Training Large ASR Encoders with Differential Privacy

Anonymous

Abstract

Self-supervised learning (SSL) methods for large speech models have proven to be highly effective at ASR. With the interest in public deployment of large pre-trained models, there is a rising concern for unintended memorization and leakage of sensitive data points from the training data. In this paper, we apply differentially private (DP) pre-training to a SOTA Conformer-based encoder, and study its performance on a downstream ASR task assuming the fine-tuning data is public. This paper is the first to apply DP to SSL for ASR, investigating the DP noise tolerance of the BEST-RQ pre-training method. Notably, we introduce a novel variant of model pruning called *gradient-based layer freezing* that provides strong improvements in privacy-utility-compute trade-offs. Our approach yields a LibriSpeech test-clean/other WER (%) of 3.78/ 8.41 with (10, 1e-9)-DP for extrapolation towards low dataset scales, and 2.81/ 5.89 with (10, 7.9e-11)-DP for extrapolation towards high scales.

Index Terms: automatic speech recognition, differential privacy, gradient clipping, model pruning

1. Introduction

Across various sub-fields of Machine Learning (ML), large scale transformer-based models [1, 2] have seen widespread adoption for modeling long-range dependencies in sequences. In automatic speech recognition (ASR), the known success of convolutions [3] prompted the introduction of the Conformer architecture [4], and later incorporation of BERT-style self-supervised learning (SSL) via the BEST-RQ pre-training method [5]. Popular ASR models are often released as modifiable checkpoints after being pre-trained on thousands of hours of crawled user-spoken utterances. Following this paradigm of pre-training ASR encoders on massive amount of data can put the model at risk for leaking sensitive information, especially when the data consist of web crawls that can contain sensitive information such as gender, dialect or identity of a speaker.

It is well-known that ML models can leak sensitive information about their training dataset, even when the data is kept private. This has been extensively discussed by works such as [6, 7, 8, 9] in natural language processing (NLP) and computer vision (CV) and later extended to the speech domain by [10, 11, 12, 13]. This paper explores methods focused on differentially private (DP) pre-training of ASR encoders (illustration in Figure 1) for mitigating the privacy leakage from trained encoders.

Differential Privacy [15] is a way to combat the privacy leakage issue, by providing theoretical guarantees about the limits of influence of any individual training point towards the final model, thereby preventing an attacker from confidently inferring whether any particular sample was used for training. One challenge associated with DP training is the stringent privacy-utility-compute trade-offs (shortened as trade-offs throughout this paper) for training large models. A straightforward technique to increase privacy is adding more noise, but that negatively affects the model performance or utility. One method to mitigate this is to increase batch size [16, 17], but this comes at the cost of increasing compute which can be expensive. Recently, works in language modeling and vision have demonstrated the utility of their DP methods being close to their non-private baselines [18, 19]. Most works are focused on improving trade-offs for the fine-tuning regime, with positive effects seen for parameter efficient techniques such as LoRA [20] that mitigate the issue of growing magnitude of DP noise as the model size increases [21, 18].

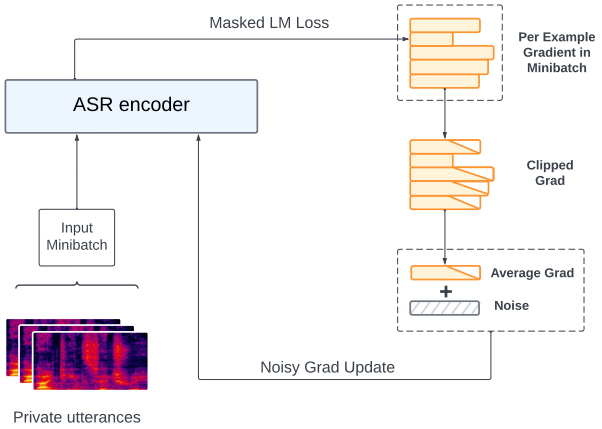

Figure 1: *The Differentially Private pre-training method for ASR encoder involving clipping per-example gradients from the minibatch, and addition of calibrated gaussian noise. Gradients with norms below clip value are not clipped, as shown above. Once private pre-training of the ASR encoder is done, fine-tuning is done publicly after attaching an ASR decoder and using CTC loss [4, 14]*

This paper is focused on DP during the pre-training regime (see Figure 1), where the challenges associated with adding larger DP noise as a result of large model sizes for full model training remain. Recently, a work in language modeling has successfully reduced the gap between private DP pre-training and the non-private baseline through the use of private tokenization and higher compute [22]. More recently, [17] discuss improving trade-offs in the setting of DP Federated Learning (FL) in ASR by using per-layer clipping [23].

So far, no work has evaluated training with DP in the SSL setting for ASR. The contribution of our work is as follows: 1) We are the first to evaluate the DP noise tolerance for the BEST-RQ setting of a large Conformer model, and 2) We introduce a new variant of model pruning called gradient-based layer freezing where we determine the model layers to freeze from a square of gradient analysis. Collectively, our recommended approach yields great improvements in utility (Word Error Rate i.e. WER in our case), while guaranteeing strong privacy of $\epsilon = 10$.

2. Background and Related Work

2.1. Differentially Private Training

Differential Privacy (DP) [15] is widely considered a gold standard for bounding and quantifying the privacy leakage of sensitive data when performing learning tasks. Intuitively, DP prevents an adversary from confidently making any conclusions about whether any particular data was used in training a model, even while having access to the model and arbitrary external side information. The formal definition of DP depends on the notion of neighboring datasets: we will refer to a pair of datasets $D, D' \in \mathcal{D}$ as neighbors if D' can be obtained from D by adding or removing one data sample.

Definition 1 ((ϵ, δ)-DP). *A (randomized) algorithm $\mathcal{A} : \mathcal{D} \rightarrow \Theta$ is (ϵ, δ)-differentially private if for all pairs of neighboring datasets $D, D' \in \mathcal{D}$, and for any $S \subseteq \Theta$ we have,*

$$P[\mathcal{A}(D) \in S] \leq exp(\epsilon) \cdot P[\mathcal{A}(D') \in S] + \delta. \quad (1)$$

Typically, differentially private training is performed using variants of DP-SGD [24], where the main distinctions from non-private training are the clipping of per-example gradients, and the addition of spherical Gaussian noise, as illustrated (for ASR pre-training) by Figure 1. This is implemented as a modification to the gradient computation during the optimization step by computing per-example gradients [25], clipping to limit their per-sample sensitivity, and the addition of calibrated Gaussian noise. Therefore, DP training is relatively independent of the exact choice of optimizer. For our experiments, we rely on the Adam optimizer with DP modifications for example-level DP. Training with DP incurs several challenges as a result of clipping and addition of noise, commonly characterized as privacy-utility-compute trade-offs (truncated as trade-offs in this paper).

2.2. Related Work

Many works [21, 16, 26] have shown that the trade-offs are substantial for training large neural networks with state-of-the-art techniques like DP-SGD [21, 24]. Consequently, there has been work [24, 27, 28, 29] on pre-training using public data for improving the utility of DP-SGD. A recent work [17] has considered DP training for ASR models, but focusing on the Federated Learning (FL) regime. Additionally, many works [30, 18, 19] have focused on privately fine-tuning neural networks (focusing largely on vision and language models (LMs)) after pre-training using public data to improve the trade-offs for DP-SGD. While it is common in literature to treat pre-training data as public, modern large model pre-training can involve sensitive data that is susceptible to be memorized and potentially leaked. There is only one recent work [22] that studies DP pre-training for LMs, and demonstrates that such models can be fine-tuned to high accuracies on downstream tasks. Related to modifications on bounding sensitivity within a training step, [31] have considered the role of gradient clipping and suggest model pruning as a strategy to improve the trade-offs.

3. Experimental Setup

For our model, we choose the 300M variant [12] of the state-of-the art ASR model architecture, Conformer XL [32]. The encoder is pre-trained on LibriLight (LL) [33] for 1M steps using self-supervised learning via the BERT-based Speech pre-Training with Random-projection Quantizer (BEST-RQ) [5]. Fine-tuning is done for 60k steps post attaching an additional projection layer on the encoder, using the LibriSpeech (LS) [34] dataset. Hyperparameter details and model architecture follow the BEST-RQ paper [5], and official dataset splits were used for training, validation and hyperparameter tuning. Pre-training takes ~1 week on Dragonfish TPUs with 8x8 topology, fine-tuning takes $\leq$ 1 day and original batch size was set at 512.

In this paper, we apply DP to the pre-training stage of our model (with LL), and assume that the fine-tuning dataset (LS) for the downstream ASR task is public. Utility is reported as test-clean/other WER on the LS dataset. We use the updated moments accountant [24, 35] for calculating our privacy guarantees. We report experiments with different DP noise multipliers in the range [1e-4, 1e-2], since we find that noise multipliers beyond 1e-2 lead to divergence (more details in Section 4.1).

Since the trade-offs with large model training can be significant, we follow the extrapolation strategy similar to recent works [16, 17]. We extrapolate the (ϵ, δ)-DP assuming the training dynamic remains unchanged upon linearly scaling minibatch size and noise multiplier (to maintain the expected signal-to-noise ratio for the gradient update) along with scaling the dataset size (for improved privacy accounting). Thus, we experiment with adding different DP noise multipliers and map the value of ϵ using the moments accountant, while the batch size, noise multiplier, and the pre-training dataset size are scaled up by the same factor as seen in Figure 2. We can see the positive effects of the scale-up factor on ϵ, thus significantly improving the privacy guarantees. In table 1, we report the precise scale-up factors for noise multipliers we consider in this paper to achieve a DP of $\epsilon = 10$ at $\delta = n^{-1.1}$, where n is the scaled-up dataset size. Note that according to recent work [36], such a level of DP can be classified in the "Tier 2: Reasonable privacy guarantees".

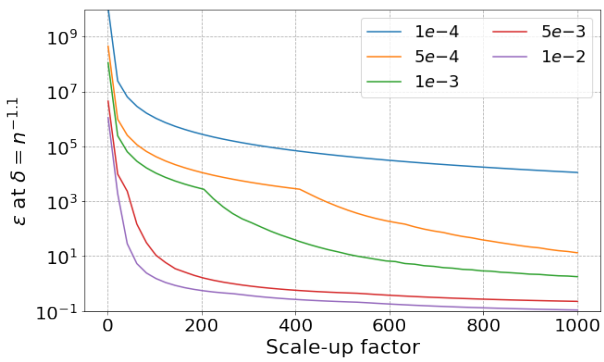

Figure 2: *Extrapolating the noise multiplier linearly with batch size and dataset size to maintain the signal-to-noise ratio and improve privacy accounting.*

Table 1: *Extrapolation factor for linearly scaling-up noise multiplier, batch size and dataset size needed for each used noise multiplier value to get DP $\epsilon = 10$ at $\delta = n^{-1.1}$, where n is the scaled-up dataset size.*

Noise multiplier	1e-4	5e-4	1e-3	5e-3	1e-2
Scale-up	5450	1070	530	105	52

4. Results

In Section 4.1, we introduce the baseline of (non-privately) pre-training the BEST-RQ 300M model. We detail preliminary modeling changes required to comply with DP training and include an analysis of the amount of DP noise tolerable for minimal performance regression of the model. Next, in Section 4.2 following [17], we incorporate per layer clipping for improved utility and noise tolerance. Lastly, we introduce our gradient-based layer freezing strategy. Our results denote a synergy between per-layer clipping and our model pruning technique, based on the compounding improvements we observe in model quality (summary of results in Table 3).

4.1. Noise tolerance of the BEST-RQ 300M model

We establish non-private baselines for the BEST-RQ 300M model, and analyze the degree of noise tolerated for minimal utility regression. As is typical for DP training, we replace batch normalization with group normalization to effectively limit per-sample contributions and avoiding mixing of batch statistics across samples [36]. After experimentation, we find the best setting of group normalization to have input rank of 3, number of groups as 1 and group norm epsilon as 1e-4, resulting in a test-clean/other WER (%) of 2.17/4.23 post fine-tuning on LibriSpeech.

Then, we experiment with choices for per-example clipping bounds, and find the bound 1.5 to be clipping almost all samples during training while providing minimal loss in performance, resulting in a WER of 2.21/4.29. We refer to this as the **non-private lower bound** result. Thus, the non-private baseline we report for BEST-RQ consists of group normalization and per-example clipping, to offer a direct comparison to the level of additive noise in our experiments. Our results for the non-private baseline, and for differing level of DP noise are reported in Table 2.

Table 2: *Noise tolerance of the BEST-RQ 300M model. Our no pre-train upper bound is WER of 4.43/11.23. Above noise multiplier* 1e-2*, the model diverges into WER of 100.*

Noise multiplier	**test-clean/other WER**
0	2.21/4.29
1e-4	2.24/4.51
5e-4	2.57/5.98
1e-3	3.54/8.31
5e-3	10.98/22.62
1e-2	15.38/29.62

Note that the performance of the model with fine-tuning from random initialization (no pre-training) is a WER of 4.43/11.23, which is the upper bound for effectively measuring the positive effects of pre-training. We refer to this result as the **no pre-train upper bound**, which is effectively the same as not applying BEST-RQ style pre-training to ConformerXL and just doing supervised training on the Librispeech dataset. As can be seen from Table 2, we start seeing significant regressions (greater than 10% relative) for noise multiplier 5e-4, where the standard extrapolation technique achieves DP $\epsilon = 10$ only at a practically prohibitive scale-up factor of 1070 (Table 1). For reference, extrapolation factor for DP $\epsilon = 10$ from noise multiplier 1e-2 is as low as 52, though with the current approach we get WER of 15.38/29.62 which is higher than the no pre-train upper bound. Our focus in the rest of the paper is to improve trade-offs for the settings with larger noise multipliers in the range [5e-4, 1e-2].

4.2. Improving the noise tolerance via warm-starting, per-layer clipping and gradient-based layer freezing

Recently, there have been increasing efforts towards improving trade-offs associated with DP training and we consider these efforts heavily in our experimentation. Ganesh et al. [37] show the importance of public pre-training for private model training, especially with an in-domain public checkpoint. Pelikan et al. [17] revive per-layer clipping and show improvements for DP in the supervised training setting of FL for ASR. A couple of recent works [38, 31] have shed light on the benefits of model pruning for DP training, by minimizing the negative effects of compounding noise affected by the model dimensionality. Thus, in order to bridge the utility gap with the non-private pre-trained baseline, we consider the following three improvements: warm-starting (WS) using public data, per-layer clipping, and our novel method of gradient-based layer freezing. Table 3 summarizes the compounding improvements on the three considered techniques.

Table 3: *Final noise tolerance WERs for BEST-RQ 300M model with our considered improvements. If we observe divergence (mainly for higher noise multipliers), we report results on fine-tuning with an early 200k step pre-trained checkpoint instead.*

Noise	**Public WS**	**+PerLayerClip**	**+LayerFreeze**
5e-4	3.82/7.65	2.78/5.9	**2.67/5.74**
1e-3	4.03/8.62	2.85/6.02	**2.81/5.89**
5e-3	6.34/13.88	3.78/8.09	**3.19/7.17**
1e-2	8.16/17.42	100/100	**3.78/8.41**

4.2.1. Warm-starting using in-domain public data (Public WS)

For our experiments, in line with prior works [39, 37] on using in-domain public data for warmstarting DP training, we treat a random 1% partition of the LibriLight (LL) train dataset as a substitute for a small amount of in-domain public data being available. Further, for improved trade-offs, we conduct the DP pre-training on the entire LL train dataset (i.e., samples in the 1% public partition are incorporated into the private training dataset, providing a marginal improvement in the privacy accounting). Fine-tuning with LibriSpeech (LS) after only (non-private) pre-training with 1% LL yields a WER of 3.88/8.94. Note that this is better than our no pre-train upper bound of

4.43/11.23, but still substantially worse than the non-private lower bound of 2.21/4.29, validating the assumption about only a small amount on in-distribution public data being available in practical scenarios.

We present the results with public warmstart in the second column in Table 3, and compared to the random initialization results in Table 2, we observe a slight regression for smaller noise multipliers $\{5\text{e-}4, 1\text{e-}3\}$, whereas a significant improvement for the higher noise multipliers $\{5\text{e-}3, 1\text{e-}2\}$.

4.2.2. *Per-Layer Clipping*

There are two commonly-used variants of per-layer clipping [23, 17], denoted by the *uniform* variant (which splits the clipping bound equally amongst all layers), and the *dim* variant (which splits the clipping bound proportional to each layer's dimension). We conducted experiments using both the variants, and but found the *dim* variant to be outperforming the uniform one (similar to results seen in [17]).

We present the results for adding per-layer clipping for DP pre-training, post public warmstarting, in the third column in Table 3. While we observed the model diverging for the highest noise multiplier of 1e-2, we notice significant improvements in model quality for all other considered values of noise multiplier, corroborating the observation in [17] regarding the usefulness of per-layer clipping in the ASR domain.

4.2.3. *Gradient-based layer freezing (LayerFreeze)*

For reducing the dimensionality of DP training, some recent works [38, 31] propose starting from a pruned model that is initialized from a publicly pre-trained checkpoint. In this work, we devise a novel one-shot variant of model pruning called **Gradient-based Layer Freezing** (Algorithm 1), where instead of removing or freezing individual parameters based on their magnitudes, we freeze them layer-wise based on the normalized squared ℓ_2 norm of their gradients observed throughout the public warmstarting phase. After this operation, we continue DP pre-training with the pruned model and entire LL dataset.

Algorithm 1 Gradient-based Layer Freezing (LayerFreeze)

Input: *Model F with params $\theta \in \mathbb{R}^M$, num layers $\{i\}_{i=1}^{L}$, Loss fn $\mathcal{L}(\theta)$ over minibatch, Num iterations T, Optimizer* `opt`*, Grad* `update()` *fn, total params per layer* `dim()` *fn, top params $p\%$,* `freeze_top_layers` *whether to freeze top layers or the rest*

$\mathbf{u_o} \leftarrow 0$ ▷ Init sq grad vector with M dim
for $t \in [T]$ **do**
  $\mathbf{g_t} \leftarrow \nabla_{\theta_t}\mathcal{L}(\theta_t)$
  $\mathbf{u_t} \leftarrow \mathbf{u_{t-1}} + \mathbf{g_t^2}$ ▷ Accumulate sq grad
  $\theta_t \leftarrow \texttt{update}(\texttt{opt}, \mathbf{g_t}, \theta_{t-1})$
for $i \in [L]$ **do**
  $\mathbf{g_{layer_i}} \leftarrow \sum_{\text{param}\in i} \mathbf{u_T}[\texttt{param}]/\texttt{dim}(i)$
$top_par \leftarrow p \cdot M$ ▷ Top num of params
$top_layers \leftarrow [\,]$
for $layer_i$, _ in `sorted` (`enumerate` ($\mathbf{g_{layer}}$)) **do**
  if $\sum \texttt{dim}(top_layers + [layer_i]) \leq top_par$ **then**
    $top_layers.append(layer_i)$
  else $break$ ▷ Sort by grad & get layers till cut off $p\%$
if `freeze_top_layers` is `True` **then**
  $fr_layers \leftarrow top_layers$
else $fr_layers \leftarrow \{i\}_{i=1}^{L} - top_layers$
return Model F with frozen layers fr_layers

Once the norms of the per-layer gradients until our public warmstarting checkpoint are accumulated, we focus on $p\%$ of the model parameters, consisting of layers with the highest normalized accumulated squared gradient norm. We perform tuning experiments by freezing layers associated with either these $p\%$ parameter, or the remaining $1-p\%$ parameters. P is treated as a hyperparameter, explored in the range $\{0.015\%, 10\%]$ as seen in Figure 3. We consistently find that DP pre-training benefits from freezing layers with the top $p\%$ parameters, where the best case is when $p = 1\%$. To shed additional light on the layers frozen in our best-case scenario, layers corresponding to the bias, scale, beta, and gamma terms were frozen. We report the results of using LayerFreeze, along with per-layer clipping and public warmstarting, in the fourth column in Table 3. It is important to note that LayerFreeze provides significant improvements in model quality in all the considered settings.

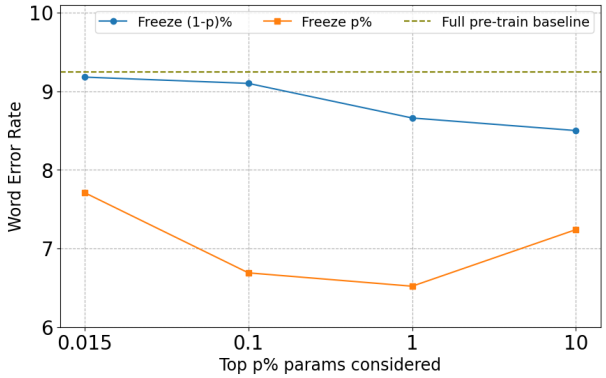

Figure 3: *Performance from tuning our LayerFreeze with different percentage of parameters frozen, while keeping the DP noise multiplier constant at* 1e-3*. Along the x-axis, we use p to refer to the % of parameters consisting of layers with the highest accumulated gradient norms. We run experiments with freezing either the p% parameters, or the remaining $(1-p)\%$. To save on compute, fine-tuning is done using an early pre-train checkpoint of 200k, assuming that the same conclusions hold for 1M.*

In summary, we obtain LibriSpeech WERs of 3.78/8.41 with (10, 1e-9)-DP for LibriLight with an extrapolation factor of 52 (low dataset scaling regime), and 2.81/5.89 with (10, 7.9e-11)-DP for LibriLight with an extrapolation factor of 530 (high dataset scaling regime).

5. Conclusion

We introduce DP to SSL for ASR, and a novel variant of model pruning called gradient-based layer freezing. Our technique improves the trade-offs for DP ASR pre-training, over improvements from public warmstarting and per-layer clipping. Overall, we demonstrate a DP training method that improves utility significantly while maintaining robust privacy guarantees under various extrapolation factors. Though our work provides a way to pre-train ASR encoders with strong DP guarantees, the extrapolations required to reach those guarantees can be limiting in some practical regimes. Improving computation trade-offs that we incur for reaching strong DP guarantees is an interesting direction we leave for future investigation.

6. References

[1] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," *Neural Information Processing Systems (NeurIPS)*, 2017.

[2] Q. Zhang, H. Lu, H. Sak, A. Tripathi, E. McDermott, S. Koo, and S. Kumar, "Transformer transducer: A streamable speech recognition model with transformer encoders and rnn-t loss," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2020.

[3] J. Li, V. Lavrukhin, B. Ginsburg, R. Leary, O. Kuchaiev, J. M. Cohen, H. Nguyen, and R. T. Gadde, "Jasper: An End-to-End Convolutional Neural Acoustic Model," in *Interspeech*, 2019.

[4] A. Gulati, J. Qin, C.-C. Chiu, N. Parmar, Y. Zhang, J. Yu, W. Han, S. Wang, Z. Zhang, Y. Wu, and R. Pang, "Conformer: Convolution-augmented Transformer for Speech Recognition," in *Interspeech*, 2020.

[5] C.-C. Chiu, J. Qin, Y. Zhang, J. Yu, and Y. Wu, "Self-supervised learning with random-projection quantizer for speech recognition," in *International Conference on Machine Learning*, 2022.

[6] R. Shokri, M. Stronati, C. Song, and V. Shmatikov, "Membership inference attacks against machine learning models," in *2017 IEEE symposium on security and privacy (SP)*. IEEE, 2017, pp. 3–18.

[7] N. Carlini, C. Liu, Ú. Erlingsson, J. Kos, and D. Song, "The secret sharer: Evaluating and testing unintended memorization in neural networks," in *28th USENIX Security Symposium*, 2019.

[8] N. Carlini, F. Tramer, E. Wallace, M. Jagielski, A. Herbert-Voss, K. Lee, A. Roberts, T. Brown, D. Song, U. Erlingsson *et al.*, "Extracting training data from large language models," in *30th USENIX Security Symposium*, 2021.

[9] N. Carlini, J. Hayes, M. Nasr, M. Jagielski, V. Sehwag, F. Tramer, B. Balle, D. Ippolito, and E. Wallace, "Extracting training data from diffusion models," in *USENIX Security Symposium*, 2023.

[10] E. Amid, O. D. Thakkar, A. Narayanan, R. Mathews, and F. Beaufays, "Extracting Targeted Training Data from ASR Models, and How to Mitigate It," in *Interspeech*, 2022.

[11] M. Jagielski, O. Thakkar, F. Tramèr, D. Ippolito, K. Lee, N. Carlini, E. Wallace, S. Song, A. G. Thakurta, N. Papernot, and C. Zhang, "Measuring forgetting of memorized training examples," in *The International Conference on Learning Representations (ICLR)*, 2023.

[12] L. Wang, O. Thakkar, and R. Mathews, "Unintended memorization in large asr models, and how to mitigate it," in *ICASSP*, 2024.

[13] M. Jagielski, O. Thakkar, and L. Wang, "Noise masking attacks and defenses for pretrained speech models," in *ICASSP*, 2024.

[14] A. Graves, S. Fernández, F. Gomez, and J. Schmidhuber, "Connectionist temporal classification: labelling unsegmented sequence data with recurrent neural networks," in *The International Conference on Machine Learning (ICML)*, 2006.

[15] C. Dwork, F. McSherry, K. Nissim, and A. Smith, "Calibrating noise to sensitivity in private data analysis," in *Theory of Cryptography Conference (TCC)*, 2006.

[16] P. Kairouz, B. McMahan, S. Song, O. Thakkar, A. Thakurta, and Z. Xu, "Practical and private (deep) learning without sampling or shuffling," in *ICML*, 2021.

[17] M. Pelikan, S. S. Azam, V. Feldman, J. Silovsky, K. Talwar, T. Likhomanenko *et al.*, "Federated learning with differential privacy for end-to-end speech recognition," *arXiv preprint arXiv:2310.00098*, 2023.

[18] D. Yu, S. Naik, A. Backurs, S. Gopi, H. A. Inan, G. Kamath, J. Kulkarni, Y. T. Lee, A. Manoel, L. Wutschitz, S. Yekhanin, and H. Zhang, "Differentially private fine-tuning of language models," in *ICLR*, 2022.

[19] Z. Bu, Y.-X. Wang, S. Zha, and G. Karypis, "Differentially private bias-term only fine-tuning of foundation models," *arXiv preprint arXiv:2210.00036*, 2022.

[20] E. J. Hu, Y. Shen, P. Wallis, Z. Allen-Zhu, Y. Li, S. Wang, L. Wang, and W. Chen, "LoRA: Low-rank adaptation of large language models," in *ICLR*, 2022.

[21] R. Bassily, A. Smith, and A. Thakurta, "Private empirical risk minimization: Efficient algorithms and tight error bounds," in *Annual Symposium on Foundations of Computer Science*, 2014.

[22] N. Ponomareva, J. Bastings, and S. Vassilvitskii, "Training text-to-text transformers with privacy guarantees," in *Findings of the Association for Computational Linguistics (ACL)*, 2022.

[23] H. B. McMahan, D. Ramage, K. Talwar, and L. Zhang, "Learning differentially private recurrent language models," in *ICLR*, 2018.

[24] M. Abadi, A. Chu, I. Goodfellow, H. B. McMahan, I. Mironov, K. Talwar, and L. Zhang, "Deep learning with differential privacy," in *The SIGSAC conference on computer and communications security*, 2016.

[25] P. Subramani, N. Vadivelu, and G. Kamath, "Enabling fast differentially private sgd via just-in-time compilation and vectorization," *NeurIPS*, 2021.

[26] X. Li, D. Liu, T. B. Hashimoto, H. A. Inan, J. Kulkarni, Y.-T. Lee, and A. Guha Thakurta, "When does differentially private learning not suffer in high dimensions?" *NeurIPS*, 2022.

[27] R. Anil, B. Ghazi, V. Gupta, R. Kumar, and P. Manurangsi, "Large-scale differentially private BERT," in *Findings of the Association for Computational Linguistics: EMNLP*, 2022.

[28] A. Kurakin, S. Song, S. Chien, R. Geambasu, A. Terzis, and A. Thakurta, "Toward training at imagenet scale with differential privacy," *arXiv preprint arXiv:2201.12328*, 2022.

[29] S. De, L. Berrada, J. Hayes, S. L. Smith, and B. Balle, "Unlocking high-accuracy differentially private image classification through scale," *arXiv preprint arXiv:2204.13650*, 2022.

[30] X. Li, F. Tramer, P. Liang, and T. Hashimoto, "Large language models can be strong differentially private learners," *arXiv preprint arXiv:2110.05679*, 2021.

[31] L. Watson, E. Gan, M. Dantam, B. Mirzasoleiman, and R. Sarkar, "Inference and interference: The role of clipping, pruning and loss landscapes in differentially private stochastic gradient descent," *arXiv preprint arXiv:2311.06839*, 2023.

[32] Y. Zhang, D. S. Park, W. Han, J. Qin, A. Gulati, J. Shor, A. Jansen, Y. Xu, Y. Huang, S. Wang *et al.*, "Bigssl: Exploring the frontier of large-scale semi-supervised learning for automatic speech recognition," *IEEE Journal of Selected Topics in Signal Processing*, 2022.

[33] J. Kahn, M. Rivière, W. Zheng, E. Kharitonov, Q. Xu, P.-E. Mazaré, J. Karadayi, V. Liptchinsky, R. Collobert, C. Fuegen *et al.*, "Libri-light: A benchmark for asr with limited or no supervision," in *ICASSP*, 2020.

[34] V. Panayotov, G. Chen, D. Povey, and S. Khudanpur, "Librispeech: An asr corpus based on public domain audio books," in *ICASSP*, 2015.

[35] I. Mironov, K. Talwar, and L. Zhang, "Rényi differential privacy of the sampled gaussian mechanism," *arXiv preprint arXiv:1908.10530*, 2019.

[36] N. Ponomareva, H. Hazimeh, A. Kurakin, Z. Xu, C. Denison, H. B. McMahan, S. Vassilvitskii, S. Chien, and A. G. Thakurta, "How to dp-fy ml: A practical guide to machine learning with differential privacy," *Journal of Artificial Intelligence Research*, 2023.

[37] A. Ganesh, M. Haghifam, M. Nasr, S. Oh, T. Steinke, O. Thakkar, A. G. Thakurta, and L. Wang, "Why is public pretraining necessary for private model training?" in *ICML*, 2023.

[38] Z. Luo, D. J. Wu, E. Adeli, and L. Fei-Fei, "Scalable differential privacy with sparse network finetuning," in *The IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021.

[39] E. Amid, A. Ganesh, R. Mathews, S. Ramaswamy, S. Song, T. Steinke, V. M. Suriyakumar, O. Thakkar, and A. Thakurta, "Public data-assisted mirror descent for private model training," in *ICML*, 2022.