Towards Sybil Resilience in Decentralized Learning

Thomas Werthenbach
Delft University of Technology
Delft, The Netherlands
T.A.K.Werthenbach@student.tudelft.nl

Johan Pouwelse
Delft University of Technology
Delft, The Netherlands
J.A.Pouwelse@tudelft.nl

***Abstract*—**

***Index Terms*—decentralized learning, sybil attack, sybil resilience**

I. Introduction

The rise of machine learning has resulted in an increasing number of everyday-life intelligent applications. As such, machine learning has been used in personal assistants [1], recommendation in social media [2] and music [3], and cybersecurity [4]. However, accurate machine learning models require large training datasets [5], [6], which can often be hard to obtain and store due to recent privacy legislation [7]. Federated learning [8] has become a promising alternative and widely adopted tool for crowd sourcing computationally expensive machine learning operations, reportedly having been used for training numerous industrial machine learning models [9]–[13]. Federated learning ensures the protection of privacy, as the user's data will not leave their device during training.

With federated learning, in contrast to centralized machine learning, training takes place on the end-users' personal devices, which are often referred to *edge devices* or *nodes*. The resulting trained models are communicated to a central server, which will combine these models using some predefined aggregation method. Such an approach offers several advantages. First, as aforementioned, by only communicating the trained models to the central server, the user's privacy is respected, while obtaining comparable performance compared to centralized machine learning [14]. While there exist attacks in which training data can be reconstructed based on the gradient of the trained models [15], [16], defense mechanisms against this attack have been proposed [17], [18]. Secondly, federated learning improves communication efficiency. As all nodes locally train on their own data, their data does not need to be communicated to the central server. It has been shown that this significantly reduces communication costs [8].

However, federated learning suffers from some disadvantages. For example, while federated learning enjoys significant improvement in terms of communication costs compared to centralized learning, the central server still aggregates the models of all participating nodes, inducing heavy communication costs and a potential bottleneck in the learning process affecting the overall convergence time [19]. Secondly, the scalability in terms of the amount of nodes heavily varies depending on the aggregation method. In secure and robust federated learning aggregation methods, the incorporation of additional nodes during aggregation may result in significantly increased computational effort for the central server [20]. Thirdly, the central server performing the aggregation poses as a single-point of failure [21]. Downtime of the central server will disrupt the overall model training process, which is especially inconvenient in architectures where edge devices are awaiting the aggregated model before starting the next training epoch. Decentralized learning, also commonly referred to as *decentralized federated learning* or *gossip learning*, is a technique aiming to resolve these issues. In decentralized learning, there is no central server performing the aggregation and the edge devices form a distributed network, e.g. a peer-to-peer network, in which each node individually performs the aggregation (see Figure 1). While the information available during aggregation is more limited relative to federated learning, it has been shown that decentralized learning has the potential to obtain similar results compared to federated learning [22]. Models are exchanged between individual devices and aggregated by each edge device depending on the aggregation method, alleviating the communicative bottleneck and single point of failure issues imposed on federated learning.

While decentralized learning solves the scalability challenges faced in federated learning, it is still vulnerable to byzantine environments [23]. Since the predefined aggregation method in decentralized learning does not have access to all models in the network, aggregation is performed with less information compared to federated learning, resulting in relatively less resistance against possible poisoning attacks [24]. Broadly studied poisoning attacks include the backdoor attack [25]–[27] and the label-flipping attack [28], [29]. The effect of these attacks can amplified through combining them with the Sybil attack [30], in which an adversary controls numerous nodes to increase its influence. For example, an adversary may deploy the Sybil attack to rapidly spread their poisoned model through the network. In this paper, we focus primarily on the use of Sybil attacks in Poisoning Attacks.

Prior work on resilience against Poisoning attacks combined with Sybil attacks in distributed machine learning has only been done in federated learning settings. One popular example of such work is *FoolsGold* [31], which aims to increase Sybil resilience through the assumption that all Sybils will broadcast similar gradients during each round of training. By dynamically adapting the aggregation weight of peers' models based on their similarity with others, *FoolsGold* shows promising results on the protection against the Sybil attack.

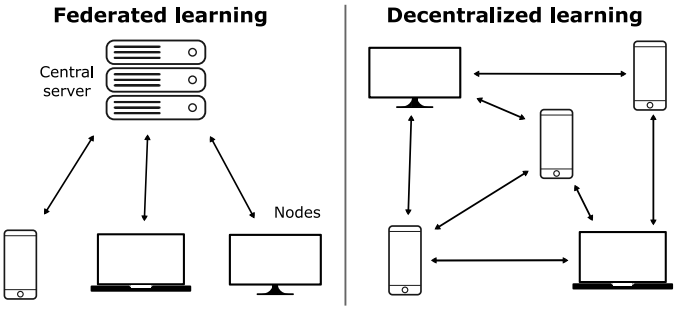

Fig. 1. Federated learning compared to decentralized learning. Arrows represent a connection between two nodes and indicates the two connecting nodes share model updates with eachother.

In this work, we first explore *FoolsGold*'s applicability to decentralized learning, using different types of Sybil attacks (classified in Section V). We then continue by introducing a novel algorithm based on *FoolsGold*'s intuitions with adaptations for increased performance in decentralized environments. More specifically, through the utilization of a probabilistic gossip mechanism knowledge spreading. Finally, we empirically evaluate this algorithm on numerous types of Sybil attacks and show its ability to obtain increased Sybil resilience.

To the best of our knowledge, there exists only a single prior work on defensive algorithms against poisoning attack in decentralized learning [32]. Furthermore, this paper is the first to study Sybil attacks in decentralized learning. In short, our contributions are the following:

- We evaluate FoolsGold, a popular Sybil resilience algorithm in federated learning, and assess its compatibility with decentralized learning in Section III.
- We identify the possible Sybil attacks in decentralized learning and illustrate their effectiveness in Section V.
- We present a pioneering algorithm for Sybil resilience in decentralized learning in Section VII, provide an empirical evaluation in Section VIII, and perform a theoretical analysis in Section IX.

II. Background

A. Federated learning

- Explain more in-depth how federated learning works → formal definitions?
- Refer to Figure 1
- Explore some implementations of popular (simple) FL algorithms.

B. Decentralized learning

- Explain more in-depth how decentralized learning works → formal definitions?
- Refer to Figure 1
- Explore some implementation of popular (simple) DL algorithms.

C. The Sybil attack

- Formal definition of Sybil attack
- In our context, most Sybil attacks may use botnets to increase their reachability and network throughput.
- Seuken and Parks on strongly and weakly benificial Sybil attacks.

III. Related work

A. FoolsGold

Explain FoolsGold [31] and show two graphs in which FoolsGold is used in both federated and decentralized settings (and show that it does not work as well in decentralized learning if there is no more than a single attack edge to every honest node).

How our work is different:

- It can be deployed in decentralized learning.
- It suffers less from the computationally expensive aggregation method. According to Foolsgold's authors, the cosine similarity function was the most expensive operation.

Furthermore, we performed an extensive evaluation of FoolsGold in both federated learning and decentralized learning. These are our results...

B. Resilient Averaging Gradient Descent

Resilient Averaging Gradient Descent (RAGD) [32] is a novel algorithm for dealing with dealing with poisoning attacks in decentralized learning.
How our work is different:

- RAGD naively assumes that Sybil model updates will be quite different compared to honest model, but this may not necessarily be the case for label-flipping attacks or backdoor attacks.
- RAGD assumes the existence of a static adjacency matrix, defining the edge weights between any two nodes. It also assumes that any attack edge has a weight of $0 < \epsilon < \frac{1}{2}$.
- We assume that nodes will not be fully connected, similar to real decentralized networks, which means that the most successful attacks will likely use Sybils in the form of a botnet.

C. Multi-Krum?

Distance based

D. Bristle?

IV. Preliminaries

1) We assume that there exists some incentive for utilizing Sybils. This may be an upper bound on the maximum amount of connections any node can have with other nodes. An alternative may be a communication bottleneck, such as network speed, which incentivizes the use of a botnet as sybils to help distribute the poisoned model more rapidly.

V. Sybil attacks in decentralized learning

VI. Threat model

VII. Design

- Explain FoolsGold (cannot assume everyone knows it)
- Pseudocode?
- Explain gossiping models → the probabilistic property occurs two-fold, ① when selecting a peer to request a model from and ② when selecting what model to send to the requesting peer.
- Add figure

VIII. Evaluation

A. Experimental setup

- DAS6 → IPv8 → Gumby
- Attacks:
  - Label-flipping attack. from [31], [33]
  - backdoor attack. from [31]
  - a little is enough? from [33]
  - fall of empires? from [33]
  - sign-flipping? from [33]
- Datasets:
  - ImageNet
  - Cifar-10
  - MNIST spin-off
  - Some other dataset
- Comparison algorithms:
  - FoolsGold
  - FedAvg
  - A few more...

B. Results

IX. Analysis

X. Discussion

XI. Conclusion

References

[1] E. V. Polyakov, M. S. Mazhanov, A. Y. Rolich, L. S. Voskov, M. V. Kachalova, and S. V. Polyakov, "Investigation and development of the intelligent voice assistant for the internet of things using machine learning," in *2018 Moscow Workshop on Electronic and Networking Technologies (MWENT)*, 2018, pp. 1–5.

[2] B. T.K., C. S. R. Annavarapu, and A. Bablani, "Machine learning algorithms for social media analysis: A survey," *Computer Science Review*, vol. 40, p. 100395, 2021. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1574013721000356

[3] X. Wang and Y. Wang, "Improving content-based and hybrid music recommendation using deep learning," in *Proceedings of the 22nd ACM International Conference on Multimedia*, ser. MM '14. New York, NY, USA: Association for Computing Machinery, 2014, p. 627–636. [Online]. Available: https://doi.org/10.1145/2647868.2654940

[4] S. A. Salloum, M. Alshurideh, A. Elnagar, and K. Shaalan, "Machine learning and deep learning techniques for cybersecurity: A review," in *Proceedings of the International Conference on Artificial Intelligence and Computer Vision (AICV2020)*, A.-E. Hassanien, A. T. Azar, T. Gaber, D. Oliva, and F. M. Tolba, Eds. Cham: Springer International Publishing, 2020, pp. 50–57.

[5] J. Prusa, T. M. Khoshgoftaar, and N. Seliya, "The effect of dataset size on training tweet sentiment classifiers," in *2015 IEEE 14th International Conference on Machine Learning and Applications (ICMLA)*, 2015, pp. 96–102.

[6] J. Hestness, S. Narang, N. Ardalani, G. F. Diamos, H. Jun, H. Kianinejad, M. M. A. Patwary, Y. Yang, and Y. Zhou, "Deep learning scaling is predictable, empirically," *CoRR*, vol. abs/1712.00409, 2017. [Online]. Available: http://arxiv.org/abs/1712.00409

[7] A. Goldsteen, G. Ezov, R. Shmelkin, M. Moffie, and A. Farkash, "Data minimization for gdpr compliance in machine learning models," *AI and Ethics*, pp. 1–15, 2021.

[8] B. McMahan, E. Moore, D. Ramage, S. Hampson, and B. A. y. Arcas, "Communication-Efficient Learning of Deep Networks from Decentralized Data," in *Proceedings of the 20th International Conference on Artificial Intelligence and Statistics*, ser. Proceedings of Machine Learning Research, A. Singh and J. Zhu, Eds., vol. 54. PMLR, 20–22 Apr 2017, pp. 1273–1282. [Online]. Available: https://proceedings.mlr.press/v54/mcmahan17a.html

[9] J. Janai, F. Güney, A. Behl, A. Geiger *et al.*, "Computer vision for autonomous vehicles: Problems, datasets and state of the art," *Foundations and Trends® in Computer Graphics and Vision*, vol. 12, no. 1–3, pp. 1–308, 2020.

[10] P. Navarro, C. Fernández, R. Borraz, and D. Alonso, "A machine learning approach to pedestrian detection for autonomous vehicles using high-definition 3d range data," *Sensors*, vol. 17, no. 12, p. 18, Dec 2016. [Online]. Available: http://dx.doi.org/10.3390/s17010018

[11] A. Hard, K. Rao, R. Mathews, F. Beaufays, S. Augenstein, H. Eichner, C. Kiddon, and D. Ramage, "Federated learning for mobile keyboard prediction," *CoRR*, vol. abs/1811.03604, 2018. [Online]. Available: http://arxiv.org/abs/1811.03604

[12] T. Yang, G. Andrew, H. Eichner, H. Sun, W. Li, N. Kong, D. Ramage, and F. Beaufays, "Applied federated learning: Improving google keyboard query suggestions," *CoRR*, vol. abs/1812.02903, 2018. [Online]. Available: http://arxiv.org/abs/1812.02903

[13] M. Chen, R. Mathews, T. Ouyang, and F. Beaufays, "Federated learning of out-of-vocabulary words," *CoRR*, vol. abs/1903.10635, 2019. [Online]. Available: http://arxiv.org/abs/1903.10635

[14] Y. Cheng, Y. Liu, T. Chen, and Q. Yang, "Federated learning for privacy-preserving ai," *Communications of the ACM*, vol. 63, no. 12, pp. 33–36, 2020.

[15] L. Lyu and C. Chen, "A novel attribute reconstruction attack in federated learning," *CoRR*, vol. abs/2108.06910, 2021. [Online]. Available: https://arxiv.org/abs/2108.06910

[16] H. Yang, M. Ge, K. Xiang, and J. Li, "Using highly compressed gradients in federated learning for data reconstruction attacks," *IEEE Transactions on Information Forensics and Security*, vol. 18, pp. 818–830, 2023.

[17] H. S. Sikandar, H. Waheed, S. Tahir, S. U. R. Malik, and W. Rafique, "A detailed survey on federated learning attacks and defenses," *Electronics*, vol. 12, no. 2, 2023. [Online]. Available: https://www.mdpi.com/2079-9292/12/2/260

[18] P. Qiu, X. Zhang, S. Ji, Y. Pu, and T. Wang, "All you need is hashing: Defending against data reconstruction attack in vertical federated learning," 2022. [Online]. Available: https://arxiv.org/abs/2212.00325

[19] J. Hamer, M. Mohri, and A. T. Suresh, "FedBoost: A communication-efficient algorithm for federated learning," in *Proceedings of the 37th International Conference on Machine Learning*, ser. Proceedings of Machine Learning Research, H. D. III and A. Singh, Eds., vol. 119. PMLR, 13–18 Jul 2020, pp. 3973–3983. [Online]. Available: https://proceedings.mlr.press/v119/hamer20a.html

[20] S. Kadhe, N. Rajaraman, O. O. Koyluoglu, and K. Ramchandran, "Fastsecagg: Scalable secure aggregation for privacy-preserving federated learning," *CoRR*, vol. abs/2009.11248, 2020. [Online]. Available: https://arxiv.org/abs/2009.11248

[21] Y. Qi, M. S. Hossain, J. Nie, and X. Li, "Privacy-preserving blockchain-based federated learning for traffic flow prediction," *Future Generation Computer Systems*, vol. 117, pp. 328–337, 2021. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0167739X2033065X

[22] I. Hegedűs, G. Danner, and M. Jelasity, "Decentralized learning works: An empirical comparison of gossip learning and federated learning," *Journal of Parallel and Distributed Computing*, vol. 148, pp. 109–124, 2021. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0743731520303890

[23] J. Hou, F. Wang, C. Wei, H. Huang, Y. Hu, and N. Gui, "Credibility assessment based byzantine-resilient decentralized learning," *IEEE Transactions on Dependable and Secure Computing*, pp. 1–12, 2022.

[24] V. Tolpegin, S. Truex, M. E. Gursoy, and L. Liu, "Data poisoning attacks against federated learning systems," in *Computer Security – ESORICS*

2020, L. Chen, N. Li, K. Liang, and S. Schneider, Eds. Cham: Springer International Publishing, 2020, pp. 480–501.

[25] E. Bagdasaryan, A. Veit, Y. Hua, D. Estrin, and V. Shmatikov, "How to backdoor federated learning," in *Proceedings of the Twenty Third International Conference on Artificial Intelligence and Statistics*, ser. Proceedings of Machine Learning Research, S. Chiappa and R. Calandra, Eds., vol. 108. PMLR, 26–28 Aug 2020, pp. 2938–2948. [Online]. Available: https://proceedings.mlr.press/v108/bagdasaryan20a.html

[26] Z. Sun, P. Kairouz, A. T. Suresh, and H. B. McMahan, "Can you really backdoor federated learning?" *CoRR*, vol. abs/1911.07963, 2019. [Online]. Available: http://arxiv.org/abs/1911.07963

[27] C. Wu, X. Yang, S. Zhu, and P. Mitra, "Mitigating backdoor attacks in federated learning," *CoRR*, vol. abs/2011.01767, 2020. [Online]. Available: https://arxiv.org/abs/2011.01767

[28] N. M. Jebreel, J. Domingo-Ferrer, D. Sánchez, and A. Blanco-Justicia, "Defending against the label-flipping attack in federated learning," 2022. [Online]. Available: https://arxiv.org/abs/2207.01982

[29] D. Li, W. E. Wong, W. Wang, Y. Yao, and M. Chau, "Detection and mitigation of label-flipping attacks in federated learning systems with kpca and k-means," in *2021 8th International Conference on Dependable Systems and Their Applications (DSA)*, 2021, pp. 551–559.

[30] J. R. Douceur, "The sybil attack," in *Peer-to-Peer Systems*, P. Druschel, F. Kaashoek, and A. Rowstron, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2002, pp. 251–260.

[31] C. Fung, C. J. M. Yoon, and I. Beschastnikh, "Mitigating sybils in federated learning poisoning," *CoRR*, vol. abs/1808.04866, 2018. [Online]. Available: http://arxiv.org/abs/1808.04866

[32] Y. Mao, D. Data, S. Diggavi, and P. Tabuada, "Decentralized learning robust to data poisoning attacks," in *2022 IEEE 61st Conference on Decision and Control (CDC)*, 2022, pp. 6788–6793.

[33] S. Farhadkhani, R. Guerraoui, N. Gupta, L. N. Hoang, R. Pinot, and J. Stephan, "Making byzantine decentralized learning efficient," 2022. [Online]. Available: https://arxiv.org/abs/2209.10931