Asking Friendly Strangers: Non-Semantic Attribute Transfer

Nils Murrugarra-Llerena and **Adriana Kovashka**
Department of Computer Science
University of Pittsburgh
{nineil, kovashka}@cs.pitt.edu

Abstract

Attributes can be used to recognize unseen objects from a textual description. Their learning is oftentimes accomplished with a large amount of annotations, e.g. around 160k-180k, but what happens if for a given attribute, we do not have many annotations? The standard approach would be to perform transfer learning, where we use source models trained on other attributes, to learn a separate target attribute. However existing approaches only consider transfer from attributes in the same domain i.e. they perform *semantic* transfer between attributes that have related meaning. Instead, we propose to perform *non-semantic* transfer from attributes that may be in different domains, hence they have no semantic relation to the target attributes. We develop an attention-guided transfer architecture that learns how to weigh the available source attribute classifiers, and applies them to image features for the attribute name of interest, to make predictions for that attribute. We validate our approach on 272 attributes from five domains: animals, objects, scenes, shoes and textures. We show that semantically unrelated attributes provide knowledge that helps improve the accuracy of the target attribute of interest, more so than only allowing transfer from semantically related attributes.

Introduction

Semantic visual attributes have allowed researchers to recognize unseen objects based on textual descriptions (Lampert, Nickisch, and Harmeling 2009; Parikh and Grauman 2011; Akata et al. 2013), learn object models expediently by providing information about multiple object classes with each attribute label (Kovashka, Vijayanarasimhan, and Grauman 2011; Parkash and Parikh 2012), interactively recognize fine-grained object categories (Branson et al. 2010; Wah and Belongie 2013), and learn to retrieve images from precise human feedback (Kumar et al. 2011; Kovashka, Parikh, and Grauman 2015). Recent ConvNet approaches have shown how to learn accurate attribute models through multi-task learning (Fouhey, Gupta, and Zisserman 2016; Huang et al. 2015) or by localizing attributes (Xiao and Jae Lee 2015; Singh and Lee 2016). However, deep learning with ConvNets requires a large amount of data to be available for the task of interest, or for a *related* task (Oquab et al. 2014). What should we do if we have a limited amount of data for the task of interest, and no data from semantically related categories? For example, let us imagine we have an entirely new domain of objects (e.g. deep sea animals) which is visually distinct from other objects we have previously encountered, and we have very sparse labeled data on that domain. Let us assume we have plentiful data from unrelated domains, e.g. materials, clothing, and natural scenes. Could we still use that *unrelated* data?

We examine how we can transfer knowledge from attribute classifiers on unrelated domains, as shown in Fig. 1. For example, this might mean we want to learn a model for the animal attribute "hooved" from scene attribute "natural", texture attribute "woolen", etc. We define semantic transfer as learning a target attribute using the remaining attributes in that same data set as source models. This is the approach used in prior work (Chen and Grauman 2014; Liu and Kovashka 2016; Han et al. 2014). In contrast, in *non-semantic* transfer (our proposed approach), we use source attributes from *other* datasets. We show that allowing transfer from diverse datasets allows us to learn more accurate models, but *only when we intelligently select how to weigh the contribution of the source models*. The intuition behind our approach is that the same visual patterns recur in different realms of the visual world, but language has evolved in such a way that they receive different names depending on which domain of objects they occur in.

We propose an attention-guided transfer network. Briefly, our approach works as follows. First, the network receives training images for attributes in both the source and target domains. Second, it separately learns models for the attributes in each domain, and then measures how related each target domain classifier is to the classifiers in the source domains. Finally, it uses these measures of similarity (relatedness) to compute a weighted combination of the source classifiers, which then becomes the new classifier for the target attribute. We develop two methods, one where the target and source domains are disjoint, and another where there is some overlap between them. Importantly, we show that when the source attributes come from a diverse set of domains, the gain we obtain from this transfer of knowledge is greater than if only use attributes from the same domain.

We test our method on 272 attributes from five datasets of objects, animals, scenes, shoes, and textures, and compare

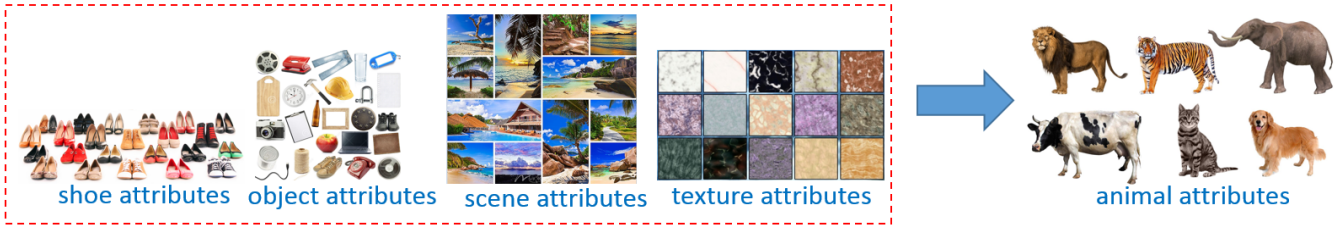

Figure 1: We study transfer of knowledge among disjoint domains. Can shoe, object, scene, and texture attributes be beneficial for learning *animal* attributes, despite the lack of semantic relation between the categories and attributes?

it with several baselines: learning using data from the target attribute only, transfer only from attributes in the same domain, uniform weighting of the source classifiers, learning an invariant representation through a confusion loss, and a fine-tuning approach. We also show qualitative results in the form of attention weights, which indicate what kind of information different target attributes borrowed.

While our target attributes come from well-defined and properly annotated datasets, our work demonstrates how non-semantic transfer can be used to learn attributes on novel domains where data is scarce, like the scenario discussed above. Our main contributions are an attention-guided transfer network, and a study of transferability of attributes across semantic boundaries.

The rest of this paper is organized as follows. Sec. 2 reviews related work on attribute learning, transfer learning, and attention networks. We describe our attention-guided transfer approach in Sec. 3. Sec. 4 shows our experimental evaluation, and we conclude in Sec. 5.

Related Work

Attribute learning. An image can portray more than one attribute, so it is natural to learn multiple attributes jointly. (Shao et al. 2015) employ multi-task learning to learn attributes for crowd scene understanding. (Fouhey, Gupta, and Zisserman 2016) recognize 3D shape attributes using a multi-label and an embedding loss. Another way for joint attribute learning is using a regularized hypergraph cut (Huang et al. 2015). Hypergraphs represent instances and can capture correlations of multiple relations (i.e. attributes).

Other approaches use localization for attribute learning. (Liu et al. 2015) learn binary face attributes using a localization component and an identity classifier followed by linear SVMs for binary attribute prediction. (Xiao and Jae Lee 2015) discover visual concepts in a sequence of attribute comparisons. (Singh and Lee 2016) improve the efficiency and accuracy of this method using a Siamese neural network with localization and ranking sub-nets.

Despite the success of end-to-end deep learning, many authors employ neural networks for feature extraction, and use these features in traditional machine learning frameworks. (Liang et al. 2015) learn a feature space using additional information from object categories, and (Gan, Yang, and Gong 2016) create category-invariant features that are helpful for attribute learning.

We study how to perform attribute transfer learning using multi-task neural networks.

Domain adaptation and transfer learning. Many researchers perform transfer learning via an invariant feature representation (Gan, Yang, and Gong 2016; Gong et al. 2012), e.g. by ensuring a network cannot distinguish between two domains in the learned feature space (Tzeng et al. 2015; Ganin and Lempitsky 2015; Long et al. 2016), training a network that can reconstruct the target domain (Ghifary et al. 2016; Kan, Shan, and Chen 2015; Bousmalis et al. 2016), through layer alignment (Chen et al. 2015) or shared layers that bridge different data modalities (Castrejon et al. 2016). Other methods (Yang, Yan, and Hauptmann 2007) perform transfer learning via parameter transfer where the source classifiers regularize the target one. (Tommasi, Orabona, and Caputo 2014) employ an adaptive least-squares SVM to transfer model parameters from source classifiers to a target domain.

Transfer learning for attributes. Prior work considers transfer learning between attributes of the same domain. (Chen and Grauman 2014) use tensor factorization to transfer object-specific attribute classifiers to unseen object-attribute pairs. (Han et al. 2014) learn a common feature space through maximum mean discrepancy and multiple kernels. (Liu and Kovashka 2016) select features from the source and target domains, and transfer knowledge using Adaptive SVM (Yang, Yan, and Hauptmann 2007) in this lower-dimensional space.

Some recent zero-shot learning work (Changpinyo et al. 2016; Xian et al. 2016; Yu and Aloimonos 2010) learns an underlying embedding space from the *seen* classes and some auxiliary information (e.g. text), and then queries this embedding with a sample belonging to a new *unseen* class, in order to make a prediction. For example, (Xian et al. 2016) use attributes and text as a class embedding. They also use a non-linear latent embedding to compute projections of image or text features, which are then merged through a Mahalanobis distance. A scoring function is learned which determines if the source domain (class descriptions) and target domain (test image) belong to the same class. Similarly, (Changpinyo et al. 2016) find an intermediate representation for text and images with dictionary learning. (Yu and Aloimonos 2010) use a topic-modeling-based genera-

tive model as an intermediate representation. Usually zero-shot learning is performed to make predictions about object categories, but it can analogously be used to predict a novel target attribute, from a set of known source attributes.

However, prior work only considers objects and attributes from the same domain. Our study differs in that we study if transferability of unrelated attributes (from different domains) is more beneficial.

Attention networks. Attention has been used for tasks such as image segmentation (Chen et al. 2016), saliency detection (Kuen, Wang, and Wang 2016), image captioning (You et al. 2016) and image question answering (Xu and Saenko 2016; Shih, Singh, and Hoiem 2016; Yang et al. 2016). The latter use an attention mechanism to decide which regions in an image are relevant to a question input. In our problem scenario, we are not concerned with image regions, but want to know which source classifiers are relevant to a target classifier. Thus, instead of image-text attention, we perform attention-guided transfer from source to target attribute classifiers.

Multi-task Attention Network for Transfer Learning

Overview. We first overview our multi-task attention network, illustrated in Fig. 2. Then, we give more details on its formulation, optimization procedure, and implementation.

An attention architecture allows us to select relevant information and discard irrelevant information. We are interested in selecting relevant source models for our target attributes (e.g. "sporty"). For example, the network might determine attributes X and Z are useful for predicting target attribute A, but attribute Y is not (Fig. 2 (b)). The learned attention weights would reflect the predicted usefulness of the source attributes for the target task.

Our network contains source and target input branches, as depicted in Fig. 2. Similarly to (Shih, Singh, and Hoiem 2016; Yang et al. 2016), we extract $fc7$ features from AlexNet for source and target images. These target (X_t) and source (X_s) visual features are embedded into a common space using a projection matrix W_{shared}, resulting in embedded features X'_t and X'_s. This common space is required to find helpful features that bridge source and target attributes. Then we learn a set of weights (classifiers) W_t and W_s which we multiply by X'_t and X'_s, to obtain attribute presence/absence scores P_t and P_s for the target and source attributes, respectively.

In order to transfer knowledge between the target and source attribute classifiers, we calculate normalized similarities W_{att} between the classifiers W_t and W_s. We refer to W_{att} as the attention weights learned in our network. We then use W_{att} as coefficients to compute a linear combination of the source classifiers W_s. By doing so, we select the most relevant source classifiers related to our target attributes. We call this resulting combined classifier W_{comb}. Finally, we compute the product of W_{comb} with the target features X'_t, to produce the final attribute presence/absence scores for the target attributes.

At training time, our network requires source and target images to find helpful knowledge to our target task. However, once the relationship between source and target attributes is captured in W_{att}, we no longer need the source images. In Fig. 2, we denote modules that are used at test time with dashed boundaries. Layers are denoted with ⬚, and ◌ represents their parameters.

Network formulation. Our network receives target (X_t) and source (X_s) visual features. We process all source and target attributes jointly, i.e. we input training image features for all attributes at the same time. These are embedded in a new common feature space:

$$X'_t = X_t W_{shared} + \vec{1} b \qquad X'_s = X_s W_{shared} + \vec{1} b \quad (1)$$

where $X_t \in \mathbb{R}^{NxD}$, $X_s \in \mathbb{R}^{NxD}$ are the features, $W_{shared} \in \mathbb{R}^{DxM}$ contains the shared embedding weights, $\vec{1} \in \mathbb{R}^{Nx1}$ is a vector of ones, $b \in \mathbb{R}^{1xM}$ is the bias term, N is the batch size, D is the number of input features, and M is the number of features of the embedding.

During backprop training, we learn target and source models W_t and W_s. Note that the target model is only used to compute its similarity to the source models, and will be replaced by a combination of source models in a later stage. We then compute P_t and P_s, which denote the probability of attribute presence/absence for the target and source attributes, respectively. These are only used so we can compute a loss during backprop (described below).

$$P_t = f(X'_t W_t) \qquad P_s = f(X'_s W_s) \quad (2)$$

where $W_t \in \mathbb{R}^{MxK}$, $W_s \in \mathbb{R}^{MxL}$ are learned model weights, f is a sigmoid function (used since we want to compute probabilities), L is the number of source attributes, and K the number of target attributes. We found it is useful to ensure unit-norm per column on W_t and W_s.

Attention weights W_{att} are calculated measuring the similarity between source classifiers W_s and target classifiers W_t. Then, a normalization procedure is applied.

$$O_{att_{i,j}} = \frac{W_{t_i}^T \cdot W_{s_j}^T}{||W_{t_i}^T|| \, ||W_{s_j}^T||}$$
$$W_{att_i} = \frac{[g(O_{att_{i,1}}), ..., g(O_{att_{i,L}})]}{\sum_{j=1}^{L} g(O_{att_{i,j}})} \quad (3)$$

where $W_{t_i}^T$ and $W_{s_j}^T$ are columns from W_t and W_s, $O_{att} \in \mathbb{R}^{KxL}$, $W_{att} \in \mathbb{R}^{KxL}$, g is a RELU function, $O_{att_{i,j}}$ is the similarity between target attribute i and source attribute j, and W_{att_i} are the attention weights for a single target attribute. We use cosine similarity in Eq. 3 to ensure distances are in the range [-1, 1].

When computing attention weights, we want to ensure we do not transfer information from classifiers that are inversely correlated with our target classifier of interest. Thus, we employ normalization over a RELU function (g in Eq. 3) and transfer information from classifiers positively correlated with the target classifier, but discard classifiers that

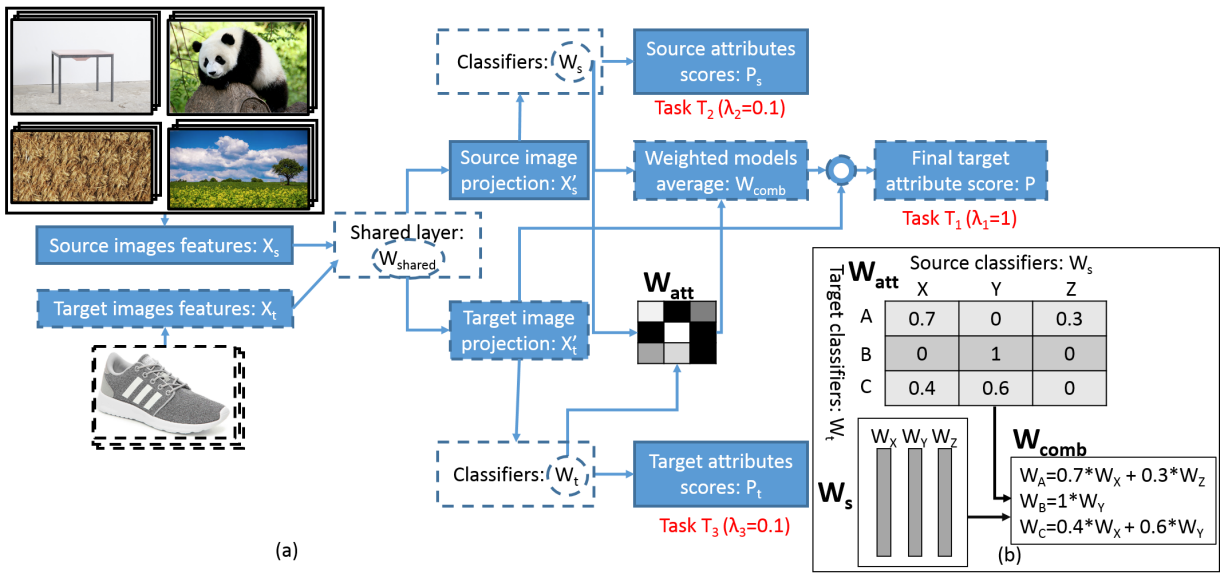

Figure 2: (a) Overview of our transfer attention network, using an example where the target attributes are from the shoes domain, and the source attributes are from the objects, scenes, animals and textures domains. Source and target images are projected through a shared layer. Then, target and source attribute models W_t and W_s are learned. An attention module selects how to weigh the available source classifiers, in order to produce a correct target attribute prediction. At test time, we only use the dashed-line modules. denotes layers, and represents their parameters. (b) Example of how source models W_s are combined into the final target attribute classifiers W_{comb}, using as coefficients the attention weights W_{att}.

are negatively correlated with it (negative similarities are mapped to a 0 weight).

Finally, a weighted combination of source models is created, and multiplied with the target image features X'_t to generate our final predictions for the target attributes:

$$W_{comb} = W_{att} W_s^T \qquad P = f(X'_t W_{comb}^T) \tag{4}$$

where $W_{comb} \in \mathbb{R}^{KxM}$ is the weighted combination of sources, and f is a sigmoid function.

Note our model is simple to train as it only requires the learning of three sets of parameters, W_{shared}, W_s and W_t.

Optimization. Our network performs three tasks. The main task T_1 predicts target attributes using attention-guided transfer, and side tasks T_2 and T_3 predict source and target attributes, respectively. Each task T_i is associated with a loss L_i. Our optimization loss is defined as

$$L = \lambda_1 * L_1 + \lambda_2 * L_2 + \lambda_3 * L_3 \tag{5}$$

where $\lambda_1 = 1$, $\lambda_2 = 0.1$, and $\lambda_3 = 0.1$.[1] Since an image can posses more than one attribute, our predictions are multi-label and we employ binary cross-entropy loss for all L_i.

For task T_2, our source image branch contains attributes from different domains. Thus an image has annotations for attributes in its domain, but not for other domains. We solve this issue with a customized cross-entropy loss (Fouhey, Gupta, and Zisserman 2016). Suppose you have N samples and L attributes. Each attribute is annotated with 0, 1 or $\varnothing$, where $\varnothing$ denotes no annotation. The customized loss is:

$$L(Y,P) = \sum_{i=1}^{N} \sum_{\substack{j=1 \\ Y_{i,j} \neq \varnothing}}^{L} Y_{i,j} log(P_{i,j}) + (1 - Y_{i,j}) log(1 - P_{i,j}) \tag{6}$$

where i is an image, j is an attribute label, $Y_{i,j} \in \{0, 1, \varnothing\}^{N,L}$ is the ground-truth attribute label matrix and $P_{i,j} \in [0, 1]^{N,L}$ is the prediction probability for image i and attribute j. The constraint $Y_{i,j} \neq \varnothing$ means attribute annotations $\varnothing$ have no effect on the loss.

Implementation. We implemented the described network using the Theano (Theano Development Team 2016) and Keras (Chollet 2015) frameworks and (Singh and Lee 2016)'s attention network. First, we did parameter exploration using 70 random configurations of learning rate and L_2 regularizer weight. Each configuration ran for five epochs with the ADAM optimizer. Then the configuration with the highest accuracy on a validation set was selected and a network with this configuration ran for 150 epochs. In the end of each epoch, the network was evaluated on a validation set, and training was stopped when the validation accuracy began to decrease. Finally, note that we have fewer target images than source images, so the target images were sampled more times.

[1] The loss weights were selected similar to other transfer learning work (Tzeng et al. 2015) where the main task has a weight of 1, and side tasks have a weight of 0.1.

Experimental Validation

We compare three types of source data for attribute transfer, i.e. three types of data that can be passed in the source branch of Fig. 2. This data can correspond to attributes from the same domain, from a disjoint domain, or from any domain. The first option corresponds to the standard manner of performing semantic (within-domain) attribute transfer (Chen and Grauman 2014; Han et al. 2014; Liu and Kovashka 2016). The latter two options represent our *non-semantic transfer* approach.

To evaluate the benefit of transfer, we also compare to a method that learns target attributes from scratch with no source data, and two standard transfer learning approaches (Tzeng et al. 2015; Oquab et al. 2014). We do not directly compare to attribute transfer methods (Chen and Grauman 2014; Han et al. 2014; Liu and Kovashka 2016) as they do not use neural nets and the comparison would not be fair.

We evaluated our method and the baselines on five domains and 272 attributes. We observe that by transferring from disjoint domains or from any domain, i.e. by performing *non-semantic transfer* without the requirement for a semantic relationship between the source and target tasks, we achieve the best results. To better understand the transfer process, we also show attention weights and determine the most relevant source domains per target domain/attribute.

Datasets

We use five datasets: Animals with Attributes (Lampert, Nickisch, and Harmeling 2009), aPascal/aYahoo Objects (Farhadi et al. 2009), SUN Scenes (Patterson et al. 2014), Shoes (Kovashka, Parikh, and Grauman 2015), and Textures (Caputo, Hayman, and Mallikarjuna 2005). The number of attributes is 85, 64, 102, 10 and 11, respectively. The total number of images is 30,475; 15,340; 14,340; 14,658 and 4,752, respectively.

For each dataset, we split the data in 40% for training the source models, 10% for training the target models, 10% for selection of the optimal network parameters, and 40% to test the final trained network on the target data. The complexity of the experimental setup is to ensure fair testing. For transfer learning among different domains (ATTENTION-DD and ATTENTION-AD below), we can increase the size of our source data split to the full dataset, but for fair comparison, we use the same split as for the ATTENTION-SD setup.

Our splits mimic the scenario where we have plentiful data from the source attributes, but limited data for the attribute of interest.

Baselines

Let D_i represent a domain and its attributes, and $D = \bigcup_{i=1}^{5} D_i$ be the union of all domains. We compare seven methods. The first are two ways of performing non-semantic transfer:

- ATTENTION-DD, which is our multitask attention network with D_i as our target domain and $D \backslash D_i$ as our source domains. We train five networks, one for each configuration of target/source.
- ATTENTION-AD, which is our multitask attention network with D_i as our target domain and D as our source domains. We again train one network for each target domain. Some attributes on the source and target branches overlap, so we assign 0 values along the diagonal of W_{att} to avoid transfer between an attribute and itself.

We compare our methods against the following baselines:

- ATTENTION-SD, which uses the same multitask attention network but applies it on attributes from only a single domain D_i, for both the source and target branches. We again train five networks, and assign values of 0 along the diagonal of W_{att}. Note that even though some form of transfer is already taking place between all target attributes due to the multi-task loss, the explicit transfer from the source domains is more effective because we have more training data for the sources than the targets.
- TARGET-ONLY, which uses the predictions P_t as the final predictions of the network, without any transfer from the source models.
- A replacement of the attention weights W_{att} with uniform weights, i.e. combining all source classifiers with the same importance for all targets. This results in baselines ATTENTION-SDU, ATTENTION-DDU and ATTENTION-ADU.
- (Tzeng et al. 2015) which learns feature representations X'_s, X'_t invariant across domains, using domain classifier and confusion losses but no attention. This results in baselines CONFUSION-DD and CONFUSION-AD.
- Approaches FINETUNE-DD and FINETUNE-AD that fine-tune an AlexNet network using source data, then fine-tune those source networks again for the target domain. This method represents "standard" transfer learning for neural networks (Oquab et al. 2014).

We found that ATTENTION-SD is a weak baseline. Thus, we replace it by an ensemble of TARGET-ONLY with ATTENTION-SD. This ensemble averages the probability outputs of these two models. We try a similar procedure for ATTENTION-DD and ATTENTION-AD, but it weakens their performance, so we use these methods in their original form.

Quantitative results

Tables 1 and 2 contain show average accuracy and F-measure, respectively. We show both per-domain and across-domains overall averages. We include F-measure because many attributes have imbalanced positive/negative data.

In both tables, we see that our methods ATTENTION-DD and ATTENTION-AD outperform or perform similar to the baselines in terms of the overall average. While the strongest baselines CONFUSION-DD and CONFUSION-AD (Tzeng et al. 2015) perform similarly to our methods for accuracy[2], our methods have much stronger F-measure (Table 2). Accuracies in Table 1 seem misleadingly high because attribute annotations are imbalanced in terms of positives/negatives and a baseline that predicts all negatives will do well in terms

[2]The top four methods have slightly different performance using three decimals.

Table 1: Method comparison using accuracy. Our ATTENTION-DD and ATTENTION-AD outperform or perform equal to the other methods on average. Best results are **bolded** per row.

	TARGET -ONLY	ATTENTION -SDU	ATTENTION -DDU	ATTENTION -ADU	ATTENTION -SD	ATTENTION -DD (ours)	ATTENTION -AD (ours)	CONFUSION -DD	CONFUSION -AD	FINETUNE -DD	FINETUNE -AD
avg animals	0.90	0.63	0.63	0.73	0.90	0.91	0.91	0.91	0.91	0.88	**0.92**
avg objects	0.92	0.89	0.89	0.89	0.92	**0.93**	**0.93**	**0.93**	**0.93**	0.91	0.92
avg scenes	**0.95**	0.93	0.93	0.93	**0.95**	**0.95**	**0.95**	**0.95**	**0.95**	**0.95**	**0.95**
avg shoes	0.88	0.70	0.71	0.79	0.89	0.90	0.90	0.90	0.90	0.75	**0.92**
avg textures	0.91	0.87	0.91	0.91	0.95	**0.99**	**0.99**	**0.99**	**0.99**	0.91	0.91
avg overall	0.91	0.80	0.81	0.85	0.92	**0.94**	**0.94**	**0.94**	**0.94**	0.88	0.92

Table 2: Method comparison using F-measure. Our approaches ATTENTION-DD and ATTENTION-AD outperform the other methods on average. Best results are **bolded** per row.

	TARGET -ONLY	ATTENTION -SDU	ATTENTION -DDU	ATTENTION -ADU	ATTENTION -SD	ATTENTION -DD (ours)	ATTENTION -AD (ours)	CONFUSION -DD	CONFUSION -AD	FINETUNE -DD	FINETUNE -AD
avg animals	0.81	0.00	0.00	0.27	0.82	0.82	**0.83**	0.82	0.82	0.69	0.79
avg objects	**0.50**	0.00	0.00	0.01	**0.50**	0.47	0.47	0.39	0.41	0.10	0.14
avg scenes	**0.28**	0.00	0.00	0.00	0.27	0.25	0.26	0.17	0.15	0.04	0.04
avg shoes	0.81	0.27	0.38	0.59	0.83	0.83	0.84	0.83	0.83	0.37	**0.87**
avg textures	0.68	0.09	0.00	0.00	0.78	**0.96**	**0.96**	0.95	0.95	0.06	0.09
avg overall	0.62	0.07	0.08	0.17	0.64	**0.67**	**0.67**	0.63	0.63	0.25	0.39

Table 3: Attention weights summed per domain for our ATTENTION-DD approach. Rows vs columns represent target vs source classifiers. The most relevant domains are **bolded** per row. − denotes ATTENTION-DD does not transfer from attributes in the same domain.

tgt/src	animals	objects	scenes	shoes	textures
animals	-	0.29	**0.56**	0.06	0.09
objects	**0.48**	-	0.44	0.04	0.04
scenes	**0.59**	0.28	-	0.07	0.06
shoes	0.19	0.35	**0.38**	-	0.08
textures	0.33	0.19	**0.44**	0.04	-

Table 4: Attention weights summed per domain for our ATTENTION-AD approach.

tgt/src	animals	objects	scenes	shoes	textures
animals	**0.43**	0.09	0.39	0.02	0.07
objects	0.26	0.21	**0.41**	0.04	0.08
scenes	0.36	0.19	**0.39**	0.02	0.04
shoes	0.10	0.30	**0.50**	0.00	0.10
textures	0.36	0.16	**0.39**	0.03	0.06

of accuracy (0.81 on average), but not in terms of F-score. Thus, the differences between the methods are larger than they seem.

It is important to highlight the success of ATTENTION-DD as it does not use any attributes from the target domain, as opposed to ATTENTION-AD. In other words, transfer is more successful when we allow information to be transferred even from domains that are semantically unrelated to the target. In addition, note that the uniform weight baselines (ATTENTION-SDU, ATTENTION-DDU and ATTENTION-ADU) are quite weak. This shows that only by selecting the source classifiers intelligently, we can perform transfer learning correctly. We see many 0 F-measure scores for ATTENTION-SDU, ATTENTION-DDU and ATTENTION-ADU because they have a bias to predict negative labels.

While FINETUNE-AD outperforms our methods for two domains in Table 1, it is weaker in terms of the overall average, and weaker in four out of five domains in Table 2.

Finally, the attention transfer methods with learned attention weights usually outperform TARGET-ONLY, which emphasizes the benefit of transfer learning. Our non-semantic transfer methods bring the largest gains.

We believe the success of our attention network is due to the combination of transfer learning via a common feature representation, and parameter transfer. The common feature representation is achieved via our shared layer, and the parameter transfer is performed via our attention-guided transfer. Finally, we believe that instance weighting also helps: this is accomplished by our choice to sample more target images than source images.

Qualitative results

In order to analyze the internal behavior of ATTENTION-DD and ATTENTION-AD, we extract and show the attention weights W_{att}. Hence, for each target classifier i, we extract the weights $W_{att_i} = (w_1, w_2, ..., w_L)$ for the source classifiers. This procedure also verifies if ATTENTION-AD is primarily using transfer from attributes in the same domain, or attributes from disjoint domains with respect to the target. Due to the large number of attributes, we group attributes by their domain. Rows represent targets, and columns sources.

In Table 3 corresponding to ATTENTION-DD, the attention weights over the source classifiers are distributed among animals, objects, and scenes. We believe that shoes attributes are not very helpful for other domains because shoes images only contain one object. Further, textures are likely not very helpful because they are a low-level representation mainly defined by edges. Interestingly, we observe that the most relevant domain for animals, shoes, and textures is scenes, and scenes is *not closely related to any of these domains*. Similarly, the most meaningful domain for objects and scenes is animals, another *semantically unrelated source domain*.

In Table 4, showing results when we perform transfer

Table 5: Interesting selected source attributes from domains disjoint from the target domain.

domain	target attribute	some relevant source attributes from [domain]
textures	aluminium	muscular [animal], made of glass [object]
	linen	handlebars [object], railroad [scene]
	lettuce leaf	lives in forest [animal]
shoes	pointy	foliage [scene]
	bright-in-color	vegetation [scene], shrubbery [scene]
	long-on-the-leg	has leg [object]
object	has stem	dirty soil [scene], feed from fields [animal]
	vegetation	dirty soil [scene]
animal	tough-skinned	stressful [scene]
	fast	scary [scene]
	hunter	studying [scene]
scene	railroad	solitary [animal]
	shrubbery	tough-skinned [animal]

from *any* domain, we observe that shoes and textures attributes do not benefit almost at all from other attributes in the same domain. On the other hand, objects, scenes, animals do benefit from semantically related attributes, but the overall within-domain model similarity is lower than 50%, again reaffirming our choice to allow non-semantic transfer.

Finally, we illustrate what visual information is being transferred across domains. In Table 5, we show relevant source attributes for several target attributes. The "aluminium" texture presents a "muscular" structure, and a color similar to "glass". The "linen" texture has edges similar to "handlebars" and "railroads". "Lettuce leaf" shows leaves' textures, so "forest" animals (which might co-occur with leaves) are helpful. For shoes attributes, "foliage" is a set of "pointy" leaves, "vegetation" and "shrubbery" are "bright-in-color", and "leg" is related to shoes that are "long-on-the-leg". For object attributes, "vegetation" and objects with a "stem" grow on "dirty soil" and animals might "feed" on them. For animal attributes, "tough skin" gives us the feeling of a "stressful" situation, "fast" animals might "scare" people, and "hunter" animals "study" the best situation to catch their prey. Finally, "railroad" scenes might be "solitary" places, and "shrubbery" is rough like "tough-skinned" animals. In other words, while source attributes are selected from disjoint domains, it is possible to explain some selections, but note that many do not have an intuitive explanation. The latter is indeed what we expect when we perform non-semantic transfer.

Conclusion

We have explored the problem of attribute transfer learning using unrelated domains. We develop an approach that transfers knowledge in a common feature space, by performing parameter transfer from source models. Our attention mechanism intelligently weights source attribute models to improve performance on target attributes. We find that attributes from a different domain than the target attributes are quite beneficial for transfer learning, and improve accuracy more than transfer from semantically related attributes. We also outperform standard transfer learning approaches.

Acknowledgement. This research was supported by a University of Pittsburgh CRDF grant. This work also used the Extreme Science and Engineering Discovery Environment (XSEDE) and the Data Exacell at the Pittsburgh Supercomputing Center (PSC), supported by National Science Foundation grants ACI-1053575 and ACI-1261721.

References

Akata, Z.; Perronnin, F.; Harchaoui, Z.; and Schmid, C. 2013. Label-embedding for attribute-based classification. In *Computer Vision and Pattern Recognition (CVPR)*. IEEE.

Bousmalis, K.; Trigeorgis, G.; Silberman, N.; Krishnan, D.; and Erhan, D. 2016. Domain separation networks. In *Advances in Neural Information Processing Systems (NIPS)*.

Branson, S.; Wah, C.; Schroff, F.; Babenko, B.; Welinder, P.; Perona, P.; and Belongie, S. 2010. Visual recognition with humans in the loop. In *European Conference of Computer Vision (ECCV)*. Springer.

Caputo, B.; Hayman, E.; and Mallikarjuna, P. 2005. Class-specific material categorisation. In *International Conference on Computer Vision (ICCV)*. IEEE.

Castrejon, L.; Aytar, Y.; Vondrick, C.; Pirsiavash, H.; and Torralba, A. 2016. Learning aligned cross-modal representations from weakly aligned data. In *Computer Vision and Pattern Recognition (CVPR)*. IEEE.

Changpinyo, S.; Chao, W.-L.; Gong, B.; and Sha, F. 2016. Synthesized classifiers for zero-shot learning. In *Computer Vision and Pattern Recognition (CVPR)*. IEEE.

Chen, C.-Y., and Grauman, K. 2014. Inferring analogous attributes. In *Computer Vision and Pattern Recognition (CVPR)*. IEEE.

Chen, Q.; Huang, J.; Feris, R.; Brown, L. M.; Dong, J.; and Yan, S. 2015. Deep domain adaptation for describing people based on fine-grained clothing attributes. In *Computer Vision and Pattern Recognition (CVPR)*. IEEE.

Chen, L.-C.; Yang, Y.; Wang, J.; Xu, W.; and Yuille, A. L. 2016. Attention to scale: Scale-aware semantic image segmentation. In *Computer Vision and Pattern Recognition (CVPR)*. IEEE.

Chollet, F. 2015. Keras.

Farhadi, A.; Endres, I.; Hoiem, D.; and Forsyth, D. 2009. Describing objects by their attributes. In *Computer Vision and Pattern Recognition (CVPR)*. IEEE.

Fouhey, D. F.; Gupta, A.; and Zisserman, A. 2016. 3D shape attributes. In *Computer Vision and Pattern Recognition (CVPR)*. IEEE.

Gan, C.; Yang, T.; and Gong, B. 2016. Learning attributes equals multi-source domain generalization. In *Computer Vision and Pattern Recognition (CVPR)*. IEEE.

Ganin, Y., and Lempitsky, V. 2015. Unsupervised domain adaptation by backpropagation. In *International Conference of Machine Learning (ICML)*. IEEE.

Ghifary, M.; Kleijn, W. B.; Zhang, M.; Balduzzi, D.; and Li, W. 2016. Deep reconstruction-classification networks for unsupervised domain adaptation. In *European Conference on Computer Vision (ECCV)*. Springer.

Gong, B.; Shi, Y.; Sha, F.; and Grauman, K. 2012. Geodesic flow kernel for unsupervised domain adaptation. In *Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE.

Han, Y.; Yang, Y.; Ma, Z.; Shen, H.; Sebe, N.; and Zhou, X. 2014. Image attribute adaptation. *IEEE Transactions on Multimedia*.

Huang, S.; Elhoseiny, M.; Elgammal, A.; and Yang, D. 2015. Learning hypergraph-regularized attribute predictors. In *Computer Vision and Pattern Recognition (CVPR)*. IEEE.

Kan, M.; Shan, S.; and Chen, X. 2015. Bi-shifting auto-encoder for unsupervised domain adaptation. In *International Conference on Computer Vision (ICCV)*. IEEE.

Kovashka, A.; Parikh, D.; and Grauman, K. 2015. Whittlesearch: Interactive image search with relative attribute feedback. *International Journal of Computer Vision (IJCV)*.

Kovashka, A.; Vijayanarasimhan, S.; and Grauman, K. 2011. Actively selecting annotations among objects and attributes. In *Interantional Conference of Computer Vision (ICCV)*. IEEE.

Kuen, J.; Wang, Z.; and Wang, G. 2016. Recurrent attentional networks for saliency detection. In *Computer Vision and Pattern Recognition (CVPR)*. IEEE.

Kumar, N.; Berg, A. C.; Belhumeur, P. N.; and Nayar, S. K. 2011. Describable visual attributes for face verification and image search. *Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*.

Lampert, C.; Nickisch, H.; and Harmeling, S. 2009. Learning to Detect Unseen Object Classes By Between-Class Attribute Transfer. In *Computer Vision and Pattern Recognition (CVPR)*. IEEE.

Liang, K.; Chang, H.; Shan, S.; and Chen, X. 2015. A unified multiplicative framework for attribute learning. In *International Conference on Computer Vision (ICCV)*. IEEE.

Liu, S., and Kovashka, A. 2016. Adapting attributes by selecting features similar across domains. In *Applications of Computer Vision (WACV)*. IEEE.

Liu, Z.; Luo, P.; Wang, X.; and Tang, X. 2015. Deep learning face attributes in the wild. In *Computer Vision and Pattern Recognition (CVPR)*. IEEE.

Long, M.; Zhu, H.; Wang, J.; and Jordan, M. I. 2016. Unsupervised domain adaptation with residual transfer networks. In *Advances in Neural Information Processing Systems (NIPS)*.

Oquab, M.; Bottou, L.; Laptev, I.; and Sivic, J. 2014. Learning and transferring mid-level image representations using convolutional neural networks. In *Computer Vision and Pattern Recognition (CVPR)*. IEEE.

Parikh, D., and Grauman, K. 2011. Relative attributes. In *International Conference of Computer Vision (ICCV)*. IEEE.

Parkash, A., and Parikh, D. 2012. Attributes for classifier feedback. In *European Conference on Computer Vision (ECCV)*. Springer.

Patterson, G.; Xu, C.; Su, H.; and Hays, J. 2014. The sun attribute database: Beyond categories for deeper scene understanding. *International Journal of Computer Vision (IJCV)*.

Shao, J.; Kang, K.; Loy, C. C.; and Wang, X. 2015. Deeply learned attributes for crowded scene understanding. In *Computer Vision and Pattern Recognition (CVPR)*. IEEE.

Shih, K. J.; Singh, S.; and Hoiem, D. 2016. Where to look: Focus regions for visual question answering. In *Computer Vision and Pattern Recognition (CVPR)*. IEEE.

Singh, K. K., and Lee, Y. J. 2016. End-to-end localization and ranking for relative attributes. In *European Conference on Computer Vision (ECCV)*. Springer.

Theano Development Team. 2016. Theano: A Python framework for fast computation of mathematical expressions. *arXiv e-prints*.

Tommasi, T.; Orabona, F.; and Caputo, B. 2014. Learning categories from few examples with multi model knowledge transfer. *Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*.

Tzeng, E.; Hoffman, J.; Darrell, T.; and Saenko, K. 2015. Simultaneous deep transfer across domains and tasks. In *International Conference on Computer Vision (ICCV)*. IEEE.

Wah, C., and Belongie, S. 2013. Attribute-Based Detection of Unfamiliar Classes with Humans in the Loop. In *Computer Vision and Pattern Recognition (CVPR)*. IEEE.

Xian, Y.; Akata, Z.; Sharma, G.; Nguyen, Q.; Hein, M.; and Schiele, B. 2016. Latent embeddings for zero-shot classification. In *Computer Vision and Pattern Recognition (CVPR)*. IEEE.

Xiao, F., and Jae Lee, Y. 2015. Discovering the spatial extent of relative attributes. In *International Conference on Computer Vision (ICCV)*. IEEE.

Xu, H., and Saenko, K. 2016. Ask, attend and answer: Exploring question-guided spatial attention for visual question answering. In *European Conference on Computer Vision (ECCV)*. Springer.

Yang, Z.; He, X.; Gao, J.; Deng, L.; and Smola, A. 2016. Stacked attention networks for image question answering. In *Computer Vision and Pattern Recognition (CVPR)*. IEEE.

Yang, J.; Yan, R.; and Hauptmann, A. G. 2007. Cross-domain video concept detection using adaptive svms. In *International Conference on Multimedia (ICM)*. ACM.

You, Q.; Jin, H.; Wang, Z.; Fang, C.; and Luo, J. 2016. Image captioning with semantic attention. In *Computer Vision and Pattern Recognition (CVPR)*. IEEE.

Yu, X., and Aloimonos, Y. 2010. Attribute-based transfer learning for object categorization with zero/one training example. In *European Conference on Computer vision (ECCV)*. Springer.