A Broader Impacts

The growing importance of in-context learning as a paradigm for leveraging LLMs on private downstream tasks has significant implications for privacy. We present the first approaches for obtaining prompts with privacy guarantees, thereby enabling the use of this learning paradigm on sensitive data. This advancement has the potential to increase trust and acceptance of LLM-based systems for private applications. Our approach PromptPATE is the first viable technique for private downstream adaptation of black-box LLMs, which enables integrations into the state-of-the-art commercial LLM APIs. We acknowledge that—as with any application that relies on DP—care must be taken when choosing the privacy parameters ε and δ since setting these incorrectly can lead to a false sense of privacy. Therefore, our work orientates at the privacy parameters that have been shown to provide reasonable protection in prior work. Thereby, we also ensure consistency and comparability in evaluations between the different appraoches.

B Limitations

Tuning Instructions and Templates. For our discrete prompts, we did not tune the instructions or templates but instead relied on a template from prior work [58]. The effectiveness and performance of our PromptPATE could potentially be further improved by tuning the instructions and templates.

Privacy Risk of Pretrained LLM: We build on pretrained LLMs to learn and deploy our private prompts. Our methods solely target the protection of the private data used for these prompts. However, it is also important to acknowledge the inherent privacy risks for data used to pretrain the LLM. We leave the pretrainig of LLMs with privacy guarantees to an orthogonal line of work.

Limited Monetary Budget for our Experiments. Due to cost limitations, we were unable to experiment with the latest and best available model, GPT4. Our experiments with GPT3-Curie in comparison to less powerful GPT3-Babbage however indicate the clear trend the our private prompts improve in performance as the non-private baseline improves due to better models. Furthermore, again due to the cost limitation, we were not able to incorporate a larger number of teachers in our experiments for PromptPATE. Therefore, the best non-private teacher baseline that we report might not be the best achievable if one had more teachers to choose from. We chose from 200 and note that with more (and potentially better teachers), not only the baseline but also the teacher ensemble's performance would get better.

Hyperparameter Tuning. To save computation costs, we did not exhaustively tune all hyperparameters in our experiments. While our approach still achieves high utility and good privacy-utility trade-offs, we acknowledge that with more hyperparameter tuning the performance together with the understanding of optimal configurations for private prompt learning could increase.

Assumption of a Trusted LLM API Provider. In our work, the API provider gets to interact with the private data, for example, through the teachers' prompts in PromptPATE. Therefore, we have to assume trust in the API provider. The privacy guarantees through our private prompt learning protect the privacy of the prompt data against users that interact with the prompted LLM. In practice, companies that are concerned about the privacy of their data with respect to the API provider could make contracts with the API providers on the use of their data or buy access plans that guarantee that data queried to the API is treated privately. We leave implementing cryptographic approaches that could relief the assumption on trusting the API provider entirely, for example, by enabling the LLM to run inference on encrypted private data to future work.

C Additional Insights into our Methods

C.1 PromptDPSGD

We present the full PromptDPSGD algorithm in Algorithm 1.

Algorithm 1: PromptDPSGD. In contrast to the standard DPSGD algorithm that updates model parameters during private training or fine-tuning, our PromptDPSGD privately updates the soft prompt parameters. We highlight these changes with respect to standard DPSGD training or fine-tuning in blue.

Require: Private downstream data $D = \{(x_i, y_i) \mid i \in [N]\}$, prompt sequence length s, embedding dimensionality e, trained LLM L with frozen parameters, loss function $\ell(L_p, x)$ for prompted LLM, **Params:** learning rate η_t, noise scale σ, sampling rate q, max gradient norm c, training iterations T.

Initialize $P_0 \in \mathbb{R}^{s \times e}$ at random
for $t \in [T]$ **do**
  Sample mini-batch B_t according to sampling rate q from D {Poisson sampling}
  For each $i \in |B_t|$, compute $\mathbf{g}_t(x_i) \leftarrow \nabla_{P_t}\, \ell(L_P, x_i)$ {Compute per sample gradient w.r.t. p_t}
  $\bar{\mathbf{g}}_t(x_i) \leftarrow \mathbf{g}_t(x_i) / \max\left(1, \frac{\|\mathbf{g}_t(x_i)\|_2}{c}\right)$ {Clip gradient}
  $\tilde{\mathbf{g}}_t \leftarrow \frac{1}{|B_t|}\left(\sum_i \bar{\mathbf{g}}_t(x_i) + \mathcal{N}\left(0, \sigma^2 c^2 \mathbf{I}\right)\right)$ {Add noise}
  $P_{t+1} \leftarrow P_t - \eta_t \tilde{\mathbf{g}}_t$ {Update soft prompt}
end for
Output p_T and compute the overall privacy cost (ε, δ).

C.2 PromptPATE

Extended Background on PATE. We include the standard Confident-GNMax Aggregator Algorithm from [37] below.

Algorithm 2: Confident-GNMax Aggregator by [37]

Require: input x, threshold T, noise parameters σ_1 and σ_2

if $\max_j\{\sum_{i\in[E]} n_{i,j}(x)\} + \mathcal{N}(0, \sigma_1^2) \geq T$ **then**
  Output $\arg\max_j\{\sum_{i\in[E]} n_{i,j}(\mathbf{x}) + \mathcal{N}(0, \sigma_2^2)\}$
else
  Output $\perp$
end if

C.3 Privacy Analysis

PromptDPSGD. Our PromptDPSGD can be seen as a repeated sampled Gaussian mechanism [1], with sampling performed over the entirety of the private prompt dataset. The difference to standard DPSGD for training or fine-tuning is that we do not update the model parameters, but the trainable embeddings for the soft prompts. This is conceptually different from standard DPSGD in terms of which parameters are updated. The privacy guarantees of the training mechanism still follow Abadi *et al.* [1], but with respect to the soft prompt embeddings: whether or not a particular data point will be included in the private training set used for tuning the prompt, the resulting soft prompt embeddings after training will be roughly the same. Especially by applying the clipping operation at every step, each mechanism's sensitivity is bounded by c. Privacy is then implemented as the trainable soft prompt embeddings are updated while adding noise noise drawn from $\mathcal{N}(0, c^2\sigma^2 I)$.

Theorem 1 (Privacy of PromptDPSGD)**.** *Let T be the total number of repetitions (training iterations) of our PromptDPSGD and the sampling rate be denoted by q. Then, there exist two constants c_1 and c_2, such that for any $\varepsilon < c_1 q^2 T$ our PromptDPSGD guarantees (ε, δ)-DP, if for any $\delta > 0$, we choose the noise according to $\sigma \geq c_2 \frac{qc\sqrt{T \log 1/\delta}}{\varepsilon}$.*

Proof. The proof follows the one by Abadi *et al.* [1], using their moments accountant that models the privacy loss as a random variable dependent on the stochastic noise added. □

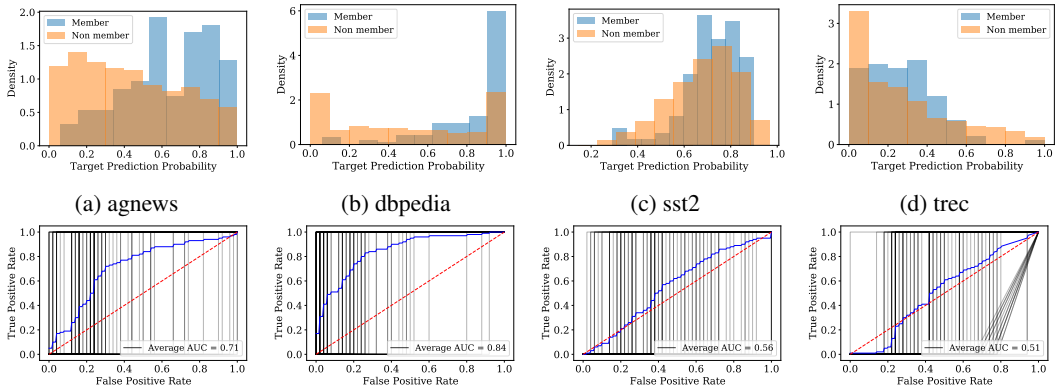
PromptPATE. Our PromptPATE relies entirely on the Confident GNMAX algorithm from Papernot *et al.* [37]. We preserve the assumption underlying the algorithm and the respective privacy analysis that the sensitivity during the voting mechanism equals one. This is done in PromptPATE by assigning *disjoint* data points from the private prompt downstream dataset to all teachers. As a consequence, the privacy analysis of our PromptPATE entirely follows Papernot *et al.* [37].

Both our PromptDPSGD and PromptPATE experience the post-processing properties of DP, *i.e.,* once trained, the privacy guarantee (ε, δ) sets an upper bound on privacy leakage for the prompt data, independent on the number and type of queries that will be posed to the final prompted LLM.

D Additional Results

D.1 Membership Inference Attacks

We present the full results of MIA against GPT3 with one-shot prompts on 4 datasets in 4.



(a) agnews (b) dbpedia (c) sst2 (d) trec

(e) agnews (f) dbpedia (g) sst2 (h) trec

Figure 4: **MIA Risk over Multiple Datasets on GPT3.** We study GPT3-babbage prompted with 100 different one-shot examples on four datasets. *top*: We present the prediction probabilities at the correct class for members (the one-shot example) and non-members (50 randomly sampled private points). The output probability for members is significantly higher than for non-member data points. *bottom*: We present the AUC-ROC curves of our MIA against the 100 prompts (gray lines) and the blue line as an average over all attacks. Given that each prompt has only one member, the resulting TPRs can only be 0% or 100% which leads to the step-shape of the gray curves. The result indicates that our attack is significantly more successful than random guessing (the red dashed line).

In addition, we also perform similar experiments on GPT2-xl with four-shot examples, with results presented in Figure 5. We replace dbpedia with cb because the input in dbpedia is usually longer than the context length of GPT2.

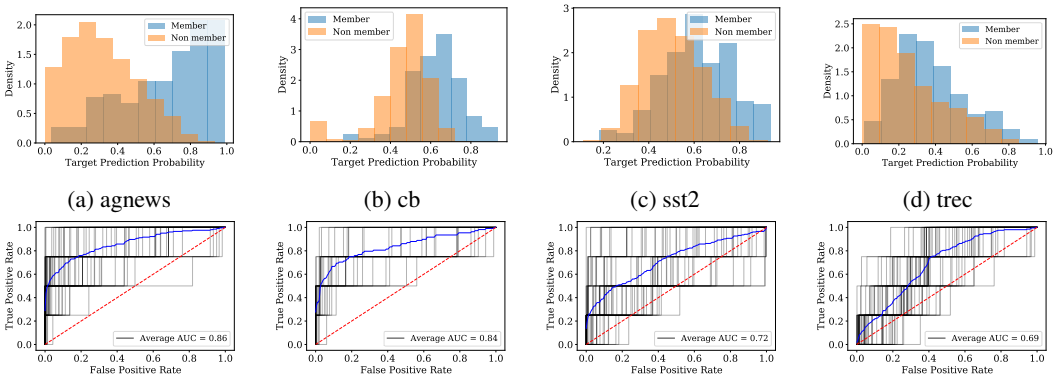
D.2 PromptPATE on Claude

We present the experiment results of PromptPATE on Claude [3]. Different from GPT3 that outputs logits over the whole vocabulary, Claude only gives us access to the next most likely token.

Experimental Setup. **Teachers:** We rely on Claude-v1 as the base LLM. We use 2-shot prompts for sst2 and agnews, 4-shot for trec and 1-shot for dbpedia. We set the maximum generated tokens to 1 and temperatures to 0. We also create an "other" category in case the moel's output does not fall under any specified categories. For each setting, we deploy 400 teacher prompts. **Private knowledge transfer:** We use the implementation of PATE's Confident GNMAX algorithm and the privacy accounting from [12] and report our algorithm's hyperparameters in Appendix E. **Student:** We limit the size of the public dataset to 200 input sequences from the respective datasets. The number of shots for students corresponds with the teachers.

D.3 More results for PromptDPSGD

We present the additional results for PromptDPSGD with $\varepsilon = 3$ on the classification tasks in Table 5.



(a) agnews (b) cb (c) sst2 (d) trec

(e) agnews (f) cb (g) sst2 (h) trec

Figure 5: **MIA Risk over Multiple Datasets on GPT2-xl (4 shot).** We study GPT2-xl prompted with 100 different four-shot examples on four datasets. *top*: We present the prediction probabilities at the correct class for members (the one-shot example) and non-members (50 randomly sampled private points). The output probability for members is significantly higher than for non-member data points. *bottom*: We present the AUC-ROC curves of our MIA against the 100 prompts (gray lines) and the blue line as an average over all attacks. Given that each prompt has only one member, the resulting TPRs can only be 0%, 25%, 50%, 75% or 100% which leads to the step-shape of the gray curves. The result indicates that our attack is significantly more successful than random guessing (the red dashed line).

	Lower Bound	Ens. Acc.	Upper Bound	Our PromptPATE		
Private	$\varepsilon = 0$	$\varepsilon = \infty$	$\varepsilon = \infty$	Public	ε	Test acc
sst2	92.7	96.0	98.0	sst2	0.048	95.7 ± 1.4
agnews	72.4	79.1	82.7	agnews	0.056	74.6 ± 1.5
trec	69.0	79.9	82.2	trec	0.068	79.3 ± 1.2
dbpedia	88.0	92.4	93.5	dbpedia	0.042	90.9 ± 0.6

Table 3: **Performance of PromptPATE on Claude.** We compare PromptPATE with three baselines: zero-shot (Lower Bound), the ensemble's accuracy (Ens. Acc), and the non-private baseline (Upper Bound) on four classification benchmarks. We find that PromptPATE achieves strong privacy protection ($\varepsilon < 0.1$ at $\delta = 10^{-6}$) and utility close to the non-private and significantly higher than the zero-shot.

E Additional Setup

E.1 PromptDPSGD

We train PromptDPSGD on NVIDIA A100 GPUs. We execute (hyper-)parameter search that takes into account learning rate (LR), max grad norm (GRAD), number of epochs (Epochs), the token length of prefix and prompt. In general, we find that the prompt and prefix token length of 10 is close to the optimal value in most cases. For the private (hyper-)parameters, in most cases we tune for $\varepsilon = 8$ and use similar (or even the same) parameters for other ε values. We set the max grad norm to 0.1 in most cases and then adjust the number of epochs (the more the better, for example, 100), and the learning rate [54][3]. The batch size is set by default to 1024.

We show the specific parameters chosen for PromptDPSGD in Table 6.

[3]We would like to thank the authors of [54] for their help, especially for the very useful and practical pieces of advice on how to tune the parameters for differential privacy from Huseyin A. Inan.

Dataset	M	Soft-Prompt (Our)		Prefix (Our)		Full-Tuning [25]		LoRA-Tuning [54]	
	P	<10K		<100K		125M		1.2M	
	G	$\varepsilon = 3$	$\varepsilon = \infty$	$\varepsilon = 3$	$\varepsilon = \infty$	$\varepsilon = 3$	$\varepsilon = \infty$	$\varepsilon = 3$	$\varepsilon = \infty$
SST2		90.48	95.64	90.37	96.33	91.86	96.40	92.60	96.60
QNLI		83.62	89.48	86.05	94.84	87.42	94.70	86.97	94.70
QQP		80.29	86.56	80.89	91.42	85.56	92.20	85.12	92.20
MNLI		73.97	82.49	80.10	90.34	82.99	90.20	82.08	90.20

Table 4: **Private classification with soft prompts and prefix for $\varepsilon = \{3, \infty\}$ and the RoBERTa$_{BASE}$ model.** We use the same setup and notation as in Table 1.

Dataset	M	Soft-Prompt (Our)	Prefix (Our)	Full-Tuning [25]
	P	<10K	<100K	125M
SST2		90.37	93.58	90.94
QNLI		87.62	89.45	89.42
QQP		82.29	83.50	87.49
MNLI		76.05	86.40	86.28

Table 5: **Private classification with soft prompts and prefix for $\varepsilon = 8$ and the RoBERTa$_{LARGE}$ model.** We use the same setup and notation as in Table 1.

E.2 PromptPATE

E.2.1 Hyperparameters for Confident-GNMax

We present our hyperparameters for Confident-GNMax in Table 7.

E.2.2 Dataset Preprocessing

sst2, trec, agnews, dbpedia and cb are taken from the repo of [58]. All other public datasets are downloaded from huggingface. To reduce the cost of quering APIs, we randomly sample 300 points from the test set to report the test accuracy. For imdb, we random select one sentence from each entry and also remove the
 tag. For qqp, we only take the column of "question 1" in the public set.

Dataset	Method	RoBERTa	BS	LR	ε	GRAD	Epochs	P-Length	Accuracy (%)
SST2	Prompt	Base	1024	0.005	∞	N/A	60	100	93.23
SST2	Prompt	Base	900	0.05	8	0.01	21	9	92.32
SST2	Prompt	Base	1024	0.005	3	0.05	100	10	86.35
SST2	Prompt	Large	2048	0.005	8	4	100	10	90.37
SST2	Prefix	Base	32	0.01	∞	N/A	60	20	94.61
SST2	Prefix	Base	1000	0.05	8	4	22	1	91.97
SST2	Prefix	Base	1024	0.01	3	0.2	100	50	90.37
SST2	Prefix	Large	2048	0.05	8	4	22	1	93.58
QNLI	Prompt	Base	1024	0.005	∞	N/A	60	128	89.48
QNLI	Prompt	Base	1024	0.005	8	0.05	100	10	84.11
QNLI	Prompt	Base	1024	0.005	3	0.1	100	50	83.62
QNLI	Prompt	Large	2048	0.01	8	0.05	100	10	87.62
QNLI	Prefix	Base	1024	0.005	∞	N/A	60	20	94.84
QNLI	Prefix	Base	1000	0.03	8	0.07	22	10	88.77
QNLI	Prefix	Base	1024	0.01	3	0.2	100	50	85.78
QNLI	Prefix	Large	2048	0.03	8	0.07	22	10	89.45
QQP	Prompt	Base	1024	0.005	∞	N/A	60	50	86.64
QQP	Prompt	Base	1024	0.05	8	0.1	10	7	82.58
QQP	Prompt	Base	1024	0.001	3	0.01	100	15	80.29
QQP	Prompt	Large	2048	0.005	8	0.05	100	10	82.29
QQP	Prefix	Base	1024	0.005	∞	N/A	60	20	91.42
QQP	Prefix	Base	1024	0.05	8	0.1	10	7	82.59
QQP	Prefix	Base	1024	0.05	3	1	15	2	80.89
QQP	Prefix	Large	2048	0.05	8	0.1	10	7	83.50
MNLI	Prompt	Base	32	0.001	∞	N/A	60	20	82.49
MNLI	Prompt	Base	1024	0.005	8	0.05	60	10	75.01
MNLI	Prompt	Base	1024	0.005	3	0.05	100	10	73.97
MNLI	Prompt	Large	2048	0.005	8	0.2	60	10	76.05
MNLI	Prefix	Base	32	0.001	∞	N/A	60	20	82.49
MNLI	Prefix	Base	1024	0.005	8	0.05	60	50	80.42
MNLI	Prefix	Base	1024	0.005	3	0.2	100	50	80.10
MNLI	Prefix	Large	2048	0.01	8	0.1	100	10	86.40

Table 6: **Detailed parameters for soft prompts and prefix.** Type is the type of training, BS represents the batch size, LR denotes the learning rate, ε is the DP guarantee, P-Length is the token length of soft-prompt or prefix.

LLM	Dataset	T	σ_1	σ_2
GPT3	sst2	180	1	20
GPT3	agnews	180	5	20
GPT3	trec	180	1	20
GPT3	dbpedia	170	1	20
Claude	sst2	390	1	50
Claude	agnews	360	1	50
Claude	trec	320	1	50
Claude	dbpedia	320	5	50

Table 7: **Detailed parameters for Confident-GNMax.**