Mitigating Embedding and Class Assignment Mismatch in Unsupervised Image Classification - Supplementary Material -

Sungwon Han[1][0000−0002−1129−760X], Sungwon Park[1][0000−0002−6369−8130], Sungkyu Park[1][0000−0002−2607−2120], Sundong Kim[2][0000−0001−9687−2409], and Meeyoung Cha[2,1][0000−0003−4085−9648]

[1] Korea Advanced Institute of Science and Technology
{lion4151, psw0416, shaun.park}@kaist.ac.kr
[2] Data Science Group, Institute for Basic Science
{sundong, mcha}@ibs.re.kr

A Code Release

Codes, training data, and the trained model are accessible via an anonymized web link: https://github.com/dscig/TwoStageUC.

B Training Details

Unsupervised Classification Task (Sec. 4.1). ResNet18 with five Norm-FC classification heads was used as the backbone network. Stage 1 of the model used a batch size of 128 and the model was trained in 5 rounds with 200 epochs per round. $w(t)$ was set to zero at first and gradually increased by 0.2 every 80 epochs, following the original work [1]. Stage 2 used the same batch size, and the model was trained with 300 epochs. Stochastic gradient descent with Nesterov momentum 0.9 was utilized as an optimizer. For the first 80 epochs, the learning rate was fixed as 0.01 then scaled-down 0.1 every 40 epochs after the first 80 epochs. Weights in Norm-FC are randomly initialized, and then fixed during training. We found that these fixed weights can avoid training loss fluctuations and lead to the fast convergence of the model. We used four data augmentation techniques: resized crop, horizontal flip, color jitter, and grayscale. Hyper-parameter λ for consistency preserving loss (L_{cp}) is set to 1. Temperatures for both stages (i.e., τ and τ_c) are set to 0.1. This choice is similar to previous studies [3,5]. The auxiliary clustering step in the original IIC model [4] was omitted to keep the number of clusters identical. In the case of STL-10, which includes both unlabeled and labeled images, we used the full dataset for training Stage 1's encoder and used the labeled images in Stage 2.

Semi-Supervised Classification Task (Sec. 4.4). WideResNet28-2 [6] was used as the backbone network for all models. This network was first pretrained with given datasets in an unsupervised fashion (based on our algorithm). We

then connected a linear classifier to the pretrained CNN network and trained the network. For the fully-supervised model, we trained the model upon pretrained weights for 180 epochs. SGD with Nesterov momentum 0.9 was used as an optimizer. The learning rate was set as 0.005 for CIFAR-10 and 0.001 for SVHN initially, then were scaled-down every 40 epochs after the first 80 epochs to 0.1. The same data augmentation technique was used except for the horizontal flip on SVHN. Affine transform was used for SVHN instead of a horizontal flip not to deform the contents. When applying our pretrained model to other semi-supervised algorithms, we ensured to follow the training details of the corresponding works.

C Hyper-parameter Analysis

Hyper-parameters affect the final classification accuracy. We show the effect of two parameters: the weight factor λ in the second stage loss from Equation 1 and the temperature τ_c in a softmax function from the normalized fully connected layer from Equation 2.

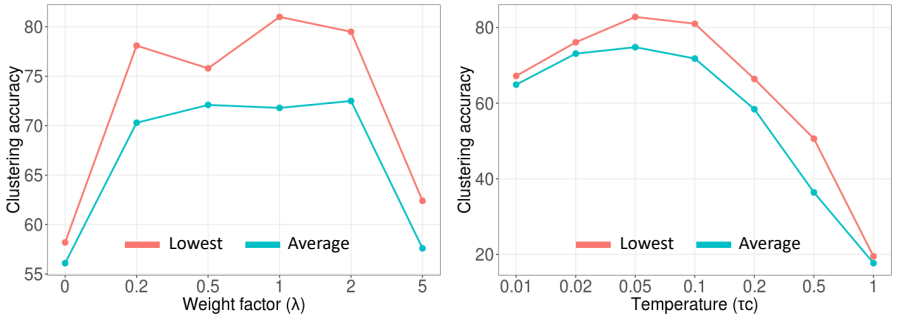
$$L_{stage2} = L_{assign} + \lambda \cdot L_{cp} \tag{1}$$

$$y_i^j = \frac{\exp(\frac{\mathbf{w}_j}{||\mathbf{w}_j||} \cdot \mathbf{v}_i / \tau_c)}{\sum_k \exp(\frac{\mathbf{w}_k}{||\mathbf{w}_k||} \cdot \mathbf{v}_i / \tau_c)} \tag{2}$$

Figure 1 shows the effect of weight and temperature changes in the fully connected layer. The red line plots the accuracy of the head with the lowest training loss among five heads, and the blue line plots the averaged accuracy over five heads. Both lines exhibit similar trends that the accuracy reaches the top for specific hyper-parameters values and drops, otherwise.

In the case of λ, the accuracy reaches the highest value at $\lambda = 1$ and decreases when the weight of consistency preserving loss changes. Consistency preserving loss can be regarded as the loss controlling the embedding quality, while L_{assign} can be regarded as that controlling the class assignment quality. Hence, λ controls the balance between the two, and extremely small or large λ values will break the equilibrium.

Meanwhile, the temperature value, τ_c, controls the degree of concentration of feature vectors projected in the unit sphere. The temperature value is critical for correctly training the classifier. A possible explanation might be the topological characteristics of the normalized vector, i.e., it has confined space, and the temperature is critical to amplifying specific signals. When τ_c becomes larger and gets closer to 1, its concentration effect diminishes, and weak signal results in poor performance. However, if τ_c gets smaller and closer to 0, the signal is hugely amplified, and small perturbations affect a lot during the training. As a result, moderate value is necessary for τ_c. By utilizing the optimal joint values of λ and τ_c upon our proposed model, the performance of unsupervised image classification could be further enhanced.



(a) Effect of weight (λ) (b) Effect of temperature (τ_c)

Fig. 1: Analysis of accuracy on the CIFAR-10 dataset across (a) the changing weight factor λ in the second stage and (b) the temperature value τ_c in the normalized fully-connected layer. The red line plots the accuracy of the head with lowest training loss among five heads, and the blue line plots the averaged accuracy over five heads.

D Further discussion on the effect of pretraining

According to the conclusion from previous work [2], pretraining may only contribute to speed up the model convergence and does not necessarily lead to an improvement in the accuracy of downstream supervised learning tasks. To investigate whether this finding also holds for the current unsupervised problem setting, we additionally conducted a simple experiment on CIFAR-10. We skipped the first stage's pretraining and only trained the second stage model with larger training epochs (300 $\rightarrow$ 1,300 epochs). However, these changes rather result in a significantly lower performance than our full model (81.0% $\rightarrow$ 59.1%), and even did not show a substantial enhancement in performance compared to our second stage only model with 300 epochs (58.6% $\rightarrow$ 59.1%). We speculate this result is due to its different nature of 'unsupervised' setting, i.e., we do not have any labels and thereby we can only solve auxiliary tasks with many sub-optimal solutions. Although longer training epochs can enable the model convergence, finding a generalizable solution beyond the auxiliary task seems a different problem.

References

1. Han, S., Xu, Y., Park, S., Cha, M., Li, C.T.: A Comprehensive Approach to Unsupervised Embedding Learning based on AND Algorithm. arXiv preprint arXiv:2002.12158 (2020)
2. He, K., Girshick, R., Dollár, P.: Rethinking imagenet pre-training. In: Proc. of the IEEE International Conference on Computer Vision (ICCV). pp. 4918–4927 (2019)
3. Huang, J., Dong, Q., Gong, S., Zhu, X.: Unsupervised deep learning by neighbourhood discovery. In: Proc. of the International Conference on Machine Learning (ICML). pp. 2849–2858 (2019)
4. Ji, X., Henriques, J.F., Vedaldi, A.: Invariant information clustering for unsupervised image classification and segmentation. In: Proc. of the IEEE International Conference on Computer Vision (ICCV). pp. 9865–9874 (2019)
5. Ye, M., Zhang, X., Yuen, P.C., Chang, S.F.: Unsupervised embedding learning via invariant and spreading instance feature. In: Proc. of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR). pp. 6210–6219 (2019)
6. Zagoruyko, S., Komodakis, N.: Wide residual networks. arXiv preprint arXiv:1605.07146 (2016)