related Work

Y. Grandvalet and Y. Bengio [1] in their experiments, has illustrated that semi-supervised learning by Entropy Minimization benefits from unlabeled data. They also introduced their findings in the context of supervised learning and introduced concepts that make the most use of unlabeled samples but also keep a check on their contribution to the robustness of the learning scheme.

H. Zhang et al. [3] introduced a simple data augmentation technique- mixup which mixes two random training samples using linear interpolation which in effect increases the generalization capacity of the network to novel data as well as increases the robustness of the model when faced with corrupt data.

S. Yun et al. [2] proposed a new data augmentation technique- CutMix which acts as a regularization technique in strong classifiers enhancing their localization ability by requiring the model to capture features so that an object can be detected even from a partial view.

3. Method

Since the experiments are performed in a supervised framework, we have implemented a deep convolutional neural network (CNN) for classification. As a baseline, we have considered the accuracy of this CNN model on labeled data using Cross-Entropy as the loss function and Stochastic Gradient Descent as the optimizer.

The semi-supervised learning is performed by splitting the training data further into a few labeled samples and the remaining as unlabeled samples. Experiments are performed by varying the number of labeled and unlabeled samples. The loss function used for this setup is based on the concepts of Entropy Minimization introduced by Y. Grandvalet and Y. Bengio [1]. The loss function $L(A, Y)$ is incorporated in our experiments in the form of Equation 1

$$L(A,Y) = \frac{-1}{m_l} \sum_{a^{(i)} \in A_l} \sum_{k=0}^{K} I\{y^{(i)} = k\} \log a_k{}^{(i)} - \frac{1}{m_u} \sum_{a^{(j)} \in A_u} \sum_{k=0}^{K} a_k{}^{(j)} \log a_k{}^{(j)} \tag{1}$$

where m_l and m_u are the number of labeled and unlabeled samples respectively and m is the total number of samples i.e. $m_l + m_u$, K is the number of classes, $A = [a^{(1)}, a^{(2)}, \ldots, a^{(m)}]$ and $a^{(i)}$ is the softmax activation for a given sample and $y^{(i)}$ is the ground truth , I is the identity function given by

$$I\{\text{condition}\}=1, \text{ if condition} = \text{True}$$
$$I\{\text{condition}\}=0, \text{ if condition} = \text{False}$$

The data augmentation is done by performing mixup on the input samples. The motivation behind using this data augmentation technique is inspired by the work of H. Zhang et al. [3]. The virtual training samples constructed by mixup for two training samples (x_i, y_i) and (x_j, y_j) drawn at random are as shown in Equation 2 and Equation 3:

$$\tilde{x} = \lambda x_i + (1 - \lambda) x_j \tag{2}$$

$$\tilde{y} = \lambda y_i + (1 - \lambda) y_j \tag{3}$$

where x_i, x_j are raw input vectors and y_i, y_j are one-hot label encodings and $\lambda \in [0,1]$.

The above semi-supervised learning and the data augmentation technique is implemented together to analyze the model's performance on the test data set by varying the labeled and unlabeled samples.

4. Experiments

The following are the experiments that were conducted. First, we have implemented a deep convolutional neural network using only the training labeled data and measured its performance on the test data. Second, applied entropy minimization for unlabeled data along with labeled data to train a semi-supervised learning model and report the performance on the test set. Third, apply data augmentation technique mixup on the training samples. Fourth, applied the above semi-supervised learning and data augmentation technique on the training data and report the performance on the test data. Fifth, conduct the experiments by varying the number of labeled and unlabeled samples. A detailed analysis of all the above experiments is as below.

As a baseline, we performed the classification of CIFAR -10 dataset using CNN. The dataset was divided into 50000 training samples and 10000 test samples. Furthermore, all the samples are labeled and thus a supervised learning model was developed for finding the baseline accuracy. The results of this experiment are illustrated in the form of loss function and classification accuracy plots [Figure 1, Figure 2] and Confusion Matrix [Figure 3].

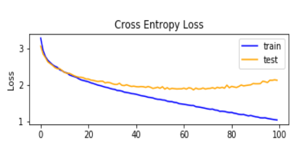

Figure 1: Baseline Cross Entropy Loss

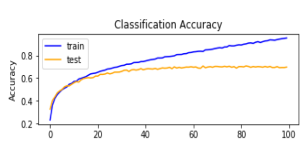

Figure 2: Baseline Classification Accuracy

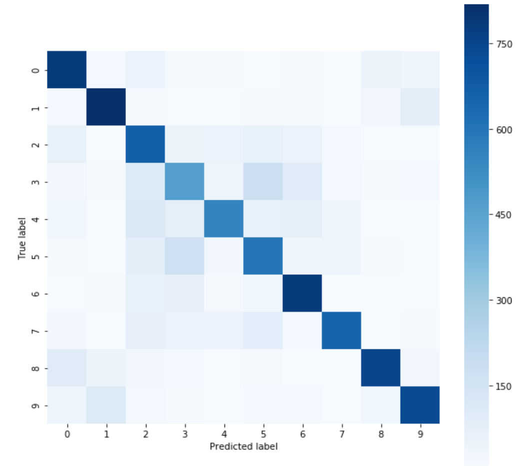

Figure 3: Baseline Confusion Matrix

Between epochs 20-30, it is observed that the model starts overfitting. This is a common problem with deep neural networks which is addressed by introducing regularization techniques.

We performed a semi-supervised learning on the training data by splitting the training data in varying number of labelled and unlabeled samples, the analysis of which can be observed in the figures below. Figure 4 & 5 gives an analysis of this experiment with 25000 labeled and unlabeled data. Figure 6 & 7 gives an analysis of this experiment with 15000 labeled and 35000 unlabeled data. Figure 8 & 9 gives an analysis of this experiment with 5000 labeled and 45000 unlabeled data. The confusion matrix for the above experiments is as shown in Figure [10, 11, 12] The CNN baseline architecture is used as basis for performing this experiment.

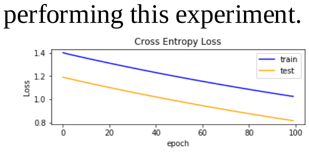

Figure 4: 25000 labeled and 25000 unlabeled samples

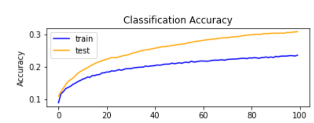

Figure 5: 25000 labeled and 25000 unlabeled samples

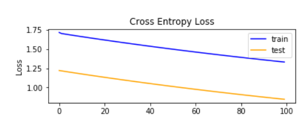

Figure 6: 15000 labeled and 35000 unlabeled samples

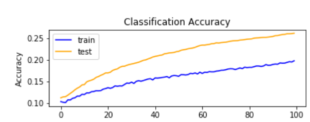

Figure 7: 15000 labeled and 35000 unlabeled samples

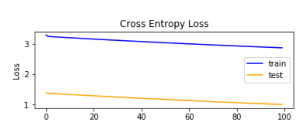

Figure 8: 5000 labeled and 45000 unlabeled samples

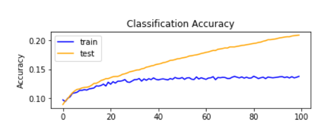

Figure 9: 5000 labeled and 45000 unlabeled samples

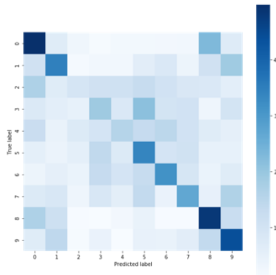

Figure 10: Confusion Matrix for 25000 labeled and 25000 unlabeled samples

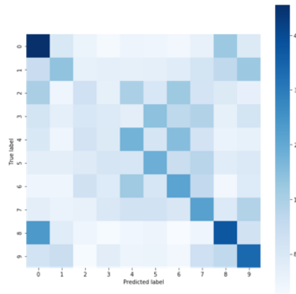

Figure 11: Confusion Matrix for 15000 labeled and 35000 unlabeled samples

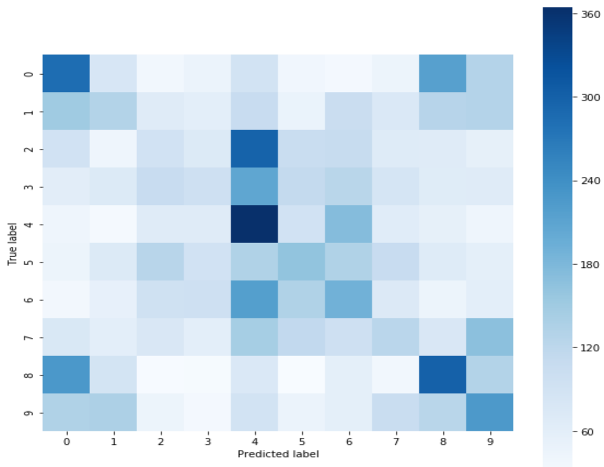

Figure 12: Confusion Matrix for 5000 labeled and 45000 unlabeled samples

In order to overcome the problem of overfitting, regularize the network and improve localization capacity of the model, we introduced mixup in the above architecture. It was seen that this heavily boosted the performance of the model and overcame the problem of overfitting. The test accuracy obtained was higher in each epoch. In a similar manner, the test loss also reduced along with the training error. The results were in the favor of mixup wherein the model performed efficiently over the labeled training dataset, the same which can be observed through the analysis [Figure- 13, 14, 15] and Table 1.

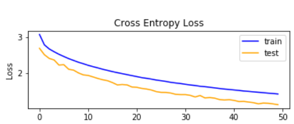

Figure 13: Mixup Cross Entropy

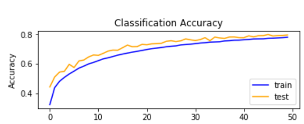

Figure 14: Mixup Classification Accuracy

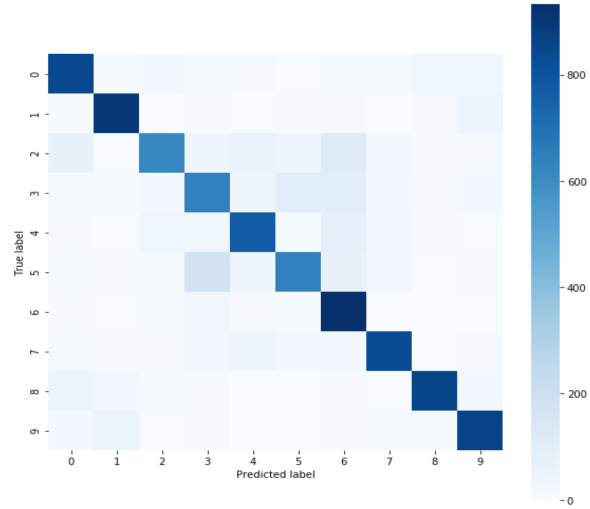

Figure 15: Confusion matrix using mixup

The results obtained from the above models further motivated us to incorporate both techniques on CNN and further analyze the performance of a semi-supervised dataset in this framework. The results obtained were, in fact, favorable to what was expected, and the model showed promising performances even with a huge number of unlabeled samples.

The model performed extraordinarily well as we increased the number of unlabeled samples thereby showing its capacity of generalization and efficiency. Figures 16-24 provides an overview of the same.

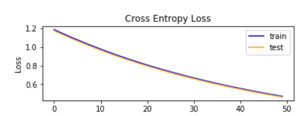

Figure 16: 25000 labeled and 25000 unlabeled samples

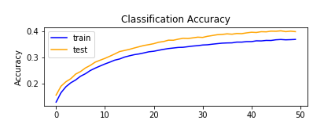

Figure 17: 25000 labeled and 25000 unlabeled samples

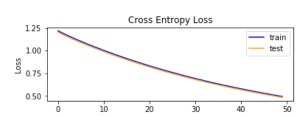

Figure 18: 15000 labeled and 35000 unlabeled samples

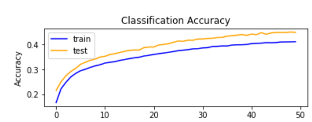

Figure 19: 15000 labeled and 35000 unlabeled samples

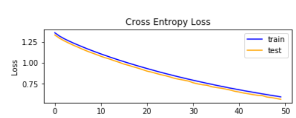

Figure 20: 5000 labeled and 45000 unlabeled samples

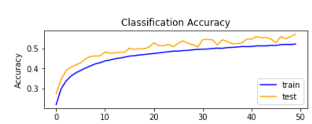

Figure 21: 5000 labeled and 45000 unlabeled samples

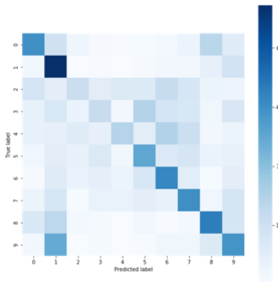

Figure 22: 25000 labeled and 25000 unlabeled samples

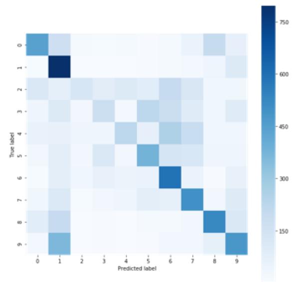

Figure 23: 15000 labeled and 35000 unlabeled samples

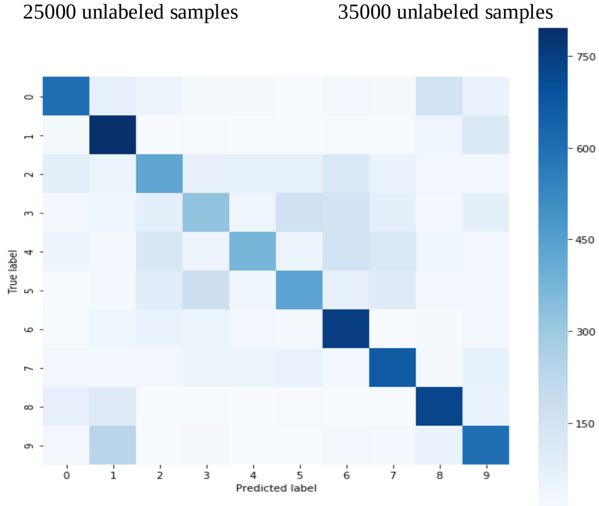

Figure 24: 5000 labeled and 45000 unlabeled samples

A detailed analysis of the experimentation with accuracies and a varying number of labeled and unlabeled samples is as shown in Table 1.

Model	Epoch	Labeled Samples	Unlabeled Samples	Accuracy (%)
Baseline (CNN)	100	NA	NA	68.100
Entropy Minimization (Semi-Supervised)	100	25000	25000	31.290
	100	15000	35000	25.060
	100	5000	45000	20.920
Mixup	50	NA	NA	79.390
CNN+Entropy Minimization+Mixup	50	25000	25000	39.780
	50	15000	35000	44.930
	50	5000	45000	57.290

Table 1: Analysis of different models with varying parameters

For performing all the experiments mentioned above, the base architecture was CNN [4]. The architecture consists of convolutional blocks stacked on top of each other, a flattening layer, a fully-connected layer and finally the softmax activations. The mixup [6] runs on the training data which are then further provided to the model for training and testing.

5. Conclusions

The Baseline model overfits over labeled training data and the accuracy obtained as a result of it is 68.1 %. Incorporating the data augmentation technique mixup helps in improving the overall performance by regularizing the network. Improved performance of 79.39 % was obtained for the same network due to mixup. Introducing semi-supervised learning by entropy minimization in the baseline model achieved lower accuracy over an increase in the number of unlabeled samples. However, a reverse trend i.e. increase in accuracy over an increase in the number of unlabeled samples was observed after the semi-supervised learning was paired along with mixup. This shows the capacity of mixup to generalize the network over unseen data and the power of entropy minimization that makes the most use of unlabeled data. Another regularization technique CutMix [5] can be used along with the above experiments as a future study to augment the localization capacity of the developed model.

6. Division of Work

I have primarily worked on implementing the semi-supervised learning by entropy minimization. This work needed the understanding of cross-entropy in the context of labeled samples and then extending this idea to incorporate unlabeled samples in this framework based on the concepts of entropy minimization. I also summarized the results obtained from this experiment by plotting the accuracy and loss functions of train and test data. I defined my own custom loss function in Keras framework for this purpose. I also worked alongside Aditya and Prashant in integrating the individual model of semi-supervised learning over CNN along with mixup. Furthermore, I worked on this report and contributed to the sections involving the summary, experiments, and results related to entropy minimization as well as the performance of the semi-supervised model along with mixup.

7. Self-Peer Evaluation Table

Jayesh Kudase	Prashant Jadhav	Aditya Chavan
20	20	20

Table 2: Self-Peer evaluation table

References

[1] Y. Grandvalet and Y. Bengio, "Semi-supervised learning by entropy minimization," *Advances in neural information processing systems,* p. 529–536, 2005.

[2] S. Yun, D. Han, S. J. Oh, S. Chun, J. Choe and Y. Yoo, "Cutmix: Regularization strategy to train strong classifiers with localizable features," *arXiv preprint arXiv:1905.04899,* 2019.

[3] H. Zhang, M. Cisse, Y. N. Dauphin and D. Lopez-Paz, "mixup: Beyond empirical risk minimization," *arXiv preprint arXiv:1710.09412,* 2017.

[4] J. Brownlee, "Discover how to develop a deep convolutional neural network model from scratch for the CIFAR-10 object classification dataset.," 2019. [Online]. Available: https://machinelearningmastery.com/how-to-develop-a-cnn-from-scratch-for-cifar-10-photo-classification/.

[5] B. Kim, "CutMix (Image Augmentation) Generator For Keras," 2019. [Online]. Available: https://github.com/DevBruce/CutMixImageDataGenerator_For_Keras.

[6] Y. Uchida, "An implementation of mixup: Beyond Empirical Risk Minimization," 2017. [Online]. Available: https://github.com/yu4u/mixup-generator.

[7] A. Krizhevsky, V. Nair and G. Hinton, "Learning multiple layers of features from tiny images," Citeseer, Tech. Rep, 2009. [Online]. Available: https://www.cs.toronto.edu/~kriz/cifar.html.