Domain-Agnostic Clustering with Self-Distillation

Mohammed Adnan[1,2], **Yani A. Ioannou**[3], **Chuan-Yung Tsai**[2], **Graham W. Taylor**[1,2]

[1]University of Guelph, [2]Vector Institute for AI, [3]University of Calgary

{madnan01, gwtaylor}@uoguelph.ca,
yani.ioannou@ucalgary.ca, kenyon.tsai@vectorinstitute.ai

Abstract

Recent advancements in self-supervised learning have reduced the gap between supervised and unsupervised representation learning. However, most self-supervised and deep clustering techniques rely heavily on data augmentation, rendering them ineffective for many learning tasks where insufficient domain knowledge exists for performing augmentation. We propose a new self-distillation based algorithm for domain-agnostic clustering. Our method builds upon the existing deep clustering frameworks and requires no separate student model. The proposed method outperforms existing domain agnostic (augmentation-free) algorithms on CIFAR-10. We empirically demonstrate that knowledge distillation can improve unsupervised representation learning by extracting richer 'dark knowledge' from the model than using predicted labels alone. Preliminary experiments also suggest that self-distillation improves the convergence of DeepCluster-v2.

1 Introduction

In recent years, the representation learning community has put much focus on learning with minimal labeled data or even no labeled data. Self-Supervised Learning (SSL) refers to the class of learning algorithms that use no human labels. Rather, SSL obtains self-supervisory signals from the data itself, often leveraging its underlying structure. These self-supervisory signals are obtained using pretext tasks such as predicting the rotation of images [1], solving jigsaw puzzles [2], mixup [3], and colorization [4]. Most of the methods use contrastive energy-based pretext tasks. SimCLR [5] proposed a simple framework for learning visual representation by minimizing the distance between different views (augmentation) of the same instance via contrastive loss. Since computing embeddings for the entire dataset is not feasible, memory bank based approaches have also been proposed. He et al. [6] proposed MoCo, a dynamic dictionary-based framework for unsupervised learning of visual representations. Other methods such as DeepCluster [7], ClusterFit [8], SwAV [9], Barlow Twins [10], and BYOL [11] are non-contrastive and use feature clustering to learn representations. They use various pretext tasks such as computing target embeddings to group similar images or minimize the redundancy between the individual components of the embedding vector. However, designing pretext tasks requires domain knowledge and hinders the application of SSL as an out-of-the-box solution for arbitrary domains where domain knowledge is not available. Thus, it is necessary to develop domain agnostic (data augmentation free) clustering and SSL algorithms.

In this work, we propose a novel self-distillation based domain-agnostic clustering algorithm which does not require any domain knowledge for designing pretext tasks or data augmentation. We also demonstrate that knowledge distillation can provide further self-supervisory signals from soft labels (i.e. dark knowledge).

Workshop on Self-Supervised Learning: Theory and Practice, NeurIPS 2021.

2 Background

Knowledge Distillation. Knowledge Distillation (KD) is a model compression method in which a smaller 'student' model is trained to mimic the behavior of a large 'teacher' model by minimizing the loss on the output class probabilities (soft labels) of the large model [12]. It has been found that the smaller model achieves similar, or often better performance than the original model. This behavior has been attributed to the presence of 'dark knowledge' present in the soft labels. Soft labels provide much more information about the semantic information present in the image. For example, given a dog image from CIFAR-10 [13], the class probability of the image being a cat will be much higher than the class probability of the image being a car. Thus, the softmax values give additional hints to the network that images of dogs and cats contain similar semantic information. Knowledge distillation also improves the loss landscape and helps find flat minima, which in turn improves generalization. Distillation has been shown to amplify regularization in the Hilbert space, and thus, it improves generalization [14]. In *self-distillation* rather than using a separate student and teacher network, a single model is used to extract 'dark knowledge' [15, 16, 17, 18, 19, 20]. The main objective in self distillation is to distill the knowledge from the deeper layers to the shallow layers of the network [21].

DeepCluster. Caron et al. proposed Deep Clustering [7] (DeepCluster) to jointly learn the neural network and cluster the resulting features by iteratively applying k-means. DeepCluster can learn generalizable features in an unsupervised manner using clustering as a pretext task. Since convolutional layers encode a strong prior for learning natural images, a randomly initialized CNN achieves 12% accuracy on ImageNet while a random prediction would be 0.1% for a randomly-initialized fully connected model [2]. DeepCluster exploits this weak signal to bootstrap the discriminative power of a CNN. Given a CNN mapping denoted by f_θ, a classifier parameterized by g_w and a training set of $X = \{x_1, x_2 \ldots, x_n\}$, it clusters the output of a CNN and subsequently clusters the output features to optimize:

$$\min_{\theta, W} \frac{1}{N} \sum_{n=1}^{N} \ell(g_w(f_\theta(x_n)), y_n). \tag{1}$$

This process is repeated iteratively until the model converges. After training, the classifier g_w is discarded, and the CNN f_θ can be used for downstream learning tasks. Since there is no mapping between assignments in two consecutive epochs, the classifier (g_w) is re-initialized after each epoch. However, training DeepCluster is not trivial, as with any method that attempts to jointly learn a discriminative classifier and labels. Pseudo-labels can collapse to one class, or it is possible that some of the clusters will be empty. The collapse happens when the CNN outputs similar features for different inputs. In either case, the training loss decreases without progress in learning discriminative features. DeepCluster-v2 [9] uses centroids given by spherical k-means to initialize the classifier weights after each epoch which improves overall stability.

Domain agnostic clustering. Current state-of-the-art Self-Supervised Learning (SSL) and clustering algorithms use data augmentation either for learning contrastive representations or as a regularizer. However, in many scenarios, data augmentation techniques cannot be used. For example, *color jittering* commonly used in SSL cannot be used on black and white x-ray images [22], and *random cropping* [23] is not relevant to histopathology images where the region of interest is significantly smaller than the image itself. Thus, it is important to develop domain agnostic SSL and clustering algorithms that don't require the design of domain-aware components such as augmentations. Recently, some methods have been proposed for domain agnostic SSL. Verma et al. [24] proposed a domain agnostic algorithm by using mixup noise at the hidden-state level. Viewmaker Network [25] uses a generative model to generate different views or augmentation, which can then be used for domain agnostic SSL.

Impact of data augmentation on clustering. For SSL techniques where data augmentation is not a critical part of the supervisory signal, one way to achieve domain agnosticism is to remove the augmentation, incurring a decrease in accuracy. However, SSL methods rely heavily on data augmentation, and their performance decreases significantly after removing data augmentation. Various studies have shown the impact of data augmentation on SSL and clustering techniques. Deshmukh et al. in [26] noted that by just removing *random crop* from the data augmentation pipeline, the accuracy on CIFAR-10 [13] drops from 84.59% to 29.88%. Tao et al. [27] also reported a decrease in the accuracy from 81.5% to 23.6% after removing data augmentations. Chen et al. also

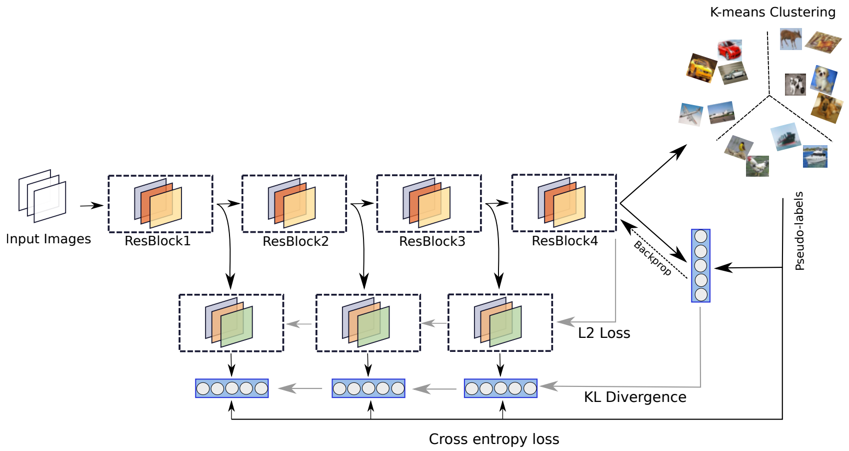

Figure 1: Schematic overview of our method. Bottleneck branches between ResBlocks act as student models and the deepest layer acts as a teacher model. Arrows in grey denote distillation loss (KL divergence and L2 loss). Each classifier has cross-entropy loss from the pseudo-labels given by k-means.

observed that data augmentation is crucial for SimCLR to learn good visual representations [5]. One explanation of the decrease in the accuracy after removing data augmentations could be attributed to the generalization property of data augmentations. Similar to explicit regularization techniques such as weight decay and dropout, data augmentations have been shown to act as an implicit regularizer that improves generalization [28].

3 Methodology

Motivation. The underlying principle behind our proposed method is that semantically similar images of different classes are often embedded close in Euclidean space and are consequently misclassified by k-means. However, soft labels may provide more information about semantically similar classes, and thus by using self-distillation, the model can be provided with further information to distinguish those inputs. Moreover, self-distillation can partially substitute the regularizing effect of data augmentation and help find flat minima, which in turn improves generalization [14].

Problem Statement. Given an unlabelled dataset $X = \{x_1, x_2, \dots x_n\}$, our objective is to learn generalizable features $f_\theta(x_i)$ without using any domain-specific data augmentation, which can then be used for learning downstream tasks by only training a linear classifier g_w.

Proposed Approach. Our proposed method uses the DeepCluster-v2 [9] framework while also introducing a self-distillation loss. We use ResNet [29] as the CNN for extracting features and introduce a self-distillation loss [15]. The ResNet architecture is modified to have an additional three bottleneck branches. Secondly, an auxiliary classifier is added on top of bottleneck branches as shown in the Figure 1. During the training phase, all three bottleneck classifiers $(q_i, i = 1, \dots, 3)$ along with the original classifier (q_c) are utilized. Bottleneck classifiers are trained as student models via distillation from the deepest classifier, which acts as the teacher model. To improve the overall performance, Zhang et al. [15] introduced three losses:

1. Loss Source 1: All bottleneck classifiers (student models) have cross entropy loss from the pseudo-labels obtained from k-means clustering denoted by $y_k(x)$:

$$\mathcal{L}_i = \sum_{\forall x} y_k(x) \log(q_i(x)) \qquad ; i \in \{1, 2, 3, c\}. \tag{2}$$

2. Loss Source 2: Kullback-Leibler (KL) divergence loss from the softmax output of the deepest classifier (teacher model). In this way, hidden knowledge from the softmax is infused in to the hidden layers for learning better representations:

$$\mathcal{L}_{KL} = \sum_{\forall x} q_c(x) \log(\frac{q_c(x)}{q_i(x)}). \tag{3}$$

3. Loss Source 3: L2 loss between the bottleneck feature map and the deepest layer features is added to provide implicit knowledge or hints to the bottleneck classifiers:

$$\mathcal{L}_{hints} = \|F_i - F_C\|_2 \,. \tag{4}$$

During training, all three distillation losses are combined with the DeepCluster loss (L_{Dc}). The overall loss is given as:

$$\mathcal{L}_{total} = \mathcal{L}_c + (1-\alpha). \sum_{i=1,2,3} \mathcal{L}_i + \alpha.\mathcal{L}_{KL} + \lambda.\mathcal{L}_{hints}. \tag{5}$$

4 Experiments

We evaluate the performance of our approach on the CIFAR-10 dataset [13]. CIFAR-10 contains 50,000 labeled training images and 10,000 test images belonging to ten different classes. We train our model on 50,000 training images without using any ground labels. In contrast to most existing unsupervised learning algorithms, we do not use any domain knowledge for data augmentations. The trained model can be used to extract general-purpose features for downstream machine learning tasks. We evaluate the trained model on 10,000 images. Implementation details are provided in the Appendix A.

Evaluation Metric. To evaluate the trained model (f_θ), we freeze the network and train a linear classifier on top of the frozen CNN using training samples from the CIFAR-10. Test data is used to evaluate the accuracy, which quantitatively measures the quality of the features learned in an unsupervised manner within the frozen CNN layers.

Results. We compare our proposed approach with existing domain agnostic or data augmentation-free clustering methods. Since most of the current clustering methods use data augmentation either for contrastive learning or as a regularizer, we compare with the performance of existing methods when trained without data augmentation. Our method outperforms other methods on CIFAR-10 and shows significant improvement over DeepCluster-v2 as shown in Table 1.

Method	Accuracy
ID [27]	18.7%
IDFD [27]	23.6%
ConCURL [26]	29.88%
DeepCluster-v2 [9]	33.27 ± 0.06 %
DeepCluster+KD (ours)	**38.00 ± 0.34%**

Table 1: Domain Agnostic Clustering on CIFAR-10. DeepCluster was trained with no data augmentation.

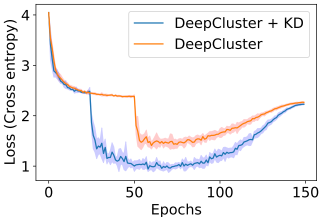

Figure 2: Evolution of the training loss. Proposed method converges faster than the DeepCluster-v2.

Stability. Preliminary experiments suggest that knowledge distillation helps in faster convergence and stability of DeepCluster training. Evaluation of the training loss with respect to the number of epochs is shown in Figure 2. All the training runs with different random initializations followed a similar trend suggesting improved stability and convergence due to self-distillation.

5 Conclusions and future work

In this work, we proposed that extracting hidden dark knowledge via self-distillation from the dataset further improves the performance of unsupervised clustering algorithms. Preliminary experiments on CIFAR-10 show that self-distillation improves the existing state-of-the-art method by 4.5%. In subsequent work, we would also benchmark the proposed algorithm on ImageNet and on the DABS: Domain Agnostic Benchmark for Self-Supervised Learning dataset [22] to study the benefits of distillation on large datasets.

References

[1] S. Gidaris, P. Singh, and N. Komodakis, "Unsupervised representation learning by predicting image rotations," 2018.

[2] M. Noroozi and P. Favaro, "Unsupervised learning of visual representations by solving jigsaw puzzles," 2017.

[3] H. Zhang, M. Cisse, Y. N. Dauphin, and D. Lopez-Paz, "mixup: Beyond empirical risk minimization," 2018.

[4] R. Zhang, P. Isola, and A. A. Efros, "Colorful image colorization," 2016.

[5] T. Chen, S. Kornblith, M. Norouzi, and G. Hinton, "A simple framework for contrastive learning of visual representations," 2020.

[6] K. He, H. Fan, Y. Wu, S. Xie, and R. Girshick, "Momentum contrast for unsupervised visual representation learning," 2020.

[7] M. Caron, P. Bojanowski, A. Joulin, and M. Douze, "Deep Clustering for Unsupervised Learning of Visual Features," *arXiv:1807.05520 [cs]*, Mar. 2019. arXiv: 1807.05520.

[8] X. Yan, I. Misra, A. Gupta, D. Ghadiyaram, and D. Mahajan, "Clusterfit: Improving generalization of visual representations," 2019.

[9] M. Caron, I. Misra, J. Mairal, P. Goyal, P. Bojanowski, and A. Joulin, "Unsupervised Learning of Visual Features by Contrasting Cluster Assignments," *arXiv:2006.09882 [cs]*, Jan. 2021. arXiv: 2006.09882.

[10] J. Zbontar, L. Jing, I. Misra, Y. LeCun, and S. Deny, "Barlow twins: Self-supervised learning via redundancy reduction," 2021.

[11] J.-B. Grill, F. Strub, F. Altché, C. Tallec, P. H. Richemond, E. Buchatskaya, C. Doersch, B. A. Pires, Z. D. Guo, M. G. Azar, B. Piot, K. Kavukcuoglu, R. Munos, and M. Valko, "Bootstrap your own latent: A new approach to self-supervised learning," 2020.

[12] G. Hinton, O. Vinyals, and J. Dean, "Distilling the knowledge in a neural network," 2015.

[13] A. Krizhevsky, "Learning multiple layers of features from tiny images," tech. rep., 2009.

[14] H. Mobahi, M. Farajtabar, and P. L. Bartlett, "Self-distillation amplifies regularization in hilbert space," 2020.

[15] L. Zhang, J. Song, A. Gao, J. Chen, C. Bao, and K. Ma, "Be your own teacher: Improve the performance of convolutional neural networks via self distillation," 2019.

[16] Y. Hou, Z. Ma, C. Liu, and C. C. Loy, "Learning lightweight lane detection cnns by self attention distillation," 2019.

[17] Z. Zhang and M. R. Sabuncu, "Self-distillation as instance-specific label smoothing," 2020.

[18] C. Yang, L. Xie, C. Su, and A. L. Yuille, "Snapshot distillation: Teacher-student optimization in one generation," 2018.

[19] M. Phuong and C. H. Lampert, "Distillation-based training for multi-exit architectures," *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 1355–1364, 2019.

[20] X. Lan, X. Zhu, and S. Gong, "Self-referenced deep learning," 2018.

[21] J. Gou, B. Yu, S. J. Maybank, and D. Tao, "Knowledge distillation: A survey," *International Journal of Computer Vision*, vol. 129, p. 1789–1819, Mar 2021.

[22] H. Jun, A. Radford, C. Igel, and C. Chute, "Dabs: A domain-agnostic benchmark for self-supervised learning," 2021.

[23] R. Takahashi, T. Matsubara, and K. Uehara, "Data augmentation using random image cropping and patching for deep cnns," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 30, p. 2917–2931, Sep 2020.

[24] V. Verma, T. Luong, K. Kawaguchi, H. Pham, and Q. Le, "Towards Domain-Agnostic Contrastive Learning," in *Proceedings of the 38th International Conference on Machine Learning*, pp. 10530–10541, PMLR, July 2021. ISSN: 2640-3498.

[25] A. Tamkin, M. Wu, and N. Goodman, "Viewmaker Networks: Learning Views for Unsupervised Representation Learning," *arXiv:2010.07432 [cs]*, Mar. 2021. arXiv: 2010.07432.

[26] A. A. Deshmukh, J. R. Regatti, E. Manavoglu, and U. Dogan, "Representation Learning for Clustering via Building Consensus," *arXiv:2105.01289 [cs]*, May 2021. arXiv: 2105.01289 version: 1.

[27] Y. Tao, K. Takagi, and K. Nakata, "Clustering-friendly Representation Learning via Instance Discrimination and Feature Decorrelation," *arXiv:2106.00131 [cs]*, May 2021. arXiv: 2106.00131 version: 1.

[28] A. Hernández-García and P. König, "Data augmentation instead of explicit regularization," 2020.

[29] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," 2015.

[30] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," 2017.

A Appendix

A.1 Implementation Details

We used PyTorch for implementing the proposed method and official DeepCluster code available at `https://github.com/facebookresearch/swav/blob/main/main_deepclusterv2.py`. For easier reproducibility of our method, we provide the hyper-parameters value for our experiment on CIFAR-10 below.

Optimizers. We used SGD for training CNN with the following hyper-parameters:

1. Base Learning Rate = $6e-2$
2. Final Learning Rate = $3e-4$
3. Momentum = 0.9
4. Weight Decay = $1e-6$
5. Epochs = 150
6. Batch size = 256
7. Warmup epochs = 5
8. Warmup start learning rate = $1e-6$

DeepCluster hyper-parameters. We removed the data augmentation and muti-crop function in the original code and use following hyper-parameters:

1. Number of prototypes = 60
2. Temperature = 0.5
3. Output feature dimension = 128
4. Hidden layer dimension = 1024
5. Number of iteration with frozen prototypes = 5000

Self-Distillation. In Equation 5 of self-distillation loss, α and λ were set to be 0.9 and $1e-5$ respectively.

Linear Classifier. We trained linear classifier using Adam optimizer [30] with learning rate = $1e-3$ for 200 epochs.