Paper Review

Large Language Models Can Be Strong Differentially Private Learners

ICLR 2022

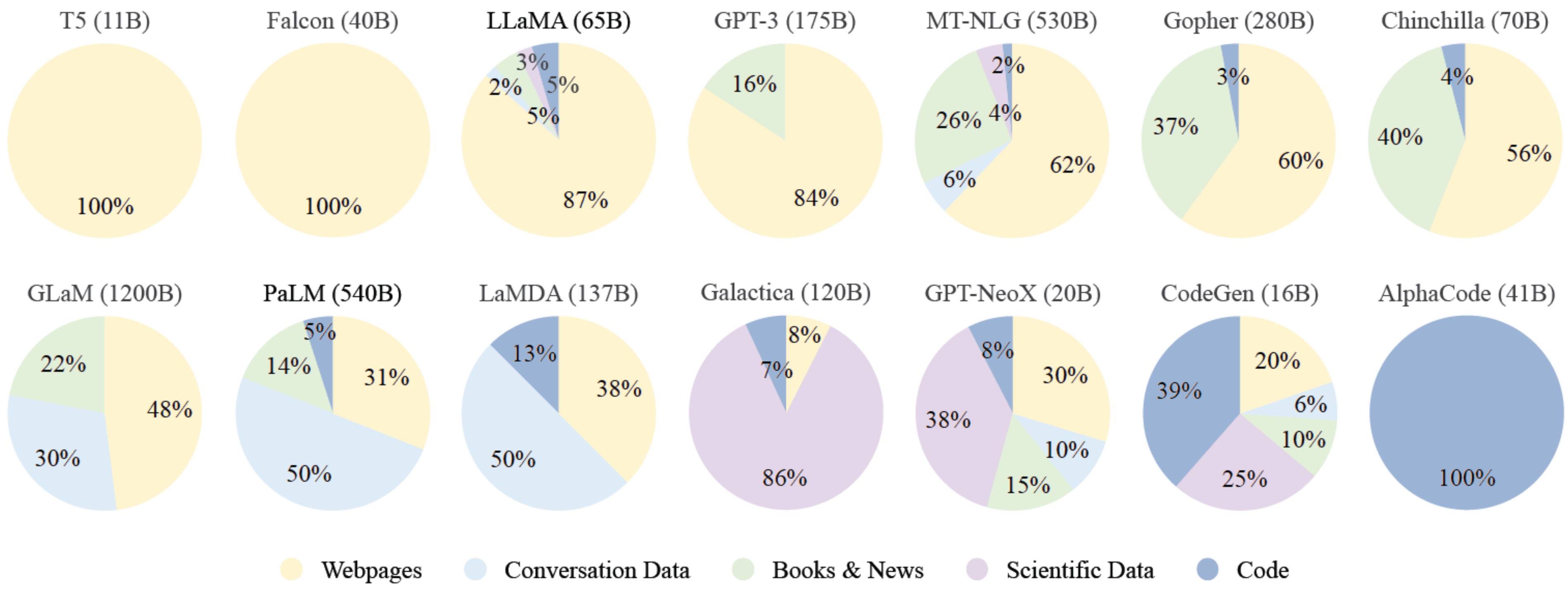
Xuechen Li[1], Florian Tramer[2], Percy Liang[1], Tatsunori Hashimoto[1]

[1]Stanford University [2]Google Research

Background

- **Privacy Preserving Deep Learning**
  - Data privacy guarantee for Large Language Models (LLM)
  - Privacy leakage from training data

Ratios of pre-training data source for LLM



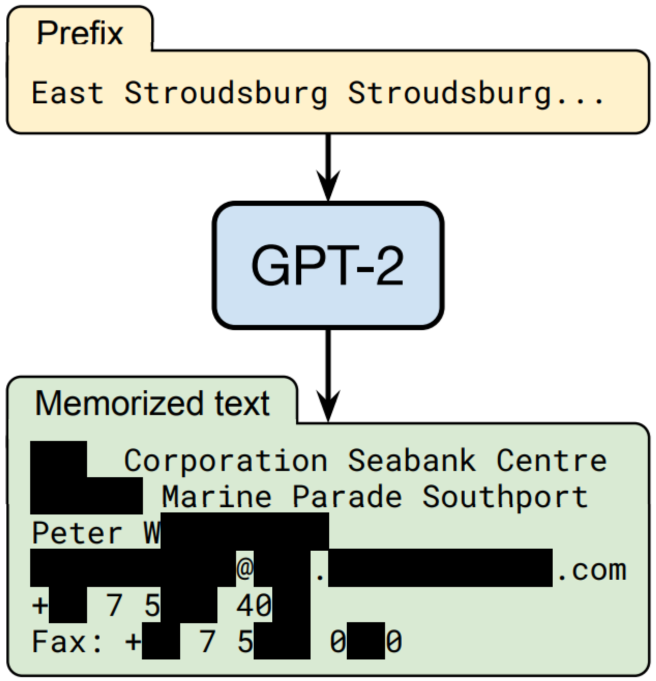
Wayne Xin Zhao et al. A Survey of Large Language Models. arXiv preprint arXiv:2303.18223, 2023.

Part 1.

Background

- **Privacy Attack**
  - Simulate the scenario of training data extraction attack
  - Language model memorization

Training data extraction attack



Categorization of memorized training examples

Category	Count
US and international news	109
Log files and error reports	79
License, terms of use, copyright notices	54
Lists of named items (games, countries, etc.)	54
Forum or Wiki entry	53
Valid URLs	50
Named individuals (non-news samples only)	46
Promotional content (products, subscriptions, etc.)	45
High entropy (UUIDs, base64 data)	35
Contact info (address, email, phone, twitter, etc.)	32
Code	31
Configuration files	30
Religious texts	25
Pseudonyms	15
Donald Trump tweets and quotes	12
Web forms (menu items, instructions, etc.)	11
Tech news	11
Lists of numbers (dates, sequences, etc.)	10

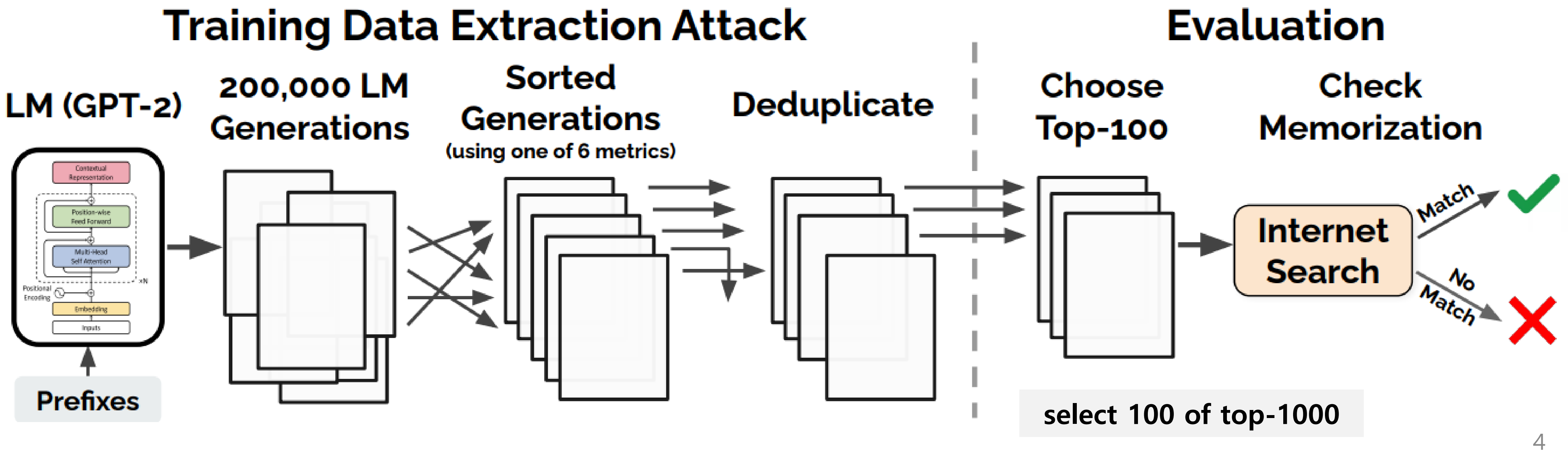
Nicholas Carlini et al. Extract Training Data from Large Language Models. USENIX Security, 2021.

Part 1.

Background

- **Privacy Attack**
  - Attack
    - Generate 256 tokens by one of sampling
    - Sort generations by one of inference metrics
  - Evaluation
    - Identify 604 unique memorized examples in total

Inference Strategy	Text Generation Strategy		
	Top-n	Temperature	Internet
Perplexity	9	3	39
Small	41	42	58
Medium	38	33	45
zlib	59	46	67
Window	33	28	58
Lowercase	53	22	60
Total Unique	191	140	273



Nicholas Carlini et al. Extract Training Data from Large Language Models. USENIX Security, 2021.

Part 1.

Background

- **Privacy Attack**
  - Training data extraction attack
  - Language model memorization

Definition 1 (Model Knowledge Extraction) *A string s is* extractable[4] *from an LM f_θ if there exists a prefix c such that:*

$$s \leftarrow \underset{s':\,|s'|=N}{\arg\max} f_\theta(s' \mid c)$$

Definition 2 (k-Eidetic Memorization) *A string s is k-eidetic memorized (for $k \geq 1$) by an LM f_θ if s is extractable from f_θ and s appears in at most k examples in the training data X: $|\{x \in X : s \subseteq x\}| \leq k$.*

Examples of $k = 1$ eidetic memorized, high entropy content that we extract

Memorized String	Sequence Length	Occurrences in Data	
		Docs	Total
Y2...[redacted]...y5	87	1	10
7C...[redacted]...18	40	1	22
XM...[redacted]...WA	54	1	36
ab...[redacted]...2c	64	1	49
ff...[redacted]...af	32	1	64
C7...[redacted]...ow	43	1	83
0x...[redacted]...C0	10	1	96
76...[redacted]...84	17	1	122
a7...[redacted]...4b	40	1	311

String Format: UUID (Universally Unique Identifier)

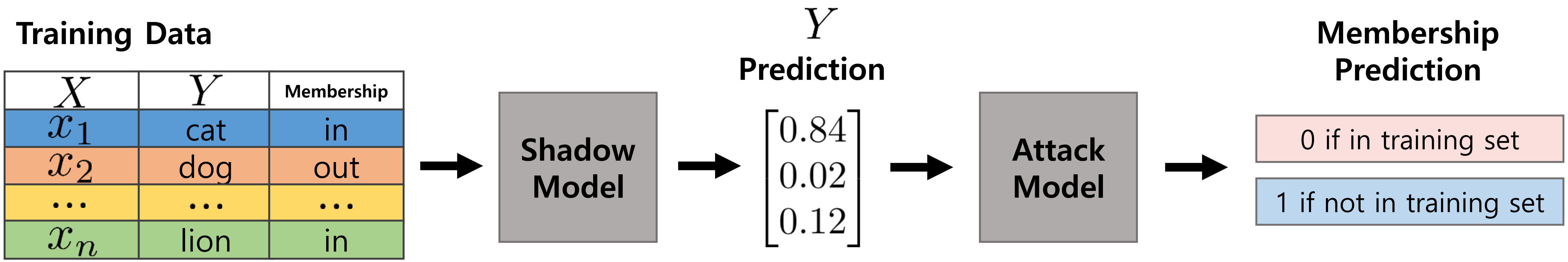
- 32 Hexadecimal numbers
- 5 Group separated by hypen(-)

Nicholas Carlini et al. Extract Training Data from Large Language Models. USENIX Security, 2021.

Part 1.

Background

- **Membership Inference Attack**
  - Shadow training
  - Baseline model provided by Machine Learning as a Service (MLaaS)



Experiment on Google-trained models

Model	*Dataset*	*Training Accuracy*	*Testing Accuracy*	*Attack Precision*
MLP	Adult	0.848	0.842	0.503
CNN	MNIST	0.984	0.928	0.517
MLP	Location	1.000	0.673	0.678
	Purchase (2)	0.999	0.984	0.505
	Purchase (10)	0.999	0.866	0.550

Model	*Dataset*	*Training Accuracy*	*Testing Accuracy*	*Attack Precision*
MLP	Purchase (20)	1.000	0.781	0.590
	Purchase (50)	1.000	0.693	0.860
	Purchase (100)	0.999	0.659	0.935
	TX hospital stays	0.668	0.517	0.657

Reza Shokri et al. Membership Inference Attacks Against Machine Learning Models. IEEE Symposium on Security and Privacy, 2017.

Background

- **Differential Privacy (DP)**
  - Deep learning adopts DP algorithm for data privacy guarantee
  - Quantify the amount of privacy
    - Privacy disclosed about individual records by the output of a valid computation
  - Data analysis
    - Can mine aggregated personal data with provable guarantees of privacy for individuals

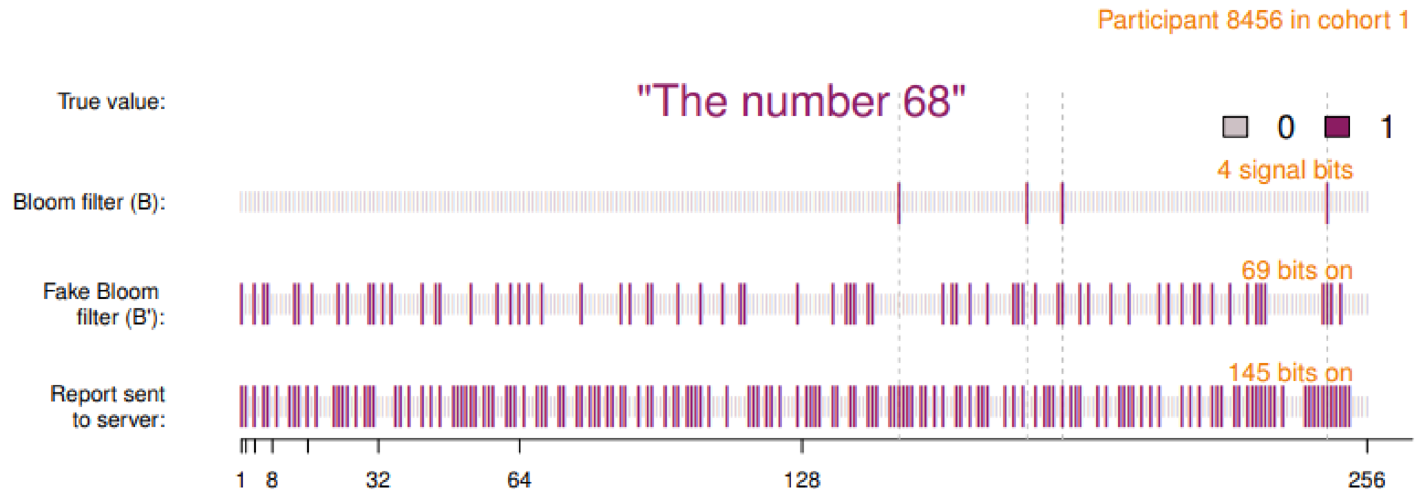
How to prevent the disclosure of private data

Measure	Purpose	Approach
Statistical Disclosure Control (SDC)	Guarantee data privacy in statistical field	Data generalization and anonymization
Computational Disclosure Control (CDC)	Data security and acess control in database system	Encryption, access control, data masking
Inference Control	Minimize disclosure of personal information during data analysis	Noising, query response distortion, data sampling

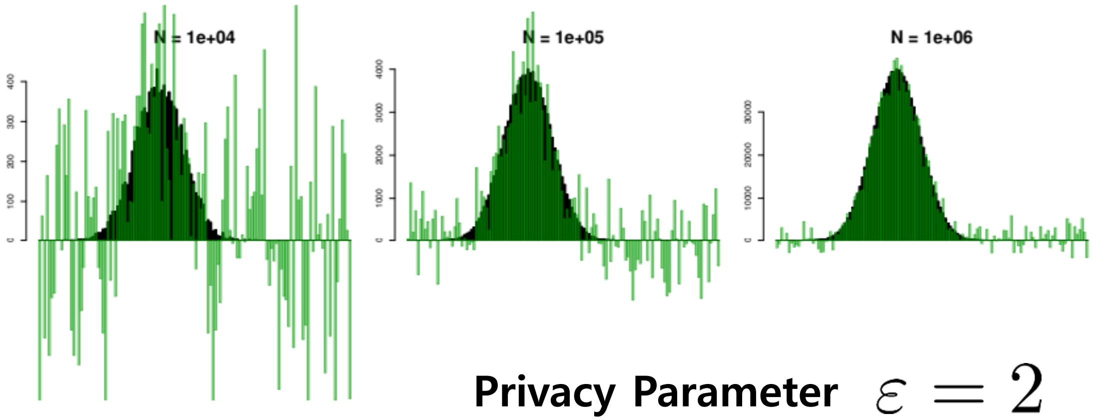
Cynthia Dwork et al. Calibrating noise to sensitivity in private data analysis. In Theory of cryptography conference, 2006.
Ross Anderson. Security Engineering — Third Edition. Wiley, 2020.
Ruobin Gong et al. Anco Hundepool. Differential Privacy for the 2020 Census: How Can We Make Data Both Private and Useful?. Special Issue 2: Differential Privacy for the 2020 U.S. Census, 2022.
Latanya Sweeney. Computational disclosure control : a primer on data privacy protection. Massachusetts Institute of Technology, 2001.

Background

- **Case of Differential Privacy (DP)**
  - Google RAPPOR (Privacy-Preserving Aggregatable Randomized Response, 2014)
    - Learning about the actual client's value v is even harder for attacker because multiple values map to the same bits in the Bloom filter
    - Attack difficulty caused by uncertainty of RAPPOR's estimated counts
  - Google uses better algorithm extending and strengthening previous work (e.g., RAPPOR)



Add noise to original data using randomized response



Privacy Parameter $\varepsilon = 2$

Cynthia Dwork et al. Calibrating noise to sensitivity in private data analysis. In Theory of cryptography conference, 2006.
Úlfar Erlingsson et al. RAPPOR: Randomized Aggregatable Privacy-Preserving Ordinal Response. ACM CCS , 2014
Andrea Bittau et al. Prochlo: Strong Privacy for Analytics in the Crowd. CoRR abs/1710.00901, 2017.

Part 1.

Background

- **Differential Privacy (DP)**
  - A mechanism A guarantees ε-differential privacy if for any pair of neighboring datasets X and X', A gives similar results t with probability

ε -differential privacy

$$\left| \ln\left(\frac{\Pr[\mathcal{T}_{\mathcal{A}}(\mathbf{x}) = t]}{\Pr[\mathcal{T}_{\mathcal{A}}(\mathbf{x}') = t]} \right) \right| \leq \epsilon.$$

Neighboring Database

$$D = D' \pm t$$

  - Privacy Budget ↑
    - Attack Difficulty ↑
    - ε ↓
    - Noise Size ↑ Accuracy ↓

$$\begin{cases} D \\ D' \end{cases}$$

Attack Target **Attacker**

Hacker by Sunardi from Noun Project (CC BY 3.0)

Cynthia Dwork et al. Calibrating noise to sensitivity in private data analysis. In Theory of cryptography conference, 2006

Background

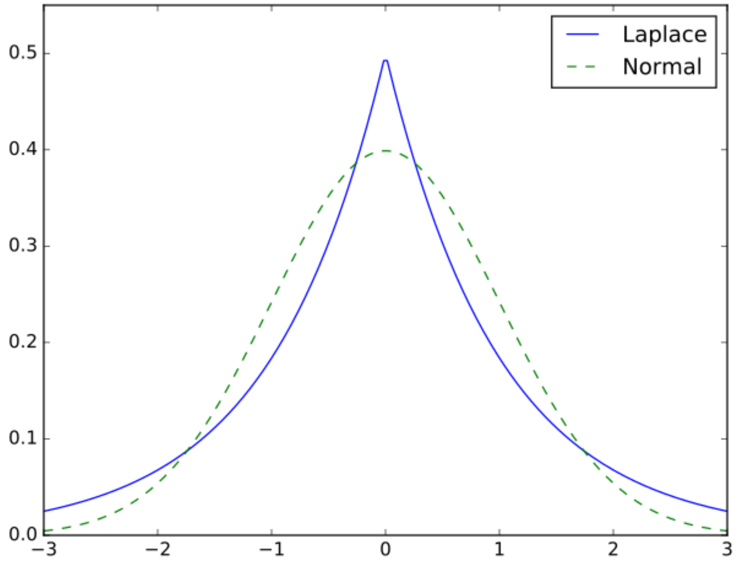
- **Differential Privacy (DP)**
  - Controlled noising mechanism to private data

$$\frac{\Pr(z+Y=t)}{\Pr(z'+Y=t)} \in \exp(\pm\frac{\|z-z'\|_1}{\lambda}). \qquad e^{\epsilon|f(\mathbf{x})-f(\mathbf{x}')|} \leq e^{\epsilon}$$

- According to Laplace Distribution $\quad Y \sim Lap(\frac{\Delta f}{\varepsilon})$
- Simplicity & Robustness

$$f(x \mid \mu, b) = \frac{1}{2b} e^{\left(\frac{|x-\mu|}{b}\right)}$$

$$\because \mu = 0,\ \sigma = \lambda,\ b = 2(\frac{\Delta f}{\varepsilon})$$



- Global Sensitivity

$$\|f(\mathbf{x}) - f(\mathbf{x}')\|_1 \leq S(f)\ .$$

https://www.johndcook.com/blog/2019/02/05/normal-approximation-to-laplace-distribution/
Cynthia Dwork et al. Calibrating noise to sensitivity in private data analysis. In Theory of cryptography conference, 2006

Background

- **Deep Learning with DP**
  - Differentially Private Stochastic Gradient Descent (DP-SGD)
    - Add Gaussian noise to gradients for individual training examples

---

Algorithm 1 Differentially private SGD (Outline)

Input: Examples $\{x_1, \ldots, x_N\}$, loss function $\mathcal{L}(\theta) = \frac{1}{N}\sum_i \mathcal{L}(\theta, x_i)$. Parameters: learning rate η_t, noise scale σ, group size L, gradient norm bound C.

Initialize θ_0 randomly

for $t \in [T]$ **do**

Take a random sample L_t with sampling probability L/N

Compute gradient

For each $i \in L_t$, compute $\mathbf{g}_t(x_i) \leftarrow \nabla_{\theta_t}\mathcal{L}(\theta_t, x_i)$

Clip gradient

$\bar{\mathbf{g}}_t(x_i) \leftarrow \mathbf{g}_t(x_i) / \max\left(1, \frac{\|\mathbf{g}_t(x_i)\|_2}{C}\right)$

Add noise

$\tilde{\mathbf{g}}_t \leftarrow \frac{1}{L}\left(\sum_i \bar{\mathbf{g}}_t(x_i) + \mathcal{N}(0, \sigma^2 C^2 \mathbf{I})\right)$

Descent

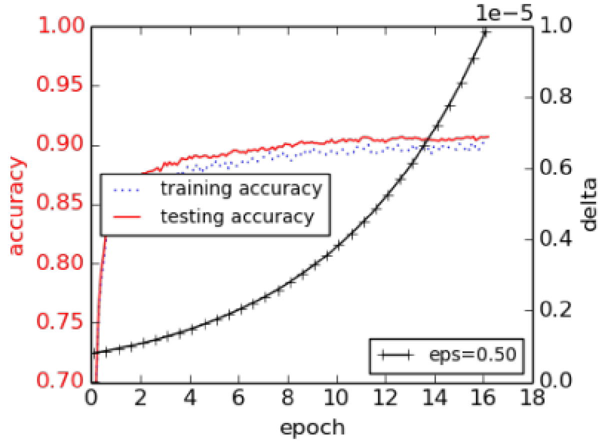
$\theta_{t+1} \leftarrow \theta_t - \eta_t \tilde{\mathbf{g}}_t$

Output θ_T and compute the overall privacy cost (ε, δ) using a privacy accounting method.

Noise clipping C

- Clip each gradient in ℓ_2 norm
- The number of Clipped batch L
- Add noise to several batches into a lot
- Then compute the average

Noise Level $\varepsilon = 0.5$



Model: LeNet-5

Dataset: MNIST

Y. LeCun et al. Gradient-based learning applied to document recognition. IEEE, 1998.
Martin Abadi et al. Deep learning with differential privacy. ACM SIGSAC, 2021.

Background

- **Large Language Model with DP-SGD**
  - DP optimization doesn't guarantee privacy-utility for large models' many parameters
  - The noise being isotropic in the high dimension of gradients

Number of trainable parameters

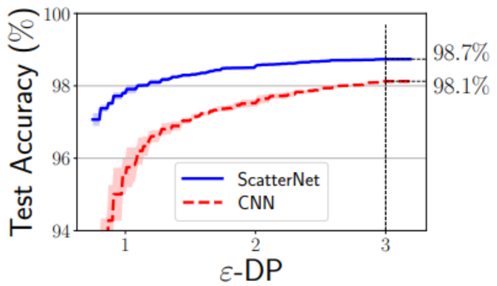
	MNIST & Fashion-MNIST	CIFAR-10
ScatterNet+Linear	40K	155K
ScatterNet+CNN	33K	187K
CNN	26K	551K / 168K

Accuracy on CIFAR-10

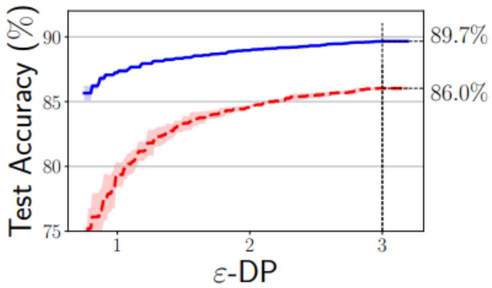
Model	Parameters	Accuracy
CNN	168K	60.7 ± 0.3
	551K	59.2 ± 0.1

Privacy Budget: $\varepsilon = 3, \delta = 10^{-5}$

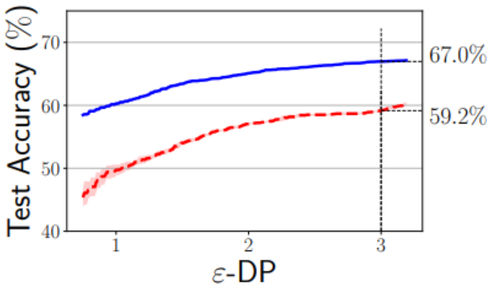
Accuracy for Privacy Budget: $\varepsilon, \delta = 10^{-5}$



(a) MNIST



(b) Fashion-MNIST



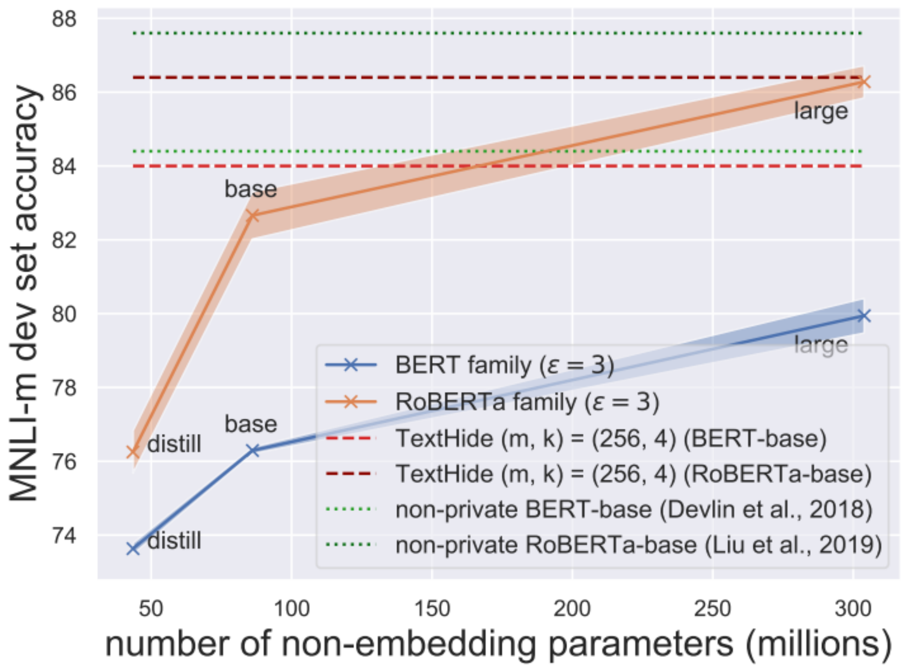
(c) CIFAR-10

Alex Krizhevsky et al. ImageNet Classification with Deep Convolutional Neural Networks. NeurIPS, 2012.
Edouard Oyallon and Stéphane Mallat. Deep Roto-Translation Scattering for Object Classifications. CVPR, 2015
Florian Tramer and Dan Boneh. Differentially Private Learning Needs Better Features (or Much More Data). ICLR, 2021.

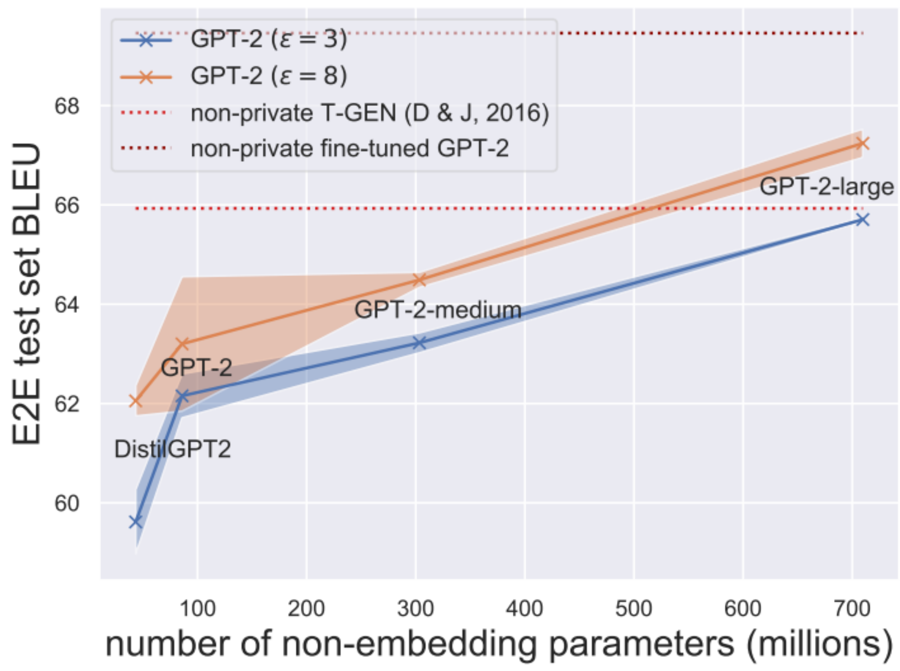
Introduction

- **Overview of our results**
  - For sentence classification, DP fine-tuning can outperform TextHide with BERT-base
    - TextHide is text encryption method tuned by heuristic privacy notions
  - For text generation, DP fine-tuning can outperform strong non-private baselines

Sentence classification on MNLI-matched



Natural language generation on E2E



Yangsibo Huang et al. TextHide: Tackling Data Privacy in Language Understanding Tasks. EMNLP, 2020.

Introduction

- **DP Fine-tuning**
  - Hyperparameter Tuning
    - Large batches lead to good performance
    - Effective Noise Multiplier σ_{eff} decreases according to this hyperparameter tuning
  - Ghost Clipping
    - This gradient norm can be computed efficiently for every example, since per-example gradients themselves need not be instantiated explicitly
  - Full Fine-tune Large Language Model with DP-Adam
    - Sentence Classification
      - Full fine-tuning with the text infilling objective outperforms other models
    - Table-To-Text Generation
      - Larger models has better performance than method optimizing few parameters
    - Chit-Chat Dialog Generation
      - Full fine-tuning with DP-Adam yields high quality competitive models

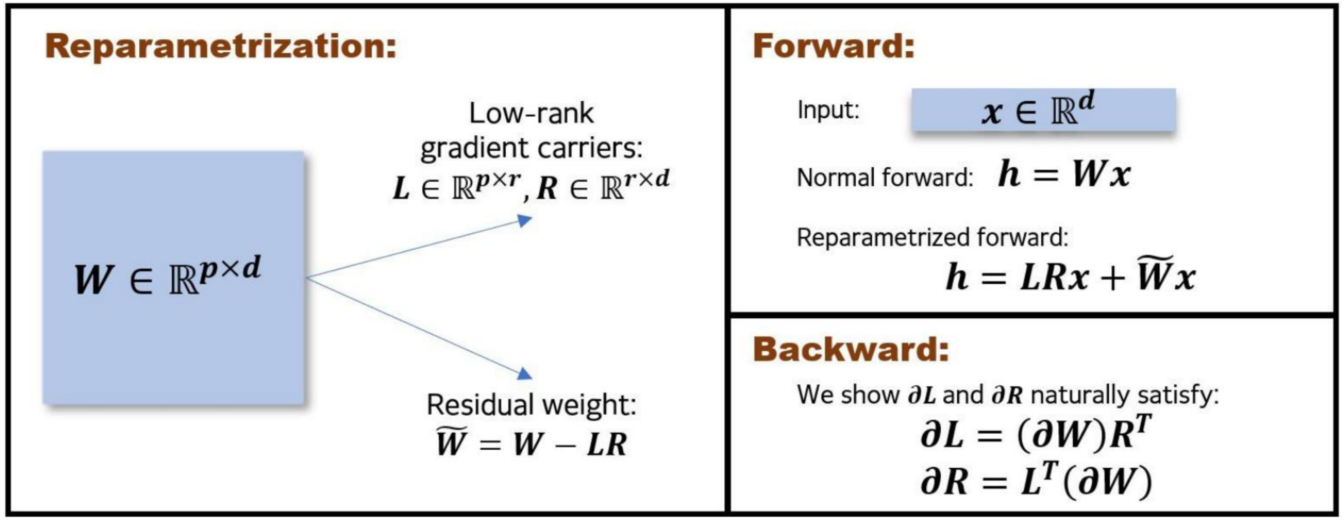
Introduction

- **DP Fine-tuning Task**
  - Sentence Classification
    - GLUE Benchmark
    - RGP (Reparametrized Gradient Perturbation)
      - Comparison DP-SGD Model
      - Reduce a mount of memory computing individual gradients

Table 1. Computation and memory costs of RGP (Algorithm 1) and DP-SGD (Abadi et al., 2016), where m is the size of mini-batch, d is the model width, r is the reparametrization rank, and K is the number of power iterations.

Method / Cost	DP-SGD	RGP
Computational cost	$\mathcal{O}(md^2)$	$\mathcal{O}(md^2 + Krd^2 + Kr^2d)$
Memory cost	$\mathcal{O}(md^2)$	$\mathcal{O}(mrd)$

Reparametrization scheme of RGP



Alex Wang et al. GLUE: A Multi-Task Benchmark and Analysis Platform for Natural Language Understanding. ICLR, 2019.
Da Yu et al. Large scale private learning via low-rank reparametrization. ICML, 2021c.

Introduction

- **DP Fine-tuning Task**
  - Table-To-Text Generation
    - BLEU & ROUGE-L
    - E2E Dataset
      - Crowdsourced dataset of 50k instances in the restaurant domain

Data format

Flat MR	NL reference
name[Loch Fyne], eatType[restaurant], food[French], priceRange[less than £20], familyFriendly[yes]	Loch Fyne is a family-friendly restaurant providing wine and cheese at a low cost. Loch Fyne is a French family friendly restaurant catering to a budget of below £20. Loch Fyne is a French restaurant with a family setting and perfect on the wallet.

Jekaterina Novikova et al. The e2e dataset: New challenges for ˇ end-to-end generation. arXiv preprint arXiv:1706.09254, 2017.

Introduction

- **DP Fine-tuning Task**
  - Chit-Chat Dialog Generation
    - Chit-Chat Dialogue Model
      - Human-like Daily Talk
      - GPT-2, DialoGPT (e.g., ChatGPT)
    - Persona-Chat dataset
      - Provide person profile
      - Consistent personality
      - Next dialogue utterance

Data format

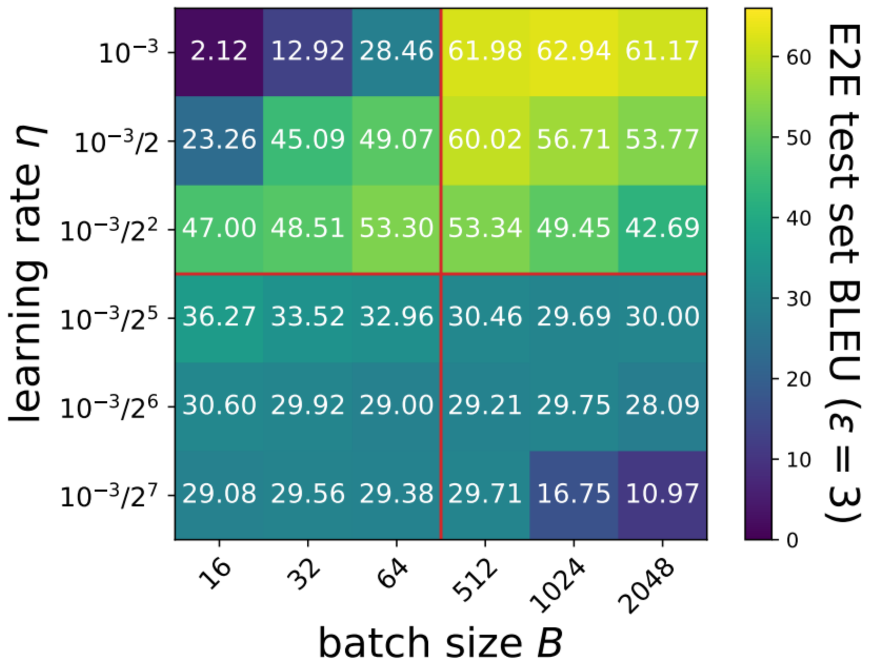
Persona 1	Persona 2
I like to ski	I am an artist
My wife does not like me anymore	I have four children
I have went to Mexico 4 times this year	I recently got a cat
I hate Mexican food	I enjoy walking for exercise
I like to eat cheetos	I love watching Game of Thrones

[PERSON 1:] Hi
[PERSON 2:] Hello ! How are you today ?
[PERSON 1:] I am good thank you , how are you.
[PERSON 2:] Great, thanks ! My children and I were just about to watch Game of Thrones.
[PERSON 1:] Nice ! How old are your children?
[PERSON 2:] I have four that range in age from 10 to 21. You?
[PERSON 1:] I do not have children at the moment.
[PERSON 2:] That just means you get to keep all the popcorn for yourself.
[PERSON 1:] And Cheetos at the moment!
[PERSON 2:] Good choice. Do you watch Game of Thrones?
[PERSON 1:] No, I do not have much time for TV.
[PERSON 2:] I usually spend my time painting: but, I love the show.

Saizheng Zhang et al. Personalizing dialogue agents: I have a dog, do you have pets too? arXiv preprint arXiv:1801.07243, 2018.
Yizhe Zhang et al. Dialogpt: Large-scale generative pre-training for conversational response generation. ACL, 2020..

Methodology

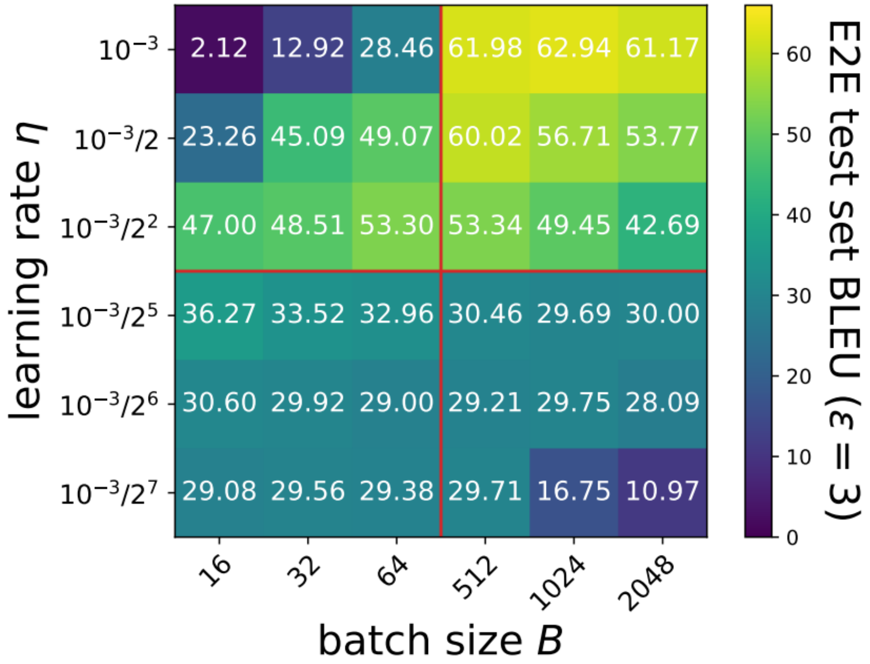
- **Batch Size, Learning Rate**
  - Private Learning
    - Fine-tune GPT-2 on E2E for table-to-text generation with DP-Adam at $\varepsilon = 3$
    - Numbers are BLEU scores on the test split of E2Es
  - General case of non-private Learning
    - LLM is typically fine-tuned with small batch sizes and learning rates with Adam

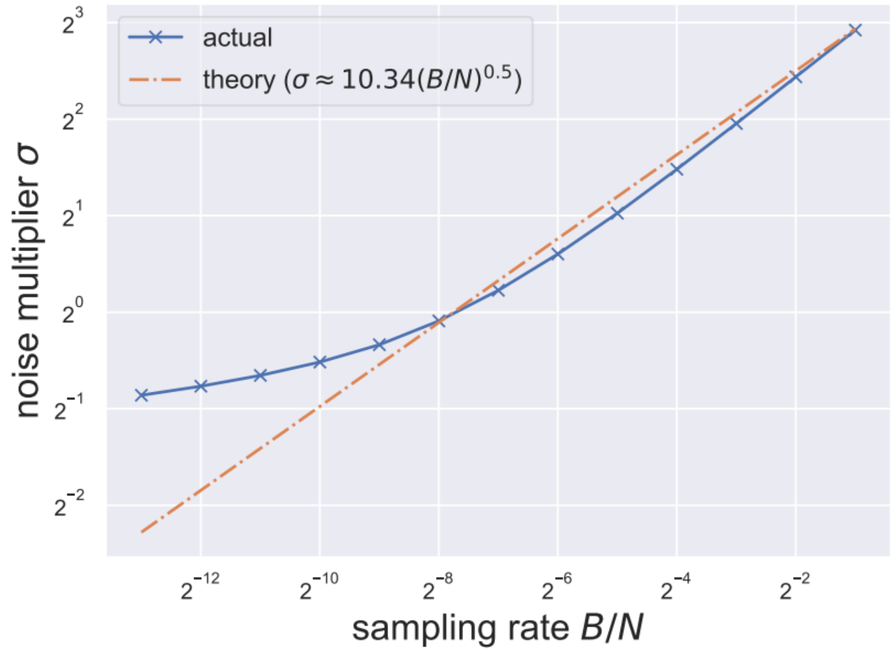


Methodology

- **Batch Size, Learning Rate**
  - Linear scaling rule for private learning
    - This rule does not generalize to batch sizes that are too small
    - Square-root relationship underestimates the noise multiplier for small batch sizes

Linear Scaling Rule: *When the minibatch size is multiplied by k, multiply the learning rate by k.*



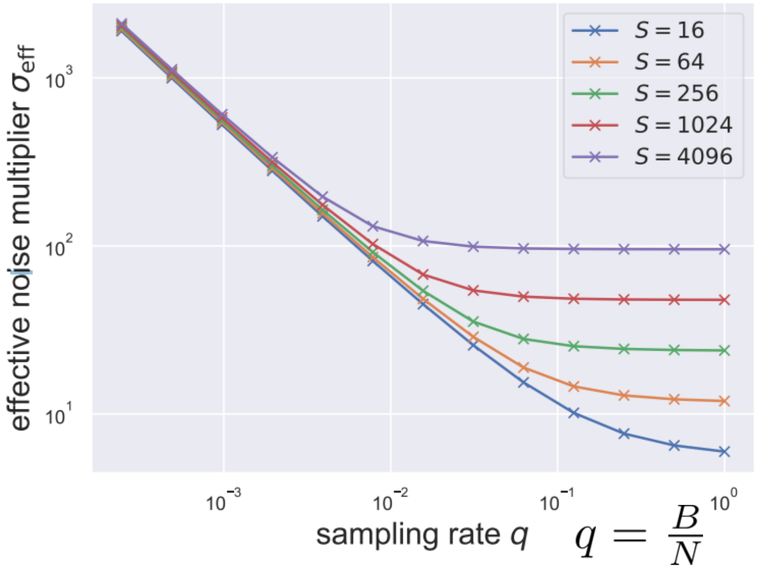


Priya Goyal et al. Accurate, Large Minibatch SGD: Training ImageNet in 1 Hour. NeurIPS, 2017.

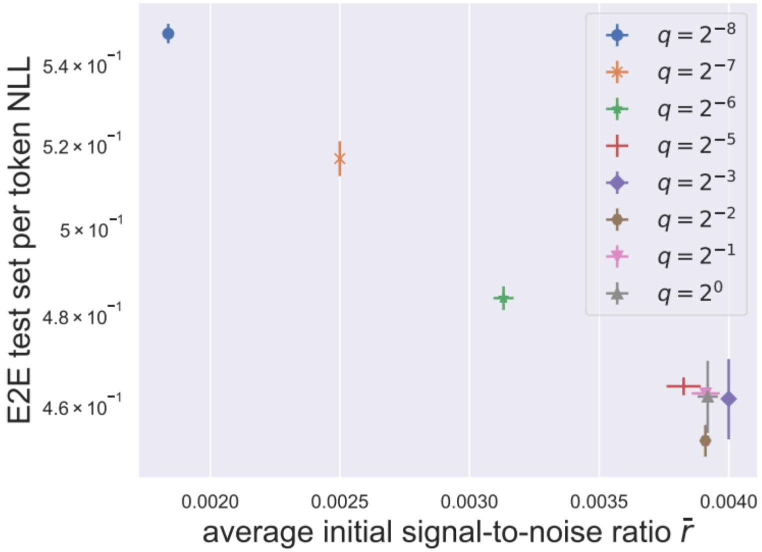
Methodology

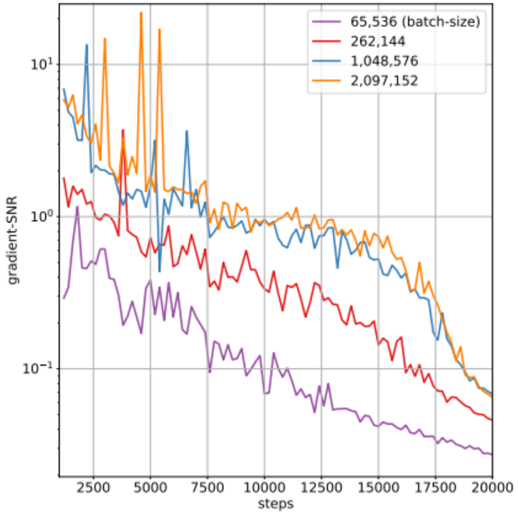
- **Batch Size, Learning Rate**
  - Increasing the batch size allows us to improve gradient-SNR
  - Expanded initial SNR leads to faster convergence of DP training
  - Effective Noise Multiplier $\sigma_{eff} = \frac{\sigma}{q} = \frac{\sigma N}{B}$
  - Signal-to-Noise Ratio $r = \|\tilde{g}\|_2 / \|\bar{z}\|_2$
    - Privacy budget $\bar{g}$ in DP-SGD/DP-Adam

$$\bar{g} = \tilde{g} + \bar{z}, \quad \tilde{g} = \frac{1}{B} \sum_{i \in \mathcal{B}} \mathrm{Clip}\left(\nabla \mathcal{L}_i, C\right), \quad \bar{z} \sim \mathcal{N}\left(0, C^2 \frac{\sigma^2}{B^2} I_p\right) = \mathcal{N}\left(0, C^2 \frac{\sigma_{\mathrm{eff}}^2}{N^2} I_p\right)$$



$q = \frac{B}{N}$





(a) Gradient-SNR

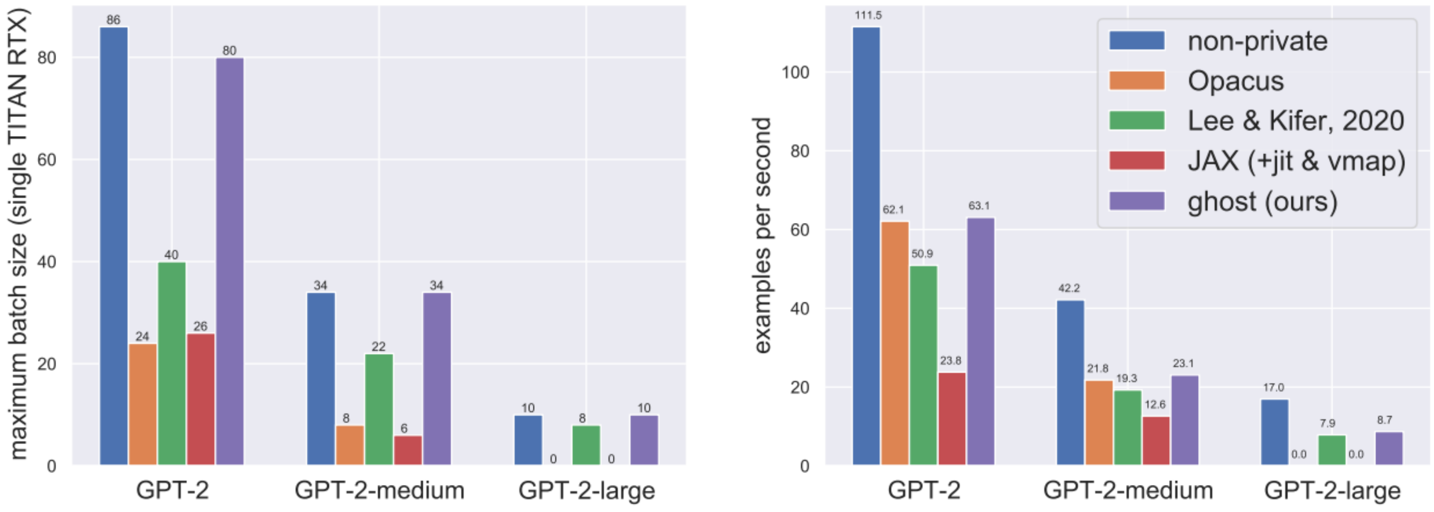
Rohan Anil et al. Large-scale differentially private BERT. EMNLP Findings, 2022.

Methodology

- **Ghost Clipping**
  - Memory saving technique that allows clipping without per-example gradients
  - Extend the Lee & Kifer (2020) by generalization of the Goodfellow (2015) trick

$$\|\nabla_W \mathcal{L}_i\|_F^2 = \text{vec}(a_i a_i^\top)^\top \text{vec}(g_i g_i^\top).$$

  - Allows fitting batches almost as large as those in non-private training



(a) Memory (b) Throughput

Clipping process

- Clip each gradient in ℓ_2 norm
- Add noise to several batches
- Then compute the average

Jaewoo Lee and Daniel Kifer. Scaling up differentially private deep learning with fast per-example gradient clipping. arXiv preprint arXiv:2009.03106, 2020.
Ian Goodfellow. Efficient per-example gradient computations. arXiv preprint arXiv:1510.01799, 2015.

Methodology

- **Ghost Clipping**
  - Extend the Lee & Kifer (2020) by generalization of the Goodfellow (2015) trick

$$\|\nabla_W \mathcal{L}_i\|_{\mathrm{F}}^2 = \mathrm{vec}(a_i a_i^\top)^\top \mathrm{vec}(g_i g_i^\top) = \|a_i\|_2^2 \, \|g_i\|_2^2 \, .$$

  - Efficient Per-Example Gradient Computations

Vanilla Gradient Norm → Goodfellow (2015) trick

Vanilla Gradient Norm

Loss function

$$L(\boldsymbol{z}^{(1)}, \ldots, \boldsymbol{z}^{(n)}, \boldsymbol{h}^{(0)}, \boldsymbol{y})$$

Gradient Norm

$$s_j^{(i)} = \sum_{k,l} \left(\frac{\partial}{\partial W_{k,l}^{(i)}} L^{(j)}\right)^2$$

Goodfellow (2015) trick

Neural Network

$$\boldsymbol{z}^{(i)} = \boldsymbol{h}^{(i-1)\top} \boldsymbol{W}^{(i)}$$
$$\boldsymbol{h}^{(i)} = \phi^{(i)}(\boldsymbol{z}^{(i)}).$$

Gradient Norm

$$\boldsymbol{s}_j^{(i)} = \left(\sum_k (\bar{Z}_{j,k}^{(i)})^2\right)\left(\sum_k (H_{j,k}^{(i-1)})^2\right).$$

Jaewoo Lee and Daniel Kifer. Scaling up differentially private deep learning with fast per-example gradient clipping. arXiv preprint arXiv:2009.03106, 2020.
Ian Goodfellow. Efficient per-example gradient computations. arXiv preprint arXiv:1510.01799, 2015.

Methodology

- **Full Fine-tuning with DP-Adam**
  - Sentence Classification
  - Fine-tuning with text infilling objective
    - Instead of predicting integer labels, we ask the model to predict textualized labels
  - Per-update speed is 3 times slower than RGP

Dataset: GLUE

Method	$\epsilon = 3$				$\epsilon = 8$			
	MNLI-(m/mm)	QQP	QNLI	SST-2	MNLI-(m/mm)	QQP	QNLI	SST-2
RGP (RoBERTa-base)	-	-	-	-	80.5/79.6	85.5	87.2	91.6
RGP (RoBERTa-large)	-	-	-	-	86.1/86.0	86.7	90.0	93.0
full (RoBERTa-base)	82.47/82.10	85.41	84.62	86.12	83.30/83.13	86.15	84.81	85.89
full (RoBERTa-large)	85.53/85.81	**86.65**	88.94	90.71	86.28/86.54	**87.49**	89.42	90.94
full + infilling (RoBERTa-base)	82.45/82.99	85.56	87.42	91.86	83.20/83.46	86.08	87.94	92.09
full + infilling (RoBERTa-large)	**86.43/86.46**	86.43	**90.76**	**93.04**	**87.02/87.26**	87.47	**91.10**	**93.81**
$\epsilon \approx$ (Gaussian DP + CLT)	2.52	2.52	2.00	1.73	5.83	5.85	4.75	4.33
$\epsilon \approx$ (Compose tradeoff func.)	2.75	2.75	2.57	2.41	7.15	7.16	6.87	6.69

Da Yu et al. Large scale private learning via low-rank reparametrization. ICML, 2021c.

Methodology

- **Full Fine-tuning with DP-Adam**
  - Table-To-Text Generation
  - Full fine-tuning GPT-2 (125 million parameters)
  - Compared with parameter-efficient approaches
    - LoRA, prefix-tuning, RGP, and fine-tuning the top 2 Transformer blocks

Dataset: E2E

Metric	DP Guarantee	Gaussian DP + CLT	Compose tradeoff func.	Method: full	LoRA	prefix	RGP	top2	retrain
BLEU	$\epsilon = 3$	$\epsilon \approx 2.68$	$\epsilon \approx 2.75$	**61.519**	58.153	47.772	58.482	25.920	15.457
	$\epsilon = 8$	$\epsilon \approx 6.77$	$\epsilon \approx 7.27$	**63.189**	**63.389**	49.263	58.455	26.885	24.247
	non-private	-	-	69.463	69.682	68.845	68.328	65.752	65.731
ROUGE-L	$\epsilon = 3$	$\epsilon \approx 2.68$	$\epsilon \approx 2.75$	**65.670**	**65.773**	58.964	65.560	44.536	35.240
	$\epsilon = 8$	$\epsilon \approx 6.77$	$\epsilon \approx 7.27$	**66.429**	**67.525**	60.730	65.030	46.421	39.951
	non-private	-	-	71.359	71.709	70.805	68.844	68.704	68.751

Jekaterina Novikova et al. The e2e dataset: New challenges for end-to-end generation. arXiv preprint arXiv:1706.09254, 2017.
Da Yu et al. Large scale private learning via low-rank reparametrization. arXiv preprint arXiv:2106.09352, 2021c.
Edward J Hu et al. Lora: Low-rank adaptation of large language models. arXiv preprint arXiv:2106.09685, 2021.
Xiang Lisa Li and Percy Liang. Prefix-tuning: Optimizing continuous prompts for generation. arXiv preprint arXiv:2101.00190, 2021.

Methodology

- **Full Fine-tuning with DP-Adam**
  - Chit-Chat Dialog Generation
    - Predict the response with the dialog history and persona description
    - Distinct challenge that the response space is intrinsically diverse, since human conversations can be informal and noise

Dataset: Persona-Chat

Model	DP Guarantee	Gaussian DP +CLT	Compose tradeoff func.	Metrics		
				F1 ↑	Perplexity ↓	Quality (human) ↑
GPT-2	$\epsilon = 3$	$\epsilon \approx 2.54$	$\epsilon \approx 2.73$	15.90	24.59	-
	$\epsilon = 8$	$\epsilon \approx 6.00$	$\epsilon \approx 7.13$	16.08	23.57	-
	non-private	-	-	17.96	18.52	-
GPT-2-medium	$\epsilon = 3$	$\epsilon \approx 2.54$	$\epsilon \approx 2.73$	15.99	20.68	-
	$\epsilon = 8$	$\epsilon \approx 6.00$	$\epsilon \approx 7.13$	16.53	19.25	-
	non-private	-	-	18.64	15.40	-
DialoGPT-medium	$\epsilon = 3$	$\epsilon \approx 2.54$	$\epsilon \approx 2.73$	**17.37**	**17.64**	2.82 (2.56, 3.09)
	$\epsilon = 8$	$\epsilon \approx 6.00$	$\epsilon \approx 7.13$	**17.56**	**16.79**	3.09 (2.83, 3.35)
	non-private	-	-	19.28	14.28	3.26 (3.00, 3.51)
HuggingFace (ConvAI2 winner)	non-private	-	-	19.09	17.51	-
HuggingFace (our implementation)	non-private	-	-	16.36	20.55	3.23 (2.98, 3.49)
Reference	-	-	-	-	-	3.74 (3.49, 4.00)

Yizhe Zhang et al. Dialogpt: Large-scale generative pre-training for conversational response generation. ACL, 2020..

Conclusion

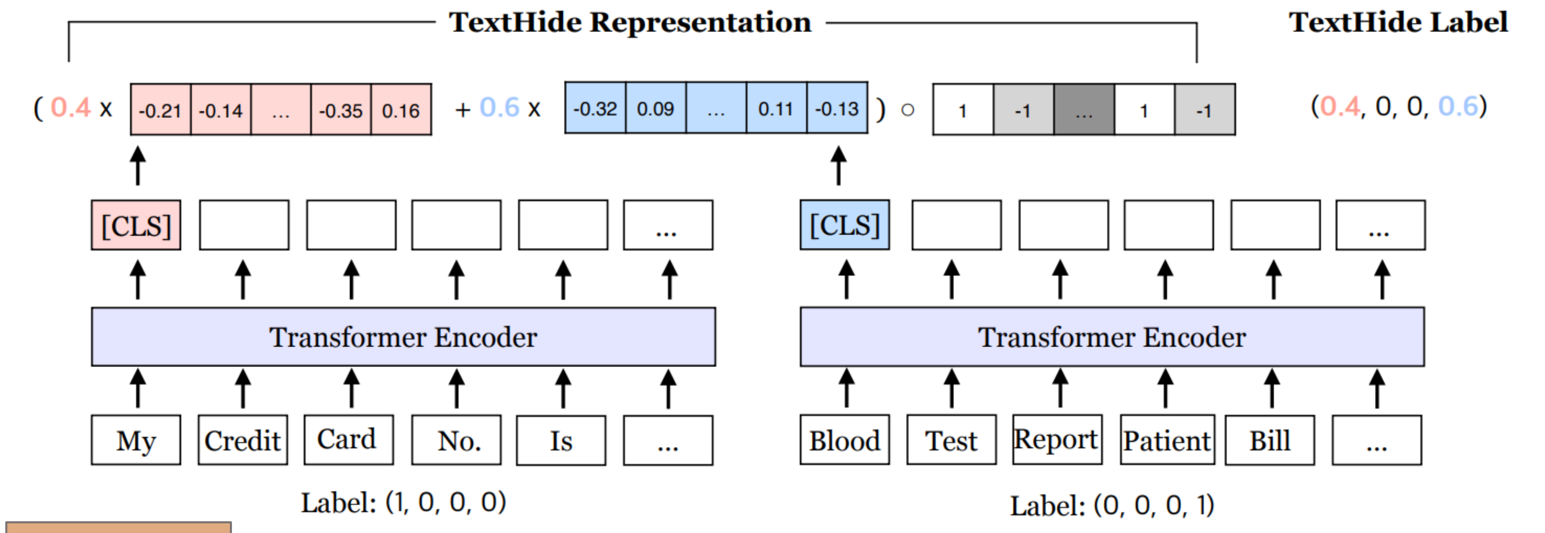
- **Full Fine-tuning Strategy with DP-Adam**
  - Larger models has competitive performance than method optimizing few parameters

- **Future Work**
  - Since DP fine-tuning generally requires substantially less private datas, we hope this will motivate organizations (e.g., federated learning with DP)

- **Limitation**
  - Should consider and create more curated public corpora for pretraining
  - Requires more transparency in reporting hyperparameter choices, analysis of hyperparameter transferability across tasks and architectures
  - Unaware of how the dimensionality of models (and pretraining) generally affect private deep learning

Appendix

- **TextHide**
  - Entry-wise mask is chosen from a randomly pre-generated pool and applied on the mixed representation
  - Training directly takes place on encrypted data and no decryption is needed
  - Attacker can't backpropagate the loss of batch through the secret mask of each client



Heuristic Method

Encryption with $k = 2$

Entry-wise mask $M = \{\sigma_1, \ldots, \sigma_m\}$

Example of different representation schemes

Query1 (CoLA): Some people consider the noisy dogs dangerous. (✓)

Baseline: Some people consider the noisy dogs dangerous. (✓)
Mix-only: Some people consider the noisy dogs dangerous. (✓)
TextHide: I know a man who hates myself. (×)

Query2 (SST-2): otherwise excellent (😄)

Baseline: otherwise excellent (😄)
Mix-only: worthy (😄)
TextHide: passive-aggressive (☹)

Yangsibo Huang et al. TextHide: Tackling Data Privacy in Language Understanding Tasks. EMNLP, 2020.

Appendix

- **DialoGPT (2020)**
  - Chit-Chat Dialogue Model (e.g., ChatGPT (2022))
  - Model Architecture Based on GPT-2

$$p(T|S) = \prod_{n=m+1}^{N} p(x_n|x_1, \cdots, x_{n-1})$$

  - Objective for Multiturn dialogue session
  - $p(T_K, \cdots, T_2|T_1)$ can be perceived as optimizing all $p(T_i|T_1, \cdots, T_{i-1})$
  - Maximum mutual information (MMI) scoring function
    - Open-domain text generation models are notorious for generating bland, uninformative samples
    - Generate a set of hypotheses using top-K sampling
    - Use $P(\text{Source}|\text{target})$ to rerank all hypotheses

Yizhe Zhang et al. Dialogpt: Large-scale generative pre-training for conversational response generation. ACL, 2020..
https://openai.com/blog/chatgpt

Part 5. Appendix

- **DP-Adam**

DP-Adam

Algorithm 1 DP-Adam

Input: Data $\mathcal{D} = \{x_i\}_{i=1}^N$, learning rate η, noise multiplier σ, batch size B, Euclidean norm threshold for gradients C, epochs E, initial parameter vector $\theta_0 \in \mathbb{R}^p$, initial moment estimates $m_0, v_0 \in \mathbb{R}^p$, exponential decay rates $\beta_1, \beta_2 \in \mathbb{R}$, avoid division-by-zero constant $\gamma \in \mathbb{R}$.
for $t \in [E \cdot N/B]$ **do**
  Draw a batch B_t via Poisson sampling; each element has probability B/N of being selected
  for $x_i \in B_t$ **do**
    $g_t(x_i) \leftarrow \nabla_{\theta_t} \mathcal{L}(x_i), \quad \tilde{g}_t(x_i) \leftarrow g_t(x_i) \cdot \min(1, C/\|g_t(x_i)\|_2)$
  end for
  $z_t \sim \mathcal{N}(0, \sigma^2 C^2 I_p)$
  $\bar{g}_t = \frac{1}{B}\left(\sum_{i=1}^N \tilde{g}_t(x_i) + z_t\right)$
  $\theta_{t+1}, m_{t+1}, v_{t+1} \leftarrow \mathrm{AdamUpdate}(\theta_t, m_t, v_t, \bar{g}_t, \beta_1, \beta_2, \gamma)$
end for
return $\theta_{TN/B}$

Algorithm 2 AdamUpdate

Input: $\theta_t, m_t, v_t, \bar{g}_t, \beta_1, \beta_2, \gamma$
$m_{t+1} \leftarrow \beta_1 \cdot m_t + (1-\beta_1) \cdot \bar{g}_t, \quad v_{t+1} \leftarrow \beta_2 \cdot v_t + (1-\beta_2) \cdot \bar{g}_t{}^2$
$\widehat{m}_{t+1} \leftarrow m_{t+1} / \left(1-\beta_1^t\right), \quad \widehat{v}_{t+1} \leftarrow v_{t+1} / \left(1-\beta_2^t\right)$
$\theta_{t+1} \leftarrow \theta_t - \alpha \cdot \widehat{m}_{t+1} / \left(\sqrt{\widehat{v}_{t+1}} + \gamma\right)$
return $\theta_{t+1}, m_{t+1}, v_{t+1}$

Adam

Algorithm 1: *Adam*, our proposed algorithm for stochastic optimization. See section 2 for details, and for a slightly more efficient (but less clear) order of computation. g_t^2 indicates the elementwise square $g_t \odot g_t$. Good default settings for the tested machine learning problems are $\alpha = 0.001$, $\beta_1 = 0.9$, $\beta_2 = 0.999$ and $\epsilon = 10^{-8}$. All operations on vectors are element-wise. With β_1^t and β_2^t we denote β_1 and β_2 to the power t.

Require: α: Stepsize
Require: $\beta_1, \beta_2 \in [0, 1)$: Exponential decay rates for the moment estimates
Require: $f(\theta)$: Stochastic objective function with parameters θ
Require: θ_0: Initial parameter vector
  $m_0 \leftarrow 0$ (Initialize 1st moment vector)
  $v_0 \leftarrow 0$ (Initialize 2nd moment vector)
  $t \leftarrow 0$ (Initialize timestep)
  while θ_t not converged **do**
    $t \leftarrow t + 1$
    $g_t \leftarrow \nabla_\theta f_t(\theta_{t-1})$ (Get gradients w.r.t. stochastic objective at timestep t)
    $m_t \leftarrow \beta_1 \cdot m_{t-1} + (1-\beta_1) \cdot g_t$ (Update biased first moment estimate)
    $v_t \leftarrow \beta_2 \cdot v_{t-1} + (1-\beta_2) \cdot g_t^2$ (Update biased second raw moment estimate)
    $\widehat{m}_t \leftarrow m_t/(1-\beta_1^t)$ (Compute bias-corrected first moment estimate)
    $\widehat{v}_t \leftarrow v_t/(1-\beta_2^t)$ (Compute bias-corrected second raw moment estimate)
    $\theta_t \leftarrow \theta_{t-1} - \alpha \cdot \widehat{m}_t/(\sqrt{\widehat{v}_t} + \epsilon)$ (Update parameters)
  end while
  return θ_t (Resulting parameters)

Jaewoo Lee and Daniel Kifer. Scaling up differentially private deep learning with fast per-example gradient clipping. arXiv preprint arXiv:2009.03106, 2020.
Florian Tramer and Dan Boneh. Differentially Private Learning Needs Better Features (or Much More Data). ICLR, 2021.