Supplementary Material for Cross-feature Contrastive Learning

Sai Aparna Aketi
Purdue University
USA
saketi@purdue.edu

Kaushik Roy
Purdue University
USA
kaushik@purdue.edu

A. Experimental Setup Details

Figure. 1 illustrates a decentralized setup with 5 agents connected in a ring topology.

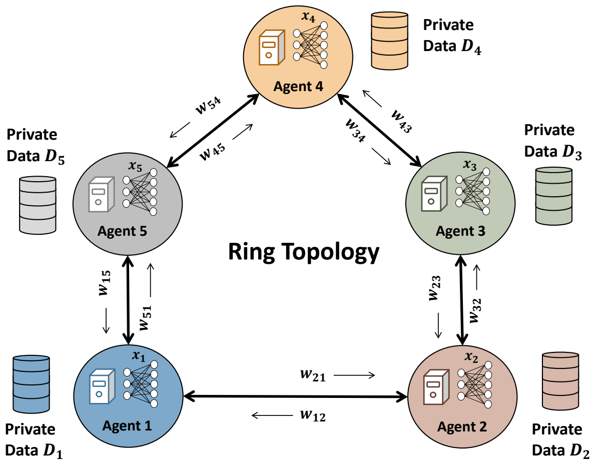


Figure 1. Decentralized training setup with 5 agents connected in a ring topology. Each agent has its own private dataset and a local model.

For the decentralized setup, we use an undirected ring, undirected Dyck graph, and undirected torus graph topologies with a uniform mixing matrix. The undirected ring topology for any graph size has 3 peers per agent including itself and each edge has a weight of $\frac{1}{3}$. The undirected Dyck topology with 32 agents has 4 peers per agent including itself and each edge has a weight of $\frac{1}{4}$. The undirected torus topology with 32 agents has 5 peers per agent including itself and each edge has a weight of $\frac{1}{5}$. All our experiments were conducted on a system with an NVIDIA A40 card with 4 GPUs. We report the test accuracy of the consensus model averaged over three randomly chosen seeds. The consensus model is obtained by averaging the model parameters across all agents using an all-reduce mechanism at the end of the training. The source code is available at https://github.com/aparna-aketi/Cross_feature_Contrastive_Loss

A.1. Datasets

In this section, we give a brief description of the datasets used in our experiments. We use a diverse set of datasets each originating from a different distribution of images to show the generalizability of the proposed techniques.

CIFAR-10: CIFAR-10 [3] is an image classification dataset with 10 classes. The image samples are colored (3 input channels) and have a resolution of 32×32. There are $50,000$ training samples with 5000 samples per class and $10,000$ test samples with 1000 samples per class.

CIFAR-100: CIFAR-100 [3] is an image classification dataset with 100 classes. The image samples are colored (3 input channels) and have a resolution of 32×32. There are $50,000$ training samples with 500 samples per class and $10,000$ test samples with 100 samples per class. CIFAR-100 classification is a harder task compared to CIFAR-10 as it has 100 classes with very few samples per class to learn from.

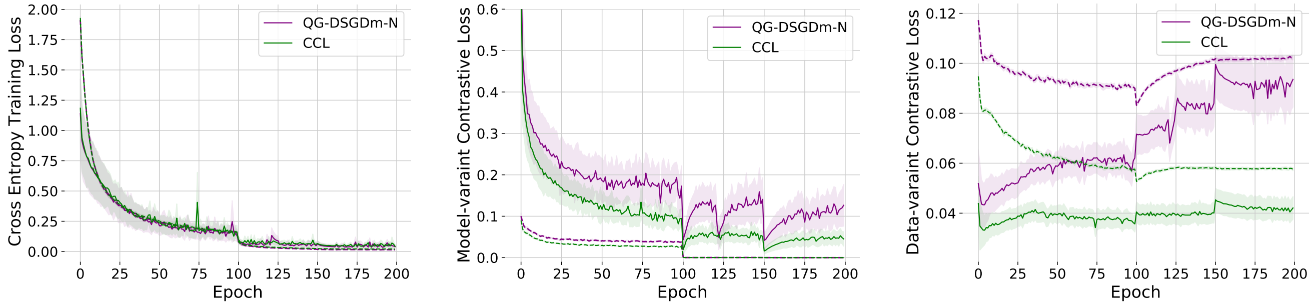
Fashion MNIST: Fashion MNIST [6] is an image classification dataset with 10 classes. The image samples are in greyscale (1 input channel) and have a resolution of 28×28. There are $60,000$ training samples with 6000 samples per class and $10,000$ test samples with 1000 samples per class.

Imagenette: Imagenette [2] is a 10-class subset of the ImageNet dataset. The image samples are colored (3 input channels) and have a resolution of 224×224. There are 9469 training samples with roughly 950 samples per class and 3925 test samples.

ImageNet: ImageNet dataset spans 1000 object classes and contains 1,281,167 training images, 50,000 validation images, and 100,000 test images. The image samples are colored (3 input channels) and have a resolution of 224×224.

A.2. Network Architecture

We replace ReLU+BatchNorm layers of all the model architectures with EvoNorm-S0 as it was shown to be better suited for decentralized learning over non-IID distributions.



(a) Cross-Entropy Training Loss ($\mathcal{L}_{ce}$)

(b) Model-Variant Contrastive Loss ($\mathcal{L}_{mv}$)

(c) Data-Variant Contrastive Loss ($\mathcal{L}_{dv}$)

Figure 2. Comparing various training loss terms for IID (dashed lines) and non-IID (solid lines) partitions of CIFAR-10 trained on ResNet-20 over a ring topology of 16 agents. We use $\alpha = 10$ for IID data and $\alpha = 0.01$ for non-IID data.

ResNet-20: For ResNet-20 [1], we use the standard architecture with $0.27M$ trainable parameters except that BatchNorm+ReLU layers are replaced by EvoNorm-S0.

ResNet-18: For ResNet-18 [1], we use the standard architecture with $11M$ trainable parameters except that BatchNorm+ReLU layers are replaced by EvoNorm-S0.

LeNet-5: For LeNet-5 [4], we use the standard architecture with $61,706$ trainable parameters.

MobileNet-V2: We use the the standard MobileNet-V2 [5] architecture used for CIFAR dataset with $2.3M$ parameters except that BatchNorm+ReLU layers are replaced by EvoNorm-S0.

A.3. Hyper-parameters

This section presents a detailed description of the hyper-parameters used in our experiments. All the experiments were run for three randomly chosen seeds. We decay the step size by 10x after 50% and 75% of the training, unless mentioned otherwise. For all the experiments, we have used a momentum of 0.9 with Nesterov, a weight decay of 0.0001, and a mini-batch size of 32 per agent.

Table 1. The value of λ_m, λ_v used for training CIFAR-10 with non-IID data using ResNet-20 architecture presented in Table 1

Agents (n)	Method	ResNet-20	
		$\alpha = 0.1$	$\alpha = 0.01$
16	CCL (ours)	$0.01, 0.0$	$0.01, 0.01$
32	CCL (ours)	$0.1, 0.1$	$0.1, 0.1$

Hyper-parameters for experiments in Table 1: All the experiments have the stopping criteria set to 200 epochs. The initial learning rate is set to 0.1. We decay the step size by $10\times$ in multiple steps at 100^{th} and 150^{th} epoch. Table 1 presents values of the scaling factor λ_m, λ_d used in the experiments.

Table 2. The value of λ_m, λ_v used for training various datasets with CCL (presented in Table 2).

Dataset	$\alpha = 0.1$	$\alpha = 0.01$
Fashion MNIST	$0.001, 0.001$	$0.01, 0.01$
CIFAR-100	$0.1, 0.1$	$0.1, 0.1$
Imagenette	$0.001, 0.001$	$1.0, 1.0$

Hyper-parameters for experiments in Table 2: All the experiments for CIFAR-100 and ImageNette have the stopping criteria set to 100 epochs and Fashion MNIST experiments have a stopping criteria of 50 epochs. The initial learning rate is set to 0.1 for CIFAR-100 and 0.01 for Fashion MNIST and Imagenette. Table 2 presents values of the scaling factor λ_m, λ_d used in the experiments.

Hyper-parameters for experiments in Table 3: All the experiments have the stopping criteria set to 200 epochs. The initial learning rate is set to 0.1. We decay the step size by $10\times$ in multiple steps at 100^{th} and 150^{th} epoch. Table 3 presents values of the scaling factor λ_m, λ_d used in the experiments. All the experiments on the Dyck and Torus graph use an averaging rate of 0.9 (instead of the default value of 1.0).

Table 3. The value of λ_m, λ_v used for training CIFAR-10 datasets with CCL on ResNet-20 over various graph topologies (presented in Table 3).

Graph	$\alpha = 0.1$	$\alpha = 0.01$
Dyck (32 agents)	$0.1, 0.1$	$0.1, 0.1$
Torus (32 agents)	$0.1, 0.1$	$0.1, 0.1$

B. Additional Results

Figure. 2 measures the different training losses for both IID and non-IID distribution with $\alpha = 0.01$ of the CIFAR-10 dataset trained on ResNet-20 architecture. We observe that the training cross-entropy loss (Fig. 2a) for IID and non-IID data converges to zero even though there is a huge gap in the validation loss. However, Fig. 2b shows that the model-variation contrastive loss for the baseline is much higher in non-IID settings compared to IID and hence is a good measure of data-heterogeneity. On the other hand, data-variant contrastive loss measures the variation in class representations across agents. Fig. 2c shows that this variation is relatively stable throughout the training process for QG-DSGDm-N (baseline) with IID Data. However, for QG-DSGDm-N with the non-IID setting, a significant increase in the variation of class representations across agents is evident. The proposed *CCL* framework explicitly minimizes the model-variant and data-variant contrastive loss. Fig. 2b shows that the CCL helps in reducing the model variance compared to QG-DSGDm-N. Fig. 2c shows that CCL has a stable variation in class representations across agents compared to QG-DSGDm-N. This results in better performance of the proposed *Cross-feature Contrastive Loss* for decentralized learning on heterogeneous data. Further, we evaluate the proposed *CCL* on the varying depth of ResNet architecture with ring topology of 16 agents as shown in Figure. 3. We observe that the proposed *CCL* framework consistently outperforms the QG-DSGDm-N baseline over varying graph sizes by an average improvement of 2.68%.

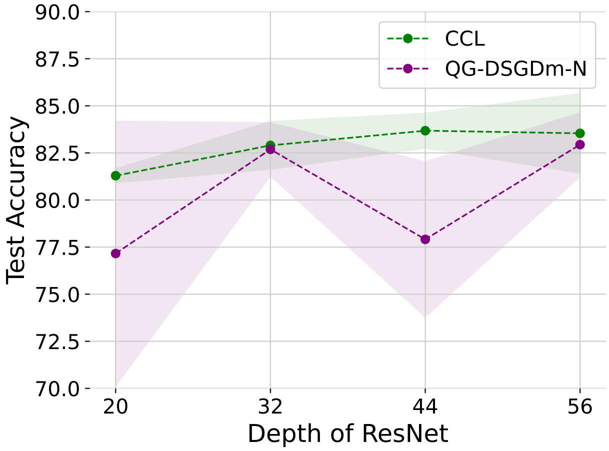


Figure 3. Test accuracy for the CIFAR-10 dataset trained on ResNet architecture with varying depth over 16-agent ring topology with a skew of $\alpha = 0.01$.

References

[1] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016. 2

[2] Hamel Husain. Imagenette - a subset of 10 easily classified classes from the imagenet dataset. *https://github.com/fastai/imagenette*, 2018. 1

[3] Alex Krizhevsky, Vinod Nair, and Geoffrey Hinton. Cifar (canadian institute for advanced research). *http://www.cs.toronto.edu/ kriz/cifar.html*, 2014. 1

[4] Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998. 2

[5] Mark Sandler, Andrew Howard, Menglong Zhu, Andrey Zhmoginov, and Liang-Chieh Chen. Mobilenetv2: Inverted residuals and linear bottlenecks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4510–4520, 2018. 2

[6] Han Xiao, Kashif Rasul, and Roland Vollgraf. Fashion-mnist: a novel image dataset for benchmarking machine learning algorithms. *arXiv preprint arXiv:1708.07747*, 2017. 1