DIFFERENTIALLY PRIVATE TRAINING OF RESIDUAL NETWORKS WITH SCALE NORMALISATION

Helena Klause\textsuperscript{1}, Alexander Ziller\textsuperscript{1,2}, Daniel Rueckert\textsuperscript{1,3}, Kerstin Hammernik\textsuperscript{1,3}, and Georgios Kaissis\textsuperscript{1,2,3}

\{helena.klause,alex.ziller,daniel.rueckert,k.hammernik,g.kaissis\} @tum.de

\textsuperscript{1}Institute for Artificial Intelligence and Informatics in Medicine, Department of Informatics, Technical University of Munich
\textsuperscript{2}Institute of Diagnostic and Interventional Radiology, Faculty of Medicine, Technical University of Munich
\textsuperscript{3}Department of Computing, Faculty of Engineering, Imperial College London

\textbf{ABSTRACT}

We investigate the optimal choice of replacement layer for Batch Normalisation (BN) in residual networks (ResNets) for training with Differentially Private Stochastic Gradient Descent (DP-SGD) and study the phenomenon of \textit{scale mixing} in residual blocks, whereby the activations on the two branches are scaled differently. Our experimental evaluation indicates that a hyperparameter search over 1-64 Group Normalisation (GN) groups improves the accuracy of ResNet-9 and ResNet-50 considerably in both benchmark (CIFAR-10) and large-image (ImageNette) tasks. Moreover, \textit{scale normalisation}, a simple modification to the model architecture by which an additional normalisation layer is introduced after the residual block’s addition operation further improves the utility of ResNets allowing us to achieve state-of-the-art results on CIFAR-10.

\section{INTRODUCTION}

Residual networks (ResNets) \cite{he2016deep} represent a standard architecture in computer vision. Their core component, the residual block, has since become a standard component of architectures responsible several victories in the ImageNet (ILSVR) challenge \cite{deng2009imagenet}. Moreover, despite newer works introducing novel architectures which outperform the original ResNets, it has recently been shown that training adaptations alone are sufficient to render their performance comparable to current-generation vision models in challenging tasks \cite{wightman2021resnet}. Differentially Private Stochastic Gradient Descent (DP-SGD), introduced by Abadi et al. \cite{abadi2016deep}, is arguably the \textit{de facto} standard technique for training neural networks under Differential Privacy (DP) constraints. The combination of ResNets and DP-SGD thus appears as a promising direction for privately training computer vision models. Indeed, the first reported training of ResNets under DP on the ImageNet large-scale vision benchmark was very recently reported \cite{kurakin2022training}. However, the application of DP-SGD results in unavoidable, information-theoretic trade-offs between privacy guarantees and model accuracy. This so-called \textit{privacy-utility trade-off} is a principal deterrent from the large-scale application of DP-SGD and mitigating it is a challenging and important task.

In the current work, we investigate two phenomena which arise in the training of ResNets with DP-SGD: ResNets utilise Batch Normalisation (BN) layers \cite{ioffe2015batch}, which are incompatible with the notion of \textit{per-sample} gradients required for DP-SGD training. The question which layer represents an optimal replacement for BN has not yet been conclusively resolved. In addition, we describe the phenomenon of \textit{scale mixing} whereby activations along the two branches of a residual block are scaled differently, interfering with convergence. Our contributions can be summarised as follows:
1. We evaluate the choice of optimal replacement layer for BN in shallow (ResNet-9) and deep (ResNet-50) architectures trained with DP-SGD on small (CIFAR-10) and large (ImageNette) image tasks through extensive (hyper-)parameter searches.

2. We show that a simple architectural adaptation, which we term scale normalisation, can alleviate the aforementioned scale mixing phenomenon and thus lead to improved accuracy.

3. We provide a number of practitioner recommendations resulting from our findings.

2 PRIOR WORK

The attempt to alleviate privacy-utility trade-offs in DP-SGD has been the subject of two distinct lines of work. Due to space constraints, we focus exclusively on the works investigating the DP-SGD algorithm as presented in Abadi et al. (2016) and omit works relying on modifications to the DP-SGD algorithm or alternative techniques for obtaining improved accuracy (such as Zhu et al. (2020) or Papernot et al. (2018)). The first approach to improving the training outcome of DP-SGD models is transfer learning, that is, fine-tuning a network which has been pre-trained on public data (Lu et al., 2021; Davody et al., 2021; Tramer & Boneh, 2021). Such works are able to utilise BN, as the layer’s parameters remain unaltered (frozen). Our work studies the situation in which transfer learning is not possible and networks have to be trained from scratch. Here, a second line of prior work focusing on model architecture or training adaptations is applicable. The topic of adapting the architecture from the ground up has been studied by Morsbach et al. (2021) and Papernot et al. (2020), who conclude that it is likely that specialised architectures for use with DP will be required. This approach is orthogonal to ours, which focuses on the question of how to best adapt existing architectures to DP-SGD. Similar adaptations of existing architectures or the training process are studied by Papernot et al. (2021), who find that the suitable choice of activation function improves model accuracy and by Dörmann et al. (2021), who demonstrate that large batch training leads to improved training accuracy. To the best of our knowledge, the work by Dörmann et al. (2021) also demonstrated the highest validation accuracy on CIFAR-10 so far, and thus represents the benchmark for our experiments. The popular suggestion (see e.g. documentation to Opacus (Yousefpour et al., 2021)) to replace BN by Group Normalisation (GN) layers with 32 groups is folklore and has –to our knowledge– not yet been systematically evaluated in prior work.

3 METHODS

3.1 INTRODUCTORY REMARKS

Our work assumes familiarity with details regarding DP and the DP-SGD algorithm, which can be found in Abadi et al. (2016). For our experimental evaluation below, we utilised two ResNet architectures. We constructed a shallow 9-layer ResNet (ResNet-9, 2,447,946 parameters) with Mish activation functions (Misra, 2020), and used the torchvision version of ResNet-50 (25,557,032 parameters) as a deep architecture. We used the CIFAR-10 small image dataset (Krizhevsky et al., 2009) and the 2019 version of the ImageNette dataset (Howard, 2019), a subset of 10 ImageNet database classes at a size of 160×160 pixels. Unless otherwise noted, all experiments below were performed in triplicate with three fixed random seeds selected randomly in advance. All architectures were trained for 50 epochs with an $L_2$-norm bound (clipping threshold) of 1.5 (derived by preliminary experimentation), to a maximum $\varepsilon$-value of <7.5 (as reported in Papernot et al. (2021)). Batch sizes for all searches ranged from 128 to 7,000. We employed the NAdam optimiser (Dozat, 2016), which extensive preliminary experiments showed to be very robust at the default learning rate of 0.001. We employed a constant learning rate schedule with reduction by half upon stagnation of the validation loss for more than three epochs. All (hyper-)parameter searches were conducted using the Optuna package (Akiba et al., 2019) using the Tree-structured Parzen Estimator search strategy and the threshold pruner. We reset the DP budget for each hyperparameter search run similar to Kurakin et al. (2022).

3.2 CHOICE OF REPLACEMENT NORMALISATION LAYER

Our first objective was the evaluation of potential replacement layers for BN, which, as described above, is incompatible with DP-SGD, as it renders activations of a batch dependent on each-other
and thus makes the notion of per-sample gradients ill-defined. Hence, we considered alternative normalisation layers which return outputs that are dependent only on individual samples, namely Instance Normalisation (IN) \citep{ulyanov2016instance}, Layer Normalisation (LN) \citep{ba2016layer} or GN \citep{wu2018group}. Of note, GN is a generalisation of Instance Normalisation (IN) and Layer Normalisation (LN), where IN is equivalent to GN when a group size of 1 chosen and to LN when a group size matching the number of features is selected. Hence, we evaluated the optimal blocks (early, mid, late, final) to obtain 4 neural networks, discussed e.g. in \citep{glorot2010understanding, zhang2019residual}. The effect of scale into the next layer. This desirable effect has been shown to yield benefits for the training dynamics of neural networks, discussed e.g. in \citep{glorot2010understanding, zhang2019residual}. Empirically, this modification results in a marked symmetrisation of the activations flowing from inputs to the operation of the residual block \(V\). For each layer \(i\), where \(i = (n, \ldots, 1)\), where \(\otimes\) denotes repeated composition. We note that \(V\) in this context can be any of the normalisation layers described above. \(V_A\) executes the function \(V_A := v_A(x_1, x_2) = x_1 + x_2\) on the inputs \((v_R(x), v_F(x))\) of the node \(A\). The output node \(O\) finally receives \(v_A(v_R(x), v_F(x))\). The entire action of the residual block \(F\) expressed as a single function from some input tensor \(x\) to some output tensor \(y\) and denoting the “contents” of each node with square brackets, can thus be represented as follows:

\[
F : x \mapsto y := I[x] \xrightarrow{V_A(Add)} O[y]. \tag{1}
\]

Diagram \([1]\) demonstrates the origin of the scale mixing phenomenon. The activations forming the inputs to \(A\) potentially have drastically different scaling, that is, value ranges and distributions. The \(V_F\) branch retains the scaling of \(I\), while \(V_R\) normalises the intermediate activations \(i\) times. Scale normalisation (ScaleNorm) alleviates this issue by introducing an additional normalisation layer \(S\) to the operation of \(V_A\), which now becomes \(V_A^S := v_A(x_1, x_2) = S(x_1 + x_2)\). We term ResNets with scale normalisation ScaleResNets in the rest of the paper. The residual block of ScaleResNets \(F^S\) can be represented as:

\[
F^S : x \mapsto y := I[x] \xrightarrow{V_A^S(ScaleNorm)} O[y]. \tag{2}
\]

Empirically, this modification results in a marked symmetrisation of the activations flowing from \(O\) into the next layer. This desirable effect has been shown to yield benefits for the training dynamics of neural networks, discussed e.g. in \citep{glorot2010understanding, zhang2019residual}. The effect of scale normalisation is demonstrated in Figure \([1]\).
Figure 1: Activation histograms of a ResNet (top row) vs. a ScaleResNet (bottom row) residual block. Observe that the activations output by $V_A$ (top right) are markedly asymmetric with substantial mass on the positive side (sample average: 0.72, standard deviation: 0.76), whereas the activations output by the scale normalisation operation $V_S^N$ (bottom right) are more symmetric about the sample mean of 0 and have unity standard deviation.

4 RESULTS

In this section, we present experimental results regarding the optimal choice of replacement layer for BN and the effect of ScaleNorm. All findings of this section are summarised in Table 1.

4.1 OPTIMAL REPLACEMENT FOR BATCH NORMALISATION

Our experiments on both the CIFAR-10 and the ImageNet dataset revealed that, while in many cases GN outperformed LN and IN, LN outperformed GN by nearly 2% in ResNet-50 on ImageNet. IN consistently performed worst and –especially for ImageNet and ResNet-50– a larger number of groups led to diminished model performance. Of note, searching over variable numbers of groups for each layer only led to marginal performance improvements of $<1\%$ in all cases (e.g. 0.8% for ResNet-9 on CIFAR-10). These findings lead to the following recommendations:

1. The (folklore) recommendation that GN with 32 groups is—in general—a good choice for training ResNets with DP-SGD should be viewed with caution, as substantial performance gains can be achieved by a simple hyperparameter search for the best number of groups including LN (i.e. 1 group).

2. Improvements from searching over a variable number of groups for each layer are existent, but marginal. With an eye towards time/resource efficiency, we thus do not recommend this strategy over a fixed number of groups for the entire network.

3. Large networks seem to favour a smaller number of groups. This result can be interpreted as an effect of the large amount of noise resulting from the total noise magnitude in DP-SGD being proportional to the square root of the number of network parameters. Here, the effect of the small number of groups seems to be a “dampening” of the noise by averaging over all feature maps.

4.2 EFFECTS OF SCALE NORMALISATION

Our findings also indicate that ScaleNorm is a beneficial architectural modification for DP-SGD training. In particular, ScaleResNets outperformed regular ResNets in both tasks and irrespective of network size. The effect was especially pronounced with the ImageNette dataset and the ResNet-50 architecture, indicating that in settings beyond benchmark tasks such as CIFAR-10 and closer to real-life scenarios, the stabilising effect of ScaleNorm on activations seems to offer added “protection”
against the larger noise magnitudes. We refer to [Papernot et al. (2021)] for a similar discussion, albeit not in conjunction with ResNets. Of note, ScaleNorm did not benefit non-DP training (where BN can be used) or IN, which was the only normalisation layer to perform better without ScaleNorm. We conjecture that this phenomenon is due to the fact that the BN layer can “learn” to counterbalance the effect of scale mixing and the lack of added noise in non-DP training and the poor general performance of IN, respectively. This also justifies why ScaleNorm has not received significant appraisal in mainstream machine learning literature. Our findings, combined with the simplicity of the architectural modification ScaleNorm represents (it can be realised in a single line of code in most deep learning libraries), lead us to recommend ScaleResNets as the standard architecture for DP-SGD training.

Table 1: Model accuracy in % on the CIFAR-10 and ImageNette datasets using the ResNet-9 and ResNet-50 model architectures with varying numbers of GN groups as well as IN and LN. Experiments were conducted either with (✓) or without (✗) ScaleNorm. All results are reported as mean and median values of three runs.

| Dataset   | Model     | ScaleNorm | LN | 8  | 16 | 32 | 64 | IN |
|-----------|-----------|-----------|----|----|----|----|----|----|
| CIFAR-10  | ResNet-9  | ✗         |    |    |    |    |    |    |
|           |           | mean      | 69.8| 69.8| 71.2| 71.3| 70.6| 66.7|
|           |           | median    | 70.5| 71.3| 71.3| 71.7| 71.1| 65.6|
| ImageNette| ResNet-9  | ✓         |    |    |    |    |    |    |
|           |           | mean      | 61.7| 62.6| 63.7| 62.8| 60.6| 54.7|
|           |           | median    | 63.2| 64.6| 64.6| 64.1| 62.0| 48.2|
| ImageNette| ResNet-50 | ✗         |    |    |    |    |    |    |
|           |           | mean      | 43.2| 42.2| 40.7| 36.4| 28.7| 15.0|
|           |           | median    | 44.2| 42.1| 37.5| 30.7| 21.0| 13.6|

5 DISCUSSION AND CONCLUSION

The broad implementation of DP to large-scale computer vision tasks will require tackling the privacy-utility trade-offs inherent to DP-SGD. Our work presents two architectural modifications, the choice of optimal normalisation layer and ScaleNorm which, despite their apparent simplicity, substantially improve the accuracy of ResNets trained with DP. Our result on CIFAR-10—to the best of our knowledge—represents a new state-of-the-art result. We report this fact not because we consider it of independent interest, but because it demonstrates that small details, such as a search over a relatively conservative range of GN groups (1-64) and a simple architectural modification were sufficient to obtain it, despite a relatively small batch size of 1024. The previous state-of-the-art result by [Dörmann et al. (2021)] required substantially larger batch sizes and gradient accumulation, which is non-trivial in DP-SGD due to sampling peculiarities, and not equally supported in all DP-SGD frameworks. Our results also corroborate the assumption that parameter efficiency beats size in DP-SGD, with ResNet-9 (an architecture primarily designed for CIFAR-10) representing an excellent choice even for the larger images of the ImageNette dataset. We consider this fact—which contradicts the trend of ever-increasing network sizes and, by extension, CO₂-emissions for their training—a satisfying conclusion from an environmental protection and resource efficiency perspective. In future work, we intend to supplement the here-presented findings with data on other residual architectures including UNets, as well as the related densely connected networks (DenseNets). Moreover, we intend to elucidate the interplay of ScaleNorm with other architecture components such as activation functions and adaptations such as initialisation as well as theoretically investigate its effect on training dynamics. In conclusion, we are hopeful that our recommendations will benefit the ongoing research into DP-optimised architectures and stimulate the application of private computer vision models in a broad variety of tasks.
REFERENCES

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, pp. 308–318, 2016.

Takuya Akiba, Shotaro Sano, Toshihiko Yanase, Takeru Ohta, and Masanori Koyama. Optuna: A next-generation hyperparameter optimization framework. In Proceedings of the 25rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 2019.

Jimmy Lei Ba, Jamie Ryan Kiros, and Geoffrey E Hinton. Layer normalization. arXiv preprint 1607.06450, 2016.

Ali Davody, David Ifeoluwa Adelani, Thomas Kleinbauer, and Dietrich Klakow. On the effect of normalization layers on differentially private training of deep neural networks. arXiv preprint 2006.10919, 2021.

Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In 2009 IEEE conference on computer vision and pattern recognition, pp. 248–255. Ieee, 2009.

Friedrich Dörmann, Osvald Frisk, Lars Nørvang Andersen, and Christian Fischer Pedersen. Not all noise is accounted equally: How differentially private learning benefits from large sampling rates. In 2021 IEEE 31st International Workshop on Machine Learning for Signal Processing (MLSP), pp. 1–6, 2021.

Timothy Dozat. Incorporating Nesterov Momentum into Adam. ICLR Workshop Track, 2016.

Xavier Glorot and Yoshua Bengio. Understanding the difficulty of training deep feedforward neural networks. In Proceedings of the thirteenth international conference on artificial intelligence and statistics, pp. 249–256. JMLR Workshop and Conference Proceedings, 2010.

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, pp. 770–778, 2016.

Jeremy Howard. Imagenette, 2019. URL https://github.com/fastai/imagenette

Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In International conference on machine learning, pp. 448–456. PMLR, 2015.

Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. 2009.

Alexey Kurakin, Shuang Song, Steve Chien, Roxana Geambasu, Andreas Terzis, and Abhradeep Thakurta. Toward training at imagenet scale with differential privacy, 2022.

Zelun Luo, Daniel J. Wu, Ehsan Adeli, and Li Fei-Fei. Scalable differential privacy with sparse network finetuning. In 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 5057–5066, 2021.

Diganta Misra. Mish: A self regularized non-monotonic activation function, 2020.

Felix Morsbach, Tobias Dehling, and Ali Sunyaev. Architecture matters: Investigating the influence of differential privacy on neural network design. CoRR, abs/2111.14924, 2021.

Nicolas Papernot, Shuang Song, Ilya Mironov, Ananth Raghunathan, Kunal Talwar, and Úlfar Erlingsson. Scalable private learning with PATE. arXiv preprint arXiv:1802.08908, 2018.

Nicolas Papernot, Steve Chien, Shuang Song, Abhradeep Thakurta, and Úlfar Erlingsson. Making the shoe fit: Architectures, initializations, and tuning for learning with privacy, 2020. URL https://openreview.net/forum?id=rJg851rYwH
Nicolas Papernot, Abhradeep Thakurta, Shuang Song, Steve Chien, and Úlfar Erlingsson.Tempered sigmoid activations for deep learning with differential privacy. *Proceedings of the AAAI Conference on Artificial Intelligence*, 35(10):9312–9321, May 2021.

Florian Tramèr and Dan Boneh. Differentially private learning needs better features (or much more data). In *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7*, 2021.

Dmitry Ulyanov, Andrea Vedaldi, and Victor Lempitsky. Instance normalization: The missing ingredient for fast stylization. *arXiv preprint arXiv:1607.08022*, 2016.

Ross Wightman, Hugo Touvron, and Hervé Jégou. Resnet strikes back: An improved training procedure in TIMM, 2021.

Yuxin Wu and Kaiming He. Group normalization. In *Proceedings of the European conference on computer vision (ECCV)*, pp. 3–19, 2018.

Ashkan Yousefpour, Igor Shilov, Alexandre Sablayrolles, Davide Testuggine, Karthik Prasad, Mani Malek, John Nguyen, Sayan Ghosh, Akash Bharadwaj, Jessica Zhao, Graham Cormode, and Ilya Mironov. Opacus: User-friendly differential privacy library in PyTorch. *arXiv preprint arXiv:2109.12298*, 2021.

Hongyi Zhang, Yann N. Dauphin, and Tengyu Ma. Residual learning without normalization via better initialization. In *International Conference on Learning Representations*, 2019.

Yuqing Zhu, Xiang Yu, Manmohan Chandraker, and Yu-Xiang Wang. Private-KNN: Practical differential privacy for computer vision. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 11854–11862, 2020.