Experiments with Reducing Algorithmic Bias in Gender Classification Systems

Emma Dwight
Mehul Raje
Agasthya Shenoy

Problem Statement and Motivation

Face recognition software is now built into most smart phones and several companies have released commercial software that perform automated facial analysis. Some such products include Face++, Google Image search and even automatic recognition of people in Facebook and Apple photo libraries. However, much of this technology is plagued by shortcomings, especially with respect to women or people of colour. We have seen news articles about iPhones not recognising people of colour or a simple Google search of a black man's name returning more references to criminal activity or crime reporting. The latest gender classification report from NIST also shows that the algorithms they evaluated performed worse for female-labeled faces than male-labeled faces (Ngan et al., 2015).

Most large scale face collection depends on face recognition algorithms. This means that any systematic error found in face detectors will inevitably affect the composition of the benchmark. As an example, the LFW dataset composed of celebrity faces is 77.5% male and 83.5% White. In response, the IARPA has released the IJB-A dataset which does not use face detectors to select images. An Algorithmic Justice League (AJL) has also been set up at MIT to combat 'exclusionary experiences and discriminatory practices' caused by algorithms.

A 2012 study (Klare et. al.) on mug shots found that a facial recognition algorithm trained exclusively on either African American or Caucasian faces recognized members of the race in its training set better than those of other races. The effect of the composition of the training set used surely matters.

Overarching Aim
This project recognises that we live in a world with terribly imperfect datasets. We demonstrate how imbalance among racial groups in training data leads to poor performance in models, and we also investigate different ways that imperfect datasets can be used well and augmented in ways that improve accuracy. Our overarching aim is to improve prediction accuracy of gender classification models using images of White, Black and Asian faces.

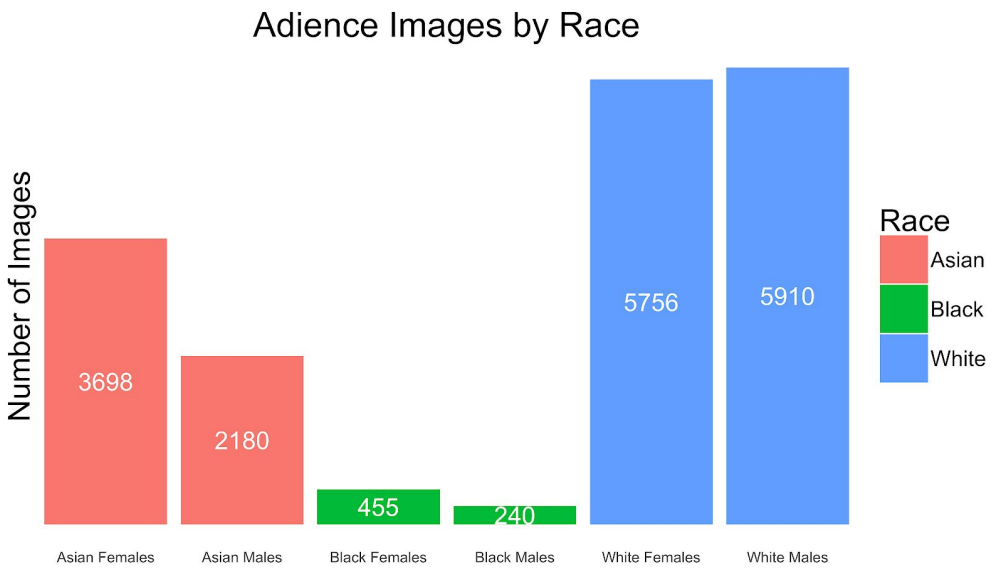
Sub-goals
- Compare accuracy of models trained on skewed and balanced datasets
- Explore data augmentation techniques to counteract imbalance in training data

- Use transfer learning to determine whether pre-trained image classification models can improve their accuracy when re-trained for a different task.

Introduction and Description of Data

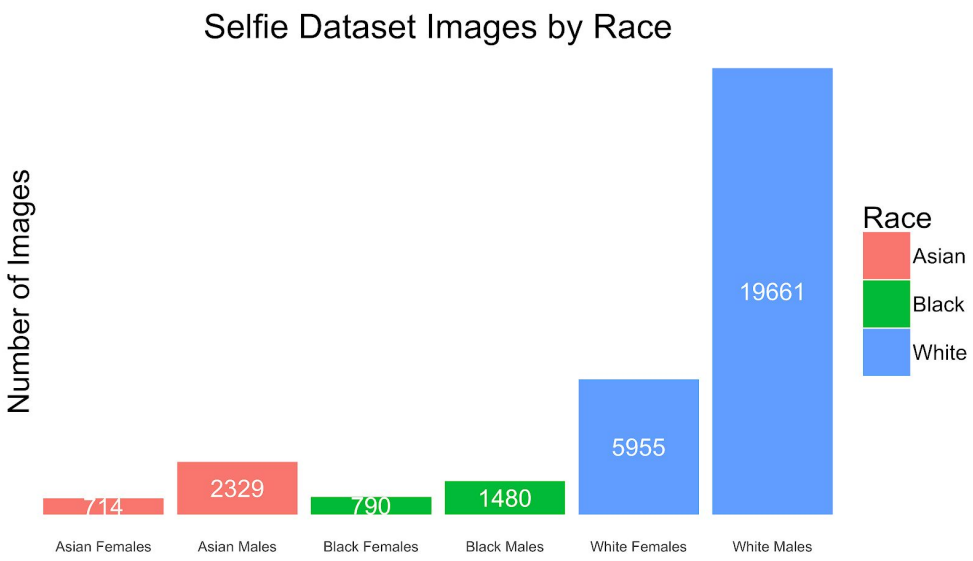
Adience Dataset

We have chosen to use the Open University of Israel's Adience dataset which "attempts to capture all the variations in appearance, noise, pose, lighting and more, that can be expected of images taken without careful preparation or posing." (www.opeau.ac.il). The 18,239 images are of faces in the wild labeled by age, gender, race/ethnicity, each cropped to 816x816 pixels. The images are of around 2,000 different people. We chose to further crop the photos to 224x224 pixels, for reasons of computational cost. Ages vary from infants to elderly, and the quality of the images is generally very poor. This is one of a small number of widely-used, well-labelled and easily available datasets used for facial recognition work, but it contains very few images of black people.

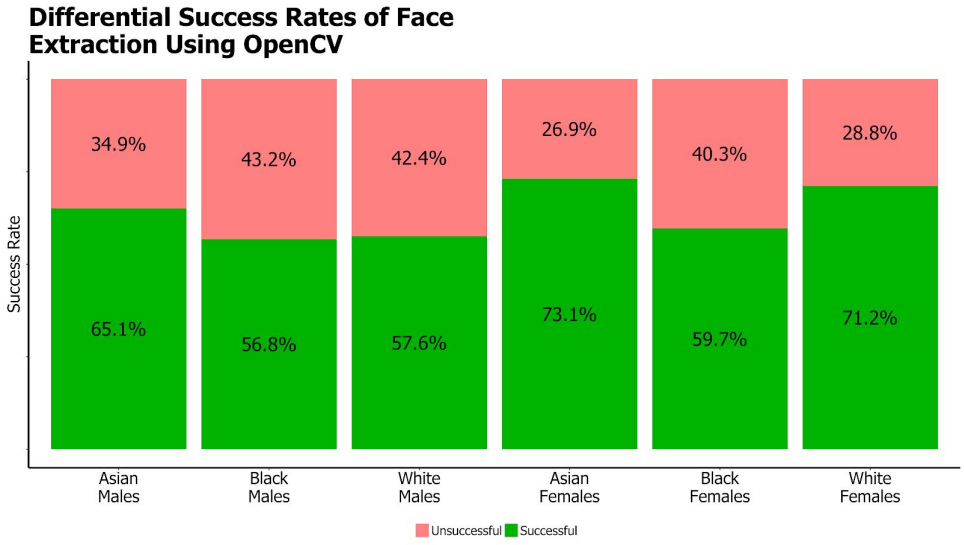


Selfie dataset

We have also created a test set of images sampled from the UCF Selfie Data Set. The reasoning for this is that we wanted to see if our model worked well on out-of-sample images from a different context. These 46,836 selfie images (originally scraped by UCF researchers from Instagram photos with a #selfie hashtag) were cropped to 816x816 using code supplied by Kally Wu. Cropping code used the opencv library to identify faces in pictures and center those faces within an 816x816 square.



The technology that underlay our pre-processing of the selfie dataset was the opencv[1] software we used to identify faces within images and crop around those faces. We found these extractions to succeed less often on faces of darker skin, which is problematic, and further increased the disparity between the subgroups in this selfie dataset. We should also note that sometimes, the cropped images opencv produced did not contain a face at all, which will limit our test-set accuracy.



[1] OpenCV (Open Source Computer Vision Library) https://opencv.org/

Literature Review

Age and Gender Classification using Convolutional Neural Networks[2]
This paper demonstrates that a convolutional neural network (CNN) with three convolutional layers and two fully-connected layers and a small number of neurons can dramatically increase performance in age and gender classification of unfiltered photographs. Although there is a relatively small existing dataset of faces with labeled age and gender, researchers show that a "shallow" CNN can still improve classification accuracy. We draw inspiration from the architecture these authors use to create our own made-from-scratch CNN, which we also then compare to transfer learning approaches on state-of-the-art pre-trained neural networks.

Gender Shades: Intersectional Accuracy Disparities in Commercial Gender Classification[3]
This paper uses the Fitzpatrick Skin Type classification system to characterize the gender and skin type distribution of IJB-A and Adience facial analysis benchmarks. It is demonstrated that these datasets are largely composed of lighter-skinned people. This paper goes on to introduce a new dataset of 1270 faces in which skin types are more phenotypically balanced than existing benchmarks. It also introduces the first intersectional demographic and phenotypic evaluation of face-based gender classification accuracy. These authors' discussion of balance in training datasets motivated us to also explore how balance and imbalance in training sets affects model performance.

Error rates	TYPE I	TYPE II	TYPE III	TYPE IV	TYPE V	TYPE VI
Microsoft	1.7%	1.1%	3.3%	0%	23.2%	25.0%
IBM	5.1%	7.4%	8.2%	8.3%	33.3%	**46.8%**

These researchers show that state-of-the-art gender classifiers perform poorly on darker-skinned faces.

---

[2] G. Levi and T. Hassner. Age and gender classification using convolutional neural networks. In 2015 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), pages 34–42, June 2015.
[3] Buolamwini, J. & Gebru, T.. (2018). Gender Shades: Intersectional Accuracy Disparities in Commercial Gender Classification. Proceedings of the 1st Conference on Fairness, Accountability and Transparency, in PMLR 81:77-91

Algorithmic decision making and the cost of fairness[4]

A machine learning system (COMPAS) system is used to classify the risk of re-offence of criminals, and used in many states for parole decisions and sentencing. It takes into account many factors and predictors (without explicitly including race) but was shown by some researchers to have higher rates of errors for people of color. Specifically, black defendants are substantially more likely than white defendants to be incorrectly classified as high risk, and among defendants who ultimately did not reoffend, black people were more than twice as likely as white people to be labeled as high-risk. This takes place in a larger conversation: What does algorithmic fairness mean? Is it about fair data, fair processes (like blinded algorithms) or fair performance?

The authors show that formal fairness constraints would require race-specific thresholds for risk, but applying an equal threshold for risk across all people leads to different rates of risk-classification by racial group. In our own context, achieving equal levels of gender classification could be achieved by artificially worsening predictions on white faces, for example. This broader conversation is important to keep in mind. We focus on improving the quality and fairness of the datasets with the hope that this will improve the fairness of our predictive accuracy.

Modeling Approach

Convolutional Neural Network Model Architecture

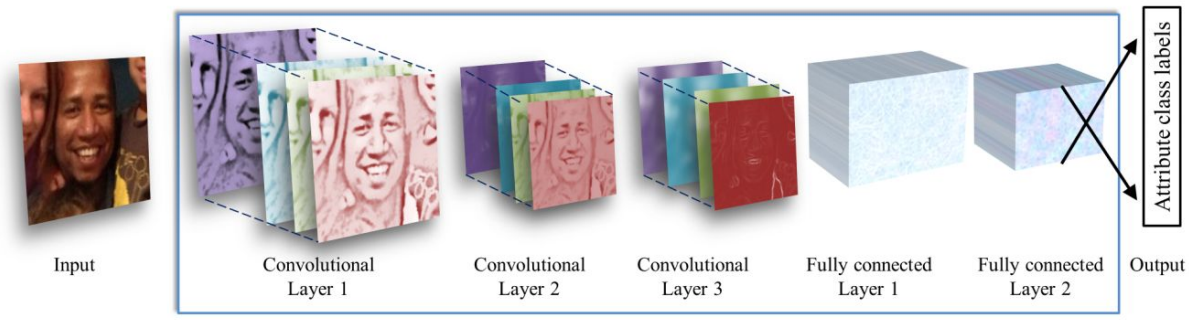
The first thing we wanted to do was to build our own CNN model from scratch before using any pre-trained networks, because we wanted to control exactly which images that network saw in training. For the architecture of this network, we closely followed the architecture used by Levi and Hassner in their published models on the adience dataset.[5] They built models that predicted both age and gender, but we focus in this project only on gender classification.

Convolutional Neural Networks are able to exploit the structure of images: pixels that are close to each other are more similar (on average) than pixels that are far apart, and objects in images are made up of smaller objects/pieces (for example, an image of a face is made up of eyes, head shape, a nose, and a mouth). A convolutional layer usually reduces the feature space of the image, but also applies a number of different filters designed to pick up on certain features of the input. Using several convolutional layers in sequence allows the network to “learn” more and more abstract/large features in each of the different filter layers. After three such layers, a “fully-connected” layer is used which then takes all of the different features extracted by the convolution and narrows down the feature space again, focussing here only on the most important features that distinguish between male and female faces. Finally, we reach a two-class

[4] Corbett-Davies, S., Pierson, E., Feller, A., Goel, S., & Huq, A. (2017). Algorithmic decision making and the cost of fairness. To appear in Proceedings of KDD'17

[5] G. Levi and T. Hassner. Age and gender classification using convolutional neural networks. In 2015 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), pages 34–42, June 2015.

output (male or female?) that uses a sigmoid activation function to assign probabilities for the face being male or female. This general architecture is diagrammed below:



Another piece of the architecture is the max-pooling layers that immediately follow the convolutional layers. These reduce the input size further, making computation of the final fully connected layer more feasible, while losing little information. Below is the actual architecture of our from-scratch network:

Layer (type)	Output Shape	Param #
conv2d_7 (Conv2D)	(None, 55, 55, 96)	14208
max_pooling2d_7 (MaxPooling2	(None, 27, 27, 96)	0
conv2d_8 (Conv2D)	(None, 27, 27, 256)	614656
max_pooling2d_8 (MaxPooling2	(None, 13, 13, 256)	0
conv2d_9 (Conv2D)	(None, 11, 11, 384)	885120
max_pooling2d_9 (MaxPooling2	(None, 5, 5, 384)	0
dense_9 (Dense)	(None, 5, 5, 512)	197120
dropout_5 (Dropout)	(None, 5, 5, 512)	0
dense_10 (Dense)	(None, 5, 5, 512)	262656
dropout_6 (Dropout)	(None, 5, 5, 512)	0
flatten_3 (Flatten)	(None, 12800)	0
dense_11 (Dense)	(None, 8)	102408
dense_12 (Dense)	(None, 2)	18

Total params: 2,076,186
Trainable params: 2,076,186
Non-trainable params: 0

With CNNs, one should always be concerned about the dangers of overfitting: if there is something irrelevant that happens to distinguish well between images of different classes (say, something in the background or perhaps the presence of earrings) then the network will only learn this feature and won't generalise well. One technique to counter this is using a "dropout" layer. We place a dropout layer after each of the dense (fully connected) layers. We give them a certain probability, and then during training, they "drop" (set to zero) each of their inputs (the outputs of the previous layer) with that probability. This forces the network to learn many

features that are able to distinguish between male and female faces, rather than overfit to just a few.

Sampling Methods

With a fixed sample size of 500 images (500 each for training and validation, from the overall 18,239 images in the adience dataset), we employed two sampling methods:

- Proportional Method: Number of samples of each gender and race proportional to the skew in the Adience data.
- Balanced Method: Equal numbers of faces of each race and gender.

We wanted to experiment with these two sampling methods in order to demonstrate how different demographic compositions of image training data affect predictive accuracy.

Data Augmentation Methods

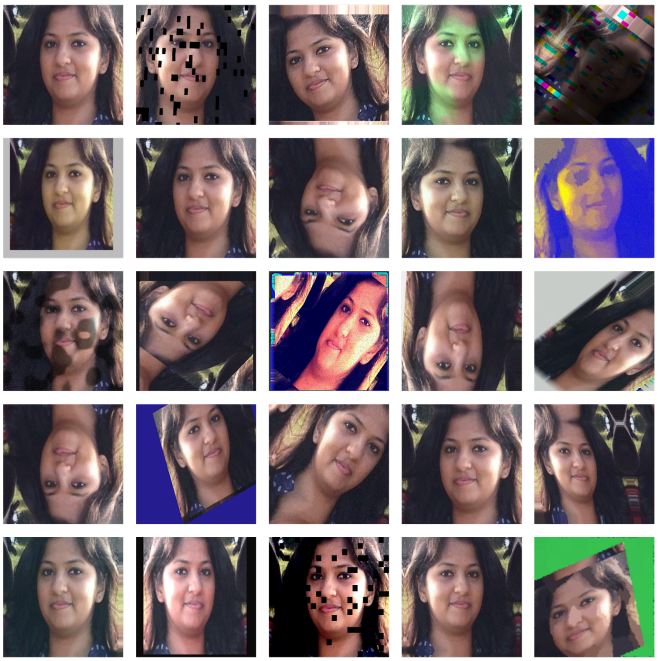
The extreme imbalance in the training data limits accuracy, especially for the minority subgroups. To artificially produce more image data we use augmentation techniques.[6] We experiment with "gentle" image augmentation using the horizontal flip, shear, and zoom options provided by keras, and with a more "extreme" range of transformations provided by the ImgAug package.[7]

Below is an example of what we term "gentle" augmentation:

[6] Luis Perez and Jason Wang. The effectiveness of data augmentation in image classification using deep learning. 2017. Available at http://cs231n.stanford.edu/ reports/2017/pdfs/300.pdf

[7] Alexander Jung. Imgaug: a library for image augmentation in machine learning experiments. Available at https://github.com/aleju/imgaug

By contrast, here is an example of what we term “extreme” augmentation:



In the first table, you can see that the features are stretched a little, and that horizontal flips are applied. But these images are all highly correlated. Though the pixels themselves are in slightly different positions, these augmentations don’t really force the network to learn which are most important and develop a real robustness to new, unseen data that will help it to improve its test set accuracy. The more extreme augmentation takes things a little too far in the opposite direction, with random color distortions, more dramatic zooms, stretches, and rotations, and some random drop-out and black-out of parts of the images. These images are much less correlated with each other, even though our code requires that only a few transformations are applied to each image. Remarkably though, the human eye can still tell that we are looking at a female face in most of them, so there should be something in the images that the network can learn from while reducing overfitting.

The augmentation strategy we ultimately used falls between these two extremes.

Project Trajectory, Results, and Interpretation

Our initial plan was to work mostly with pre-trained models and employ transfer learning to teach them our gender classification task. We came to realise though, that some of these pre-trained image classification models are trained for goals and on data vastly different than our own, and so their performance varied wildly. We also found ourselves more interested in the dramatic imbalance of images in our training set, and because we didn’t know exactly what images the pre-trained network had seen, we wouldn’t be able to disentangle the effect of our own training images on these networks from the images they’ve already seen.

Choosing instead to focus on training a smaller, more specialised network from scratch allowed us to demonstrate far more clearly the effects of differently-balanced training datasets, and the effects of our data augmentation strategy.

We created a test set of the 16,239 adience images that didn't appear in our two random samples. These are the "leftover" images that our model has not yet seen.

Overall Validation and Test Accuracy

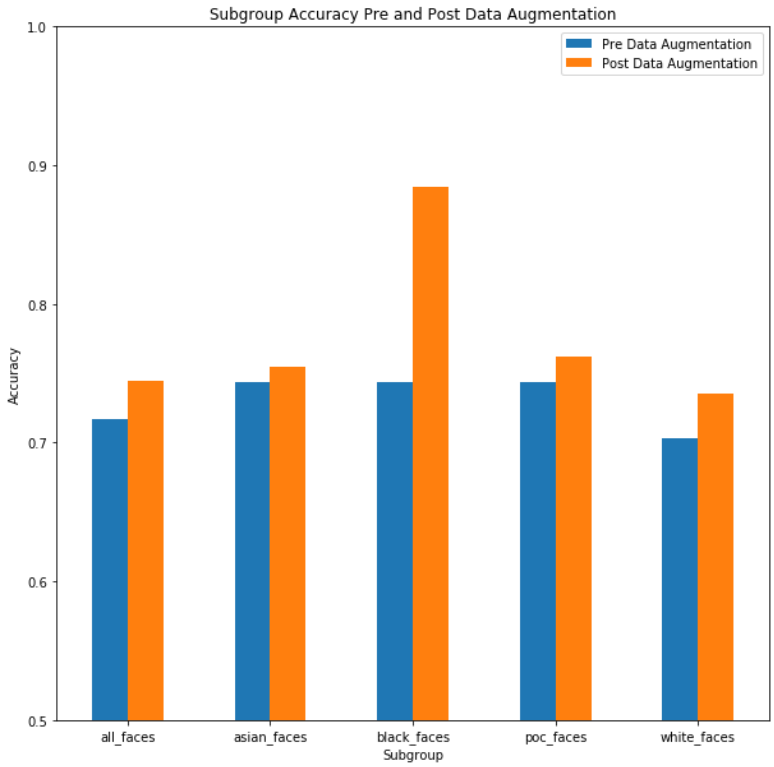
Model	**Validation Accuracy (overall)**	**Test Accuracy (overall)**
From-scratch	0.784	0.777
VGGFace Transfer Learning	0.948	0.937
InceptionV3 Transfer Learning	0.728	0.736

The VGGFace Transfer Learning model has the highest test accuracy. It should be noted, however, that these comparisons are not exact; that is, each model has vastly different architecture and was trained on different sized training sets. VGGFace, for example, was trained on a 2.6 million face dataset with 2000+ classes, and InceptionV3 was trained on 14.7 million images with over 1000 classes. When applying transfer learning techniques, each of these networks were trained with over 13 million trainable parameters.

In this context, the accuracy score for our from-scratch model, trained on roughly 2000 images (reduced to 224x224 resolution) with only two classes and 2.7 million trainable parameters, seems much more competitive.

This suggests that specializing the architecture of a smaller neural network for a specific task can yield better results than repurposing a large network designed for a tangentially related task.

Effect of data augmentation on gender classification accuracy in the from-scratch model
Artificially balancing datasets using data augmentation yields gains in accuracy. Our results show improved performance in all subgroups when the model was trained with augmented images of minority faces.



Accuracy on Subgroups

Below we compare performance on both the Adience and Selfie test sets between the two models with the highest overall accuracy.

	VGGFace		Made from Scratch (post augmentation)	
Race	**Selfie Data**	**Adience Test Set**	**Selfie Data**	**Adience Test Set**
Asian	0.659	0.948	0.437	0.755
Black	0.395	0.892	0.562	0.884
White	0.689	0.934	0.403	0.736

Conclusions and Possible Future Work

We showed how imbalanced training data can lead to poor and biased model performance, and demonstrated ways that researchers can do the best they can when using such training sets (using image augmentation). In broader discussions about how to reduce racial bias in facial recognition, the first plan-of-attack should surely be to construct more balanced training datasets in the first place, a goal which our research only tangentially informs.

Our small from-scratch model also performed admirably when compared to state-of-the-art image classification models (VGG Face and Inception), considering how few layers it had and

how few images it saw, which indicates that specialised networks that are well-designed for their particular task (in our case, gender classification) punch well above their weight.

Recommendations for Future Research

A natural extension of our project would be to experiment with how using adversarial images might increase the robustness of our gender classification model. Similarly to our approach to data augmentation, these might be generated only for non-white faces in order to bolster the accuracy of these predictions.

Borrowing ideas from boosting, and from strategies discussed throughout the course, we think that increasing the weights of images for non-white faces, or re-training the model on non-white faces, might help to lift the accuracy for these predictions. If there are only a few images of that particular subgroup, telling the model to pay extra close attention to these, through higher weights or some other method, might help to narrow the gaps in predictive accuracy.

Our biggest recommendation for practitioners is to over-sample training images from people of color groups in order to build models that work well for them (sample at rates not proportional to the population). If someone were building a facial recognition model designed to be used on Harvard students, for example, creating a training set that is, at the very least, representative of the racial and gender demographics of this population is a great starting point, but oversampling from people of color from whom few images would otherwise be sampled would be ideal. Adding another white face most likely won't teach the network anything new, but adding another non-white face adds a lot of information as the network hasn't seen many such faces.

We also urge future researchers to publish accuracy rates of their models for different racial groups in their test set, not just overall accuracy rates. We hope that this would encourage the researchers to be more mindful of any disparities in their model performance, and allow others to hold them more accountable. More awareness and discussion of these issues would hopefully lead to more research about how to improve algorithmic fairness in the future.