Privacy-preserving Data Deduplication for Enhancing Federated Learning of Language Models

Aydin Abadi*
Newcastle University

Vishnu Asutosh Dasu*
Pennsylvania State University

Sumanta Sarkar*
University of Warwick

*Equal Contribution

Duplicated Data and LLMs

- Quality of training data has a significant impact on ML algorithms
- Duplicated text is prevalent in vast text datasets used for LLM training
- These duplicates adversely effect LLMs by increasing perplexity and training time [Lee et al., ACL 2022]
  - C4 dataset has a 61 word sequence repeated verbatim 61,036 times
  - Deduplication improved perplexity by up to 10%

[Lee et al., ACL 2022] K. Lee, D. Ippolito, A. Nystrom, C. Zhang, D. Eck, C. Callison-Burch, and N. Carlini, “Deduplicating training data makes language models better,” 2022.

Federated Learning and Deduplication

- In federated learning, a server orchestrates the training of a global model on datasets held by multiple clients
- Server only aggregates gradient updates and cannot see client data
- Prior work [Lee et al., ACL 2022] deduplicates data in the centralized setting
- **How do we deduplicate datasets in federated learning while preserving data privacy?**

[Lee et al., ACL 2022] K. Lee, D. Ippolito, A. Nystrom, C. Zhang, D. Eck, C. Callison-Burch, and N. Carlini, “Deduplicating training data makes language models better,” 2022.

Our Solution

- Our solution, **Efficient Privacy-Preserving Multi-Party Deduplication** (EP-MPD), is based on a new variant of **Private Set Intersection** (PSI) protocols

- EP-MPD securely removes all pairwise duplicates from datasets of two or more clients in federated learning

- We introduce the notion of **Group Private Set Intersection** (G-PSI), which is used as a building block for EP-MPD

Background

Private Set Intersection (PSI)

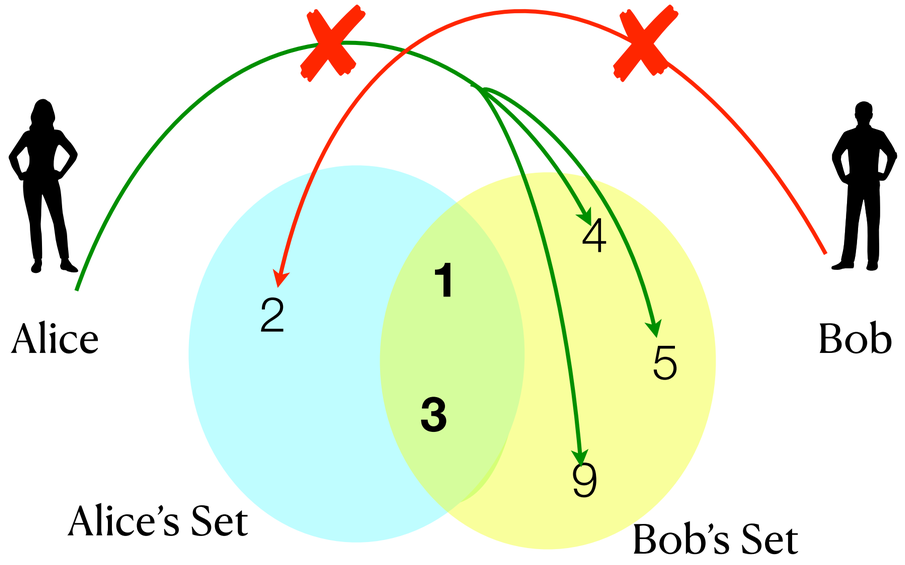
- PSI lets mutually distrustful parties compute intersection of their private sets such that nothing beyond the intersection is reveal

- Example:

  - During PSI computation, Bob must not learn 2 and Alice must not learn 4,5, and 9

Intersection of sets A and B: 1, 3

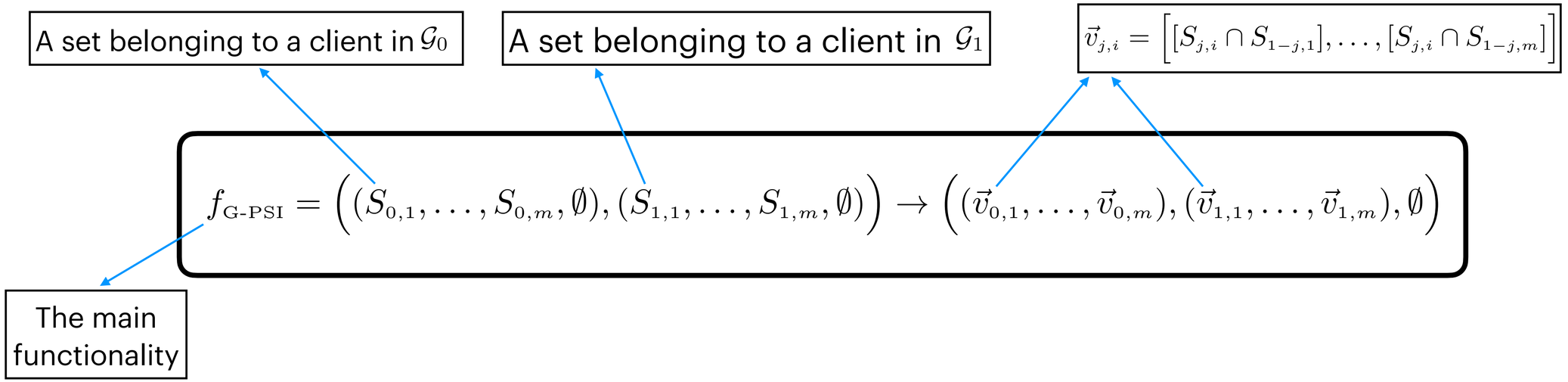
$A \cap B$



Our Solution

G-PSI as a building block of EP-MPD

- G-PSI operates on two groups of users (clients)
- G-PSI allows all clients in one group to find the pairwise intersection of their set with the sets of all clients in the other group



Our Solution

G-PSI Implementation Details

- We develop two protocols that meet the requirements of **G-PSI**

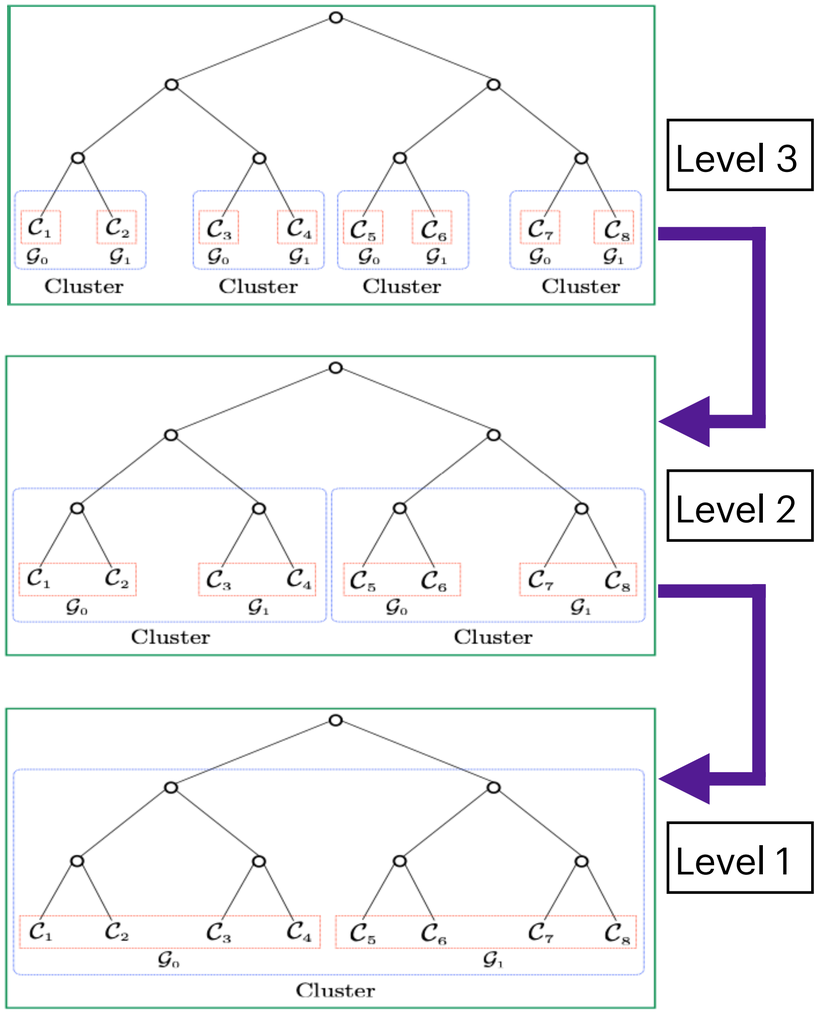
1. **EG-PSI (I)**: Uses private key encryption and requires a Trusted Execution Environment (TEE) to find shared encrypted data, thus requiring very little processing time

2. **EG-PSI (II)**: Uses public key encryption and requires a TEE to only encrypt data. Requires more processing time but TEE plays a smaller role

Our Solution

EP-MPD from G-PSI

- EP-MPD constructs a binary tree where the leaf nodes represent clients
- At each level, clusters of clients are formed where the left and right subtrees represent two groups
- Starting from the lowest level, EP-MPD iteratively invokes G-PSI on the clusters as we move up the tree
- Once we reach the root of the tree, all pairwise duplicates have been removed



EP-MPD and Federated Learning

1. Each user locally removes duplicates

2. All users run EP-MPD to remove pairwise duplicates

3. All users join the federated learning protocol

- *Parties*. A set of clients $\{\mathcal{C}_1, \ldots, \mathcal{C}_m\}$.
- *Server*. Holds the initial model $\boldsymbol{\theta}$.
- *Inputs*. Sets $S_1, \ldots, S_m$, where each S_i belongs to client $\mathcal{C}_i$, $1 \leq i \leq m$, and m is a power of two.
- *Outputs*. A global updated model Θ.

---

1) *Local Deduplication.* Each client runs a deduplication algorithm on their local dataset. At the end, client $\mathcal{C}_i$ receives an updated dataset S_i'.
2) *Global Deduplication.*
   a) All the clients participate in the EP-MPD as described in Figure 4.
   b) Each client $\mathcal{C}_i$ gets updated set S_i'', such that

$$\sum_{i=1}^{m} S_i'' = \bigcup_{i=1}^{m} S_i'.$$

3) *Federated learning.*
   a) The server and clients agree upon an FL protocol for training.
   b) The server initiates the learning by sharing the initial model $\boldsymbol{\theta}$ with each client.
   c) Each client $\mathcal{C}_i$ trains on their local dataset S_i'' and updates $\boldsymbol{\theta}$ to $\boldsymbol{\theta}_i$.
   d) The clients and server aggregate the local models $\boldsymbol{\theta}_i$ trained by the clients.
   e) The server outputs the global updated model Θ for the next training round.

Experiments

- We implement EP-MPD in Python with AES-128 CBC as a PRP for EG-PSI (I) and EC-OPRF for EG-PSI (II)
- We create a federated learning setup to fine-tune LLMs with 10 clients
- We evaluate with 7 text datasets and vary the percentage of duplicates up to 30%
- We compare the perplexity and GPU training time before and after deduplication with EP-MPD
- We use the GPT-2 Medium and GPT-2 Large models

Results

- **EP-MPD** is up to **14x** faster that naive application of two-party PSI

Client Count	EP-MPD		Naive Approach (Two-party PSI)
	EP-MPD$^{(I)}$	EP-MPD$^{(II)}$	
10	162.2	1529.8	616.8
20	339.8	3062.4	2537.4
30	506.4	4596.0	5890.8
40	806.6	6113.5	9884.0
50	1160.3	7653.2	15974.1

Results

- **Perplexity** improves up to **19%** after deduplication

Model	Dataset	Duplication Percentage						Deduplicated
		30%		20%		10%		
		PP	IR (%)	PP	IR (%)	PP	IR (%)	PP
GPT-2 Medium	Haiku	3.78	4.36	3.7	2.37	3.65	1.09	3.61
	Rotten Tomatoes	2.4	3.62	2.36	2.0	2.35	1.67	2.31
	Short Jokes	3.96	5.34	3.89	3.79	3.83	2.31	3.74
	Poetry	6.24	14.47	6.51	18.07	5.77	7.61	5.33
	IMDB	13.17	7.1	12.57	2.67	12.49	2.05	12.23
	Sonnets	15.83	13.64	15.63	12.54	14.23	3.88	13.67
	Plays	34.32	18.3	34.89	19.62	28.12	–	28.04
GPT-2 Large	Haiku	3.26	11.29	3.25	10.83	2.98	2.78	2.89
	Rotten Tomatoes	2.65	16.79	2.61	15.29	2.53	12.81	2.21
	Short Jokes	4.11	7.84	4.03	5.86	3.94	3.64	3.79
	Sonnets	8.52	5.58	8.4	4.27	8.02	–	8.04

Results

- Total GPU **training time** (minutes) improves up to **27%** after deduplication

Model	Dataset	Duplication Percentage						Deduplicated
		30%		20%		10%		
		Time	IR (%)	Time	IR (%)	Time	IR (%)	Time
GPT-2 Medium	Haiku	111.64	23.06	101.67	15.51	93.82	8.44	85.9
	Rotten Tomatoes	162.79	21.7	151.54	15.89	138.76	8.14	127.46
	Short Jokes	396.62	27.85	338.69	15.51	313.35	8.68	286.15
	Poetry	101.33	22.55	94.25	16.73	86.87	9.66	78.48
	IMDB	2103.93	23.99	1945.94	17.82	1772.44	9.77	1599.24
	Sonnets	33.13	27.95	28.53	16.33	26.14	8.68	23.87
	Plays	31.48	22.9	29.38	17.39	26.95	9.94	24.27
GPT-2 Large	Haiku	21.0	22.05	19.65	16.69	18.02	9.16	16.37
	Rotten Tomatoes	70.74	23.08	65.26	16.63	59.91	9.18	54.41
	Short Jokes	340.75	22.93	313.65	16.27	288.86	9.08	262.63
	Sonnets	13.91	20.92	12.89	14.66	11.87	7.33	11.0

Conclusion

- We develop **EP-MPD**, a protocol that efficiently removes all pairwise duplicates in federated learning while preserving data privacy
- We observe up to **19%** improvement in perplexity and up to **27%** improvement in GPU training time
- For future work, one could remove **near duplicates** in federated learning
- Contact: vdasu@psu.edu

Paper

Source Code