Google Research

Sparsity-Preserving Differentially Private Training of Large Embedding Models

Badih Ghazi[1], Yangsibo Huang[1, 2, 3], Pritish Kamath[1], Ravi Kumar[1], Pasin Manurangsi[1], Amer Sinha[1], Chiyuan Zhang[1]
[1] Google Research [2] Princeton University [3] Princeton Language and Intelligence

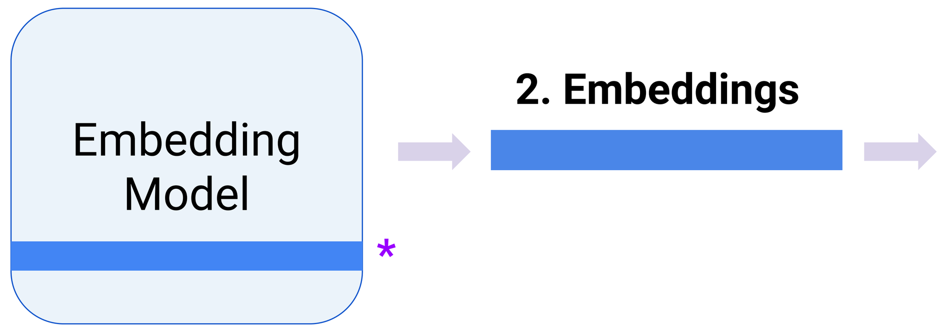
PRINCETON UNIVERSITY

PRINCETON Language + Intelligence

Introduction

Embedding Models Process (Private) Non-numerical Inputs

1. **Non-numerical data**
   - Demographic data
   - Private chats
   - Medical history
   - ...

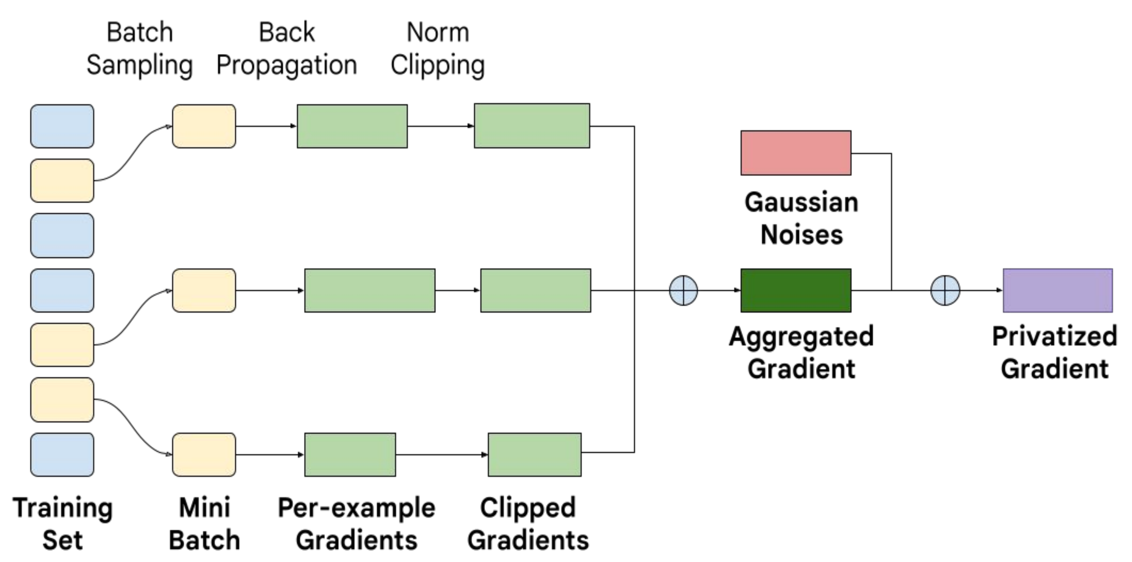


3. **Downstream tasks**
   - Recommender systems
   - Chatbots
   - ML-powered diagnosis
   - ...

* sparse lookup ⟶ sparse gradients (leveraged by customized APIs such as Google TPUs for efficiency)

Protect Privacy: Differentially Private SGD (DP-SGD) [1]

Adds Dense Noise to Gradients to Protect Privacy



sparse gradients ⟶ dense gradients (more computation)

Our Main Contributions:

We propose sparsity-preserving DP training algorithms for Large Embedding Models and achieve:

Recommendation tasks

★ > 10^5× reduction in gradient size in recommendation tasks (# non-zero embedding gradients rows) while maintaining accuracy

★ Translates to > 20x wall-clock time improvement

Natural language understanding tasks

★ > 50x gradient size reduction while maintaining accuracy

★ Outperforms the LoRA method [3]

★ Larger improvements in multilingual models

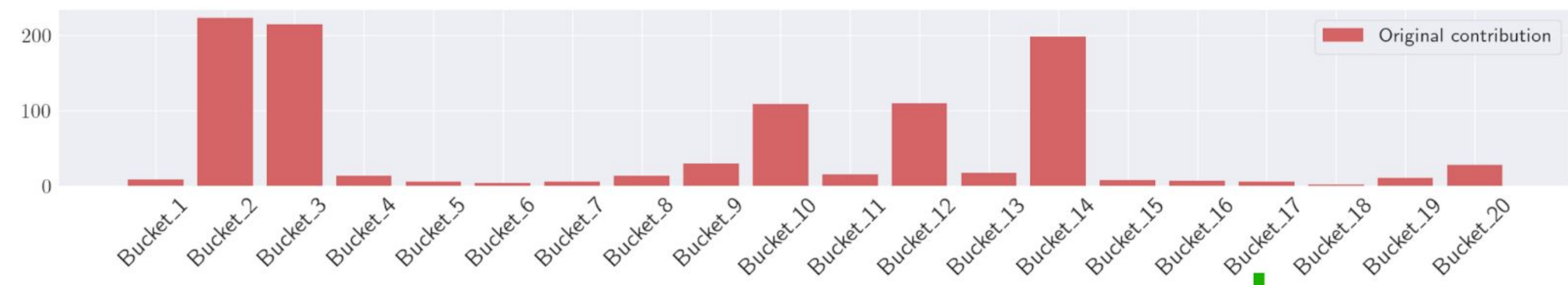
Method

Our Proposal: Adaptive Filtering-Enabled Sparse Training (DP-AdaFEST)

We extend standard DP-SGD with an extra mechanism at each iteration to privately select the "top features":

1. Compute how many examples **contributed** to each non-numerical feature "bucket";
2. Restrict the total contribution from each example by **clipping** their counts;
3. **Add Gaussian noise** to the contribution count of each feature bucket;
4. **Select** only the features to be included in the gradient update that have a count above a given threshold (a sparsity-controlling parameter) to be included in the gradient update, thus maintaining sparsity.

DP-AdaFEST is DP: the privacy cost can be easily computed by composing it with the standard DP-SGD iterations (§ 3.3 in paper)



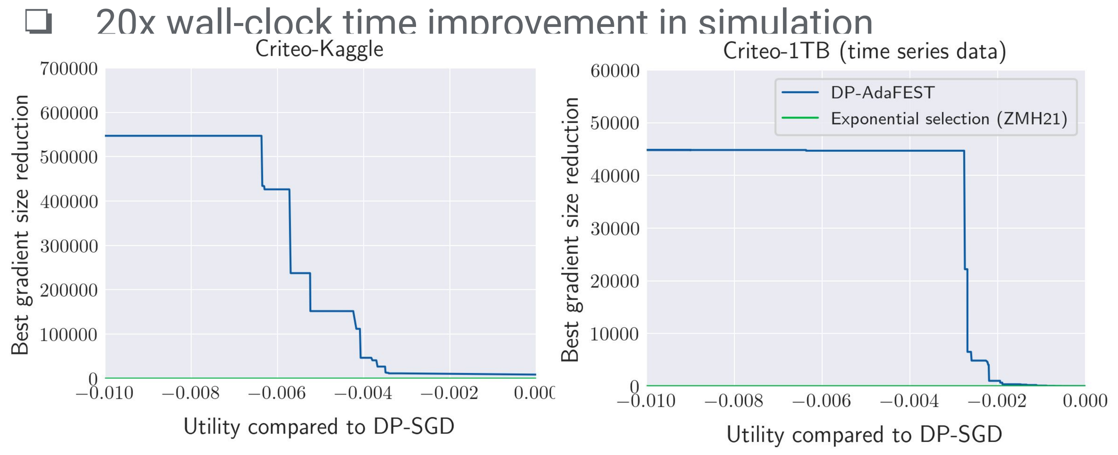
Computes privacy-preserving contribution scores for each bucket

Only buckets in purple are selected for gradient update

Results

Recommendation Tasks

- Criteo-Kaggle & Criteo-1TB (Time-series). Vocab size: 1.7M.
- > 10^5× reduction in gradient size w/ comparable utility
- 20x wall-clock time improvement in simulation



Natural Language Understanding Tasks

- SST, QNLI, QQP from GLUE benchmark [2]. Vocab size: ~50k.
- ~50x reduction in gradient size w/ comparable utility (due to already condensed vocabulary)

Comparison w/ LoRA [3]

- DP-AdaFEST achieves sparser gradients compared to LoRA, which adapts weight matrices using low-rank approximation
- DP-AdaFEST benefits from the efficient embedding lookup via customized APIs. LoRA would not be able to leverage them (it requires relatively expensive matrix multiplication)

Acc. compared to DP-SGD	Best gradient size reduction	
	DP-AdaFEST	LoRA
-0.001	**17.41×**	5.91×
-0.005	**62.14×**	23.64×
-0.01	**62.14×**	47.28×

Larger Improvement on Multilingual Models

Acc. compared to DP-SGD	Best gradient size reduction	
	RoBERTa (|V|: 50k) [4]	XLM-R (|V|: 200k) [5]
-0.001	17.41×	**19.84×**
-0.005	62.14×	**73.42×**
-0.01	62.14×	**162.13×**

Conclusion

Takeaways

- We effectively address the "destroyed gradient sparsity" challenge when applying general-purpose DP-SGD to large-scale embedding models, via the proposal of **DP-AdaFEST**.
- DP-AdaFEST achieves a substantially sparser gradient in recommendation tasks, with a reduction in gradient size of over **10^5×** (translates into 20x wall-clock time improvement) compared to the dense gradient produced by vanilla DP-SGD, while maintaining comparable levels of accuracy.
- DP-AdaFEST is also **more effective than LoRA** in reducing the gradient size when applied to natural language understanding tasks.

Future Work

- Leverage specialized hardware to further optimize the computational performance and speed up the training
- Integration of our methods with non-centralized training paradigms (e.g., Federated Learning)

References

[1] Deep Learning with Differential Privacy. Abadi et al., CCS 2016
[2] GLUE: A Multi-Task Benchmark and Analysis Platform for Natural Language Understanding. Wang et al., ICLR 2019
[3] LoRA: Low-rank Adaptation of Large Language Models. Hu et al., ICLR 2022
[4] RoBERTa: A Robustly Optimized BERT Pretraining Approach. Liu et al., arxiv preprint 2019
[5] Unsupervised Cross-lingual Representation Learning at Scale. Conneau et al., ACL 2020