Addressing the Loss-Metric Mismatch with Adaptive Loss Alignment Supplementary Material

Chen Huang [1] **Shuangfei Zhai** [1] **Walter Talbott** [1] **Miguel Angel Bautista** [1] **Shih-Yu Sun** [1] **Carlos Guestrin** [1] **Josh Susskind** [1]

1. More Experiments

Image classification on CIFAR-100: We train and evaluate ALA for the metric of classification error on the CIFAR-100 dataset (Krizhevsky, 2009). As in CIFAR-10, we divide the training set of CIFAR-100 randomly into a new training set of 40k images and a validation set of 10k images, for loss controller learning. The 10k testing images are used for evaluation. We compare with the recent methods that use the full 50k training images and their optimal hyper-parameters. For ALA, multi-network training is adopted by default for robust online policy learning. Each network is trained via Momentum-SGD.

Table S1 reports classification errors using different ResNet (He et al., 2016) architectures. For all network architectures, ALA outperforms both hand-designed loss functions, *e.g.*, L-Softmax (Liu et al., 2016), and the adaptive loss function that acts as a differentiable metric surrogate in L2T-DLF (Wu et al., 2018). This validates the benefits of directly optimizing the evaluation metric using ALA.

Face verification on LFW: We evaluate the performance of our ALA-based metric learning method on a face verification task using the LFW dataset (Huang et al., 2007). The LFW verification benchmark contains 6,000 verification pairs. For a fair comparison with recent approaches we train ALA using the same 64-layer ResNet architecture proposed in (Liu et al., 2017; Wang et al., 2018) as our main model. We follow the small training data protocol (Huang et al., 2007) and train and validate on the popular CASIA-WebFace dataset (Yi et al., 2014) which contains 494,414 images of 10,575 people. The training images with identities appearing in the test set are removed. Our ALA controller is trained to optimize the verification accuracy metric on the validation set.

[1]Apple Inc., Cupertino, United States. Correspondence to: Chen Huang <chen-huang@apple.com>.

Proceedings of the 36^{th} International Conference on Machine Learning, Long Beach, California, PMLR 97, 2019.

Table S1. Classification error (%) on CIFAR-100 dataset. 10-run average and standard deviation are reported for ALA.

Method	ResNet-8	ResNet-20	ResNet-32
cross-entropy	39.79	32.33	30.38
L-Softmax (Liu et al., 2016)	38.93	31.65	29.56
L2T-DLF (Wu et al., 2018)	38.27	30.97	29.25
ALA	**37.78±0.09**	**30.54±0.07**	**29.06±0.09**

Table S2. Face verification accuracy (%) on LFW dataset. All methods use the same training data and network architectures.

Method	Accuracy
Softmax loss	97.88
Softmax+Contrastive (Sun et al., 2014)	98.78
Triplet loss (Schroff et al., 2015)	98.70
L-Softmax loss (Liu et al., 2016)	99.10
Softmax+Center loss (Wen et al., 2016)	99.05
SphereFace (A-Softmax) (Liu et al., 2017)	99.42
CosFace (LMCL) (Wang et al., 2018)	99.33
Triplet + ALA (Focal weighting)	99.49
Triplet + ALA (Distance mixture)	**99.57**

Table S2 compares ALA to recent face recognition methods on LFW. These methods often adopt a strong but hand-designed loss function to improve class discrimination. In contrast, ALA adaptively controls the triplet loss function (Schroff et al., 2015), achieving state-of-the-art performance even for different parameterizations, where we specifically studied focal weighting (Lin et al., 2017) and distance mixture formulations. These results further verify the advantages of ALA to directly optimize for the target metric regardless of the specific formulation of loss function to be controlled.

2. More Analyses

Baseline comparisons: Table S3 compares some related baselines in both classification and metric learning tasks to further highlight the benefits of ALA. In particular, we compare with the contextual bandit method and population-based training (PBT) (Jaderberg et al., 2017). The two baselines follow the same experimental settings on respective datasets, as detailed in the main paper.

The contextual bandit method changes loss parameters (*i.e.*, actions) according to the current training states, similar to an online hyperparameter search scheme. Following the same loss parameterizations for classification and metric learning, the method increases weights for those confusing class pairs and evaluation metric-improving distance functions respectively, and otherwise downweights them. This is similar to our one-step RL setting except that in ALA, actions affect future states, making it an RL problem. Table S3 illustrates that one-step RL-based ALA consistently outperforms the heuristic contextual bandit method. We believe more advanced bandit algorithms can work better, but RL has the capacity to learn flexible state-transition dynamics. Moreover, our RL setting can be extended to use multi-step episodes (Figure S1). This allows to model longer-term effects of actions, while contextual bandits always obtain immediate reward from a single action.

Recall that we train in parallel 10 child models by default, for robust ALA policy learning. We are thus interested to see how this compares to PBT techniques (using the same 10 child models). Table S3 shows that PBT does not help as much as ALA, which suggests the learned ALA policy is more powerful than model ensembling or parameter tuning. We will show later (in Figure S1) that ALA can achieve competitive performance even with 1 child model which enjoys higher learning efficiency.

Ablation study: Table S4 shows the results of ablation studies on the design choices of *ALA loss controller* and *state representation*. As in Table S3, we experiment with the example tasks of classification and metric learning under the same settings. Looking at the top cell of Table S4, we find that switching from 2-layer loss controller to 1-layer leads to a consistent performance drop; on the other hand, the 3-layer loss controller does not help much. The bottom cell of Table S4 quantifies the effects of the four components of our policy state s_t. We can see that it is relatively more important to keep the historical sequence of validation statistics (besides the ones at current timestep) and the current loss parameters Φ_t in the state representation. The relative change of validation statistics (from their moving average) and the normalized iteration number also have marginal contributions.

Computational cost: Under the classification and metric learning tasks considered in the paper, our simultaneous (single) model training and ALA policy learning often incur an extra $20\% - 50\%$ cost as indexed by wall-clock time over regular model training. However, this overhead is often canceled out by the convergence speedup of the main model. Our multi-model training together with policy learning is able to achieve stronger performance with modest additional ($\sim 30\%$) computational overhead for policy learning, at the cost of using distributed training to collect replay episodes.

Table S3. Baseline comparisons for CIFAR-10 classification and metric learning on Stanford Online Products (SOP) dataset. We report classification error (%) with ResNet-32 and Recall(%)@k=1 for the two tasks respectively. For metric learning, ALA is trained with the 'Margin' framework and with the loss parameterization of 'Distance mixture'. We compare ALA to contextual bandit and population-based training (PBT) baselines.

Classification		Metric learning	
Method	Error↓	Method	Recall↑
cross-entropy	7.51	Triplet (Schroff et al., 2015)	66.7
L2T (Fan et al., 2018)	7.10	Margin (Wu et al., 2017)	72.7
L2T-DLF (Wu et al., 2018)	6.95	ABE-8 (Kim et al., 2018)	76.3
ALA	**6.79**	Margin + ALA	**78.9**
Contextual bandit	7.34	Contextual bandit	73.1
PBT (Jaderberg et al., 2017)	7.29	PBT (Jaderberg et al., 2017)	73.6

Table S4. Ablation studies of ALA loss controller design (2-layer MLP by default) and state representation s_t. Experiments of CIFAR-10 classification and metric learning on SOP are conducted with the same settings as in Table S3. Performance degradation in comparison to default ALA method is indicated by positive Δ of classification error (%) and negative Δ of Recall(%)@k=1.

Classification		Metric learning	
Method	ΔError	Method	ΔRecall
ALA (1-layer MLP)	+0.06	Margin+ALA (1-layer MLP)	-0.5
ALA (3-layer MLP)	-0.03	Margin+ALA (3-layer MLP)	-0.1
ALA (s_t w/o history)	+0.11	Margin+ALA (s_t w/o history)	-1.4
ALA (s_t w/o Δstatistics)	+0.04	Margin+ALA (s_t w/o Δstatistics)	-0.2
ALA (s_t w/o Φ_t)	+0.05	Margin+ALA (s_t w/o Φ_t)	-0.6
ALA (s_t w/o iter#)	+0.02	Margin+ALA (s_t w/o iter#)	-0.3

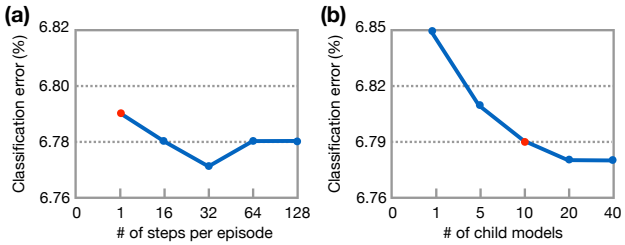

Figure S1. Sample efficiency of our RL approach for ALA (validation metric as reward). Classification error of ResNet-32 is reported on CIFAR-10. Good performance can be achieved by our default RL settings (red dots) with one-step episodes and 10 child model training that are sample efficient.

This is much more efficient than those meta-learning methods, *e.g.*, (Fan et al., 2018; Zoph & Le, 2017) that learn the policy by training the main model to convergence multiple times (*e.g.*, 50 times).

Sample efficiency: Figure S1 illustrates the sample efficiency of ALA's RL approach in the example task of CIFAR-10 classification. We train the ResNet-32 model and use the default reward based on the validation metric. Figure S1(a) shows that using episodes consisting of a single training step suffices to learn competent loss policies with good per-

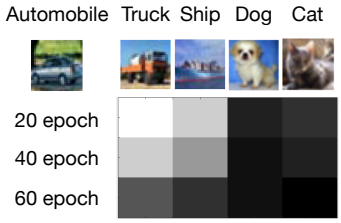

Figure S2. Evolution of class correlation scores $\Phi_t(i,j)$ on CIFAR-10 (with ResNet-32 network). Light/dark color denotes positive/negative values. Our policy modifies the class correlation scores in a way that forms a hierarchical classification curriculum by merging similar classes and gradually separating them.

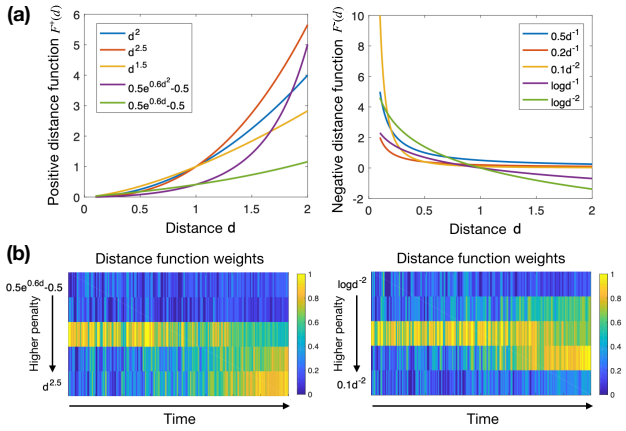

Figure S3. (a) Our positive distance function $F_i^+(\cdot)$ and negative distance function $F_i^-(\cdot)$ in metric learning. (b) Evolution of distance function weights $\Phi_t(i)$ on Stanford Online Products dataset. Our policy gives gradually larger weights to those high-penalty distance functions, which implies an adaptive and soft "hard-mining" curriculum.

formance. Figure S1(b) further shows improvements from parallel training with multiple child models that provide more episodes for policy learning. We empirically choose to use 10 child models, which only incurs an extra $\sim 30\%$ time cost for policy learning, thus striking a good performance tradeoff.

Policy visualization for classification: Figure S2 illustrates the ALA policy learned for classification, which performs actions to adjust the loss parameters in Φ_t (*i.e.*, class correlations) dynamically. We observe that the ALA controller tends to first merge similar classes with positive $\Phi_t(i,j)$, and then gradually discriminates between them with negative $\Phi_t(i,j)$. This indicates a learned curriculum that guides model learning to achieve both better optimization and generalization.

Policy visualization for metric learning: We visualize the learned ALA policy for metric learning under a parametric loss formulation that mixes different distance functions. Figure S3(a) first shows the distance functions $F_i^+(\cdot)$ and $F_i^-(\cdot)$ we apply to distance d^+ (between anchor and positive instances) and distance d^- (between anchor and negative instances), respectively. Specifically, $F_i^+(d) \in \{d^2, d^{2.5}, d^{1.5}, 0.5e^{0.6d^2} - 0.5, 0.5e^{0.6d} - 0.5\}$ defines 5 increasing distance functions to penalize large d^+, and $F_i^-(d) \in \{0.5d^{-1}, 0.2d^{-1}, 0.1d^{-2}, \log d^{-1}, \log d^{-2}\}$ defines 5 decreasing distance functions to penalize small d^-. We empirically found our performance is relatively robust to the design choices of distance functions (within $\pm 0.05\%$ verfication accuracy on LFW among our early trials), as long as they differ. The ability to learn adaptive weightings over these distance functions plays a more important role.

Figure S3(b) demonstrates the evolution of weights $\Phi_t(i)$ over our distance functions on the Stanford Online Products dataset. Note that while the weights for our default distance functions d^2 and $0.5d^{-1}$ are both initialized as 1, our ALA controller learns to assign larger weights to those high-penalty distance functions over time. This implies an adaptive "hard mining" curriculum learned from data that is more flexible than hand-designed alternatives.

3. Limitations

In this work we studied multiple evaluation metric formulations (classification accuracy and AUCPR for the classification settings, and Recall@k and verification accuracy for metric learning). While this includes non-decomposable metrics, we did not extend to more complex scenarios that might reveal further benefits of ALA. In future work we plan to apply ALA to multiple simultaneous objectives, where the controller will need to weigh between these objectives dynamically. We would also like to examine cases where the output of a given model is an input into a more complex pipeline, which is common in production systems (*e.g.*, detection→alignment→recognition pipelines). This requires further machinery to be developed for making reward evaluation efficient enough to learn the policy jointly with training the different modules.

Another area where ALA can be further developed is to make it less dependent on specific task types and loss/metric formulations. Ideally, a controller can be trained through continual learning to handle different scenarios flexibly. This would enable the use of ALA in distributed crowd learning settings where model training gets better and better over time.

Finally, an interesting area to study further is how ALA behaves in dynamically changing environments where available training data can change over time (*e.g.*, life-long learning, online learning, meta-learning). Ideally, ALA is suited to tackle these challenges, and we will continue to explore this in future work.

References

Fan, Y., Tian, F., Qin, T., Li, X.-Y., and Liu, T.-Y. Learning to teach. In *International Conference on Learning Representations*, 2018.

He, K., Zhang, X., Ren, S., and Sun, J. Deep residual learning for image recognition. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

Huang, G. B., Ramesh, M., Berg, T., and Learned-Miller, E. Labeled faces in the wild: A database for studying face recognition in unconstrained environments. Technical report, University of Massachusetts, Amherst, 2007.

Jaderberg, M., Dalibard, V., Osindero, S., Czarnecki, W. M., Donahue, J., Razavi, A., Vinyals, O., Green, T., Dunning, I., Simonyan, K., Fernando, C., and Kavukcuoglu, K. Population based training of neural networks. *arXiv preprint arXiv:1711.09846*, 2017.

Kim, W., Goyal, B., Chawla, K., Lee, J., and Kwon, K. Attention-based ensemble for deep metric learning. In *The European Conference on Computer Vision (ECCV)*, pp. 760–777, 2018.

Krizhevsky, A. Learning multiple layers of features from tiny images. Technical report, University of Toronto, 2009.

Lin, T.-Y., Goyal, P., Girshick, R., He, K., and Dollár, P. Focal Loss for Dense Object Detection. In *International Conference on Computer Vision (ICCV)*, 2017.

Liu, W., Wen, Y., Yu, Z., and Yang, M. Large-margin softmax loss for convolutional neural networks. In *Proceedings of The 33rd International Conference on Machine Learning*, pp. 507–516, 2016.

Liu, W., Wen, Y., Yu, Z., Li, M., Raj, B., and Song, L. Sphereface: Deep hypersphere embedding for face recognition. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

Schroff, F., Kalenichenko, D., and Philbin, J. Facenet: A unified embedding for face recognition and clustering. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 815–823, 2015.

Sun, Y., Chen, Y., Wang, X., and Tang, X. Deep learning face representation by joint identification-verification. In *Advances in Neural Information Processing Systems*, 2014.

Wang, H., Wang, Y., Zhou, Z., Ji, X., Li, Z., Gong, D., Zhou, J., and Liu, W. CosFace: Large margin Cosine loss for deep face recognition. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.

Wen, Y., Zhang, K., Li, Z., and Qiao, Y. A discriminative feature learning approach for deep face recognition. In *The European Conference on Computer Vision (ECCV)*, 2016.

Wu, C.-Y., Manmatha, R., Smola, A. J., and Krähenbühl, P. Sampling matters in deep embedding learning. In *International Conference on Computer Vision*, 2017.

Wu, L., Tian, F., Xia, Y., Fan, Y., Qin, T., Jian-Huang, L., and Liu, T.-Y. Learning to teach with dynamic loss functions. In *Advances in Neural Information Processing Systems 31*, pp. 6467–6478, 2018.

Yi, D., Lei, Z., Liao, S., and Li, S. Z. Learning face representation from scratch. *CoRR*, abs/1411.7923, 2014.

Zoph, B. and Le, Q. V. Neural architecture search with reinforcement learning. In *International Conference on Learning Representations*, 2017.