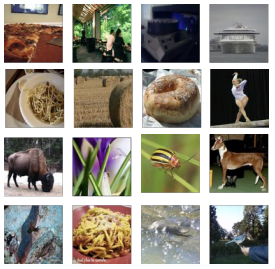
Supplementary: Meta-Meta Classification for One-Shot Learning

Arkabandhu Chowdhury[1], Dipak Chaudhari[2], Swarat Chaudhuri[2], and Chris Jermaine[1]

[1]Rice University, [2]University of Texas, Austin

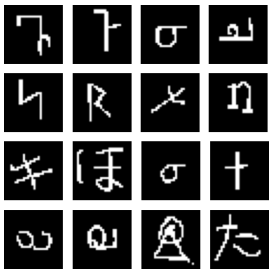
1. Data Sets

In our experiments, we have used four publicly available data sets. A short description of these data sets is given below.



(a) ILSVRC-2012

(b) CUB-2011 Birds



(c) Omniglot

(d) Aircraft

Figure 1: Training examples sampled from the data sets used in our experiments.

ILSVRC-2012. [5] A data set of natural images of 1000 diverse categories, the most commonly used Imagenet data set, primarily released for 'Large Scale Visual Recognition Challenge' (Figure 1a). Randomly chosen 900 classes are used for training and the remaining 100 classes are used for testing.

CUB-200-2011. [7] A data set for fine-grained classification of 200 different bird species, an extended version of the CUB-200 dataset (Figure 1b). All 200 classes are used for cross-domain testing.

Omniglot. [3] A data set of images of 1623 handwritten characters from 50 different alphabets, with 20 examples per class (Figure 1c). Randomly chosen 1200 characters are used for training and 423 classes are used for testing.

Aircraft. [4] A dataset of images of aircrafts spanning 102 model variants, with 100 images per class (Figure 1d). Randomly chosen 87 classes are used for training and 15 classes are used for testing.

Note that in each data set, the validation set is included in the training split.

Data Processing. All images apart from those from Omniglot are resized into 84x84 resolution. Omniglot images are resized into 28x28 resolution and color-inverted for faster training.

2. Sampling one-shot problem instances

Let the training split of n_{mtrn} classes for meta-training be called S_{mtrn} and test split of n_{mtst} classes for meta-testing be called S_{mtst}. Meta-learning relies on a distribution of learning problems which are sampled as follows.

Meta-training. We sample one class from S_{mtrn} and designate it as a "positive" class, called $C_{mtrn}^{<pos>}$. Then we sample 50 "negative" classes, called $C_{mtrn,trn}^{<neg>}$, from the remaining (n_{mtrn}-1) classes of S_{mtrn}. Now, the support set or training set, D_{trn}, is generated by sampling one image from the selected positive class, and 50 images from the 50 negative classes. The sampling from negative classes is done with replacement, so some negative classes may have multiple samples, and some may not be represented in D_{trn}. For the query set or test set, D_{tst}, we use the same positive class and again sample 50 "negative" classes, $C_{mtrn,tst}^{<neg>}$, from the remaining (n_{mtrn}-1) classes of S_{mtrn}. Then D_{tst} is generated by sampling 50 images from the positive class, and 50 images from the negative classes with replacement. Note that $C_{mtrn}^{<pos>}$ is a single class whereas $C_{mtrn,trn}^{<neg>}$ and $C_{mtrn,tst}^{<neg>}$ are sets of 50 classes.

So, in total, we sample 151 images which constitute one learning problem for meta-training as follows:
D_{trn}: 1 from $C_{mtrn}^{<pos>}$, 50 from $C_{mtrn,trn}^{<neg>}$
D_{tst}: 50 from $C_{mtrn}^{<pos>}$, 50 from $C_{mtrn,tst}^{<neg>}$.

Meta-testing. Sampling learning problems for meta-testing is exactly the same as for meta-training, only difference

being we use S_{mtst} instead of S_{mtrn} data. So, here also, we sample 151 images which constitute one learning problem for meta-testing as follows:
D_{trn}: 1 from $C_{mtst}^{<pos>}$, 50 from $C_{mtst,trn}^{<neg>}$
D_{tst}: 50 from $C_{mtst}^{<pos>}$, 50 from $C_{mtst,tst}^{<neg>}$.

Note that when we run MAML on clusters, for each cluster, the positive image comes from that particular cluster and all other images come from the entire meta-training set, S_{mtrn}.

3. Architecture and Hyperparameters

For pretraining a feature extractor, we used a modified ResNet-152 [2] model where, in order to minimize number of features and reduce computation cost, we changed the number of output channels of the convolutions from [64, 128, 256, 512] to [64, 64, 128, 256] and block expansion from 4 to 2. We empirically found that the features produced by ResNet-152 resulted in more meaningful clusters through k-means clustering than some of the other pre-trained models we tried, such as, ResNet-18 and ResNet-50. For future improvement, a thorough experimental study on the choice of feature extractors and the clustering algorithm could be done.

For MAML, we used the commonly-used four-layer convolutional network [1]. While we acknowledge that the meta-learners have shown improved performance with resnet-18 architecture [6], we kept the basic four-layer convnet for consistency in our method as well as the comparing methods. Since we design our method as a binary classifier, we avoided batch-normalization. This is because while doing the testing during meta-learning, batch-normalization can learn to distinguish the negative images from the positive ones without training, provided negative images come from multiple different negative classes.

In order to realize the meta-aggregation function, we used a fully connected feed forward network with two hidden layers of 256 units. The input to the meta-aggregator is the concatenation of the 512 dimensional pretrained ResNet-152 feature vector of the 'positive' support image and the binary logits of the corresponding 'negative' query images coming from individual learners. We used dropout with probability 0.9 for the input layer and 0.6 for the hidden layers during training.

Following the hyper-parameters used in the MAML work [1], we also used a meta batch-size of 4 for all of the problems. We trained our model for 60,000 iterations in all of the data sets except for Aircraft where we used 40,000 iterations since Aircraft data set is significantly smaller than others. For all of the classification tasks, we used 5 inner gradient steps with a learning rate of 0.0001 and a meta-learning rate of 0.001.

Choosing the right number of clusters k can be a bit tricky with k-means clustering as it is not often possible to find a distinct *'elbow point'*. We experimented the classification performance with increasing k logarithmaclly with a step of 2 for all of our data sets, and reported the results till we found a significant improvement before the accuracy reaches a saturation. We found $k = 16$ to be, in general, a good value for the number of clusters, however, it can depend a lot on the size and the nature of the data. For example, since Aircraft data is significantly smaller than the rest, we noticed no significant improvement (at least 1% in test accuracy) when we moved from $k = 8$ to $k = 16$, hence we showed results till 8 clusters for this particular data set.

Finally, we tuned the learning rate schedule and weight decay and we used ADAM optimizer to train all the models. Other details and the complete set of hyperparameters used are included in the source code which we will make public.

References

[1] Chelsea Finn, Pieter Abbeel, and Sergey Levine. Model-agnostic meta-learning for fast adaptation of deep networks. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, pages 1126–1135. JMLR. org, 2017.

[2] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[3] Brenden M Lake, Ruslan Salakhutdinov, and Joshua B Tenenbaum. Human-level concept learning through probabilistic program induction. *Science*, 350(6266):1332–1338, 2015.

[4] S. Maji, J. Kannala, E. Rahtu, M. Blaschko, and A. Vedaldi. Fine-grained visual classification of aircraft. Technical report, 2013.

[5] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander C. Berg, and Li Fei-Fei. ImageNet Large Scale Visual Recognition Challenge. *International Journal of Computer Vision (IJCV)*, 115(3):211–252, 2015.

[6] Eleni Triantafillou, Tyler Zhu, Vincent Dumoulin, Pascal Lamblin, Utku Evci, Kelvin Xu, Ross Goroshin, Carles Gelada, Kevin Swersky, Pierre-Antoine Manzagol, et al. Meta-dataset: A dataset of datasets for learning to learn from few examples. *arXiv preprint arXiv:1903.03096*, 2019.

[7] P. Welinder, S. Branson, T. Mita, C. Wah, F. Schroff, S. Belongie, and P. Perona. Caltech-UCSD Birds 200. Technical Report CNS-TR-2010-001, California Institute of Technology, 2010.