

Supplementary Materials: MetaRepair: Learning to Repair Deep Neural Networks from Repairing Experiences

1 EXPERIMENTAL DETAIL

1.1 Environments

We implement MetaRepair with Python 3.7 based on PyTorch 1.12. All the experiments are conducted on the same workstation with a 12-core i9-10920X @ 3.50GHz CPU (256GB RAM) and two NVIDIA A5000 GPUs (24GB memory each). The operation system is Ubuntu 20.04.

1.2 Training & testing details

For all the experiments, MetaRepair is trained for 150 epoches with a batch size of 5. During meta-training, the α and β are empirically set to -0.03 and -30 for confidence calculation as Eq. 8. In detail, we sample $N = 2000$ tasks for meta-training while the number of tasks for meta-test, i.e., M in Algorithm 1, is calculated based on number of test examples, D_{test} size and batch size, i.e., $M = \frac{n(\mathcal{D}_{fail}^{target})}{B \cdot n(D_{test})}$ where $n(\cdot)$ is the example counting operation. For the D_{train} and D_{test} in a task, we select one available example from each class to form D_{train} that is semantic-cover, while D_{test} are set 50 and 200 for CIFAR-10 and *tiny*-ImageNet respectively.

1.3 Repairing Experience Setup

As briefly stated in Sec. 5.1.1, we select gaussian noise, zoom blur, fog, brightness and all digital error corruptions as target corruptions for repairing in the main experiments. During the experiments, we set the target corruption is the only inaccessible corruption while other 18 corruptions are all utilized as repairing experiences. It should be noted that such setup is still reasonable since all the data that can be accessed from the repairing experiences are the collected few examples which still follows the real-world setup for DNN repair problem. As for the experiences in the experiments of gradually add correlated corruptions and test generalization performance, we keep only utilizing those corruptions that not involved as repairing targets.

1.4 Others

For the Inception v3 features utilized to calculate the experience confidence, we directly adopt the torchvision [3] official implementation that has pretrained by ImageNet [2]. Another experimental detail needs to be noted is that we only set D_{train} as semantic-cover which means examples of D_{train} cover all the classes of dataset, while the examples of D_{test} are randomly sampled without consideration of example class.

2 LIMITATIONS

Although the proposed L2R and corresponding MetaRepair achieves promising repairing performance for different DNNs on both CIFAR-10 and *tiny*-ImageNet datasets. We point out here two main limitations of the proposed L2R strategy.

First of all, even our approach is generalizable to different type of corruptions, we note that the crucial repairing capability is acquired with the availability of repairing experiences, especially

those highly correlated ones. While we only utilize the few collected failures from each corruption as experiences, the corruptions themselves are commonly not easy or require effort for accessing. As one of the possible amendment for utilizing L2R under such situation is to manually synthesize the correlated data like [4].

Another limitation of the proposed L2R is the time consumed. On the one hand, repairing DNN by optimization requires DNN retraining or fine-tuning which is time-consuming. On the other hand, the meta-learning is well-known of its slow episodic training strategy. Potentially, more efficient way for repairing DNN is to repair DNN with test-time adaptation [1] which aims to project the corrupted data into the clean domain while freezing the DNN during repairing process.

3 NOTATION TABLE

t	the “target” meta-test corruption
k	the index of k -th meta-train corruption
$F_{\theta}(\cdot)$	the pretrained DNN with parameter θ
$\mathcal{P}_{clean}, \mathcal{P}_{corrupt}$	the clean and corrupted data distribution
D_{train}, D_{test}	the clean train & test dataset
$D_{corrupt}$	the overall corrupted dataset
$D_{collect}, D_{fail}$	the collected and inaccessible corrupted data
D_{train}, D_{test}	the train and test data of a meta-learning task
$\hat{\theta}, \hat{\theta}_{meta}$	the updated and meta-learned DNN parameter
ξ, ξ_{meta}	the meta-knowledge before and after meta-learning
$q_{corrupt}, p_{corrupt}$	the corruption-wise FID and sampling probability

4 MORE BASIC GENERALIZATION RESULTS

For the basic generalization results of other three categories of corruptions, we show them in the next page from which the consistent generalization can be observed. Note we do not conduct more one-for-all generalization experiments cause the only variation is the support repairing experiences, while such variation can be easily derived with the cross-corruption FID results.

REFERENCES

- [1] Jin Gao, Jialing Zhang, Xihui Liu, Trevor Darrell, Evan Shelhamer, and Dequan Wang. 2023. Back to the source: Diffusion-driven adaptation to test-time corruption. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 11786–11796.
- [2] Alexlex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. 2017. Imagenet classification with deep convolutional neural networks. *Commun. ACM* 60, 6 (2017), 84–90.
- [3] Sébastien Marcel and Yann Rodriguez. 2010. Torchvision the machine-vision package of torch. In *Proceedings of the 18th ACM international conference on Multimedia*. 1485–1488.
- [4] Bing Yu, Hua Qi, Qing Guo, Felix Juefei-Xu, Xiaofei Xie, Lei Ma, and Jianjun Zhao. 2021. Deeprepair: Style-guided repairing for deep neural networks in the real-world operational environment. *IEEE Transactions on Reliability* 71, 4 (2021), 1401–1416.

Table 1: Basic generalization results for blur corruptions with zoom blur as repairing basis.

Corruption \ Model	CIFAR-10					tiny-ImageNet				
	Base	DB	GB	MB	GAB	Base	DB	GB	MB	GAB
DenseNet	80.56	58.95	55.80	71.35	65.22	17.88	15.82	13.29	17.70	16.20
ConvNeXt	78.20	62.08	60.69	75.37	67.00	29.06	14.12	14.01	20.90	22.68
VAN	78.01	62.36	59.31	73.75	63.16	30.11	15.77	14.17	29.27	28.00

Table 2: Basic generalization results for weather corruptions with fog as repairing basis.

Corruption \ Model	CIFAR-10				tiny-ImageNet			
	Base	SW	FR	SP	Base	SW	FR	SP
DenseNet	50.38	65.73	50.88	60.99	13.24	23.55	15.19	20.15
ConvNeXt	52.60	65.55	51.15	62.32	15.12	25.70	18.10	19.78
VAN	50.72	62.37	55.58	60.30	15.88	23.57	16.88	20.48

Table 3: Basic generalization results for image property corruptions with brightness as repairing basis.

Corruption \ Model	CIFAR-10			tiny-ImageNet		
	Base	CT	SA	Base	CT	SA
DenseNet	59.81	42.18	50.34	18.72	12.84	15.92
ConvNeXt	60.17	45.45	53.42	21.69	15.60	15.50
VAN	60.41	44.36	55.90	20.30	15.14	14.65