Towards Data and Model Confidentiality in Outsourced Machine Learning

A Dissertation submitted in partial fulfillment of the requirements for the degree of Doctor of Philosophy

Sagar Sharma
Data Intensive Analysis and Computing (DIAC) Lab

Dr. Keke Chen, Advisor

Committee Members:
Dr. Xiaoyu Liu,
Dr. Krishnaprasad Thirunarayan,
Dr. Junjie Zhang

Contents

I. Overview
   1. Outsourcing Scenarios
   2. Privacy Concerns
   3. Confidential Machine Learning (CML) in Outsourcing
   4. The DMC Process of Constructing CML Frameworks
   5. Challenges

II. My Work
   1. Scope and Contributions
   2. PrivateGraph: Confidential Spectral Analysis of Large Encrypted Graphs
   3. SecureBoost: Confidential Boosting with Random Linear Classifiers

III. Concluding Remarks

IV. Related Publications
I. Overview

Outsourcing scenarios ...

- Bigger data
- Several sources
- Several forms
  - Better / more complex analytics
  - Larger Storage
  - Larger Computation Resources
Outsourcing Scenarios..

Data Owner Outsourcing to a Public Cloud

Cloud provides:
1. A sense of infinite storage and computation resources!
2. 24/7 availability!

Data Owner’s benefits:
1. Data and computation intensive model learning
2. Improvement of services
   • Friend suggestions
   • Better movie recommendations
3. User targeting

Users benefits:
1. Storage and ML-services
   • health-care insights,
   • movie recommendations,
   • social networking

Outsourcing scenarios – A real-world example

KHealth: A health informatics provider
Privacy concerns

Outside the trusted boundary of the data owners, risks of following:

1. Private data theft and re-selling
2. Misuse of PII
3. Proprietary/intellectual model theft and misuse
4. Model-based attacks: Model to training data

Confidential Machine Learning (CML) in outsourcing

1. Privacy of sensitive data (feature values, labels, statistics);
2. Training powerful ML models with protected data;
3. Confidentiality of learned models - model-based attacks;
4. High-quality models and practical cost;
5. Fairly distributed workload amongst the involved parties.
Challenges

2. Differential Privacy (DP) does not meet the outsourcing requirements
   • Different threat model that involves untrusted data and model consumers
3. Hybrid composition with DMC promising but presents major challenges:
   • Decomposition of the target algorithm and successful mappings to crypto/privacy primitives;
   • Dealing with "crypto-unfriendly" components and algorithms;
   • Challenges of switching/mixing between different primitives (data encoding, precision, etc.).
4. Intrinsically expensive algorithms, such as DNN learning, and for massive data scalability, the DMC process becomes impractical. [Mohassel et al.]
5. Balancing trade-offs amongst simplicity of implementing algorithms, costs, quality of learned models, and cost distribution.
II. My Work

Thesis

Constructing confidential machine learning (CML) frameworks for outsourcing involves carefully choosing or designing algorithms that are crypto-friendly and mapping them to an assortment of cryptographic and privacy primitives to optimize the overall cost. However, when the target machine learning algorithms are intrinsically expensive or require massive data scalability, one may be forced to relax the desired security level and adopt the more efficient perturbation and transformation mechanisms to preserve the confidentiality of related data and models.
Scope and contributions

1. Three confidential frameworks for outsourcing:
   a. Unsupervised learning (spectral analysis)
   b. Supervised learning (boosting)
   c. Image-based deep learning

2. Features:
   a. Overcome the challenges of constructing CML frameworks with the DMC process;
   b. Apply alternative perturbation techniques when DMC becomes impractical;
   c. Preserve both data and model confidentiality along with high model quality (utility);
   d. Cost practicality and scalability for real-life settings and fair distribution of workload;
   e. Extensive empirical evaluation of cost, privacy, and utility;
   f. Backed by formal cost and security analysis, publicly available demos/working implementations, and published papers.

PrivateGraph: Confidential Spectral Analysis of Encrypted Private Graphs
PrivateGraph Framework

**PrivateGraph Framework**

**DMC Process:**
1. Chooses approximate algorithms
2. Maps Components to AHE/SHE, LWE masking and DP

**O(n^3)**
1. Replaces expensive eigen-decomposition with approximate algorithms of **Lanczos** and **Nystrom**
2. Provable privacy and practical results
3. Storage and O(n^3) operations outsourced to cloud
4. Minimal data owner and contributors’ involvement (O(n) operations only)
5. The final model known only to the data-owner.

**Confidential eigen-approximation**

**Lanczos Decomposition and Mapping**

1. Generate random vector b_0
2. For i=1 to t compute \( b_i = Ab_i \) → cost O(N^2) in the cloud
   other vector based operations → cost O(N)
3. End for
4. Postprocessing \( (b_0, ..., b_t) \), to get top-k eigenvectors → cost O(N)
   \( k, t \ll N \), e.g., \( k=10 \), and \( t=30 \)

Crypto-friendly E(\( Ab \)) homomorphically
With AHE: \( b_i \) unencrypted; \( b'_i = b_i + \lambda \)

\( \lambda \leftarrow \) novel noise vector generation
- O(n) cost to recover \( Ab \)
- indistinguishable from uniform random noise vector, based on learning with error (LWE).

Crypto-unfriendly Euclidian norm involving square roots performed locally over plaintext
Confidential eigen-approximation

Nystrom Decomposition and Mapping

1. Randomly shuffle rows/columns: switching both i and j rows, and i and j columns at the same time.
2. Decompose $W$ to get top-k eigenvectors in $U_{m\times k}$ and top-k diagonal eigenvalue matrix $L_{k\times k}$.
3. Compute $CU^4$ to get top-eigenvectors of $A$.

Possible exposure of the eigenstructure of $W$ through $U_{m\times k}L_{k\times k}^{-1}U_{k\times m}^T$; set $V = U_{m\times k}L_{k\times k}^{-1}$.

Random-Projection Perturbation $V'_{m\times k} = (V + \Delta)R$

$\Delta$ is a uniformly random matrix
$R$ is a $k \times k$ invertible random matrix.

Optimization: Sparse submission

How many fake $E(0)$ elements?

1. Bin-based DP to balance between sparsity and privacy.
2. To satisfy both node and edge-level privacy.
3. Bins determined by % of nodes ---- enough nodes in each bin, e.g., 50 per bin.
4. Noise level depends on $\epsilon$ and $\Delta = \max(F(A_1) - F(A_2))$ which is reduced by our binning.
Cost Analysis

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>Cloud</th>
<th>Data Owner</th>
<th>Comm. cost</th>
</tr>
</thead>
<tbody>
<tr>
<td>Lan-AHE</td>
<td>$O(tN^2)$</td>
<td>$O((t + h)N)$</td>
<td>$O((t + h)N)$</td>
</tr>
<tr>
<td>Lan-SHE</td>
<td>$O(tN^2)$</td>
<td>$O(2tN)$</td>
<td>$O(2tN)$</td>
</tr>
<tr>
<td>Ny-AHE</td>
<td>$O(Nkm)$</td>
<td>$O(m^2 + mk^2 + Nk)$</td>
<td>$O(2Nk + 2km + m^2)$</td>
</tr>
<tr>
<td>Ny-SHE</td>
<td>$O(Nkm)$</td>
<td>$O(m^2 + mk^2 + Nk)$</td>
<td>$O(Nk + km + m^2)$</td>
</tr>
</tbody>
</table>

N --- number of nodes  
k --- number of top eigen-vectors  
t --- Lanczos iterations  
m --- Nystrom samples  
h --- security parameter in LWE masking 80 for 80 bit security

Achieves the design goal: Cloud – $O(n^2)$, and client – $O(n)$

Results Summary

A. Clustering accuracy; $t$ – Lanczos iteration, $m$ – sampling size for Nystrom

<table>
<thead>
<tr>
<th>Datasets</th>
<th>N</th>
<th>Accuracy</th>
<th>m</th>
<th>t</th>
</tr>
</thead>
<tbody>
<tr>
<td>Facebook</td>
<td>3999</td>
<td>82%</td>
<td>396</td>
<td>30</td>
</tr>
<tr>
<td>Twitter</td>
<td>76244</td>
<td>90%</td>
<td>3080</td>
<td>25</td>
</tr>
<tr>
<td>Gplus</td>
<td>102100</td>
<td>92%</td>
<td>8168</td>
<td>30</td>
</tr>
</tbody>
</table>

B. Increase in number of edges due to sparse encoding ($\epsilon=1.0$)

| Dataset | nbins | nodes/bin | orig. $|E|$ | pert. $|E|$ | % Inc. |
|---------|-------|-----------|-------|-------|-------|
| Facebook| 100   | 40        | 84243 | 99965 | 18.66 |
| Twitter | 1000  | 76        | 1242390| 1527286| 22.93 |
| GPlus   | 2000  | 52        | 12113501| 13228599| 9.21  |

Compare this to $N^2$  
15,673,681 for FB  
155 times
Results Summary

C. Data owner’s costs

D. Storage at Cloud

<table>
<thead>
<tr>
<th>Format</th>
<th>Facebook</th>
<th>Twitter</th>
<th>GPlus</th>
</tr>
</thead>
<tbody>
<tr>
<td>dense</td>
<td>3.78 GB</td>
<td>1.35 TB</td>
<td>2.43 TB</td>
</tr>
<tr>
<td>sparse</td>
<td>24.41 MB</td>
<td>372.87 MB</td>
<td>3.15 GB</td>
</tr>
</tbody>
</table>

Saving ratio 155 3.621 771

Results for AHE schemes
RLWE schemes faster but heavier storage/communication

SecureBoost: Confidential Boosting with Random Linear Classifiers
SecureBoost Framework

1. Random linear classifiers (RLCs) to simplify and reduce the protocols
2. Fully protects users’ data (both feature values and labels) and the learned models
3. Only a minimum Leakage function released to CSP
4. Two constructions of SecureBoost: HE + GC and SecSh + GC

Decision stumps – not “crypto-friendly”

Training:

1. Examine all possible thresholds/splits for all variables
2. Select attribute and threshold that maximizes Information Gain (ID3)
   \[ \text{Info Gain} = S(\text{parent}) - \{\text{Avg.}(S(\text{class1}), S(\text{class 2}))\} \]
3. No DS implementation over encrypted data available.
Random Linear Classifiers (RLC) – more crypto-friendly

1. Boundary planes $w^T x + b$
2. Random w and b
3. $w^T x + b > 0$ then class +1; otherwise class -1.

*With standardized data, $b \in [-2, 2]$, and each element of $w \in [-1, 1]^k$, a valid RLC in 1~2 tries i.e. accuracy > 50%

Decomposition and Mapping

For $X_{x \times n}$ training dataset, Form a matrix Z such that $z_{ij} = x_i y_i$

Only 1 matrix–vector multiplication $\rightarrow$ AHE/SHE
Only n comparisons! $\rightarrow$ Minimum GC

Cost Analysis

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>HE+GC</td>
<td>UserCloud</td>
<td>$O(nk)$</td>
<td>-</td>
<td>$O(pk)$</td>
<td>$O(nk)$</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td>CSP</td>
<td>$O(pn)$</td>
<td>$O(pn)$</td>
<td>-</td>
<td>$O(pn)$</td>
<td>-</td>
<td>$O(nk)$</td>
</tr>
<tr>
<td>SecSh+GC</td>
<td>UserCloud</td>
<td>$O(pk)$</td>
<td>-</td>
<td>$O(pn)$</td>
<td>$O(pk)$</td>
<td>-</td>
<td>$O(nk)$</td>
</tr>
<tr>
<td></td>
<td>CSP</td>
<td>$O(pn)$</td>
<td>-</td>
<td>$O(pk)$</td>
<td>$O(p(n + k))$</td>
<td>-</td>
<td>$O(pk)$</td>
</tr>
</tbody>
</table>

n --- number of training records
k --- number of training dimensions
p --- number of RLCs tried
b --- number of encoding bits

**HE + GC**
1. Cloud takes all of storage
2. Cloud-heavy CSP-light computations

**SecSh + GC**
1. Cloud – CSP equal storage
2. CSP takes a significant workload
3. Users benefit the most, no encryption cost!
Security Analysis: The leakage function $I_t$

$I_{t,i} = (h_t(x_i) == y_i)$ for each RLC $t = 1 \ldots p$ leaked to CSP

Bit $I_{t,i}$ indicate if the $t^{th}$ RLC correctly classifies sample $x_i$

Implications:

After $p$ iterations, for each sample $x_i$, a characterization vector (CV) is formed

![Diagram of characterization vectors]

An intuitive attack:
- If $c_2 == c_4$, $x_1 = x_4$
- Closeness of $c_i$ and $c_j$ $\rightarrow$ closeness of $x_i$ and $x_j$

Security Analysis: The leakage function $I_t$

Average Euclidean distance between record-pairs generating CV pairs differing by $k$ bits

Similarity of CVs does not infer similarity of training records!
Results Summary

A. Datasets properties

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Instances</th>
<th>Attributes</th>
<th>Adaboost Accuracy</th>
<th>Number of decision stumps</th>
</tr>
</thead>
<tbody>
<tr>
<td>ionosphere</td>
<td>351</td>
<td>34</td>
<td>92.02% +/- 4.26%</td>
<td>50</td>
</tr>
<tr>
<td>credit</td>
<td>1,000</td>
<td>24</td>
<td>74.80% +/- 3.50%</td>
<td>100</td>
</tr>
<tr>
<td>spambase</td>
<td>4,601</td>
<td>57</td>
<td>92.31% +/- 4.40%</td>
<td>75</td>
</tr>
<tr>
<td>epileptic</td>
<td>11,500</td>
<td>179</td>
<td>86.95% +/- 3.40%</td>
<td>200</td>
</tr>
<tr>
<td>synthetic</td>
<td>150,000</td>
<td>10</td>
<td>89.51% +/- 2.10%</td>
<td>75</td>
</tr>
</tbody>
</table>

B. Convergence of SecureBoost

![Graph showing convergence of SecureBoost](image)

About 200 base classifiers!

Results Summary

C. Boosting with DS vs. with RLC

![Comparison of model quality](image)

D. Boosting with different linear classifiers

![Comparison of model convergence](image)
Results Summary
E. Comparison with the state-of-the-art SecureML

Disguised-Nets: Image Disguising for Confidential Outsourced Deep Learning
Confidential DNN Learning

1. HE or GC based solutions (DMC process) are impractical and limited to evaluations only.
   - SecureML [Mohassel et al.]
   - CryptoNets [Xie et al.]
2. Cloud-client partitioning: Vulnerable to GAN-based attacks [Hitaj et al.]

Visual Re-identification Attack

Aimed at compromising the visual privacy of protected images

Visual privacy = (1 – accuracy of the DNN examiner)
Class-membership Attack

Learn weather a certain class of images was used as training examples

1. Given a DNN model and known output labels \( \{c | c \in C\} \);
2. Probe the model with a set of images, \( \{t_i | i = 1..m\} \), in a target class \( c \);
3. Observe model’s output to determine if \( c \in \) i.e. if \( c \) is in-training;

Any deep learning framework that exposes the model (a good quality model) is subject to this attack.

Our Idea

1. Transform image blocks with random orthogonal / projection matrices
2. Pseudo-random permutation (\( \pi \)) of image blocks
3. Additive Perturbation with a noise matrix
Disguised-Nets Framework

1. High-quality models
2. Practical Cost
3. Resiliency to GAN-based, visual re-identification, and class-membership attacks
4. No need for one to alter or tailor the existing DNN architectures

Image disguising mechanisms

Some MNIST samples
Image disguising mechanisms

Some CIFAR-10 samples

Results summary

A. Parameter settings and CNN architectures

<table>
<thead>
<tr>
<th>Datasets</th>
<th>Mechanisms</th>
<th>Block size</th>
<th>Noise Level</th>
<th>Architecture</th>
</tr>
</thead>
<tbody>
<tr>
<td>MNIST</td>
<td>block-wise RMT + Permutation</td>
<td>7 x 7</td>
<td>100</td>
<td>Simple</td>
</tr>
<tr>
<td>FASHION</td>
<td>block-wise RMT + Permutation</td>
<td>7 x 7</td>
<td>100</td>
<td>Simple</td>
</tr>
<tr>
<td>CIFAR-10</td>
<td>block-wise RMT</td>
<td>2 x 2</td>
<td>25</td>
<td>ResNet</td>
</tr>
<tr>
<td>LFW</td>
<td>block-wise RMT</td>
<td>2 x 2</td>
<td>50</td>
<td>ResNet</td>
</tr>
</tbody>
</table>

B: Results of applying image disguising mechanisms

<table>
<thead>
<tr>
<th>Datasets</th>
<th>Model Accuracy With Disguise</th>
<th>Model Accuracy Without Disguise</th>
</tr>
</thead>
<tbody>
<tr>
<td>MNIST</td>
<td>96.6 +/- 0.4%</td>
<td>96.7 +/- 0.2%</td>
</tr>
<tr>
<td>FASHION</td>
<td>85.1 +/- 0.6</td>
<td>88.7 +/- 0.3%</td>
</tr>
<tr>
<td>CIFAR-10</td>
<td>89.3% +/- 0.1%</td>
<td>93.4% +/- 0.2%</td>
</tr>
<tr>
<td>LFW</td>
<td>90.6 +/- 1.3%</td>
<td>94.3 +/- 2.0%</td>
</tr>
</tbody>
</table>

C: Associated cost

- **Insufficient transformation cost!**
  - A few milliseconds per image
- **Low communication and storage cost**
  - Only 2 to 5 times the original image size
Results summary

D. Model Quality

Results summary

E. Resilience to Visual Re-identification Attack
Results summary

F. Resilience to Class-membership Attack

**Effective on unprotected models.**
In-training class-wise Fano factor is significantly higher.

**Ineffective on Disguised-Nets models.**
In-training class-wise Fano factor is indistinguishable from that of Out-training.

III. Concluding Remarks
Conclusion

1. Construction of CML frameworks for outsourcing crucial but not straight-forward and faces several challenges.

2. Three CML frameworks with these desired characteristics:
   a) Preserve both data and model confidentiality, generate robust models
   b) Cost practicality and scalability for real-life settings
   c) Fair work-load distribution between the involved parties
   d) Backed with cost and security analysis, and working demos and prototypes

3. Crypto-friendliness of the chosen algorithms and their components essential for practicality of the CML frameworks.

4. Reliance on cheaper perturbation techniques for intrinsically expensive algorithms and massive scalability.

Future Research Directions

1. Use of emerging trusted execution environment (TEE) hardware solutions in constructing CML frameworks;
2. Real-life adaptation of confidential frameworks (e.g. Privacy + AI + Healthcare);
3. Theoretical explanation of Disguised-Nets-like perturbation techniques and their potentiality in other learning objectives;
4. Further analysis on resilience of the the perturbation mechanisms;
5. An easy-to-use common library for CML algorithms;
6. In-depth study of our proposed class-membership attacks.
Related Publications

Sagar Sharma and Keke Chen, “Confidential Boosting with Random Linear Classifiers for Protected User-generate Data” Accepted by ESORICS, Luxembourg 2019

Sagar Sharma, James Powers, and Keke Chen “PrivateGraph: Privacy-Preserving Spectral Analysis of Encrypted Graphs in the Cloud”, IEEE TKDE, 2018

Sagar Sharma, Keke Chen, and Amit Sheth, “Towards Practical Privacy-Preserving Analytics for IoT and Cloud-Based Healthcare Systems”, IEEE Internet Computing, 2018

Sagar Sharma and Keke Chen, “Image Disguising for Privacy-preserving Deep Learning”, ACM CCS Poster Session, Toronto 2018

Sagar Sharma and Keke Chen, Privacy-Preserving Boosting with Random Linear Classifiers, ACM CCS Poster Session, Toronto 2018


ACKNOWLEDGEMENT

Dr. Keke Chen
Advisor

Dr. Xiaoyu Liu

Dr. Krishnaprasad Thirunarayan
Committee Members

Dr. Junjie Zhang

And friends and family.
Questions/Comments.