A. Implementation details

A.1. Clothing1M

As most previous works, we used ResNet-50 architecture, but did not utilize ImageNet pre-training. For self-supervised pre-training, we used a SimCLR implementation\(^1\) in PyTorch\(^2\), trained on 8 NVIDIA 2080 Ti GPUs for 750 epochs. We trained the network using the AdamW optimizer\([1]\). DivideMix For DivideMix, we used a weight decay of 0.001, and a batch size of 32. As in the case of CIFAR, the warm-up period is five epochs. We trained the network for 120 epochs, with initial learning rate of 0.002, reduced by a factor of 10 after 40 epochs. For each epoch, we sampled 1000 mini-batches from the training data with same amount of samples of every class (according to noisy label). We set \(\lambda_U = 0\). Since a large amount of data is available, we found that increasing value of the threshold to \(\tau = 0.7\) improves the performance of the network.

ELR+ For ELR+, we used the default hyperparameters, except for reduced learning rate (0.001).

A.2. WebVision

DivideMix For WebVision, we also used ResNet-50 architecture. For self-supervised pre-training, we used a SimCLR implementation\(^2\) in PyTorch\([2]\), trained on 8 NVIDIA 2080 Ti GPUs for 1000 epochs. We trained the network using the AdamW optimizer\([1]\) with a weight decay of 0.001, and a batch size of 32. The warm-up period is one epoch. We trained the network for 80 epochs, with initial learning rate of 0.002, reduced by a factor of 10 after 40 epochs. We set \(\lambda_U = 0\).

B. Noise detection analysis

To evaluate the quality of noise detection, in Fig. B.1 we present the ROC-AUC score of noise detection and the effective noise rate, defined as the share of noisy samples in the labeled part of the dataset. C2D demonstrates multiple desired properties including a higher initial score, a much faster rise in separability score as well as a more stable decrease in effective noise level, and eventually a higher overall score and lower noise level. Moreover, even though C2D and the baseline both suffer from decrease in the ROC-AUC score due to overfitting, C2D demonstrated a lower gap between the peak and final scores than the baseline.

References


---

\(^*\)Equal contribution.

\(^1\)https://github.com/HobbitLong/SupContrast

\(^2\)https://github.com/HobbitLong/SupContrast
Figure B.1: Training time ROC-AUC scores (left) and effective noise rates (right). C2D demonstrates higher initial score, faster rise, and more stable decrease in effective noise level.