Learning With High-Level Attributes

Thao Nguyen
thaonguyen@cs.stanford.edu

12 June 2019

Abstract

Increasing quantity and availability of human knowledge offers an opportunity to improve joint human-machine performance as well as to give more causal feedback to users. This project explores different training settings where high-level attributes are included and a potential benefit that it can offer in terms of data efficiency. We experiment with fine-grained classification on the Caltech Birds Dataset, where domain knowledge could be of great value.

1 Introduction

Human experts are increasingly being assisted by machine learning models, especially in high-stake decision-making settings. The current setup is inefficient in several ways. For example, limited interpretability of a model's predictions makes it difficult for human experts to confidently incorporate them into their own work-flows. In addition, model training typically relegates human experts to the task of labeling examples, and the experts don't have any opportunity to impart their domain-specific knowledge to the model during the learning process.

Several existing studies have shown that unlike humans, artificial neural networks can often pay a lot of attention to arbitrary, non-causal details and miss the broader picture (e.g. bias towards texture in object recognition [4], or insensitive to shuffling of image parts [2]). Thus, we seek to study a structured way to train models to incorporate high-level supervision in reasoning about the final outputs. Although this increased modularity comes with additional costs of metadata collection, we hope to show that models trained this way will offer better defense against adversarial inputs as well as data-efficiency. Note that this is different from feature engineering, which focuses on easily computable features. Instead, we want to study more abstract, high-level features that are difficult to define a labelling function for.

This project focuses on the fine-grained classification setting, and in particular, the Caltech UCSD Birds 200-2011 dataset (CUB-200-2011 [7]). Fine-grained classification is challenging because of the difficulty of finding discriminative features, which often rely on domain knowledge, while neural networks tend to learn only the high-level abstraction of the images. Thus, this may be an appropriate task to demonstrate the extent to which neural networks pick up on useful non-causal correlations and how that strategy would not work as well with limited training data, and how training with high-level attributes might help mitigate those issues.

2 Related Work

In terms of incorporating high-level features, existing works have tried to extract concepts from the internal representations of models. For example, [5] find the vector that maximally separates two concepts suggested by humans at some level of the neural net and then measure the derivative along that vector of the final prediction for interpretability. In [1], for each image in the Broden dataset and each convolutional unit, the authors take the map of activations of the unit and match it with the mask pixel-wise annotation from the dataset, to check if representations at different layers disentangle different categories of meaning. Most closely related to our approach is [3], in which the network is forced to learn to segment the raw scan into a tissue map before making a final diagnosis.

3 Dataset

The CUB-200-2011 dataset [7] contains 11,788 images of birds across 200 categories in total. The list of species were obtained using an online field guide, images were harvested using Flickr and annotated by Mechanical Turk workers. There is an official train test split that people often use for benchmarking, which yields 5794 test images and 5994 train images, 20% of which are used for validation in this project, thus training and validation set sizes are 4796 and 1198 respectively.

The dataset comes with 312 binary attributes, all are visual in nature, mostly involving color, pattern of shape of a body part (e.g. attribute belly color group contains 15 different color choices). The ratio of positive labels to negative labels is $1 : 9$, which may not be a concern when attributes are used in conjunction with the images for predicting class labels. However, when class labels are used to predict attributes (details can be found in section 4.2.3), we encounter a class imbalance learning problem and thus experiment with weighting the gradient updates to correspond to the class ratio.

Besides, each attribute also includes indicator for one of four certainty levels: 1 (not visible), 2 (guessing), 3 (probably), 4 (definitely). If we consider categories 2 and 3 to be 'uncertain' then on average attributes for all images are 34.4% uncertain. Here is a more detailed breakdown of the certainty level distribution in the training set: not visible - 11%, guessing - 7% , probably - 28%, definitely - 54%, and among the positive attribute labels: not visible - 0%, guessing - 8% , probably - 34%, definitely - 58%. This means that the binary label (present/ not present) in itself may be noisy, affecting how effective they can be as a source of high-level supervision.

4 Methods

Whenever images are involved in training, we use InceptionV3 model [6], which has yielded state-of-the-art results on many fine-grained classification tasks and visual recognition ones in general. Because of the relatively limited official training data, the InceptionV3 model has been pretrained on ImageNet and all layers are then fine-tuned for the tasks of interest with a reduced learning rate. L2 regularization is used to prevent overfitting in all models.

4.1 Baselines

A natural baseline is using only bird images to train an InceptionV3 model [6], demonstrating the extent to which using visual cues alone can help distinguish among 200 classes of birds, without the need for domain knowledge of the birds' parts.

Another baseline is classification using only attribute labels for each image, fed through a feed-forward neural network with no hidden layer. This helps us gauge how informative the attributes are by themselves in determining the final class of birds. We experiment with adding hidden layers and more non-linear activations (and increase regularization accordingly) but don't obtain any improvement in performance.

4.2 Training With Attributes

Attributes might not be enough. It's probable that the list of human-generated attributes doesn't fully capture all of the information in the raw input, and this is especially true in our case when human labelers may not necessarily be experts in recognizing 200 classes of birds. Recognizing the importance of images as a rich source of information, we design two training algorithms that leverage both images and attributes, either in the same stage or at different stages.

4.2.1 Cotraining

Cotraining is adding an auxiliary task of predicting the attributes from the input images, in addition to predicting the bird class. Thus, in total the network is supposed to output $20+312 = 332$ predictions, 20 logits for the 20 bird classes (from which we will do a softmax), and 312 sigmoid ouputs for the 312 binary attributes. This is motivated by the fact that these attributes are relevant to the main task and thus learning them at the same time will reinforce useful representations within the network.

4.2.2 Bottleneck

To reduce the chance of the network picking up random correlations between image pixels and bird classes, we force it to use images to predict the binary attributes first. Then, the logit predictions for attributes are fed into the second baseline model as described in section 4.1, to output the final class of birds. The two stages are trained separately. This procedure may incur considerable errors, especially if the two models are not perfect at their respective tasks, but we hope that by forcing the network to learn intermediate high-level features first, this will offer a more causal explanation of how it determines the final class from the input images. Furthermore, another potential issue is that the first stage may suffer from class imbalance problem, as we have observed during training, and reweighting the gradient updates (i.e. updates by positive-labelled attributes are increased by 9 times) is found to offer improvement in overall accuracy.

4.2.3 Learning Curve

To see how data-efficient the different models are, we experiment with randomly removing 25%, 50% and 75% of the training data and train with a smaller dataset. We hypothesize that models that incorporate high-level attributes will be more robust to this reduction in information and generalize better, as compared to those that use only images.

5 Experiments

5.1 Setup

We use SGD optimizer with a starting learning rate of 0.0001, which is reduced by 10 folds if the validation loss stops improving for 10 epochs. We train all models for 1000 epochs and pick the best one based on validation accuracy. A batch size of 64 is used for most models. We also tune the L2 regularization coefficient as a hyperparameter, especially for models with more parameters such as the ones used for cotraining and bottleneck.

5.2 Results

Amount of data	Simple Finetune	Cotraining	Bottleneck	Only Attributes
100%	73.3%	73.5%	4.86%*	47.5%
75%	68.7%	69.4%	0.604%	44.1%
50%	61.9%	60.8%	0.570%	40.3%
25%	40.4%	41.3%	0.777%	28.3%

Table 1: How accuracy is affected with varying amount of data (*=weighted loss)

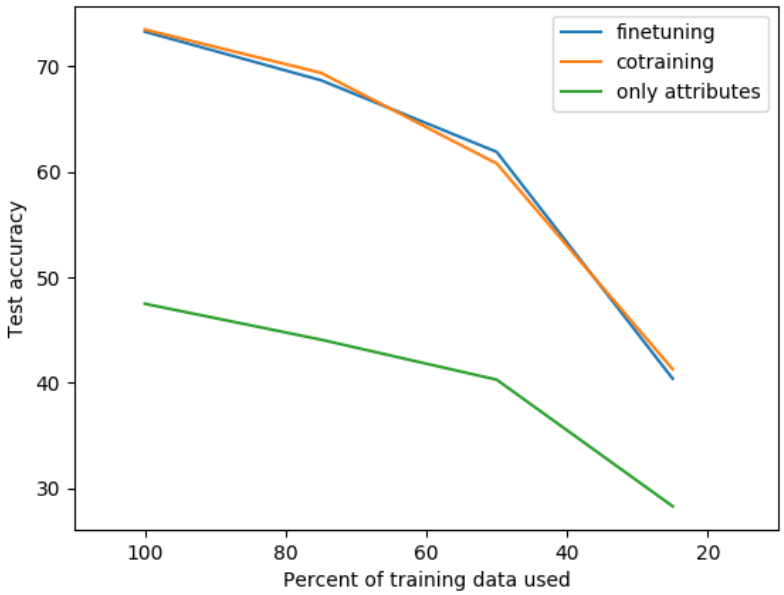

Figure 1: Learning curve of different training methods when data is a constraint (refer to table 1 for specific accuracy numbers)

Bottleneck method, while performing better than random guessing ($100 * 1/200 = 0.5\%$), still lags far behind compared to the rest of the methods. This also shows that information from raw images is still necessary for this task.

As seen from Figure 1, performance of cotraining in general shows less sharp of a decline than that of simple finetuning as amount of training data becomes increasingly limited. Training with only attributes, though yielding a lower performance that the other two methods, seems to be the most data-efficient. More work can be done on raising the performance of using only attributes, such as through adaptive feature selection for each class, but overall we expect that this method will be more robust to small training sets since it's less likely to pick up arbitrary pixel correlations that others using raw images are susceptible to.

Cotraining	Bottleneck	Only Attributes
73.6%	3.56%	43.0%

Table 2: Accuracy when certainty-calibrated attribute labels are used

From table 2, it seems that converting binary labels to certainty-calibrated labels doesn't help either, it may be better to just remove the most noisy labels from the supervision source.

6 Conclusion

This project explores the use of high-level attributes in fine-grained classification, a task that proves challenging to humans on average and thus naturally relies on domain knowledge. Being more of an exploratory project, we are left with many related questions to answer, one of which is whether objective ground-truth attributes are necessary for high-level supervision to bring about significant benefits that outweight the additional data collection. Secondly, in this domain of computer vision, does the attribute need to be localized? More output inspection and interpretation is needed to make sure that the model can pinpoint the location of the relevant attributes in the input image and not just guessing based on correlation with class labels.

Future work would include replicating the experiments on another bigger dataset, hopefully in another domain such as medical imaging, in addition to exploring more alternatives to denoising attributes, such as simply removing attributes with more than 40% of uncertainty. In addition, it would be interesting to look at model efficiency. Since InceptionV3 has a fairly complex architecture with more than 300 layers and 23M parameters, training on raw images alone already yields a sufficiently good performance for 200-class classification. In application scenarios where there are constraints on storage and inference speed, we may want to make use of a simpler model with fewer parameters. We hope that using attributes will help to close the performance gap of such models with the state-of-the-art.

Code and experiments can be found at https://github.com/thaonguyen19/CUB_supervision.

References

[1] David Bau, Bolei Zhou, Aditya Khosla, Aude Oliva, and Antonio Torralba. Network dissection: Quantifying interpretability of deep visual representations. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6541–6549, 2017.

[2] Wieland Brendel and Matthias Bethge. Approximating cnns with bag-of-local-features models works surprisingly well on imagenet. *arXiv preprint arXiv:1904.00760*, 2019.

[3] Jeffrey De Fauw, Joseph R Ledsam, Bernardino Romera-Paredes, Stanislav Nikolov, Nenad Tomasev, Sam Blackwell, Harry Askham, Xavier Glorot, Brendan O'Donoghue, Daniel Visentin, et al. Clinically applicable deep learning for diagnosis and referral in retinal disease. *Nature medicine*, 24(9):1342, 2018.

[4] Robert Geirhos, Patricia Rubisch, Claudio Michaelis, Matthias Bethge, Felix A Wichmann, and Wieland Brendel. Imagenet-trained cnns are biased towards texture; increasing shape bias improves accuracy and robustness. *arXiv preprint arXiv:1811.12231*, 2018.

[5] Been Kim, Martin Wattenberg, Justin Gilmer, Carrie Cai, James Wexler, Fernanda Viegas, and Rory Sayres. Interpretability beyond feature attribution: Quantitative testing with concept activation vectors (tcav). *arXiv preprint arXiv:1711.11279*, 2017.

[6] Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jon Shlens, and Zbigniew Wojna. Rethinking the inception architecture for computer vision. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2818–2826, 2016.

[7] C. Wah, S. Branson, P. Welinder, P. Perona, and S. Belongie. The Caltech-UCSD Birds-200-2011 Dataset. Technical report, 2011.