Mean Teacher SSL Image Segmentation: Does Increasing the Number of Labelled Images Improve Performance

Student ID: 22197823

1 Introduction

Supervised neural network models are able to produce incredible results, and for some tasks can perform better than humans [1]. But training these models requires a large corpus of labelled data. In the real world this is a problem, labelling data is very expensive. A prime example is medical imaging which requires expensive machinery to produce the image, and then requires a well trained individual to label the data, and also comes with the complication that there maybe privacy concerns using this labelled data. This paper focuses on semantic image segmentation, the task of labelling and classifying every pixel in an image. Again, this type of problem is very expensive to generate a labelled dataset because now every pixel must be labelled, not just the image.

Semi Supervised Learning (SSL) [2] is an approach to get around this by learning from both the available labelled and unlabelled data. The goal is to extract useful information from the unlabelled data to improve the model, saving resource on labelling the dataset. There are many SSL methods: Pseudo Labelling [3] uses the prediction from your model as the label for unlabelled data. Mix-Match [4] extends Pseudo Labelling by applying many different augmentations to the input, and then combining all the predictions to generate a label. Mean-Teacher [5] uses a second *Teacher* network (which is the exponential moving average of the primary *Student* network) to generate labels for unlabelled data.

In this paper we will be using a Mean-Teacher [5] SSL method to train a UNet model on semantic image segmentation on the OxfordIIIT-Pet Database [6]. We attempt to answer: Does the mean-teacher SSL algorithm work for semantic image segmentation; and how does the model's performance vary with differing amounts of labelled data.

2 Methods

2.1 Dataset Creation

Our models are trained on the OxfordIIIT-Pet dataset which has over 7000 images of cats and dog, each image is labelled with a trimap classifying each pixel as (1) animal, (2) background or (3) boundary between animal and background. Our model will be classifying each pixel as either animal or background. Therefore we interpret class (3) to be equal to animal (1), giving us a binary label mask. This can been seen in *Fig 1.*

We partitioned 20% of the images for the validation set, 10% for the test set, and the remaining 70% for the training set. It's worth noting when the dataloader is created an argument is required to indicate what percentage of the training images will be labelled. Unlabelled data is artificially created by discarding the label masks. Labelled data is chosen sequentially from a static list of all the training images, therefore labelled images aren't added randomly each time and we can compare different datasets of different labelled percentages.

2.2 Dataloader and Augmentation

Any images loaded from the dataloader are first normalised so that the whole dataset has a mean and standard deviation of $(0, 1)$ respectively to help with convergence when training. Images are then resized to 64x64 using bi-linear interpolation enabling training to be completed in batches.

The dataloader samples uniformly without replacement from it's respective dataset. For example a dataset with 20% labelled and 80% unlabelled, each mini-batch has an expectation of containing 20% labelled and 80% unlabelled images.

The images in each mini-batch are augmented with a random transform that can alter: hue, saturation, Gaussian blur and add Gaussian noise. Examples can be seen in *Fig 1.* Applying augmentation increases the generalizability of our models [7].

2.3 UNet Model

We are using the UNet [8] architecture for our network with a depth of 3. Our network is shallower than most UNet models due to limitations on our computation resource.

Our model takes input batches of dimension [B,3,64,64], where B is the batch size. The UNet can take an arbitrarily sized images with 3-channels, but to enable training with batches we resize all images to 64x64 pixels.

The model has two parts, the encoder and the decoder. The encoder is constructed from 3x3 double convolutional layers with padding, followed by max-pool down-sampling. There are 3 of these layers, each layer reducing the spatial dimension and increasing the channel dimensions to finally give dimensions of [B,256,16,16]. The decoder is constructed from up-sampling 2x2 up-convolutions followed by 3x3 double convolutional layers. In addition there are 2 skips from the encoder to the decoder. The decoder increases the spatial dimensions and decreases the channel dimension. Finally a 1x1 kernel is applied to the output from the final double-conv layer to output dimensions [B,2,64,64].

UNet performs well on image segmentation [8] for a number of reasons. Firstly the architecture naturally outputs an image so it can classify each pixel. Secondly being a convolutional network with skips it is able to extract features in the image, eg. shapes, with a high receptive field.

2.4 Mean-Teacher Training

Mean-Teacher Overview A diagram of our mean-teacher method can be seen in *Fig 2*. The idea of this is to train our primary model, the *Student*, only on labelled data initially so it has some performance in prediction. Then we use a second model, the *Teacher*, which is the exponential moving average of the student model. The student continues to learn from labelled data using the supervised loss, and now introduce an unsupervised loss for all images between the teacher's prediction of a non-augmented image vs the student's prediction from an augmented image. Due to no augmentation applied to the teacher model the idea is the teacher performs better than the student during training.

Dice Loss For both the classification cost (supervised loss) and the consistency cost (unsupervised loss) we use a modified version of the Dice Loss proposed in [9]. Our dataset is not well balanced between background and animal, so we use the modified dice loss to put more cost on classifying the animal incorrectly compared to the background. More formally the dice loss we use is:

$$D(P,T) = 1 - \frac{2\sum_i^N p_i t_i}{\sum_i^N p_i^2 + \sum_i^N t_i^2} \tag{1}$$

where $P \in [0,1]^{64\times64}$ is the softmax of the prediction of the animal, and $T \in \{0,1\}^{64\times64}$ is the hard target label of the animal, 1 representing animal. N is the total number of pixels (in our case 64^2). This is a differentiable function allowing us to minimise D through stochastic gradient descent.

Classification and Consistency Cost Classification cost is the supervised loss (L_s) during training, it is the dice loss between the student softmax prediction and the ground truth label. L_s is only calculated for labelled data.

Consistency Cost is the unsupervised loss (L_u), it is the weighted Dice Loss between the student softmax prediction and teacher hard prediction target: $L_u = w_t D(P,T)$, where w_t is the consistency weighting during epoch t ramping up from 0 to 1.5 exponentially between epochs 8 and 33 then remaining at 1.5

However, we are training two types of models, one is fully supervised with only labels, in this case we only calculate the classification cost, giving the total supervised loss $L = L_s$. When we are training semi-supervised models we have a total semi-supervised loss $L = L_s + L_u$.

Training Training is completed for a total of 100 epochs with a batch size of 32 using the Adam optimize with learning rate 0.001 and momentum of 0.9. Each training iteration uses the loss L defined in the previous section.

The student and teacher model are initialised with the same random Gaussian weights. Each epoch the Teacher parameters are updated with the exponential moving average of the student's weights: $\hat{\theta}_t = \alpha\hat{\theta}_{t-1} + (1-\alpha)\theta_t$, where $\hat{\theta}_t$, and θ_t are the weights of the teacher and student model respectively after epoch t.

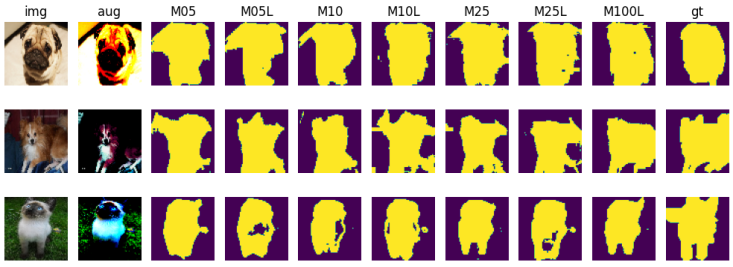

Fig. 1. (img) resized original image, (aug) augmented image, (MXX) predictions from each trained model, (gt) ground truth

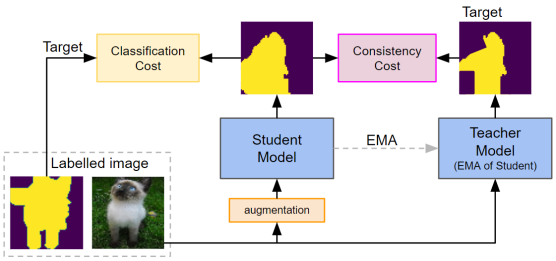

Fig. 2. Mean Teacher learning method

3 Experiments

Different Datasets To complete our experiments we create 7 different training datasets of varying sizes, and label to unlabelled ratio detailed in *Table 1.* For example D10L consists of only labelled images but consists of 10% of the total training data, whereas dataset D10 consists of the same 10% labelled data but with all the remaining training data unlabelled.

For each of these datasets we train a model for 100 epoch. If the dataset only comprises of labelled data we use the supervised training method detailed in *Section 2.4*, and if the dataset comprises of both labelled and unlabelled the SSL Mean Teacher training method is used detailed in the same section.

Dataset Name	D5L	D5	D10L	D10	D25L	D25	D100L
Model Name	M5L	M5	M10L	M10	M25L	M25	M100L
Labelled %	5%	5%	10%	10%	25%	25%	100%
Unlabelled %	0%	95%	0%	90%	0%	75%	0%

Table 1. Training datasets split of labelled and unlabelled data, and what models are trained on what dataset. Percentages are the percentage of the total available training data (70% of all data, remaining 30% is for validation and test sets)

Evaluation Metrics To evaluate our models during training we calculate Pixel *Accuracy* and Intersection over Union (IOU) every 5th epoch on the validation set. Later in the report we calculate the evaluation metrics on the test set.

Accuracy is the number of correctly predicted pixels over the total number of pixels: $A(P,T) = \sum(P == T)/N$, where $P,T \in \{0,1\}^{64\times64}$ are the hard predictions of animal, and the ground truth of animal pixels respectively.

IOU is the number of correctly predicted animal pixels, over the number of predicted animal pixels plus the number of ground truth animal pixels: $IOU(P,T) = \sum^{N} p_i * t_i/(\sum p_i + \sum t_i)$. This is similar to the dice loss because it focuses on the number of correctly predicted animal pixels.

3.1 Experiments

Experiment 1 - Validation Performance vs Epochs (M25L, M25) The first experiment is investigate how evaluation metrics perform through training. Primarily is there a difference between M25L (supervised with 25% labelled) and M25 (semi-supervised with 25% labelled and 75% unlabelled).

Results are shown on the Left and Middle in *Figure 3*. As expected accuracy is much higher than IOU due to not only focusing on animal correctness. IOU for M25L is a lot more noisy and peaks around epoch 45, while M25 has a much more stable performance, peaking at epoch 80. Going forward we will use the highest performing models from epoch 45 and 80 for M25 and M25L respectively.

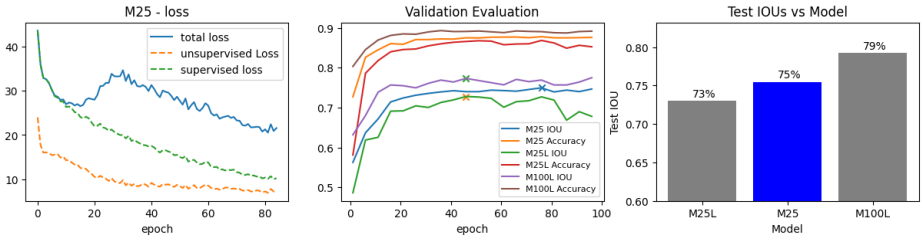

Fig. 3. Left: Loss components of M25 during training. **Middle**: Validation accuracy and IOU of M25, M25L and M100L during training. **Right**: IOU Test performance between M25L, M25 and M100L models

Experiment 2 - Training with Additional Unlabelled Data (M25L, M25, M100L) To investigate the benefit of adding unlabelled data we compare the lower bound (M25L) and upper bound (M100L) models to M25 on Test IOU performance. M25L is trained on 25% labelled training data, M100L is trained on 100% labelled training data. M25 is trained on 25% labelled and 75% unlabelled from training set. Therefore any performance change from M25L to M25 is due to the additional unlabelled data.

Looking at *Fig 3* on the right, we can see M25 is between the upper and lower bounds. This provides evidence M25 is learning useful features from the 75% unlabelled data, but as expected is performing less than the upper bound.

Experiment 3 - Varying the Amount of Labelled Data (All Models) To assess how model performance changes when varying the amount of labelled data, we repeat Experiment 2 with 5% (D5L + D5) and 10% (D10L + D10) labelled data. The results are shown in *Table 2*. We see mixed results for this experiments. When only 5% of the data is labelled we see a 0.7 percentage point (0.7pts) improvement in test IOU from model M05L to M05. But for 10% labelled data we see a 0.5pts reduction in performance when training with unlabelled data.

3.2 Results

Model Bound %	5%	10%	25%
Lower Bound	67.8%	70.5%	73.0%
SSL - MXX	68.5%	70.0%	75.5%
Upper Bound	79.2%	79.2%	79.2%

Table 2. Test IOU of all models and dataset labelled percentage

To summarise the results from the above experiments and *Table 2*, the best performing models were trained between 40-80 epochs, after this we can see over-fitting in the supervised models. For example in the Middle of *Fig 3* M25L validation IOU starts to decrease indicating over-fitting. But the semi-supervised models do not act in the same way. For example M25 in the same figure does not decrease. This could be the consistency cost keeping the models more generalizable for longer.

We also see evidence training with additional unlabelled data can improve performance. We saw this in particular with Model M25 (trained with 25% training data labelled) performing 2.5pts better than M25L. But when reducing the amount of labelled data we saw mixed results. Using unlabelled data with 5% labelled data (M5), we saw marginal increase in performance, and 10% (M10) the performance marginally decreased. We hypothesise this is because the student model requires a certain amount of labelled data to learn enough that the EMA Teacher model can make useful predictions from the unlabelled data.

3.3 Discussion and Further Work

A consequence of our findings is that even if there is not enough labelled data to improve performance when adding unlabelled data, there still maybe a benefit of the model becoming more generalizable.

There are a number of questions this paper has not been able to answer. How does performance vary as different amounts of unlabelled data is included? I believe augmentation plays a critical role in the teacher being able to provide useful labels for the student, therefore further work investigating different augmentation variations could provide useful insights into SSL.

Finally, the original trimap from the OxfordIIIT-Pet dataset has 3 classes, one of these classes is neither animal nor background but we interpreted it as animal. This could have introduced noise into our labelling causing the results to be dampened.

4 Conclusion

From this paper we conclude it is possible to benefit from additional unlabelled data. But we hypothesise when using the mean teacher method, there is a minimum amount of labelled data required to improve performance from additional unlabelled data, but the models could still benefit by being more generalizable.

This is an encouraging result when, in some fields, unlabelled data is ubiquitous and acquiring labelled data is very expensive.

References

1. Authors: *De Man R, Gang GJ, Li X, Wang G.*, Title: *Comparison of deep learning and human observer performance for detection and characterization of simulated lesions.* Publisher: *J Med Imaging (Bellingham)*, Year: 2019
2. Authors: *Chapelle, Olivier, Scholkopf, Bernhard, and Zien, Alexander*, Book: *Semi-Supervised Learning.* Publisher: *MIT Press*, Year: 2006.
3. Author: *Dong-Hyun Lee*, Title: *Pseudo-Label - The Simple and Efficient Semi-Supervised Learning Method for Deep Neural Networks*, In: *www.ResearchGate.net*
4. Authors: *Berthelot, David and Carlini, Nicholas and Goodfellow, Ian and Papernot, Nicolas and Oliver, Avital and Raffel, Colin A*, Title: *MixMatch: A Holistic Approach to Semi-Supervised Learning*, Year: 2019
5. Authors: *Antti Tarvainen and Harri Valpola*, Title: *Weight-averaged consistency targets improve semi-supervised deep learning results*, Year: 2017, In: *CoRR*
6. Authors: *Omkar M Parkhi and Andrea Vedaldi and Andrew Zisserman and C. V. Jawahar*, Title: *Cats and Dogs*, Book: *IEEE Conference on Computer Vision and Pattern Recognition*, Year: 2012
7. Authors: *Connor Shorten, Taghi M. Khoshgoftaar*, Title: *A survey on Image Data Augmentation for Deep Learning*, Year: 2019, Publisher: *Journal of Big Data*
8. Authors: *Olaf Ronneberger, Philipp Fischer, and Thomas Brox*, Title: *U-Net: Convolutional Networks for Biomedical Image Segmentation*, Year: 2015, Publisher: *CoRR*
9. Authors: *Fausto Milletari, Nassir Navab Seyed-Ahmad, and Ahmadi*, Title: *V-Net: Fully Convolutional Neural Networks for Volumetric Medical Image Segmentation*