Appendix for "Deep Combinatorial Aggregation"

Yuesong Shen [1,2] **Daniel Cremers** [1,2]
[1] Technical University of Munich, Germany
[2] Munich Center for Machine Learning, Germany
{yuesong.shen, cremers}@tum.de

A Pseudo-code for DCA training

Algorithm 1 depicts the pseudocode for training a layerwise DCA model.

Algorithm 1: Layerwise DCA training

Inputs : Dataset $\mathcal{D}$, neural network f with n sets of parameters $\boldsymbol{\Theta} = (\theta^1_{1:n}, \dots, \theta^L_{1:n})$, with θ^l_i the i-th parameter instance of layer l.

def *sampleDCAProposal(*$\boldsymbol{\Theta}$*)*:
  Draw random indices $i_1, \dots, i_L \in [1:n]$;
  return $\hat{\boldsymbol{\theta}} = (\theta^1_{i_1}, \dots, \theta^L_{i_L})$;
repeat
  foreach *minibatch* $B \in \mathcal{D}$ **do**
    $\hat{\boldsymbol{\theta}} \leftarrow$ sampleDCAProposal($\boldsymbol{\Theta}$);
    loss $\leftarrow$ feedForward($f, \hat{\boldsymbol{\theta}}, B$);
    $\Delta\hat{\boldsymbol{\theta}} \leftarrow$ loss.backward();
    $\boldsymbol{\Theta} \leftarrow$ updateDCAParameters($\boldsymbol{\Theta}, \Delta\hat{\boldsymbol{\theta}}$);
  end
until *end of training*;

B Additional experimental results

B.1 In-domain results on SVHN dataset

Table 1 summarizes the results of in-domain image classification on SVHN dataset.

Table 1: In-domain image classification results on SVHN. Similar to the case of CIFAR-10, modelwise DCA has the best predictive performance and uncertainty estimation, while trunkwise DCA has slightly better results w.r.t. deep ensemble while producing a richer set of model proposals.

	Accuracy ↑	NLL ↓	ECE ↓	Brier ↓
Standard	0.9590 ± 0.0012	0.1786 ± 0.0038	0.0176 ± 0.0011	0.0654 ± 0.0016
MC dropout	0.9644 ± 0.0005	0.1353 ± 0.0017	0.0053 ± 0.0006	0.0547 ± 0.0006
SWAG	0.9631 ± 0.0010	0.1431 ± 0.0013	0.0061 ± 0.0004	0.0568 ± 0.0009
Deep ensemble	0.9694 ± 0.0002	0.1257 ± 0.0003	0.0086 ± 0.0007	0.0487 ± 0.0001
Trunk. DCA	0.9698 ± 0.0009	0.1212 ± 0.0036	0.0061 ± 0.0005	0.0474 ± 0.0013
Model. DCA	$\underline{0.9713 \pm 0.0005}$	$\underline{0.1128 \pm 0.0014}$	$\underline{0.0055 \pm 0.0005}$	$\underline{0.0452 \pm 0.0007}$

36th Conference on Neural Information Processing Systems (NeurIPS 2022).

B.2 Additional plots for distributional shift experiment

Figure 1 reports Brier scores and ECEs of the distributional shift experiments on CIFAR-10-C.

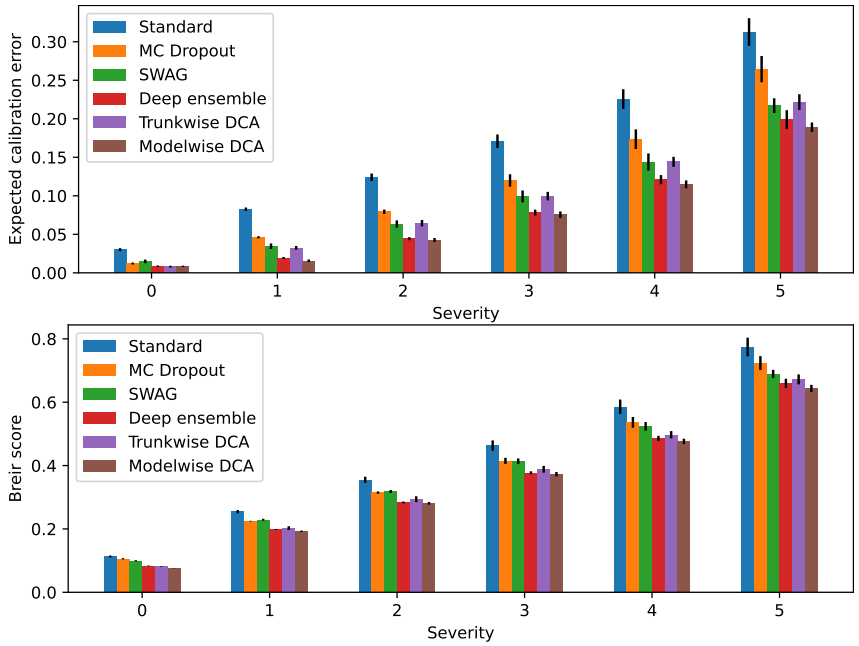

Figure 1: Expected calibration errors and Brier scores of image classification on CIFAR-10-C with various degrees of corruption severities. Modelwsie DCA again has the best overall performance. Here trunkwise DCA has worse results compared to deep ensemble according to these metrics. This is contrary to the conclusion based on negative log-likelihood.

B.3 Additional quantitative results for out-of-distribution experiment

Table 2: FPR95, detection error, AUPR-in, AUPR-out, and AUROC of out-of-distribution detection task using models trained on CIFAR-10 to distinguish between CIFAR-10 (in-domain) and SVHN (OOD) test data. Modelwise DCA consistently yields the best results. All values are in percentage.

	FPR@95% ↓	Detection error ↓	AUROC ↑	AUPR-In ↑	AUPR-Out ↑
Standard	18.0 ± 2.2	16.7 ± 1.6	89.0 ± 1.5	84.8 ± 2.2	93.2 ± 1.0
MC dropout	23.9 ± 3.0	19.1 ± 2.2	85.9 ± 2.0	82.1 ± 2.6	90.9 ± 1.2
SWAG	14.3 ± 2.6	13.4 ± 1.4	92.1 ± 1.5	89.3 ± 1.7	$\underline{95.1 \pm 1.1}$
Deep ensemble	14.0 ± 1.8	12.8 ± 0.9	$\underline{92.5 \pm 0.9}$	90.0 ± 1.0	$\underline{95.4 \pm 0.6}$
Trunk. DCA	$\underline{11.5 \pm 2.1}$	12.2 ± 0.8	$\underline{93.0 \pm 1.0}$	90.5 ± 1.0	$\underline{95.7 \pm 0.7}$
Model. DCA	$\underline{10.8 \pm 2.2}$	$\underline{11.1 \pm 0.6}$	$\underline{93.4 \pm 0.9}$	$\underline{91.5 \pm 0.8}$	$\underline{95.7 \pm 0.7}$

Table 2 provides quantitative metrics for the OOD detection task using CIFAR-10 as the in-domain dataset and SVHN as the OOD outliers. The following metrics are reported:

False positive rate at 95% true positive rate (FPR@95%) A threshold metric which reports the false positive rate at the threshold where the true positive rate is 95%;

Detection error The minimum of average values of false positive rate and false negative rate evaluated at different decision boundaries d: $\min_{d \in [0,1]} (\text{FPR}|_d + \text{FNR}|_d)/2$;

Area under the receiver operating characteristic curve (AUROC) This metric reports the area under the receiver operating characteristic (ROC) curve;

Area under the precision–recall curve (AUPR-in / AUPR-out) This metric measures the area under the precision–recall (AUPR) curve, AUPR-in reports the AUPR where the in-domain data are taken as positive samples, whereas AUPR-out instead treats OOD data as positive samples and in-domain data as negative ones.

B.4 Additional results for ablation study on consistency enforcing loss

Table 3 displays additional results for channelwise, layerwise and blockwise DCA variants.

Table 3: Comparison of DCA training using negative log-likelihood (NLL) loss and consistency enforcing loss (CEL) for channelwise DCA, layerwise DCA, and blockwise DCA models.

	Accuracy	NLL	ECE
Channelwise DCA (NLL)	0.9336 ± 0.0007	0.2190 ± 0.0030	0.0204 ± 0.0013
Channelwise DCA (CEL)	0.9324 ± 0.0014	0.2198 ± 0.0061	0.0233 ± 0.0011
Layerwise DCA (NLL)	0.9342 ± 0.0012	0.2264 ± 0.0088	0.0228 ± 0.0015
Layerwise DCA (CEL)	0.9339 ± 0.0018	0.2190 ± 0.0089	0.0223 ± 0.0015
Blockwise DCA (NLL)	0.9375 ± 0.0004	0.2079 ± 0.0061	0.0161 ± 0.0017
Blockwise DCA (CEL)	0.9357 ± 0.0018	0.2089 ± 0.0031	0.0200 ± 0.0023

C Performance of individual DCA model proposals

To further analyze the effect of the consistency enforcing loss for DCA models, we inspect the predictions of the individual model copies from modelwise DCAs training with and without the consistency enforcing loss. The results are reported in Table 4.

Table 4: Results for individual model copies from modelwise DCAs trained using negative log-likelihood (NLL) loss and consistency enforcing loss (CEL) on CIFAR-10. Results for baseline standard training are included for reference.

	Accuracy ↑	NLL ↓	ECE ↓
Standard	0.9264 ± 0.0020	0.2507 ± 0.0078	0.0301 ± 0.0020
DCA individual (NLL)	0.8964 ± 0.0075	0.3561 ± 0.0229	0.0364 ± 0.0029
DCA individual (CEL)	0.9026 ± 0.0055	0.3055 ± 0.0161	0.0195 ± 0.0022

We observe that individual models trained with the consistency enforcing loss have better results compared to the NLL case, which shows that CEL improves the training of individual components. Compared to the baseline standard training, individual model proposals from trained DCA models have worse classification performance, however their aggregation yields superior results both in terms of accuracy and uncertainty estimation.

In Table 4, we also see that results from individual models trained without CEL have larger standard deviations, which hints that the predictions might be more diverse in this case. To further confirm this, we employ the following metrics to quantify the diversity of individual predictions [1] (assuming we have M individual models and C output classes, KL denotes the KL-divergence, Var denotes the variance, and for a given input x, $p_c^m(x)$ denotes the predicted probability of class c from model m):

Pairwise KL divergence

$$D_{pw}(x) = \frac{1}{M(M-1)} \sum_{i=1}^{M} \sum_{j=1}^{M} \mathrm{KL}(\mathbf{p}^i(x) \| \mathbf{p}^j(x)); \tag{1}$$

Classwise variance

$$D_{var}(x) = \sum_{c=1}^{C} \operatorname*{Var}_{m \in \{1,\dots,M\}} \left[p_c^m(x) \right]; \tag{2}$$

Jensen-Shannon divergence

$$D_{js}(x) = \frac{1}{M} \sum_{i=1}^{M} \mathrm{KL}\left(\mathbf{p}^i(x) \middle\| \frac{1}{M} \sum_{j=1}^{M} \mathbf{p}^j(x) \right). \tag{3}$$

The results are collected in Table 5 and confirm that individual models trained without CEL have more diverse predictions.

Table 5: Diversity metrics for individual model copies from modelwise DCAs trained using negative log-likelihood (NLL) loss and consistency enforcing loss (CEL) on CIFAR-10. Larger values indicate more diverse predictions. Classwise variance uses Bessel's correction to get unbiased estimations.

	Pairwise KL	Classwise variance	JS divergence
DCA individual (NLL)	0.0077 ± 0.0002	0.0011 ± 0.0000	0.0031 ± 0.0001
DCA individual (CEL)	0.0052 ± 0.0002	0.0007 ± 0.0000	0.0020 ± 0.0001

D Additional results using VGG and DenseNet

In previous experiments we focus on the preactivation ResNet-20 base model. Since the DCA approach does not rely on specific choice of network architecture, we expect the conclusions to hold for other network structures. To empirically verify this, we conduct additional in-domain experiments on CIFAR-10 using two more architectures: VGG-16 [4] and DenseNet-40 [3]. The results are summarized in Table 6 for the VGG backbone and in Table 7 for the DenseNet case. We see that modelwise DCA consistently achieves the best results both in terms of accuracy and uncertainty estimation for all tested network architectures.

Table 6: In-domain image classification results on CIFAR-10 using VGG-16 backbone.

	Accuracy ↑	NLL ↓	ECE ↓	Brier ↓
Standard	0.9342 ± 0.0004	0.3436 ± 0.0105	0.0512 ± 0.0009	0.1147 ± 0.0016
MC dropout	0.9379 ± 0.0013	0.2474 ± 0.0063	0.0341 ± 0.0018	0.0980 ± 0.0025
Deep ensemble	0.9494 ± 0.0006	0.1862 ± 0.0010	0.0142 ± 0.0007	0.0779 ± 0.0010
Model. DCA	0.9509 ± 0.0006	0.1687 ± 0.0013	0.0143 ± 0.0005	0.0745 ± 0.0009

Table 7: In-domain image classification results on CIFAR-10 using DenseNet-40 backbone.

	Accuracy ↑	NLL ↓	ECE ↓	Brier ↓
Standard	0.9306 ± 0.0033	0.2604 ± 0.0097	0.0353 ± 0.0022	0.1082 ± 0.0047
MC dropout	0.9284 ± 0.0017	0.2128 ± 0.0041	0.0119 ± 0.0010	0.1049 ± 0.0007
Deep ensemble	0.9477 ± 0.0013	0.1649 ± 0.0017	0.0087 ± 0.0005	0.0791 ± 0.0008
Model. DCA	0.9503 ± 0.0006	0.1486 ± 0.0011	0.0063 ± 0.0004	0.0735 ± 0.0005

E Text classification experiments

Additionally, we consider the performance of DCA for natural language processing tasks. Especially, we address the task of text classification on the AG News dataset [5] using a Seq2seq model with attention [2]. We adapt from an existing implementation[1] and collect the results in Table 8. We observe that modelwise DCA achieves the best accuracy when trained with the NLL loss. Using the consistency enforcing loss, modelwise DCA achieves the best ECE.

Table 8: In-domain text classification results on AG News.

	Accuracy ↑	NLL ↓	ECE ↓	Brier ↓
Standard	0.9147 ± 0.0018	0.2490 ± 0.0058	0.0157 ± 0.0036	0.1291 ± 0.0033
Deep ensemble	0.9192 ± 0.0017	0.2323 ± 0.0043	0.0098 ± 0.0018	0.1224 ± 0.0019
DCA (CEL)	0.9140 ± 0.0026	0.2446 ± 0.0065	0.0081 ± 0.0039	0.1285 ± 0.0031
DCA (NLL)	0.9213 ± 0.0015	0.2312 ± 0.0041	0.0145 ± 0.0022	0.1199 ± 0.0014

[1] `https://github.com/AnubhavGupta3377/Text-Classification-Models-Pytorch`

References

[1] T. Abe, E. K. Buchanan, G. Pleiss, R. Zemel, and J. P. Cunningham. Deep ensembles work, but are they necessary? In *NeurIPS*, 2022.

[2] D. Bahdanau, K. Cho, and Y. Bengio. Neural machine translation by jointly learning to align and translate. In *ICLR*, 2015.

[3] G. Huang, Z. Liu, L. van der Maaten, and K. Q. Weinberger. Densely connected convolutional networks. In *CVPR*, 2017.

[4] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. In *ICLR*, 2015.

[5] X. Zhang, J. J. Zhao, and Y. LeCun. Character-level convolutional networks for text classification. In *NIPS*, 2015.