Privacy Attacks on Decentralized Learning Systems that Exchange Chunked Models

Author: Halil Betmezoğlu
Prof.: Jérémie Decouchant
Supervisor: Bart Cox

1 Motivation & Background

Decentralized Learning: Nodes train on local datasets and share model parameters with neighbors.

Model Chunking: Model parameters are split into smaller parts before sharing [1].

Problem: Keeping sensitive data local does not guarantee privacy. Information can still be leaked through shared chunked model parameters. Potential privacy attacks are *membership inference* and *linkability*.

Objectives

1. Enhance the linkability attack using Hungarian matching and limited data access
2. Evaluate privacy impact of model chunking against membership inference and linkability.
3. Compare static, cyclic, and random chunking methods under various conditions.

2 Methodology

Attacks

- **Membership Inference**: Determine if a specific sample was in the training set of a node using shadow models [2].
- **Linkability**: Infer the origin node of a chunk [1, 3].

Chunking Methods

- **Static**: Share the same chunk to each neighbor every round.
- **Cyclic**: Rotate the chunk that is shared to a neighbor over rounds.
- **Random**: Randomly select which chunk to share.
- **None**: Share all model parameters.

Experiments

- 16-node topologies:
  - 3-regular for membership inference
  - 8-regular for linkability
- Lenet model
- MNIST and CIFAR-10 datasets

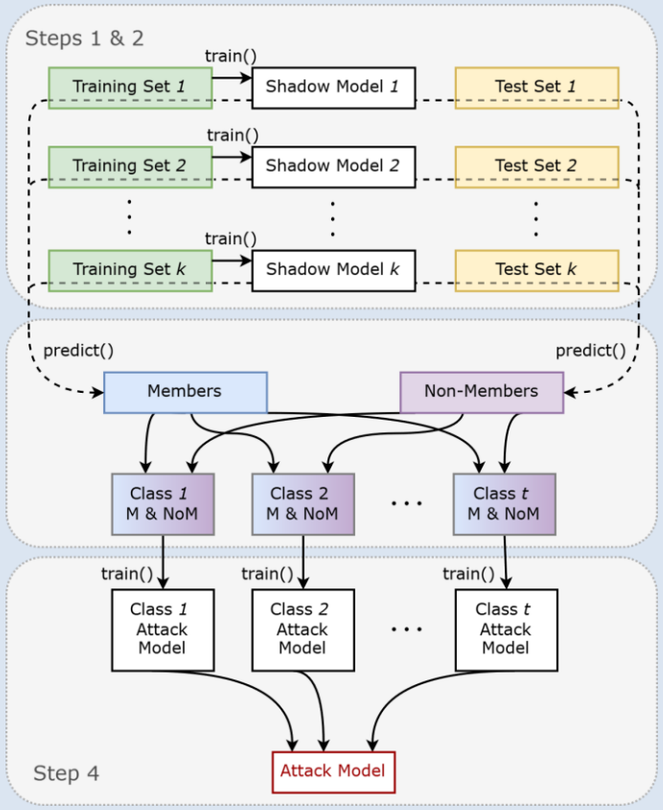
3 Privacy Attacks

Membership Inference

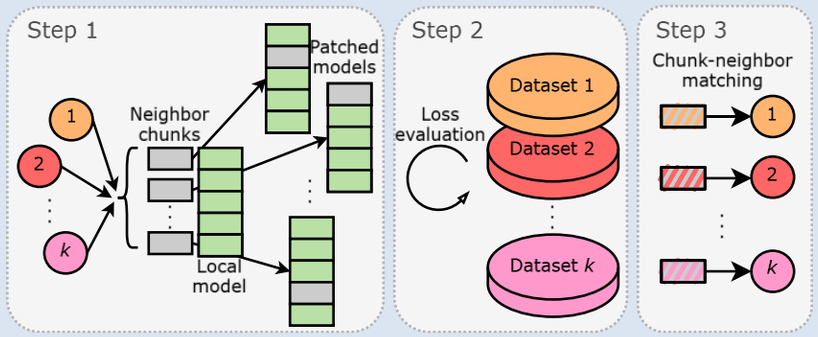
Goal: Determine if a sample was in the training set of the model.

Linkability

Goal: Determine which node a given chunk has originated from.
Chunk-neighbor matching: Minimum-loss and Hungarian matching can be used.



↓ **Figure:** Linkability Attack. ↓ ↑ **Figure:** Membership Inference Attack. ↑



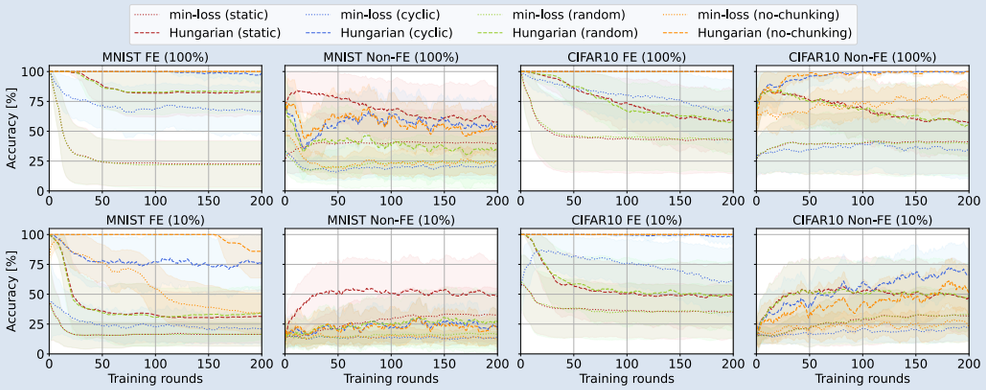
4 Results & Discussion

Membership Inference Attack Results

Chunking Method	Average AUC MNIST (%)	Average AUC CIFAR-10 (%)	Average AUC non-FE (%)	Average AUC FE (%)
Static	55.52 ± 0.41	82.89 ± 0.55	60.48 ± 0.47	77.93 ± 0.49
Cyclic	50.14 ± 0.92	78.78 ± 3.42	58.09 ± 2.22	70.82 ± 2.12
Random	54.70 ± 0.86	82.57 ± 0.46	60.76 ± 0.38	76.50 ± 0.94
None	50.20 ± 0.56	54.70 ± 0.86	57.42 ± 1.04	71.40 ± 1.09

- Chunking increases vulnerability to membership inference, especially with static and random chunking

Linkability Attack Results



- Static and random chunking reduce linkability with full epochs.
- Hungarian matching consistently outperforms minimum-loss matching.

In both attacks:

- Avoiding full epochs reduces attack performance.
- Cyclic chunking and no chunking show similar behavior, as do static and random chunking.
- Attack performance depends on data heterogeneity and chunking strategy.

5 Limitations

- Lack of randomness in MNIST limits attack effectiveness.
- Both datasets have 10 classes. A higher number of classes would improve attack performance.
- LeNet model lacks complexity, especially for CIFAR-10 data.
- Experiments use IID data, non-IID conditions would improve attack accuracy.
- Shadow models rely on a disjoint subset of the data, a more realistic attack could create its own dataset through target model queries.
- Low model accuracies in experiments with CIFAR-10, which can be improved by running with more data and iterations.

6 Conclusion

Model chunking does not eliminate privacy risks. It reduces linkability under certain conditions but increases vulnerability to membership inference.

Future work can investigate other privacy attacks, expand work on membership inference and linkability, and investigate other defenses.

[1] S. Biswas, M. Even, A. - M. Kermarrec, L. Massoulié, R. Pires,R. Sharma, and M. de Vos. "Noiseless Privacy-Preserving De-centralized Learning". In: *Proceedings on Privacy Enhancing Technologies* 2025.1 (Jan. 2025), pp. 824–844.
[2] R. Shokri, M. Stronati, C. Song, and V. Shmatikov. *Membership Inference Attacks against Machine Learning Models.* 2017.
[3] T. Lebrun, A. Boutet, J. Aalmoes, and A. Baud. "MixNN: protection of federated learning against inference attacks by mixing neural network layers". In: *Proceedings of the 23rdACM/IFIP International Middleware Conference.* Middleware'22. ACM, Nov. 2022, pp. 135–147.