# Identifying and removing bias from Neural Nets

Abhimanyu Yadav
Columbia University
New York, NY
ay2412@columbia.edu

Siddharth Ramakrishnan
Columbia University
New York, NY
spr2310@columbia.edu

## 1. Introduction

Project source code. Image datasets are known to contain a lot of biases and these biases are learned by neural nets when trained on these datasets. If the contribution from these biased variations become overwhelming, the neural network loses its ability to generalize well on unseen data, especially from different domains. For instance, if we used the IMDB dataset for a task such as age prediction, we observe that the dataset is biased by having younger age groups dominated by women and having older age groups dominated by men. Using traditional training techniques, this bias is picked up by neural net. Existing literature such as those from [3] have demonstrated how Joint Learning and Unlearning (JLU) has been used to make the model resilient to such biases thereby improving its generalizing power. A quick look at the architecture below shows how the authors used a VGG-M base architecture [9] to carry out this task and use the embeddings of the feature representations (highlighted in yellow) to demonstrate the bias learned by the network. If we observe the embeddings on the left in Fig. 2 , we see that by only focusing on the primary task, the neural net has implicitly learned to differentiate on gender even though it was never exposed to gender labels. The neural network picks up on gender information so that when it sees a female, it predicts lower age and when it sees a male it predicts a higher age and ends up over-fitting on the IMDB dataset, but performs bad in generalization. On unlearning using JLU, the embeddings as shown in Fig. 2 show no learning of gender and thus they would generalize well. In this paper, we implement this approach from the ground up on our own CNN architecture and apply it on a dataset we have created by scraping the web. We use sampling bias to expose the neural network to bias and then use unlearning. Furthermore, we test the effects of the hyperparameter $\alpha$ on the embeddings.

## 2. Related Work

Dataset bias is a heavily studied topic and prevalent in many well known datasets. Image datasets contain biases

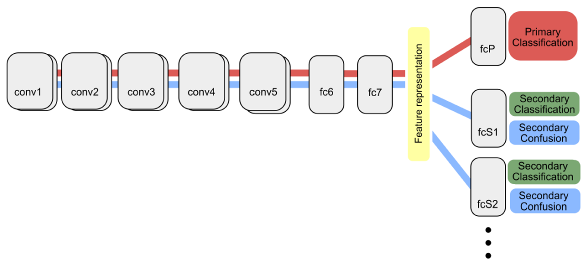

Figure 1. JLU Architecture [3]

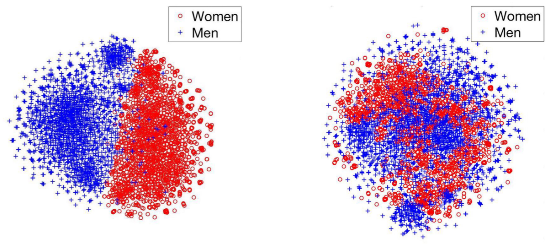

Figure 2. Gender learned (left) and gender unlearned (right) [3]

[11],[10] and the removal of such biases will improve model generalization. A lot of research has been conducted to remove bias caused by spurious variations. The Visual Geometry Group at Oxford arXived their work [3] on removing bias from neural network embeddings. They propose a systematic framework for identifying and removing bias from datasets. A problem that they formulate in the framework is to do gender agnostic age classification on IMDB dataset, and we plan to carry out this implementation but for a dataset that we are creating, *Cricbuzz* dataset [2] supplemented with more data from *CricAu* [1] dataset and with our own convolutional neural network architecture.
The datasets consists of images of cricketers, labelled with gender and age. This labelled data is obtainable on their corresponding websites subjected to scraping, cleaning and preprocessing. [3] uses secondary datasets that have spu-

rious variations such as pose affecting the age classification for which the authors used the AFLW [8] dataset and the IMDB dataset itself. Neural networks are extensively used in computer vision for tasks such as predicting age and gender from images. Convolutional neural networks were used to learn representations to estimate age and gender, where the data is limited.[7]. The proposed methodology described in the later sections draw inspiration from this work. Multi task learning approaches have shown a lot of promise in solving challenges like this in various domains. Improvements in generalization have been observed in fields like signal processing [4],[6] natural language processing [5]. The aim of this paper is to use a multi task neural network to solve the primary task of age classification and unlearning the secondary task of gender classification.

## 3. Approach

### 3.1. Data Preparation

We create our own dataset for the task. The data was scraped from two popular cricket websites namely Cricbuzz and Cricau, which contain the most recent information of cricketers. Scraping was carried out using Scrapy in which, spiders were written to crawl these sites and obtain images and metadata. To obtain the age and gender information, regular expression and tag processing was used. One of the most significant challenges faced was verifying the ground truth age and gender based the scraped values. The images were not time-stamped and as a result there were many examples where there were discrepancies between the true age and the age of the cricketer when the image was taken. To alleviate this, we use Microsoft Face API which looks at an image and predicts the age and gender (among other things). By inspecting the scraped results and the API results, the authors have heuristically determined the age and gender of the person by allowing a margin of error of around 5 years. Any spurious results post this processing, was dealt with manually. Consequently, the resulting dataset has 5011 examples with around 500 women. The dataset is highly skewed in favour of males. The experiments are carried out on three datasets derived from this distribution and are listed below.

1. **Original Dataset-OD**
   This contains 4511 males and 500 females. We can see that this distribution is more spread on males as opposed to females. Fig. 3

2. **Biased Dataset-BD**
   This contains 1017 males and 482 females. We can see that this distribution contains only females from ages 16-35 and males from 40-70. A complement of this dataset is created for testing the model to see if it unlearns this bias. Fig. 4

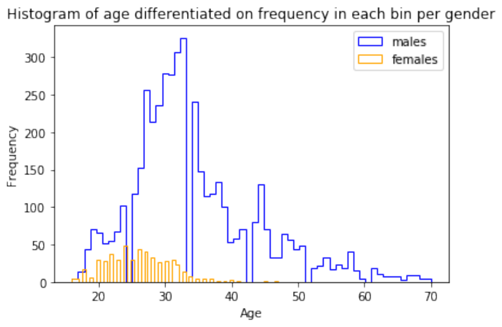

Figure 3. Original Dataset Histogram

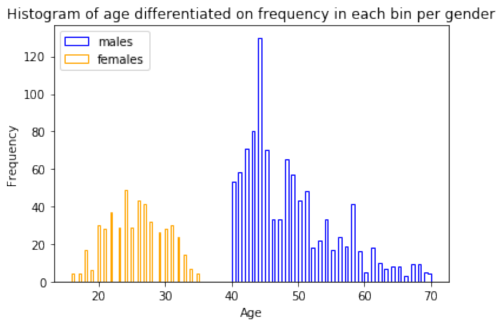

Figure 4. Biased Dataset

3. **Equally Balanced Dataset-EBD**
   Here we have an equal number of males and females in each age group (478 each). Since the histograms overlap exactly over each other, we just have one color representing both genders. Fig. 5

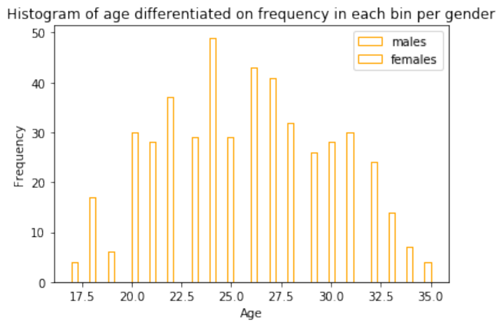

Figure 5. Balanced Dataset

### 3.2. Method

We use the joint learning and unlearning approach as highlighted in this [3] paper. As mentioned in section-3.1, our dataset consists of images labelled with age and gender. We use a multi-task learning architecture as shown in Fig. 6. The input to the network is a 96 X 96 X 3 image and there

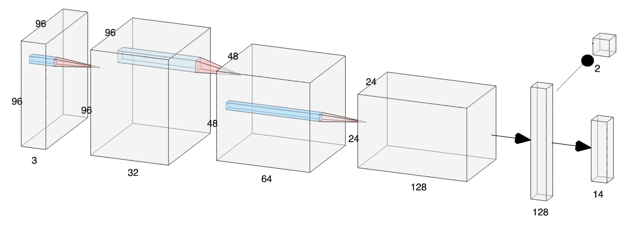

Figure 6. CNN Architecture

two outputs, gender prediction and age prediction. The network predicts gender out of male and female and has soft predictions in the output such as 0.6, 0.4 determining the probability the network assigns to the gender of the image. For age, we divide the age group 16-72 into 14 bins of length of 4 years each. The neural network predicts similar to gender by assigning soft labels to each of these 14 categories. It is important to note that although we have age and gender labels for the same image, the method described would work even if we had an age only dataset and a gender only dataset.

Let us denote the 128-dimensional output vector as $\theta_{repr}$. The neural network is therefore taking in a 96 X 96 X 3 $=$ 27648 dimensional feature vector and learning $\theta_{repr}$. The learning is guided by the loss functions as described below. Let $X_{age}, y_{age}$ denote the input image and age label from the age dataset. Let $X_{gen}, y_{gen}$ denote the input image and gender label from the gender dataset. The age label and gender label would be one hot vectors. Let $y_{\text{conf}}$ = (0.5, 0.5) be a gender confusion constant label. Let $p^k_{age}$ denote the probability the network assigns to the bin $k$ where $k \in [14]$ for age classification. Similarly, let $p^k_{gen}$ denote the probability the network assigns to the bin $k$ where $k \in [2]$ for gender classification. $\theta_{age}$ be the learned age classifier $\theta_{gen}$ be the learned gender classifier. With this notation, we define 3 losses:

$$L_{age}(X_{age}, y_{age}, \theta_{repr}, \theta_{age}) = \sum_{k=1}^{14} \mathbb{1}[y_{age} = p^k_{age}] \log(p^k_{age})$$

$$L_{gen}(X_{gen}, y_{gen}, \theta_{repr}, \theta_{gen}) = \sum_{k=1}^{2} \mathbb{1}[y_{gen} = p^k_{gen}] \log(p^k_{gen})$$

$$L_{\text{confusion}}(X_{gen}, y_{\text{conf}}, \theta_{repr}, \theta_{gen}) = \sum_{k=1}^{2} \mathbb{1}[y_{\text{conf}} = p^k_{gen}] \log(p^k_{gen})$$

$L_{age}$ is a simple softmax loss which would make sure that the learned representation $\theta_{repr}$ is such that age classification is very accurate. Similarly, $L_{gen}$ would ensure that learned representations $\theta_{repr}$ would make gender classification accurate. The interesting loss function to consider is

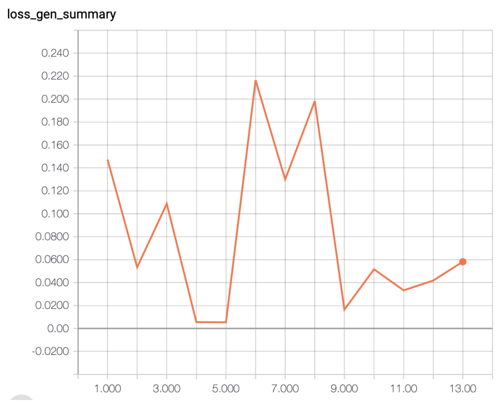

Figure 7. Fluctuating loss in gender prediction

$L_{\text{confusion}}$ which would ensure that $\theta_{repr}$ is such that the network is always confused about gender. It does so by always supplying the correct label as (0.5, 0.5) in the final softmax. Clearly, $L_{gen}$ and $L_{\text{confusion}}$ stand in direct opposition of each other and thus can't be optimized together. Also, recall from section-1 that our goal is to do gender agnostic age classification and hence we define two tasks which we optimize for alternatively. Primary task is to minimize $L_{age} + \alpha L_{\text{confusion}}$ where $\alpha$ is a hyper-parameter that determines how much gender confusion we want to introduce. Secondary task would be to minimize $L_{gen}$.

**Algorithm 1** Joint Learning and Unlearning (JLU)

```
1: procedure TRAINING
2:     for epochs do
3:         while $L_{gen}$ is not optimized do
4:             $\min(L_{gen})$
5:         $\min(L_{age} + \alpha L_{\text{confusion}})$
```

To evaluate the correctness of the training procedure, we created a training, validation and testing split of OD. An interesting visualization we obtained from the training process was the variation of $L_{gen}$ during training as shown in 7. As we alternate between the primary and the secondary task, the duel between $L_{gen}$ and $L_{\text{confusion}}$ is responsible for the results obtained. At the end of training, when tested on hold out set for age prediction, the accuracy was 0.99 whereas when tested on gender prediction, the accuracy was 0.53 which proves that the training was successful.

## 4. Results

The focus of our experimentation was broadly to analyze whether or not $\theta_{repr}$ learned gender while being trained for the task of image prediction. We visualize $\theta_{repr}$ by passing a small amount of data through the trained network and

| Model | Accuracy |
|---|---|
| M1 | 46% |
| M2 | 49% |

Table 1. Test results on EBD after training M1 and M2 on ED

recording the value $\theta^i_{repr}$ for input example i. We then take all $\theta^i_{repr}$'s obtained project them down to 2-d using t-SNE. We use Tensorflow Embedding Projector with the default parameters to produce the t-SNE embeddings.

## 4.1. Unlearning on extreme bias

The goal of this experiment is to test the hypothesis whether gender unlearning is useful in removing bias from a neural network trained on an extremely biased dataset. So, in this experiment we consider the dataset BD, create a training, and validation split. We then train a model (M-1) to optimize only for age prediction without any gender unlearning. We train another model (M-2) on the same dataset to optimize for age prediction while unlearning gender with $\alpha$ set to 1 where the training is done according to JLU algorithm as described above. We finally test the models on the dataset EBD. The test accuracies we obtained are shown in Table-1. We notice a 3% increase in classification accuracy because of the use of unlearning. This is because without unlearning, the neural network tends to pick on the bias in the dataset where there are only younger females and older males, and in order to optimize for the training set, the network starts giving importance to gender while making an age prediction. In a way, the network believes, that if it is a woman, she must be young and if it is a guy, he must be old. This is expected because that is what we expose the neural network to during training. The improvement in results by unlearning gender is due to the introduction of confusion loss which penalizes the network for making any polarized inference about gender.

We also used the dataset BD to construct t-SNE embeddings for both models M1 and M2. For M1, the results are shown in Figure-8. The red dots correspond the females while the blue dots correspond to males. We can see that the embeddings are clearly separable by gender (the line is drawn to highlight the separation) even though the network was never exposed to gender. As described above, this is because of the bias in the dataset which is picked up the neural network. For M2, the results are shown in Figure-8. We can see that the embeddings are not separable by gender anymore. This is because by unlearning, we have restricted the network to draw inferences about gender and as a result when M2 is used to make age predictions in the wild, it would do so better because it would not base its inferences on gender.

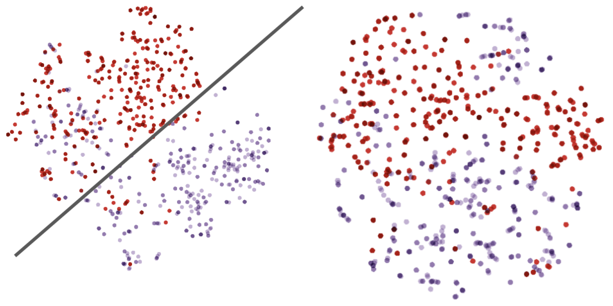
Figure 8. Embeddings are separable by gender in case of M1 (left) but not in case of M2 (right)

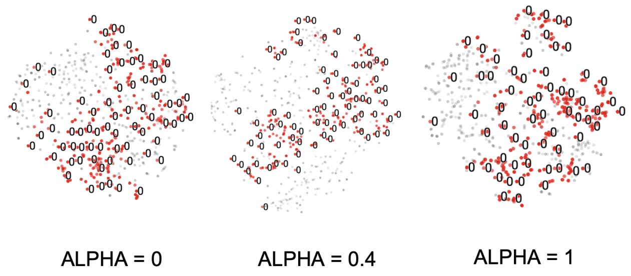

Figure 9. Increasing $\alpha$ helps in mixing the gender embeddings

## 4.2. Effect of $\alpha$ on the embeddings

The goal of this experiment is to test the effect of the hyperparameter $\alpha$ in confusing the network on gender and creating mixed up embeddings not separable by gender. For the purpose of this test, we train another model M3 with JLU training setting the value of $\alpha$ to 0.4. Note that the model M1, since there is no unlearning, represents $\alpha = 0$ setting and M2 as discussed before was trained for $\alpha$ set to 1. We compute t-SNE embeddings for the images in EBD dataset by passing them each of the 3 models separately. The resultant embeddings are shown in Figure-9, where females are marked by 0 and rest are males. These results are in line with the expectations that increasing $\alpha$ will promote gender confusion and thus make the embeddings less separable by gender.

## Individual Contributions

- **Siddharth** Data Preparation, Scraping, Microsoft Face API verification
- **Abhimanyu** Implemented JLU from scratch, ran experiments and worked on producing the t-SNE embeddings.
- **Siddharth and Abhimanyu** Worked on JLU and creating samples of the data.

## References

[1] CricAu. `https://www.cricket.com.au`.

[2] CricBuzz. `https://www.cricbuzz.com`.

[3] M. Alvi, A. Zisserman, and C. Nellaker. Turning a blind eye: Explicit removal of biases and variation from deep neural network embeddings. In *Workshop on Bias Estimation in Face Analytics, ECCV*, 2018.

[4] R. Caruana. Multitask learning. *Machine learning*, 28(1):41–75, 1997.

[5] R. Collobert and J. Weston. A unified architecture for natural language processing: Deep neural networks with multitask learning. In *Proceedings of the 25th international conference on Machine learning*, pages 160–167. ACM, 2008.

[6] L. Deng, G. Hinton, and B. Kingsbury. New types of deep neural network learning for speech recognition and related applications: An overview. In *Acoustics, Speech and Signal Processing (ICASSP), 2013 IEEE International Conference on*, pages 8599–8603. IEEE, 2013.

[7] G. Levi and T. Hassner. Age and gender classification using convolutional neural networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 34–42, 2015.

[8] P. M. R. Martin Koestinger, Paul Wohlhart and H. Bischof. Annotated Facial Landmarks in the Wild: A Large-scale, Real-world Database for Facial Landmark Localization. In *Proc. First IEEE International Workshop on Benchmarking Facial Image Analysis Technologies*, 2011.

[9] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

[10] T. Tommasi, N. Patricia, B. Caputo, and T. Tuytelaars. A deeper look at dataset bias. In *Domain Adaptation in Computer Vision Applications*, pages 37–55. Springer, 2017.

[11] A. Torralba and A. A. Efros. Unbiased look at dataset bias. In *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*, pages 1521–1528. IEEE, 2011.