A APPENDIX

A.1 FEW-SHOT LEARNER META-TRAINING PROTOCOLS

In the following, the meta-training protocols for the various few-shot learners used in the experiments including MAML, ProtoNets, and CNAPS will be detailed.

A.1.1 DATASETS

***mini*ImageNet** *mini*ImageNet is a subset of the larger Imagenet dataset (Russakovsky et al., 2015) created by Vinyals et al. (2016). It consists of 60,000 color images that is sub-divided into 100 classes, each with 600 instances. The images have dimensions of 84×84 pixels. Ravi & Larochelle (2017) standardized the 64 training, 16 validation, and 20 test class splits. *mini*ImageNet has become a defacto standard dataset for benchmarking few-shot image classification methods with the following classification task configurations: (i) 5-way, 1-shot; (ii) 5-way, 5-shot.

META-DATASET META-DATASET (Triantafillou et al., 2020) is composed of ten (eight train, two test) image classification datasets. We augment Meta-Dataset with three additional held-out datasets: MNIST (LeCun et al., 2010), CIFAR10 (Krizhevsky & Hinton, 2009), and CIFAR100 (Krizhevsky & Hinton, 2009). The challenge constructs few-shot learning tasks by drawing from the following distribution. First, one of the datasets is sampled uniformly; second, the "way" and "shot" are sampled randomly according to a fixed procedure; third, the classes and support / query instances are sampled. Where a hierarchical structure exists in the data (ImageNet or Omniglot), task-sampling respects the hierarchy. In the meta-test phase, the identity of the original dataset is not revealed and the tasks must be treated independently (i.e. no information can be transferred between them). Notably, the meta-training set comprises a disjoint and dissimilar set of classes from those used for meta-test. META-DATASET is presently, the "gold standard" for evaluating few-shot classification methods. Full details are available in Triantafillou et al. (2020).

In our experiments, we excluded the Omniglot, Textures, Fungi, and Traffic Signs datasets from evaluation because their test splits are too small to allow for a fair assessment of the attack's generalization, even though the attacks reduced the classification accuracy on those datasets to approximately zero in the *Support Specific* case.

A.1.2 MAML META-TRAINING PROTOCOL

We meta-trained our implementation of MAML with identical network configuration, hyper-parameters, and training protocol as prescribed in Finn et al. (2017). The meta-trained models attained the following accuracy:

*mini*ImageNet 5-way, 1-shot: $47.2 \pm 1.7\%$

*mini*ImageNet 5-way, 5-shot: $61.3 \pm 0.9\%$

A.1.3 PROTONETS META-TRAINING PROTOCOL

We meta-trained our implementation of Prototypical Networks with identical network configuration, hyper-parameters, and training protocol as prescribed in Snell et al. (2017). The meta-trained models attained the following accuracy:

*mini*ImageNet 5-way, 1-shot: $46.8 \pm 0.6\%$

*mini*ImageNet 5-way, 5-shot: $65.1 \pm 0.5\%$

A.1.4 CNAPS META-TRAINING PROTOCOL

For all the CNAPS experiments, we use the code provided by the the CNAPS authors (Requeima et al., 2019b). We made modifications to the code to enable various adversarial attacks. We follow an identical dataset configuration and meta-training process as prescribed in Requeima et al. (2019b).

A.1.5 FINE-TUNING PROTOCOL

We pretrain two separate feature extractors (ResNet18 (He et al., 2016) and MNASNet (Tan et al., 2019)) off-line on the training split of the ILSVRC 2012 dataset from the META-DATASET benchmark on images of size 84×84 pixels. The final classifier layer of this pre-trained model is then removed and the remaining layers serve as an embedding function. Note that the meta-training phase is not required when fine-tuning. During meta-testing, for each task, the removed final layer is replaced with an untrained fully-connected layer with input size being equal to the embedding function output dimension and output size equal to the way of the task. The support set data from the test task is then used to train the new final classifier layer and FiLM layers (Perez et al., 2018; Requeima et al., 2019a) inserted into the feature extractors to adapt the pretrained classifier weights to the current task using stochastic gradient descent. This network trained on the support set can now be evaluated using the query set data from the task.

In particular, we use the following hyper-parameters in the fine tuning experiments: Stochastic Gradient Descent with learning rate 0.05, momentum 0.9, and weight decay 0.001 for 50 iterations.

A.2 QUERY ATTACKS

We present our algorithm for performing query attacks with PGD in Algorithm A.1. Using this algorithm, we attack MAML and ProtoNets with settings that match the experiments by (Goldblum et al., 2019) to ensure that our attack performs approximately the same, as expected. The attack settings used are $L = 20$, $\epsilon = \frac{8}{255}$, $\gamma = \frac{2}{255}$, with an untargeted loss function. Our results are shown in Table A.1 and the relevant results from (Goldblum et al., 2019) are shown in Table A.2. Our models perform similarly when presented with clean data and when attacked using PGD, as expected.

Algorithm A.1 PGD for Query Attack

Require:
- I_{min}: Minimum image intensity
- I_{max}: Maximum image intensity
- L: Number of iterations
- ϵ: Perturbation amount
- γ: Step size
- $D_S \equiv \{\boldsymbol{x}, \boldsymbol{y}\}$
- $D_Q \equiv \{\boldsymbol{x}^*, \boldsymbol{y}^*\}$
- ▷ We use cross-entropy loss for $\mathcal{L}$.

procedure PGDQ(D_S, D_Q, f, g)
  $\boldsymbol{\delta} \sim U(-\epsilon, \epsilon)$
  $\tilde{\boldsymbol{x}}^* \leftarrow \text{clip}(\boldsymbol{x}^* + \boldsymbol{\delta}, I_{min}, I_{max})$
  for $n \in 1, ..., L$ **do**
    $\boldsymbol{\delta} \leftarrow \text{sgn}(\nabla_{\tilde{\boldsymbol{x}}^*} \mathcal{L}(f(\tilde{x}^*, g(x, y)), y^*)$
    $\tilde{\boldsymbol{x}}^* \leftarrow \text{clip}(\tilde{\boldsymbol{x}}^* + \gamma\boldsymbol{\delta}, I_{min}, I_{max})$
    $\tilde{\boldsymbol{x}}^* \leftarrow x + \text{clip}(\tilde{\boldsymbol{x}}^* - \boldsymbol{x}^*, -\epsilon, \epsilon)$
  end for
  return $\tilde{\boldsymbol{x}}^*$
end procedure

Table A.1: The classification accuracy (%) when performing our query attack against MAML and ProtoNets models in the 5-way, 1-shot and 5-shot configurations on *mini*ImageNet. Results are averaged over 500 tasks and PGD settings were L=20, with $\epsilon = \frac{8}{255}$, $\gamma = \frac{2}{255}$. All figures are percentages and the $\pm$ sign indicates the 95% confidence interval over tasks.

	MAML		ProtoNets	
	Clean	Adversarial	Clean	Adversarial
*mini*ImageNet 5-way, 1 shot	47.0+/-0.3	0.0+/-0.0	46.6+/-0.3	0.0+/-0.0
*mini*ImageNet 5-way, 5 shot	60.7+/-0.1	0.0+/-0.0	64.7+/-0.1	0.0+/-0.0

Table A.2: Results reproduced from Goldblum et al. (2019) where possible. The table shows classification accuracy (%) when performing a query attack against MAML and ProtoNets models in the 5-way, 1-shot and 5-shot configurations on *mini*ImageNet. Results were tested on 150000 samples. PGD settings were L=20, with $\epsilon = \frac{8}{255}$, $\gamma = \frac{2}{255}$. All figures are percentages.

	MAML		ProtoNets	
	Clean	Adversarial	Clean	Adversarial
*mini*ImageNet 5-way, 1 shot	45.04	0.03	43.26	0.00
*mini*ImageNet 5-way, 5 shot	60.25	0.03	70.23	0.00

A.3 Additional Details and Experiments

In our experiments, all the input images were re-scaled to have pixel values between -1 and 1. We considered perturbations using the ℓ_∞ norm, on a scale of $[-1, 1]$, so that $\epsilon = 0.1$ corresponds to allowing $\pm 10\%$ or an absolute change of ± 0.2 to the intensity of each pixel in an image.

We calculated the perturbation step size γ to depend on ϵ and the maximum number of iterations, so that $\gamma = r\frac{\epsilon}{L}$, where r is a scaling coefficient. We observed that the optimal values for r depend on the numbers of shots, and varies with ϵ and L. The results of our tuning experiments are provided for ProtoNets in Table A.3 - Table A.5. In general, larger values of r (i.e. larger step sizes) performed better as the number of PGD iterations increased. Although the *single* loss strategy did not perform well for 1-shot classification, its performance increased at higher shots, often out-performing the *all* strategy for sufficiently large numbers of PGD iterations, even though the *Specific* accuracy did not go to 0.0%. The *all* strategy performed significantly better than *single* when L was low.

Table A.3: Accuracy of ProtoNets 5-way 1-shot, with perturbation size $\epsilon = 0.05$ when varying the loss function (targeted *all* or untargeted *single*), PGD iterations (given in the column headers) and step size ratio (r), over 100 tasks. Seed query set size is fixed at $13N$. Clean accuracy is $47.5 \pm 2.0\%$. All figures are percentages and the $\pm$ sign indicates the 95% confidence interval over tasks. Bold text indicates the lowest score.

		20		50		100		200		500	
	r	Specific	General	Specific	General	Specific	General	Specific	General	Specific	General
all	0.25	7.4±0.9	17.0±0.5	7.0±0.9	16.8±0.5	6.9±1.0	16.7±0.5	6.8±0.9	16.8±0.5	6.9±0.9	16.6±0.5
	0.5	**3.4±0.6**	13.2±0.4	1.5±0.3	12.2±0.4	1.2±0.3	11.8±0.4	1.3±0.3	11.8±0.4	1.3±0.3	11.8±0.4
	1	3.7±0.6	**12.9±0.4**	**0.8±0.2**	10.9±0.4	0.3±0.2	10.1±0.4	0.1±0.1	9.9±0.4	0.1±0.1	10.1±0.4
	1.5	3.9±0.7	**12.9±0.4**	0.9±0.3	**10.8±0.4**	**0.1±0.1**	9.6±0.4	**0.0±0.0**	9.3±0.4	**0.0±0.0**	9.6±0.4
	2	4.9±0.8	13.5±0.4	1.2±0.3	**10.8±0.4**	0.2±0.1	9.4±0.4	**0.0±0.0**	9.2±0.4	**0.0±0.0**	9.3±0.4
	3	7.1±1.1	14.4±0.4	1.5±0.4	11.0±0.4	0.2±0.1	**9.3±0.4**	**0.0±0.0**	**9.0±0.4**	**0.0±0.0**	**8.8±0.3**
single	0.25	32.8±2.1	34.0±0.6	29.9±1.9	32.6±0.5	30.6±2.0	33.8±0.6	30.6±1.9	34.1±0.6	31.3±2.0	35.0±0.6
	0.5	26.1±1.8	27.7±0.5	21.6±1.6	25.6±0.5	21.2±1.7	25.0±0.5	20.9±1.7	25.5±0.5	21.2±1.7	26.1±0.5
	1	21.3±1.6	23.2±0.5	15.0±1.5	18.8±0.5	13.5±1.3	17.4±0.4	12.6±1.3	17.7±0.5	12.0±1.4	17.4±0.5
	1.5	20.3±1.7	22.1±0.5	13.1±1.5	16.4±0.4	10.9±1.2	14.7±0.4	9.2±1.1	14.3±0.4	8.2±1.1	13.6±0.4
	2	**19.5±1.6**	**21.7±0.5**	12.6±1.4	15.4±0.4	10.0±1.2	13.2±0.4	7.8±1.0	12.3±0.4	6.4±1.0	11.9±0.4
	3	20.8±1.8	22.4±0.5	**11.9±1.3**	**14.9±0.4**	**8.9±1.1**	**12.0±0.4**	**6.4±1.0**	**10.7±0.4**	**5.0±0.8**	**9.6±0.4**

Table A.4: Accuracy of ProtoNets 5-way 5-shot, with perturbation size $\epsilon = 0.05$ when varying the loss function (targeted *all* or untargeted *single*), PGD iterations (given in the column headers) and step size ratio (r), over 100 tasks. Seed query set size is fixed at $7N$. Clean accuracy is $64.2 \pm 1.6\%$. All figures are percentages and the $\pm$ sign indicates the 95% confidence interval over tasks. Bold text indicates the lowest score.

		20		50		100		200		500	
	r	Specific	General	Specific	General	Specific	General	Specific	General	Specific	General
all	0.25	**1.1±0.2**	9.8±0.2	0.1±0.0	9.0±0.2	**0.0±0.0**	8.7±0.2	**0.0±0.0**	8.7±0.2	**0.0±0.0**	8.7±0.2
	0.5	1.5±0.3	10.3±0.2	**0.0±0.0**	8.1±0.2	**0.0±0.0**	7.8±0.2	**0.0±0.0**	7.6±0.2	**0.0±0.0**	7.7±0.2
	1	1.9±0.5	9.8±0.2	0.1±0.0	**8.0±0.2**	**0.0±0.0**	7.4±0.2	**0.0±0.0**	6.9±0.1	**0.0±0.0**	7.0±0.1
	1.5	2.3±0.5	**9.7±0.2**	0.1±0.1	8.1±0.2	**0.0±0.0**	**7.1±0.2**	**0.0±0.0**	6.8±0.1	**0.0±0.0**	6.6±0.1
	2	3.3±0.6	10.6±0.2	0.1±0.1	8.2±0.2	**0.0±0.0**	7.2±0.2	**0.0±0.0**	6.6±0.1	**0.0±0.0**	6.5±0.1
	3	6.4±0.9	11.9±0.2	0.2±0.1	8.1±0.2	**0.0±0.0**	7.3±0.2	**0.0±0.0**	**6.5±0.1**	**0.0±0.0**	**6.3±0.1**
single	0.25	36.0±1.6	38.6±0.3	35.9±1.7	39.3±0.3	34.9±1.7	39.3±0.3	33.5±1.9	38.7±0.3	33.5±1.9	39.3±0.4
	0.5	24.7±1.4	26.3±0.3	17.3±1.4	20.7±0.3	14.6±1.2	18.4±0.3	12.2±1.2	17.0±0.3	10.8±1.2	16.6±0.3
	1	18.3±1.4	19.7±0.3	9.1±1.1	11.2±0.2	5.9±0.7	8.2±0.2	3.7±0.5	6.5±0.2	2.9±0.5	6.1±0.2
	1.5	**17.9±1.4**	**19.0±0.3**	7.7±1.0	9.5±0.2	4.5±0.6	6.2±0.2	2.4±0.3	4.4±0.1	1.6±0.3	3.9±0.1
	2	18.5±1.4	19.4±0.3	**7.5±1.0**	**9.1±0.2**	**4.0±0.6**	5.5±0.1	**2.0±0.3**	3.7±0.1	1.2±0.2	3.1±0.1
	3	19.8±1.4	21.3±0.3	8.0±1.0	9.6±0.2	4.1±0.6	**5.4±0.1**	**2.0±0.3**	**3.3±0.1**	**1.1±0.2**	**2.6±0.1**

Unnormalized Small-Scale Results To supplement Fig. 4 in Section 4.1, we provide the unnormalized in Tables A.6 and A.7, for the 1-shot and 5-shot scenarios, respectively. All adversarial query points used in the swap attacks achieved approximately 100% fooling rates when presented to the learner as evasion attacks.

Fraction of Poisoned Patterns In addition to Fig. 5a in Section 4.1 of the paper, we provide similar plots for MAML, ProtoNets and CNAPs on 5-way, 1-shot problems. Note that for these

Table A.5: Accuracy of ProtoNets 5-way 10-shot, with perturbation size $\epsilon = 0.05$ when varying the loss function (targeted *all* or untargeted *single*), PGD iterations (given in the column headers) and step size ratio (r), over 100 tasks. Seed query set size is fixed at $6N$. Clean accuracy is $71.7 \pm 1.1\%$. All figures are percentages and the $\pm$ sign indicates the 95% confidence interval over tasks. Bold text indicates the lowest score.

		20		50		100		200		500	
	r	Specific	General	Specific	General	Specific	General	Specific	General	Specific	General
all	0.25	1.7±0.2	**8.5±0.1**	0.4±0.1	7.6±0.1	0.2±0.1	7.5±0.1	0.2±0.1	7.5±0.1	0.2±0.1	7.5±0.1
	0.5	2.0±0.3	**8.5±0.1**	0.2±0.1	7.2±0.1	0.0±0.0	6.9±0.1	0.0±0.0	6.9±0.1	0.0±0.0	7.0±0.1
	1	2.3±0.4	**8.5±0.1**	0.2±0.1	**7.1±0.1**	0.0±0.0	6.5±0.1	0.0±0.0	6.4±0.1	0.0±0.0	6.5±0.1
	1.5	2.9±0.5	8.9±0.2	0.2±0.1	**7.1±0.1**	0.0±0.0	6.7±0.1	0.0±0.0	6.2±0.1	0.0±0.0	6.2±0.1
	2	4.6±0.6	10.1±0.2	0.4±0.1	7.2±0.1	0.0±0.0	**6.4±0.1**	0.0±0.0	**6.0±0.1**	0.0±0.0	6.0±0.1
	3	8.0±1.0	12.5±0.2	0.7±0.2	7.3±0.1	0.1±0.0	6.5±0.1	0.0±0.0	6.1±0.1	0.0±0.0	**5.9±0.1**
single	0.25	23.6±1.4	24.3±0.2	16.5±1.2	18.2±0.2	13.4±1.2	15.6±0.2	11.2±1.1	14.1±0.2	11.8±1.0	15.1±0.2
	0.5	17.3±1.5	17.9±0.2	8.3±0.9	9.3±0.2	5.6±0.7	6.7±0.1	3.8±0.5	5.2±0.1	3.0±0.4	4.8±0.1
	1	15.8±1.4	**16.5±0.2**	6.3±0.7	**7.1±0.1**	3.7±0.5	4.6±0.1	2.3±0.3	3.2±0.1	1.6±0.2	2.6±0.1
	1.5	16.6±1.3	17.0±0.2	6.4±0.7	7.2±0.1	3.5±0.5	**4.4±0.1**	2.1±0.3	2.9±0.1	1.4±0.2	2.2±0.1
	2	17.9±1.3	18.3±0.2	7.0±0.8	7.7±0.1	3.8±0.5	4.6±0.1	2.1±0.3	**2.8±0.1**	1.3±0.2	2.1±0.1
	3	19.0±1.2	19.7±0.2	7.8±0.8	8.4±0.2	4.2±0.6	5.1±0.1	2.2±0.3	2.9±0.1	1.4±0.2	**2.0±0.1**

Table A.6: The classification accuracy for a variety of attacks against MAML and ProtoNets models in the 5-way, 1-shot *mini*ImageNet configuration, averaged over 500 tasks with $M = 20N$. All support images were perturbed. PGD settings were L=100, with $\gamma = 0.0015$ for $\epsilon = 0.05$, and $\gamma = 9.4\text{e}{-4}$ for $\epsilon = 0.0314$. All figures are percentages and the $\pm$ sign indicates the 95% confidence interval over tasks.

	ϵ	Label Shift	Noise	Support Specific	Support General	Swap
Protonets	0.0314	13.0±0.2	46.3±0.3	1.3±0.1	9.8±0.2	19.2±0.2
(Clean: 47.5±0.3)	0.05	13.0±0.2	43.8±0.3	1.1±0.1	9.4±0.2	18.2±0.2
MAML	0.0314	20.6±0.2	46.9±0.3	0.8±0.1	8.9±0.2	12.0±0.2
(Clean: 46.9±0.3)	0.05	20.6±0.2	46.2±0.3	0.7±0.1	8.8±0.2	11.2±0.2

Table A.7: The classification accuracy (%) for a variety of attacks against MAML and ProtoNets models in the 5-way, 5-shot *mini*ImageNet configuration, averaged over 500 tasks with $M = 20N$. All support images were perturbed. PGD settings were L=100, with $\gamma = 0.0015$ for $\epsilon = 0.05$, and $\gamma = 9.4\text{e}{-4}$ for $\epsilon = 0.0314$. All figures are percentages and the $\pm$ sign indicates the 95% confidence interval over tasks.

	ϵ	Label Shift	Noise	Support Specific	Support General	Swap
Protonets	0.0314	8.9±0.1	64.2±0.1	0.6±0.1	6.2±0.1	19.4±0.1
(Clean: 64.6±0.1)	0.05	8.9±0.1	58.4±0.1	0.9±0.1	6.4±0.1	19.5±0.1
MAML	0.0314	19.6±0.1	60.6±0.1	1.2±0.1	7.2±0.1	9.1±0.1
(Clean: 61.4±0.1)	0.05	19.6±0.1	60.0±0.1	1.5±0.1	7.3±0.1	9.3±0.1

results, MAML and ProtoNets were performing classification on *mini*ImageNet, whereas CNAPS was performing classification on ILSVRC-2012, which is a more difficult problem.

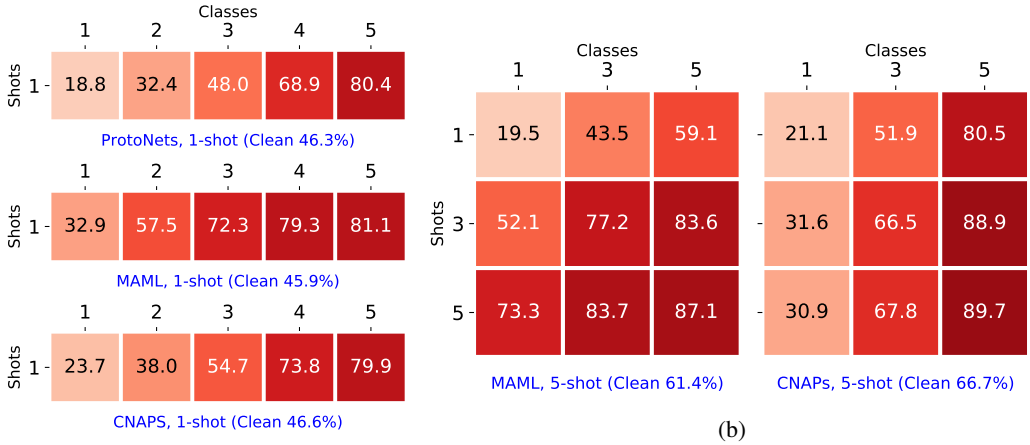

Figure A.1: The relative drop in 5-way classification accuracy of ProtoNets, MAML and CNAPS as the number of poisoned classes and poisoned shots within those classes are varied when performing support set attacks for (a) 1-shot and (b) 5-shot problems. Darker colors indicate a stronger attack. Attacks were calculated with $\epsilon = 0.05$, $\gamma = 0.0015$, $L = 200$, averaged over 250 tasks. The 1-shot problems used $M = 13N$, whereas the 5-shot problems used $M = 7N$. The ProtoNets, 5-shot scenario can be found in the main body of the paper.

A.4 Additional Large-Scale Attack Results

Here, we present the unnormalized numbers for Fig. 6 in Table A.8. We also present results for a similar attack, perpetrated against ProtoNets with FiLM in Table A.9, showing that our attack is also effective against ProtoNets in a large-scale scenario.

Table A.8: Accuracy of CNAPS on the META-DATASET benchmark in the *Clean*, *Specific* and *General* scenarios when attacking with an adversarial support set, with $\epsilon = 0.05$, $\gamma = 0.0015$, $L = 100$, averaged over 500 tasks, with all classes, but only 20% of the shots poisoned. All figures are percentages and the $\pm$ sign indicates the 95% confidence interval over tasks.

	Clean	Specific	General
ilsvrc_2012	49.5±1.2	0.2±0.0	10.8±0.1
omniglot	85.1±0.9	23.6±1.9	-
aircraft	69.2±1.1	0.0±0.0	9.1±0.2
cu_birds	66.3±1.0	0.1±0.0	6.6±0.3
dtd	57.4±0.8	1.6±0.2	-
quickdraw	69.0±0.9	2.3±0.1	12.3±0.1
fungi	42.6±1.2	0.0±0.0	-
vgg_flower	83.9±0.7	1.5±0.2	16.2±0.6
traffic_sign	61.8±0.9	0.3±0.1	-
mscoco	41.5±1.1	0.4±0.0	9.6±0.1
mnist	89.5±0.5	11.9±0.8	32.0±0.2
cifar10	70.2±0.6	0.1±0.0	13.0±0.1
cifar100	48.7±1.1	0.5±0.0	6.1±0.1

Table A.9: Accuracy of ProtoNets with FiLM on the META-DATASET benchmark in the *Clean*, *Specific* and *General* scenarios when attacking with an adversarial support set, with $\epsilon = 0.05$, $\gamma = 0.0015$, $L = 100$, averaged over 500 tasks, with all classes, but only 20% of the shots poisoned. All figures are percentages and the $\pm$ sign indicates the 95% confidence interval over tasks.

	Clean	Specific	General
ilsvrc_2012	54.0±1.1	1.5±0.1	15.0±0.1
omniglot	90.9±0.7	42.9±2.6	-
aircraft	75.9±1.0	0.0±0.0	7.2±0.2
cu_birds	72.2±0.9	2.0±0.2	9.7±0.4
dtd	64.4±0.8	7.8±0.6	-
quickdraw	75.1±0.9	33.8±1.1	42.3±0.2
fungi	43.6±1.1	0.9±0.1	-
vgg_flower	89.6±0.6	10.4±0.8	27.9±1.0
traffic_sign	71.9±0.8	4.2±0.3	-
mscoco	40.3±1.1	2.8±0.1	10.5±0.1
mnist	91.2±0.5	58.3±1.3	68.6±0.2
cifar10	73.1±0.7	1.6±0.1	13.5±0.1
cifar100	56.6±1.2	2.4±0.1	7.9±0.2