ML Project

Amir Loewenthal 205629124

Ron Keller 312501703

Introduction

In this project, we evaluated the performance of a deep learning ensemble method called “mean teacher” from the paper of Tarvainen, Antti, and Harri Valpola. "Mean teachers are better role models: Weight-averaged consistency targets improve semi-supervised deep learning results." arXiv preprint arXiv:1703.01780 (2017). We implemented the paper’s algorithm and compare the results of the algorithm to a known image recognition algorithm, and a novel improvement we proposed to this algorithm. We evaluate the three algorithms on 20 different datasets to give varied and confident results. In addition, we managed statistical tests to conclude which algorithm performed the best on different datasets.

The algorithm

Algorithm description

The algorithm proposes a regularization method that aims to improve the performance of supervised and semi-supervised learning problems. The algorithm trains two different sets of weights. One set is called the student model and the second set is called the teacher model. Both models are trained with noise on the input and then the MSE between the teacher and the student predictions is called “consistency cost” and it is considered as part of the loss function (In unsupervised learning this is the loss function alone).

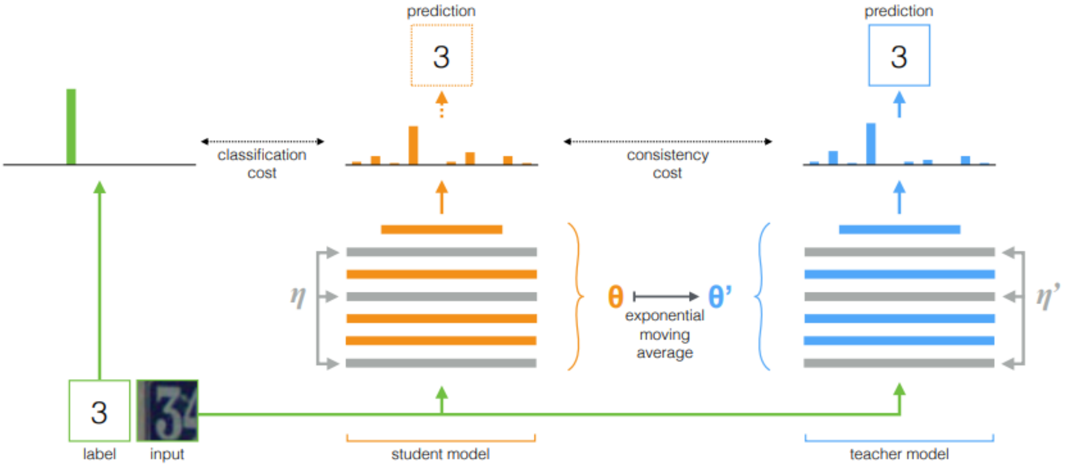


figure explanation is taken from the paper:

The Mean Teacher method. The figure above is taken from the paper and it illustrates the algorithm architecture: it depicts a training batch with a single labeled example. Both the student and the teacher model evaluate the input applying noise (η, η0) within their computation. The softmax output of the student model is compared with the one-hot label using classification cost and with the teacher output using consistency cost. After the weights of the student model have been updated with gradient descent, the teacher model weights are updated as an exponential moving average of the student weights. Both model outputs can be used for prediction, but at the end of the training, the

teacher’s prediction is more likely to be correct. A training step with an unlabeled example would be similar, except no classification cost would be applied
The novelty of this algorithm is in the way that the teacher model updates its weights: The algorithm uses an exponential moving average (EMA) of the student model weights. The EMA method is been taken from temporal ensembling with one difference. Instead of averaging over the predictions the mean teacher algorithm average over the weights.

algorithm inner CNN architecture taken from the paper:

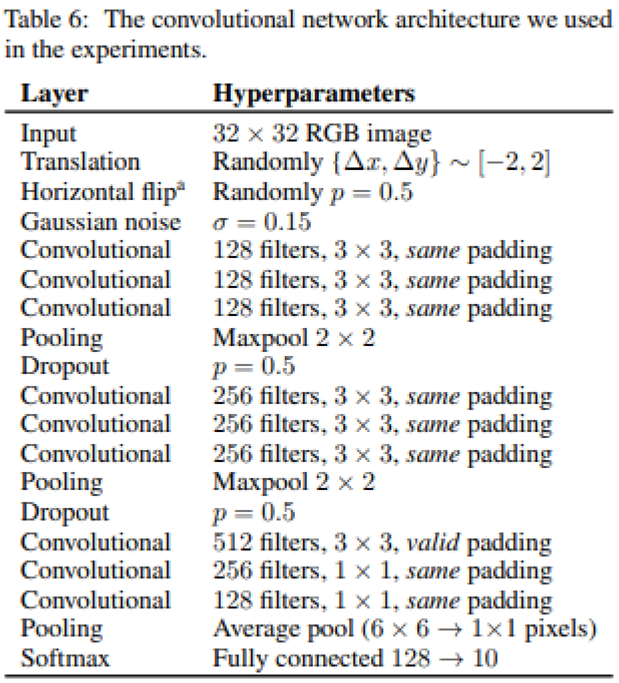


Table 6: The convolutional network architecture we used in the experiments.

Layer	Hyperparameters
Input	32×32 RGB image
Translation	Randomly $\{\Delta x, \Delta y\} \sim [-2, 2]$
Horizontal flip[a]	Randomly $p = 0.5$
Gaussian noise	$\sigma = 0.15$
Convolutional	128 filters, 3×3, *same* padding
Convolutional	128 filters, 3×3, *same* padding
Convolutional	128 filters, 3×3, *same* padding
Pooling	Maxpool 2×2
Dropout	$p = 0.5$
Convolutional	256 filters, 3×3, *same* padding
Convolutional	256 filters, 3×3, *same* padding
Convolutional	256 filters, 3×3, *same* padding
Pooling	Maxpool 2×2
Dropout	$p = 0.5$
Convolutional	512 filters, 3×3, *valid* padding
Convolutional	256 filters, 1×1, *same* padding
Convolutional	128 filters, 1×1, *same* padding
Pooling	Average pool ($6 \times 6 \rightarrow 1 \times 1$ pixels)
Softmax	Fully connected $128 \rightarrow 10$

Advantages
1.The algorithm is good on large datasets and in online learning in contrast to temporal ensembling.
2.The algorithm can be used in both supervised and semi-supervised environments.

Instead of sharing the weights between the student and the teacher, we are updating the teacher model using EMA weights of the student, so now it can aggregate information after every step instead of every epoch, In addition, since the weight averages improve all layer outputs, not just the top output, which gives us several advantages:

4.More accurate model weights.
5.The target model has better intermediate representations
6.More accurate target labels lead to a faster feedback loop between the student and the teacher models, resulting in better test accuracy
7.The approach scales to large datasets and online learning

Disadvantage
Since the teacher model improves its performance based on the improvement of the student there is high coupling between the two models.

Our improvement

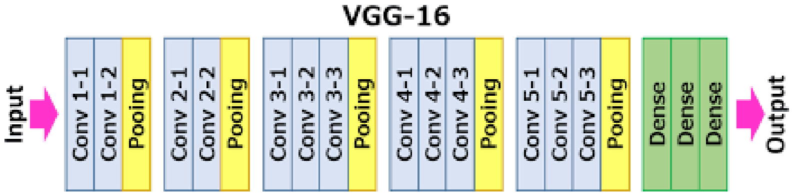
Description

In the previous section, we present the disadvantages of the mean teacher algorithm. The coupling between the two models was one of them. To overcome this problem, we proposed our improved algorithm “Double Mean Teacher”.
Instead of using a single teacher, our method will train two teachers. Each teacher will train differently. The first teacher will use the EMA of the student weights in a similar way to the teacher in the mean teacher algorithm. While the second teacher will update its weights with gradient descent.
The total cost of the model will be built from the classification cost of the student, and the average between the consistency costs of the two teachers.

We thought that by using multiple teachers while some of them are updated separately with gradient descent instead of EMA weights of the student and averaging the total cost. Our model will be able to overcome the high coupling between the student and the teacher. The average between all those teachers will help our student model to keep important information from all teachers and improve the generality of the model even further than when using a single teacher.

Baseline

As a baseline, we choose a known image recognition architecture. We choose VGG16, an image recognition algorithm that we worked with on a previous assignment in this course. An illustration of the VGG 16 network is presented here:



In addition to the well-known baseline, in the paper, they address both supervised and semi-supervised problems, and the baseline VGG can't handle semi-supervised problems so we also ran the PHI model presented in the paper which is just equal to the mean teacher model but the weights of teacher are same as the student and no EMA update is used, in the paper, they used also two baselines (GAN and PHI model) and compared results with GAN on the supervised and with the PHI model as a baseline in the semi-supervised.

Evaluation

The Datasets:

We choose 20 datasets to run our experiments on them. As Lior recommended, we split the larger datasets into several smaller datasets, so they can be processed more easily. Note that in the table when there is X_1, X_2 for dataset name X. this means that those are splits of the original dataset to several small datasets.
note: In addition to the 20 datasets presented below, we also ran cifar_100_i, i∊{1-5} also with semi-supervised configuration so we can see if the mean-teacher,double-mean teacher are better than the baseline for our semi-supervised which is the PHI-model.

note2: Some of the datasets can be found under the ‘Datasets’ folder inside the git repository. Other datasets can be downloaded via the data_load command as explained in the Github’s readme file.

	Dataset name	Shape	Num of samples	Classes
0	Beans	(32, 32, 3)	1295	3
1	Casava	(64, 64, 3)	9430	5
2	Cifar100_1	(32, 32, 3)	12000	20
3	Cifar100_2	(32, 32, 3)	12000	20
4	Cifar100_3	(32, 32, 3)	12000	20
5	Cifar100_4	(32, 32, 3)	12000	20
6	Cifar100_5	(32, 32, 3)	12000	20
7	Cmater	(32, 32, 3)	6000	10
8	Ctb_1	(64, 64, 3)	2051	70
9	Ctb_2	(64, 64, 3)	2134	70
10	Ctb_3	(64, 64, 3)	1848	60
11	Oxford_1	(64, 64, 3)	3493	51
12	Oxford_2	(64, 64, 3)	4696	51
13	Rps	(100, 100, 3)	2892	3
14	Coloret	(64, 64, 3)	5000	8
15	shvn_1	(32,32,3)	51765	4
16	shvn_2	(32,32,3)	34565	4
17	stl_1	(96,96,3)	5200	4
18	stl_2	(96,96,3)	5200	4
19	stl_3	(96,96,3)	5200	4

Parameters

For each of the different algorithms, we chose different parameters for the hyperparameters tuning as stated in the assignment we did 10-fold cross-validation with 50 random searches while each search is checked on 3-fold cross-validation to find the best hyperparameters:

hyperparameters for our baseline model:
batch_size: [32,64,128]
learning rate: [0.1, 0.01, 0.001, 0.05]
type of pooling layers: [max, avg]

We choose these specific parameters for parameter tuning since these parameters are considered to be the baseline for parameter tuning in regard to deep learning algorithms.

for the hyperparameters of the PHI model, they are like the mean-teacher just without the EMA rate.

hyperparameters for our mean-teacher/double-mean-teacher:
dropout rate of the student model: [0.1, 0.2, 0.3, 0.4]
batch_size: [32,64,128]
dropout rate of the teacher model: [0.1, 0.2, 0.3, 0.4]
EMA rate: [0.999, 0.95, 0.92, 0.98]

The reasons behind choosing these hyper params:
1. EMA rate: the EMA rate is affecting the weights update of the teacher similar to the learning rate in the baseline model. so it makes sense to try different values for the EMA rate.

2.dropout rate: the dropout purpose is to add noise to the input of the teacher/student so they will have slightly different images to learn from which results in more robust learning.

3. batch size: it's good practice to try the model learning on different batch sizes to find a good balance between the computation time and the accuracy of the gradient update.

Results

All the results are presented in the file “Results/Experiments_results_supervised.xlsx” inside the git repository.
there are also the results on the semi-supervised run-in ”Results/Experiments_results_semisupervised.xlsx” in the git repository.

Statistical tests

We ran a Friedman test on the results of the experiments over the 20 datasets and the three algorithms.

We choose AUC as the metric for the test. We run the experiments for both the supervised and semi-supervised environments.

(code for the statistical tests can be found in the git repository)

Supervised

The Friedman test returns a p-value of 0.035 therefore for alpha=0.05 we can reject the null hypothesis.

Since we reject the null hypothesis we ran posthoc tests.

We chose the Nemenyi Post-Hoc test to compare the performance of the algorithms two algorithms at a time. The results of the Nemenyi test are provided in the following table

	Baseline	MT-Model	Double MT
Baseline	1	0.190777	0.0367
MT-Model	0.190777	1	0.691266
Double MT	0.0367	0.691266	1

We can see from the results that for alpha=0.05 our improved algorithm- the Double Mean Teacher outperforms the Mean Teacher algorithm significantly.

Semi-Supervised

The Friedman test returns a p-value of 0.006 therefore for alpha=0.01 we can reject the null hypothesis.

Since we reject the null hypothesis we ran posthoc tests.

We chose the Nemenyi Post-Hoc test to compare the performance of the algorithms two algorithms at a time. The results of the Nemenyi test are provided in the following table

	PHI-model	MT-Model	Double MT
PHI-model	1	0.254114	0.004467
MT-Model	0.254114	1	0.254114
Double MT	0.04467	0.254114	1

We can see from the results that for alpha=0.01 our improved algorithm- the Double Mean Teacher outperforms the PHI algorithm significantly.

Conclusions

As we mentioned above we run our algorithms over 20 different datasets. From observing the results of our experiments we came to the following conclusions:

-Our Double Mean Teacher Algorithm improves the performance on different datasets. The accuracy of our improved model was better on many of the datasets. In addition, as presented in the statistical tests this improvement is statistically significant.

-The performance on problems with a different number of classes: Since our experiments were on a large variety of datasets we observe through our experiments that datasets with a small number of labels (svhn, rps) tend to have higher accuracy than datasets with a large number of labels (cbt,ciphar100). Those differences appeared through all the algorithms but were highly noticeable on the Mean Teacher algorithm. This leads us to conclude that the Mean Teacher algorithm performs better on datasets with a small number of labels.

-When evaluating the models on semi-supervised data we couldn't use the VGG model so as we described above we used the PHI model for baseline.
Here it was clear that the mean-teacher and the double-mean-teacher outperformed the phi model, so we can conclude that in the semi-supervised area those improvements were really effective and boosted the results.