Doing the Best We Can with What We Have: Multi-Label Balancing with Selective Learning for Attribute Prediction

Emily M. Hand, Carlos Castillo and Rama Chellappa
University of Maryland, College Park
College Park, MD
{emhand, carlos, rama} @umiacs.umd.edu

Abstract

Attributes are human describable features, which have been used successfully for face, object, and activity recognition. Facial attributes are intuitive descriptions of faces and have proven to be very useful in face recognition and verification. Despite their usefulness, to date there is only one large-scale facial attribute dataset, CelebA (Liu et al. 2015). Impressive results have been achieved on this dataset, but it exhibits a variety of very significant biases. As CelebA contains mostly frontal idealized images of celebrities, it is difficult to generalize a model trained on this data for use on another dataset (of non celebrities). A typical approach to dealing with imbalanced data involves sampling the data in order to balance the positive and negative labels, however, with a multi-label problem this becomes a non-trivial task. By sampling to balance one label, we affect the distribution of other labels in the data. To address this problem, we introduce a novel Selective Learning method for deep networks which adaptively balances the data in each batch according to the desired distribution for each label. The bias in CelebA can be corrected for in this way, allowing the network to learn a more robust attribute model. We argue that without this multi-label balancing, the network cannot learn to accurately predict attributes that are poorly represented in CelebA. We demonstrate the effectiveness of our method on the problem of facial attribute prediction on CelebA, LFWA, and the new University of Maryland Attribute Evaluation Dataset (UMD-AED), outperforming the state-of-the-art on each dataset (Liu et al. 2015).

Introduction

Attributes are semantic features which have been used for many different applications, ranging from face verification to action recognition (Kumar et al. 2009; Zheng et al. 2014). Accurately predicting facial attributes (e.g. gender, hair color, eye color, etc.) from images is a very difficult problem, and has become of recent interest in the computer vision community with the introduction of CelebA (Liu et al. 2015). CelebA is the first and only widely available large-scale facial attribute dataset. Since it's introduction, there has been a lot of progress made on the problem of facial attribute recognition from images. Deep CNNs have proven to be very effective in attribute prediction, with state-of-the-art methods using CNNs for feature extraction (Liu et al. 2015; Rudd, Gunther, and Boult 2016; Wang, Cheng, and Feris 2016).

Impressive results have been achieved on CelebA (Liu et al. 2015), however this dataset exhibits some extreme biases. It consists of mostly frontal, high-quality, posed images of celebrities, which is not representative of real-world imagery. Models trained on this data - without accounting for its biases - are not likely to perform well on another domain (e.g. images of non celebrities, or low quality images). A traditional method for handling imbalanced data is to sample the data so as to balance a particular label. However, for the multi-label setting, the data cannot be balanced in this way. Sampling the data so that one label is balanced changes the distribution for the other labels. In an ideal world, we would have access to unbiased, balanced, precisely-labeled datasets, but with CelebA as the only widely available, large-scale attribute dataset, we must find ways to overcome its biases in order to move towards a solution to the attribute prediction problem.

In (Rudd, Gunther, and Boult 2016), the authors try to adjust for the imbalance in CelebA by introducing a mixed objective loss, which adjusts the back-propagation weights according to a given target distribution. This method is a step in the right direction, but does not take full advantage of deep networks. Deep CNNs use batch learning, and so the label imbalance must be addressed in each batch. We argue that in order to truly account for the bias in CelebA, label balancing must be performed at the batch level when training a CNN. To this end, we propose a domain-adaptive batch re-sampling method for training CNNs, which we call Selective Learning. Selective Learning adapts each batch separately for every attribute according to a given target distribution for that attribute. There are many under-represented attributes in CelebA, including *bald*, *mustache*, *gray hair*, etc, and several over-represented attributes, such as *young*, and *nobeard*. In order to remove the bias in CelebA, Selective Learning adapts the batches for each attribute, allowing the model to learn from balanced data in every batch. This multi-label balancing allows our deep network to learn features which truly represent the facial attributes, not just the bias in the training data.

We evaluate the proposed Selective Learning on CelebA, LFWA, and a new dataset: UMD-AED. UMD-AED consists of 3338 images, each labeled with a subset of the 40 binary attributes from CelebA. Each of the 40 attributes has 50 positive samples and 50 negative samples in UMD-AED. A model which has learned a robust representation for facial attributes, should perform well on any test set, no matter the distribution for each attribute. Our model, trained with Selective Learning, outperforms the state-of-the-art on each dataset.

Related Work

Attributes

Attributes - human describable features - have been used for many different computer vision tasks, such as activity, object, and face recognition (Cheng et al. 2013; Duan et al. 2012; Farhadi et al. 2009; Hwang, Sha, and Grauman 2011; Jayaraman, Sha, and Grauman 2014; Zhang et al. 2014a; Zheng et al. 2014). Facial attributes - *gender*, *hair color*, *eye color*, etc. - have been successfully used in face verification and recognition (Kumar et al. 2009; 2011). The problems of recognizing *gender* and *age* have been studied extensively for many years (Fu, Guo, and Huang 2010; Ng, Tay, and Goi 2012). (Kumar et al. 2009) first introduced the concept of facial attributes for face verification, following up on that work with (Kumar et al. 2011). They used 65 - and then 73 - binary attributes as face descriptors. Even before this, Kumar et. al used attributes for image search in their FaceTracer work, predicting attributes using a combination of SVMs and Adaboost (Kumar, Belhumeur, and Nayar 2008). With the release of two face datasets with attribute labels, great advances have been made in the recognition of facial attributes in the past few years, with deep networks achieving impressive results (Abdulnabi et al. 2015; Huang et al. 2016) (Levi and Hassner 2015; Liu et al. 2015; Zhang et al. 2014a; Zhong, Sullivan, and Li 2016). Pose Aligned Networks for Deep Attributes (PANDA) combines part-based models with deep CNNs to learn features for specific parts in a specific pose. PANDA is used for human attribute prediction - full body attributes - and performs well on unconstrained images (Zhang et al. 2014a). (Zhang et al. 2014b) uses three attributes: *gender*, *smiling* and *wearing glasses* to improve facial landmark localization. (Ranjan, Patel, and Chellappa 2015) uses multi-task learning to jointly learn face detection, landmarks, pose, and *gender* combining features from intermediate layers in order to learn the different tasks. (Ehrlich et al. 2016) uses a RBM for multi-task attribute learning, achieving state-of-the-art results on several large-scale attribute datasets. (Liu et al. 2015) uses two deep CNNs, one for localizing the face in the image (LNet), and one for attribute prediction (ANet). They introduce the CelebA dataset, a large-scale attribute-labeled dataset of face images. Their method, LNet+ANet, outperformed PANDA and FaceTracer on the CelebA dataset (Kumar, Belhumeur, and Nayar 2008; Zhang et al. 2014a). Using wearable cameras, collecting face tracks with weather and location metadata, (Wang, Cheng, and Feris 2016) achieves state-of-the-art results on attribute prediction by first training a verification network on the wearable camera data, then fine-tuning the network for attribute prediction. Our network, AttCNN, requires no pretraining on external data, and outperforms (Liu et al. 2015) as well as (Wang, Cheng, and Feris 2016) on CelebA.

Domain Adaptation

There have been many different methods for domain adaptation over the years (Patel et al. 2015). Object recognition has benefitted from domain adaptation methods, especially since the introduction of a benchmark dataset (Saenko et al. 2010). Semi-supervised approaches have dominated the field with dictionary learning methods (Bo, Ren, and Fox 2011; Yang et al. 2011), and metric learning methods (Saenko et al. 2010). Many unsupervised approaches have been introduced as well, with dictionary (Lu, Chellappa, and Nasrabadi 2015) and manifold-based methods (Gopalan, Li, and Chellappa 2011; Gong et al. 2012) being the most popular. Face recognition can easily be framed as a domain adaptation problem with faces in different poses and with different illuminations and resolutions (Ho and Gopalan 2014; Qui et al. 2012; Shekhar et al. 2013).

(Chen et al. 2015) tackles the problem of domain adaptation of clothing attributes from ideal images to images taken in unconstrained environments. They use a two-stream CNN to model the two domains in separate paths, using connections between the two paths to ensure that the features are similar for both domains. This method works in supervised and unsupervised settings. In (Ganin and Lempitsky 2014), the feature extraction portion of the network feeds into two different predictor portions: the class predictor, and the domain predictor. As the domain predictor backpropagates the error, it reverses the gradient when it passes through the feature extraction portion of the network. This allows the network to learn the class labels while keeping the feature distributions for the two domains similar. (Rudd, Gunther, and Boult 2016) addresses the problem of dataset bias in a multi-label setting. The authors introduce a Mixed-Objective Optimization Network (MOON) for attribute recognition, by weighting the back-propagation error for each attribute according to a given target distribution. Unlike MOON, which does not account for the label imbalance in the batches, Selective Learning performs a domain-adaptive re-sampling at the batch level, so that each batch fits the target distribution for each attribute.

In domain adaptation, there are two sets of data: the source and the target, both consisting of images and labels. Let X_S and X_T be the source and target images, and Y_S and Y_T be the source and target labels respectively. In supervised domain adaptation, the model has access to X_S, Y_S, X_T, and Y_T at training time. In unsupervised domain adaptation, the model has access to X_S, Y_S, and X_T, but Y_T is unknown. In the proposed Selective Learning approach, the model has access to X_S, Y_S and has some idea of what the distribution for Y_T will be, but it does not have access to X_T during training. We argue that this is a form of domain adaptation, as we are adjusting the learning procedure on the source data to account for a desired target label distribution. (Rudd, Gunther, and Boult 2016) uses a similar formulation.

Facial attributes were first introduced to describe faces (Fu, Guo, and Huang 2010; Kumar, Belhumeur, and Nayar

Layer	Parameters/Activation/Pooling/Norm
Conv1	75 7x7 Filters, Stride 4 ReLU Max Pooling 3x3, Stride 2 Norm 5x5
Conv2	200 5x5 Filters ReLU Max Pooling 3x3, Stride 2 Norm 5x5
Conv3	300 3x3 Filters ReLU Max Pooling 5x5, Stride 2 Norm 5x5
FC1	512 Units ReLU Dropout 50%
FC2	512 Units ReLU Dropout 50%
FC3	40 Units

Table 1: AttCNN Architecture. Conv1 is the bottom layer, and FC3 is the top and final layer producing 40 outputs.

2008; Ng, Tay, and Goi 2012). With the introduction of deep networks, the focus has been shifted to improving face verification methods using attributes, however we cannot say for certain that these describable facial features play a discriminative role in face recognition (Rudd, Gunther, and Boult 2016). Most deep networks pre-train on external datasets – usually labeled with identity – and then fine-tune the weights for the task of attribute prediction. These networks have learned a representation for identity, which does not necessarily translate to a representation for facial attributes. We will see this with our comparison to the state-of-the-art method from (Wang, Cheng, and Feris 2016), which was pre-trained with verification data. In this work, we choose to focus on improving attribute models solely for the purpose of accurately describing faces using their physical features, rather than for use in a face verification system.

The proposed Selective Learning approach performs multi-label balancing in each training batch, and we provide results on CelebA, LFWA, and a new evaluation dataset, UMD-AED. Selective Learning outperforms the state-of-the-art on CelebA – by 0.11% on average – on LFWA – by 2.5% on average – and on UMD-AED – by over 11% on average.

Proposed Method

Multi-Task Attribute CNN

For attribute prediction, we use a multi-task deep attribute CNN (AttCNN) implemented in Caffe (Jia et al. 2014). Table 1 shows the AttCNN architecture. FC3 is the output layer, with 40 nodes, one for each attribute. A sigmoid cross-entropy loss is used to facilitate training of the AttCNN. At test time, we apply the sigmoid function to FC3, taking values above 0.5 as positive instances of an attribute, and values below 0.5 as negative attribute responses.

Selective Learning

We introduce a novel Selective Learning method which adaptively balances each batch according to the desired distribution for each label in a multi-task learning framework. In other words, Selective Learning performs multi-label balancing of the training data. Consider, for example, the two attributes *bald*, and *male*. We intuitively know that the distribution for *male* is much more balanced than that for *bald*, and so we would expect to see more positive instances of *male* than *bald*. If we were to train a separate model for each attribute, we would be able to sample the data such that our model for *bald* could learn from a more balanced set. However, in a multi-task setting, where we learn all attributes at once, it is much more difficult to handle these imbalances. Selective Learning offers a solution to this problem by adaptively balancing each label in every batch of data according to a target distribution for that label.

Batch Balancing For each label (attribute) in each batch, if the distribution for that label does not match the desired target distribution, then we must adapt the batch accordingly. For each label, there are three cases: 1) the batch distribution is equal to the target distribution, 2) the label is over-represented, and 3) the label is under-represented. If the batch distribution for a label is equal to the target distribution, then we do nothing, and the Selective Learning batch (SL batch) is the same as the original batch for that label.

If a label is over-represented in a batch, that means there are more positive instances and fewer negative instances than if the batch followed the target distribution. When there are too many positive instances in the original batch, we take a random subset from the positive samples according to the target distribution and add those to the SL batch, ignoring the rest of the positive samples. For example, if we have a batch of size 100, with 70 positive instances, and a balanced target distribution, then we sample 50 positive instances, ignoring the other 20. At this point, the SL batch contains a subset of the positive samples from the original batch.

We must now adjust the negative instances for the given label. Since the positive instances are over-represented, we were able to simply sample from the positive instances to meet the target distribution, but there are not enough negative instances to meet the target distribution. Instead, we weight the negative samples so they effectively match the target distribution. Using the same example from above, we have 30 negative samples in the original batch, so in the SL batch, we weight the negative samples by $\frac{5}{3}$ so that the negative samples effectively match the balanced target distribution. That is, the SL batch contains a subset of the positive samples, and all the negative samples, with an additional weight attached to them. If a label is under-represented, we reverse the above process, sampling from the negative instances and weighting the positive instances.

Implementation Selective Learning can be used with any loss function in a deep network. Here we describe the implementation details of the method.

For each label (attribute) a, we have some target distribution $P_T(a)$ and some batch distribution $P_B(a)$. If $P_T(a) = P_B(a)$, i.e. the batch distribution for a matches the target

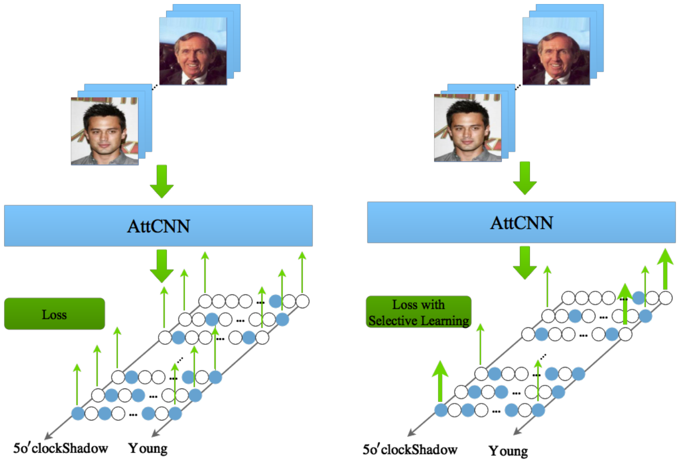


Figure 1: Visualization of the proposed Selective Learning (right) and normal learning without batch balancing (left). A blue node is a positive instance of an attribute and a white node is a negative instance of an attribute. The green upward pointing arrows indicate the back-propagation error. In a loss without Selective Learning (left), every attribute in every sample has the same weight, as indicated by the arrows all being the same thickness. In a loss with selective learning (right), we see that some attributes in some samples are not used for learning (they have no back-propagation arrows), and some samples have a higher weight (thicker green arrows) to account for imbalance. The two losses are demonstrated on *5o'clockShadow* and *Young*, two of the most imbalanced attributes in CelebA.

distribution, then the loss is calculated normally and the back-propagation error is unchanged. In practice, the SL batch is constructed by adding weights to every sample in the original batch.

Let $|B|$ be the size of the batch. If $P_B(a=1) > P_T(a=1)$, i.e. a is over-represented in the batch, the SL batch consists of all the samples from the original batch with weights to reduce the number of positive instances, and to increase the effective number of negative instances. Specifically, a random subset of $P_T(a=1)|B|$ positive instances are given a weight of 1, with all other positive instances given a weight of 0. The negative samples are each weighted by $\frac{P_T(a=0)}{P_B(a=0)}$ giving the negative samples the same effect as if they matched the target distribution.

Similarly, if $P_B(a=1) < P_T(a=1)$, i.e. a is under-represented in the batch, the SL batch consists of all the samples from the original batch with weights to reduce the number of negative instances, and to increase the effective number of positive instances. A random subset of $P_T(a=0)|B|$ negative instances are given a weight of 1, with all other negative instances given a weight of 0. The positive samples are each weighted by $\frac{P_T(a=1)}{P_B(a=1)}$ giving them the same effect as if they matched the target distribution.

Selective Learning allows the network to learn from adapted batches for each label (or attribute), so that all labels – even under-represented and over-represented labels – are learned as if the data matched a desired target distribution. Selective Learning is capable of both turning off back-propagation, and re-weighting the error for any attribute in any sample. Training of deep networks is done on a batch-by-batch basis, and so it makes sense to perform weighting and balancing at the batch-level. In MOON (Rudd, Gunther, and Boult 2016), each sample is re-weighted according to the target distribution, and so individual batch distributions are not taken into account, which leads to imbalances in training.

Figure 1 is a visualization of Selective Learning in comparison with normal multi-task learning. The left side shows a multi-task loss without Selective Learning, and the right side shows a multi-task loss with selective learning. Two attributes are highlighted: *5 o'clock shadow* and *young*. We can see that both are highly imbalanced, and with Selective Learning, each attribute is learned from its adapted batch, effectively removing the imbalance.

In our experiments, we apply Selective Learning to the proposed AttCNN, which uses a sigmoid cross-entropy loss. In the following section we demonstrate the effectiveness of Selective Learning on several challenging attribute datasets. We note that Selective Learning is extremely versatile and can be applied to any multi-label problem. It can easily be used for tasks other than facial attribute prediction, such as facial landmark detection (where nose points may be over-represented and ear points may be under-represented), body part localization (where some body parts may be occluded more than others), face verification across pose (where frontal is extremely over-represented) or any multi-task problem where the training data is imbalanced. Selective Learning can also be used to combine data from several different sources, with some, or no common labels for use in training a deep network, since it adaptively balances every batch for each label.

Experiments

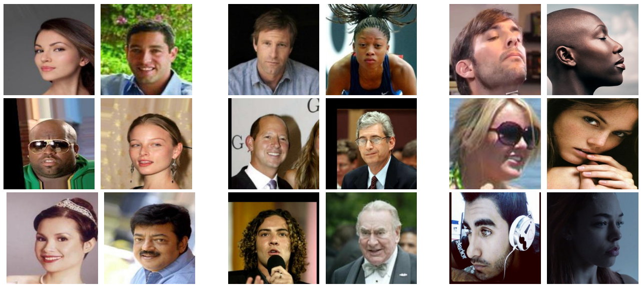
~~Data~~

We use three datasets in our experiments: CelebA, LFWA, and UMD-AED - a new evaluation dataset.

CelebA CelebA contains roughly $200,000$ images, with $160,000$ for training and $20,000$ each for validation and testing (Liu et al. 2015). Each image in CelebA is labeled with 40 binary attributes. CelebA consists of mostly frontal, posed images of celebrities. Sample images from CelebA can be seen in figure 2a.

LFWA LFWA is a much smaller dataset, containing only $13,143$ images labeled with the same 40 attributes from CelebA (Liu et al. 2015). LFW was originally created for face verification and attribute labels were later added creating LFWA. LFWA consists of still images of celebrities, so it is very similar to CelebA. Sample images from LFWA can be seen in figure 2b.

For each attribute, the percentage of positive labels is plotted for both LFWA and the CelebA train split in figure 3. We can see that LFWA exhibits some of the same imbalances as CelebA, though not to the same extreme, likely due to the size of the dataset. For instance, *black hair*, *blond hair*,



(a) CelebA (b) LFWA (c) UMD-AED

Figure 2: Sample images from (a) CelebA, (b) LFWA, and (c) UMD-AED.

heavy makeup, and *high cheekbones* are even more underrepresented in LFWA than in CelebA. So, if a model learned to prefer to output 0 for those attributes, then it would perform better on LFWA than on CelebA, without truly having learned a representation for those attributes.

University of Maryland Attribute Evaluation Dataset In order to better evaluate an attribute model, we constructed a new evaluation dataset, UMD-AED. UMD-AED contains 2,800 face images, each labeled with a subset of the 40 attributes from CelebA and LFWA. UMD-AED was collected in such a way that each attribute has the same number of positive and negative samples, hence why not every attribute is labeled in each image. Specifically, every attribute has 50 positive and 50 negative samples. Though UMD-AED is a small dataset, it is extremely effective at highlighting weakness in attribute models, as we will see in our experiments. With deep learning dominating almost every field in computer vision, most work is concerned with the quantity of data, rather than the quality. In our collection of UMD-AED, we focused on quality data which would effectively test the attribute representations learned by deep networks. By quality we mean that UMD-AED represents a wide variety of data, with low and high quality images, extreme lighting and poses, as well as different ages and skin tones, as can be seen in figure 2c.

UMD-AED was constructed by performing an image search with each of the 40 attributes as search terms, running the face detector from (Ranjan, Patel, and Chellappa 2015), and hand-curating the resulting face images. UMD-AED is much more representative of real-world data than CelebA or LFWA. As we will demonstrate in our experiments, to compare performance of attribute models on the test split of CelebA if they were trained on CelebA is optimistic. Evaluating models on UMD-AED will provide a much more unbiased metric for success of attribute prediction algorithms. If a model has learned a true representation for an attribute, then it can be expected to perform well on UMD-AED. We will make this dataset publicly available so that future work on attribute prediction can be evaluated on a balanced, real-world dataset.

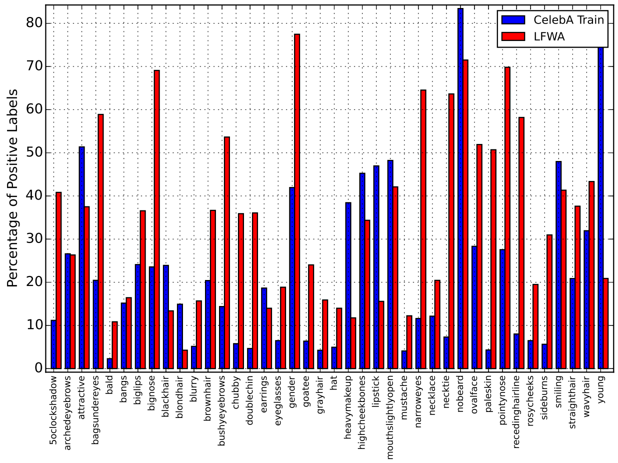


Figure 3: Percentage of positive attribute labels for CelebA train, and LFWA.

Method	**Accuracy**
LNet+ANet (Liu et al. 2015)	87.30
Walk and Learn (Wang, Cheng, and Feris 2016)	88.15
MOON (Rudd, Gunther, and Boult 2016)	90.94
AttCNN (Ours)	**90.97**

Table 2: Average attribute accuracy on the CelebA test set.

AttCNN

We train AttCNN directly on the CelebA training set, without any pre-training. As preprocessing steps, we subtract the training mean from each image, and take a random crop of 227x227 from the original image of size 256x256. The network weights are learned from scratch – starting with random initialization – using only the CelebA training set. AttCNN is trained for 22 epochs with batches of size 200, using a sigmoid cross-entropy loss.

We compare our AttCNN to the state of the art methods in table 2. AttCNN is comparable with the three previous state-of-the-art methods for attribute prediction: MOON (Rudd, Gunther, and Boult 2016), LNet+ANet (Liu et al. 2015), and Walk & Learn (Wang, Cheng, and Feris 2016). Table 2 shows that AttCNN outperforms all three methods on average. This is an impressive feat, as AttCNN has fewer than 6 million parameters, and is trained from scratch, whereas the most recent state-of-the-art, MOON, has 138 million parameters and is pre-trained on a large-scale object-recognition dataset, and both LNet+ANet and Walk & Learn are pre-trained on identification and verification data.

We argue that the success of AttCNN is due to training directly on attribute data. All three of the previous state-of-the-art networks have too many parameters to train directly from the 160,000 images in the train split of CelebA. With AttCNN as our base network, we demonstrate the effectiveness of the proposed Selective Learning approach in the following section.

Method	Average Accuracy
MOON$_{\text{Balanced}}$	**86.33**
AttCNN$_{\text{Balanced}}$	85.05

Table 3: Average attribute accuracy on the CelebA test set using the balanced networks.

Method	Average Accuracy
AttCNN$_{\text{P(a) = train}}$	**91.05**
AttCNN$_{\text{P(a) = test}}$	91.07

Table 4: Average attribute accuracy on the CelebA test set using AttCNN with target distributions given by the CelebA train ($P(a) = train$) and test ($P(a) = test$) sets. AttCNN$_{\text{P(a) = train}}$ is bolded as it is the new state-of-the-art on CelebA.

Selective Learning

We test the proposed Selective Learning method on CelebA, LFWA, and UMD-AED, and then compare with the state-of-the-art MOON method.

For our first experiment, we train AttCNN using Selective Learning with a balanced target distribution. We denote this model as AttCNN$_{\text{Balanced}}$. We train AttCNN$_{\text{Balanced}}$ for 22 epochs and we use batches of size 200 just as with the original AttCNN. Table 3 shows that AttCNN$_{\text{Balanced}}$ performs comparably to, though not as well as the balanced MOON on the CelebA test set. However, we believe this to be an artifact of the extreme imbalance in CelebA, which is not being effectively removed by MOON, as we will demonstrate in our experiments on LFWA and UMD-AED.

We perform two experiments adapting training of AttCNN to the CelebA training distribution (AttCNN$_{\text{P(a)=train}}$) and to the CelebA test distribution (AttCNN$_{\text{P(a)=test}}$), and present the results in table 4. Using Selective Learning with P(a)=train, we improve on the state-of-the-art for the CelebA test set with 91.05% average attribute prediction accuracy. This improvement highlights the need for label balancing at the batch-level as even slight changes in distributions within batches results in decreased performance. With P(a)=test, we see a small improvement, but we normally do not have access to the distribution of the test set. We provide this result to highlight the fact that the bias in CelebA extends from the training set to the validation and test sets. If the bias was less severe in the CelebA test set, we would see a larger improvement when adjusting for the testing distribution.

We tested the balanced and unbalanced MOON, as well as AttCNN, AttCNN$_{\text{P(a)=train}}$, and AttCNN$_{\text{Balanced}}$ on LFWA.

Method	Average Accuracy
MOON$_{\text{UnBalanced}}$	68.98
MOON$_{\text{Balanced}}$	70.49
AttCNN	71.21
AttCNN$_{\text{P(a)=train}}$	71.49
AttCNN$_{\text{Balanced}}$	**73.03**

Table 5: Average attribute accuracy on LFWA using MOON and AttCNN.

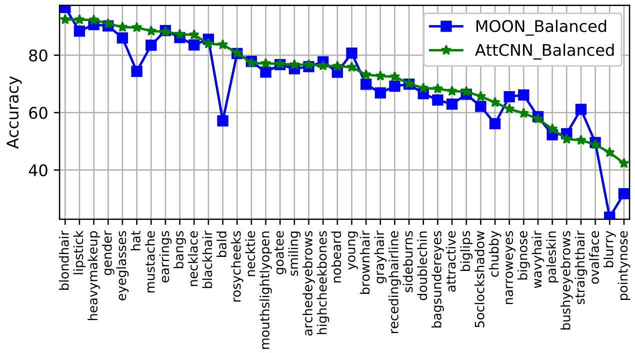


Figure 4: Results for AttCNN$_{\text{Balanced}}$ and MOON$_{\text{Balanced}}$ on LFWA. Best viewed in color.

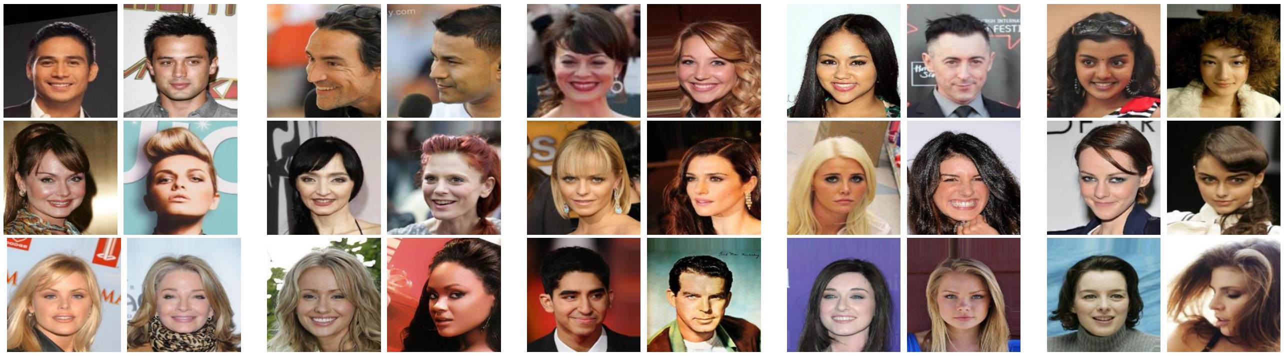
Method	Average Accuracy
MOON$_{\text{UnBalanced}}$	56.36
MOON$_{\text{Balanced}}$	59.46
AttCNN	66.85
AttCNN$_{\text{P(a)=train}}$	67.40
AttCNN$_{\text{P(a) = 0.5}}$	**71.11**

Table 6: Average attribute accuracy on UMD-AED using MOON and AttCNN.

The results are reported in table 5. AttCNN$_{\text{Balanced}}$ outperforms MOON by over 2.5%. We see that even just adapting each batch to align with the distribution of the training data (AttCNN$_{\text{P(a)=train}}$), outperforms both the unbalanced and the balanced MOON. Figure 4 shows the prediction accuracy for each attribute on LFWA using the balanced MOON and AttCNN$_{\text{Balanced}}$. We can see that the two curves are very close, except for a few attributes: *hat*, *bald*, *gray hair*, *chubby*, *blurry*, and *pointy nose*. We can see from figure 3 that these attributes were much more under-represented in CelebA than in LFWA, and so the bias of CelebA appears to have negatively affected the performance of MOON on LFWA. This same bias seems to have positively affected MOON on the CelebA test set, as seen in table 3.

For a less biased evaluation of the proposed attribute model, we test on UMD-AED, and these results are presented in table 6 as well as figure 6. We see that AttCNN$_{\text{Balanced}}$ outperforms MOON on almost every attribute. Here we truly see the effect of the extreme imbalance in CelebA on MOON, with many attributes achieving roughly 50% accuracy. In table 6 AttCNN$_{\text{Balanced}}$ outperforms the balanced MOON by a significant margin – over 11%, and AttCNN$_{\text{Balanced}}$ gives a 4% improvement over AttCNN. From this result, on a dataset with an even distribution for every attribute, and a better representation of real-world images, we can see that Selective Learning addresses the problem of multi-label balancing for deep networks trained on imbalanced data.

Our evaluation of AttCNN$_{\text{Balanced}}$ on UMD-AED not only highlights the effectiveness of our method, but also indicates areas for improvement. Both MOON and AttCNN$_{\text{Balanced}}$ struggle with *oval face*, *attractive*, *high cheekbones*, *arched*



(a) Oval Face (b) Attractive (c) High Cheekbones (d) Arched Eyebrows (e) Lipstick

Figure 5: Samples from CelebA train set with bad or ambiguous labeling for (a)*oval face*, (b)*attractive*, (c)*high cheekbones*, (d)*archedeyebrows*, and (e)*lipstick*. Positive labeled images are in the left columns, and negative labeled images are in the right columns. There is an obvious bias towards celebrities in this data. From (b), it is unclear what distinguishes an attractive person from an unattractive person.

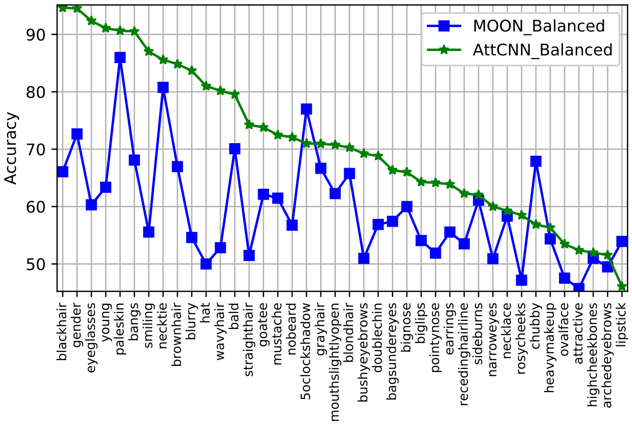


Figure 6: Results for AttCNN$_{\text{Balanced}}$ and MOON$_{\text{Balanced}}$ on UMD-AED. Best viewed in color.

eyebrows, and *lipstick*. All of these are very subjective attributes, with the exception of *lipstick*, and so there is likely some noise in the CelebA labels. Exploring the dataset, we find that this is exactly the case. We provide some sample images from CelebA in figure 5 to demonstrate the noisy labeling. Figures 5a-5e show samples with both positive and negative labels for the above attributes, with positive labels on the left and negative labels on the right. In many cases it is impossible to determine why one image has a positive label and another has a negative label. All of the subjects in figure 5c appear to have high cheekbones, but half of them are labeled as not having them. The negatively labeled samples in figure 5d have more arch in their eyebrows than the positively labeled samples. We argue that these subjective attributes should be removed from the attribute prediction task, as the goal is to accurately describe a face using attributes, and highly subjective attributes will not help with this cause.

Though *lipstick* is not a subjective attribute, we decided to perform the same analysis due to the poor performance of both MOON and AttCNN on this attribute. We found that the labels were just as noisy for *lipstick* as for the subjective attributes. Figure 5e shows samples from CelebA labeled with lipstick and not lipstick. None of the women in figure 5e are wearing lipstick, and yet half of them are labeled as such. Even with multi-label balancing using Selective Learning, there is no way to correct for this much noise in the labels. It is clear from these analyses that the next step in attribute prediction research is to collect a new large-scale dataset with more precise labels for training.

Conclusion

We introduced a novel Selective Learning technique for multi-label balancing of biased training data, and demonstrated its effectiveness on the problem of facial attribute prediction, improving on the state-of-the-art. Selective Learning adapts every training batch for each attribute according to a desired target distribution, allowing for balanced training with each batch. Since deep learning methods are trained on a batch-by-batch basis, it only makes sense to apply label balancing at the batch level. To test the capabilities of Selective Learning, we introduced a new evaluation dataset - UMD-AED. UMD-AED has an even distribution for each attribute, allowing for evaluation of attribute models in a balanced setting.

We introduced AttCNN, a deep network with fewer than 6 million parameters which is trained directly from CelebA. AttCNN outperformed the three previous state-of-the-art methods on CelebA, without pre-training on an external dataset. Training AttCNN with Selective Learning, we outperform the state-of-the-art on CelebA, LFWA, and UMD-AED, by 0.11%, 2.54%, and 11.65% respectively. The performance of our model on UMD-AED highlights the effectiveness of Selective Learning in allowing a deep network to learn a true representation of the data, rather than just the bias of the training set. UMD-AED will be made publicly available so that future research on attribute prediction can be evaluated on a balanced dataset.

Selective Learning can be applied to any multi-label problem which uses deep networks, including face verification across pose, facial landmark localization, and body part detection and localization, among many others. Though we demonstrate Selective Learning using a Sigmoid Cross-Entropy Loss, it can be used with any loss function. It can also be used to combine data from different sources with few or no common labels, since every batch is adapted for each label, no learning will occur for a particular label if it is not represented in the batch. Selective Learning is an extremely versatile method that can be applied to many problems, and will help ease the difficulty associated with multi-label balancing in large-scale datasets, which are needed to train deep networks.

Acknowledgment

This research is based upon work supported by the Office of the Director of National Intelligence (ODNI), Intelligence Advanced Research Projects Activity (IARPA), via IARPA R&D Contract No. 2014-14071600012. The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily represent- ing the official policies or endorsements, either expressed or implied, of the ODNI, IARPA, or the U.S. Government. The U.S. Government is authorized to reproduce and distribute reprints for Governmental purposes notwithstanding any copyright annotation thereon.

References

Abdulnabi, A. H.; Wang, G.; Lu, J.; and Jia, K. 2015. Multi-task cnn model for attribute prediction. *arXiv preprint*.

Bo, L.; Ren, X.; and Fox, D. 2011. Hierarchical matching pursuit for image classification: Architecture and fast algorithms. *NIPS*.

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

Cheng, H. T.; Sun, F. T.; Griss, M.; Davis, P.; Li, J.; and You, D. 2013. Nuactive: Recognizing unseen new activities using semantic attribute-based learning. *ICMSAS*.

Duan, K.; Parikh, D.; Crandall, D.; and Grauman, K. 2012. Discovering localized attributes for fine-trained recognition. *CVPR*.

Ehrlich, M.; Shields, T. J.; Almaev, T.; and Amer, M. R. 2016. Facial attributes classification using multi-task representation learning. *CVPR*.

Farhadi, A.; Endres, I.; Hoiem, D.; and Forsyth, D. 2009. Describing objects by their attributes. *CVPR*.

Fu, Y.; Guo, G.; and Huang, T. S. 2010. Age synthesis and estimation via faces: A survey. *PAMI*.

Ganin, Y., and Lempitsky, V. 2014. Unsupervised domain adaptation by backpropagation. *arXiv preprint*.

Gong, B.; Shi, Y.; Sha, F.; and Grauman, K. 2012. Geodesic flow kernel for unsupervised domain adaptation. *CVPR*.

Gopalan, R.; Li, R.; and Chellappa, R. 2011. Domain adaptation for object recognition: An unsupervised approach. *ICCV*.

Ho, H., and Gopalan, R. 2014. Model-driven domain adaptation on product manifolds for unconstrained face recognition. *IJCV*.

Huang, C.; Li, Y.; Loy, C. C.; and X, T. 2016. Learning deep representation for imbalanced classification. *CVPR*.

Hwang, S. J.; Sha, F.; and Grauman, K. 2011. Sharing features between objects and their attributes. *CVPR*.

Jayaraman, D.; Sha, F.; and Grauman, K. 2014. Decorrelating semantic visual attributes by resisting the urge to share. *CVPR*.

Jia, Y.; Shelhamer, E.; Donahue, J.; Karayev, S.; Long, J.; Girshick, R.; Guadarrama, S.; and Darrell, T. 2014. Caffe: Convolutional architecture for fast feature embedding. *arXiv preprint*.

Kumar, N.; Belhumeur, P.; and Nayar, S. 2008. Facetracer: A search engine for large collections of images with faces. *ECCV*.

Kumar, N.; Berg, A. C.; Belhumeur, P. N.; and Nayar, S. K. 2009. Attribute and simile classifiers for face verification. *ICCV*.

Kumar, N.; Berg, A. C.; Belhumeur, P. N.; and Nayar, S. K. 2011. Describable visual attributes for face verification and image search. *PAMI*.

Levi, G., and Hassner, T. 2015. Age and gender classification using convolutional neural networks. *CVPR*.

Liu, Z.; Luo, P.; Wang, X.; and Tang, X. 2015. Deep learning face attributes in the wild. *ICCV*.

Lu, B.; Chellappa, R.; and Nasrabadi, N. M. 2015. Incremental dictionary learning for unsupervised domain adaptation. *BMVC*.

Ng, C. B.; Tay, Y. H.; and Goi, B. M. 2012. Vision-based human gender recognition: A survey. *arXiv preprint*.

Patel, V. M.; Gopalan, R.; Li, R.; and Chellappa, R. 2015. Visual domain adaptation: A survey of recent advances. *Signal Processing Magazine*.

Qui, Q.; Patel, V. M.; Turaga, P.; and Chellappa, R. 2012. Domain adaptive dictionary learning. *ECCV*.

Ranjan, R.; Patel, V. M.; and Chellappa, R. 2015. Hyperface: A deep multi-task learning framework for face detection, landmark localization, pose estimation, and gender recognition. *arXiv preprint*.

Rudd, E.; Gunther, M.; and Boult, T. 2016. Moon: A mixed objective optimization network for the recognition of facial attributes. *ECCV*.

Saenko, K.; Kulis, B.; Fritz, M.; and Darrell, T. 2010. Adapting visual category models to new domains. *ECCV*.

Shekhar, S.; Patel, V. M.; Nguyen, H. V.; and Chellappa, R. 2013. Generalized domain-adaptive dictionaries. *CVPR*.

Wang, J.; Cheng, Y.; and Feris, R. S. 2016. Walk and learn: Facial attribute representation learning from egocentric video and contextual data. *CVPR*.

Yang, M.; Zhang, L.; Feng, X.; and Zhang, D. 2011. Fisher discrimination dictionary learning for sparse representation. *ICCV*.

Zhang, N.; Paluri, M.; Ranzato, M. A.; Darrell, T.; and Bourdev, L. 2014a. Panda: Pose aligned networks for deep attribute modeling. *CVPR*.

Zhang, Z.; Luo, P.; Loy, C.; and Tang, X. 2014b. Facial landmark detection by deep multi-task learning. *ECCV*.

Zheng, J.; Jiang, Z.; Chellappa, R.; and Phillips, J. P. 2014. Submodular attribute selection for action recognition in video. *NIPS*.

Zhong, Y.; Sullivan, J.; and Li, H. 2016. Face attribute prediction using off-the-shelf cnn features. *ICB*.