This ICCV workshop paper is the Open Access version, provided by the Computer Vision Foundation.
Except for this watermark, it is identical to the accepted version;
the final published version of the proceedings is available on IEEE Xplore.

Bridging the gap between debiasing and privacy for deep learning

Carlo Alberto Barbano*, Enzo Tartaglione*†, Marco Grangetto*
{name.surname}@unito.it
*University of Turin, Torino, Italy
†Télécom Paris, Paris, France

Abstract

The broad availability of computational resources and the recent scientific progresses made deep learning the elected class of algorithms to solve complex tasks. Besides their deployment, two problems have risen: fighting biases in data and privacy preservation of sensitive attributes. Many solutions have been proposed, some of which deepen their roots in the pre-deep learning theory. There are many similarities between debiasing and privacy preserving approaches: how far apart are these two worlds, when the private information overlaps the bias?
In this work we investigate the possibility of deploying debiasing strategies also to prevent privacy leakage. In particular, empirically testing on state-of-the-art datasets, we observe that there exists a subset of debiasing approaches which are also suitable for privacy preservation. We identify as the discrimen the capability of effectively hiding the biased information, rather than simply re-weighting it.

1. Introduction

In the latest years Deep Learning (DL) models have received a huge interest from the research community. The determinant factor towards such a huge success certainly lies in the recent deployment of high-performance hardware resources (like TPUs), the relative easiness of training complex models by simply minimizing a global objective function through gradient descent-based approaches, and the typical broad data availability. Overall, deep models are considered as "universal problem solving tools" [36]. However, these models show vulnerabilities towards privacy preservation in data: model-inversion attacks are able to retrieve sensitive information from models trained to solve some specific tasks [29]. In order to discipline AI for data privacy, the European Union is currently drafting the General Data Protection Regulation (GDPR) [1], defining a set constraints that DL models must satisfy towards guaranteeing data privacy.

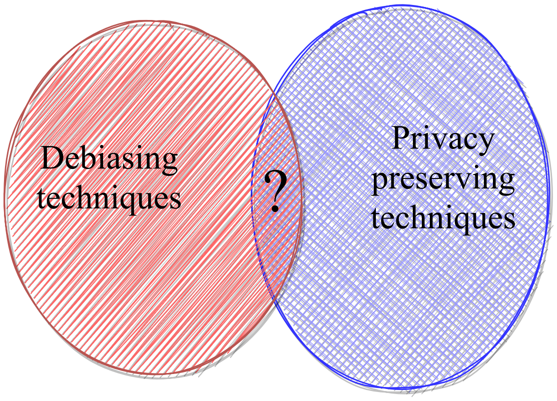


Figure 1: Is there some debiasing technique which is also privacy-preserving?

The problem of data privacy in Artificial Intelligence (AI) based algorithms deepens its roots before the vast uprising of DL models, providing solutions to this problem proposing approaches when aggregating data like K-anonymity [25, 37], at training like differential privacy [11, 19, 30, 35, 44] and homomorphic encryption [15, 32] and at inference time like information-theoretic privacy [17]. Of course, these approaches have also been successfully deployed in DL scenarios, proposing a large quantity of variants.

Focusing on guaranteeing privacy at training time, we find approaches working on the input of the trained model (meaning that the private features are removed at the source itself), on the output (meaning that the information is tentatively hidden at the output of the model) or on the various update steps of the model itself, introducing noise which de-correlates the private information. Looking at this categorization for the approaches guaranteeing privacy in DL, we observe significant resemblances with state-of-the-art debiasing approaches: how different debiasing is from guaranteeing data privacy, when the bias overlaps the feature we desire to keep private?

In this work we investigate differences between privacy-preserving and debiasing techniques, defining conditions for which a debiasing approach can be also used in a privacy-preserving scenario. In particular, we conduct a study on a synthetic dataset where we have direct and noiseless control on the correlation between features we desire to keep private and target, comparing four different debiasing representants. We observe that, despite the semantic closeness of the the two concepts, *not all* the debiasing approaches can also be deployed for privacy preservation, providing insights for the outcomes. We also conduct experiments on real dataset, validating the results achieved on the synthetic dataset. The key point of this work is not to glorify debiasing strategies over the privacy preserving ones, but to bridge these two world, showing that some debiasing techniques can also be used for privacy preservation (hence, the evaluation of the effectiveness of debiasing approaching over privacy preservation ones is out of the scope of this work).

The rest of the work is structured as follows. In Sec. 2 we provide an overview over some of the most important privacy preservation approaches. Then, in Sec. 3 we provide also a review over the debiasing categories of algorithms, selecting four candidates and addressing our testing setup. In Sec. 4 we perform an empirical evaluation and we address some considerations over the nature of the tested debiasing algorithms and, in Sec. 5, the conclusions are drawn.

2. Privacy in deep learning

Privacy-preserving approaches ideally aim at hiding some information, making it un-recoverable (or difficult to recover) from a potential attacker. The concept of privacy-aware learning is not novel in machine learning. One of the very first works in such an area was published in the far 1965 by Warner [41]. In particular, they were suggested privacy-preserving methods for survey sampling. Following this path, in the 70s many works have been proposed on different areas, like census taking and analysis of tabular data by Fellegi [13]. Very recently, thanks to the increase of computational capabilities, many works have been proposed on privacy-preserving in computational frameworks. We can divide these into the following categories.

Data anonymization. These approaches address the problem of collecting data from many different sources making impossible to beck-tracing their source. To these approaches, the most common approach, especially in the medical domain, we find vanilla data anonymization: this simple approach consists in simply hiding the sensitive metadata information consisting the information to be kept private, according to the most recent GDPR [1]. Such data cleaning procedure is standard for releasing medical imaging, where the original DICOM file format, by standard, contains sensitive information for the patients, like name, birth date and gender of the patient. However, this is certainly not sufficient to prevent back-tracing information: Narayan and Shmatikov, for example, were able to recover sensitive anonymized information from the Netflix prize dataset [29].

K-anonymization. More advanced and safe data aggregation approaches consist, for example, in guaranteeing the so-called k-anonimity. Sweeney proposed a framework for which anonymity of data is guaranteed when compared to $k-1$ others, and it is mainly thought to fight re-identification, guaranteeing the redundancy of similar features [37]. A large limit of this technique consists in its low performance on high-dimensional data, which is a common setup in DL scenarios.

Homomorphic encryption. This is a special category of encryption which allows users to perform computation directly on encrypted data, without the need of decrypting those [15]. Despite such an approach, by definition, is able to discourage mining of private information from the attackers, its computational complexity is also very high, limiting its deployment in real scenarios [32].

Multi-party computation. A current challenge for deep learning, especially in the medical field, lies in the impossibility of publicly sharing data. Due to physical, ethical and legal constraints, it might happen data can not leak outside the infrastructure where they have been created [26]. Towards this end, federated learning-based approaches are uprising: they consist in having the dataset distributed across many infrastructures. Each of these train independently a DL model and occasionally exchange information about the trained model, which might involve quantities like the model's parameters or the gradients [22]. This approach, if achieved in a round-robin fashion, averages naturally the information related to the private features. A major drawback of this approach, however, lies in the need for intensive communication between the infrastructures, which significantly slows the training process [28].

Differential privacy. Differential privacy is a very general approach to withhold private information from individuals in a set of data. In general, we can say that if data belonging to different individuals (or in our context, belonging to different private classes) are sufficiently close to be each other indistinguishable, the private information can not be retrieved. Behind this very simple yet effective idea, a number of approaches have been proposed and can be categorized in four groups.

- *Perturbing the input data.* Introducing a proper perturbation to the data themselves can hide the private target information. To this class belong centralized approaches [11] and recently, de-centralized alternatives have been proposed as well [12, 21].
- *Perturbing the output of the trained model.* This approach consists in applying a sufficiently large noise to the output of the model such that the samples belonging to different private classes are each other indistinguishable [19]. However, efficiently computing the noise to be applied in a high-dimensional scenario is not straightforward due to the non-convexity of the objective function: to this end, convex proxies have been recently proposed to overcome this obstacle [30, 31].
- *Perturbing the gradient update.* Applying a specific noise to the update signal for the model is possible to enhance differential privacy in the model. Towards this approach, many proposals, ranging from the deployment of a distributed framework [35] to the design of momentum-based optimizers accounting for the private class membership [2] have been proposed. The main drawback of these strategies lies in the low convergence and high computation complexity required.
- *Perturbing the target labels.* Finally, deploying noise to the target labels for the learning task can also be deployed to hide the private information, despite such an approach is mainly meant to boost gradient perturbation approaches [43, 44].

3. Debiasing in deep learning

In this section we first provide an overview over standard approaches taken for debiasing, then we bridge debiasing and privacy preserving algorithms and finally we descrive the overall test setup to empirically evaluate the effectiveness of debiasing strategies to also preserve privacy.

3.1. Overview

In the recent years a large number of debiasing approaches have been proposed: we can categorize them as follows.

De-biasing from the data source. It is known that datasets are typically affected by biases. In their work, Torralba and Efros [40] show how biases affect some of the most commonly used datasets, drawing considerations on the generalization performance and classification capability of the trained ANN models. Following a similar approach, Tommasi *et al.* [39] conduct experiments reporting differences between a number of datasets and verifying how final performances vary when applying different de-biasing strategies in order to balance data. Working at the dataset level is in general a critical aspect, and greatly helps in understanding the data and its structure [10].

Ensembling approaches. A typical debiasing strategy consists in training a pool of model and evaluating as an outcome a common score between these. The training strategy in this case can be addressed in many different ways. For example, ReBias [6] consists in solving a min-max problem, where the target is to promote the independence between the network prediction and all biased predictions. This comes at the cost of training multiple models and to solve the non-trivial min-max problem.

Identifying the known unknowns. Towards enhancing robustness in DL models to biases, it is uprising the challenge of finding the so-called “known unknowns” [5]. These consists in features unintentionally caught by the DL models which unexpectedly drive the model towards having a high confidence score over a wrong outcome. Identifying the “known unknowns” [5] and optimize on those using a neural networks ensemble is the approach proposed in LearnedMixin [9].

De-biasing within the deep model. Another approach attempts to achieve debiasing at training time, within the trained model itself, without the need to train extra models. This approach includes the inclusion of some corrective loss term. This can be exploited at two different levels.

- *Correcting the loss.* Some specific re-weighting of the loss function can improve generalization capability of the model. Recently, RUBi has been proposed [8], where logit re-weighting is proposed to lower the bias impact in the learning process.
- *Regularizing on the model’s bottleneck.* Typically, in the DL models, it can be identified some bottleneck layer, where the dimensionality of the extracted features is minimal. In such a space, some regularization approach on the features is possible: for example,

Table 1: Similarities between privacy preservation strategies and debiasing approaches.

Privacy preserving	Debiasing
Data anonymization	Debiasing from data source
K-anonymization	Identifying known unknowns
Multi-party	Ensembling
Differential privacy	Debiasing from data source
	Debiasing within the deep model

EnD [38] proposes a regularization term, where biased features are each other disentangled, while the unbiased target ones are, on the contrary, entangled.

3.2. Relationship between debiasing and privacy preservation

From a high-level view, there are many resemblances between the privacy preserving approaches discussed in Sec. 2 and typical debiasing strategies, reviewed in Sec. 3.1.

Let us define $\boldsymbol{x}_i$ as the input data for our DL model and $\mathcal{T}(\boldsymbol{x}_i)$ as the target class associated to $\boldsymbol{x}_i$. We can identify further attributes of the data, depending on the context:

- for debiasing purpose, we indicate with $\mathcal{B}(\boldsymbol{x}_i)$ the bias label associated to $\boldsymbol{x}_i$. This indicates any attribute (e.g. gender) which constitutes a bias in the training data;
- for privacy preserving purpose, we indicate with $\mathcal{P}(\boldsymbol{x}_i)$ the private class label associated to $\boldsymbol{x}_i$. This can represent any private attribute (e.g. identity).

Given that in this work we aim at analysing the usage of debiasing techniques in privacy preserving contexts, we assume that these two notions are equivalent. Hence, from now on, we will refer to $\mathcal{P}(\boldsymbol{x}_i)$ only.

Looking at approaches directly working at the data source $\boldsymbol{x}_i$, data anonymization strategies and differential privacy (where input is perturbed) share similar concepts to those exploited within the debiasing at the source ones. In particular, in both cases the input data $\boldsymbol{x}_i$ is altered, preserving the information related to $\mathcal{T}(\boldsymbol{x}_i)$ but erasing (in the case of privacy preservation approaches) or re-weighting (for debiasing approaching) the information related to $\mathcal{P}(\boldsymbol{x}_i)$.

K-anonymization, in a broad perspective, resembles the research for unknown unknowns: if non-trivial common features between biased data can be found, then we can say there are "known unknowns", which is solved when these data are each other confused. Similarly, in K-anonymization there is an explicit constraint on the number of samples to be compared to look at these correlations.

Another interesting analogy builds-up between multi-party computation and ensembling approaches. In both cases, a pool of models is trained and they are deployed altogether at inference time. However, while in multi-party computation data are typically data de-centralized, ensembling approaches allow data centralization. The two approaches meet under the federated learning roof, where multiple DL instances are trained in parallel, and occasionally synchronized.

One of the broadly-used approach for privacy preservation, differential privacy, meets debiasing within the deep models approaches with significant similarities. Indeed, adding regularization constraints at training time for the DL model, as a general concept overlaps differential privacy strategies, where perturbation on the output/labels/gradients is applied [7, 20]. In this case, however, the difference is subtle. Differential privacy confuses two examples such that , if $\mathcal{P}(\boldsymbol{x}_i) = \mathcal{P}(\boldsymbol{x}_j)$, the probability of recovering such an information is as low as some ε. On the contrary, debiasing within the DL model strategies reweight/remodel correlations between data, not necessarily wiping-out the information related to the bias class membership.

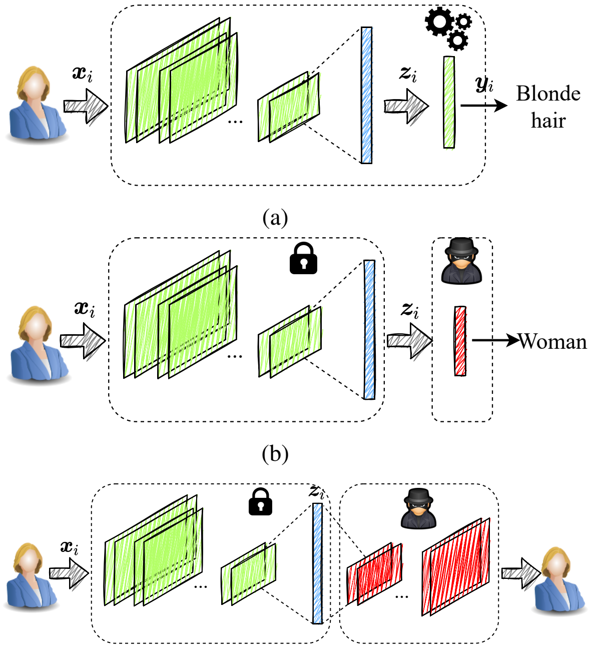
An overview of the resemblances between privacy preserving strategies and debiasing is visualized in Table 1. Overall, we can say that while privacy preserving approaches erase or hide some information to prevent an attacker can recover it, debiasing approaches reweight it. Are there debiasing techniques which completely remove the private information? We have selected four debiasing techniques, representing the macro-categories when deploying a training for a model, as discussed in Sec.3.1:

- LearnedMixin [9] as a technique to identify known unknowns;
- RUBi [8] for debiasing within the deep model, adding a corrective term to the loss.
- ReBias [6] as ensembling approach;
- EnD [38] for debiasing within the deep model, regularizing the bottleneck.

3.3. Testing framework

In order to assess the presence/absence of private information on the trained DL models, we design a model inversion-like and membership inference strategy. In such a frame, the attacker attempts to infer some attributes or private class membership from the output of a DL model, or to exactly reconstruct the input [42]. Hence, our general framework consists of two main steps.

1. *Train the model.* In this step, we train the DL model (Fig. 2a). In this phase, standard learning strategy is used, and eventually a debiasing strategy can be deployed besides training, attempting to hide the information related to $\mathcal{P}(\boldsymbol{x}_i)\forall i$. In this work, we name the accuracy measured on the target classes *Target Accuracy*.



(a)

(b)

(c)

Figure 2: Standard training on some target feature, like hair color recognition (a), gender membership recovery (b) and input reconstruction (c). In the image, in green are the layers deployed at training time (where parallelograms are convolutional layers while the rectangular box is a fully-connected layer), in red the layer trained by the attacker to obtain the private information from the bottleneck layer and in blue a plain reshaping layer.

2. *Attack.* After train is completed, an attacker attempts to recover the information of $\mathcal{P}(\boldsymbol{x}_i)$. This is typically conducted at the output of the model. Considering that most of the typically deployed DL models do not explicitly have a non-linear activation function at their output (softmax's effect is of mere normalization in range $(0; 1)$, hence to retrieve the top-1 class it is sufficient to find the maximum value of the logits), the classification fully-connected layer is a linear mapping from the previous layer's output (we name it *bottleneck layer*). Hence, we extract the output $\boldsymbol{z}_i$ from the bottleneck layer and we train a classifier to retrieve $\mathcal{P}(\boldsymbol{x}_i)$ (Fig. 2b). We choose to address the attack on $\boldsymbol{z}_i$ because its features are typically richer and the dimensionality is still sufficiently low. We indicate the accuracy measured on the private classes with *Private Class Accuracy*. Besides the private class membership, we can also attempt to recover the original input $\boldsymbol{x}_i$ using a similar approach as proposed in [14] (Fig. 2c).

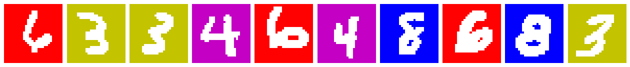


Figure 3: **Biased-MNIST** dataset: the background colors highly correlate with the digit classes, according to the value of $\rho \in [0.1; 1.0]$ (the higher, the most the correlation).

4. Experiments

In this section, we present the experiments we conducted using different debiasing techniques in order to assess whether they can also prevent private information leakage. We perform our experiments on four datasets: *Biased-MNIST, CelebA, IMDB Face dataset, SIIM-FISABIO-RSNA*.[1]

Setup All the following setups are taken from the cited known literature, which we also take as as a reference.
For Biased-MNIST, we use the network architecture proposed by Bahng *et al.* [6], consisting of four convolutional layers with 7 × 7 kernels. We use the Adam optimizer with a learning rate of 10^{-4}, a weight decay of 10^{-4} and a batch size of 256. We train for 80 epochs. We do not use any data augmentation scheme.
For CelebA and IMDB, we deploy a ResNet-18 model [16], trained with a learning rate of 10^{-4} and a batch size of 256. We train for 50 epochs. On IMDB, we follow [23, 38] by binning the age values in the intervals 0-19, 20-24, 25-29, 30-34, 34-39, 40-44, 45-49, 50-54, 55-59, 60-64, 65-69, 70-100, proposed by Alvi *et al.* [3].
For SIIM-FISABIO-RSNA we split the dataset in a training set comprising the 85% of the scans, a validation set of 5% scans and a test set of the remaining 10%. We train a DenseNet-121 model [18] to classify over two classes: “Negative for Pneumonia” and “Typical Appearance”. The training has been performed using SGD, with an initial learning rate of 0.1, decayed by a factor 10 after no improvement over the validation set loss has been detected for 5 consecutive epochs. The training stops when the learning rate drops below 10^{-3}. We use batch size 16 with momentum of 0.9 and weight decay of 10^{-4}.

4.1. Color leakage in Biased-MNIST

As base benchmark, we employ the synthetic dataset Biased-MNIST, recently proposed by Bahng *et al.* [6]. In these experiments, our goal is to assess whether the color-related information is leaked by the models. By using synthetic data, which we have full control over, we are able to perform an in-depth comparison of different techniques under varying level of difficulty. This dataset is constructed

[1] The code is available at https://github.com/EIDOSlab/bridging-debiasing-privacy-deep-learning.

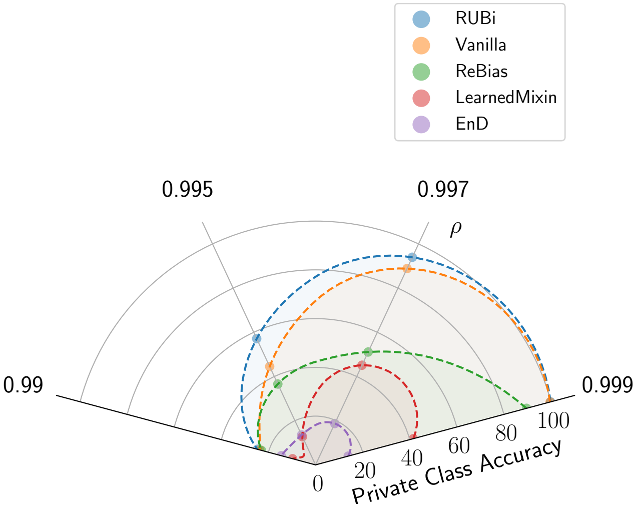


Figure 4: **Biased-MNIST private class accuracy.** The closer a curve is to the origin of the polar plot, the better the corresponding technique is at preventing private information leakage.

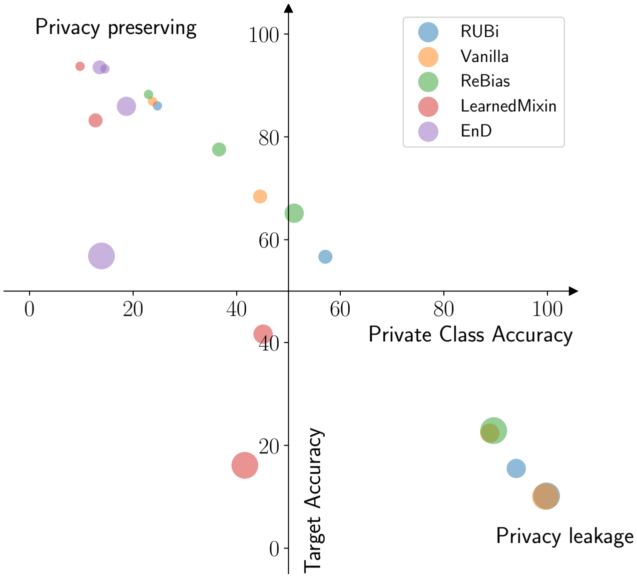


Figure 5: **Private class accuracy vs target accuracy.** Larger markers indicate higher values of ρ.

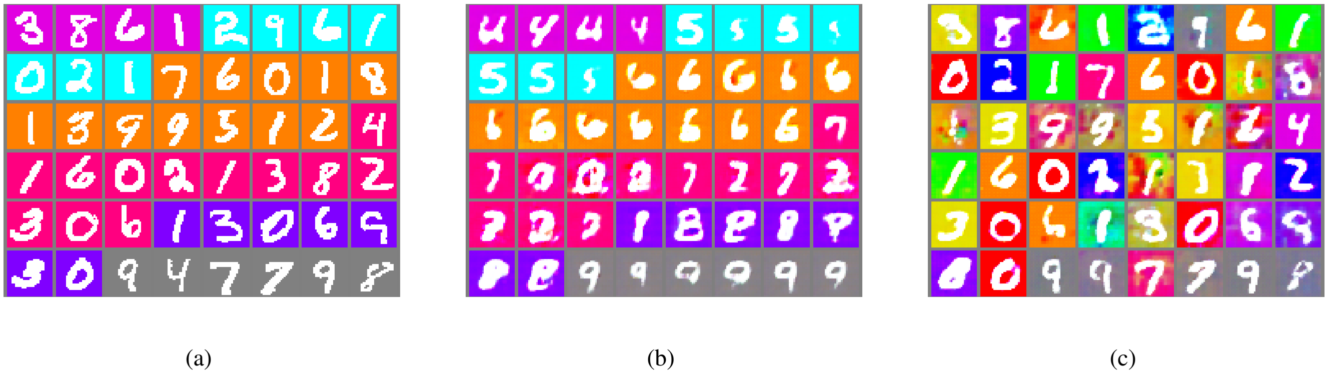
from the MNIST dataset [24] by injecting a color into the images background, as shown in Fig. 3. Each digit is associated with one of ten pre-defined colors, which will be our bias/private feature. The correlation between a digit and a background color is determined by the hyperparameter ρ. For example, a value of $\rho = 0.99$ means that a digit will have the same color 99% of the times. To vary the level of difficulty in the dataset, we select $\rho \in \{0.990, 0.995, 0.997, 0.999\}$, as done in [6]. An *unbiased* testing set is constructed following the same criterion, with $\rho = 0.1$. Given the low correlation between color and digit class in the unbiased test set, models must learn to classify shapes instead of colors in order to reach a high accuracy.

Results First of all, we measure the accuracy on the digit classification for all the techniques. As expected, the results obtained by a vanilla model heavily suffer from the color bias, especially when ρ is higher (10.4% with $\rho = 0.999$, 33.4% with $\rho = 0.997$). All of the debiasing techniques show an improvement with respect to the baseline model, with EnD and ReBias showing the highest gap in the most difficult setting (52.30% and 22.7% respectively, with $\rho = 0.999$). We expect an attack to be trivial on a vanilla model, and we also hypothesize that it could be prevented by some of the debiasing techniques. Fig. 4 shows the private class accuracy obtained by the linear classifier at the different values of ρ. The vanilla model shows in fact a significant leakage of color-related information, as the attack reaches almost 100% accuracy in the higher range of ρ. Surprisingly, not even RUBi manages to prevent an attack, obtaining performances even worse than vanilla. Considering all of the difficulty settings, the techniques which better prevent privacy leakages are LearnedMixin and EnD. In order to rank the different techinques, in Fig. 5 we compare the private class accuracy and the target accuracy. Debiasing algorithms which are able to avoid leakages while retaining (or improving) the target accuracy are found in the top left portion of the plot. From this analysis, we find EnD to be the best performing technique, followed by LearnedMixin and Rebias.

We now concentrate on the best technique (EnD), and we further assess the absence of a privacy leakage by conducting a model attack, as pictured in Fig. 2c. Fig. 6 shows the reconstructed images. When using a vanilla encoder, the color is fully preserved (and the digit is transformed into the corresponding training class). On the other hand, with a EnD-regularized encoder, the digit information is preserved while the color information is almost completely removed, as it seems to be randomly guessed by the decoder.

4.2. Gender leakage from face images

Next, we focus on gender information leakage on real facial images on two different tasks: face attributes classification and age prediction. For the first task, we employ CelebA [27], a dataset of 202,599 images, which provides 40 binary attributes for every image. As target attributes, we use the hair color and the presence of makeup. As private class membership we use the gender attribute (male or female). This choice is dictated by the fact that there is a high correlation between these attributes (i.e. most women have blond hair or wear heavy makeup in this dataset). For age prediction, instead, we use the IMDB Face dataset [33].


(a)

(b)

(c)

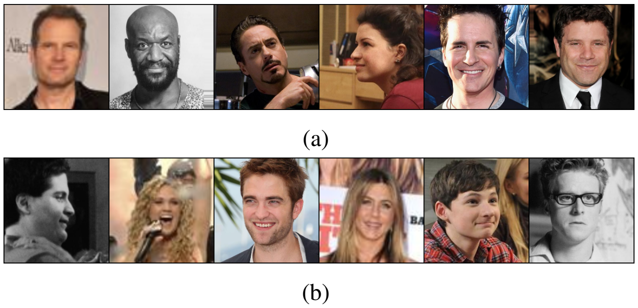
Figure 6: **attack on Biased-MNIST:** (a) ground truth images (b) decoder trained from a privacy leaking encoder (c) decoder trained with a *EnD*-regularized encoder.

Even though CelebA provides an age attribute (*Young*), we prefer IMDB as it provides a real age value, allowing for regression tasks. This dataset contains 460,723 face images, which besides age, are also annotated with gender information. Conforming to [23, 38], 20% of the IMDB dataset is used as test set, selecting samples with age 0-29 or 40+. To introduce some kind of bias in the training data, the remaining data is then split into two training subset, named extreme-biased (EB): *EB1* which contains women aged 0-29 and men with aged 40+, and *EB2* which contains men aged 0-29 and women 40+. An example of the EB1 and EB2 training sets is shown in Fig. 7.

Results Results for the CelebA dataset are presented in Tab. 2. As for the Biased-MNIST experiments, we observe an increase of the target accuracy when employing EnD for both of the classification tasks. Compared to the baseline, we also observe a significant decrease in the accuracy of the attack. Considering that the provided gender attribute is binary, an accuracy of 50% represents a random guess by the attacker, meaning that there is no private information leakage. The same considerations apply to the age regression task on the IMDB dataset. Tab. 3 shows the results. Here, we obtain an accuracy of around 50% on both the training sets. We further investigate the effect of the debiasing technique on the model, by analyzing the distribution of the latent space of a vanilla model compared to a regularized model. We fit a gaussian distribution on the principal component of the embeddings computed on the IMDB dataset. Fig. 8 shows the distributions. We observe that, while in the vanilla model the two distribution $\mathcal{N}_m\,(-0.42, 0.27)$, $\mathcal{N}_f\,(0.42, 0.47)$ are clearly separate, they are almost overlapping in the regularized model ($\mathcal{N}_m\,(-0.09, 0.93)$, $\mathcal{N}_f\,(-0.09, 0.91)$).

Table 2: **CelebA target accuracy** *(higher is better)* **and private class accuracy** *(lower is better)*.

Task	Method	Target	Private Class
Hair Color	Vanilla	70.25	59.20
	EnD [38]	**91.21**	**50.00**
Makeup	Vanilla	62.00	80.56
	EnD [38]	**75.93**	**63.89**

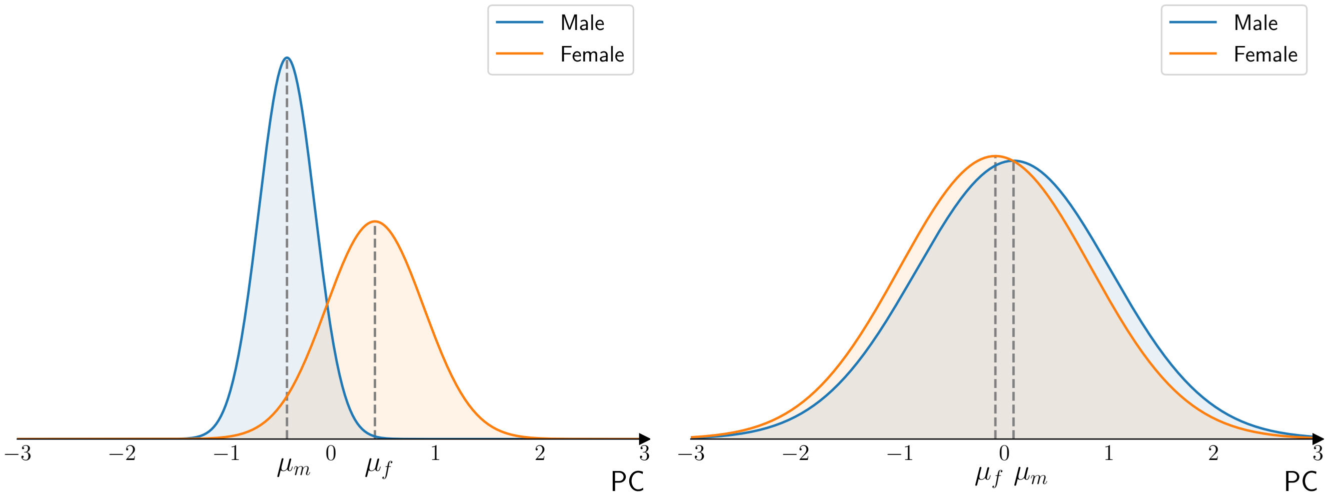

(a)

(b)

Figure 7: **IMDB train splits:** EB1 (a) and EB2 (b).

Table 3: **IMDB target accuracy** *(higher is better)* **and private class accuracy** *(lower is better)*. On age detection, gender is guessed correctly 50% of the times, which is equal to random guessing.

Split	Method	Target	Private Class
EB1	Vanilla	77.17	82.36
	EnD [38]	**80.15**	**49.95**
EB2	Vanilla	61.97	63.74
	EnD [38]	**78.80**	**50.05**



(a)

(b)

Figure 8: **Gaussian fit on the principal component** (PC) of the IMDB embeddings using a vanilla model (a) and a *EnD*-regularized model (b).

Table 4: **SIIM-FISABIO-RSNA target accuracy** *(higher is better)* **and private class accuracy** *(lower is better)*.

Method	Target	Private Class
Vanilla	78.12	**87.1**
EnD (low) [38]	**78.21**	63.4
EnD (high) [38]	78.02	55.3

4.3. Gender leakage in medical data

We also test the capability of removing information considered private on a medical dataset. SIIM-FISABIO-RSNA is a dataset[2] comprising more than 6k chest X-ray (CXR) scans in DICOM format, anonymized according to the current GDPR guidelines. For study purposes, however, the metadata associated to these scans comprises information about the gender, which will be used as private class. The scans are converted using the meta-information contained in the DICOM files, and rescaled to 448 × 448 resolution.

Results Results are provided in Tab. 4. In this case, for EnD, we provide two different results: one is achieved with a small weight for the regularization (specifically, it weights over the 1% on the total objective function minimized - low) while another has a higher weight (10% - high). Also in this case we observe that from a vanilla approach we are able to recover the information about the gender with a good accuracy (above 87%) while the effect of EnD drops as the weight of the regularization term increases. Differently from the previous scenarios, the performance in this case is not significantly affected: this is explained from the natural disentanglement between gender and the given medical task (presence of pneumonia and typical COVID presence). However, the gender information is still naturally forwarded to the bottleneck layer, which is postulated as plausible by some works in the literature [4, 34].

5. Conclusion

In this work we have shed some light over the possibility of bridging debiasing and privacy-preserving approaches for deep learning. In particular, we have reviewed some salient privacy preserving approaches and categorizing the most popular debiasing approaches, evidencing resemblances naturally rising between the two worlds. To address our investigation, we have considered the special case in which debiasing algorithms consider the private information as the bias for the learning problem. We have conducted some empirical evaluations from which we evidenced that, under our constraint, there *exists* a non-empty class of debiasing algorithms which can be deployed for both purposes. In particular, if the given debiasing algorithm is also able to hide the private information rather than simply re-weighting it, then it can be successfully deployed for privacy preservation. The investigation on whether the sufficient condition also hold is left as future work.

[2] https://www.kaggle.com/c/siim-covid19-detection/data

References

[1] Regulation (eu) 2016/679 of the european parliament and of the council of 27 april 2016 on the protection of natural persons with regard to the processing of personal data and on the free movement of such data, and repealing directive 95/46/ec (general data protection regulation). *OJ*, L 119:1–88, 4.5.2016. 1, 2

[2] Martin Abadi, Andy Chu, Ian Goodfellow, H Brendan McMahan, Ilya Mironov, Kunal Talwar, and Li Zhang. Deep learning with differential privacy. In *Proceedings of the 2016 ACM SIGSAC conference on computer and communications security*, pages 308–318, 2016. 3

[3] Mohsan Alvi, Andrew Zisserman, and Christoffer Nellåker. Turning a blind eye: Explicit removal of biases and variation from deep neural network embeddings. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 0–0, 2018. 5

[4] Devansh Arpit, Stanisław Jastrzundefinedbski, Nicolas Ballas, David Krueger, Emmanuel Bengio, Maxinder S. Kanwal, Tegan Maharaj, Asja Fischer, Aaron Courville, Yoshua Bengio, and Simon Lacoste-Julien. A closer look at memorization in deep networks. In *Proceedings of the 34th International Conference on Machine Learning - Volume 70*, ICML'17, page 233–242. JMLR.org, 2017. 8

[5] Joshua Attenberg, Panos Ipeirotis, and Foster Provost. Beat the machine: Challenging humans to find a predictive model's "unknown unknowns". *Journal of Data and Information Quality (JDIQ)*, 6(1):1–17, 2015. 3

[6] Hyojin Bahng, Sanghyuk Chun, Sangdoo Yun, Jaegul Choo, and Seong Joon Oh. Learning de-biased representations with biased representations. In *International Conference on Machine Learning (ICML)*, 2020. 3, 4, 5, 6

[7] Arnaud Berlioz, Arik Friedman, Mohamed Ali Kaafar, Roksana Boreli, and Shlomo Berkovsky. Applying differential privacy to matrix factorization. In *Proceedings of the 9th ACM Conference on Recommender Systems*, pages 107–114, 2015. 4

[8] Remi Cadene, Corentin Dancette, Matthieu Cord, Devi Parikh, et al. Rubi: Reducing unimodal biases for visual question answering. In *Advances in neural information processing systems*, pages 841–852, 2019. 3, 4

[9] Christopher Clark, Mark Yatskar, and Luke Zettlemoyer. Don't take the easy way out: Ensemble based methods for avoiding known dataset biases. In Kentaro Inui, Jing Jiang, Vincent Ng, and Xiaojun Wan, editors, *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing, EMNLP-IJCNLP 2019, Hong Kong, China, November 3-7, 2019*, pages 4067–4080. Association for Computational Linguistics, 2019. 3, 4

[10] Ekin D. Cubuk, Barret Zoph, Dandelion Mane, Vijay Vasudevan, and Quoc V. Le. Autoaugment: Learning augmentation strategies from data. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019. 3

[11] Cynthia Dwork and Jing Lei. Differential privacy and robust statistics. In *Proceedings of the forty-first annual ACM symposium on Theory of computing*, pages 371–380, 2009. 1, 3

[12] Úlfar Erlingsson, Vasyl Pihur, and Aleksandra Korolova. Rappor: Randomized aggregatable privacy-preserving ordinal response. In *Proceedings of the 2014 ACM SIGSAC conference on computer and communications security*, pages 1054–1067, 2014. 3

[13] Ivan P Fellegi. On the question of statistical confidentiality. *Journal of the American Statistical Association*, 67(337):7–18, 1972. 2

[14] Matt Fredrikson, Somesh Jha, and Thomas Ristenpart. Model inversion attacks that exploit confidence information and basic countermeasures. In *Proceedings of the 22nd ACM SIGSAC conference on computer and communications security*, pages 1322–1333, 2015. 5

[15] Craig Gentry. Fully homomorphic encryption using ideal lattices. In *Proceedings of the forty-first annual ACM symposium on Theory of computing*, pages 169–178, 2009. 1, 2

[16] 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. 5

[17] Hsiang Hsu, Shahab Asoodeh, and Flavio P Calmon. Information-theoretic privacy watchdogs. In *2019 IEEE International Symposium on Information Theory (ISIT)*, pages 552–556. IEEE, 2019. 1

[18] Gao Huang, Zhuang Liu, Laurens Van Der Maaten, and Kilian Q Weinberger. Densely connected convolutional networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4700–4708, 2017. 5

[19] Roger Iyengar, Joseph P Near, Dawn Song, Om Thakkar, Abhradeep Thakurta, and Lun Wang. Towards practical differentially private convex optimization. In *2019 IEEE Symposium on Security and Privacy (SP)*, pages 299–316. IEEE, 2019. 1, 3

[20] Bargav Jayaraman and Lingxiao Wang. Distributed learning without distress: Privacy-preserving empirical risk minimization. *Advances in Neural Information Processing Systems*, 2018. 4

[21] Peter Kairouz, Sewoong Oh, and Pramod Viswanath. Extremal mechanisms for local differential privacy. *Advances in neural information processing systems*, 27:2879–2887, 2014. 3

[22] Renuga Kanagavelu, Zengxiang Li, Juniarto Samsudin, Yechao Yang, Feng Yang, Rick Siow Mong Goh, Mervyn Cheah, Praewpiraya Wiwatphonthana, Khajonpong Akkarajitsakul, and Shangguang Wang. Two-phase multi-party computation enabled privacy-preserving federated learning. In *2020 20th IEEE/ACM International Symposium on Cluster, Cloud and Internet Computing (CCGRID)*, pages 410–419. IEEE, 2020. 2

[23] Byungju Kim, Hyunwoo Kim, Kyungsu Kim, Sungjin Kim, and Junmo Kim. Learning not to learn: Training deep neural networks with biased data. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019. 5, 7

[24] Yann LeCun, Corinna Cortes, and CJ Burges. Mnist handwritten digit database. *ATT Labs [Online]. Available: http://yann.lecun.com/exdb/mnist*, 2, 2010. 6

[25] Kristen LeFevre, David J DeWitt, and Raghu Ramakrishnan. Mondrian multidimensional k-anonymity. In *22nd International conference on data engineering (ICDE'06)*, pages 25–25. IEEE, 2006. 1

[26] Yehida Lindell. Secure multiparty computation for privacy preserving data mining. In *Encyclopedia of Data Warehousing and Mining*, pages 1005–1009. IGI global, 2005. 2

[27] Ziwei Liu, Ping Luo, Xiaogang Wang, and Xiaoou Tang. Deep learning face attributes in the wild. In *Proceedings of International Conference on Computer Vision (ICCV)*, December 2015. 6

[28] Eleftheria Makri, Dragos Rotaru, Nigel P Smart, and Frederik Vercauteren. Epic: efficient private image classification (or: Learning from the masters). In *Cryptographers' Track at the RSA Conference*, pages 473–492. Springer, 2019. 2

[29] Arvind Narayanan and Vitaly Shmatikov. Robust de-anonymization of large sparse datasets. In *2008 IEEE Symposium on Security and Privacy (sp 2008)*, pages 111–125. IEEE, 2008. 1, 2

[30] NhatHai Phan, Yue Wang, Xintao Wu, and Dejing Dou. Differential privacy preservation for deep auto-encoders: an application of human behavior prediction. In *Thirtieth AAAI Conference on Artificial Intelligence*, 2016. 1, 3

[31] NhatHai Phan, Xintao Wu, and Dejing Dou. Preserving differential privacy in convolutional deep belief networks. *Machine learning*, 106(9):1681–1704, 2017. 3

[32] M Sadegh Riazi, Mohammad Samragh, Hao Chen, Kim Laine, Kristin Lauter, and Farinaz Koushanfar. {XONN}: Xnor-based oblivious deep neural network inference. In *28th {USENIX} Security Symposium ({USENIX} Security 19)*, pages 1501–1518, 2019. 1, 2

[33] Rasmus Rothe, Radu Timofte, and Luc Van Gool. Deep expectation of real and apparent age from a single image without facial landmarks. *International Journal of Computer Vision*, 126(2-4):144–157, 2018. 6

[34] Vitaly Shmatikov and Congzheng Song. What are machine learning models hiding? 8

[35] Reza Shokri and Vitaly Shmatikov. Privacy-preserving deep learning. In *Proceedings of the 22nd ACM SIGSAC conference on computer and communications security*, pages 1310–1321, 2015. 1, 3

[36] Sho Sonoda and Noboru Murata. Neural network with unbounded activation functions is universal approximator. *Applied and Computational Harmonic Analysis*, 43(2):233–268, 2017. 1

[37] Latanya Sweeney. k-anonymity: A model for protecting privacy. *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, 10(05):557–570, 2002. 1, 2

[38] Enzo Tartaglione, Carlo Alberto Barbano, and Marco Grangetto. End: Entangling and disentangling deep representations for bias correction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13508–13517, 2021. 4, 5, 7, 8

[39] Tatiana Tommasi, Novi Patricia, Barbara Caputo, and Tinne Tuytelaars. A deeper look at dataset bias. In *Domain adaptation in computer vision applications*, pages 37–55. Springer, 2017. 3

[40] Antonio Torralba, Alexei A Efros, et al. Unbiased look at dataset bias. In *CVPR*, page 7. Citeseer, 2011. 3

[41] Stanley L Warner. Randomized response: A survey technique for eliminating evasive answer bias. *Journal of the American Statistical Association*, 60(309):63–69, 1965. 2

[42] Xi Wu, Matthew Fredrikson, Somesh Jha, and Jeffrey F Naughton. A methodology for formalizing model-inversion attacks. In *2016 IEEE 29th Computer Security Foundations Symposium (CSF)*, pages 355–370. IEEE, 2016. 4

[43] Liyang Xie, Kaixiang Lin, Shu Wang, Fei Wang, and Jiayu Zhou. Differentially private generative adversarial network. *arXiv preprint arXiv:1802.06739*, 2018. 3

[44] Lei Yu, Ling Liu, Calton Pu, Mehmet Emre Gursoy, and Stacey Truex. Differentially private model publishing for deep learning. In *2019 IEEE Symposium on Security and Privacy (SP)*, pages 332–349. IEEE, 2019. 1, 3