Study of the impact of the unlabeled to labeled data ratio on the accuracy of Self-training with Noisy Student algorithm

Tzu Yi Chuang
M.S. in Computer Engineering
Columbia University
tc3075@columbia.edu

Kuan Yu Ko
M.A. in Statistics
Columbia University
kk3376@columbia.edu

Pierre Andurand
M.S. in Computer Science
Columbia University
pa2570@columbia.edu

Abstract

To investigate the effect of the unlabeled to labeled data ratio (the "ratio" or "rate") on accuracy in [1], two types of experiments are run on the *MNIST*[4], *CIFAR10*[5] and *SVHN*[6] datasets. The first tests investigate if a better accuracy could be achieved by running a semi supervised algorithm on the full training dataset by initialising the weights to the ones trained by a supervised algorithm on the full dataset. A supervised learning algorithm is trained on the full training dataset. A Self-Training with Noised Student algorithm (STNS) is then run on the dataset. For different values of the ratio, we start by using the trained supervised learning algorithm on the full dataset to predict the pseudo label on unlabeled data. A noised student model is then trained on the full dataset made up of true and pseudo labels. The un-noised teacher model then uses those weights to predict the new pseudo labels on unlabeled data. The last two steps are repeated. The second series of tests investigate the optimal amount of unlabeled data to be added to a training set in order to improve accuracy on a given supervised algorithm by running a STNS. We start with a fixed small-size training set of labeled data, and then add a quantity of unlabeled data defined by the rate. We test on Resnet20 and a simpler convolutional neural network, and vary the number of Student loops, and level of noise for the Student (Data Augmentation or not). We find that adding data augmentation significantly worsens the results and that adding too many loops bring some instability, and as a result it is better to choose 3 Student loops rather than 10. For algorithms that are not trained for long, the accuracy is a downward sloping function of rate, meaning that is it better to add an extra 20% of unlabeled data rather than 5 times more. However, for algorithms that have been well trained and optimised, there is no clear pattern and therefore no clear optimal values of rate. Our code and full tests results are available at `github.com/PierreAndurand2/DL_final`.

1 Introduction

Deep learning algorithms found great success in image recognition [2]. However, nearly all the state-of-the-art models are trained with supervised learning, which means that these models can't utilize the large amount of unlabeled data. By showing the models only labeled images, we limit ourselves from making use of unlabeled images available in much larger quantities to improve accuracy and robustness of state-of-the-art models. In the previous work of [3], the model trained on 3B weakly labeled images from social media, improved the accuracy of image classification and object detection task. In 2019, the Self-Training with Noisy Student (STNS) model presented by [1] achieved a higher accuracy on the ImageNet dataset than other state-of-the-art algorithms by making use of 300M unlabeled images. This method can be thought of as Knowledge Expansion rather than

Knowledge Distillation [2], where the student model learns in more difficult environments created by adding noise on images and model. The objective of this report is to test the results of the STNS model on three simple datasets relative to the equivalent fully supervised algorithm, and to find the optimal range of ratio to use between unlabeled and labeled data in order to maximize accuracy.

2 Related Work

Self-training Noisy Student algorithm The self-training process consists of the following steps: Firstly, a Teacher model is trained using labeled data (true labels), and then used to predict unlabeled data which become pseudo labeled data (pseudo labels). The concatenation of true and pseudo labeled data is then used to train a noised Student model and the procedure is repeated a number of times. This method, introduced by [1], is claimed to reach higher accuracy on ImageNet than any other state-of-the-art algorithm by using the information contained in unlabeled data, and by making the algorithm learn better thanks to the added noise.

Semi-supervised Learning Semi-supervised learning have delivered promising results in many areas by harnessing the information of unlabeled data. A good book reference is added in [7].

3 Methods

Datasets The datasets selected are *MNIST*, *CIFAR-10* and *SVHN*. The *MNIST* dataset consists of 60,000 training images and 10,000 testing images of low-resolution (28x28) representing handwritten digits from 0 to 9 with grayscale level. *CIFAR-10* is composed of 50,000 training and 10,000 testing color images of low-resolution (32x32). The *CIFAR-10* classes are: airplanes, cars, birds, cats, deer, dogs, frogs, horses, ships, and trucks. Therefore it has a higher complexity than *MNIST*. *SVHN* is a real-world image dataset of street house numbers. The dataset is composed of 73,257 training images, and 26,032 testing images. They are low-resolution (32x32) colour images. It can be seen as similar in flavor to *MNIST* (e.g., the images are of small cropped digits), but comes from more complex natural scene images). SVHN is obtained from house numbers in Google Street View images [8]. All datasets have 10 classes.

Algorithm The STNS algorithm follows the subsequent steps:
Inputs: Labeled images $\{(x_1, y_1), (x_2, y_2), ..., (x_n, y_n)\}$ and unlabeled images$\{\hat{x_1}, \hat{x_2}, ..., \hat{x_n}\}$
(a) Learn a teacher model which minimizes the CE-loss on labeled images (true labels).
(b) Use the teacher model to predict pseudo labels for unlabeled images (pseudo labels).
(c) Learn a noised student model which minimizes the CE-loss on the concatenation of true and pseudo labels
(d) Iterative training: Use the student as a teacher and go back to step (b).

Tests and evaluation criteria Cross-entropy loss is selected for the training process, but the evaluation criteria is top-1 accuracy as a function of the ratio. We also check the sensitivity of the accuracy results to the following parameters: number of Teacher-Student loops in the STNS, learning rate of Student model (0.001 vs 0.0001), and if the Student model uses images coming from data augmentation or not.
Two types of tests are selected. In the **first test**, the full training set is split between labeled and unlabeled data as a function of the ratio. We then investigate if the performance of STSN is better than the fully supervised model, and what the optimal range of the ratio is. This method could be useful when one cannot access new unlabeled data and wants to get the best accuracy possible on testing data given chosen training dataset and algorithm. The teacher model trained on the full dataset predicts pseudo labels (step (a) modified, and step (b)), and then 3, or 6 loops of 10 epochs of steps (c) and (d) are computed. The **second test** takes a fixed small-sized balanced sample of the training set as labeled data (5000 images), and then adds part of the balance of the training set as unlabeled data as a function of the ratio. We then run STSN on that training set and compare accuracies in order to find the optimal range of the ratio, and determine how much unlabeled data could be useful to add if one only had a small labeled training set. We used *MNIST* with a Simple CNN in the Milestone report, and in this final report we investigate ResNet20 and the Simple CNN on *CIFAR-10* and *SVHN*.

4 Architecture

Simple CNN One of the Teachers used is a simple Convolutional Neural Network composed of the following layers:
2 Conv2D with 32 (3,3) filters followed by ReLu activation function
MaxPooling2D (2,2)
Dropout(0.25)
2 Conv2D with 64 (3,3) filters followed by ReLu activation function
MaxPooling2D (2,2)
Dropout(0.25)
Flatten
Dense(512) followed by ReLu
Dense(10) followed by softmax activation function.

Resnet20 ResNet20 is used for *CIFAR-10* and *SVHN*. We used the code for Resnet20 that can be applied to CIFAR10 and SVHN from Keras (https://keras.io/examples/cifar10_resnet/). The original Resnet paper can be found in [9], and its modification to fit our datasets can be found in [10].

The student model has the Dropout(0.5) added before the last layer in order to add some noise. We also investigate Data Augmentation (DA) and No Data Augmentation (NDA) for data fed to the Student. The Simple CNN is used on the 3 datasets.

5 Experiment Results

In [1], the authors used 1.3M images from Imagenet 2012 ILSVRC, and 130M filtered unlabeled images from the JFT dataset. In their second ablation study, they found that reducing the unlabeled dataset by a factor of 16 had no impact on accuracy. This corresponds to a ratio of unlabeled/labeled of 6.25. The authors note that performance drops when the ratio is further decreased, concluding that "a large amount of unlabeled data leads to better performance". As we are using much simpler datasets with less classes, we will mainly be interested in ratios lower than 6. And we will study if increasing the ratio also betters performance or if it has the opposite effect on those datasets.

Implementation Details We use the Adams optimizer with cross-entropy loss for the training process, and output both the cross-entropy loss as well as overall accuracy while testing. We then compare the impact of different rates on the accuracy, and check the improvement of the resulting accuracy relative to the fully supervised model. We use soft pseudo labels (softmax). A learning rate of 0.001 is used for the training of the Teacher, and 2 learning rates are investigated for the Student: 0.001 and 0.0001. For Test1, we investigate Data Augmentation and 3 Student loops on *SVHN* and *CIFAR10*. We also investigate No Data Augmentation with 6 Student loops. For Test2, we investigate 3 loops vs 10 loops for *SVHN* and *CIFAR10* for both Resnet20 and the Simple CNN algorithms.

For the fully supervised model, we choose an amount of epochs after which the accuracy stabilizes, ie 36 for *MNIST* and 100 for *CIFAR10* and *SVHN*.

We use Jupyter notebook with Python and Keras. The ratios used are [0, 0.2, 0.5, 0.75, 1, 2.5, 5, 7.5, 9]. We did the tests on 8 GPUs on AWS in parallel.

Results The conclusion reached by the Milestone report was that by looking at the accuracy as a function of rate for *MNIST* and *CIFAR10* using the Simple CNN with 10 Student loops, the optimal range of the ratio of unlabeled to labeled data was [0.1-0.3].
Unfortunately this time, by using a more complex algorithm such as Resnet, the conclusion is a lot less clear. The accuracy is not always improved, and the slope of the accuracy as a function of rate does not have a clear pattern as it is a function of many other parameters: the algorithm used, the

dataset used, and if we feed Data Augmentation to the Student. Increasing the number of Student loops above three appears to add noise to the pattern. A learning rate for the Student of 0.0001 looks to be better than 0.001.
We note that by using very few epochs (5-10) for the training of the fully supervised algorithm, and only one Student loop, there is an increase in accuracy for both Test1 and Test2, and all charts are downward sloping. This probably means that when there is a lot of easy improvement to be made, using STNS is beneficial, and the optimal ratio is [0.1-0.5]. We selected 4 figures (1-4) here to illustrate this point, and all other tests are in the "Code_final_project" notebook. This notebook shows our code and results of all those simple tests. For longer tests, they were made on different GPUS, and we collected all the results in the "results" notebook. NDA stands for No Data Augmentation. DA stands for Data Augmentation.

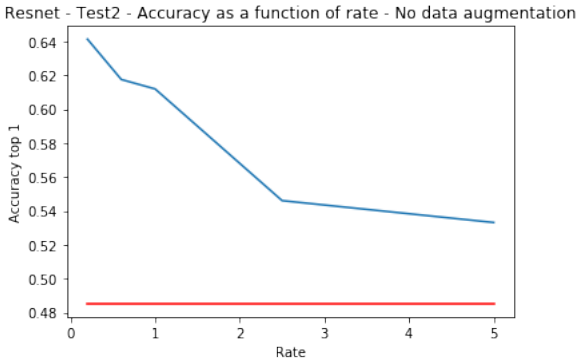

Figure 1: Resnet-CIFAR10-test2-NDA-10 epochs training-1 loop

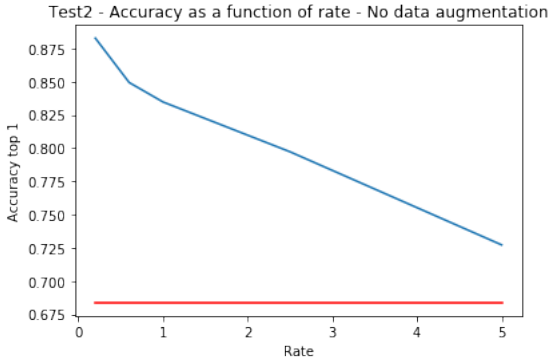

Figure 2: Resnet-HN-test2-NDA-10 epochs training-1 loop

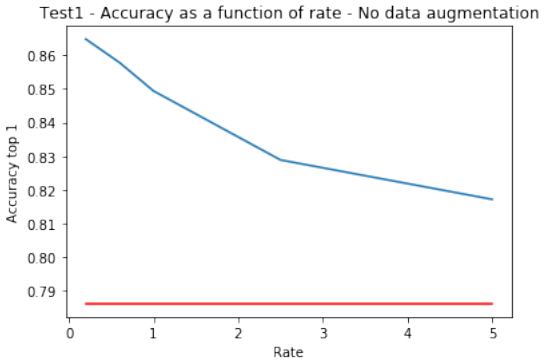

Figure 3: Resnet-CIFAR10-test1-NDA-10 epochs-1 loop

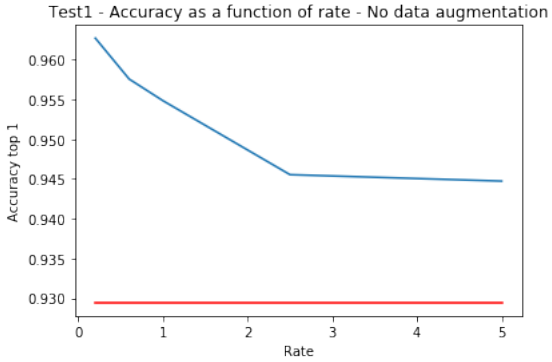

Figure 4: Resnet-HN-test1-NDA-10 epochs training-1 loop

The figures below use lr=0.0001 for the Student, and 100 epochs training for the first Teacher on fully labeled data. We only selected No Data Augmentation, as adding Data Augmentation worsens the results. All tests results can be found in the "results" notebook.

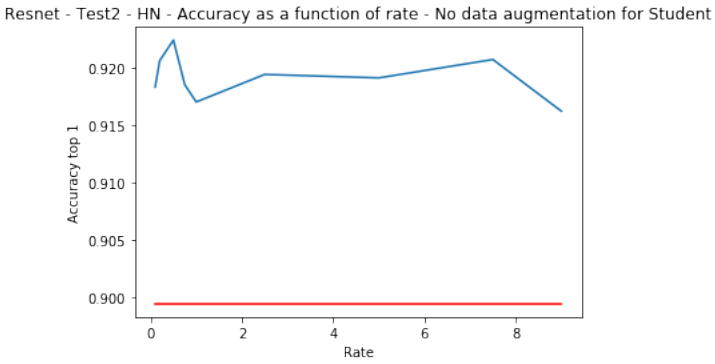

Figure 5: Resnet-HN-test2-NDA-3 loops

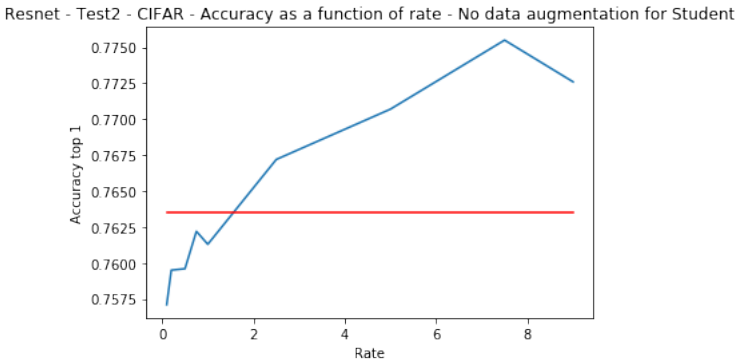

Figure 6: Resnet-CIFAR10-test2-NDA-3 loops

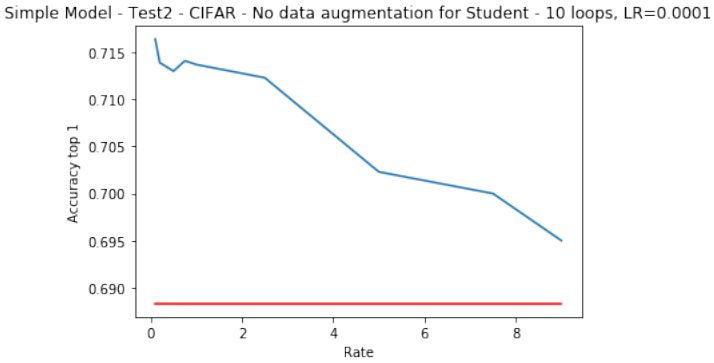

Figure 7: Simple CNN-CIFAR10-test2-NDA-10 loops

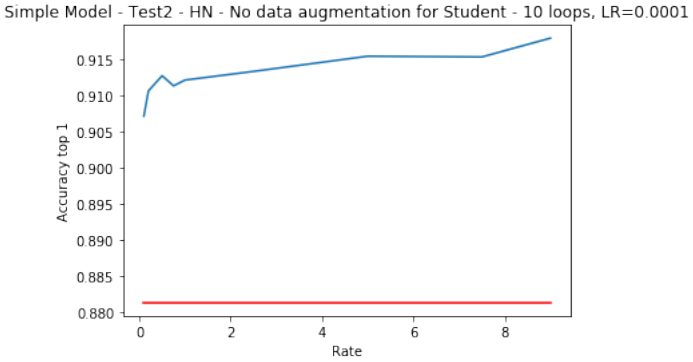

Figure 8: Simple CNN-HN-test2-NDA-10 loops

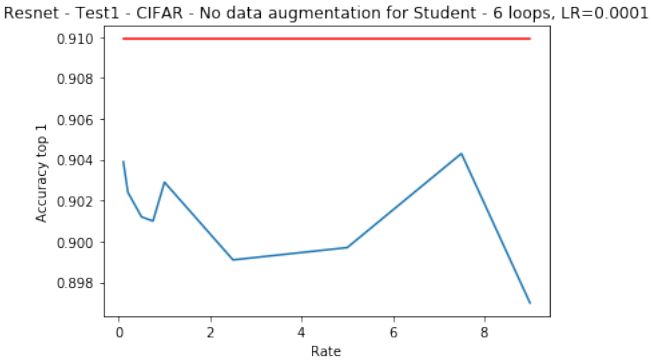

Figure 9: Resnet-CIFAR-test1-NDA-6 loops

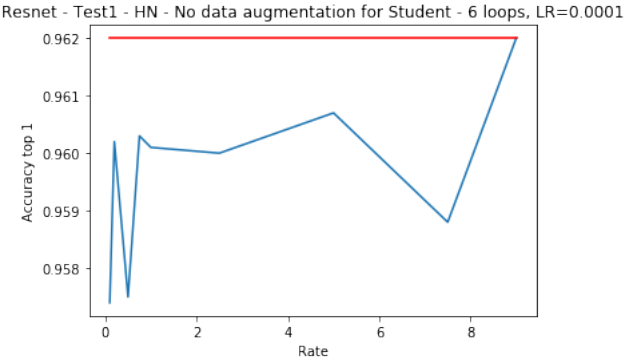

Figure 10: Resnet-HN-test1-NDA-6 loops

6 Conclusion

In this study, we experimented on the impact of the ratio between unlabeled data and labeled data on accuracy in STNS on *CIFAR10* and *SVHN* using the ResNet20 algorithm mainly. We can conclude that on the datasets and algorithms used it is better not to use Data Augmentation on the Student, as doing so can give us worse results than the supervised algorithm. A learning rate of 0.0001 for the student looks better than 0.001. Using more than 3 loops for the Student does not seem to add much, and if anything seems to be adding noise. On Test1, we do not see any improvement from Resnet using STNS on both datasets (Fig 9 and 10), and it does not seem to be dependent on the ratio. We cannot find a clear pattern for the accuracy as a function of the ratio (Fig 5-8). It is a function of the algorithm, datasets and hyper parameters used.

References

[1] Qizhe Xie and Minh-Thang Luong and Eduard Hovy and Quoc V. Le (2019) Self-training with Noisy Student improves ImageNet classification. *arXiv* 1911.04252

[2] Geoffrey Hinton, Oriol Vinyals, and Jeff Dean (2013) Distilling the knowledge in a neural network. *arXiv preprint axXiv* 1503.02531

[3] Dhruv Mahajan, Ross Girshick, Vignesh Ramanathan, Kaiming He, Manohar Paluri, Yixuan Li, Ashwin Bharambe, and Laurens van der Maaten (2018) Exploring the limits of weakly supervised pretraining. *arXiv* 1805.00932

[4] LeCun Yann, Cortes Corinna (2010), MNIST handwritten digit database.

[5] Alex Krizhevsky (2009), Learning Multiple Layers of Features from Tiny Images.

[6] Yuval Netzer, Tao Wang, Adam Coates, Alessandro Bissacco, Bo Wu, Andrew Y. Ng (2011) Reading Digits in Natural Images with Unsupervised Feature Learning. *NIPS Workshop on Deep Learning and Unsupervised Feature Learning*

[7] van Engelen, J.E., Hoos, H.H. A survey on semi-supervised learning. Mach Learn 109, 373–440 (2020). https://doi.org/10.1007/s10994-019-05855-6

[8] http://ufldl.stanford.edu/housenumbers/

[9] K. He, X. Zhang, S. Ren and J. Sun, "Deep Residual Learning for Image Recognition," in CVPR, 2016

[10] https://towardsdatascience.com/resnets-for-cifar-10-e63e900524e0