Deep Relational Metric Learning Supplementary Material

Wenzhao Zheng*, Borui Zhang*, Jiwen Lu†, Jie Zhou
Department of Automation, Tsinghua University, China
Beijing National Research Center for Information Science and Technology, China
{zhengwz18, zhang-br21}@mails.tsinghua.edu.cn; {lujiwen, jzhou}@tsinghua.edu.cn

A. Results using the evaluation protocol [8]

Though we followed the standard evaluation protocol [12, 13, 22] and used a constrained experimental setting for fair comparisons with existing deep metric learning methods, the conclusions can still be questioned due to the lack of a validation set and the uninformative evaluation metric [8]. To improve the credibility of our experimental evaluation, we additionally performed experiments on the CUB-200-2011 [15] and Cars196 [6] dataset by strictly following the new evaluation protocol [8].

Specifically, we employed a BN-Inception [4] network pretrained on ImageNet [10] as the trunk model. We set the dimension of the final embedding to 128 and use a batch size of 32 for training. To prevent direct test set feedback, we performed a 4-fold cross-validation on the training subset to search for the hyperparameters. We used the first half of the classes as the training subset and the rest as the test subset, and then evenly split the training subset into four partitions based on the number of classes. During each validation, we employed one of the four partitions for training the rest for evaluation. We used the average accuracy on the four validation sets as feedback to tune the hyperparameters.

For testing, we reported the performance in separated and concatenated setting. For the separated setting, we directly computed the performance of the four 128-dim embeddings obtained using the model trained in each fold and reported the average results. For the concatenated setting, we concatenated the four aforementioned embedding for each sample to obtain a 512-dim embeddings for evaluation. We employed the Precision@1 (R/P@1), the R-Precision (RP), and the Mean Average Precision at R (MAP@R) as the evaluation metric. We direct interesting readers to the original paper [8] for more details.

Table 1 and 2 shows the results of the baseline methods and the proposed DRML framework on the CUB-200-2011 and Cars196 dataset, respectively. We use red numbers to denote the best results and blue numbers to denote the second best results. We applied our framework to the triplet loss [18], the ProxyAnchor loss [5], and Cosface [16]. We see that our DRML framework still consistently boosts the performance of existing methods and further achieves the state-of-the-art result under the new evaluation protocol, which verifies the effectiveness of the proposed relation-aware embedding.

B. Performance on large-scale datasets

We further conducted experiments on the ImageNet dataset [10] to evaluate the generalization of the proposed method to large-scale datasets. Table 3 shows the results of our DRML framework applied existing deep metric learning methods. As the original papers did not reported the performance on the ImageNet dataset, the results in Table 3 are based on our reproduction[1]. We observe that the ProxyAnchor loss with random sampling (PA) [5] is the best baseline method. The triplet loss with semi-hard sampling (i.e., TSH) [11] achieves better results than the softmax baseline while the margin loss with distance-weighted sampling (MDW) [19] achieves worse results, though MDW consistently outperforms TSH on small-scale datasets like the CUB-200-2011 and Cars196 datasets. We can see this trend on the middle-scale Stanford Online Products [13] dataset as the two methods achieve comparable performance. Despite the changing ranking of performance on datasets of different scales, our DRML framework can uniformly improve the performance of various methods, which shows that the effectiveness of our framework generalizes well to the benchmark scale.

C. Visualization of the embedding space

Figure 1 shows the qualitative result of the proposed DRML-MDW on the CUB-200-2011 dataset. We em-

*Equal contribution.
†Corresponding author.

[1]Code: https://github.com/zbr17/DRML

Table 1. Results using the new protocol on the CUB-200-2011 dataset.

Method	Concatenated (512-dim)			Separated (128-dim)		
	R/P@1	RP	MAP@R	R/P@1	RP	MAP@R
Pretrained	51.1	24.9	14.2	50.5	25.1	14.5
Contrastive [3]	67.2 ± 0.5	36.9 ± 0.3	26.2 ± 0.3	58.6 ± 0.5	31.5 ± 0.2	20.7 ± 0.2
ProxyNCA [7]	66.1 ± 0.3	35.5 ± 0.2	24.6 ± 0.2	58.3 ± 0.3	30.6 ± 0.1	19.7 ± 0.1
Margin [19]	65.5 ± 0.5	35.0 ± 0.2	24.1 ± 0.3	56.2 ± 0.4	29.5 ± 0.2	18.6 ± 0.2
N. Softmax [21]	65.4 ± 0.2	36.0 ± 0.2	25.2 ± 0.2	58.5 ± 0.2	31.7 ± 0.2	20.9 ± 0.2
ArcFace [2]	67.1 ± 0.3	37.2 ± 0.2	26.4 ± 0.2	60.1 ± 0.2	32.3 ± 0.1	21.4 ± 0.1
FastAP [1]	63.6 ± 0.2	34.5 ± 0.2	23.7 ± 0.2	55.9 ± 0.3	29.8 ± 0.2	19.1 ± 0.2
SNR [20]	67.3 ± 0.5	36.9 ± 0.2	26.1 ± 0.2	58.8 ± 0.3	31.6 ± 0.2	20.8 ± 0.2
MS [17]	66.0 ± 0.2	35.9 ± 0.1	25.2 ± 0.1	58.5 ± 0.2	31.4 ± 0.1	20.6 ± 0.1
MS+Miner [17]	65.8 ± 0.3	36.0 ± 0.2	25.2 ± 0.2	58.2 ± 0.2	31.3 ± 0.2	20.5 ± 0.2
SoftTriple [9]	66.2 ± 0.4	36.5 ± 0.2	25.6 ± 0.2	59.6 ± 0.4	32.1 ± 0.2	21.3 ± 0.2
Triplet [18]	**64.4** ± 0.4	34.6 ± 0.4	23.8 ± 0.4	56.0 ± 0.3	29.6 ± 0.3	18.8 ± 0.3
DRML-Triplet	64.2 ± 0.5	**34.8** ± 0.4	**24.1** ± 0.3	**56.3** ± 0.4	**30.0** ± 0.5	**19.3** ± 0.4
ProxyAnchor [5]	65.2 ± 0.2	36.0 ± 0.2	25.3 ± 0.1	56.6 ± 0.1	30.5 ± 0.1	19.8 ± 0.2
DRML-PA	**66.5** ± 0.1	**36.8** ± 0.2	**26.0** ± 0.2	**59.5** ± 0.2	**32.0** ± 0.3	**21.2** ± 0.2
Cosface [16]	67.2 ± 0.4	37.4 ± 0.2	26.5 ± 0.2	59.8 ± 0.3	32.1 ± 0.2	21.6 ± 0.2
DRML-Cosface	**69.2** ± 0.3	**37.8** ± 0.2	**27.2** ± 0.2	**60.2** ± 0.3	**33.0** ± 0.2	**22.3** ± 0.3

Table 2. Results using the new protocol on the Cars196 dataset.

Method	Concatenated (512-dim)			Separated (128-dim)		
	R/P@1	RP	MAP@R	R/P@1	RP	MAP@R
Pretrained	46.9	13.8	5.9	43.3	13.4	5.6
Contrastive [3]	81.6 ± 0.4	35.7 ± 0.4	25.5 ± 0.4	69.4 ± 0.2	28.2 ± 0.2	17.6 ± 0.2
ProxyNCA [7]	83.3 ± 0.4	36.6 ± 0.3	26.4 ± 0.4	70.9 ± 0.6	28.6 ± 0.3	18.0 ± 0.3
Margin [19]	82.1 ± 2.4	34.7 ± 2.2	24.1 ± 2.3	71.0 ± 2.7	27.6 ± 1.5	16.8 ± 1.5
N. Softmax [21]	83.6 ± 0.3	36.6 ± 0.2	26.4 ± 0.2	72.9 ± 0.2	29.6 ± 0.1	18.9 ± 0.1
ArcFace [2]	84.0 ± 0.2	35.4 ± 0.3	25.2 ± 0.3	73.7 ± 0.4	28.6 ± 0.1	18.1 ± 0.1
FastAP [1]	78.2 ± 0.7	33.4 ± 0.7	22.9 ± 0.7	64.7 ± 0.6	26.4 ± 0.4	15.8 ± 0.4
SNR [20]	81.9 ± 0.4	35.4 ± 0.4	25.1 ± 0.5	70.2 ± 0.4	27.9 ± 0.4	17.4 ± 0.3
MS [17]	85.3 ± 0.3	38.0 ± 0.6	27.8 ± 0.8	73.7 ± 1.0	29.4 ± 0.6	18.8 ± 0.7
MS+Miner [17]	84.6 ± 0.3	37.7 ± 0.4	27.6 ± 0.4	72.9 ± 0.3	29.5 ± 0.4	18.9 ± 0.4
SoftTriple [9]	83.7 ± 0.2	36.3 ± 0.2	26.1 ± 0.2	73.0 ± 0.2	29.4 ± 0.1	18.7 ± 0.1
Triplet [18]	77.5 ± 0.6	32.9 ± 0.5	22.1 ± 0.5	63.9 ± 0.4	26.1 ± 0.3	15.2 ± 0.3
DRML-Triplet	**78.8** ± 0.3	**33.2** ± 0.4	**22.8** ± 0.5	**64.0** ± 0.2	**26.2** ± 0.3	**15.5** ± 0.1
ProxyAnchor [5]	83.3 ± 0.4	35.7 ± 0.3	25.7 ± 0.4	73.7 ± 0.4	29.4 ± 0.3	18.9 ± 0.2
DRML-PA	**85.7** ± 0.5	**36.0** ± 0.2	**26.1** ± 0.2	**76.6** ± 0.4	**29.8** ± 0.3	**19.3** ± 0.2
Cosface [16]	85.3 ± 0.2	36.7 ± 0.2	26.9 ± 0.2	74.1 ± 0.2	28.5 ± 0.1	18.2 ± 0.1
DRML-Cosface	**86.4** ± 0.3	**38.7** ± 0.4	**29.2** ± 0.3	**75.7** ± 0.3	**30.2** ± 0.2	**20.0** ± 0.1

ployed the Barnes-Hut t-SNE [14] algorithm to visualize the learned embedding space and magnify specific regions for clear demonstration. We color the boundary of each image using different colors to represent the ground truth class label. We observe that even though the classes in the test subset are not seen during training, our method can still accurately measure their semantic differences. Moreover, the images in the CUB-200-2011 dataset possess small interclass differences and large intraclass variations, yet our framework still effectively clusters together instances from the same class using the learned relation-aware embeddings despite all these difficulties.

Table 3. Experimental results on the ImageNet dataset.

Method	R/P@1	R@2	P@2	RP	MAP@R	NMI
Softmax Baseline	53.7	63.8	50.9	25.5	33.9	71.8
Margin-DW [19]	46.3	56.5	45.5	23.7	33.1	74.6
DRML-MDW	**48.9**	**59.2**	**48.3**	**25.1**	**34.4**	**75.4**
Triplet-SH* [11]	54.9	64.5	55.2	32.0	41.3	78.3
DRML-TSH	**55.8**	**65.3**	**55.3**	**32.3**	**41.5**	**78.6**
ProxyAnchor [5]	66.4	74.1	66.4	44.2	52.2	82.2
DRML-PA	**68.0**	**75.0**	**67.6**	**46.3**	**53.9**	**82.9**

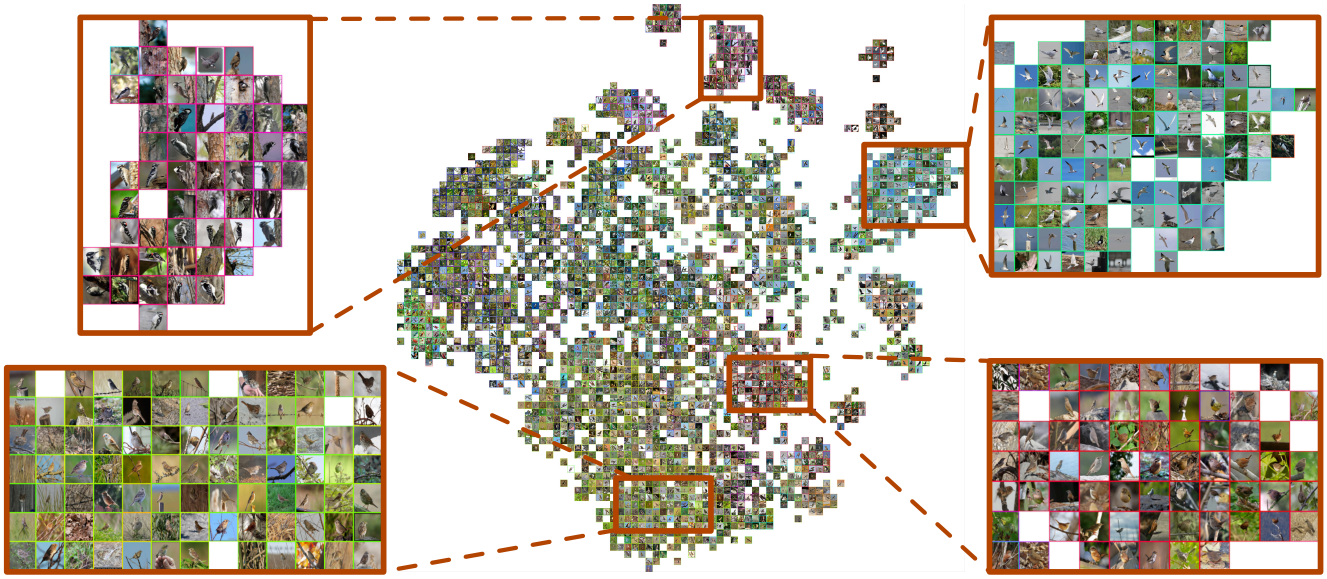

Figure 1. Qualitative result of the proposed DRML-MDW method on the test subset of the CUB-200-2011 dataset, where we magnify specific regions for clear demonstration. (Best viewed on a monitor when zoomed in.)

References

[1] Fatih Cakir, Kun He, Xide Xia, Brian Kulis, and Stan Sclaroff. Deep metric learning to rank. In *CVPR*, pages 1861–1870, 2019.

[2] Jiankang Deng, Jia Guo, Niannan Xue, and Stefanos Zafeiriou. Arcface: Additive angular margin loss for deep face recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4690–4699, 2019.

[3] Raia Hadsell, Sumit Chopra, and Yann LeCun. Dimensionality reduction by learning an invariant mapping. In *CVPR*, pages 1735–1742, 2006.

[4] Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *ICML*, pages 448–456, 2015.

[5] Sungyeon Kim, Dongwon Kim, Minsu Cho, and Suha Kwak. Proxy anchor loss for deep metric learning. In *CVPR*, pages 3238–3247, 2020.

[6] Jonathan Krause, Michael Stark, Jia Deng, and Li Fei-Fei. 3d object representations for fine-grained categorization. In *ICCVW*, pages 554–561, 2013.

[7] Yair Movshovitz-Attias, Alexander Toshev, Thomas K. Leung, Sergey Ioffe, and Saurabh Singh. No fuss distance metric learning using proxies. In *ICCV*, pages 360–368, 2017.

[8] Kevin Musgrave, Serge Belongie, and Ser-Nam Lim. A metric learning reality check. In *ECCV*, 2020.

[9] Qi Qian, Lei Shang, Baigui Sun, and Juhua Hu. Softtriple loss: Deep metric learning without triplet sampling. In *ICCV*, 2019.

[10] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, et al. Imagenet large scale visual recognition challenge. *IJCV*, 115(3):211–252, 2015.

[11] Florian Schroff, Dmitry Kalenichenko, and James Philbin. Facenet: A unified embedding for face recognition and clustering. In *CVPR*, pages 815–823, 2015.

[12] Kihyuk Sohn. Improved deep metric learning with multi-class n-pair loss objective. In *NIPS*, pages 1857–1865, 2016.

[13] Hyun Oh Song, Yu Xiang, Stefanie Jegelka, and Silvio Savarese. Deep metric learning via lifted structured feature embedding. In *CVPR*, pages 4004–4012, 2016.

[14] Laurens Van Der Maaten. Accelerating t-sne using tree-

based algorithms. *JMLR*, 15(1):3221–3245, 2014.

[15] Catherine Wah, Steve Branson, Peter Welinder, Pietro Perona, and Serge J Belongie. The Caltech-UCSD Birds-200-2011 dataset. Technical Report CNS-TR-2011-001, California Institute of Technology, 2011.

[16] Hao Wang, Yitong Wang, Zheng Zhou, Xing Ji, Dihong Gong, Jingchao Zhou, Zhifeng Li, and Wei Liu. Cosface: Large margin cosine loss for deep face recognition. In *CVPR*, pages 5265–5274, 2018.

[17] Xun Wang, Xintong Han, Weilin Huang, Dengke Dong, and Matthew R Scott. Multi-similarity loss with general pair weighting for deep metric learning. In *CVPR*, pages 5022–5030, 2019.

[18] Kilian Q Weinberger and Lawrence K Saul. Distance metric learning for large margin nearest neighbor classification. *JMLR*, 10(2):207–244, 2009.

[19] Chao-Yuan Wu, R Manmatha, Alexander J Smola, and Philipp Krähenbühl. Sampling matters in deep embedding learning. In *ICCV*, pages 2859–2867, 2017.

[20] Tongtong Yuan, Weihong Deng, Jian Tang, Yinan Tang, and Binghui Chen. Signal-to-noise ratio: A robust distance metric for deep metric learning. In *CVPR*, pages 4815–4824, 2019.

[21] Andrew Zhai and Hao-Yu Wu. Classification is a strong baseline for deep metric learning. *arXiv preprint arXiv:1811.12649*, 2018.

[22] Wenzhao Zheng, Zhaodong Chen, Jiwen Lu, and Jie Zhou. Hardness-aware deep metric learning. In *CVPR*, pages 72–81, 2019.