Adapting Attributes by Selecting Features Similar across Domains

Siqi Liu Adriana Kovashka
University of Pittsburgh
siqiliu, kovashka@cs.pitt.edu

Abstract

Attributes are semantic visual properties shared by objects. They have been shown to improve object recognition and to enhance content-based image search. While attributes are expected to cover multiple categories, e.g. *a dalmatian and a whale can both have "smooth skin", we find that the appearance of a single attribute varies quite a bit across categories. Thus, an attribute model learned on one category may not be usable on another category. We show how to adapt attribute models towards new categories. We ensure that positive transfer can occur between a source domain of categories and a novel target domain, by learning in a feature subspace found by feature selection where the data distributions of the domains are similar. We demonstrate that when data from the novel domain is limited, regularizing attribute models for that novel domain with models trained on an auxiliary domain (via Adaptive SVM) improves the accuracy of attribute prediction.*

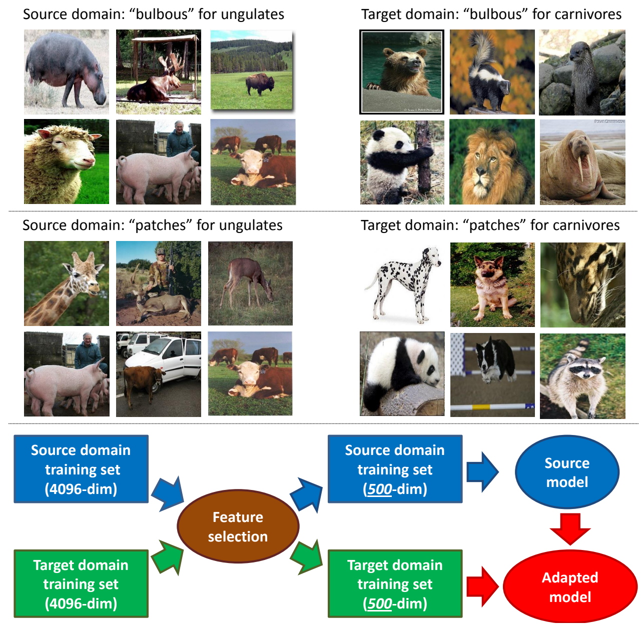

Figure 1: Examples of large discrepancy in attribute appearance between domains, and an overview of our approach.

1. Introduction

Attributes are semantic visual properties of the world which, similarly to adjectives, describe concepts that multiple object categories share. For example, animals might be "furry", a person might be "smiling", and a landscape might be "natural". Attributes have been successfully used to provide a rich and meaningful representation for recognition [9, 20] and image search [19, 17]. Since attributes are shared across categories, they are particularly useful when data is scarce. This is because learning a model for one attribute affects all the category models for objects that possess the attribute. Therefore, learning attribute models is a good investment for object recognition in the case of limited data [9, 18]. Further, attributes are useful since we can never have exhaustive visual data when learning about the world, yet when we encounter unfamiliar objects, it is useful to be able to say something about them, even if our system does not know their names.

One very appealing application of attributes is zero-shot learning, where a computer vision system can learn to automatically categorize a new previously unseen class without any training data, based on a textual attribute-based description. The system requires models for the attributes involved in the description, but they can be learned on data from other classes. For example, we can learn to recognize a polar bear by knowing that it is "white", "furry", "bulbous" *etc.*, after having learned models for these attributes from images of other animals [20]. Thus, zero-shot learning allows a vision system to generalize beyond what it has seen.

In order for applications such as zero-shot recognition and attribute-based search to work, an attribute must have the same semantic meaning and representation across different categories. In other words, we should be able to learn "white", "furry" and "bulbous" on one set of classes, and apply the learned models to another set with accuracy similar to what it would be on the first set. Otherwise, the attribute models we learn do not actually capture the concept behind the attribute name that humans associate with the name. For example, a model for "furry" might capture a concept that is just correlated with the attribute in question, such as the image background. This would limit the success of zero-shot learning since the latter depends on human-provided textual descriptions.

Consider the images shown in Figure 1. The figure shows positive images for the "bulbous" attribute for two domains of animals, ungulates and carnivores. The ungulate "bulbous" images look quite different than the carnivore ones. The right-hand side shows animals with black and white patches, as well as furry ones. Therefore, even to a human it may not be clear what the left and right-hand side have in common. The same holds for the "patches" attribute: the patches on pandas and dalmatians look very different from the giraffe patches. This variation indicates we cannot expect an attribute model learned on one domain of classes to perform well on another. Yet if we could efficiently *adapt* existing attribute models to new potentially very different domains, we can successfully and quickly learn attributes across different domains.

Thus, domain adaptation is an important aid in attribute learning. Yet attribute adaptation has largely been ignored in the vision community. The few exceptions rely on the existence of a generic model which is not be available in our case [16], do not consider the case of zero overlap between the categories in the source and target domains [7], or use a complex method which we find does not perform well in experiments [12].

We study the adaptability of attributes across domains, and propose a method for efficient adaptation. An overview of the method is shown in the lower part of Figure 1. We wish to learn a model of an attribute for a novel domain (called the "target" domain) with limited data from that domain, and a model trained on an existing domain (called "source" or "auxiliary"). For example, we can learn a model of "bulbous" on sea animals with (1) limited images of sea animals and (2) a "bulbous" model on land animals. We develop techniques for making adaptation more feasible, by selecting features that are both similar across domains.

Through experiments, we found that adapting a model trained on the source domain to the target domain through Adaptive SVM [31] has an advantage over directly applying a model trained only on the source domain. Feature selection helps improve the performance further so that it outperforms a model trained only on the target domain when the training data is scarce.

2. Related Work

Attributes and attribute learning. Attributes [20, 9, 4, 24] are semantic visual properties of objects, *e.g*. "furry", "shiny", "wooden", etc. Attributes allow recognition to go beyond labeling to describing objects. For example, when a robot encounters a new object, even if it cannot put the correct label on it, it can say "It looks like an airplane, but I don't see the wing," as suggested in [9]. Attributes also enable recognition of new objects from a textual description [20, 9, 22, 14]. Some applications of attributes have been for search [19, 17], actively learning object categories [23, 18], recognition of scenes [24] and unusual objects [26]. Attributes allow more efficient learning of category models because they are shared across categories, so multiple categories are impacted when an attribute models is learned [18].

In most of the above works (except those which perform zero-shot recognition such as [9, 20, 22, 14]), attribute models are learned and used on the same dataset, usually using data from all available object classes. However, since attributes are semantic properties, they should be learnable across object category boundaries. If a computer vision system truly understood the meaning of "furry", we should be able to learn a model of "furry" on one domain of objects, and apply it on another. Farhadi *et al*. [9] and Jayaraman *et al*. [14] show how to learn more accurate models that avoid learning data artifacts that are correlated with the attribute, in order to improve the accuracy of attribute models when a model is applied on a different dataset. In contrast, we examine how we can adapt attribute models trained on one domain to successfully work on another.

Domain adaptation and transfer learning. Transfer learning [2, 28] involves applying an existing model in the learning of a related category. For instance, given a model of "motorcycle" and just a small number of images for "bicycle", we can learn about as good a model as with a large number of images for "bicycle" [2]. Similarly, domain adaptation [31, 11, 10, 13, 1, 32, 5, 30] allows one to learn an accurate model of a concept adapted to the novel target domain, in cases where simply applying the model learned on the source domain does not suffice. For example, [10] consider accurately recognizing monitors "in the wild", after learning a model for "monitor" from clean product images on Amazon.com. Supervised domain adaptation transforms an existing model using a small number of labeled examples from the target category. Unsupervised adaptation uses unlabeled examples from the target domain and finds a feature space where the two domains are similar, then learns a model in that space.

Transfer learning and adaptation aim to optimize how we use small amounts of data in new domains or of new categories. This aim of making smart use of limited data is in the same vein as efficient attribute-based object category learning. It seems natural to combine these methods, yet we are aware of only three works that study attribute adaptation. One is the attribute personalization approach of Kovashka and Grauman [16] which learns a generic model of an attribute from the crowd and adapts it towards individual search users. In other words, they learn a "least common denominator" model for an attribute such as "formal shoes", and adapt it towards each particular user's notion of this concept. In contrast to [16], our approach does not assume a generic model is available, so instead of producing a specialized model from an "umbrella" model, it takes one model and adapts it towards a category that is potentially visually very different.

Another relevant work is Han *et al.*'s Image Attribute Adaptation (IAA) approach [12]. They propose to learn a shared multi-kernel representation for the source and target domains by minimizing their discrepancy. Assuming images close in the feature space possess similar attributes, they use local linear regression to predict the attributes of unlabeled training images in the target domain. Finally, multiple kernel regression with an $\ell_{2,p}$-norm loss function is learned to predict the attributes of testing images. While IAA tries to bring closer the images in the two domains by learning a multi-kernel representation, our method tries to bring them closer through feature selection. IAA learns multiple attributes together, but our method learns them separately. After mapping the original feature space to the multi-kernel feature space, IAA learns one model for the source and target domain. In contrast, our method learns two different models for the source and target domain while keeping them as close as possible through regularization. For practical use, IAA is very complex: besides different choices of base kernels that can be made, there are four trade-off parameters for the objective function and one parameter for the norm. In contrast, our approach is significantly simpler (only one parameter needs to be set when using a linear kernel). Further, as we show in experiments, our method produces better results.

In a recent work, Chen *et al.* [7] also adapt attributes across domains. However, the different domains in their case are clothing items shown in clean product images as opposed to a less restricted street environment. Yet in [7], the categories of clothing items are still the same (e.g., "plaid shirt"). In contrast, we consider the case of disjoint categories between the source and target domains. The problem is more challenging because when we learn an attribute on a category such as "bear" and try to predict it on "dolphin", the visual appearance shift is much larger than in plaid shirts on Amazon as opposed to plaid shirts on Facebook. Further, [7] use an expensive deep learning formulation, and we use simple linear kernel adaptation.

Feature selection. Researchers have studied how to ensure that the features used for various vision tasks are informative [3, 33, 21]. [25, 8] show how feature selection improves recognition performance. [27, 29, 6] exploit feature selection for domain adaptation.

In contrast to prior work, our approach uses a much simpler and less expensive criterion for feature selection. However, any method that finds a feature subspace where the two domains are close can be used in our formulation. Our contribution is the integration of feature selection in an adaptive SVM, for the problem of learning attributes across domains. We show that our adaptation method predicts attributes on novel domains more accurately than alternative methods.

3. Approach

We assume we are given two domains of the same high-level concept. For example, this could be male and female humans, ungulates and carnivores, *etc*. In all cases, we learn models for each attribute individually. We assume a scenario where we are given a classifier for an attribute A and domain S, and want to efficiently learn a classifier for domain T. This is useful because it matches a situation when we have learned an attribute on some categories, and we want to use this attribute model on a visually and semantically different set of categories. To adapt an existing classifier, we will use the Adaptive SVM framework of [31], but only using a subset of the features which we will select automatically. We provide details for both the adaptation and feature selection next.

3.1. Adapting the classifier

In [31], the authors propose Adaptive SVM (A-SVM). A-SVM is a modification of an SVM formulation that allows one to learn a new classifier while regularizing with an existing one.

Let $\boldsymbol{x}_i$ denote the feature representation of image i and y_i denote its binary label (1 or -1) which says whether this image contains the attribute of interest. Let $D_T = \{\boldsymbol{x}_1, y_1\}, \dots, \{\boldsymbol{x}_{N_T}, y_{N_T}\}$ denote the data in the target domain, which consists of N_T samples. Let $D_S = \{\boldsymbol{x}_1^S, y_1^S\}, \dots, \{\boldsymbol{x}_{N_S}^S, y_{N_S}^S\}$ denote the data in the source domain, where N_S is the number of samples. Also, let $f_s(\boldsymbol{x})$ denote a model learned on the source domain. Let $f_\Delta(\boldsymbol{x}) = \boldsymbol{w}_\Delta^T \boldsymbol{x}$ denote the "delta" or change function between the source model and its transformed version which results from adapting it with the target data, so the *adapted* classifier is $f(\boldsymbol{x}) = f_s(\boldsymbol{x}) + f_\Delta(\boldsymbol{x})$, which can be learned using the source classifier and data in the target domain, as:

$$\min_{\boldsymbol{w}_\Delta} \quad \frac{1}{2}\|\boldsymbol{w}_\Delta\|^2 + C\sum_{i=1}^{N_T}\xi_i \tag{1}$$
$$\text{subject to} \quad y_i f_s(\boldsymbol{x}_i) + y_i \boldsymbol{w}_\Delta^T \boldsymbol{x}_i \geq 1 - \xi_i,$$
$$\xi_i \geq 0, \quad \forall i = 1, \dots, N_T$$

Note that in this formulation, $\boldsymbol{w}_\Delta$ denotes the change of the parameters, hence the deviation of the adapted model from the source model. In other words, the new adapted model is learned on the target domain, but it is encouraged to stay close to the model for the source domain. This regularization with the source model is useful since the target domain data is scarce and learning a model from this data alone can be unreliable.

To learn $f_s(\boldsymbol{x})$ in the source domain, we use the standard SVM formulation, which is similar to Eq. (1) but does not take an existing model into account:

$$\begin{aligned} \min_{\boldsymbol{w}} \quad & \frac{1}{2}\|\boldsymbol{w}\|^2 + C\sum_{i=1}^{N_S} \xi_i \\ \text{subject to} \quad & y_i \boldsymbol{w}^T \boldsymbol{x}_i^S \geq 1 - \xi_i, \\ & \xi_i \geq 0, \quad \forall i = 1, \ldots, N_S \end{aligned} \tag{2}$$

We use the linear A-SVM implementation of [31], but a kernelized version of A-SVM is also possible.

Yang *et al*. [31] show that A-SVM allows faster learning of concepts for a new domain as compared to starting from scratch or directly applying a classifier from the old domain. We wish to determine whether this holds in the domain of attributes, *i.e*., whether attributes are adaptable across domain boundaries. Note that the formulation in Equation (1) makes the assumption that the two domains are similar enough that positive transfer can occur between them. This assumption is valid in the case of learning TV-related concepts such as "weather", as in [31], but it may not hold in the case of learning attributes across different types of animals. Therefore, we need to ensure that positive transfer can occur between domains. We discuss this next.

3.2. Feature selection

In general, different domains are not guaranteed to be similar. If they are too dissimilar, regularizing the classifier for one domain with that learned on another would be counter-productive. To overcome this problem, we wish to bring the source and target domains closer, by performing a form of feature selection.

We only use those features for learning the new classifier that are similar across domains. We do this separately for those images in the source and target domains that contain attribute A, namely D_S^+ and D_T^+, and for the images that do not contain them, D_S^- and D_T^-. More formally, let j denote a feature dimension and $x_{i,j}$ denote the j-th component of $\boldsymbol{x}_i$. Let

$$\begin{aligned} s_j^+ &= \frac{|\mu(j, D_S^+) - \mu(j, D_T^+)|^2}{\sigma^2(j, D_S^+) + \sigma^2(j, D_T^+)} \\ s_j^- &= \frac{|\mu(j, D_S^-) - \mu(j, D_T^-)|^2}{\sigma^2(j, D_S^-) + \sigma^2(j, D_T^-)} \end{aligned} \tag{3}$$

where

$$\mu(j, D_S^+) = \frac{1}{|D_S^+|} \sum_{\boldsymbol{x}_i \in D_S^+} x_{i,j}$$

$$\sigma^2(j, D_S^+) = \frac{1}{|D_S^+|} \sum_{\boldsymbol{x}_i \in D_S^+} (x_{i,j} - \mu(j, D_S^+))^2$$

and other values of μ and σ^2 are defined similarly.

We rank all feature dimensions j with respect to their scores s_j^+ and s_j^- separately, in ascending order, average the ranks, and pick the $K = 500$ feature dimensions with smallest ranks. Thus, we are picking the feature dimensions that are most similar between the two domains.

Then $\boldsymbol{x}_i$ in Eq. (1) and (2) becomes

$$(x_{i,l(1)}, x_{i,l(2)}, \ldots, x_{i,l(K)})$$

where $l(r), r = 1, \ldots, K$ are the original indices of the K feature dimensions most similar across the two domains. In other words, the adapted model is now trained with lower-dimensional target data and regularized with a source model trained on lower-dimensional source data.

In addition to the feature spaces being similar, we also explore another useful criterion for feature selection: feature discriminativity. We want to pick features that are discriminative in both the source and target domains, hence differ between the positive and negative data:

$$\begin{aligned} d_j^S &= -\frac{|\mu(j, D_S^+) - \mu(j, D_S^-)|^2}{\sigma^2(j, D_S^+) + \sigma^2(j, D_S^-)} \\ d_j^T &= -\frac{|\mu(j, D_T^+) - \mu(j, D_T^-)|^2}{\sigma^2(j, D_T^+) + \sigma^2(j, D_T^-)} \end{aligned} \tag{4}$$

We can use the feature similarity and discriminativity criteria separately, or we can combine the two criteria and pick those features for which the average rank computed from the ranks based on s_j^+, s_j^-, d_j^S and d_j^T is smallest.

In the next section, we show how these feature selection criteria help attribute adaptation succeed.

4. Experimental Validation

We first describe the data and features we use and the baselines against which we compare our approach. We then present our experimental findings.

4.1. Dataset

We use the Animals with Attributes dataset of Lampert *et al*. [20]. Within it, we select two domains, which constitute two animal orders: Artiodactyla (ungulates or hoofed animals) and Carnivora (carnivores). The former consists of 10 species, while the latter consists of 21 species. For sampling convenience we omit the last species, raccoon, from the latter. We treat the former as the source and the latter as the target domain.

We randomly sample $N_S = 1000$ images from the source domain and $N_T = \{20, 40, 60, 80, 100\}$ images from the target domain for training, and another 1000 images from the target domain for testing. The same number of images are sampled from each species. We repeat each experiment 10 times with different training and testing sets, and report the average results over all trials.

Attribute	Source	Target
black	0.5	0.65
white	0.4	0.5
brown	0.7	0.7
gray	0.3	0.4
patches	0.4	0.3
spots	0.4	0.25
bulbous	0.7	0.4
lean	0.3	0.65
smelly	0.7	0.4
muscle	0.7	0.55
forager	0.5	0.5
mountains	0.4	0.25
domestic	0.4	0.4

Table 1: Attributes and their distributions on the source and target domains. The value shown denotes what fraction of species in the corresponding domain contain the attribute.

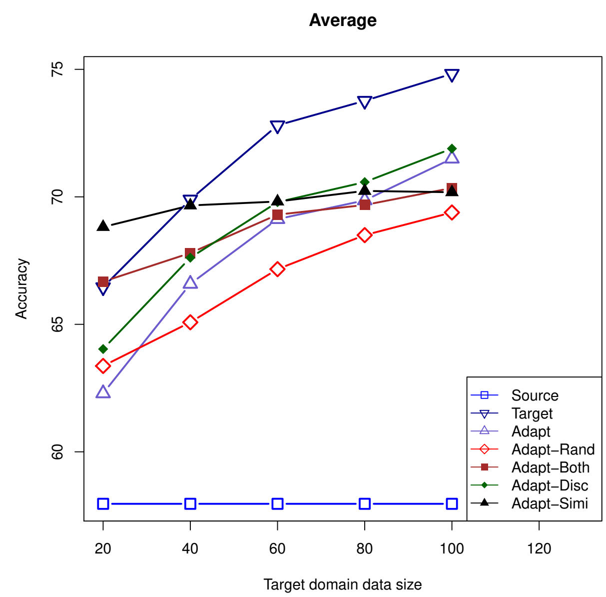

Figure 2: Attribute prediction accuracy averaged over all 13 attributes.

To make experiments more meaningful, we only conduct experiments using the 13 attributes that occur in at least 25 percent and no more than 75 percent of the species in both domains. These attributes are shown in Table 1.

4.2. Features

We use the DECAF features provided with the dataset released by [20]. These features correspond to the *fc7* layer of CaffeNet trained on images from the ImageNet 2012 challenge. These features can be extracted on novel images using the Caffe package from Berkeley [15]. Each instance is a 4096-dimensional vector with a unit ℓ_2-norm.

4.3. Baselines

We compare our approach to three baselines:

- SOURCE, a model trained with 1000 data points on the source domain using the original 4096-dimensional CNN features;
- TARGET, trained using between 20 and 100 data points on the target domain using the original 4096-dimensional CNN features; and
- IAA, the attribute adaptation approach of [12], a multiple kernel model trained with data points on both the source and target domain using the 4096-dimensional CNN features.

We examine several versions of our method:

- ADAPT, a 4096-dimensional model which directly applies A-SVM as in Section 3.1, without applying any feature selection;
- ADAPT-SIMI, a 500-dimensional model which applies feature selection as in Eq. (3);
- ADAPT-DISC, a 500-dimensional model which applies feature selection as in Eq. (4);
- ADAPT-BOTH, a 500-dimensional model which applies both similarity and discriminativity for feature selection, as in Eq. (3) and (4); and
- ADAPT-RAND, a 500-dimensional model which selects features randomly.

The parameters C in our models are tuned in the range $\{2^{-15}, 2^{-12}, \ldots, 1, \ldots, 2^{12}, 2^{15}\}$ with five-fold cross-validation. For IAA, we tried different parameter settings as in [12]: α and μ were varied in the range $\{10^{-4}, 10^{-2}, 1, 10^2, 10^4\}$; and p in $\{0.5, 1.0, 1.5\}$. For IAA only, instead of the time-consuming cross-validation, we report the best results we can get among all the parameter settings considered in the original paper. Thus, our results on IAA are generous to this baseline.

4.4. Results

Figure 2 shows the attribute prediction performance of the linear SVM methods (all methods in Section 4.3 except IAA), averaged over all 13 attributes. Again, all of these methods are tested on data from the target domain. We observe that the model trained on source data performs quite poorly, which indicates that the source and target domains are quite distinct. We also observe that the naive adaptation method (ADAPT) fails to outperform a method trained with just a small number of data points on the target data (TARGET). In other words, the source and target domains are so different that on average, with straight-forward adaptation techniques, there is no benefit of using information

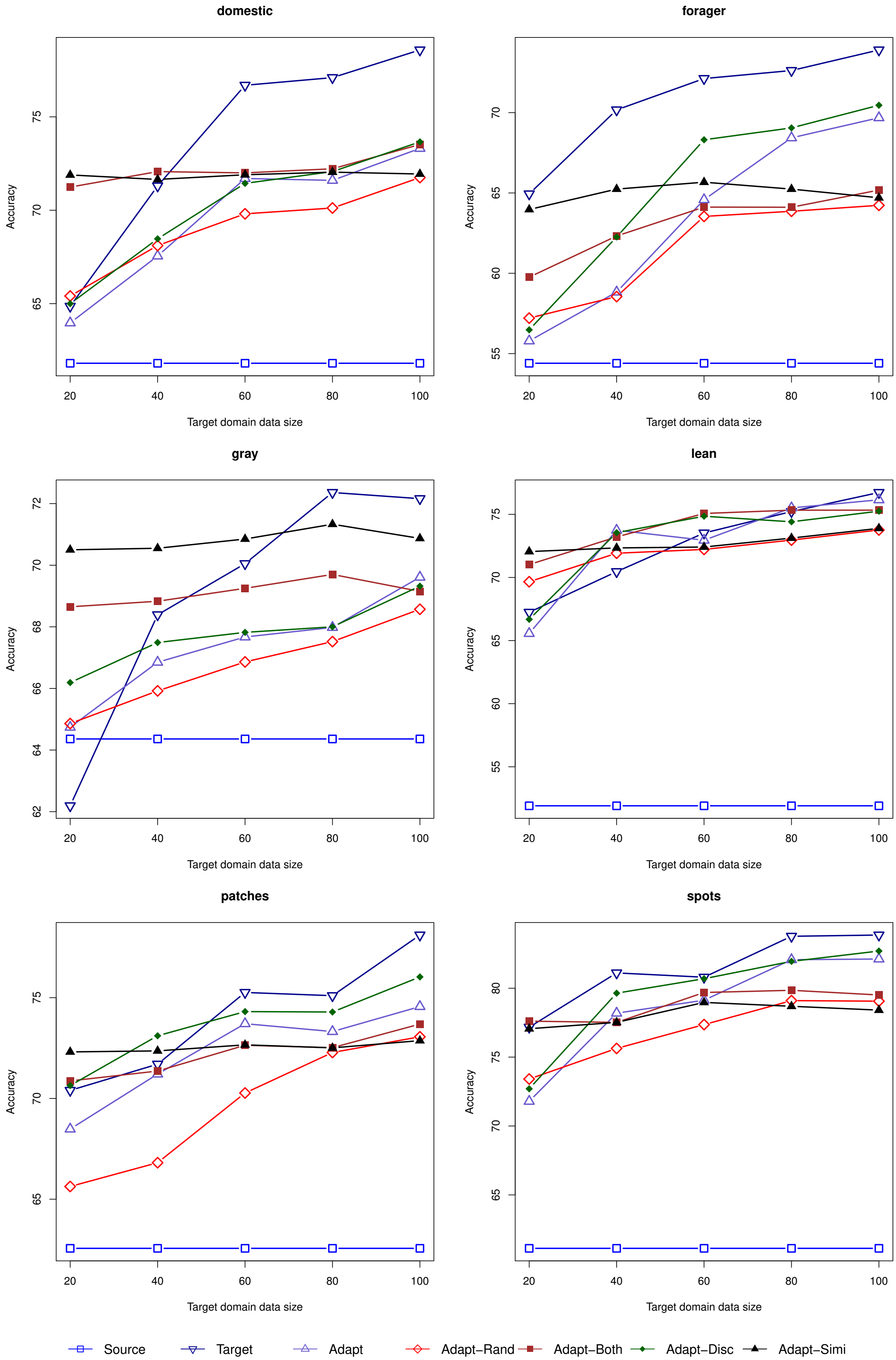

Figure 3: Attribute prediction results for a subset of the attributes.

Attribute	SOURCE	TARGET	ADAPT-SIMI
black	52.0	65.1	65.4
brown	63.0	70.2	71.4
bulbous	55.3	61.7	66.3
mountains	64.9	76.3	74.1
muscle	62.1	59.4	67.0
smelly	38.7	59.6	54.5
white	61.4	64.9	68.0

Table 2: Accuracy on the attributes not shown in Figure 3, using 20 training samples on the target domain.

from the source domain for regularization.

The feature selection methods ADAPT-DISC and ADAPT-SIMI and their combination ADAPT-BOTH all improve upon the naive adaptation method (ADAPT) when a small amount of data if available. However, the ADAPT-SIMI method is best. This indicates that the similarity of feature spaces between the two domains is more important for adaptation than feature discriminativity. As a sanity check, the 500-dimensional model which uses random feature selection (ADAPT-RAND) fails to improve upon direct adaptation, except at 20 training instances.

We developed the attribute adaptation method for situations when data from the target domain is scarce. Therefore, we are most interested in the left-most part of Figure 2, *i.e*. the case when only 20 data samples from the target domain are available for training. ADAPT-SIMI, the method which selects features such that they are similar between the two domains, achieves the best results when little target data is available, and it outperforms the model trained on target data only (TARGET). When more data becomes available, feature discriminativity (ADAPT-DISC) and having data from the domain of interest (TARGET) become more important. In other words, when sufficient data is available, one should directly learn on data from the domain of interest, but *when data from the target domain is limited, the best strategy is to adapt a source model, after selecting features that are common between the two domains.*

In Figure 3, we include a representative subset of the methods' performance on individual attributes. We have several findings.

- SOURCE performs worst in almost all cases, confirming that there is a large discrepancy between the source and target domains.
- For some attributes, *e.g*. "forager", TARGET performs the best even when the training data set is small. The reason might be that the source and target domains are so different that it is almost impossible to get any transferable knowledge from the source domain. For other attributes, *e.g*. "gray", even at 60 target data points it is preferable to adapt from a source domain.

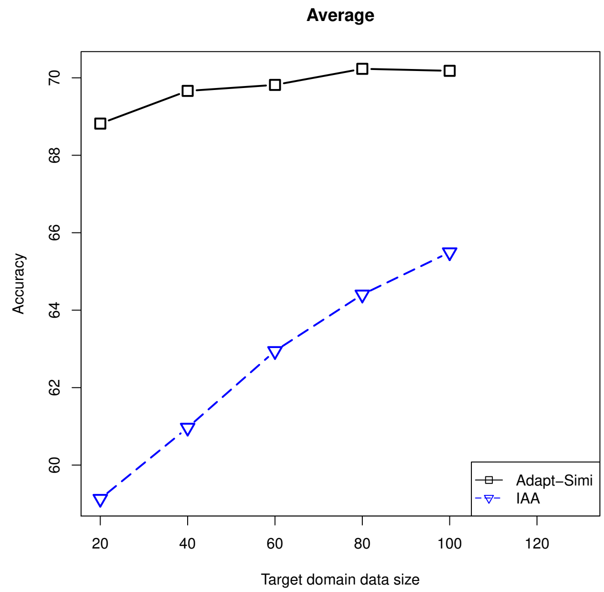

Figure 4: Comparison to another attribute adaptation method, [12].

- ADAPT-DISC outperforms ADAPT in most cases, which in turn outperforms ADAPT-RAND, meaning that feature selection helps.
- ADAPT-SIMI's performance increases slower than ADAPT-DISC's as the target domain data size increases, because the former learns the common part between the two domains, and adding more target domain data mostly provides more variations in the target domain, leaving the common part unchanged.

Due to the space limitation, we only show the more interesting per-attribute curves in Figure 3. The performance of ADAPT-SIMI, TARGET and SOURCE on the remaining attributes is shown in Table 2. We conclude that for nine of the 13 attributes, ADAPT-SIMI outperforms TARGET when little data is available. The three where we underperform it are all attributes that are not visual ("forager", "mountains", "smelly") so it is hard to learn any model for those.

Figure 4 shows the comparison of our approach ADAPT-SIMI to the IAA approach of [12], averaged over all attributes. We see that our method learns significantly more accurate attribute models than IAA.

Considering the large amount of attributes we can learn in practice, it is important to be able to learn from a small labeled dataset. On a dataset of 100 attributes, getting 100 labels per attribute using a majority vote over 5 labels acquired on MTurk for 1 cent per label would cost $500. Based on the trend of the curves in all figures, the smaller the training sample, the larger our method's relative gain over other methods, demonstrating the practicality of our method in common situations of limited data.

5. Conclusion

We presented a method for adapting an attribute model learned on one domain, *e.g*. land animals, to be used on another domain, *e.g*. sea animals. Our method helps positive transfer between the source and target domains by selecting features that are similar between the two domains, thus finding a feature subspace where the two domains are close. We demonstrate that when the data in the domain of interest is scarce, and an attribute model on a different domain is available, it is best to utilize that model and adapt it for the new domain. In addition, we showed that our method achieves more accurate results than an alternative attribute adaptation method.

As future work, we plan to study how we can best visualize the attribute models we have learned. Further, we will consider how we can transfer knowledge between different attributes, including ones which may not seem to have a semantic connection.

References

[1] R. Aljundi, R. Emonet, D. Muselet, and M. Sebban. Landmarks-based kernelized subspace alignment for unsupervised domain adaptation. In *CVPR*, 2015. 2

[2] Y. Aytar and A. Zisserman. Tabula Rasa: Model Transfer for Object Category Detection. In *ICCV*, 2011. 2

[3] J. Bins and B. A. Draper. Feature Selection from Huge Feature Sets. In *ICCV*, 2001. 3

[4] S. Branson, C. Wah, F. Schroff, B. Babenko, P. Welinder, P. Perona, and S. Belongie. Visual Recognition with Humans in the Loop. In *ECCV*, 2010. 2

[5] R. Caseiro, J. F. Henriques, P. Martins, and J. Batista. Beyond the shortest path: Unsupervised domain adaptation by sampling subspaces along the spline flow. In *CVPR*, 2015. 2

[6] M. Chen, K. Q. Weinberger, and J. Blitzer. Co-training for domain adaptation. In *NIPS*, 2011. 3

[7] Q. Chen, J. Huang, R. Feris, L. M. Brown, J. Dong, and S. Yan. Deep domain adaptation for describing people based on fine-grained clothing attributes. In *CVPR*, 2015. 2, 3

[8] C. M. Christoudias, R. Urtasun, and T. Darrell. Unsupervised Feature Selection via Distributed Coding for Multi-view Object Recognition. In *CVPR*, 2008. 3

[9] A. Farhadi, I. Endres, D. Hoiem, and D. A. Forsyth. Describing Objects by Their Attributes. In *CVPR*, 2009. 1, 2

[10] B. Gong, Y. Shi, F. Sha, and K. Grauman. Geodesic Flow Kernel for Unsupervised Domain Adaptation. In *CVPR*, 2012. 2

[11] R. Gopalan, R. Li, and R. Chellapa. Domain Adaptation for Object Recognition: An Unsupervised Approach. In *ICCV*, 2011. 2

[12] Y. Han, Y. Yang, Z. Ma, H. Shen, N. Sebe, and X. Zhou. Image Attribute Adaptation. *IEE Transactions on Multimedia*, 16:1115–1126, June 2014. 2, 3, 5, 7

[13] J. Hoffman, B. Kulis, T. Darrell, and K. Saenko. Discovering Latent Domains for Multisource Domain Adaptation. In *ECCV*, 2012. 2

[14] D. Jayaraman, F. Sha, and K. Grauman. Decorrelating Semantic Visual Attributes by Resisting the Urge to Share. In *CVPR*, 2014. 2

[15] Y. Jia, E. Shelhamer, J. Donahue, S. Karayev, J. Long, R. Girshick, S. Guadarrama, and T. Darrell. Caffe: Convolutional architecture for fast feature embedding. In *ACM Multimedia*, pages 675–678, 2014. 5

[16] A. Kovashka and K. Grauman. Attribute Adaptation for Personalized Image Search. In *ICCV*, 2013. 2, 3

[17] A. Kovashka, D. Parikh, and K. Grauman. WhittleSearch: Interactive Image Search with Relative Attribute Feedback. *IJCV*, 115:185–210, 2015. 1, 2

[18] A. Kovashka, S. Vijayanarasimhan, and K. Grauman. Actively Selecting Annotations Among Objects and Attributes. In *ICCV*, 2011. 1, 2

[19] N. Kumar, A. C. Berg, P. N. Belhumeur, and S. K. Nayar. Describable Visual Attributes for Face Verification and Image Search. *TPAMI*, 33(10):1962–1977, 2011. 1, 2

[20] C. Lampert, H. Nickisch, and S. Harmeling. Learning to Detect Unseen Object Classes By Between-Class Attribute Transfer. In *CVPR*, 2009. 1, 2, 4, 5

[21] M. H. Nguyen and F. de la Torre. Optimal feature selection for support vector machines. *Pattern Recognition*, 2009. 3

[22] D. Parikh and K. Grauman. Relative Attributes. In *ICCV*, 2011. 2

[23] A. Parkash and D. Parikh. Attributes for Classifier Feedback. In *ECCV*, 2012. 2

[24] G. Patterson and J. Hays. SUN Attribute Database: Discovering, Annotating, and Recognizing Scene Attributes. In *CVPR*, 2012. 2

[25] H. Peng and F. Long. Feature Selection Based on Mutual Information: Criteria of Max-Dependency, Max-Relevance, and Min-Redundancy. *TPAMI*, 2005. 3

[26] B. Saleh, A. Farhadi, and A. Elgammal. Object-Centric Anomaly Detection by Attribute-Based Reasoning. In *CVPR*, 2013. 2

[27] S. Satpal and S. Sarawagi. Domain adaptation of conditional probability models via feature subsetting. In *KDD*, 2007. 3

[28] M. Stark, M. Goesele, and B. Schiele. A Shape-based Object Class Model for Knowledge Transfer. In *ICCV*, 2009. 2

[29] S. Uguroglu and J. Carbonell. Feature selection for transfer learning. In *Machine Learning and Knowledge Discovery in Databases*. Springer, 2011. 3

[30] J. Xu, S. Ramos, D. Vazquez, and A. M. Lopez. Domain adaptation of deformable part-based models. *TPAMI*, 36(12):2367–2380, 2014. 2

[31] J. Yang, R. Yan, and A. G. Hauptmann. Cross-Domain Video Concept Detection Using Adaptive SVMs. In *ACM Multimedia*, 2007. 2, 3, 4

[32] T. Yao, Y. Pan, C.-W. Ngo, H. Li, and T. Mei. Semi-supervised domain adaptation with subspace learning for visual recognition. In *CVPR*, 2015. 2

[33] L. Yu and H. Liu. Efficient Feature Selection via Analysis of Relevance and Redundancy. *JMLR*, 2004. 3