PRIVATE LANGUAGE MODELS VIA TRUNCATED LAPLACIAN MECHANISM

TIANHAO HUANG*, TAO YANG*, IVAN HABERNAL,
LIJIE HU, AND DI WANG†

Corresponding to di.wang@kaust.edu.sa

INTRODUCTION

In this paper, we propose a novel private embedding method called the high dimensional truncated Laplacian mechanism. Specifically, we introduce a non-trivial extension of the truncated Laplacian mechanism, which was previously only investigated in one-dimensional space cases. Theoretically, we show that our method has a lower variance compared to the previous private word embedding methods. To further validate its effectiveness, we conduct comprehensive experiments on private embedding and downstream tasks using three datasets. Remarkably, even in the high privacy regime, our approach only incurs a slight decrease in utility compared to the non-private scenario.

Comparison of Private Embedding

Original:	Oh and we came on a Saturday night around	11:30	for	context.	(→Privacy Leakage)
Trlaplace:	Oh and we came on a Saturday night around	9:30pm	for	<unk>	(→Private and Fluent)
Laplace:	Oh and we came on a Saturday night around	around	for	<unk>	(→Semantic Problem)
Gaussian:	Oh and we came on a Saturday night around	11:30	for	<unk>	(→Privacy Leakage)

An example of (private) text re-write for different mechanisms with $\epsilon = 0.1$.

PRIVATE EMBEDDING VIA TRUNCATED LAPLACIAN MECHANISM

Generally speaking, for each token w_i, to achieve DP, our approach consists of three steps. First, each token w_i is mapped to an d-dimensional pre-trained word embedding $\phi(w_i)$. And we perform a clipping step to get a clipped embedding:

$$\text{CLIPEmb}(w_i) = \phi(w_i) \min\{1, \frac{C}{\|\phi(w_i)\|_2}\}, \tag{1}$$

where the threshold $C > 0$ is a hyper-parameter. In the second step, we add some random noise to the clipped embedding vector to make it satisfy DP. Finally, we will perform the projection step by finding the nearest word $\hat{w}_i$ to the perturbed and clipped embedding vector within the embedding space:

$$\hat{w}_i = \arg\min_{w \in \mathcal{W}} \|\phi(w) - \text{CLIPEmb}(w_i) - \eta\|_2, \tag{2}$$

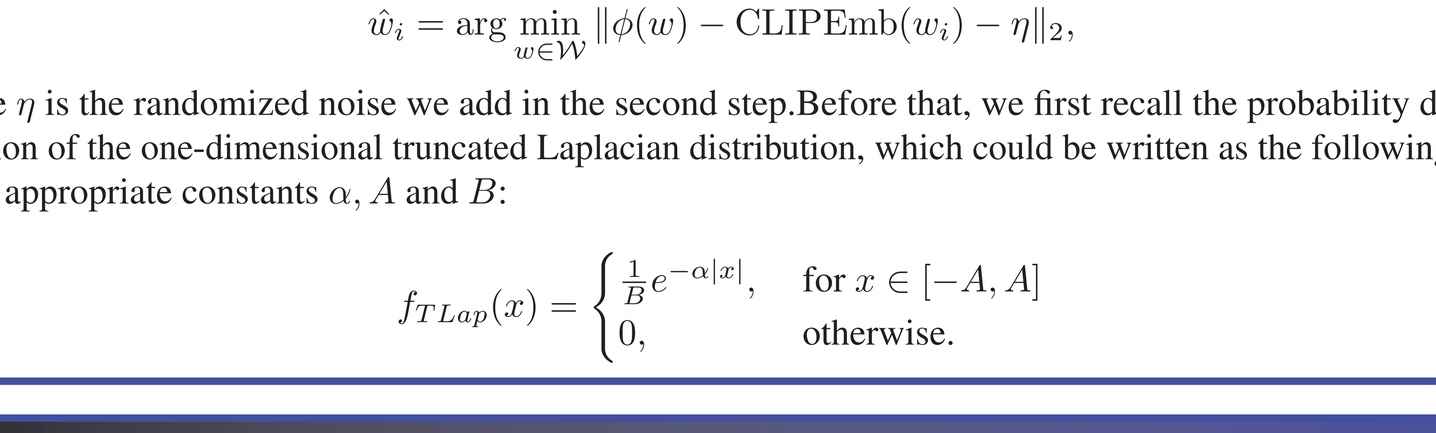
where η is the randomized noise we add in the second step.Before that, we first recall the probability density function of the one-dimensional truncated Laplacian distribution, which could be written as the following with some appropriate constants α, A and B:

$$f_{TLap}(x) = \begin{cases} \frac{1}{B} e^{-\alpha|x|}, & \text{for } x \in [-A, A] \\ 0, & \text{otherwise.} \end{cases} \tag{3}$$

PRIVACY TEST

Performance under GloVe for the non-private case and the three mechanisms, where the privacy budget ranges from 0.05 to 0.5.

		Original	Gaussian				Laplacian				TrLaplacian			
Privacy budget ϵ		∞	0.05	0.1	0.2	0.5	0.05	0.1	0.2	0.5	0.05	0.1	0.2	0.5
	Loss↓	2.95	51.25	26.66	9.92	5.97	51.43	37.86	15.35	7.31	**2.89**	**2.86**	**2.84**	**3.04**
	Rouge1↑	92.37	14.01	59.52	83.61	89.06	13.02	43.30	75.77	86.98	**92.44**	**92.43**	**92.41**	**92.25**
Yahoo	**BLEU↑**	8.501	**9.286**	8.418	8.489	8.499	9.132	8.287	8.474	8.493	8.499	**8.500**	**8.497**	**8.504**
	N_w ↑	0.703	0.072	0.511	0.595	0.628	0.066	0.334	0.566	0.642	**0.706**	**0.682**	**0.666**	**0.662**
	BERT-S↑	0.975	0.849	0.908	0.955	0.963	0.839	0.889	0.942	0.959	**0.976**	**0.971**	**0.971**	**0.971**
	Loss↓	3.07	34.67	21.62	10.61	5.98	36.00	34.64	14.86	7.38	**2.98**	**2.99**	**3.02**	**2.94**
	Rouge1↑	89.40	15.97	48.89	76.48	84.97	12.60	14.68	66.62	82.08	**89.45**	**89.47**	**89.34**	**89.54**
Yelp	**BLEU↑**	8.934	**8.976**	8.850	8.926	8.930	8.607	8.916	8.913	8.928	8.931	**8.935**	**8.936**	**8.936**
	N_w ↑	0.706	0.144	0.381	0.608	0.694	0.052	0.138	0.525	0.646	**0.705**	**0.721**	**0.722**	**0.725**
	BERT-S↑	0.973	0.874	0.895	0.943	0.964	0.855	0.874	0.927	0.952	**0.971**	**0.973**	**0.971**	**0.972**



Curves of Loss, Rouge1 and BERTScore with different privacy budget ϵ for Yelp datasets.

PRELIMINARIES

Definition 1. *Given a domain of dataset $\mathcal{X}$. A randomized algorithm $\mathcal{A} : \mathcal{X} \mapsto \mathcal{R}$ is (ε, δ)-differentially private (DP) if for all adjacent datasets S, S' with each sample is in $\mathcal{X}$ and for all $T \subseteq \mathcal{R}$, the following holds*

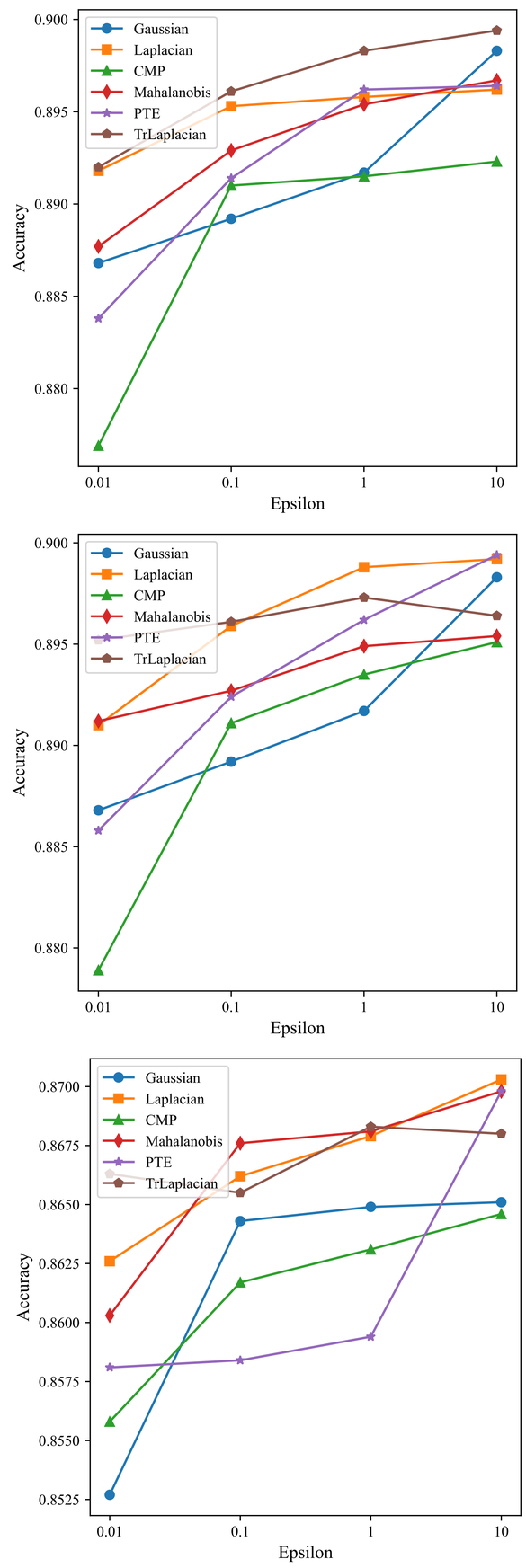
$$\Pr(\mathcal{A}(S) \in T) \leq \exp(\varepsilon) \Pr(\mathcal{A}(S') \in T) + \delta.$$

When $\delta = 0$, we call the algorithm $\mathcal{A}$ is ε-DP.

Definition 2. *Given a discrete vocabulary $\mathcal{W}$, a randomized algorithm $\mathcal{A} : \mathcal{W} \mapsto \mathcal{R}$ is word-level (ϵ, δ)-differentially private (DP) if for all pair of words $w, w' \in \mathcal{W}$ and for all $T \subseteq \mathcal{R}$ we have $\mathbb{P}(\mathcal{A}(w) \in T) \leq e^{\epsilon} \mathbb{P}(\mathcal{A}(w') \in T) + \delta$. When $\delta = 0$, we call the algorithm $\mathcal{A}$ is ϵ-DP.*

In this paper, we assume the user holds a sentence $s = w_1 w_2 \cdots w_n$ with n words. And we aim to design an (ϵ, δ)-DP algorithm, which is private w.r.t. each word w_i.

UTILITY TEST



Classification accuracy results for private fine-tuning across various embeddings and privacy levels.