Introduction

For the purpose of assignment project I decided to take a scientific stance and, together with dr. Stanislaw Jastrzebski and mgr. Maciej Szymczak, we delved into the method commonly refered to as deep ensembles [1] - a very simple, at least in the statement, technique to boost neural networks performance.

More formally the statement is: we train M models to obtain optimized weights θ_1 to θ_M and for a new input sample, the prediction is computed as an average prediction of these models

$$F_{\theta_1,\ldots,\theta_M}(\mathrm{x}) = \frac{1}{M}\sum_{i=1}^{M} f_{\theta_i}(\mathrm{x})$$

Obvious problem is that it requires M times longer training and inference time. The ultimate goal of our research is to develop a method of simulating effects of deep ensembles while training a single model. There is recent research in that field, notably Snapshot Ensembles [5] and Fast Geometric Ensembling [3] but we aim to achieve similar effects by interfere in the early phase of training, which itself is crucial period by means of the generalization as shown in [6].

Results

We have conducted experiments involving mainly CIFAR-10 dataset [8] and SimpleCNN architecture. In this section I will briefly describe two of them: firstly we will ask whether starting epoch of ensembling is relevant and secondly, whether ensemble handles noise in training and testing data.

Delayed Ensembling

The thesis that ensemble owe its performance to diversity of its constituents seems strongly plausible. As described in [1] and [2] the loss surface of neural networks is full of distinct local minima and, surprisingly, while there is no linear path (on which cost remains roughly the same) between any two of them, still there is some path connecting the two, what is a base for FGE method aforementioned above. Our hypothesis states that the early phase of training is indeed crucial for the phenomenon that each member of the ensemble locates a separate minimum.

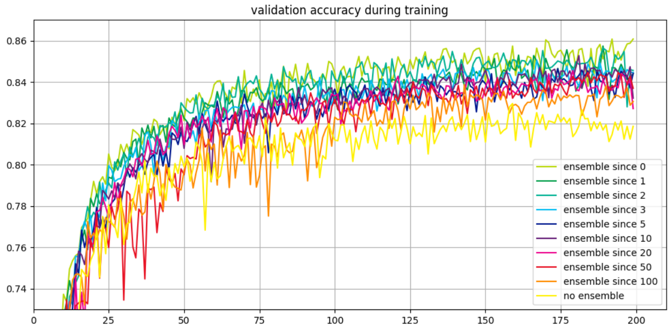

Figure 1: We see that to get most from model ensembling, we should not delay it.

In order to support that, we have visualized how the starting epoch of model ensembling influences the separation of minima (Figure 2) and overall performance (Figure 1). We see that earlier ensembling yields better results but that if ensemble has not started before around 10th epoch, we irrevocably lose diversity of ensemble members - training trajectories are intertwined and all of these oscillates in the same region.

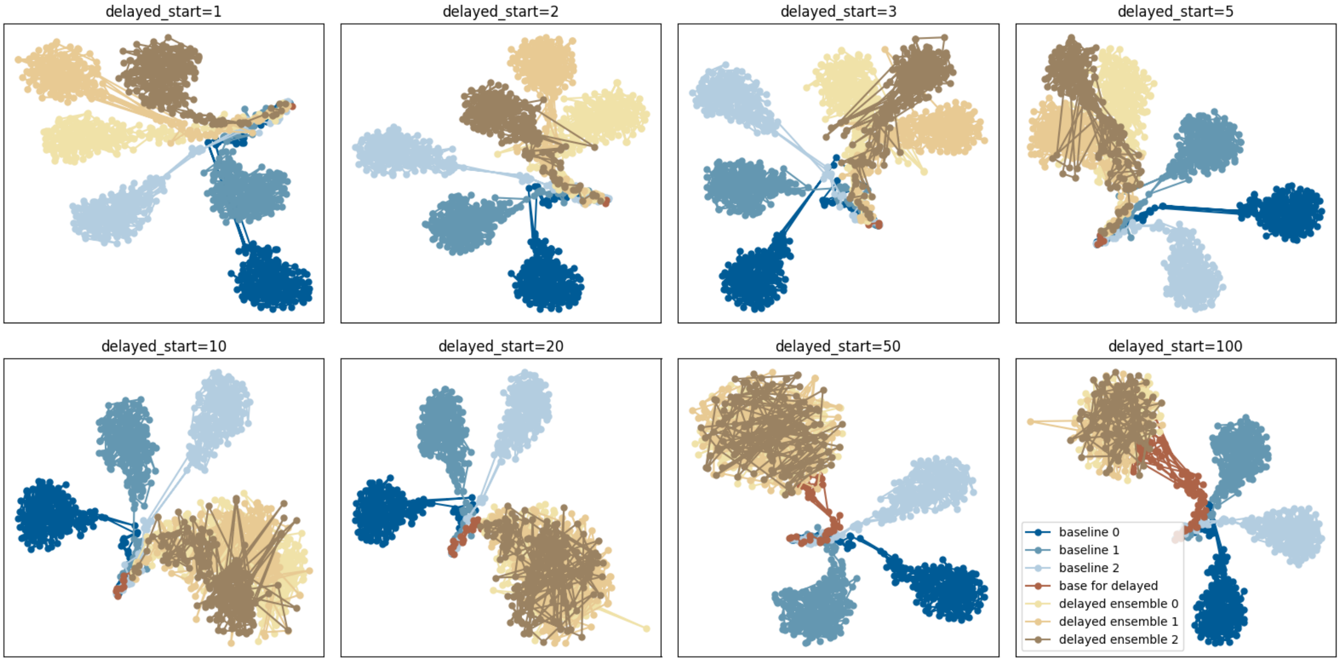

Figure 2: On each plot we see t-SNE projection of seven training trajectories: blueish represent baseline model while crimson and brownish represent delayed model.

Noisy Labels

The second experiment has been conducted in order to determine, if an ensemble of models is less prone to noise in training data, i.e. to data consisting some incorrectly labeled examples, and if it is capable to generalize out of distribution, i.e. when test data comes from different distribution than model is trained on. For testing we use corrupted version of CIFAR-10 from [4].

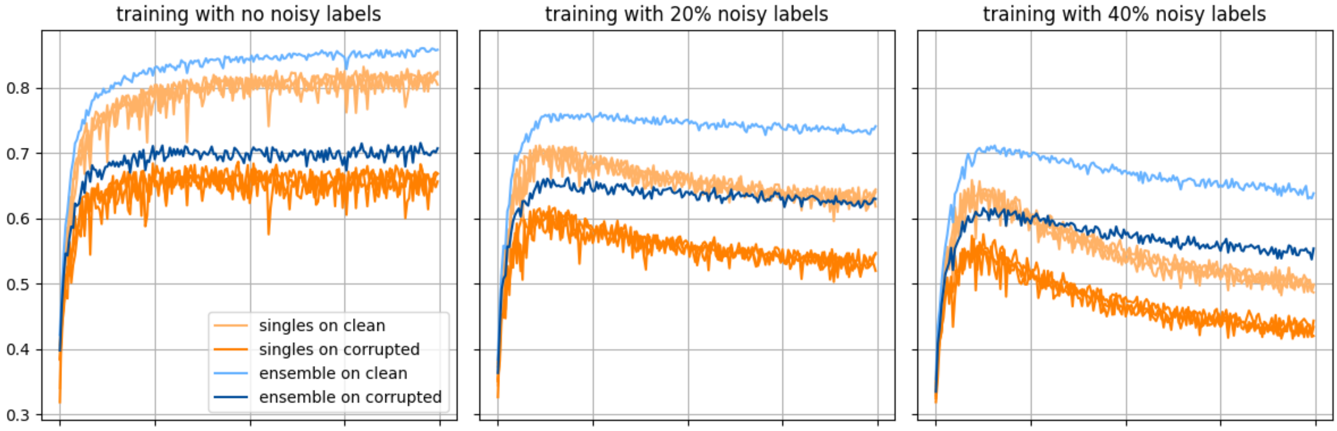

Figure 3: Accuracy plots show that ensemble is quite robust to noised training data.

In order to investigate that we measured models accuracy (Figure 3) and entropy of theirs predictions (Figure 4). We see that as noise in training labels increases, performance of a single model decreases much faster than that of an ensemble. Additionally relative difference between in and out of distribution generalization is getting smaller. From entropy dynamic we can conclude that ensemble is way more stable with its predictions as its level saturates faster than that for single models. It indicates that singles are biased towards some (possibly not only one) classes, they systematically convey less information what means they operate on less complex intrinsic models and ensemble is more resistant to that.

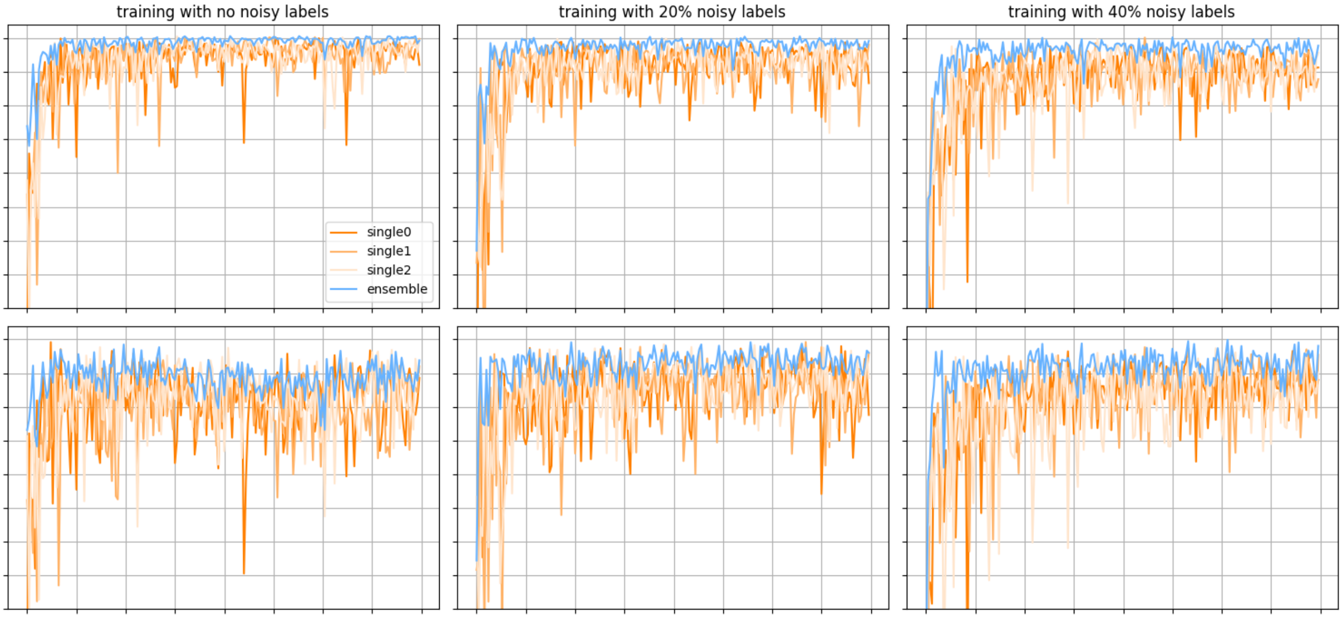

Figure 4: Entropy of predictions for clean (top row) and corrupted (bottom) test sets during training.

Future Work

Clearly, ensembles robustness is the thing that makes this method special. While it is too early to draw conclusions, next steps in our research will be to take closer look at representational differences between single models through CKA as in [7] and exploit the phenomenon of memorization further with the method described by [9].

References

[1] Stanislav Fort, Huiyi Hu, and Balaji Lakshminarayanan. *Deep Ensembles: A Loss Landscape Perspective*. 2020. arXiv: `1912.02757 [stat.ML]`.

[2] Stanislav Fort and Stanislaw Jastrzebski. *Large Scale Structure of Neural Network Loss Landscapes*. 2019. arXiv: `1906.04724 [cs.LG]`.

[3] Timur Garipov et al. *Loss Surfaces, Mode Connectivity, and Fast Ensembling of DNNs*. 2018. arXiv: `1802.10026 [stat.ML]`.

[4] Dan Hendrycks and Thomas Dietterich. *Benchmarking Neural Network Robustness to Common Corruptions and Perturbations*. 2019. arXiv: `1903.12261 [cs.LG]`.

[5] Gao Huang et al. *Snapshot Ensembles: Train 1, get M for free*. 2017. arXiv: `1704.00109 [cs.LG]`.

[6] Stanislaw Jastrzebski et al. *The Break-Even Point on Optimization Trajectories of Deep Neural Networks*. 2020. arXiv: `2002.09572 [cs.LG]`.

[7] Simon Kornblith et al. *Similarity of Neural Network Representations Revisited*. 2019. arXiv: `1905.00414 [cs.LG]`.

[8] Alex Krizhevsky. *Learning Multiple Layers of Features from Tiny Images*. 2009.

[9] Jiaming Song et al. *Robust and On-the-fly Dataset Denoising for Image Classification*. 2020. arXiv: `2003.10647 [cs.LG]`.