Explainability and Bias

Dependable AI Assignment-2

Kwanit Gupta, B19EE046[a]

[a]*Indian Institute of Technology, Jodhpur*

March, 2023

Note

I attempted only the 2nd Question, where I picked CVPR 2017's "Aggregated Residual Transformations for Deep Neural Networks" paper for Problem Task and NeurIPS 2016's "Equality of Opportunity in Supervised Learning" paper for post-processing based bias mitigation. The Sections and Subsections will thereby explain about the necessary variations and ablations alongside bias metrics associated alongside traditional metrics.

---

1. Question-2

- Select a recent paper on a state-of-the-art performing model and reproduce the results on any one dataset mentioned in the paper.
- Do you observe any bias? Explain the type of bias you observed. [10 Marks]
- Try to mitigate the bias using the bias mitigation technique. In this, you have to select the paper related to bias mitigation and use it to mitigate the bias you found in part B. Report the metrics values used in the paper. You are advised to select a cognitive bias mitigation paper to mitigate the cognitive bias in the computer vision task you select in part A.
- Try another approach of your own to mitigate the bias using two techniques, either DATA method (Pre-Processing) or ALGORITHMIC method. Report the values of the same metric you used in part C for these techniques also.
- Compare the bias mitigation techniques used in parts C and D(a), and D(b) by taking in support of bias metrics.
- Report the changes you observed before and after applying bias mitigation techniques.

1.1. Part-1 (CVPR 2017's ResNext Model)

The authors propose three types of residual units:

- The Basic Residual Unit has two convolutional layers with the same number of filters
- Bottleneck Residual Unit has three convolutional layers with decreasing numbers of filters.
- The Grouped Residual Unit is similar to the Bottleneck Residual Unit, but it has groups of filters instead of single filters.

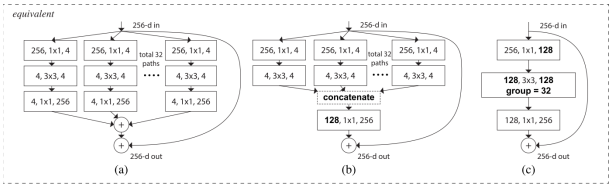

Figure 1: Residual Units in ResNeXt

The authors then introduce the Aggregated Residual Transformation (ART) block, which is composed of multiple residual units. The ART block uses a "collect-and-aggregate" strategy to improve the accuracy and efficiency of the network.

For Evaluation Metrics (to also showcase the nature of bias), I used the following:-

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{1}$$

$$Precision = \frac{TP}{TP + FP} \tag{2}$$

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} \tag{3}$$

$$\text{support}(y) = \sum_{i=1}^{n} 1(y_i = y) \tag{4}$$

$$TPR = \frac{TP}{TP + FN} \tag{5}$$

$$FPR = \frac{FP}{FP + TN} \tag{6}$$

$$PPV = \frac{TP}{TP + FP} \tag{7}$$

$$NPV = \frac{TN}{TN + FN} \tag{8}$$

Where TP is True Positive, FP is False Positive, TN is True Negative and FN is False Negative.
Following are the reproduced results of ResNext architecture with different variations and same parameters as used by authors:-

Metrics	50_32 × 4	101_32 × 8	101_64 × 4
Accuracy	67%	65%	64%
M-TPR	67.19%	64.88%	63.57%
M-FPR	3.64%	3.9%	4.04%
M-PPV	69.05%	66.04%	63.87%
M-NPV	96.35%	96.09%	95.95%
m-TPR	67.2%	64.88%	63.57%
m-FPR	3.64%	3.9%	4.04%
m-PPV	67.2%	64.88%	63.57%
m-NPV	67.2%	64.88%	63.57%

Table 1: ResNeXt Multiclass classification metrics on CIFAR10

Difference in normal metrics & that of Macro and Micro (represented as M and m respectively) is in the representation with respect to Multiclass scenario. Macro metrics focus on weighing each class whereas micro metrics focus on weighing each sample.

1.2. Part-2 Observed Source of Bias

From the paper's point of view, the architecture provides the following reasons for biasing:-

- Grouping: The ResNeXt model groups filters into cardinality groups, which allows for increased model capacity without a significant increase in computational cost. However, if the grouping is not balanced across different classes, it can lead to bias in the model.
- Cardinality: The cardinality parameter determines the number of groups into which the filters are divided. If the cardinality is too low, the model may not have enough capacity to learn complex patterns in the data, whereas if it is too high, it may lead to overfitting and bias.
- Feature Correlation: In the ResNeXt model, feature maps are aggregated by concatenating them along the channel dimension, followed by a bottleneck layer. This aggregation can cause the model to learn correlated feature representations, leading to biased predictions. In CIFAR-10, some of the classes have similar visual features, such as a bird and an airplane or a truck and an automobile. This similarity in features can lead to biased predictions if the model learns to correlate the features.
- Lack of Diversity: In some ResNeXt models, the authors observed that the diversity of filters learned in the initial layers was limited. This lack of diversity can result in the model having a biased view of the input data, leading to biased predictions. In CIFAR-10, the dataset has several classes, each with different visual features. The lack of diversity in the initial filters can result in biased predictions, especially for the classes that are not well-represented in the initial layers.

Hereby, I observed the following biases too:-

- Texture Bias: If the images in the dataset are highly textured, the model might tend to learn texture-based features more than other features, leading to poor generalization.
- Contrast Bias: If the contrast between the foreground and background in the images is high, the model might tend to focus on the contrast rather than other features, leading to poor generalization.
- Color Bias: If the images in the dataset have a particular color bias, the model might tend to learn features based on that color, leading to poor generalization.
- Occlusion Bias: If the images in the dataset have occlusions or clutter, the model might tend to learn features based on the occlusion or clutter, leading to poor generalization.
- Illumination Bias: If the images in the dataset have a particular illumination, the model might tend to learn features based on that illumination, leading to poor generalization.

1.3. Part-3 NeurIPS 2016 Paper

The paper begins with a formalization of the fairness constraint being considered. The authors consider a binary classification setting where each data point (x,y) consists of a feature vector x and a binary label y (e.g. a positive or negative label).

They also assume the existence of a protected attribute A that determines group membership. For example, A could be a person's gender or race, and the protected attribute is referred to as "privileged" if it corresponds to the historically advantaged group.

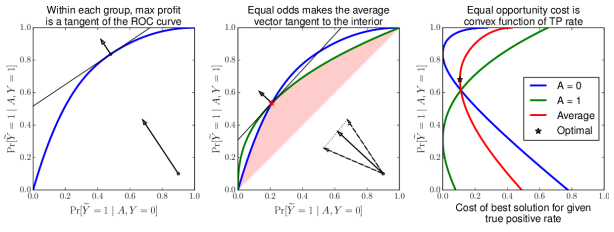

Figure 2: Equality of Opportunity in Supervised Learning

The authors formalize the notion of "equality of opportunity" in the following way: a classifier satisfies the equality of opportunity constraint with respect to a protected attribute A if the true positive rate (TPR) for group A is equal to the TPR for the complementary group (i.e., the non-privileged group), when conditioning on the outcome y.

More formally, the authors define two groups: the privileged group (A=1) and the non-privileged group (A=0). They define TPR as the probability of correctly predicting a positive outcome for each group:

$$TPR(A = 1) = P(y = 1|A = 1, h(x) = 1) \quad (9)$$

$$TPR(A = 0) = P(y = 1|A = 0, h(x) = 1) \quad (10)$$

where h(x) is the output of the classifier on input x. The equality of opportunity constraint requires that TPR(A=1) = TPR(A=0), when conditioned on the outcome y. That is,

$$P(y = 1|A = 1, h(x) = 1) = P(y = 1|A = 0, h(x) = 1) \quad (11)$$

The authors propose two algorithms for achieving fairness under the equality of opportunity constraint: (1) a post-processing algorithm, and (2) an algorithm that incorporates fairness into the training process.

The post-processing algorithm takes as input a pre-trained classifier h and an additional threshold parameter, and outputs a new classifier h' that satisfies the equality of opportunity constraint. The algorithm works by adjusting the threshold for the privileged group and the non-privileged group separately, such that the TPR is equal for both groups.

The algorithm can be described as follows:

- Compute TPR(A=1) and TPR(A=0) for the original classifier h and the current threshold value.
- If TPR(A=1) > TPR(A=0), decrease the threshold for the privileged group (A=1) until the TPRs are equal.
- If TPR(A=0) > TPR(A=1), increase the threshold for the non-privileged group (A=0) until the TPRs are equal.
- Output the resulting classifier h'.

The second algorithm incorporates fairness into the training process by adding a regularization term to the loss function that penalizes deviations from the equality of opportunity constraint. Specifically, the authors propose a regularizer based on the difference between the TPRs for the privileged and non-privileged groups, defined as:

$$R(h) = |TPR(A = 1) - TPR(A = 0)| \quad (12)$$

Since CIFAR-10 doesn't have any protected attributes, I personally set "pet animals" as a protected attribute with 4 animal classes and the other 2 animal classes as non-pet animals.

Metrics	50_32 × 4	101_32 × 8	101_64 × 4
Accuracy	61%	58%	55%
M-TPR	NaN%	NaN%	NaN%
M-FPR	4.7%	4.9%	5.34%
M-PPV	29.25%	28.26%	27.48%
M-NPV	95.25%	95.03%	94.65%
m-TPR	60.9%	58.45%	54.95%
m-FPR	8.39%	8.05%	8.42%
m-PPV	60.9%	58.45%	54.95%
m-NPV	60.9%	58.45%	54.95%

Table 2: ResNeXt's Protected Multiclass metrics on CIFAR10 before Equality Odds

1.4. Part-4 My Devised Bias Mitigation Strategies

Figure 3: Random Sharpness Enhancement

Metrics	50_32 × 4	101_32 × 8	101_64 × 4
M-TPR	NaN%	NaN%	NaN%
M-FPR	3.87%	4.96%	5.34%
M-PPV	31.72%	28.26%	27.48%
M-NPV	96.12%	95.03%	94.65%
m-TPR	67.04%	58.45%	54.95%
m-FPR	5.75%	8.05%	8.42%
m-PPV	67.04%	58.45%	54.95%
m-NPV	67.04%	58.45%	54.95%

Table 3: ResNeXt's Protected Multiclass metrics on CIFAR10 after Equality Odds

Metrics	50_32 × 4	101_32 × 8	101_64 × 4
Accuracy	61%	66%	60%
M-TPR	NaN%	NaN%	NaN%
M-FPR	4.32%	3.9%	4.65%
M-PPV	18.64%	18.64%	18.16%
M-NPV	95.67%	96.1%	95.34%
m-TPR	61.05%	65.75%	59.8%
m-FPR	4.3%	4.75%	6.35%
m-PPV	61.05%	65.75%	59.8%
m-NPV	61.05%	65.75%	59.8%

Table 4: ResNeXt's Non-Protected Multiclass metrics on CIFAR10 before Equality Odds

The ideations that I implemented with the ResNeXt architecture, were primarily based on mitigating the Color, Texture and Illumination Bias via Random AutoContrast and Random Sharpness enhancements into images with the probability of 0.5, i.e either yes or no.

Figure 4: Random Contrast Enhancement

Metrics	50_32 × 4	101_32 × 8	101_64 × 4
M-TPR	NaN%	NaN%	NaN%
M-FPR	0%	0%	0%
M-PPV	NaN%	NaN%	NaN%
M-NPV	100%	100%	100%
m-TPR	100%	100%	100%
m-FPR	0%	0%	0%
m-PPV	100%	100%	100%
m-NPV	100%	100%	100%

Table 5: ResNeXt's Non-Protected Multiclass metrics on CIFAR10 after Equality Odds

Following were the results of those variations:-

Metrics	50_32 × 4	101_32 × 8	101_64 × 4
Accuracy	69%	64%	66%
M-TPR	68.73%	63.82%	65.74%
M-FPR	3.47%	4.01%	3.8%
M-PPV	69.34%	64.74%	66.85%
M-NPV	96.52%	95.98%	96.19%
m-TPR	68.73%	63.83%	65.74%
m-FPR	3.47%	4.01%	3.8%
m-PPV	68.73%	63.83%	65.74%
m-NPV	68.73%	63.83%	65.74%

Table 6: ResNeXt's Multiclass metrics on CIFAR10 with Random Sharpness

Metrics	50_32 × 4	101_32 × 8	101_64 × 4
Accuracy	68%	66%	64%
M-TPR	68.05%	65.5%	64.38%
M-FPR	3.55%	3.83%	3.95%
M-PPV	68.73%	66.77%	64.72%
M-NPV	96.44%	96.16%	96.04%
m-TPR	68.05%	65.5%	64.39%
m-FPR	3.55%	3.83%	3.95%
m-PPV	68.05%	65.5%	64.39%
m-NPV	68.05%	65.5%	64.39%

Table 7: ResNeXt's Multiclass metrics on CIFAR10 with Random AutoContrast

Metrics	50_32 × 4	101_32 × 8	101_64 × 4
Accuracy	67%	66%	66%
M-TPR	67.43%	66.38%	65.65%
M-FPR	3.61%	3.73%	3.81%
M-PPV	67.95%	67.17%	66.78%
M-NPV	96.38%	96.26%	96.18%
m-TPR	67.43%	66.38%	65.65%
m-FPR	3.61%	3.73%	3.1%
m-PPV	67.43%	66.38%	65.65%
m-NPV	67.43%	66.38%	65.65%

Table 8: ResNeXt's Multiclass metrics on CIFAR10 with both Random Sharpness and AutoContrast

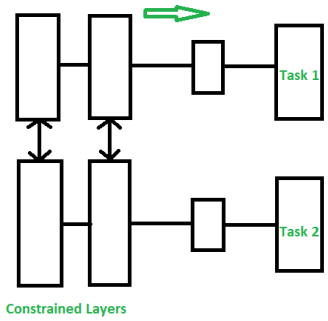

Figure 5: MultiTask Learning

For Architecture related bias mitigation, I mounted a reconstruction MLP at the penultimate embedding feature layer, just after pooling was done globally via minimizing MSE. It was done so that the embeddings space retains the original discriminative characteristics because of which they can become different to each other.

Following were the results of those variations (only tried on 50_32 $\times$ 4d):-

Metrics	$w_{ce} = w_{mse}$	$w_{ce} > w_{mse}$	$w_{ce} < w_{mse}$
Accuracy	67%	68%	70%
M-TPR	66.76%	68.36%	70.24%
M-FPR	3.69%	3.51%	3.3%
M-PPV	66.77%	68.89%	70.29%
M-NPV	96.3%	96.48%	96.69%
m-TPR	66.76%	68.36%	70.25%
m-FPR	3.69%	3.51%	3.3%
m-PPV	66.76%	68.36%	70.25%
m-NPV	66.76%	68.36%	70.25%

Table 9: ResNeXt's Multiclass metrics on CIFAR10 with MLP-based reconstruction head

1.5. Comparitive Analysis

"Equal Opportuniy" primarily focussed on the protected attribute that was accounted in CIFAR-10 as subset, but my main motivation was to facilitate the bias mitigation process throughout the CIFAR-10 dataset, which was greatly done by Multi-Task CNN architecture.

Although I cannot comment on the comparison between my devised techniques and "Equal Opportunity", but I provided the corresponding tables for protected v/s non-protected samples. For the scenario of Data Preporcessing v/s Model-related, the majority performance boost and bias mitigation was observed in the later part despite the former actual providing the necessary real life scenarios of texture, color and illumination mixtures. This was primarily reasonable since the embedding layer is the discriminative and refined representation of the feature space formed by CIFAR-10. So, tweeking that with the reconstruction loss was a great call to retain the original class-wise characteristics.