SUPERCM: REVISITING CLUSTERING FOR SEMI-SUPERVISED LEARNING

Durgesh Singh, Ahcène Boubekki, Robert Jenssen, Michael C. Kampffmeyer

Department of Physics and Technology, UiT The Arctic University of Norway, Tromsø

ABSTRACT

The development of semi-supervised learning (SSL) has in recent years largely focused on the development of new consistency regularization or entropy minimization approaches, often resulting in models with complex training strategies to obtain the desired results. In this work, we instead propose a novel approach that explicitly incorporates the underlying clustering assumption in SSL through extending a recently proposed differentiable clustering module. Leveraging annotated data to guide the cluster centroids results in a simple end-to-end trainable deep SSL approach. We demonstrate that the proposed model improves the performance over the supervised-only baseline and show that our framework can be used in conjunction with other SSL methods to further boost their performance.

Index Terms— Clustering, Semi-supervised learning, Gaussian mixture models

1. INTRODUCTION

Traditional deep learning has achieved state-of-the-art performance on various tasks at the cost of large-scale supervised training data. However, it is difficult to obtain such a dataset in many applications, due to an expensive and time-consuming annotation process [1]. Several approaches have tackled this dependency by exploiting information from the otherwise abundant unlabeled data, and thereby improving the existing model performance. Semi-supervised Learning (SSL) [2] is one such paradigm that addresses the problem of label scarcity. SSL methods depend on the clustering assumption [3] and mostly leverage consistency regularization [4, 5, 6, 7] or entropy minimization [8, 9, 10, 11] to enforce the same. However, the success of consistency regularization and entropy minimization methods depends on the choice of an appropriate perturbation strategy or the quality of the estimated pseudo-labels, respectively. This has led to complex training mechanisms that incorporate different perturbation strategies and rely on various pseudo-label heuristics to achieve greater performance [6, 11].

Visual Intelligence publications are financially supported by the Research Council of Norway, through its Centre for Research-based Innovation funding scheme (grant no. 309439), and Consortium Partners. The authors are part of the UiT Machine Learning Group: https://machine-learning.uit.no.

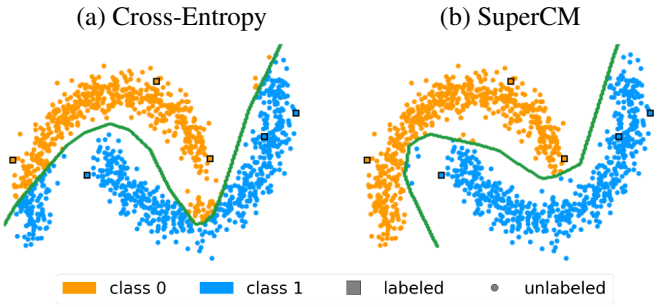

Fig. 1. Comparison of Cross-Entropy and SuperCM on the two moons dataset for 3 labeled examples per class out of a total of 1600 datapoints. Each approach leverages a feature extractor consisting of a fully connected neural network with three hidden layers, each with 10 neurons.

In this work, we take a different approach and study the problem of SSL from the clustering perspective. Recent deep clustering approaches partition high dimension input features by performing clustering and representation learning simultaneously [12]. Owing to the recent success of these methods, we develop a novel training approach that takes inspiration from the Gaussian mixture model (GMM), and utilizes a one layer auto-encoder called the clustering module (CM) [13] for the SSL task. Our method is called Semi-**Super**vised **C**lustering **M**odule (SuperCM), which does not rely on a complex training scheme and achieves performance improvement with respect to its supervised-only baseline. As an illustration, we show in Figure 1, how SuperCM compares to a model trained with a vanilla Cross-Entropy (CE), and how it results in better separation of the two moons dataset. The SuperCM, due to the differentiable nature of the CM, can be further incorporated as a regularizer in other gradient-based SSL methods to boost their performance, thus opening new avenues in clustering-based SSL research. We summarize the contribution of our work as follows:

- We develop a simple approach that builds on a differentiable clustering module and explicitly enforces the clustering assumption in the SSL task.

- Our SuperCM is complimentary to other SSL methods and improves their performance significantly when the number of annotated samples is low.

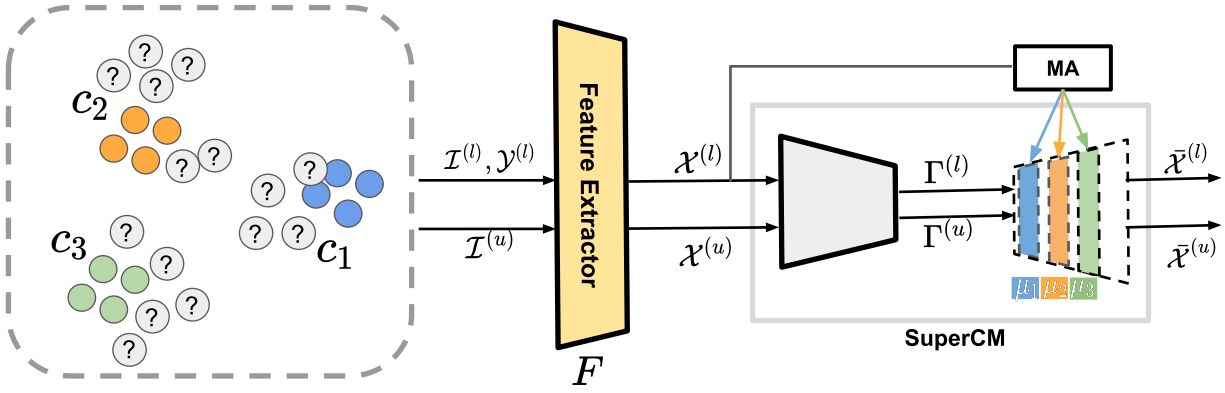

Fig. 2. Architecture of the SuperCM. The feature extractor and the CM's encoder are trained with gradient descent using both labeled (colored) and unlabeled (gray) data. The centroids of the CM are updated as the class-wise moving average (MA) of the labeled data.

2. RELATED WORKS

To set the stage for SuperCM, we discuss relevant deep SSL and clustering approaches. For a more extensive survey the interested reader is referred to [3, 12].

2.1. Semi-Supervised Learning

We first discuss representative methods based on consistency regularization and entropy minimization for the SSL task. Consistency regularization methods encourage invariant predictions for different input perturbations of the same unlabeled data point. One prominent example of consistency-based methods is Virtual Adversarial Training (VAT) [6], which uses the concept of adversarial attacks for consistency regularization. It aims to find a perturbation to the input data in an adversarial direction, and enforces the consistency between the model predictions for the original input and its perturbation. Moreover, entropy minimization [8] encourages low entropy on the model predictions for the unlabeled data. One representative method of entropy minimization is Pseudo-label [9] that generates high-confidence proxy labels for the unlabeled data to guide the training. While these approaches have shown performance improvement in the supervised-only baseline, they rely on complex training strategies to achieve the same.

2.2. Deep Clustering

In this section, we discuss deep clustering approaches [12], which aim to cluster high dimensional data with the help of neural networks. Early methods [14, 15, 16] perform iterative training by minimizing the clustering loss for learning deep features and using the updated features to estimate new cluster labels. However, this alternating training strategy can be sub-optimal. Recent approaches perform joint representation learning and clustering simultaneously to obtain more cluster-friendly embedding. One prominent approach of simultaneous clustering is the CM [13], which uses a one-layer auto-encoder with a deep feature extractor and performs clustering by minimizing a GMM-based clustering loss along with a suitable representation learning loss. We briefly describe the CM in the next section and discuss how it can be extended to the learning in the SSL scenario.

3. METHOD

3.1. Clustering Module

As the key building block of our SSL approach, we first describe the CM introduced in [13]. The model aims to maximize a differentiable, rephrased version of the $\mathcal{Q}$-function of a Gaussian mixture model. The loss function of the CM can be stated as follows:

$$\mathcal{L}_{\text{CM}} = \frac{1}{N}\left(\sum_{i=1}^{N}||\boldsymbol{x}_i - \bar{\boldsymbol{x}}_i||^2 + \sum_{i=1}^{N}\sum_{k=1}^{K}\gamma_{ik}(1-\gamma_{ik})||\boldsymbol{\mu}_k||^2 - \sum_{i=1}^{N}\sum_{\substack{k,l=1 \\ k\neq l}}^{K}\gamma_{ik}\gamma_{il}\boldsymbol{\mu}_k^T\boldsymbol{\mu}_l + \sum_{k=1}^{K}(1-\alpha_k)\log\tilde{\gamma}_k\right) \quad (1)$$

where $N > 0$ is the number of data points and $K > 0$ is the number of clusters. An input data $\boldsymbol{x}_i \in \mathcal{X} \subset \mathbb{R}^d$ has a label $y_i \in \mathcal{Y}$ and its reconstruction by the CM is denoted $\bar{\boldsymbol{x}}_i \in \mathbb{R}^d$. The CM's encoder, with softmax activation, outputs the set $\Gamma = \{\gamma_{ik} = P(y_i = k|\boldsymbol{x}_i)\}$ which corresponds to the posterior probabilities of the cluster assignment, also known as cluster responsibilities. The centroids of the model, denoted $\boldsymbol{\mu}_k \in R^d$, are the weights of the CM's decoder. Finally, the cluster probabilities are controlled by a Dirichlet prior over the $\tilde{\gamma}_k = \mathbb{E}_{\mathcal{X}}(\gamma_{ik})$ with concentration $\alpha > 1$.

3.2. SuperCM

An intuitive SSL extension of the CM would be to prepend a feed-forward neural network to the CM module and leverage

Table 1. Results for Top-1 accuracy with CE and SuperCM for different SSL base models on the CIFAR-10 dataset. None denotes the SSL setting without a base model where performance of CE and SuperCM is compared. Bold numbers indicate statistically significant improvements (t-test, $p < 0.05$).

	600 labels		4000 labels	
SSL base model	CE	SuperCM	CE	SuperCM
None	56.94±0.46	**62.14±1.40**	78.65±0.45	**82.26±0.26**
Pseudo-Label [9]	61.05 ±1.25	**65.19 ±2.52**	84.97±0.22	85.19 ±0.47
VAT [6]	68.43 ±0.89	**75.23 ±3.92**	86.82 ±0.19	86.69 ±0.11

the supervised data through the CE loss over the posterior probabilities. However, we observe that this could lead to trivial solutions as the training of the backbone is too fast. We overcome this obstacle by learning the centroids as a class-wise average of the labeled data instead of gradient descent. Specifically, we rely on a moving average to prevent frequent noisy updates. Computing the centroids this way also prevents them from collapsing and makes the Dirichlet prior along with its hyper-parameter α unnecessary. We call this approach SuperCM and show a schematic representation of the model in Figure 2.

At each iteration t, the input consists of $n^{(l)}$ labeled data pairs $\{\mathcal{I}^{(l)}, \mathcal{Y}^{(l)}\}$ and $n^{(u)}$ unlabeled data $\mathcal{I}^{(u)}$. Both inputs are first transformed by a feature extractor F into $\mathcal{X}^{(l)} = F\big(\mathcal{I}^{(l)}\big)$ and $\mathcal{X}^{(u)} = F\big(\mathcal{I}^{(u)}\big)$. The centroids are then updated using the labeled data as follows:

$$\boldsymbol{\mu}_k = \frac{t-1}{t}\boldsymbol{\mu}_k + \frac{1}{t}\frac{1}{n^{(l)}}\sum_{i=1}^{n^{(l)}} \mathbb{1}_{(y_i^{(l)}=k)}\boldsymbol{x}_i^{(l)} \tag{2}$$

Finally, the labeled and unlabeled data are concatenated and passed through the CM to obtain the cluster probabilities $\Gamma^{(l+u)}$ as well as the reconstructions $\bar{\mathcal{X}}_{l+u}$.

The loss function of SuperCM combines a standard CE loss applied on the cluster responsibilities of the labeled data with the CM loss applied on both types of data. The SuperCM can also be used as a regularizer for existing SSL models. The combined loss can be stated as follows:

$$\begin{aligned}\mathcal{L}_{\text{SuperCM}}^{\text{SSL}} &= \mathbf{CE}(\Gamma^{(l)}, \mathcal{Y}^{(l)}) \\ &+ \beta \cdot \mathcal{L}_{\text{CM}}(\mathcal{X}^{(l+u)}, \Gamma^{(l+u)}, \bar{\mathcal{X}}^{(l+u)}) \\ &+ \delta \cdot \mathcal{L}_{\text{SSL}}\end{aligned} \tag{3}$$

where $\beta \geq 0$ and $\delta \geq 0$ are weights of the CM loss and of the loss of the SSL base model, respectively[1].

4. EXPERIMENTS

In this section, we compare the performance of the SuperCM as an SSL model, and as a regularizer for other SSL baselines. We follow the recommendations of [17] for data prepossessing, model architecture, and training protocol.

[1]Code available at https://github.com/Durgesh93/SuperCM.git

4.1. Experimental Setting

Data We use CIFAR-10 [18] for all the experiments. It consists of 60000, 32×32 color images distributed into ten classes. We use data augmentation random-crop, flip and Gaussian noise. The dataset is divided into training, validation, and test sets containing 50000, 5000, and 10000 images, respectively.
Architecture We use the Wide-ResNet-28-2 [19] architecture with 1.5M parameters as backbone for all the models. The architecture returns feature vectors before the linear classifier of dimension 128. When SuperCM is involved, the classifier is replaced with the CM's one-layer autoencoder.
Training In our experiments, the model is trained using the Adam [20] optimizer for 500000 iterations with batch size 100. The learning rate is decayed once with a factor of 0.1 after 400000 iterations. The hyper-parameters β and δ are tuned over the validation dataset.
Baseline There are two types of baselines for our experiments. For the SSL setting, the baseline is supervised-only training with the CE loss, i.e. $\delta = 0$ and $\beta = 0$ in Eq. (3). For the SSL regularization setting, we use VAT and Pseudo-label as base models, i.e. $\delta > 0$ and $\beta = 0$ in Eq. (3).
Evaluation The final model is computed based on stochastic weight averaging [21] for all the experiments. Model selection is based on the best Top-1 accuracy on the validation set. Overall model performance is measured using Top-1 accuracy on the test set. We report mean and standard deviation over five runs trained with different random seeds.

4.2. Results

Table 1 summarizes the results for the 600 and 4000 label settings on the CIFAR-10 dataset. Without the SSL base model, i.e., $\delta = 0$, SuperCM significantly improves the performance of the CE baseline by 5.2% and 3.6% for 600 and 4000 labels respectively in the SSL setting. When SuperCM is used as a regularizer of a base model, the accuracy of the base model significantly increases for the 600 labels setting. Specifically, SuperCM improves the accuracy of the Pseudo-Label baseline by 4.14% and that of VAT by 6.8%. However, we do not observe significant improvement for the 4000 labels setting. Our hypothesis for the results in the low supervision setting is that- the SSL base models benefit from the well-separated

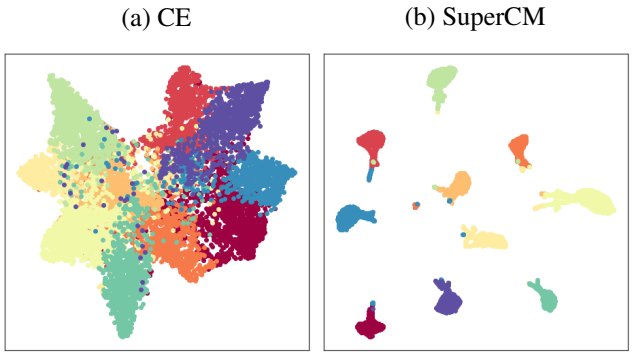

Fig. 3. UMAP plots for backbone features of the models trained with CE and SuperCM.

clustering obtained by SuperCM, when it is difficult to obtain a reliable supervisory signal with the CE loss in the training. A visualization of the well-separated features learned by SuperCM is provided in Sec. 5.1.

5. ANALYSIS

In this section, we analyze different aspects of the SuperCM and discuss the results.

5.1. Feature Visualization

We show in Figure 3, the UMAP[22] representations of the feature space learned by the models trained with CE and SuperCM for the CIFAR-10 600 labels configuration. It is evident that SuperCM yields more separated and compact classes, which leads to better generalization and thereby better performance.

5.2. Varying Data Amounts

We evaluate the performance of our method on CIFAR-10 for different training label sizes ranging from 250 to 4000 labels, with and without VAT as the SSL base model. The results are summarized in Figure 4.

Without the SSL base model, SuperCM improves the supervised-only baseline (CE) for all levels of supervision. However, we observe that improvements diminish as the number of labels drops significantly. In this case, we do not expect a large performance improvement from SuperCM alone, as the supervision is extremely scarce and not sufficient to learn a cluster-friendly embedding for the overarching SSL task. However, as the number of labels increases, we see significant improvements compared to the supervised-only baseline ranging from 3% to 6%.

Similarly, incorporating SuperCM as a regularizer improves the performance of VAT significantly for all except the 2000 and 4000 label settings. It is interesting to observe that SuperCM regularization improves the VAT baseline by around 10% in the 250 label setting, where VAT benefits from the online clustering regularization of the SuperCM. We also

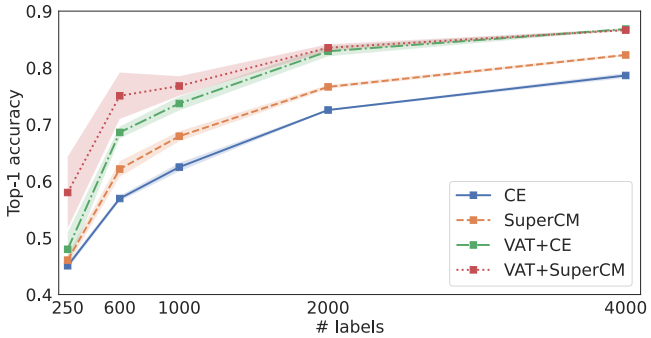

Fig. 4. Training with CE and SuperCM for different levels of supervision.

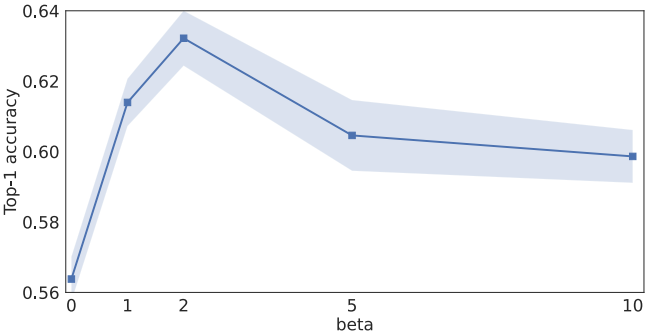

Fig. 5. Top-1 accuracy of SuperCM trained with 600 labels and different values of the hyper-parameter β.

see performance improvements of around 6% and 3% in the case of the 600 and 1000 label settings.

These results highlight the benefit of our method without relying on a complex training strategy, unlike other prominent approaches in the SSL research.

5.3. Hyper-parameter Sensitivity

Without a base model, the SuperCM uses a single hyper-parameter β. To study the sensitivity of the model toward β, we trained SuperCM with 600 labels and varying values of β on the CIFAR-10 dataset. Figure 5 shows the Top-1 accuracy of the model trained with different values of β, where $\beta = 0$ represents the training with only supervised loss (CE). As CM is incorporated ($\beta > 0$), we observe considerable performance improvement to existing CE baseline. However, we further observe that as the weight for the CM increases and less weight is given to the CE, performance slowly decreases.

6. CONCLUSION

In this paper, we present a simple end-to-end framework for SSL. Our training strategy benefits from the built-in clustering capability of the CM module and does not rely on complex training schemes. Facilitated by the differentiable CM, our method can be integrated into any gradient based SSL method as an unsupervised regularizer, paving the way to new versatile SSL approaches.

7. REFERENCES

[1] Martin J. Willemink, Wojciech A. Koszek, Cailin Hardell, Jie Wu, Dominik Fleischmann, Hugh Harvey, Les R. Folio, Ronald M. Summers, Daniel L. Rubin, and Matthew P. Lungren, "Preparing medical imaging data for machine learning," *Radiology*, vol. 295, no. 1, pp. 4–15, Apr. 2020.

[2] Olivier Chapelle, Bernhard Scholkopf, and Alexander Zien, Eds., *Semi-Supervised Learning*, The MIT Press, Sep. 2006.

[3] Xiangli Yang, Zixing Song, Irwin King, and Zenglin Xu, "A survey on deep semi-supervised learning," *Computing Research Repository (CoRR)*, vol. abs/2103.00550, 2021.

[4] Antti Tarvainen and Harri Valpola, "Mean teachers are better role models: Weight-averaged consistency targets improve semi-supervised deep learning results," in *Advances in Neural Information Processing Systems (NeurIPS)*, 2017, vol. 2017-December, pp. 1196–1205.

[5] Samuli Laine and Timo Aila, "Temporal ensembling for semi-supervised learning," *5th International Conference on Learning Representations (ICLR)*, 2017.

[6] Takeru Miyato, Shin ichi Maeda, Masanori Koyama, and Shin Ishii, "Virtual adversarial training: A regularization method for supervised and semi-supervised learning," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 41, pp. 1979–1993, 2019.

[7] Ben Athiwaratkun, Marc Finzi, Pavel Izmailov, and Andrew Gordon Wilson, "There are many consistent explanations of unlabeled data: Why you should average," in *International Conference on Learning Representations (ICLR)*, 2019.

[8] Yves Grandvalet and Yoshua Bengio, "Semi-supervised learning by entropy minimization," in *Advances in Neural Information Processing Systems (NeurIPS)*, 2005.

[9] Dong-Hyun Lee, "Pseudo-label : The simple and efficient semi-supervised learning method for deep neural networks," *ICML 2013 Workshop : Challenges in Representation Learning (WREPL)*, 07 2013.

[10] Hieu Pham, Qizhe Xie, Zihang Dai, and Quoc V. Le, "Meta pseudo labels," *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 11552–11563, 2021.

[11] Xiaohua Zhai, Avital Oliver, Alexander Kolesnikov, and Lucas Beyer, "S4l: Self-supervised semi-supervised learning," *IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 1476–1485, 2019.

[12] Sheng Zhou, Hongjia Xu, Zhuonan Zheng, Jiawei Chen, Zhao Li, Jiajun Bu, Jia Wu, Xin Wang, Wenwu Zhu, and Martin Ester, "A Comprehensive Survey on Deep Clustering: Taxonomy, Challenges, and Future Directions," Jun. 2022.

[13] Ahcène Boubekki, Michael C. Kampffmeyer, Ulf Brefeld, and Robert Jenssen, "Joint optimization of an autoencoder for clustering and embedding," *Machine Learning*, vol. 110, pp. 1901–1937, 2021.

[14] Junyuan Xie, Ross Girshick, and Ali Farhadi, "Unsupervised deep embedding for clustering analysis," in *Proceedings of the 33rd International Conference on International Conference on Machine Learning (ICML)*, 2016, vol. 48, pp. 478–487.

[15] Mathilde Caron, Piotr Bojanowski, Armand Joulin, and Matthijs Douze, "Deep clustering for unsupervised learning of visual features," in *European Conference on Computer Vision (ECCV)*, 2018.

[16] Junnan Li, Pan Zhou, Caiming Xiong, and Steven Hoi, "Prototypical contrastive learning of unsupervised representations," in *International Conference on Learning Representations (ICLR)*, 2021.

[17] Avital Oliver, Augustus Odena, Colin Raffel, Ekin D. Cubuk, and Ian J. Goodfellow, "Realistic evaluation of deep semi-supervised learning algorithms," *Advances in Neural Information Processing Systems (NeurIPS)*, vol. 2018-December, pp. 3235–3246, Apr. 2018.

[18] Alex Krizhevsky, "Learning multiple layers of features from tiny images," 2009.

[19] Sergey Zagoruyko and Nikos Komodakis, "Wide residual networks," in *Proceedings of the British Machine Vision Conference (BMVC)*. Sep. 2016, pp. 87.1–87.12, BMVA Press.

[20] Diederik P. Kingma and Jimmy Ba, "Adam: A method for stochastic optimization," in *3rd International Conference on Learning Representations (ICLR), San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, 2015.

[21] Pavel Izmailov, Dmitrii Podoprikhin, Timur Garipov, Dmitry P. Vetrov, and Andrew Gordon Wilson, "Averaging weights leads to wider optima and better generalization," in *Thirty-Fourth Conference on Uncertainty in Artificial Intelligence (UAI)*. 2018, pp. 876–885, AUAI Press.

[22] Leland McInnes and John Healy, "UMAP: uniform manifold approximation and projection for dimension reduction," *Computing Research Repository (CoRR)*, vol. abs/1802.03426, 2018.