This work was performed during 2019 Fall and 2020 Spring semester. Stacey would like to acknowledge the 2019 - 2020 IBM PhD fellowship recently awarded to her and the support from IBM Almaden under the Enterprise AI, Systems & Solutions Research Group (director: Sandeep Gopisetty).
• Motivation
• Deep Learning & Federated Training
• Privacy Leakage in Federated Learning Systems
• Differential Privacy for Iterative ML Training Mechanisms
  • $\alpha$-Condensed Local Differential Privacy
• LDP-Fed: Federated Learning with Local Differential Privacy
  • LDP Module, $k$-Selection Module
• LDP-Fed Performance & Features
• Conclusion
In 2020 there will be 40x more bytes of data than there are stars in the observable universe.

DOMO report

Infographic source: rightedge
Machine learning

5 Year Growth Rate: 34%

• Published patent applications for Patent Classification G06N “Computer Systems Based on Specific Computational Models” grew at a compound annual rate of 34% from 2013 to 2017.

• This includes machine learning and artificial neural networks.

Forbes article

<table>
<thead>
<tr>
<th>Company</th>
<th>2017 Published Applications</th>
</tr>
</thead>
<tbody>
<tr>
<td>IBM</td>
<td>654</td>
</tr>
<tr>
<td>Microsoft</td>
<td>139</td>
</tr>
<tr>
<td>Google</td>
<td>127</td>
</tr>
<tr>
<td>LinkedIn</td>
<td>70</td>
</tr>
<tr>
<td>Facebook</td>
<td>66</td>
</tr>
<tr>
<td>Intel</td>
<td>52</td>
</tr>
<tr>
<td>Fujitsu</td>
<td>49</td>
</tr>
</tbody>
</table>

Data Science platforms that support machine learning are predicted to grow at a 13% CAGR through 2021.
REACTION: DEMAND FOR PRIVACY

CCPA: What Does It Mean For AI (Artificial Intelligence)?

By Katherine Bindley
Updated Nov 22, 2019 11:56 pm ET

AI Has a Big Privacy Problem and Europe’s New Data Protection Law Is About to Expose It

By David Meyer
May 25, 2018 6:52 AM EDT
Activation function converts input signals to an output signal

An activation function is applied to the sum of the product of input signals and their corresponding weights.
Learning Process

1. Shuffle data and divide into batches
2. Feed batches forward through the network
3. Calculate Error
4. Backpropagate the error
5. Use gradients to update weights
1. Same model structure & parameters are initialized at each participant

2. Each participant conducts local training on their private dataset, resulting in updated parameters

3. Locally updated model parameters are sent to the parameter server

4. Server aggregates the parameter updates using Federated Averaging

5. New, aggregated parameters are broadcast to all participants
Membership Inference Attacks:

Given training dataset $D$, and a model $M$ trained on $D$, and a data point $x$.

Can an attacker determine if $x \in D$?

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Attack Accuracy[1]</th>
</tr>
</thead>
<tbody>
<tr>
<td>Purchase History (100 class)</td>
<td>72.4%</td>
</tr>
<tr>
<td>Texas Hospital Stays</td>
<td>66.4%</td>
</tr>
<tr>
<td>CIFAR-100 (AlexNet)</td>
<td>85.1%</td>
</tr>
</tbody>
</table>

Definition

Differential Privacy [1]: A randomized mechanism $K$ provides $(\epsilon, \delta)$-differential privacy if for any two neighboring databases $D_1$ and $D_2$ that differ in only a single entry and $\forall S \subseteq Range(K)$

$$\Pr(K(D_1) \in S) \leq e^{\epsilon} \cdot \Pr(K(D_2) \in S) + \delta$$

If $\delta = 0$, $K$ is said to satisfy $\epsilon$-differential privacy.

***Limits the impact that any one instance can have on the mechanism output***

Composition Property

Sequential Composition property\(^1\): Let \(f_1, f_2, \ldots, f_n\) be \(n\) algorithms such that for each \(i \in [1, n]\), \(f_i\) satisfies \((\epsilon_i, \delta_i)\)-differential privacy. Then,

Releasing the outputs of \(f_1(D), f_2(D), \ldots, f_n(D)\) satisfies \((\sum_{i=1}^n \epsilon_i, \sum_{i=1}^n \delta_i)\)-DP.

***Multiple passes on a dataset causes additive privacy loss in differential privacy***

Definition

$\epsilon$-LDP \cite{ding2017collecting}: A randomized mechanism $\Psi$ provides $\epsilon$-local differential privacy where $\epsilon > 0$, if and only if for any inputs $v_1, v_2$ in universe $\mathcal{U}$ and $\forall y \in \text{Range}(\Psi)$, we have:

$$\Pr[\Psi(v_1) = y] \leq e^\epsilon \cdot \Pr[\Psi(v_2) = y]$$

**Protects the raw value (input to $\Psi$) from privacy inference according to $\epsilon$**

\(\alpha\)-CONDENSED LOCAL DIFFERENTIAL PRIVACY

Definition

\(\alpha\)-CLDP \(^1\): A randomized mechanism \(\Phi\) provides \(\alpha\)-condensed local differential privacy where \(\alpha > 0\), if and only if for any inputs \(v_1, v_2\) in universe \(\mathcal{U}\) and \(\forall y \in \text{Range}(\Phi)\), we have:

\[
\Pr[\Phi(v_1) = y] \leq e^{\alpha \cdot d(v_1,v_2)} \cdot \Pr[\Phi(v_2) = y]
\]

***Protects the raw value (input to \(\Phi\)) from privacy inference according to \(\alpha\) and \(d(\cdot,\cdot)\)***

LDP-FED: FEDERATED LEARNING WITH LOCAL DIFFERENTIAL PRIVACY

**Client Perspective**

1. Participants initialize local DNN instances and model parameters with local LDP Modules to protect privacy.
2. The server waits to receive $k$ parameter updates.
3. Each participant $P_i$ locally computes training gradients according to $D_i$.
4. Each $P_i$ perturbs their gradients according to their local instance of the LDP Module.
5. The $k$-Client Selection Module accepts update from each $P_i$ with probability $q = k/N$.
6. Each participant waits to receive aggregated parameter updates from the parameter server and then updates its local DNN.
7. Each $P_i$ proceeds to step 2 to start the next iteration.

**Server Perspective**

1. The parameter server generates initial model parameters $\theta_0$ and sends them to each participant.
2. The server waits to receive $k$ parameter updates randomly selected by the $k$-Client Selection Module.
3. Once parameter updates are received, the Aggregation Module aggregates the updates, i.e., averages the gradient updates to determine new model parameters.
4. The parameter server updates model parameters and sends updated values back to participants to update their local DNNs.

---

**Diagram Steps:**

1. **Conduct local training**
2. **LDP Module perturbs update**
3. **Private updates sent to k-Client Selection Module**
4. **Selection Module sends updates to Server**
5. **Selected updates are aggregated**
6. **New model parameters are sent back**
7. **Local models are updated**
• Individual participants locally define LDP-Module in LDP-Fed
  • Privacy guarantee, privacy mechanism parameters

• Privacy risk is not uniform:
  • Smaller datasets \([1]\)
  • Minority group representation \([2][3]\)

• Privacy requirements may not be uniform

<table>
<thead>
<tr>
<th>Target Population</th>
<th>Attack Accuracy ([^3])</th>
</tr>
</thead>
<tbody>
<tr>
<td>Aggregate</td>
<td>70.14%</td>
</tr>
<tr>
<td>Male Images</td>
<td>68.18%</td>
</tr>
<tr>
<td>Female Images</td>
<td>76.85%</td>
</tr>
<tr>
<td>White Race Images</td>
<td>62.77%</td>
</tr>
<tr>
<td>Racial Minority Images</td>
<td><strong>89.90%</strong></td>
</tr>
</tbody>
</table>


• Privacy requirement: guarantee $\alpha$-CLDP for each participant in FL training of DNN

• Must partition $\alpha$ into $E$ small budgets! (one for each of the $E$ total iterations) such that

$$\alpha = \sum_{i=0}^{E-1} \alpha_i$$

• Let $\theta_i$ = # of parameter updates to be uploaded to the parameter server at iteration $i$ and $\alpha_i$ be the allocated portion of the overall privacy budget. We then set

$$\alpha_p = \frac{\alpha_i}{|\theta_i|}$$

• $\alpha_p$ is the privacy budget when applying $\alpha$-CLDP to each parameter update in $\theta_i$
Basic implementation of $\alpha$-CLDP in FL divides the budget by (1) number of iterations and (2) number of parameters in the model:

$$\alpha_p = \frac{\alpha}{qE|\theta|}$$

Approach in $\alpha$-CLDP-Fed is to reduce (2) to only upload a subset of the parameters $\theta_i$ at each iteration and therefore increase the budget $\alpha_p$ (and corresponding accuracy) for parameters which are uploaded.

In LDP-Fed: $\theta_i$ corresponds to 1 layer of the DNN with earlier iterations updating later layers and proceeding iterations moving backward through the network.

Number of iterations and portion of the privacy budget allocated to an individual layer $\ell$ is directly proportionate to the size of that layer (with a minimum of 1 iteration).
LDP MODULE: CYCLES

- LDP-Fed cycles further control when different parameter updates are shared.
- Each cycle is implemented in terms of iteration rounds.

Let $c'$ = number of cycles. One cycle is then $\frac{E}{c'}$ rounds.

- Rounds within each cycle are then allocated to individual layers in the same manner, with number of rounds allocated being proportional to layer size.

- In LDP-Fed, the default cycle value is set to 5.
Let $\theta_i$ be the parameters selected for upload by the LDP Module at iteration $i$.

For each parameter $p \in \theta_i$ the LDP Module then applies the appropriate LDP Mechanism; for $\alpha$-CLDP-Fed...

### Exponential Mechanism

Exponential Noise Mechanism$^{[1]}$: Let $v \in \mathcal{U}$ be the raw user data, and let the Exponential Mechanism $\Phi_{EM}$ take as input $v$ and output a perturbed value in $\mathcal{U}$, i.e. $\Phi_{EM} : \mathcal{U} \rightarrow \mathcal{U}$. Then, $\Phi_{EM}$ that produces output $y$ with the following probability satisfies $\alpha$-CLDP.

$$\forall y \in \mathcal{U}: \Pr[\Phi_{EM}(v) = y] = \frac{e^{-\alpha \cdot d(v,y)}}{\sum_{z \in \mathcal{U}} e^{-\alpha \cdot d(v,z)}}$$

***Add noise to each parameter value to achieve $\alpha$-condensed local differential privacy***

Conventional FL systems do not query every participant in every round
  • Efficiency
  • Availability (WiFi, power, etc.)

Training in LDP-Fed: only $k \leq N$ participants' parameter updates selected per round

Discarded updates do not introduce any privacy cost

Sampling Amplification

Allows for a tighter bound of $\alpha = \sum_{i=0}^{E-1} q \cdot \alpha_i$ where $q = \frac{k}{N} \leq 1$. 
**RESULTS: LDP-FED ACCURACY IN DL VS LDP-BASIC**

\[ \alpha = 1.0 \]

- CLDP-Basic: below the random guess baseline of 10% \( \Rightarrow \) applying the privacy budget uniformly leads to untenable accuracy loss.
- LDP-Fed Single Layer approach significantly improves performance
- LDP-Fed’s proportionate budget and iteration allocation further improves accuracy by an additional \( \sim 2\% \)
\( \alpha = 1 \)

- Adversarially equivalent \( \epsilon \) computed from [1], \( \delta = 10^{-5} \)
- \( \alpha \)-CLDP-Fed outperforms DPSGD by \( \sim 6.8\% \) and Hybrid-One by \( \sim 3.5\% \)
- \( \Phi_{EM} \) in the LDP Module can be applied in parallel compared to cost of optimizer efficiency in DPSGD and Hybrid-One
- LDP-Fed requires no heavy cryptographic protocols
LDP-FED SYSTEM FEATURES

<table>
<thead>
<tr>
<th>Privacy-Preserving Federated Learning Method</th>
<th>Efficient</th>
<th>Locally Defined Privacy Guarantee</th>
<th>Protection from Inference Attacks</th>
<th>Handles Complex Models</th>
</tr>
</thead>
<tbody>
<tr>
<td>SMC [1]</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$\epsilon$-DP Parameter Sharing [2]</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Local Optimizer [3]</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Hybrid-One [4]</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$\alpha$-CLDP-Fed</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### Privacy-Preserving Federated Learning Method

- **Locally Defined Privacy Guarantee**
  - Each participant can locally define their privacy guarantee and privacy parameters.
  - Allows for adherence to local policies and compute restrictions.

- **Protection from Inference Attacks**
  - Protects from inference in the context of the complete training lifecycle.

- **Handles Complex Models**
  - LDP-Fed layer approach allows for the system to maintain utility and control noise with large models.

---


• LDP-Fed: a novel FL approach with communication efficient LDP
  • An edge system for distributed and collaborative training with a large population of clients
  • Participants efficiently train complex models + formal privacy protection
  • Participants customize their LDP privacy budget locally
• The \( \alpha \)-CLDP-Fed algorithm extends traditional LDP intended for single categorical values, to handle high dimensional, continuous, and large scale model parameter updates
• LDP-Fed parameter selection approach prevents LDP noise from overwhelming model updates \( \rightarrow \) balancing utility, privacy trade-off
• Comparison of LDP-Fed with the state-of-the-art privacy-preserving FL approaches in both accuracy and system features.