Using Early Readouts to Mediate Featural Bias in Distillation

Rishabh Tiwari
Google Research, India
rishabhtiwari@google.com

Durga Sivasubramanian
IIT Bombay
durgas@cse.iitb.ac.in

Anmol Mekala
University of Massachusetts Amherst
amekala@umass.edu

Ganesh Ramakrishnan
IIT Bombay
ganesh@cse.iitb.ac.in

Pradeep Shenoy
Google Research, India
shenoypradeep@google.com

Abstract

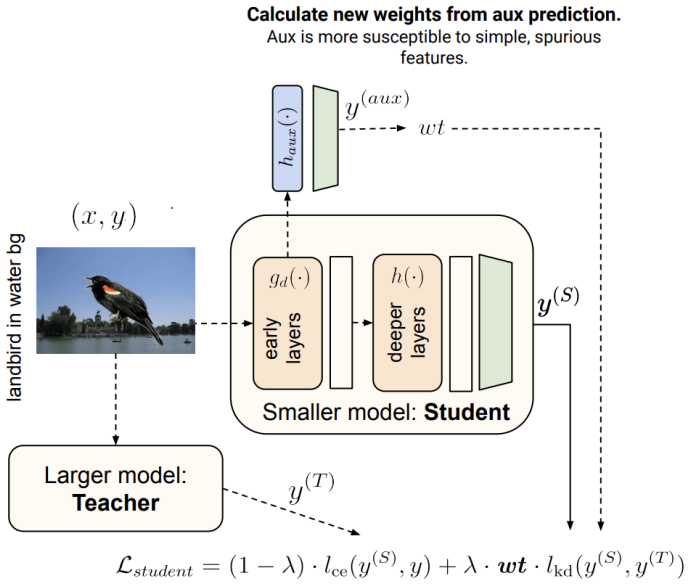
Deep networks tend to learn spurious feature-label correlations in real-world supervised learning tasks. This vulnerability is aggravated in distillation, where a student model may have lesser representational capacity than the corresponding teacher model. Often, knowledge of specific spurious correlations is used to reweight instances & rebalance the learning process. We propose a novel early readout mechanism whereby we attempt to predict the label using representations from earlier network layers. We show that these early readouts automatically identify problem instances or groups in the form of confident, incorrect predictions. Leveraging these signals to modulate the distillation loss on an instance level allows us to substantially improve not only group fairness measures across benchmark datasets, but also overall accuracy of the student model. We also provide secondary analyses that bring insight into the role of feature learning in supervision and distillation.

1. Introduction

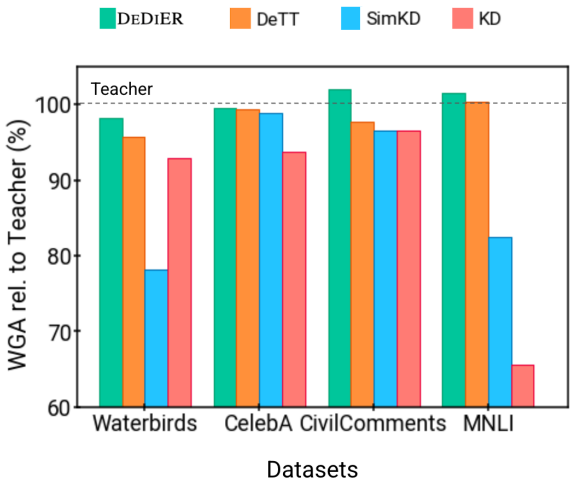
Deep networks trained via supervision tend to preferentially learn simple features with weak label correlations [7, 11, 28]. This weakness is significantly magnified in real-world applications where limited data or sampling biases may introduce spurious correlations between unrelated features and desired labels. As an example, the Waterbirds benchmark [25] challenges DNNs' dependence on habitat or background cues for classifying bird species. These challenges are significantly worsened in distillation, where both teacher model biases and student model capacity limitations may worsen fairness measures in the student [1, 5, 23].

We focus on the problem of learning debiased student models in a distillation setting. Bias is typically studied by tracking model performance by relevant subgroups of data, *e.g.*, groups where a spurious attribute is in agreement with, or in conflict with the true label. Directly optimizing such measures of bias (*e.g.*, Group DRO [25]) is limited by the need for annotations of group knowledge or feature dimensions known to be spuriously correlated. Recent work instead uses errors of a pre-trained model as a proxy for identifying problem instances, and a subsequent round of training with upweighted losses for those instances [19, 20]. These methods need to carefully titrate early stopping criteria for identifying erroneous instances, since DNNs typically overfit on training data and rapidly drive training error rates to zero.

We propose DEDIER (**De**biased **Di**stillation using **E**arly **R**eadouts) wherein we introduce two key innovations for debiasing distilled student models: **a) Early readouts** – a new source of differential information about spurious feature correlations at an instance level, and **b) Confidence weighting** – which allows for fine-grained adjustments to distillation on a per-instance level, graded by prediction confidence. Our primary insight is that spurious features are often learned at earlier levels of a neural network [11, 15, 31]; thus, prediction errors at early layers can provide significant information about feature bias in the network. Indeed, we show that "reading out" (*i.e.*, predicting with a linear decoder) instance labels from earlier representations disproportionately errs on instances that defy known spurious correlations (Fig. 2 (a, c)), indicating overdependence on those features. Further, the readouts are often *confidently* wrong on those conflicting instances as compared to other errors (Fig. 2 (b, d)). This leads naturally into our proposal for bias-mitigated distillation: modulate the teacher signal by a function of early readout confidence margin (Fig. 1a). A significant strength of our proposal is that it can be easily attached to any standard model training procedure. This means that the early readouts provide information about the specific model being trained, and also evolve through the training procedure, unlike previous approaches which depend on static identification of instances



(a) A visual illustration of DEDIER



(b) Worst-group performances of the different KD methods.

Figure 1. (a) We use predictions from an auxiliary layer applied on top of early features to determine the weights for the distillation loss. Errors from the readouts are disproportionately from learned spurious features. (b) Comparison of Worst Group Accuracies (WGA) relative to that of the Teacher's. DEDIER is best in being able to match the Teacher's WGA.

by a separate pre-trained model.

Summing up, we make the following contributions:

- We propose *early readouts* as a novel source of information for identifying the risk of bias at the instance level. Our approach does not rely on any specialized knowledge of the dataset, such as group membership or spurious attribute values in instances.
- We propose a flexible, margin-based method for reweighing the teacher-matching loss in distillation, using these early readouts, to significantly mitigate bias in student models. Our approach outperforms SOTA on not just fairness measures but also overall accuracy on well-studied debiasing benchmarks (Fig. 1b).
- We demonstrate the generality of our instance-weighting scheme by evaluating on 4 different datasets having different group biases and compositions.

2. Related Work

2.1. Bias & Group-Fairness

Machine learning models are typically trained with a goal of maximizing average performance. However, there can be hidden biases present in the data which can inadvertently be perpetuated or even amplified by ML models [3, 16]. In particular, recent literature highlights the tendency of DNNs to focus on easy-to-learn features ("simplicity bias", see *e.g.*, [7, 24, 30]). This inherent weakness of DNNs can magnify existing spurious correlations in the data by preferentially learning those features. Various algorithms for distributionally robust optimization (DRO) have been proposed to address this issue [6, 8, 14], but their formulation is too conservative. More recent work [25, 29] proposed a general formulation which explicitly optimizes over group labels. Setlur *et. al.* [27] build on these ideas without using group labels and without sacrificing overall accuracy. Other work proposes simple repeated training recipes [20] that aim to reduce errors from previous trained models as a proxy for debiasing models.

2.2. Knowledge Distillation

Knowledge distillation (KD) uses a larger *teacher* model to help small *student* models learn more than they would while using cross-entropy loss over hard labels. This network compression is most commonly achieved by making the student mimic the teacher's final output layer logits [12]. Other variants [10] also distil the teacher's intermediate feature maps for a more thorough transfer of the teacher's knowledge. SimKD [4] transplants the teacher's discriminative head to the student in addition to performing feature distillation, thus aiming for a closer matching of the teacher performance.

2.3. Bias and Distillation

The robustness and bias of compressed models has been investigated in several works. Pruning-based compression has been observed to worsen model performance on under-represented groups [13] and worsen OOD performance [5]. KD-trained small models have been observed to have amplified bias [1] and worsened OOD performance [5]. The picture that emerges from these works is that smaller models learnt via different compression methods such as pruning, KD, *etc*., tend to rely on spurious correlations while aiming to match the average accuracy of larger models. Although the larger models are less dependent on spurious correlations, this knowledge is not transferred completely during KD; this is the problem that we aim to solve.

Mitigating of bias in distillation has been previously attempted by the softening of teacher labels based on sample hardness [5], data augmentation via mixup on protected characteristics [1], and transplanting (robust) teacher layers [19]. Du *et. al.* [5] employ an ensemble of models of various sizes to decide sample hardness, whereas [1] depends on the availability of specific group annotations; thus these approaches are inadequate for our setting. Closest to our setting is [19]; we compare against and significantly outperform this approach.

3. Preliminaries

We aim to build a classifier model $f(\cdot)$ using training data $D = (x_i, y_i) \mid i \in (1, \cdots, n)$. Here, $x_i \in \mathcal{X}$ denotes the input of the i^{th} instance, and $y_i \in \mathcal{Y}$ the corresponding label. The model f is parameterized by $\theta \in \Theta$, and our goal is to learn a mapping $f_\theta : \mathcal{X} \to \mathcal{Y}$. We define the 0-1 loss function as $l_{0-1}(x, y; \theta) = \mathbf{1}[f_\theta(x) \neq y]$ representing the error incurred. In a standard classification setting, we seek to minimize $\mathbb{E}\left[\ell_{0-1}(x, y; \theta)\right]$, *i.e*., the 0-1 loss over the entire dataset.

Knowledge Distillation: Suppose we have access to a pre-trained model (teacher) that generates logits denoted as $y^{(T)} = \mathcal{T}(\boldsymbol{x})$. Then a new model (student) could be trained using a "teacher matching" objective that involves minimizing the KL-divergence between the student's logits $y^{(S)}$ and those of the teacher $y^{(T)}$ [12]. The training objective typically used to train a student model is:

$$\mathcal{L}_{student} = \sum_{D} \left((1 - \lambda) l_{ce} + \lambda l_{kd}\right) \tag{1}$$

where $l_{kd} = \tau^2 KL\big(y^{(S)}, y^{(T)}\big)$ is the teacher-matching loss, τ is a temperature parameter that controls the softening of the KL-divergence term, and $l_{ce} = H\big(y^{(S)}, y\big)$ is a supervised learning loss on the model's predictions $y^{(S)}$ compared to the true labels y. H is typically cross-entropy loss, a tractable relaxation of the 0-1 loss mentioned above.

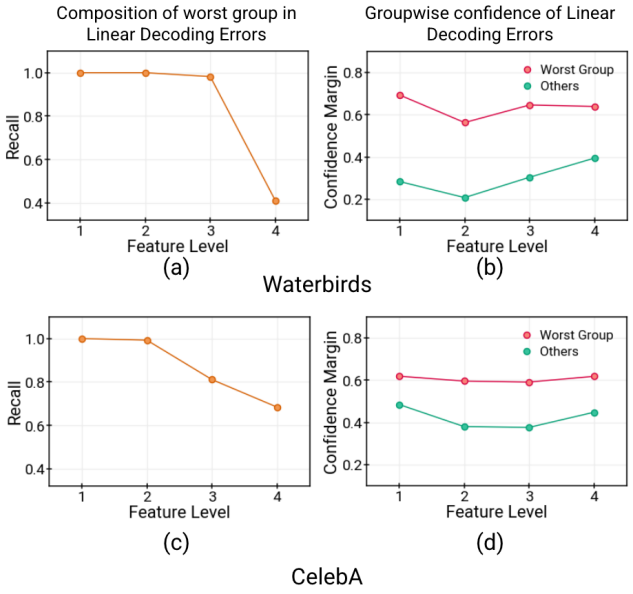


Figure 2. Early readout errors recall worst group instances (left) and worst group readouts are more confident, across layers (right). We measure linear decoding error and confidence margins at each layer, after 1 epoch. (a, c): We observe that nearly all worst group instances are misclassified ($\sim$ 100% recall) with more recall in earlier layers. (b, d): show that error instances from minority groups have significantly higher confidence margin compared to other groups. See text for more details.

Finally, λ is a hyperparameter controlling the contribution of the above two loss components.

Groups and spurious correlations: We assume that the dataset D consists of groups $g \in G$, and that each data point in D belongs to one of these groups in G. The groups are defined in terms of attribute value and label pairs where the attribute value may be spuriously correlated with the label in the supplied dataset D. As an example, in the Waterbirds dataset, the background of a given image (*e.g*., water) may be heavily associated with a specific category of foreground objects (waterbirds); however, this correlation is clearly incidental, not causal. Thus, the relevant groups for this dataset consist of different pairings of background and foreground objects. Continuing our Waterbirds example, landbirds on water are heavily misclassified, as the spurious feature (water) is in conflict with the true label. We refer to such groups interchangeably in the text as "worst groups" (groups with worst accuracy, typically) or "minority groups" (since such conflicting examples are in the minority in typical datasets).

Group fairness: Minimizing the overall 0-1 loss on a dataset does not necessarily guarantee group-wise fairness (*i.e*., comparable performance on all groups in the data); indeed, in practice, DNNs are heavily swayed by spurious correlations, leading to errors when the spurious feature's

prediction is in conflict with the label. One way of addressing this is by defining a group fairness loss such as Group DRO [25]: $\max_{g \in G} \mathbb{E}\left[\ell_{0-1}(x, y; \theta) \mid g\right]$. By minimizing this *worst group* accuracy, one can achieve equitable performance across groups. Alternatively, one could use the above loss as a measure of performance for comparing methods on their group fairness at test-time. Similar objectives can be framed for distillation as well.

Fairness without group labels: Directly optimizing a groupwise loss requires group information for each training instance, which may be limiting or even infeasible in practice. In addition to substantial labeling costs, a concern is that only some spurious correlations may be captured via explicit labeling. Instead, recent work uses model prediction errors to identify problematic instances, as a proxy for more generally identifying spurious correlations. JTT [20] trains a standard supervised classifier, collects instance errors from that classifier, and then trains a fresh classifier with a large fixed multiplier on the loss from those instances. The hope is that errors from the first model signal instances where spurious features contradict the label; by upweighting those instances, JTT shows improvements in group fairness measures even though no group information was used in training. DeTT [19] uses a similar strategy for upweighting teacher loss for error instances in distillation, by using an early-stopped ERM model for identifying error instances. These approaches depend on proxy models trained via careful early stopping criteria, and only capture a fraction of the problematic instances in the dataset.

4. Debiased Distillation with DEDIER

We now describe DEDIER, a novel method for distilling knowledge from a debiased teacher without using group information in the training data. We identify *early readouts*–label predictions from early network representations–as a novel signal for overdependence on spurious features, and leverage this signal into a graded adjustment of distillation loss on a per-instance level (Fig. 1a). Since these signals are directly accessible from the model being trained, rather than proxy models, we can develop a dynamic weighting scheme for distillation wherein our adjustments adapt to the model as it is being learned.

We first motivate the use of early readouts for automatically identifying problem instances, and more broadly, groups of instances that may suffer from spurious feature correlations (Fig. 2). We then show how *confidence margins* from early readouts can be flexibly transformed into a graded weighting scheme for distillation (Fig. 3).

4.1. Early Readouts for Precise Identification

Our key insight is that early representations in the network provide significantly richer information about errors and spurious correlations, compared to the final predictions made by a classifier. This insight is based on previous work that suggests that DNNs are led astray by "simple" features (the so-called simplicity bias [28]); further, that simple features are typically learned early in the network stack, and spread throughout later layers [11].

To illustrate this idea clearly, we trained a ResNet-18 [9] model on the Waterbirds and CelebA datasets [21,25,32] for one epoch, and trained linear decoders that operate on the representation from each layer of the network. We examined how error, and error + confidence, can be used to differentiate minority group instances from other groups. We summarize our observations next:

Early errors signal worst groups: In Fig. 2 (a) we observe, how on the Waterbirds dataset, nearly all minority instances are incorrectly classified at earlier network layers ($\sim 100\%$ recall), although this error drops rapidly at the final layer even after a single epoch of training. This is already a substantial advantage over previous approaches such as JTT & DeTT, that use error instances at the final layer of an ERM classifier to identify minority instances. The finding is replicated on the CelebA dataset (Fig. 2 (c)). This also confirms the hypothesis that early readouts give a strong indication of the network's dependence on spuriously correlated features.

Erroneous but confident: Next we examine whether the *confidence margin* associated with incorrectly labeled instances provide further discriminative information. For a probabilistic classifier output $\mathbf{p} = \{p_1, \cdots, p_K\}$, where p_k denotes the probability of label k, we define the confidence margin as the difference between the top two values of p_k:

$$p_{max} = \max_{p_k \in \mathbf{p}} p_k \tag{2}$$

$$\texttt{cm}(\mathbf{p}) = p_{max} - \max_{p_k \in \mathbf{p} \setminus p_{max}} p_k \tag{3}$$

In Fig. 2 (b, d) we present the average confidence margin on incorrectly classified instances, broken down into minority group and other groups (Waterbirds). We see that confidence margin further helps us discriminate between the two sets of erroneous instances, with minority group errors associated with significantly higher margin. Again, this differential is larger in earlier rather than later layers.

Advantages of early readouts: These results clearly bring out the following points: (a) Errors from early readouts, and in particular, the associated confidence margins, are significantly more informative than errors at final layers of the model (*e.g.*, [19,20]). (b) Since the readouts are a cheap way of monitoring network performance at an instance level, we can design *dynamic* adjustment techniques that use readouts from the classifier being learned, throughout the course of its training, instead of a fixed upweighting of a pre-selected set of instances from a proxy model's output [19,20]. As we show in our results, this weighing scheme is more effective with earlier layers.

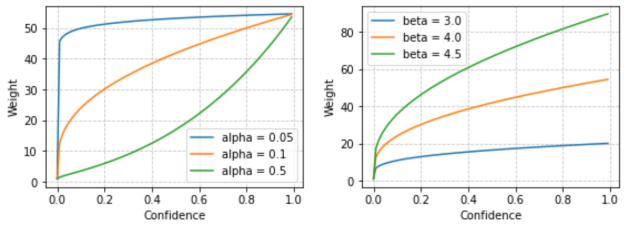


Figure 3. Weighing scheme as a function of confidence of instances mispredicted by early readouts, for different values of α and β. a) shows weighing scheme for different values of α keeping β=4. b) shows weighing scheme for different values of β keeping α=0.1

4.2. Confidence-Based Weighting for KD

In our experiments we observed that using mispredictions as the signal of an instance being from a worst group results in low-precision (∼10-20%) worst group identification. We counteract this by upweighing based on the confidence margin given Fig. 2 (b, d)'s observation of higher confidence margin for worst group instances. This helps further increase the relative contribution of worst group instances to the loss, without using group labels.

Fig. 1a sketches the broad outline of our proposal. Let the representation learned by the neural network at a given depth d be denoted by $g_d(x)$. In order to obtain early readouts, we add an auxiliary classifier layer h_a on top of g_d. The early readouts are then the classifier probabilities $\mathbf{p}^{(aux)} = h_{aux}(g_d(x))$, with an associated predicted label $y^{(aux)}$, for a data sample $(x, y) \in D$. We train the new auxiliary network for R epochs to obtain correct readouts. Then based on the early readouts, we devise a weighting scheme as follows:

$$\boldsymbol{wt} = \begin{cases} 1, & \text{if } y^{(aux)} = y \\ e^{\beta \cdot \texttt{cm}(\mathbf{p}^{(aux)})^{\alpha}}, & \text{otherwise} \end{cases} \tag{4}$$

where $\texttt{cm}(\cdot)$ is the confidence margin described in Eq. (3). In other words, if the auxiliary prediction is correct, we do not change the instance weight; however, on erroneous predictions from the auxiliary network, we scale the associated (incorrect) confidence margin through an appropriately parametrized function to generate a weight.

The hyperparameters (α, β) control the variation of weights with confidence margin, and the maximum weight associated with any misclassified point, respectively. In Fig. 3 (left) we present the influence of α when $\beta = 4$ is kept fixed; for lower α there is a sharp, substantial change in weights after a certain confidence value, and for higher α the changes are gradual. On the other hand, β simply scales up the weight as a function of confidence for a given alpha (right panel).

Algorithm 1 The DEDIER approach: learning student $\mathcal{S}$, given dataset $\mathcal{D} = (x_i, y_i) \mid i \in (1 \cdots N)$ and teacher $\mathcal{T}$.

Hyperparameters: Distillation loss fraction λ, depth d of early readout, parameters α and β for calculating the weights.

$\mathcal{S} = h(g_d(\cdot))$: break student down into early layers g_d of depth d and remaining deeper layers h.
Let $h_{aux}(\cdot)$ be the auxiliary network
We augment dataset D with weights as $\mathcal{D}_w \leftarrow (x_i, y_i, wt_i) \mid i \in (1 \cdots n)$, where $(wt_1 \cdots wt_N) \leftarrow (1 \cdots 1)_n$

for each $e \in \{1 \cdots E\}$ **do**
    Train student model $\mathcal{S}$ using the loss described in Eq. 5.
    if $e\%L == 0$ **then**:
        Train $h_{aux}(g_d(\cdot))$ for R epochs on dataset D using standard cross-entropy.
        for each $(x, y, wt) \in \mathcal{D}_w$ **do**
            $y^{(\text{aux})} = h_{aux}(g_d(x))$
            Update wt according to Eq. 4.
        end for
    end if
end for

4.3. Dynamic Reweighing for Distillation

We augment the dataset D with the weights wt and obtain $\mathcal{D}_w = \{(x_i, y, wt_i) | i \in \{i \cdots n\}\}$. We perform knowledge distillation using a loss similar to the one described in Eq. 1, with our new augmented dataset D_w:

$$\mathcal{L}_{student} = \sum_{D_w} [(1 - \lambda) \cdot l_{ce} + \lambda \cdot \boldsymbol{wt} \cdot l_{kd}] \tag{5}$$

where wt are instance-specific weights as described above. We present the complete algorithm in Algorithm 1. Unlike previous approaches, we update weights w_i throughout the training period (specifically, by retraining the auxiliary readout network $h_{aux}(\cdot)$ every L epochs). This means that our weighting scheme is not only based on the properties of the model being trained (as opposed to a pretrained proxy model [19, 20]), but also that it adapts during the training process. In particular, relative accuracies of different groups may change during training, and our approach seamlessly adapts the upweighting according to the needs of the current model (*c.f.*, Sec. 6.3).

5. Experiment Setup

5.1. Datasets

We evaluate our method and various baselines on four debiasing benchmarks: *Waterbirds*, *CelebA*, *MultiNLI* and

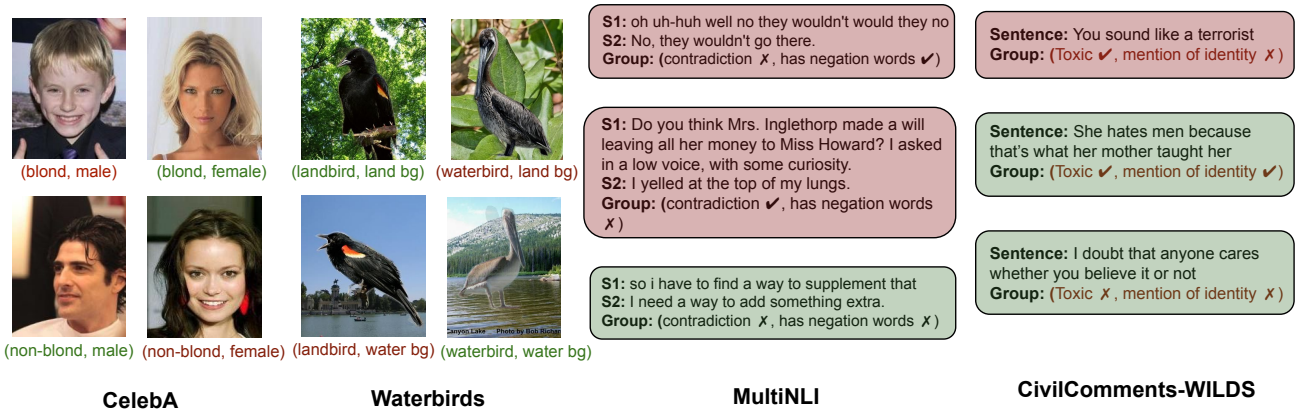


Figure 4. Groups in the four datasets. Groups which follow the correlation are in green and ones in conflict with the correlation are in red.

CivilComments-WILDS. Each of these are classification tasks where instances include a feature that is spuriously correlated with the label, as illustrated in Figure 4. In other words, the feature or attribute value is often associated with specific labels without actually having any causal relationship with the label. This sets up a natural grouping of (label, attribute-value) pairs, where the attribute value may indicate a label that is consistent with, or in contrast to the true label. Empirically, models perform worst on groups with label-attribute conflict.

Waterbirds [25, 32]: Binary classification for bird images, with "waterbird" and "landbird" classes. Since each category is often pictured on a stereotypical background, the background (water, land) is spuriously correlated with the label. The dataset also contains small numbers of bias-conflicting examples for training and evaluation.

CelebA [21]: Binary classification of face images in order to identify hair color: "blond" or "non-blond" [25]. In the dataset, most blond-haired celebrities are female, setting up a spurious correlation between gender and hair color.

MultiNLI [33]: Each input sample consists of two sentences, and the task is to determine whether the meaning of the second sentence entailed by, neutral with, or contradicts the first sentence. The presence or absence of negation words in the second sentence can misleadingly influence the prediction task [25].

CivilComments-WILDS [2, 18]: The objective is to categorize whether an online comment is toxic or not. In the dataset, the label is unintentionally correlated to references to specific demographic characteristics such as gender (male, female), ethnicity (White, Black), sexual orientation (LGBTQ), and religious affiliations (Muslim, Christian, and others). We use a binary indicator (appearance of words related to demographic identities in the comment) as the spuriously label-correlated attribute for defining groups.

5.2. Evaluation Metric

We use the standard train, validation and test splits of each dataset. We assume that validation and test set have group information. As is common practice, we use group information of validation data for tuning hyperparameters, and group information of test data for measuring the performance of methods. We report *Average Accuracy* and *Worst Group Accuracy* (WGA) on the unseen test dataset. *Average Accuracy* is the percentage of correctly predicted data points in the test set. For each dataset, test data is divided into groups as per Sec. 5.1. Given these groups, WGA is the accuracy of the group with the worst performance for a given (method, dataset) combination.

5.3. Baselines

Apart from standard knowledge distillation described in Sec. 3, we compare against the following:

Just Train Twice (JTT) [20]: This is designed for mitigating bias in supervised learning, without access to group information. The model is first trained for a small number of epochs n, and misclassified instances are selected for follow-up interventions. A second classifier is trained from scratch with the loss from previously identified instances scaled up by a single fixed scalar λ. This method is limited by the dependence on tuning n and λ, and lack of fine-grained distinction between instances.

Group DRO [25]: This is also for supervised learning, with the use of both class and group labels during training. It aims to minimize the worst-group accuracy under early stopping. The need for annotations for group information is a major limitation; however, this method can be considered a skyline for methods that don't use group information.

KD with a reused teacher classifier (SimKD) [4]: This is designed to improve knowledge distillation in general. This approach trains the student to mimic the final feature map (pre-classifier layer representation) from the teacher.

Method	Annotation	Teacher	Waterbirds		CelebA		CivilComments-WILDS		MultiNLI	
			Avg Acc.	WGA	Avg Acc.	WGA	Avg Acc.	WGA	Avg Acc.	WGA
Teacher	No	No	92.6	91.4	92.7	90.0	86.1	76.7	81.4	77.7
Group DRO	Yes	No	87.0 ± 1.65	81.1 ± 1.60	93.0 ± 0.31	86.4 ± 1.27	82.2 ± 0.80	77.3 ± 0.81	50.7 ± 1.73	47.5 ± 1.33
ERM	No	No	75.0 ± 1.07	33.8 ± 3.57	**95.9** ± 0.12	44.1 ± 1.41	**91.0** ± 0.47	57.5 ± 5.23	50.7 ± 1.04	14.1 ± 5.11
JTT	No	No	86.7 ± 0.50	80.5 ± 0.53	86.4 ± 4.65	77.8 ± 2.48	84.0 ± 3.21	60.0 ± 2.06	55.1 ± 0.89	25.2 ± 3.44
KD ($\lambda = 0.5$)	No	Yes	87.9 ± 0.90	67.9 ± 0.68	95.6 ± 0.25	62.0 ± 4.51	87.5 ± 2.94	69.1 ± 6.47	57.5 ± 0.50	46.2 ± 0.17
KD ($\lambda = 1$)	No	Yes	88.6 ± 0.31	84.9 ± 0.47	93.7 ± 0.06	84.4 ± 1.15	85.0 ± 0.50	74.0 ± 1.04	57.3 ± 0.31	51.0 ± 1.39
SimKD	No	Yes	82.1 ± 1.10	71.4 ± 2.15	93.0 ± 0.31	89.0 ± 0.35	87.0 ± 0.40	74.0 ± 2.25	70.7 ± 0.92	64.1 ± 1.62
DeTT	No	Yes	90.1 ± 0.62	87.5 ± 1.25	92.8 ± 0.35	89.5 ± 0.71	86.7 ± 0.56	75.0 ± 2.56	78.9 ± 0.49	77.9 ± 0.06
DEDIER (Ours)	No	Yes	**92.1** ± 0.39	**89.8** ± 0.47	93.2 ± 0.06	**89.6** ± 1.67	84.4 ± 0.39	**78.3** ± 0.80	**80.1** ± 0.24	**78.9** ± 0.28

Table 1. Comprehensive comparison of methods across datasets. Rows represent various baselines, alongside the Teacher model and DEDIER (gray=supervised learning, white=distillation). Columns show average accuracy and worst group accuracy (WGA) on unseen test data, grouped by dataset. DEDIER substantially improves WGA compared to other distillation baselines, while simultaneously beating them on overall accuracy on 3 out of 4 datasets. We also consistently outperform Group DRO which uses group information in optimizing worst-group accuracy for a supervised setting.

Subsequently, the teacher's classification layer is grafted onto the student model, resulting in a much closer transfer of the teacher's knowledge. To address differences in output size between the teacher and student models, a projector layer is introduced following the student model's feature layer which leads to a small increase in the student's size. Though it does not specifically address debiasing, since larger teacher models are less biased, the better distillation mitigates bias to some extent. The limitations of this method are the need for changing the student architecture and increasing the student size.

Debiasing via Teacher Transplantation (DeTT) [19]: This is the baseline closest to our setting. It aims to mitigate bias, while simultaneously improving teacher matching, combining the approaches of both JTT and SimKD. They 1) use an unbiased teacher, 2) distil the teacher's feature map using a projection layer, and append the teacher classifier layer (from SimKD), 3) assign higher weights during training to initially misclassified samples (from JTT). The limitations of JTT and SimKD are inherited in DeTT.

5.4. Model Architecture and Training

We use the Resnet-18 [9] architecture for the vision datasets, and DistilBERT [26] for the text datasets. The teacher model for knowledge distillation is Resnet-50 [9] and BERT [17] for the vision & text datasets respectively. We adopt a similar training approach as [19], training MultiNLI and CivilComments-WILDS datasets using AdamW [22] and vision datasets using SGD optimizers. We do not use learning rate schedules. The learning rates, weight decay and position of the auxiliary layer, obtained after grid search, are listed in the appendix, along with the choice of hyperparameters α, β. To reduce the search space, we keep the training interval (L) and number of training epochs for auxiliary layer (R) fixed at 1. The appendix also includes sensitivity analysis for the new hyperparameters. We train Waterbirds for 100, CelebA for 60, MultiNLI for 10 and CivilComments-WILDS for 10 epochs respectively. We employ validation worst group accuracy for early stopping. We use a ResNet depth-1 block as the auxiliary network for vision datasets, and a two layer neural network for the text datasets. We train the auxiliary network at the end of each epoch. We implement DEDIER with $\lambda = 1$.

6. Results

Tab. 1 compares DEDIER on the standard benchmark datasets and the baselines described in Section 5.3. We report *Average Accuracy* and *Worst Group Accuracy* (WGA). We aim to make significant gains in WGA, while ideally not worsening overall accuracy.

6.1. DEDIER: Debiased and Accurate

DEDIER achieves substantial gains over baselines in *Worst Group Accuracy* across datasets. An impressive result is that DEDIER also improves *Average Accuracy* across different datasets (except for the CivilComments dataset), showing that our learned classifiers are overall more robust in addition to being less biased.

This is because, unlike DeTT or JTT, we do not simply reweight a predetermined set of misclassified points that have been specified by some previous model; instead, DEDIER dynamically adapts the loss function through the mechanism of refreshing the early readout model every epoch. Thus, our classifiers are incentivized throughout the training process to focus on different problem instances, rather than an *a priori* notion of worst group as per Sec. 5.1. See Sec. 6.3 for further analysis of this key finding. We also observe that the earliest readouts give the best WGA improvements, refer to Appendix B.3 for more details.

Waterbirds groups	Teacher	DeTT	SimKD	DEDIER
(waterbird, water bg)	94.3	92.6 ± 0.70	89.4 ± 0.06	94.1 ± 0.86
(landbird, land bg)	91.6	90.0 ± 0.06	92.1 ± 0.46	**89.8** ± 0.46
(waterbird, land bg)	91.7	**88.3** ± 0.81	**71.4** ± 2.15	92.1 ± 0.40
(landbird, water bg)	**91.4**	88.8 ± 1.70	84.6 ± 0.79	90.6 ± 0.67
CelebA groups				
(blond, female)	94.3	92.6 ± 1.14	92.2 ± 0.46	92.7 ± 1.48
(non-blond, male)	92.9	92.3 ± 0.58	93.0 ± 0.46	93.2 ± 0.42
(non-blond, female)	92.1	93.0 ± 0.78	93.2 ± 0.35	93.1 ± 0.52
(blond, male)	**90.0**	**89.5** ± 0.71	**89.0** ± 0.35	**89.6** ± 1.96

Table 2. Groupwise breakdown of accuracy for Waterbirds and CelebA datasets. Shown are the teacher model, DEDIER and KD baselines DeTT and SimKD. Gray rows indicate spurious feature values supporting label; white rows indicate conflict. Boxed text highlights worst-group performance in each column.

Note that DEDIER achieves better WGA than GroupDRO [25] which assumes availability of group information. For CivilComments and CelebA, we note that the debiasing baselines (JTT and DeTT) perform worse than ERM on average accuracy.

6.2. Removing Spurious Correlations

We analyze the performance of DEDIER, alongside the Teacher model and baselines DeTT & SimKD, across different subgroups in the held-out test set (Tab. 2). The table distinguishes *bias-confirming* groups where the spurious feature value *supports* predicting the label (gray rows), and *bias-contradicting* groups where the spurious feature value and label are in apparent conflict (white rows). For instance, in Waterbirds, a water background may (incorrectly) influence the classifier to predict a waterbird as label, and in CelebA, female gender may influence the classifier to predict blond hair color. These influences are helpful in the gray rows, and harmful in the white rows.

We note the following key findings: a) DEDIER beats baselines DeTT & SimKD on not just worst group accuracy (Tab. 1), but nearly all groups that have bias-conflicting feature values (Tab. 2), and b) interestingly, our worst-group performance on Waterbirds is on a group that has a *bias-confirming* combination of spurious feature and label. This confirms that our training procedure has successfully reduced the dependence on the spurious feature, without sacrificing overall accuracy, compared to baseline methods.

6.3. Dynamics of Learning in DEDIER

A key design aspect of DEDIER is the use of readouts *from the model being trained* in order to refine the distillation loss (Eq. (5)), in contrast to JTT and DeTT. This means that the debiasing can dynamically vary throughout the training process to reflect the current state of the model.

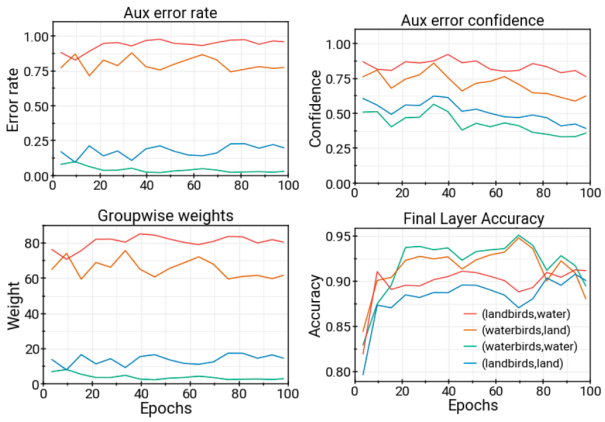


Figure 5. Evolution of reweighting during distillation (Waterbirds dataset). Top row shows error rate, and confidence of error instances, at the early readout broken down by groups. As expected, conflicting groups have high error rates due to spurious features; through the distillation process, the overconfidence reduces. Bottom row shows average weights for each group in the distillation loss (Eq. (5)), and the error rate at final layer. At the end of training, the groupwise accuracies are reconciled.

We illustrate these training dynamics in Fig. 5 for the Waterbirds dataset. We note the following interesting findings: 1) groups with spurious-feature/ label conflict have the highest early readout error (top left), and this is sustained through the training period, 2) the confidence associated with these errors reduces gradually over time (top right), 3) the average weights associated with each group changes as their performance (readout errors + confidence) changes at the auxiliary layer (bottom left v/s top left), 4) through the process of debiased distillation in DEDIER, conflicting and non-conflicting group accuracies are eventually reconciled, to be in close agreement (bottom right).

7. Conclusion

We presented DEDIER, which debiases distillation by automatically identifying and mitigating student models' dependence on spurious correlations. We observe that labels decoded from earlier layers disproportionately and confidently fail on instances with conflict between label and spurious features. This novel finding shows that early readouts flag the risk of overdependence on spurious features, and leads naturally to a scheme for graded, per-instance adjustments to the distillation loss. We show via extensive experiments that DEDIER not only improves worst group accuracy but also overall accuracy across benchmark datasets, and that the adjustments to the distillation loss adapt to the evolution of the student model. In future work, we hope to explore the quality and robustness of features learnt by the student network, and the applicability of these ideas to broader settings such as domain generalization.

References

[1] Jaimeen Ahn, Hwaran Lee, Jinhwa Kim, and Alice Oh. Why knowledge distillation amplifies gender bias and how to mitigate from the perspective of distilbert. In *Proceedings of the 4th Workshop on Gender Bias in Natural Language Processing (GeBNLP)*, pages 266–272, 2022. 1, 3

[2] Daniel Borkan, Lucas Dixon, Jeffrey Sorensen, Nithum Thain, and Lucy Vasserman. Nuanced metrics for measuring unintended bias with real data for text classification. In *World Wide Web (WWW)*, pages 491–500, 2019. 6

[3] Joy Buolamwini and Timnit Gebru. Gender shades: Intersectional accuracy disparities in commercial gender classification. In *FAT*, 2018. 2

[4] Defang Chen, Jian-Ping Mei, Hailin Zhang, Can Wang, Yan Feng, and Chun Chen. Knowledge distillation with the reused teacher classifier. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 11933–11942, 2022. 2, 6

[5] Mengnan Du, Subhabrata Mukherjee, Yu Cheng, Milad Shokouhi, Xia Hu, and Ahmed Hassan Awadallah. Robustness challenges in model distillation and pruning for natural language understanding. *arXiv preprint arXiv:2110.08419*, 2021. 1, 3

[6] John Duchi and Hongseok Namkoong. Learning models with uniform performance via distributionally robust optimization. *arXiv preprint arXiv:1810.08750*, 2018. 2

[7] Robert Geirhos, Jörn-Henrik Jacobsen, Claudio Michaelis, Richard Zemel, Wieland Brendel, Matthias Bethge, and Felix A Wichmann. Shortcut learning in deep neural networks. *Nature Machine Intelligence*, 2(11):665–673, 2020. 1, 2

[8] Tatsunori Hashimoto, Megha Srivastava, Hongseok Namkoong, and Percy Liang. Fairness without demographics in repeated loss minimization. In *International Conference on Machine Learning*, pages 1929–1938. PMLR, 2018. 2

[9] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016. 4, 7

[10] Byeongho Heo, Jeesoo Kim, Sangdoo Yun, Hyojin Park, Nojun Kwak, and Jin Young Choi. A comprehensive overhaul of feature distillation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 1921–1930, 2019. 2

[11] Katherine Hermann and Andrew Lampinen. What shapes feature representations? exploring datasets, architectures, and training. *Advances in Neural Information Processing Systems*, 33:9995–10006, 2020. 1, 4

[12] Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the knowledge in a neural network, 2015. 2, 3

[13] Sara Hooker, Nyalleng Moorosi, Gregory Clark, Samy Bengio, and Emily Denton. Characterising bias in compressed models. *arXiv preprint arXiv:2010.03058*, 2020. 3

[14] Dirk Hovy and Anders Søgaard. Tagging performance correlates with author age. In *Annual Meeting of the Association for Computational Linguistics*, 2015. 2

[15] Jie Hu, Li Shen, Samuel Albanie, Gang Sun, and Andrea Vedaldi. Gather-excite: Exploiting feature context in convolutional neural networks. *Advances in neural information processing systems*, 31, 2018. 1

[16] David Jurgens, Yulia Tsvetkov, and Dan Jurafsky. Incorporating dialectal variability for socially equitable language identification. In *Annual Meeting of the Association for Computational Linguistics*, 2017. 2

[17] Jacob Devlin Ming-Wei Chang Kenton and Lee Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of naacL-HLT*, volume 1, page 2, 2019. 7

[18] Pang Wei Koh, Shiori Sagawa, Henrik Marklund, Sang Michael Xie, Marvin Zhang, Akshay Balsubramani, Weihua Hu, Michihiro Yasunaga, Richard Lanas Phillips, Irena Gao, Tony Lee, Etienne David, Ian Stavness, Wei Guo, Berton A. Earnshaw, Imran S. Haque, Sara Beery, Jure Leskovec, Anshul Kundaje, Emma Pierson, Sergey Levine, Chelsea Finn, and Percy Liang. WILDS: A benchmark of in-the-wild distribution shifts. In *International Conference on Machine Learning (ICML)*, 2021. 6

[19] Jiwoon Lee and Jaeho Lee. Debiased distillation by transplanting the last layer. *arXiv preprint arXiv:2302.11187*, 2023. 1, 3, 4, 5, 7

[20] Evan Z Liu, Behzad Haghgoo, Annie S Chen, Aditi Raghunathan, Pang Wei Koh, Shiori Sagawa, Percy Liang, and Chelsea Finn. Just train twice: Improving group robustness without training group information. In *International Conference on Machine Learning*, pages 6781–6792. PMLR, 2021. 1, 2, 4, 5, 6

[21] Ziwei Liu, Ping Luo, Xiaogang Wang, and Xiaoou Tang. Deep learning face attributes in the wild. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 3730–3738, 2015. 4, 6

[22] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *International Conference on Learning Representations*, 2019. 7

[23] Michal Lukasik, Srinadh Bhojanapalli, Aditya Krishna Menon, and Sanjiv Kumar. Teacher's pet: understanding and mitigating biases in distillation. *arXiv preprint arXiv:2106.10494*, 2021. 1

[24] Junhyun Nam, Hyuntak Cha, Sungsoo Ahn, Jaeho Lee, and Jinwoo Shin. Learning from failure: De-biasing classifier from biased classifier. *Advances in Neural Information Processing Systems*, 33:20673–20684, 2020. 2

[25] Shiori Sagawa, Pang Wei Koh, Tatsunori B. Hashimoto, and Percy Liang. Distributionally robust neural networks for group shifts: On the importance of regularization for worst-case generalization. In *International Conference on Learning Representations (ICLR)*, 2020. 1, 2, 4, 6, 8

[26] Victor Sanh, Lysandre Debut, Julien Chaumond, and Thomas Wolf. Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter. *arXiv preprint arXiv:1910.01108*, 2019. 7

[27] Amrith Setlur, Don Dennis, Benjamin Eysenbach, Aditi Raghunathan, Chelsea Finn, Virginia Smith, and Sergey Levine. Bitrate-constrained DRO: Beyond worst case robustness to unknown group shifts. In *NeurIPS 2022 Workshop on*

Distribution Shifts: Connecting Methods and Applications, 2022. 2

[28] Harshay Shah, Kaustav Tamuly, Aditi Raghunathan, Prateek Jain, and Praneeth Netrapalli. The pitfalls of simplicity bias in neural networks. *Advances in Neural Information Processing Systems*, 33:9573–9585, 2020. 1, 4

[29] Tasuku Soma, Khashayar Gatmiry, and Stefanie Jegelka. Optimal algorithms for group distributionally robust optimization and beyond. *arXiv preprint arXiv:2212.13669*, 2022. 2

[30] Damien Teney, Ehsan Abbasnejad, Simon Lucey, and Anton Van den Hengel. Evading the simplicity bias: Training a diverse set of models discovers solutions with superior ood generalization. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 16761–16772, 2022. 2

[31] Rishabh Tiwari and Pradeep Shenoy. Overcoming simplicity bias in deep networks using a feature sieve. In Andreas Krause, Emma Brunskill, Kyunghyun Cho, Barbara Engelhardt, Sivan Sabato, and Jonathan Scarlett, editors, *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pages 34330–34343. PMLR, 23–29 Jul 2023. 1

[32] C Wah, S Branson, P Welinder, P Perona, and S Belongie. The Caltech-UCSD Birds-200-2011 dataset. Technical report, California Institute of Technology, 2011. 4, 6

[33] Adina Williams, Nikita Nangia, and Samuel Bowman. A broad-coverage challenge corpus for sentence understanding through inference. In *Association for Computational Linguistics (ACL)*, pages 1112–1122, 2018. 6