Privacy-Preserving Natural Language Processing

Lecture 7 – Differentially-Private Stochastic Gradient Descent

Prof. Dr. Ivan Habernal

June 12, 2025

www.trusthlt.org

Chair of Trustworthy Human Language Technologies (TrustHLT)

Ruhr University Bochum & Research Center Trustworthy Data Science and Security

Recap

What we covered so far

- Pure $(\varepsilon, 0)$ differential privacy
- Central and Local DP
- Approximate (ε, δ)-DP
- Mechanisms: Laplace, Exponential, Randomized response, Gaussian
- Post processing and composition

Today

Let's finally do some supervised machine learning (neural networks)

Trained models (their weights) can leak training data

N. Carlini et al. (2020). **"Extracting Training Data from Large Language Models".** In: *arXiv preprint*

Recall from Lecture 1: Extracting attack by Carlini et al. (2020) — recovered training examples from GPT-2

- by prompting it with short strings sampled from the public Internet
- then manually checking whether these strings can also be found with a Google search

Simply prompting the model with data sampled from the model's training distribution (GPT-2 was trained on some unknown text sampled from the Internet), and (reasonably) assuming that any string memorized by the model is also contained in Google's search index

Model inversion attack (Fredrikson, Jha, and Ristenpart, 2015)

Figure 1: An image recovered using a new model inversion attack (left) and a training set image of the victim (right). The attacker is given only the person's name and access to a facial recognition system that returns a class confidence score.

M. Fredrikson, S. Jha, and T. Ristenpart (2015). **"Model Inversion Attacks that Exploit Confidence Information and Basic Countermeasures".** In: *Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security*. Denver, Colorado: ACM, pp. 1322–1333

- exploit confidence values exposed by the APIs
- attacker can produce a recognizable image of a person, given only API access to a facial recognition system and the name of the person whose face is recognized by it

Stochastic Gradient Descent recap

Stochastic Gradient Descent recap

Finding the best model’s parameters

Training as optimization

$$\mathcal{L}(\Theta) = \frac{1}{n} \sum_{i=1}^{n} L(f(\boldsymbol{x}_i; \Theta), y_i)$$

The training examples are fixed, and the values of the parameters determine the loss

The goal of the training algorithm is to set the values of the parameters Θ, such that the value of $\mathcal{L}$ is minimized

$$\hat{\Theta} = \underset{\Theta}{\operatorname{argmin}} \, \mathcal{L}(\Theta) = \underset{\Theta}{\operatorname{argmin}} \frac{1}{n} \sum_{i=1}^{n} L(f(\boldsymbol{x}_i; \Theta), y_i)$$

(Online) Stochastic Gradient Descent

1: **function** SGD($f(\boldsymbol{x}; \Theta)$, $(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n)$, $(\boldsymbol{y}_1, \ldots, \boldsymbol{y}_n)$, L)
2: **while** stopping criteria not met **do**
3: Sample a training example $\boldsymbol{x}_i, \boldsymbol{y}_i$
4: Compute the loss $L(f(\boldsymbol{x}_i; \Theta), \boldsymbol{y}_i)$
5: $\hat{\boldsymbol{g}} \leftarrow$ gradient of $L(f(\boldsymbol{x}_i; \Theta), \boldsymbol{y}_i)$ wrt. Θ
6: $\Theta \leftarrow \Theta - \eta_t \hat{\boldsymbol{g}}$
7: **return** Θ

Loss in line 4 is based on a **single training example** $\rightarrow$ a rough estimate of the corpus loss $\mathcal{L}$ we aim to minimize

The noise in the loss computation may result in inaccurate gradients

Minibatch Stochastic Gradient Descent

function mbSGD($f(\boldsymbol{x};\Theta)$, $(\boldsymbol{x}_1, \dots, \boldsymbol{x}_n)$, $(\boldsymbol{y}_1, \dots, \boldsymbol{y}_n)$, L)

  while stopping criteria not met **do**

    Sample m examples $\{(\boldsymbol{x}_1, \boldsymbol{y}_1), \dots (\boldsymbol{x}_m, \boldsymbol{y}_m)\}$

    $\hat{\boldsymbol{g}} \leftarrow 0$

    for $i = 1$ to m **do**

      Compute the loss $L(f(\boldsymbol{x}_i;\Theta), \boldsymbol{y}_i)$

      $\hat{\boldsymbol{g}} \leftarrow \hat{\boldsymbol{g}} +$ gradient of $\frac{1}{m} L(f(\boldsymbol{x}_i;\Theta), \boldsymbol{y}_i)$ wrt. Θ

    $\Theta \leftarrow \Theta - \eta_t \hat{\boldsymbol{g}}$

  return Θ

How to privatize SGD with DP

What can we privatize in the SGD algorithm by DP?

function SGD($f(\boldsymbol{x};\Theta)$, $(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n)$, $(\boldsymbol{y}_1, \ldots, \boldsymbol{y}_n)$, L)

  while stopping criteria not met **do**

    Sample a training example $\boldsymbol{x}_i, \boldsymbol{y}_i$

    Compute the loss $L(f(\boldsymbol{x}_i;\Theta), \boldsymbol{y}_i)$

    $\hat{\boldsymbol{g}} \leftarrow$ gradient of $L(f(\boldsymbol{x}_i;\Theta), \boldsymbol{y}_i)$ wrt. Θ

    $\Theta \leftarrow \Theta - \eta_t \hat{\boldsymbol{g}}$

  return Θ

What can we privatize in the SGD algorithm by DP?

function SGD($f(\boldsymbol{x}; \Theta)$, $(\boldsymbol{x}_1, \dots, \boldsymbol{x}_n)$, $(\boldsymbol{y}_1, \dots, \boldsymbol{y}_n)$, L)
  while stopping criteria not met **do**
    Sample a training example $\boldsymbol{x}_i$, $\boldsymbol{y}_i$
    Compute the loss $L(f(\boldsymbol{x}_i; \Theta), \boldsymbol{y}_i)$
    $\hat{\boldsymbol{g}} \leftarrow$ gradient of $L(f(\boldsymbol{x}_i; \Theta), \boldsymbol{y}_i)$ wrt. Θ
    $\Theta \leftarrow \Theta - \eta_t \hat{\boldsymbol{g}}$
  return Θ

- Privatize input
- Privatize output
- Privatize learning

How to privatize SGD with DP

Problem 1: Unbounded gradient and unbounded sensitivity

Unbounded sensitivity of gradient

Standard SGD

1: $\ldots$

2: $\boldsymbol{g}(\boldsymbol{x}_i) \leftarrow \nabla \mathcal{L}(\theta_t, \boldsymbol{x}_i)$ $\triangleright$ Compute gradient

3: $\ldots$

Clip the gradient vector by **per-example** ℓ_2 norm

$$\boldsymbol{g}(\boldsymbol{x}_i) \leftarrow \nabla \mathcal{L}(\theta_t, \boldsymbol{x}_i)$$

$$\bar{\boldsymbol{g}}(\boldsymbol{x}_i) \leftarrow \frac{\boldsymbol{g}(\boldsymbol{x}_i)}{\max\left(1, \frac{\|\boldsymbol{g}(\boldsymbol{x}_i)\|_2}{C}\right)}$$

where $C \in \mathbb{R}$ is a clipping constant (hyper-parameter)

How to privatize SGD with DP

Problem 2: Too many steps for simple composition

Running several mechanisms on the same data

function SGD($f(\boldsymbol{x}; \Theta)$, $(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n)$, $(\boldsymbol{y}_1, \ldots, \boldsymbol{y}_n)$, L)
    while stopping criteria not met **do**
        $\ldots$
    return Θ

Composition theorems: Running the same or various privacy mechanisms on the same data

Basic composition — "epsilons and deltas add up"

For $k \in \mathbb{N}$, the composition of k mechanisms (each of them is (ε, δ)-DP) gives $(k\varepsilon, k\delta)$-DP

This would lead to an excessively high overall budget

Running several mechanisms on the same data

Basic composition — “epsilons and deltas add up”

For k steps (each (ε, δ)-DP): $(k\varepsilon, k\delta)$-DP

Theorem III.3 in C. Dwork, G. N. Rothblum, and S. Vadhan (2010). **“Boosting and Differential Privacy”.** In: *2010 IEEE 51st Annual Symposium on Foundations of Computer Science*. Las Vegas, USA: IEEE, pp. 51–60

k-fold adaptive composition of an (ε, δ)-DP mechanism

Advanced composition — using smaller overall budget

For $\delta' > 0$ and $\varepsilon' = \varepsilon\sqrt{2k \ln(1/\delta')} + k\varepsilon(\exp(\varepsilon) - 1)$ the composite mechanism is $(\varepsilon', k\delta + \delta')$-DP

Great news: Advanced composition gives us quadratic improvement wrt. number of steps k

- $\approx \sqrt{k} \cdot \varepsilon$ instead of simple $k \cdot \varepsilon$

How to privatize SGD with DP

Trick 3: Sub-sampling helps to reduce the budget in each step

Privacy amplification by sub-sampling

Let's define a **sampling function** that takes a dataset $D_{\text{in}} \in \mathcal{X}$ and produces another dataset $D_{\text{out}} \in \mathcal{X}$ as follows:

- For each entry t from D_{in} the function draws a binary value at random
  - We draw 'zero or one' using a Bernoulli random variable $\text{Ber}(\beta)$ parametrized by $\beta \in (0, 1)$
- If it's 1, this entry t will end up in the output dataset D_{out}
- If it's 0, this entry is ignored

Important: For each entry t the Bernoulli trial is independent of other entries

This is also known as **Poisson sampling**

Privacy amplification by sub-sampling

Let's have an $(\varepsilon_1, \delta_1)$-DP algorithm $\mathcal{A}_1$

We propose a new algorithm $\mathcal{A}_2$ that works in two steps:

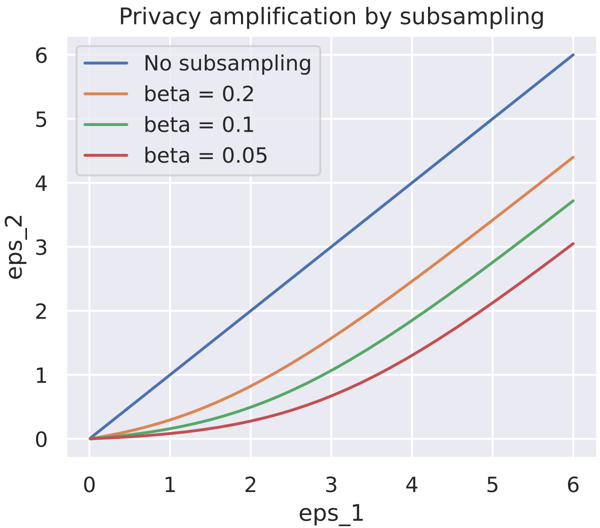
1. Sub-sample our dataset D using Poisson sampling (with parameter β)
2. Run $\mathcal{A}_1$ on this smaller dataset

Then $\mathcal{A}_2$ is $(\varepsilon_2, \delta_2)$-DP, where

$$\varepsilon_2 = \ln\left(1 + \beta\left[\exp(\varepsilon_1) - 1\right]\right) \qquad \delta_2 = \beta\delta_1$$

Proof in the appendix of N. Li, W. Qardaji, and D. Su (2012). **"On Sampling, Anonymization, and Differential Privacy Or, K-Anonymization Meets Differential Privacy".** In: *Proceedings of the 7th ACM Symposium on Information, Computer and Communications Security*. Seoul, South Korea: ACM, pp. 32–33; there are a few 'nasty' typos.

How much we can 'save' on the privacy budget?



Why is Poisson sampling relevant for SGD?

Recall Mini-batch SGD!

function mbSGD($f(\boldsymbol{x}; \Theta)$, $(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n)$, $(\boldsymbol{y}_1, \ldots, \boldsymbol{y}_n)$, L)
    while stopping criteria not met **do**
        Sample m examples $\{(\boldsymbol{x}_1, \boldsymbol{y}_1), \ldots (\boldsymbol{x}_m, \boldsymbol{y}_m)\}$
        $\ldots$

- We usually use small ‘batches’ which are somehow randomly subsampled from the training dataset
- We can replace the minibatch sampling with Poisson sampling!

DP-SGD

DP-SGD algorithm

function DP-SGD($f(\boldsymbol{x}; \Theta)$, $(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n)$, $|L|$ — 'lot' size, T — # of steps)

  for $t \in (1, 2, \ldots, T)$ **do**

    Add each training example to a 'lot' L_t with probability $|L|/n$

    for each example in the 'lot' $\boldsymbol{x}_i \in L_t$ **do**

      $\boldsymbol{g}(\boldsymbol{x}_i) \leftarrow \nabla \mathcal{L}(\theta_t, \boldsymbol{x}_i)$ ▷ Compute gradient

      $\bar{\boldsymbol{g}}(\boldsymbol{x}_i) \leftarrow \boldsymbol{g}(\boldsymbol{x}_i) / \max\left(1, \|\boldsymbol{g}(\boldsymbol{x}_i)\| / C\right)$ ▷ Clip gradient

      $\tilde{\boldsymbol{g}}(\boldsymbol{x}_i) \leftarrow \bar{\boldsymbol{g}}(\boldsymbol{x}_i) + \mathcal{N}(0, \sigma^2 C^2 \boldsymbol{I})$ ▷ Add noise

    $\hat{\boldsymbol{g}} \leftarrow \frac{1}{|L|} \sum_{k=1}^{|L|} \tilde{\boldsymbol{g}}(\boldsymbol{x}_k)$ ▷ Gradient estimate of 'lot' by averaging

    $\Theta_{t+1} \leftarrow \Theta_t - \eta_t \hat{\boldsymbol{g}}$ ▷ Update parameters by gradient descend

  return Θ

Stochastic gradient descent with differential privacy

Setup: A set of labeled i.i.d. examples — like tabular data (each example = single person)

Privacy 'accountant' — utilizes composition of DP

- Computes the privacy cost at each access to the training data (gradient computation)
- Accumulates this cost as the training progresses

Tightest privacy by numerical integration to get bounds on the **moment generating function** of the **privacy loss random variable** for all moments ≤ 32

M. Abadi, A. Chu, I. Goodfellow, H. B. McMahan, I. Mironov, K. Talwar, and L. Zhang (2016). **"Deep Learning with Differential Privacy".** In: *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security*. Vienna, Austria: ACM, pp. 308–318

Recap of DP-SGD

- DP-SGD 'de-facto' standard for supervised training with DP
- Implemented in Opacus, Tensorflow privacy, and other libs

M. Abadi, A. Chu, I. Goodfellow, H. B. McMahan, I. Mironov, K. Talwar, and L. Zhang (2016). **"Deep Learning with Differential Privacy".** In: *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security.* Vienna, Austria: ACM, pp. 308–318

What makes it tricky?

- Remember: Data points **must** be independent (privacy-wise)
- Scalability: Per-example gradient norm and clipping is super slow

The Obvious Application: Supervised Training

DP-SGD across various NLP tasks

Setup:

Although DP-SGD had been used in language modeling, the community lacked a thorough understanding of its usability across different NLP tasks

Research questions:

- Which models and training strategies provide the best trade-off between privacy and performance on different NLP tasks?
- How exactly do increasing privacy requirements hurt the performance?

M. Senge, T. Igamberdiev, and I. Habernal (2022). **“One size does not fit all: Investigating strategies for differentially-private learning across NLP tasks”.** In: *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*. Ed. by Y. Goldberg, Z. Kozareva, and Y. Zhang. Abu Dhabi, United Arab Emirates: Association for Computational Linguistics, pp. 7340–7353

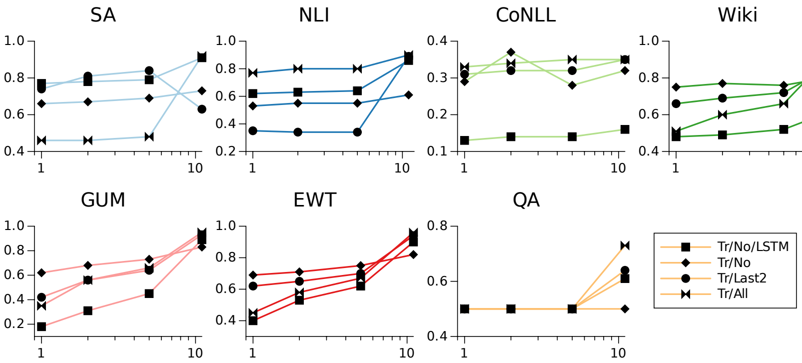
DP-SGD across various NLP tasks: Datasets

Task	Dataset	Size	Classes
SA	IMDb	50k documents	2
NLI	SNLI	570k pairs	3
NER	CoNLL'03	$\approx$ 300k tokens	9
NER	Wikiann	$\approx$ 320k tokens	7
POS	GUM	$\approx$ 150k tokens	17
POS	EWT	$\approx$ 254k tokens	17
QA	SQuAD 2.0	150k questions	$\star$

Table 1: Datasets and their specifics. $\star$ SQuAD contains 100k answerable and 50k unanswerable questions, where answerable questions are expressed as the span positions of their answer.

M. Senge, T. Igamberdiev, and I. Habernal (2022). **"One size does not fit all: Investigating strategies for differentially-private learning across NLP tasks".** In: *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*. Ed. by Y. Goldberg, Z. Kozareva, and Y. Zhang. Abu Dhabi, United Arab Emirates: Association for Computational Linguistics, pp. 7340–7353

DP-SGD across various NLP tasks: Results



M. Senge, T. Igamberdiev, and I. Habernal (2022). **"One size does not fit all: Investigating strategies for differentially-private learning across NLP tasks".** In: *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*. Ed. by Y. Goldberg, Z. Kozareva, and Y. Zhang. Abu Dhabi, United Arab Emirates: Association for Computational Linguistics, pp. 7340–7353

Figure 1: Comparison of BERT performances (macro F_1 score) per dataset with varying privacy budget $\varepsilon \in \{1, 2, 5, \infty\}$ on the x-axis (note the $\log$ scale).

The Less Obvious Application: Language Models

Early DP language models

H. B. McMahan, D. Ramage, K. Talwar, and L. Zhang (2018). **"Learning Differentially Private Recurrent Language Models".** In: *Proceedings of the 6th International Conference on Learning Representations.* Vancouver, BC, Canada, pp. 1–14

Motivated by the problem of training models for next-word prediction in a mobile keyboard; used this as a running example

Early DP language models: Neighboring datasets

H. B. McMahan, D. Ramage, K. Talwar, and L. Zhang (2018). **"Learning Differentially Private Recurrent Language Models".** In: *Proceedings of the 6th International Conference on Learning Representations.* Vancouver, BC, Canada, pp. 1–14

Most prior work on differentially private machine learning deals with example-level privacy

— Two datasets D and D' are defined to be adjacent if D' can be formed by adding or removing a **single training example** from D

But:

— A sensitive word or phrase may be typed several times by an individual user, but it should still be protected

Early DP language models: Neighboring datasets

McMahan, Ramage, Talwar, and Zhang (2018) thus **defined**:

Definition: User-adjacent datasets

Let D and D' be two datasets of training examples, where each example is associated with a user. Then, D and D' are adjacent if D' can be formed by adding or removing **all of the examples associated with a single user** from D.

D contains training examples, each associated with a user, e.g., $D = \{A_1, A_2, B_1, B_2\}$ where $\{A, B\}$ are the users. Then D' can be formed by adding or removing all examples from one user, e.g., $D' = \{A_1, A_2\}$, or $D' = \{A_1, A_2, B_1, B_2, C_1\}$, but not $\{A_1, A_2, B_1\}$

H. B. McMahan, D. Ramage, K. Talwar, and L. Zhang (2018). **"Learning Differentially Private Recurrent Language Models".** In: *Proceedings of the 6th International Conference on Learning Representations*. Vancouver, BC, Canada, pp. 1–14

Early DP language models: Training the model with DP

Their private algorithm relies heavily on two prior works

- FederatedAveraging (or FedAvg) algorithm of McMahan et al. (2016), which trains deep networks on user-partitioned data
- the moments accountant of Abadi et al. (2016), which provides tight composition guarantees for the repeated application of the Gaussian mechanism combined with amplification-via-sampling

H. B. McMahan, E. Moore, D. Ramage, S. Hampson, and B. A. y Arcas (2016). **"Federated Learning of Deep Networks using Model Averaging".** In: *arXiv preprint*

Early DP language models: Training the model with DP

FedAvg was introduced by McMahan et al. (2016) for federated learning, where the goal is to train a shared model while leaving the training data on each user's mobile device. Instead, devices download the current model and compute an update by performing local computation on their dataset.

Most importantly, the algorithm naturally forms per-user updates based on a single user's data, and these updates are then averaged to compute the final update applied to the shared model on each round.

This structure makes it possible to extend the algorithm to provide a user-level differential privacy guarantee.

H. B. McMahan, E. Moore, D. Ramage, S. Hampson, and B. A. y Arcas (2016). **"Federated Learning of Deep Networks using Model Averaging".** In: *arXiv preprint*

Early DP language models: Training the model with DP

To achieve differential privacy:

- A) They use random-sized batches where we select users independently with probability q, rather than always selecting a fixed number of users.
- B) They enforce clipping of per-user updates so the total update has bounded ℓ_2 norm.
- C) (They use different estimators for the average update)
- D) They add Gaussian noise to the final average update.

H. B. McMahan, D. Ramage, K. Talwar, and L. Zhang (2018). **"Learning Differentially Private Recurrent Language Models".** In: *Proceedings of the 6th International Conference on Learning Representations.* Vancouver, BC, Canada, pp. 1–14

Early DP language models: Data and evaluation

Data

Used a large public dataset of Reddit posts

Each post in the database is keyed by an author, so they group the data by these keys in order to provide user-level privacy.

763,430 users each with 1600 tokens

Evaluation

– LSTM language model (1.35M params)

– They evaluate using `AccuracyTop1`, the probability that the word to which the model assigns highest probability is correct

H. B. McMahan, D. Ramage, K. Talwar, and L. Zhang (2018). **"Learning Differentially Private Recurrent Language Models".** In: *Proceedings of the 6th International Conference on Learning Representations.* Vancouver, BC, Canada, pp. 1–14

Early DP language models: Results

H. B. McMahan, D. Ramage, K. Talwar, and L. Zhang (2018). **“Learning Differentially Private Recurrent Language Models”.** In: *Proceedings of the 6th International Conference on Learning Representations.* Vancouver, BC, Canada, pp. 1–14

model		data			
σ	S	users K	$\tilde{C}$	ϵ	AccT1
0.000	∞	763430	100	∞	17.62%
0.003	**15**	763430	5000	4.634	17.49%
0.006	**10**	763430	1667	2.314	17.04%
0.012	**15**	763430	1250	2.038	16.33%

When Things Go Tricky

Poisson subsampling versus just batches?

The ‘standard’ random shuffling method for iterating over batches providing a weaker privacy guarantee for the training data than Poisson sampling.

— Experiments with Neural Machine Translation

T. Igamberdiev, D. N. L. Vu, F. Kuennecke, Z. Yu, J. Holmer, and I. Habernal (2024). **“DP-NMT: Scalable Differentially Private Machine Translation”.** In: *Proceedings of the 18th Conference of the European Chapter of the Association for Computational Linguistics: System Demonstrations.* Ed. by N. Aletras and O. De Clercq. St. Julians, Malta: Association for Computational Linguistics, pp. 94–105

Datasets

— WMT-16 (DE-EN) language pair

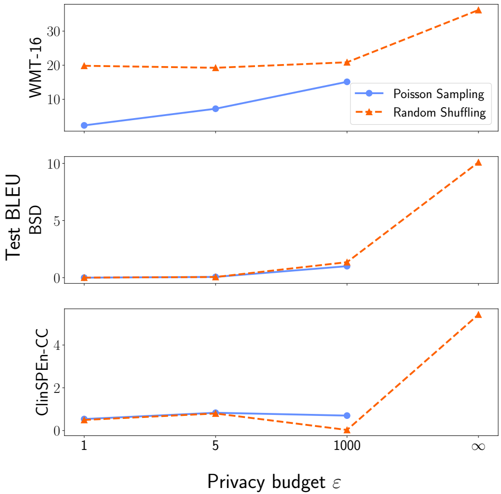
— Business Scene Dialogue corpus (BSD), a collection of fictional business conversations in various scenarios (e.g. “face-to-face”, “phone call”, “meeting”), Japanese and English

— ClinSPEn-CC, a collection of parallel COVID-19 clinical cases in English and Spanish

Dataset	Lang. Pair	# Trn.+Vld.	# Test
WMT-16	DE-EN	4,551,054	2,999
BSD	JA-EN	22,051	2,120
ClinSPEn-CC	ES-EN	1,065	2,870

T. Igamberdiev, D. N. L. Vu, F. Kuennecke, Z. Yu, J. Holmer, and I. Habernal (2024). **“DP-NMT: Scalable Differentially Private Machine Translation”.** In: *Proceedings of the 18th Conference of the European Chapter of the Association for Computational Linguistics: System Demonstrations.* Ed. by N. Aletras and O. De Clercq. St. Julians, Malta: Association for Computational Linguistics, pp. 94–105

Results



T. Igamberdiev, D. N. L. Vu, F. Kuennecke, Z. Yu, J. Holmer, and I. Habernal (2024). **“DP-NMT: Scalable Differentially Private Machine Translation”.** In: *Proceedings of the 18th Conference of the European Chapter of the Association for Computational Linguistics: System Demonstrations.* Ed. by N. Aletras and O. De Clercq. St. Julians, Malta: Association for Computational Linguistics, pp. 94–105

When Things Go Very Tricky

Private information in text?

our understanding of what is *private information* in textual data is still very limited

Applications of DP — guarantee to each individual *data point*

For textual data, a single data point will often be a sentence or document.

However, this does not mean that there is a one-to-one mapping from *individuals* to sentences and documents. For instance, multiple documents could potentially refer to the same individual, or contain the same piece of sensitive information that would break the assumption of each data point being independent.

T. Igamberdiev, D. N. L. Vu, F. Kuennecke, Z. Yu, J. Holmer, and I. Habernal (2024). **"DP-NMT: Scalable Differentially Private Machine Translation".** In: *Proceedings of the 18th Conference of the European Chapter of the Association for Computational Linguistics: System Demonstrations.* Ed. by N. Aletras and O. De Clercq. St. Julians, Malta: Association for Computational Linguistics, pp. 94–105

Private information in text?

"In this paper, we discuss the mismatch between the narrow assumptions made by popular data protection techniques (data sanitization and differential privacy), and the broadness of natural language and of privacy as a social norm."

"We argue that existing protection methods cannot guarantee a generic and meaningful notion of privacy for language models. We conclude that language models should be trained on text data which was explicitly produced for public use."

H. Brown, K. Lee, F. Mireshghallah, R. Shokri, and F. Tramèr (2022). **"What Does it Mean for a Language Model to Preserve Privacy?"** In: *2022 ACM Conference on Fairness, Accountability, and Transparency*. New York, NY, USA: ACM, pp. 2280–2292

Private information in text?

The approach to preserving privacy in LMs has been to attempt complete removal of private information from training data (data sanitization), or to design algorithms that do not memorize private data, such as algorithms that satisfy differential privacy (DP)

Both methods make explicit and implicit assumptions about the structure of data to be protected, the nature of private information, and requirements for privacy, that do not hold for the majority of natural language data.

H. Brown, K. Lee, F. Mireshghallah, R. Shokri, and F. Tramèr (2022). **"What Does it Mean for a Language Model to Preserve Privacy?"** In: *2022 ACM Conference on Fairness, Accountability, and Transparency*. New York, NY, USA: ACM, pp. 2280–2292

Private information in text?

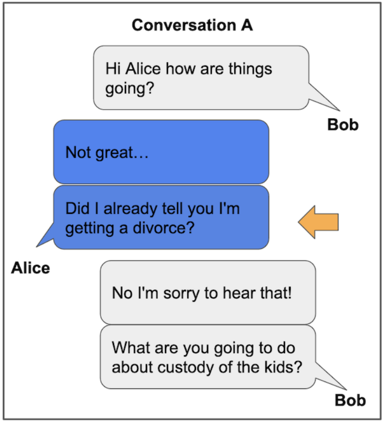
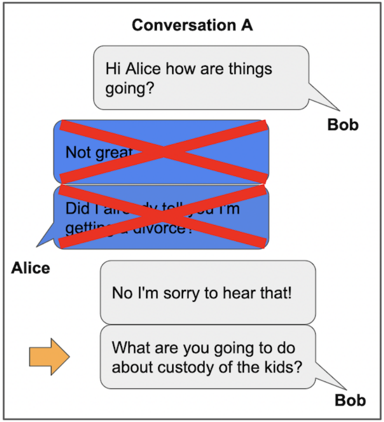


Figure 2: Original conversation. Private information indicated by orange arrows.

H. Brown, K. Lee, F. Mireshghallah, R. Shokri, and F. Tramèr (2022). **"What Does it Mean for a Language Model to Preserve Privacy?"** In: *2022 ACM Conference on Fairness, Accountability, and Transparency*. New York, NY, USA: ACM, pp. 2280–2292

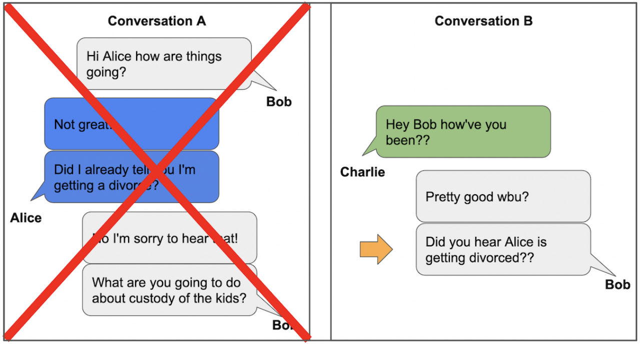
Private information in text?



H. Brown, K. Lee, F. Mireshghallah, R. Shokri, and F. Tramèr (2022). **"What Does it Mean for a Language Model to Preserve Privacy?"** In: *2022 ACM Conference on Fairness, Accountability, and Transparency*. New York, NY, USA: ACM, pp. 2280–2292

Figure 3: Alice's messages removed. Bob's last message still includes her private information.

Private information in text?



H. Brown, K. Lee, F. Mireshghallah, R. Shokri, and F. Tramèr (2022). **"What Does it Mean for a Language Model to Preserve Privacy?"** In: *2022 ACM Conference on Fairness, Accountability, and Transparency*. New York, NY, USA: ACM, pp. 2280–2292

Figure 4: The whole original conversation is removed. Conversation B still contains Alice's private information though she is not in the conversation.

License and credits

Credits

Ivan Habernal

Content from ACL Anthology papers licensed under CC-BY
`https://www.aclweb.org/anthology`

Partly inspired by lectures from Gautam Kamath