Mitigate Pruning Bias In ResNet18 Through CV Techniques and Strategies

Jonathan Peral Gort[jperalgo@sfu.ca], Daniel Vuksan[danielv@sfu.ca]

Abstract.

We investigate bias in compressed neural networks by replicating and extending Hooker's *Characterising Bias in Compressed Models* (Hooker et al., 2020). Using a ResNet18 baseline and a 90%-pruned variant, we perform binary classification on the CelebA task to determine who is blonde or otherwise. Additionally, we identify Compression Identified Exemplars (CIE) as predominantly low-frequency samples - and demonstrate it based on results shown by our Exploratory Data Analysis - and perform performance analysis on them. We then mitigate bias by augmenting the pruned ResNet18 with a ViT encoder block, improving CIE Test Set scores by an observable amount (0.46 $\rightarrow$ 0.51 F1-score). Our work highlights low-frequency performance analysis on compressed models, and shines some light on possible ways to mitigate the bias introduced post-compression of models.

1 Problem Statement and Motivation

Motivation In the current times of Machine Learning, innovation and AI, we have observed how this fascinating industry has transformed the modern World. We have gotten many things: from self-driving cars that use Computer Vision models (Kirillov et al., 2023) to vast amounts of information easily accessible through prompt chats (DeepSeek-AI et al., 2025; Vaswani et al., 2023; Brown et al., 2020). The World of Machine Learning and AI is expanding. There is no doubt about that. However, we have observed that in terms of scalability, we still have not managed to make compact models with state-of-the-art performance. We can all look back at when smartphones were initially introduced, and realize the massive impact having portable, relatively compute-strong, pocket-sized devices had on all of us. We like to think that, in a similar trend, once AI figures out the scalability problem, it will open up a whole new set of opportunities. This is precisely why we have chosen to replicate the Characterizing Bias in Compressed Models paper (Hooker et al., 2020), and additionally carry out a set of experiments that expand on the paper.

Problem Statement Our task is a binary classification one. We are looking to determine whether individuals from the CelebA dataset (Liu et al., 2015) are either blonde or non-blonde. We will achieve this by building a ResNet18 architecture, trained on the full CelebA dataset. This ResNet18 model will act as our baseline. Furthermore, we will also train a pruned version of the ResNet18 baseline model, which will have a similar training protocol, save for the difference that it will be pruned to a sparsity of 0.90 - i.e. only 10% of its original set of weights will be left. Subsequently, once we have our baseline non-pruned and pruned models we will determine Compression Identified Exemplars (CIE) (Hooker et al., 2020). Once our CIE are determined, we will show evidence of how CIE are mostly composed of low-frequency samples in the dataset. We will also look at how compressed models perform relative to our non-pruned baseline ResNet18. And finally, we will perform some experiments to mitigate bias on the CIE performance.

Review Of Relevant Topics

Attention The Attention Layer is an essential component of the ViT encoder architecture. Their significance stems from their ability to focus on specific parts of the inputs sequence when making predictions. The way it works is through computing a weight sum of input features, where the weights are determined by the relevance of each feature to the current task. Such relevance is determined by using the dot product between a query (Q) and keys (K), the most common form being Scaled-Dot-Product Attention:

$$\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V$$

where V is the value matrix, and d_k the dimension of the keys.

Modal CIE As proposed by the by the Characterising Bias In Copmpressed Models paper (Hooker et al., 2020), Modular CIE is the simplest form to determine CIE. In the context, of our binary classification task, they are simply determined when the inference results of our baseline ResNet18 do not agree with those from its compressed variants.

$$\text{CIE}(x, t) = \begin{cases} 1, & \text{if } y^M(x, 0) \neq y^M(x, t) \\ 0, & \text{otherwise} \end{cases}$$

Where *x* is a sample in our dataset, *y* is the model's inference result, *t* the level of compression (0.90 in our case), and *M* is the class predicted most frequently by both models - i.e. blonde, since it is a binary classification problem.

2 Related Works

As previously mentioned, this paper is heavily influenced by Hooker's *Characterising Bias in Compressed Models* (Hooker et al., 2020), and thus to fully understand it, we recommend giving that paper a read.

3 BaseLine Results

3.1 Non-Pruned ResNet18

Training Procedure

According to Hooker's paper (Hooker et al., 2020), we train in approximately 10,000 steps. With our hyperparameters, we managed to come close at 10,700 steps.

- Our training split is 90% on training set, and 10% on test set.
- K-FOLDS = 4
- EPOCHS = 5
- BATCH-SIZE = 256

Pruning

No pruning is applied on this model.

Results

Loss Train across 4-Folds *Refer to Fig. 1*

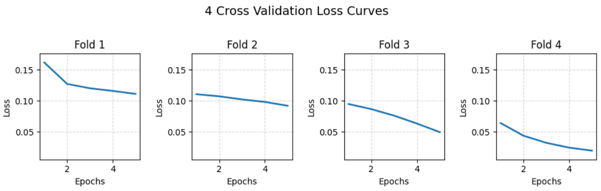

Fig. 1: Loss per Epochs (in each fold)

Test Set Performance

- Accuracy: 0.95
- Precision: 0.85
- Recall: 0.80
- F1-Score: 0.83
- Number of CIE: -

3.2 Pruned ResNet18 (sparsity=0.9)

Training Procedure

Similarly trained as it's non-pruned counterpart.

- Our training split is 90% on training set, and 10% on test set.
- K-FOLDS = 4
- EPOCHS = 5
- BATCH-SIZE = 256

Pruning

Pruning is applied at a rate of 0.1087, every 500 steps; starting after 1000 steps. This ensures us a sparsity of 0.90.

Results

Loss Train across 4-Folds *Refer to Fig. 2*

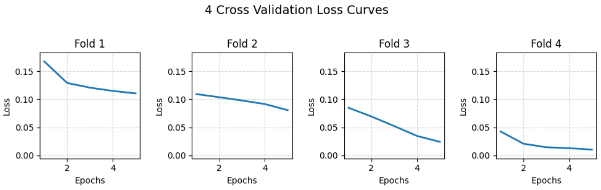

Fig. 2: Loss per Epochs (in each fold)

Test Set Performance

- Accuracy: 0.95
- Precision: 0.83
- Recall: 0.80
- F1-Score: 0.81
- Number of CIE: 760

CIE Test Set Performance

- Accuracy: 0.44
- Precision: 0.45
- Recall: 0.48
- F1-Score: 0.46

4 Experimental Setup

4.1 Data And Exploratory Data Analysis (EDA)

We conducted EDA on the CelebA dataset (Liu et al., 2015). The dataset contains 202,599 total number of samples. In this paper, we specifically chose to focus on three key attributes: *is_blonde*, *is_young*, *gender*. With these three attributes in mind, we showed the distribution of the full dataset, and CIE. These subgroup evaluations allowed us to investigate how compression affects performance disparities across different demographic segments. *Refer to Figure 3 for Unitary Distribution; Figure 4 for 2-Class Intersectional Distribution; Figure 5 for Blonde and Non-Blonde Intersectional Distribution.*

4.2 Metrics

Inference Performance The metrics we will be using to compare the performance of our models is accuracy, precision, recall and f1-score.

- **Accuracy**:

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

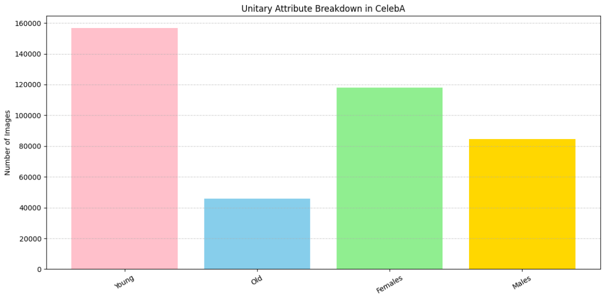

Fig. 3: Unitary Attribute Distribution in CelebA

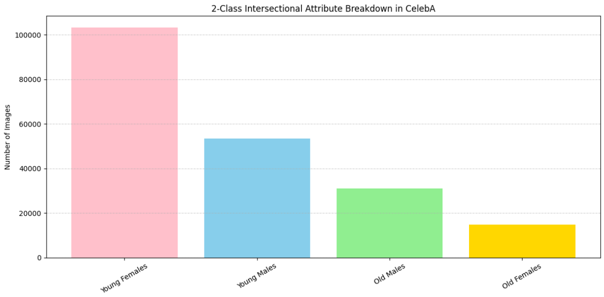

Fig. 4: 2-Class Intersectional Attribute Breakdown in CelebA

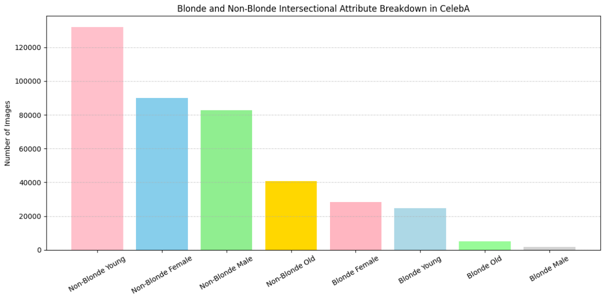

Fig. 5: Blonde and Non-Blonde Intersectional Attribute Breakdown in CelebA

- **Precision**:

$$\text{Precision} = \frac{TP}{TP + FP}$$

- **Recall** (Sensitivity or True Positive Rate):

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

- **F1-score**:

$$\text{F1} = 2 \cdot \frac{\text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}}$$

Where *TP*=True Positives, *TN*=True Negatives, *FP*=False Positives, *FN*=False Negatives.

4.3 Experiment Structure

Our experiment methodology goes as follows: We first define the experiment we wish to work on, found on the Experiments List section; we proceed to justify our reasoning with a hypothesis. We then make the required architectural changes to our base pruned ResNet18, effectively modifying it, train it on our sample dataset of the dataset and compare the results to those yielded by our pruned ResNet18 on the sample dataset. **Note:** the subset size is kept fixed throughout the experimentation. We use 4-fold Cross-Validation to reduce the chance of ”getting lucky” with the subset we chose. If an experiment shows promising results on the sample dataset, we perform it on the full dataset training on a single A100 GPU.

4.4 Experiments List

The following is a tentative list of the experiments we wish to carry out.

- Adding ViT Encoder architecture
- Minority Class Oversampling

5 Experiments

5.1 Experiment 1: [Adding ViT Encoder Block to the top of Pruned ResNet18]

Experiment Procedure Explanation

As per the title, in this experiment we make hybrid between our baseline pruned ResNet18 and a ViT Encoder Block. In order to fine-tune this model, we freeze all the weights on the pruned ResNet18 bottom part of the architecture. Thus, performing weight optimization only on the Encoder.

Justification and Hypothesis

Our hypothesis was that with fine-tuning, our model will mitigate some of the bias of the pruned model. The reason as to why we chose to integrate a ViT Encoder Block on top of the baseline architecture - i.e. late in inference - was due to its feasibility, it is much easier to implement, and computationally lighter. Additionally, we suspect that a late Vit Encoder is capable of detecting high-level relationships.

Architecture Change

We introduce these changes to the baseline architecture:

- Added ViT Encoder Block on top of pruned ResNet18

Results

Loss Train across 4-Folds *Refer to Fig. 6*

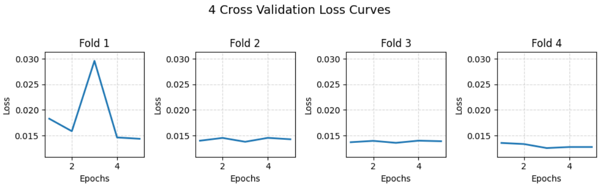

Fig. 6: Loss per Epochs (in each fold)

Test Set Performance

- Accuracy: 0.95
- Precision: 0.82
- Recall: 0.82
- F1-Score: 0.82
- Number of CIE: 793 *(higher does not necessarily mean worse)*

CIE Test Set Performance

- Accuracy: 0.45
- Precision: 0.45
- Recall: 0.58
- F1-Score: 0.51

Conclusion Comparing to the baseline pruned ResNet18, we can observe a overall increase in performance in the CIE Test Set. We suspect if we scale up our dataset, this will continue to improve.

6 Future Work

Although, we have run out of time, we really we wish we could finish all of the experiments proposed in the experiments list. Notably, fine-tuning on minority class CIE in order to determine whether that would reduce bias on the CIE test set. We also want to highlight that we see promising results on expanding on the experiment performed by using Multi-head Latent Attention mechanism (DeepSeek-AI et al., 2024) as they are faster, and perform better than the Self-Attention mechanism we used on our Encoder. Additionally, it would be promising seeing the results of having an earlier-in-inference ViT Encoder, rather than a late one in our architecture, as it could potentially improve on low-level detail capture, rather than high-level semantic details.

References

Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. Language models are few-shot learners. *Preprint*, arXiv:2005.14165.

DeepSeek-AI, Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Ruoyu Zhang, Runxin Xu, Qihao Zhu, Shirong Ma, Peiyi Wang, Xiao Bi, Xiaokang Zhang, Xingkai Yu, Yu Wu, Z. F. Wu, Zhibin Gou, Zhihong Shao, Zhuoshu Li, Ziyi Gao, Aixin Liu, Bing Xue, Bingxuan Wang, Bochao Wu, Bei Feng, Chengda Lu, Chenggang Zhao, Chengqi Deng, Chenyu Zhang, Chong Ruan, Damai Dai, Deli Chen, Dongjie Ji, Erhang Li, Fangyun Lin, Fucong Dai, Fuli Luo, Guangbo Hao, Guanting Chen, Guowei Li, H. Zhang, Han Bao, Hanwei Xu, Haocheng Wang, Honghui Ding, Huajian Xin, Huazuo Gao, Hui Qu, Hui Li, Jianzhong Guo, Jiashi Li, Jiawei Wang, Jingchang Chen, Jingyang Yuan, Junjie Qiu, Junlong Li, J. L. Cai, Jiaqi Ni, Jian Liang, Jin Chen, Kai Dong, Kai Hu, Kaige Gao, Kang Guan, Kexin Huang, Kuai Yu, Lean Wang, Lecong Zhang, Liang Zhao, Litong Wang, Liyue Zhang, Lei Xu, Leyi Xia, Mingchuan Zhang, Minghua Zhang, Minghui Tang, Meng Li, Miaojun Wang, Mingming Li, Ning Tian, Panpan Huang, Peng Zhang, Qiancheng Wang, Qinyu Chen, Qiushi Du, Ruiqi Ge, Ruisong Zhang, Ruizhe Pan, Runji Wang, R. J. Chen, R. L. Jin, Ruyi Chen, Shanghao Lu, Shangyan Zhou, Shanhuang Chen, Shengfeng Ye, Shiyu Wang, Shuiping Yu, Shunfeng Zhou, Shuting Pan, S. S. Li, Shuang Zhou, Shaoqing Wu, Shengfeng Ye, Tao Yun, Tian Pei, Tianyu Sun, T. Wang, Wangding Zeng, Wanjia Zhao, Wen Liu, Wenfeng Liang, Wenjun Gao, Wenqin Yu, Wentao Zhang, W. L. Xiao, Wei An, Xiaodong Liu, Xiaohan Wang, Xiaokang Chen, Xiaotao Nie, Xin Cheng, Xin Liu, Xin Xie, Xingchao Liu, Xinyu Yang, Xinyuan Li, Xuecheng Su, Xuheng Lin, X. Q. Li, Xiangyue Jin, Xiaojin Shen, Xiaosha Chen, Xiaowen Sun, Xiaoxiang Wang, Xinnan Song, Xinyi Zhou, Xianzu Wang, Xinxia Shan, Y. K. Li, Y. Q. Wang, Y. X. Wei, Yang Zhang, Yanhong Xu, Yao Li, Yao Zhao, Yaofeng Sun, Yaohui Wang, Yi Yu, Yichao Zhang, Yifan Shi, Yiliang Xiong, Ying He, Yishi Piao, Yisong Wang, Yixuan Tan, Yiyang Ma, Yiyuan Liu, Yongqiang Guo, Yuan Ou, Yuduan Wang, Yue Gong, Yuheng Zou, Yujia He, Yunfan Xiong, Yuxiang Luo, Yuxiang You, Yuxuan Liu, Yuyang Zhou, Y. X. Zhu, Yanhong Xu, Yanping Huang, Yaohui Li, Yi Zheng, Yuchen Zhu, Yunxian Ma, Ying Tang, Yukun Zha, Yuting Yan, Z. Z. Ren, Zehui Ren, Zhangli Sha, Zhe Fu, Zhean Xu, Zhenda Xie, Zhengyan Zhang, Zhewen Hao, Zhicheng Ma, Zhigang Yan, Zhiyu Wu, Zihui Gu, Zijia Zhu, Zijun Liu, Zilin Li, Ziwei Xie, Ziyang Song, Zizheng Pan, Zhen Huang, Zhipeng Xu, Zhongyu Zhang, and Zhen Zhang. 2025. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *Preprint*, arXiv:2501.12948.

DeepSeek-AI, Aixin Liu, Bei Feng, Bin Wang, Bingxuan Wang, Bo Liu, Chenggang Zhao, Chengqi Dengr, Chong Ruan, Damai Dai, Daya Guo, Dejian Yang, Deli Chen, Dongjie Ji, Erhang Li, Fangyun Lin, Fuli Luo, Guangbo Hao, Guanting Chen, Guowei Li, H. Zhang, Hanwei Xu, Hao Yang, Haowei Zhang, Honghui Ding, Huajian Xin, Huazuo Gao, Hui Li, Hui Qu, J. L. Cai, Jian Liang, Jianzhong Guo, Jiaqi Ni, Jiashi Li, Jin Chen, Jingyang Yuan, Junjie Qiu, Junxiao Song, Kai Dong, Kaige Gao, Kang Guan, Lean Wang, Lecong Zhang, Lei Xu, Leyi Xia, Liang Zhao, Liyue Zhang, Meng Li, Miaojun Wang, Mingchuan Zhang, Minghua Zhang, Minghui Tang, Mingming Li, Ning Tian, Panpan Huang, Peiyi Wang, Peng Zhang, Qihao Zhu, Qinyu Chen, Qiushi Du, R. J. Chen, R. L. Jin, Ruiqi Ge, Ruizhe Pan, Runxin Xu, Ruyi Chen, S. S. Li, Shanghao Lu, Shangyan Zhou, Shanhuang Chen, Shaoqing Wu, Shengfeng Ye, Shirong Ma, Shiyu Wang, Shuang Zhou, Shuiping Yu, Shunfeng Zhou, Size Zheng, T. Wang, Tian Pei, Tian Yuan, Tianyu Sun, W. L. Xiao, Wangding Zeng, Wei An, Wen Liu, Wenfeng Liang, Wenjun Gao, Wentao Zhang, X. Q. Li, Xiangyue Jin, Xianzu Wang, Xiao Bi, Xiaodong Liu, Xiaohan Wang, Xiaojin Shen, Xiaokang Chen, Xiaosha Chen, Xiaotao Nie, Xiaowen Sun, Xiaoxiang Wang, Xin Liu, Xin Xie, Xingkai Yu, Xinnan Song, Xinyi Zhou, Xinyu Yang, Xuan Lu, Xuecheng Su, Y. Wu, Y. K. Li, Y. X. Wei, Y. X. Zhu, Yanhong Xu, Yanping Huang, Yao Li, Yao Zhao, Yaofeng Sun, Yaohui Li, Yaohui Wang, Yi Zheng, Yichao Zhang, Yiliang Xiong, Yilong Zhao, Ying He, Ying Tang, Yishi Piao, Yixin Dong, Yixuan Tan, Yiyuan Liu, Yongji Wang, Yongqiang Guo, Yuchen Zhu, Yuduan Wang, Yuheng Zou, Yukun Zha, Yunxian Ma, Yuting Yan, Yuxiang You, Yuxuan Liu, Z. Z. Ren, Zehui Ren, Zhangli Sha, Zhe Fu, Zhen Huang, Zhen Zhang, Zhenda Xie, Zhewen Hao, Zhihong Shao, Zhiniu Wen, Zhipeng Xu, Zhongyu Zhang, Zhuoshu Li, Zihan Wang, Zihui Gu, Zilin Li, and Ziwei Xie. 2024. Deepseek-v2: A strong, economical, and efficient mixture-of-experts language model. *Preprint*, arXiv:2405.04434.

Sara Hooker, Nyalleng Moorosi, Gregory Clark, Samy Bengio, and Emily Denton. 2020. Characterising bias in compressed models. *Preprint*, arXiv:2010.03058.

Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C. Berg, Wan-Yen Lo, Piotr Dollár, and Ross Girshick. 2023. Segment anything. *arXiv:2304.02643*.

Ziwei Liu, Ping Luo, Xiaogang Wang, and Xiaoou Tang. 2015. Deep learning face attributes in the wild. In *Proceedings of International Conference on Computer Vision (ICCV)*.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2023. Attention is all you need. *Preprint*, arXiv:1706.03762.