Supplementary Information for "Privacy-preserving, Efficient and Effective Machine Learning"

Chuhan Wu[1], Fangzhao Wu[2*], Tao Qi[1], Yongfeng Huang[1*], and Xing Xie[2]

[1]Department of Electronic Engineering, Tsinghua University, Beijing 100084, China
[2]Microsoft Research Asia, Beijing 100080, China
[*]Correspondence: fangzwu@microsoft.com, yfhuang@tsinghua.edu.cn

Supplementary Materials

Datasets

The details of the datasets we used in our work are introduced as follows. The MNLI and QNLI datasets are used for natural language inference, i.e., judging the entailment relations between two (short) texts. The QQP dataset is used for paraphrase by identifying whether a pair of questions are semantically equivalent. The SST-2 dataset is for binary text sentiment classification, i.e., identifying the sentiment polarities of texts. The task on MNLI is three-way classification, while is binary classification on QNLI, QQP, and SST-2. The training, validation, and test splits of these datasets are given. The MIND dataset is a large-scale news recommendation dataset, which contains the news impression logs of 1 million users on Microsoft News in a period of six weeks (from October 12 to November 22, 2019). The task is predicting whether a candidate news article within a test impression will be clicked by a user given the historical clicked news of this user. The samples in the last week are used as the test set, samples on the day before the test period are for validation, and the rest are used for training. The detailed statistics of these datasets are provided (Extended Data Tables 1 and 2).

Experimental Settings

Since there is no user information in the text classification datasets (MNLI, QNLI, QQP, and SST-2), we use regard different samples are kept by independent clients. On the MIND dataset, we partition the training dataset based on user IDs, and the samples from different users are placed on different clients. To implement the models, we use the transformers library (version 4.11) to import the parameters pretrained language models. We randomly sample 128 clients to participate in each round of federated learning and the standard FedAvg framework is used[1]. For the full finetuning and GEP methods, the gradient norm clip threshold is 10, and is 1 for other parameter-efficient methods (FedPrompt and RGP). The prompt prefix length is 16 and the encoder hidden dimension is 64. Without specific mention, the default privacy budget ε is 5 and the probability δ is 1e-3. We use Adam[2] as the optimizer. We list the detailed hyperparameter settings on different datasets (Extended Data Table 3). We repeat each experiment 5 times and report the mean results with standard deviations or 95% confidence intervals.

Performance Metrics

The performance metric on the text classification datasets is accuracy, which is defined as follows:

$$\text{Accuracy} = \frac{\text{TP}+\text{TN}}{\text{TP}+\text{FP}+\text{TN}+\text{FN}}, \tag{1}$$

where TP, FP, TN, and FN are true positive, false positive, true negative, and false negative, respectively. The performance metrics on the MIND dataset are used for evaluating the ranking quality, and we use AUC, MRR, nDCG@5, and nDCG@10 as the measurements. They are formulated as follows:

$$\text{AUC} = \frac{\sum_{\text{p}\in\mathscr{P}}\sum_{\text{n}\in\mathscr{N}}\text{I}[\text{P(p)} > \text{P(n)}]}{|\mathscr{P}||\mathscr{N}|}, \tag{2}$$

$$\text{MRR} = \frac{1}{\text{N}_\text{p}}\sum_{\text{i}=1}^{\text{N}_\text{p}}\frac{1}{\text{Rank}(\text{p}_\text{i})}, \tag{3}$$

$$\text{nDCG@K} = \frac{\sum_{\text{i}=1}^{\text{K}}(2^{\text{r}_\text{i}}-1)/\log_2(1+\text{i})}{\sum_{\text{i}=1}^{\text{N}_\text{p}}1/\log_2(1+\text{i})}, \tag{4}$$

where $P(\cdot)$ denotes the predicted probability of a candidate news article to be clicked. $\mathscr{P}$ and $\mathscr{N}$ stand for the sets of clicked and non-clicked samples, respectively. $I[\cdot]$ is an indicator function. r_i is the relevance score for the i-th ranked news, which is 1 if clicked else 0. The metric nDCG@K is computed based on users' real click behaviors on the top K news ranked by the recommender system.

Basic Model Settings

For text classification tasks, the basic model we used is only the pretrained language model with a single classifier layer on top of the output hidden representations[3]. For the news recommendation task, we employ the framework of PLM-NR[4] that first uses pretrained language models to encode candidate news and clicked news, then uses a user encoder with attention mechanism[5] to learn user representations, and finally evaluate the relevance between news embeddings and user embeddings via inner product to compute the click scores.

Experimental Environment

We conduct experiments on a cloud Linux machine installed with the Ubuntu 16.04 operating system. It has 16 Tesla V100 GPUs and each of them has 32GB of memory. The CPU type is Intel(R) Xeon(R) Platinum 8168 CPU @ 2.70GHz. The Python version is 3.6.9. The total machine memory is 256GB.

Influence of Privacy Protection Intensity

Here we evaluate the performance of FedPrompt under different privacy budget ε and leakage probability δ (Extended Data Figure 1). We observe that the performance sacrifice is larger when a smaller privacy budget is used and a smaller leakage probability is tolerated. Nevertheless, FedPrompt is still informative under very strong privacy protection requirement (e.g., $\varepsilon = 2$ and $\delta = 1e-5$). It shows FedPrompt is effective in achieving the tradeoff between model utility and privacy protection.

Comparison with Communication-efficient Federated Learning Baselines

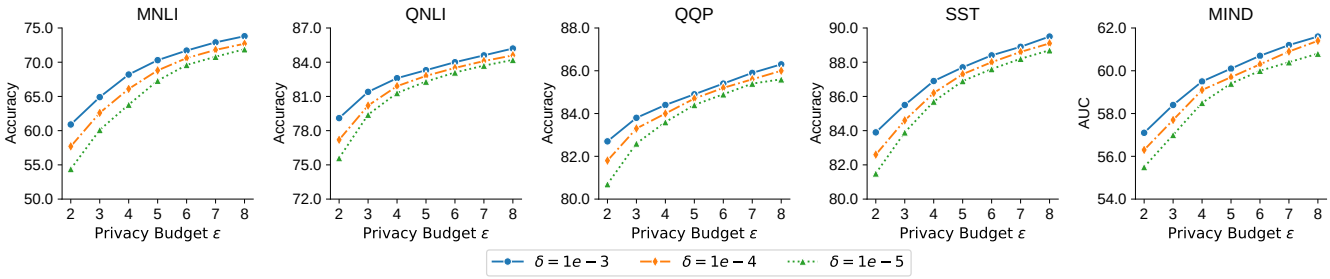
Although many communication-efficient federated learning methods are proposed in recent years, we argue that they usually cannot reduce the communication cost so effectively as FedPrompt. We compare FedPrompt with several widely used communication-efficient federated learning methods, including FedDropout[6], FetchSGD[7], SCAFFOLD[8], FedPAQ[9], and finetuning the distilled model TinyBERT[10], in terms of their performance under LDP (Extended Data Figure 2) and the average communication cost per client in each round (Extended Data Figure 3). We find FedPrompt achieves the best performance under LDP settings, meanwhile having the lowest communication cost. This is because other compared methods cannot effectively reduce the number of communicated parameters and the norm of their gradients cannot be effectively reduced. Thus, the LDP noise added to their model updates can substantially hurt their performance.

Gradient Norm Analysis

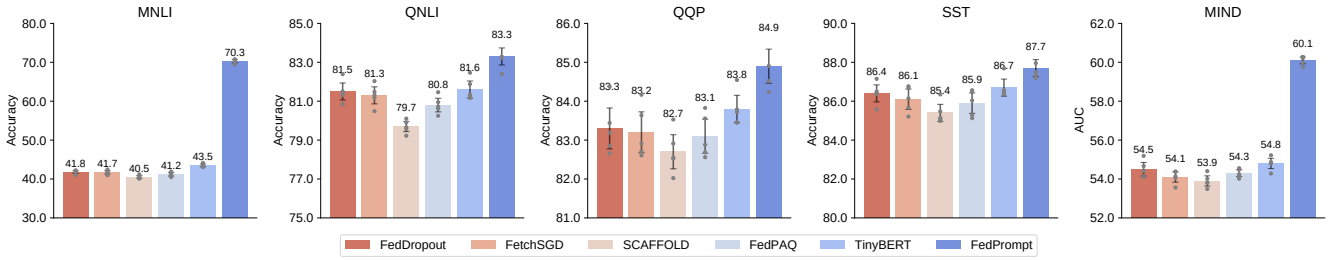
We compare the gradient norm distributions of the full finetuning method and our FedPrompt method (Extended Data Figure 4). The results indicate that the L_2 norms of local model updates in FedPrompt are much smaller than those in the full model finetuning method. Thus, the noise intensity added to local prompt updates is much weaker, thereby FedPrompt has a better model utility.

References

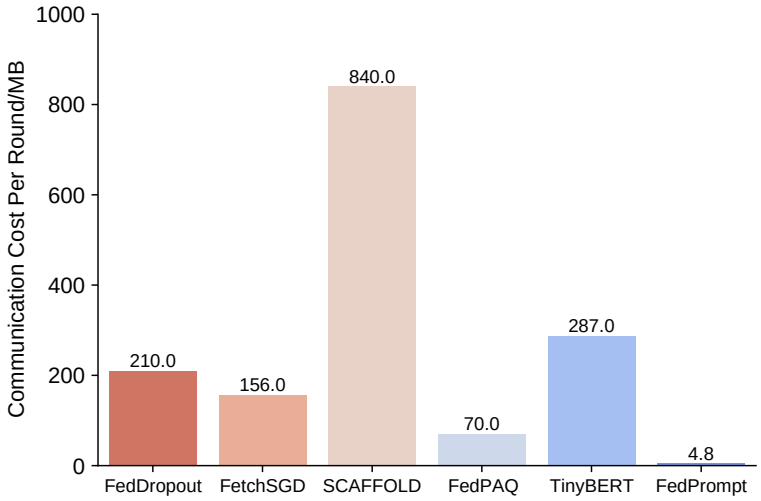
1. McMahan, B., Moore, E., Ramage, D., Hampson, S. & y Arcas, B. A. Communication-efficient learning of deep networks from decentralized data. In *AISTATS*, 1273–1282 (2017).
2. Kingma, D. P. & Ba, J. Adam: A method for stochastic optimization. In *ICLR* (2015).
3. Devlin, J., Chang, M.-W., Lee, K. & Toutanova, K. Bert: Pre-training of deep bidirectional transformers for language understanding. In *NAACL-HLT*, 4171–4186 (2019).
4. Wu, C., Wu, F., Qi, T. & Huang, Y. Empowering news recommendation with pre-trained language models. In *SIGIR*, 1652–1656 (2021).
5. Yang, Z. *et al.* Hierarchical attention networks for document classification. In *NAACL-HLT*, 1480–1489 (2016).
6. Caldas, S., Konečny, J., McMahan, H. B. & Talwalkar, A. Expanding the reach of federated learning by reducing client resource requirements. *arXiv preprint arXiv:1812.07210* (2018).
7. Rothchild, D. *et al.* Fetchsgd: Communication-efficient federated learning with sketching. In *ICML*, 8253–8265 (PMLR, 2020).
8. Karimireddy, S. P. *et al.* Scaffold: Stochastic controlled averaging for federated learning. In *ICML*, 5132–5143 (PMLR, 2020).
9. Reisizadeh, A., Mokhtari, A., Hassani, H., Jadbabaie, A. & Pedarsani, R. Fedpaq: A communication-efficient federated learning method with periodic averaging and quantization. In *AISTATS*, 2021–2031 (PMLR, 2020).
10. Jiao, X. *et al.* Tinybert: Distilling BERT for natural language understanding. In *EMNLP Findings*, 4163–4174 (2020).



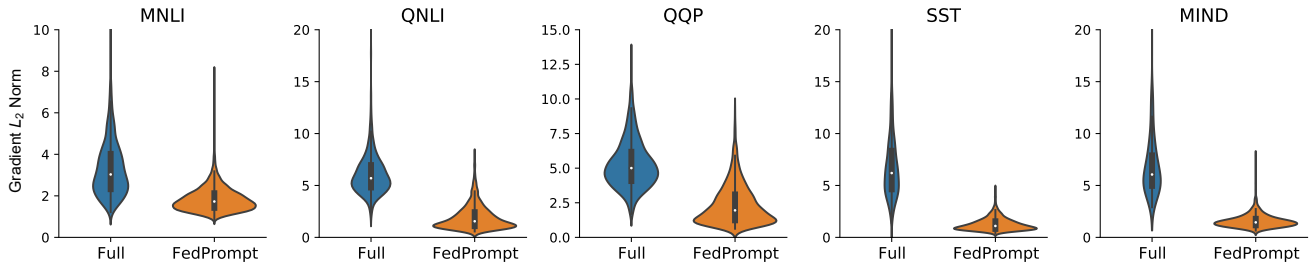
Extended Data Figure 1. The performance of FedPrompt under different privacy budget ε and different leakage probability δ. BERT-Base is used as the basic model. A lower budget or a lower leakage probability indicates stronger privacy protection, which yields a larger performance decrease.



Extended Data Figure 2. The performance of different communication-efficient federated learning methods under differentially private settings. The mean scores with 95% confidence intervals are presented (n=5 independent experiments). FedPrompt consistently outperforms other compared methods ($p < 0.05$).



Extended Data Figure 3. The average communication cost per client in each round. FedPrompt has the lowest communication cost among all compared methods.



Extended Data Figure 4. The norm distributions of the local model updates in full finetuning and FedPrompt. The violin plot includes the minimum, maximum, median, first quartile, and third quartile values of the gradient norms. FedPrompt has much smaller gradient norms than the full model update method.

	#train	#val	#test	avg. text len.
MNLI	392,702	9,832	9,847	29.81
QNLI	104,743	5,463	5,463	36.56
QQP	363,846	40,430	390,965	22.25
SST-2	67,349	872	1,821	9.79

Extended Data Table 1. Statistics of the MNLI, QNLI, QQP, and SST-2 datasets. The average text length means the average number of tokens in each sample.

# users	1,000,000	# impressions	15,777,377
# news	161,013	# news click behaviors	24,155,470
avg. title len.	11.52	# training samples	2,186,683
# validation samples	365,200	# test samples	2,341,619

Extended Data Table 2. Statistics of the MIND dataset. The average title length means the average number of tokens in the news title. Each sample is composed of a user's historical clicked news articles and the click/non-click behaviors on a candidate news article.

Hyperparameters	MNLI	QNLI	QQP	SST-2	MIND
full model learning rate	1e-5	2e-5	1e-5	2e-5	2e-6
prompt model learning rate	1e-4	2e-4	1e-4	2e-4	1e-4
optimizer	Adam	Adam	Adam	Adam	Adam
prompt prefix length	16	16	16	16	16
prompt encoder hidden size	64	64	64	64	64
epoch	20	20	20	20	3

Extended Data Table 3. The hyperparameter settings on different datasets.