Dependable AI Assignment 2 Report

Mitigating Biases in EfficientNet-v2 using Meta Orthogonalization

- Soumik Roy, B20AI042

Overview

About the Research Papers

For this assignment I have referred to 2 research papers namely “EfficientNetV2: Smaller Models and Faster Training” and “Debiasing Convolutional Neural Networks via Meta Orthogonalization”. The first paper talks about an efficient CNN architecture in terms of speed and size, giving State of the Art accuracies in computer vision tasks. The second research paper talks about mitigation of Bias in CNN based architectures using a technique called Meta-Orthogonalization. I have combined the implementations of both the research papers in order to mitigate Biases in EfficientNet-v2 using Meta Orthogonalization.

EfficientNetV2: Smaller Models and Faster Training 🔗
Debiasing Convolutional Neural Networks via Meta Orthogonalization 🔗

Part A

The EfficientNetV2 model

For this part, I chose the research paper EfficientNetV2: Smaller Models and Faster Training by authors Mingxing Tan and Quoc V. Le. It talks about the new CNN based architecture they come up with, which gives accuracies similar to Resnet, EfficientNet-B0 and other SOTA architectures , with faster training speeds and smaller size.

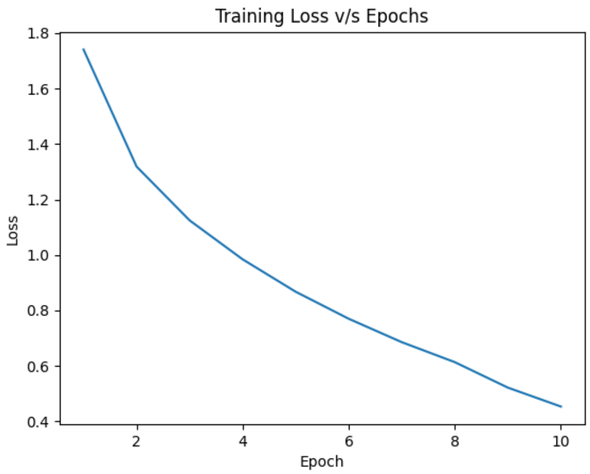
To develop this family of models, the authors use a combination of training-aware neural architecture search and scaling, to jointly optimize training speed and parameter efficiency. The models were searched from the search space enriched with new ops such as Fused-MBConv. Their experiments show that EfficientNetV2 models train much faster than state-of-the-art models while being up to 6.8x smaller. The training can be further sped up by progressively increasing the image size during training, but it often causes a drop in accuracy.

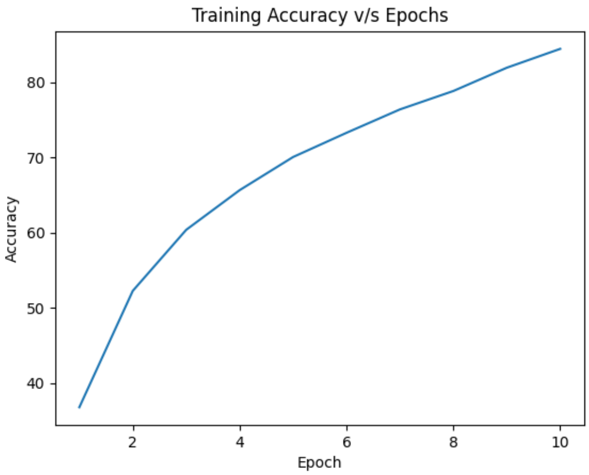
Dataset Used

The authors tested on several different datasets including CIFAR10, CIFAR100, ImageNet etc. I trained the model on the CIFAR-100 dataset to reproduce the results demonstrated in the research paper.

Results

I trained the model for 10-epochs, and the training curves are as follows :

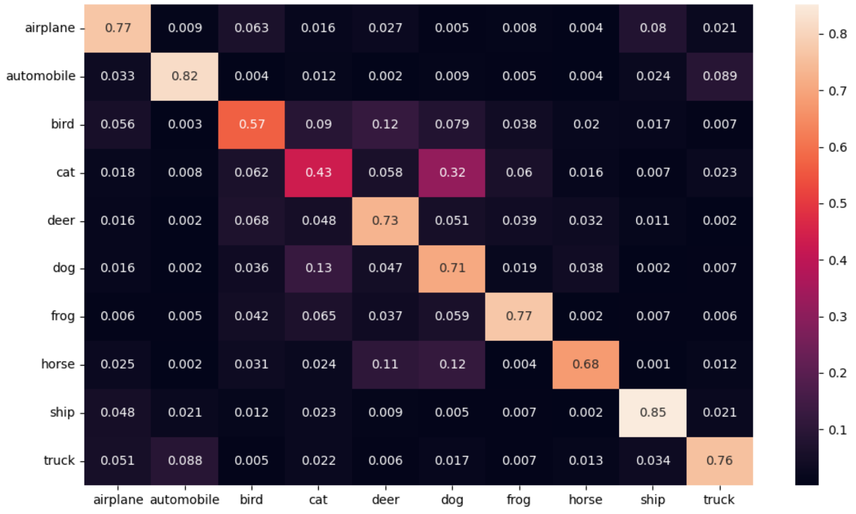




Part B

Observing Bias in the previous Model

After training the EfficientNet-v2 in the previous part, I observed the predictions it was making. The confusion matrix for the model prediction, for the 10 most mispredicted classes is as follows :



	airplane	automobile	bird	cat	deer	dog	frog	horse	ship	truck
airplane	0.77	0.009	0.063	0.016	0.027	0.005	0.008	0.004	0.08	0.021
automobile	0.033	0.82	0.004	0.012	0.002	0.009	0.005	0.004	0.024	0.089
bird	0.056	0.003	0.57	0.09	0.12	0.079	0.038	0.02	0.017	0.007
cat	0.018	0.008	0.062	0.43	0.058	0.32	0.06	0.016	0.007	0.023
deer	0.016	0.002	0.068	0.048	0.73	0.051	0.039	0.032	0.011	0.002
dog	0.016	0.002	0.036	0.13	0.047	0.71	0.019	0.038	0.002	0.007
frog	0.006	0.005	0.042	0.065	0.037	0.059	0.77	0.002	0.007	0.006
horse	0.025	0.002	0.031	0.024	0.11	0.12	0.004	0.68	0.001	0.012
ship	0.048	0.021	0.012	0.023	0.009	0.005	0.007	0.002	0.85	0.021
truck	0.051	0.088	0.005	0.022	0.006	0.017	0.007	0.013	0.034	0.76

As we can see, the confusion matrix shows that the model is predicting bird and cat wrongly most frequently. Further it is **biased in predicting cats as dogs, dogs as cat, and birds as deer very oftenly.**

This is a type of Sample bias, as the model had probably been trained on the training data such that above classes were not highly distinguishable from each other, or were quite specific. Hence while predicting on testing data, whenever images of the above classes, which looked similar were encountered, the model made mistakes.

Part C

Mitigating Biases using Meta-Orthogonalization

For this part I chose the research paper, "Debiasing Convolutional Neural Networks via Meta Orthogonalization" by Kurtis Evan David, Qiang Liu and Ruth Fong. While deep learning models often achieve strong task performance, their successes are hampered by their inability to disentangle spurious correlations from causative factors, such as when they use protected attributes (e.g., race, gender, etc.) to make decisions. In the paper, the authors tackle the problem of debiasing convolutional neural networks (CNNs) in such instances. Building off of existing work on debiasing word embeddings and model interpretability, their Meta Orthogonalization method encourages the CNN representations of different concepts (e.g., gender and class labels) to be orthogonal to one another in activation space while maintaining strong downstream task performance. Through a variety of experiments, the authors systematically test their method and demonstrate that it significantly mitigates model bias and is competitive against current adversarial debiasing methods.

Meta-Orthogonalization

The Debiasing method can be decomposed into three separate losses used during training:

1. Classification Loss ($\mathcal{L}_{\text{class}}$) – original task loss to train the CNN, e.g. cross entropy.
2. Concept Loss ($\mathcal{L}_{\text{concept}}$) – log-loss to learn image concept vectors at a specific layer.
3. Debias Loss ($\mathcal{L}_{\text{debias}}$) – our regularization term to induce orthogonal concepts

The value of Debias Loss is as below :

$$\mathcal{L}_{\text{debias}}(\beta') = \sum_{c} \left(\frac{\beta_c'^{\top} \nu}{||\beta_c'||_2 ||\nu||_2} \right)^2$$

We simultaneously learn our concept embeddings βc, rather than at the end of training, because βc is directly used in the proposed regularization $\mathcal{L}_{\text{debias}}$. Assuming βc is learned using SGD, at every iteration, βc is updated to

$$\beta_c' = \beta_c - \alpha \nabla_{\beta_c} \mathcal{L}_{\text{concept}}(c, \theta)$$

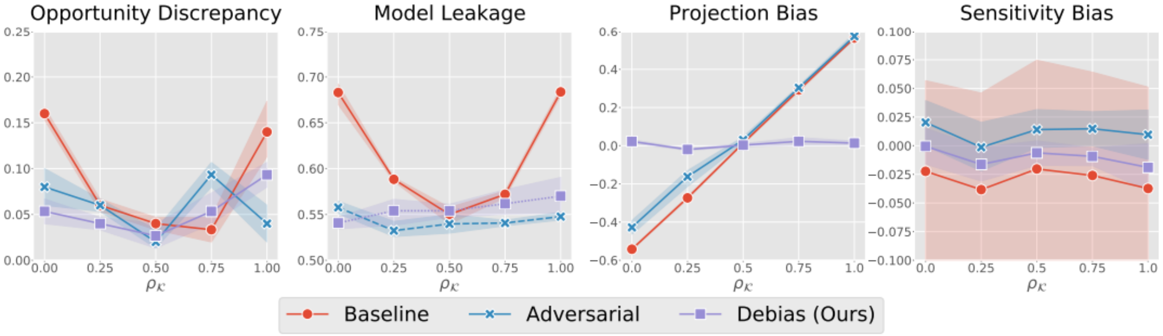
The log loss computed by is itself a function of θ, and thus β'c is also a function of θ which can now be used to regularize the CNN:

Finally the minimisation function looks like :

$$\min_{\theta,\beta} \quad \mathcal{L}_{\text{class}}(\theta) + \sum_{c} \mathcal{L}_{\text{concept}}(c, \beta) + \gamma \mathcal{L}_{\text{debias}}(\beta')$$

Results

Using the new loss function, we can debias our model and train it as proposed. The results of the debiased model using 4 different metrics namely : Opportunity Discrepancy, Model Leakage, Projection Bias and Sensitivity bias are as below. We use the class "Cat" for checking the biasing metrics. This is because the "Cat" class was the most mispredicted class and hence the bias was most seen in this class. We also run a training loop using Adversarial Debiasing instead of Metha-Orthogonalization. So we will compare Baseline model, Adversarially debiased model, and our Metha-Orthogonalization model with each other using the above metrics.

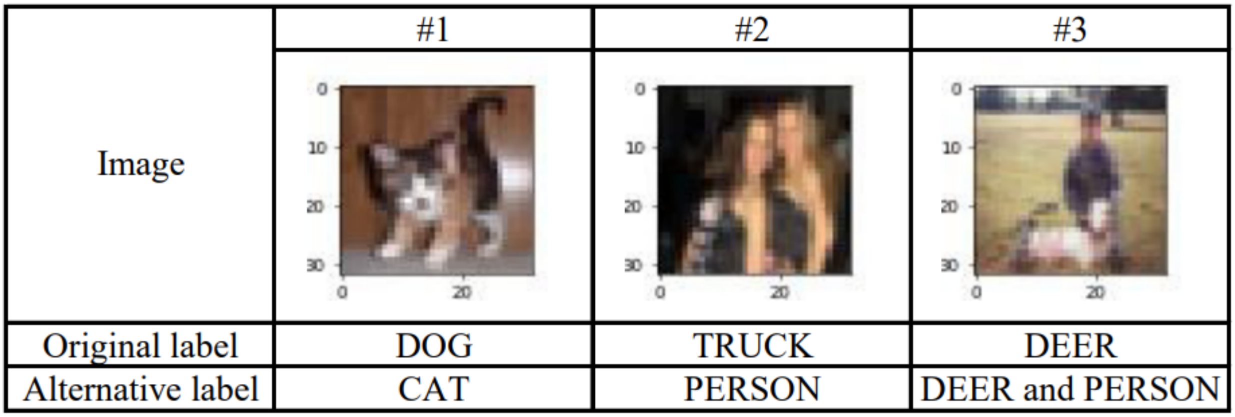


We see our model consistently does better than the standard training (Baseline), and is comparable to adversarial debiasing. Optimal Projection and Sensitivity Bias (right two curves) should be at the y = 0 line, which is closest to our model.

Part D

Mitigating Biases using Data Preprocessing

According to the study done in paper Autocleansing : unbiased estimation of deep learning with mislabeled data , alot of images (13%) in the CIFAR-100 dataset have been mislabelled. Some examples are shown below :



	#1	#2	#3
Image			
Original label	DOG	TRUCK	DEER
Alternative label	CAT	PERSON	DEER and PERSON

0
10
20
30
0
20
0
10
20
30
0
20
0
10
20
30
0
20

So we can re label these images to their original names, and train our model again .This would help in mitigating some of the bias.

Mitigating Biases Algorithmically

In the work done by the authors, they have already used another Debiasing technique - the Adversarial Debiasing, which works good in debiasing, however not as good as the Meta-Orthogonalization technique. In this method, the classifier model is learned to maximize prediction accuracy and simultaneously reduce an adversary's ability to determine the protected attribute from the predictions. This approach leads to a fair classifier as the predictions cannot carry any group discrimination information that the adversary can exploit.

We are mitigating the bias stochastically in the proposed algorithm. We can however also add a term of momentum in the Debias loss function which takes in consideration the Debias Loss of the previous step. THis can be helpful as it will take in consideration that the model is debiasing step by step and hence the model in the previous step was less biased than the raw model.

Part E

Comparing all techniques

If we compare all the techniques, then we will observe that the Meta orthogonalization technique works better than the data preprocessing debiasing technique. This is so because in the data preprocessing method we follow, we can only relabel the wrongly labeled data, which is quite small in comparison to the size of our CIFAR-100 dataset, hence it leads to only little bit debiasing, and that too only for some classes for which more images were incorrectly labeled.

Also the Meta-Orthogonalization technique works a little better than Adversarial Debiasing, as we saw in the plot added in the Part C. However if we apply the momentum part to our meta orthogonalization loss function, then debiasing would improve even further, and it would become more efficient in debiasing.