TripleBlind: A Privacy Preserving Framework for Decentralized Data and Algorithms

Gharib Gharibi **Babak Poorebrahim Gilkalaye** **Ravi Patel** **Andrew Rademacher**

David Wagner **Jack Fay** **Gary Moore** **Steve Penrod** **Greg Storm**

Riddhiman Das

TripleBlind, Inc.
research@tripleblind.ai

Abstract

Developing efficient data-driven applications, primarily using deep learning, requires access to large and diverse datasets. However, sharing and collecting sensitive data is extremely challenging due to privacy, ethical, and legal concerns. To address these challenges, we present *TripleBlind*, a practical privacy-preserving framework for creating and consuming data-driven applications from decentralized data and algorithms. TripleBlind provides a set of automated, high-level APIs that enable (1) extracting conclusions from remote data without moving it outside the owner's firewall, (2) training sophisticated AI models from decentralized data, and (3) consuming trained models for secure and efficient inference-as-a-service without compromising the privacy of either the model or the data. We focus in this tool demo on two tasks: First, we train a ResNet-34 model using decentralized medical image data without "seeing" the raw data. Second, we utilize our secure multi-party computation protocol to run real-time inference over the public Internet.

1 Motivation

Artificial intelligence (AI), especially deep learning, has led to significant advances in an ever-growing number of domains [2, 6]. This success heavily relies not only on the availability of large amounts of data but also its diversity–which is critical for training high-performance, fair, and generalizable models. For example, a recent study [3] that investigated more than 400 models for detecting COVID-19 determined that each one of them was fatally flawed. It found that the majority of the models were trained on small, single-origin samples with limited diversity. The study concluded that *it is imperative to develop efficient and semi-automated data exchange and processing pipelines to ensure easy and rapid access to high-quality, diverse data without compromising its privacy*. Therefore, there exist pressing demands for sharing multi-institutional and even multi-national data to build reliable data-driven applications. However, it is still profoundly challenging to share and utilize such sensitive data due to privacy, ethical, and legal concerns [1, 8, 10, 12, 13, 17].

Prior work. To address private data sharing challenges, Federated Learning (FL) [7] and Split Learning (SL) [14] were developed to train models on decentralized data without the need to centrally store it. Compared to FL, SL reduces the communication and computation overhead by splitting the model into two parts distributed between the data owner (client) and the consumer (server) and only exchanges the output of one layer between the two instead of the entire model. However, SL trains the model sequentially among the clients, leading to impractical training times.

Peer reviewed and Accepted at NeurIPS 2021 Demo Track

To address inference-as-a-service privacy challenges, several studies suggested the use of cryptographic methods, most notably fully-homomorphic encryption (FHE) [4] and secure multi-party computation (MPC) [16]. However, the significant computational overhead of FHE and its limited support for many functions (e.g., sigmoid) make it unsuitable for real-world applications [9]. In contrast, secure MPC-based inference methods have illustrated some promising results [5, 11, 15].

Overall, existing privacy-preserving methods face two main challenges: (1) they introduce computational overhead and (2) they lack proper tool support; for instance, the complexity of secure MPC techniques and absence of tool support put them out of reach for most machine learning practitioners.

2 The technology demonstrated: TripleBlind

TripleBlind is a privacy-preserving framework for building and consuming data-driven applications. It provides a set of automated, high-level APIs that enable (1) extracting conclusions from remote data without moving it outside the owner's firewall, (2) training AI models on decentralized data, and (3) utilizing trained models for efficient inference-as-a-service without compromising the privacy of either the model or the data. Through TripleBlind, we aim to advance the state-of-the-art of privacy-preserving methods and provide them via a set of simple and user-friendly APIs. A video demo highlighting the main features of TripleBlind is located at `www.tripleblind.ai/neurips2021`

The elements of novelty We introduce and illustrate in this demo three contributions:

Blind Learning (BL). A decentralized training approach built on top of Split Learning. It introduces several novelties over FL and SL. BL splits the model between the server and clients, and thus it provides two advantages over FL. First, it reduces the communication overhead since it only exchanges a single layer of outputs between the clients and the server. Second, it reduces the computational requirements at the clients since they train only one part of the model compared to the entire model in FL. Compared to SL, our approach trains the clients' models in parallel leading to more efficient training times. Most importantly, BL addresses direct data leakage by augmenting the utility loss function with a distance correlation metric that optimizes the client-side models to prevent sharing unnecessary information with the server, hindering data reconstruction attacks.

Privophy. A secure MPC library that introduces several cryptographic primitives and protocols to evaluate any arbitrary function. Secure MPC enables two or more parties to jointly compute a function over their input data without revealing their data to each other. We focus in this demo on our secure MPC inference for neural networks, and we particularly illustrate, through the live demo, the efficiency and ease-of-use of our encrypted inference service. Some preliminary results on Privophy's performance compared to the current state-of-the-art techniques are listed on the demo website.

Software Development Kit (SDK). A toolset that provides complete scripting control of our services, including Blind Learning and Privophy. It is installed on the end user's device (e.g., a data scientist's workstation) to manage the organization's assets (e.g., data and models) or to operate on other organizations' assets for training, inferences, or analysis. The SDK supports Python, R, and provides command-line utilities to interface with the rest of the ecosystem. The audience will interact with our system using some SDK scripts provided in Jupyter notebooks, as we explain in the following.

3 Demo plan

Our demo will take place in the following order:
(1) Motivate the overall work and the need for systems like TripleBlind
(2) Explain the underlying methodology of Blind Learning and Privophy
(3) Illustrate the usage of our toolset using a live example in a Jupyter notebook to train an image classifier using the CIFAR-10 dataset (distributed over three remote machines). The purpose here is to teach the audience how to interact with TripleBlind
(4) Invite the audience to interact with our system. More details are below.
(5) Answer the audience questions while they interact with our system.

Audience interaction details. We will invite members of the audience to interact with our system. The participants will play one of two roles: (1) a data scientist that trains a model using remote decentralized datasets owned by other organizations or (2) a data owner that wishes to produce predictions of their data using a remote model owned by another organization. In both cases, the participants will run the examples in a real setup over the public Internet. The datasets and models will be placed on individual Google Cloud instances representing different organizations.

The participants will be given access to Jupyter notebooks running online. The notebooks will present four tasks: image classification, tabular data classification, multi-modal (images and text) classification, and a remote inference task. The notebooks will be pre-loaded with the scripts necessary to run these tasks. Nevertheless, the audience will have complete freedom to experiment with our APIs, e.g., change the hyperparameters, datasets, and architecture. The notebooks are also available at the moment for the reviewers (please visit the demo site).

References

[1] California Consumer Privacy Act. (ccpa), 2018. URL `https://oag.ca.gov/privacy/ccpa`.

[2] Mahbubul Alam, Manar D Samad, Lasitha Vidyaratne, Alexander Glandon, and Khan M Iftekharuddin. Survey on deep neural networks in speech and vision systems. *Neurocomputing*, 417:302–321, 2020.

[3] Derek Driggs, Ian Selby, Michael Roberts, Effrossyni Gkrania-Klotsas, James HF Rudd, Guang Yang, Judith Babar, Evis Sala, Carola-Bibiane Schönlieb, and AIX-COVNET collaboration. Machine learning for covid-19 diagnosis and prognostication: lessons for amplifying the signal while reducing the noise, 2021.

[4] Craig Gentry and Dan Boneh. *A fully homomorphic encryption scheme*, volume 20. Stanford University Stanford, 2009.

[5] Chiraag Juvekar, Vinod Vaikuntanathan, and Anantha Chandrakasan. {GAZELLE}: A low latency framework for secure neural network inference. In *27th {USENIX} Security Symposium ({USENIX} Security 18)*, pages 1651–1669, 2018.

[6] William Lotter, Abdul Rahman Diab, Bryan Haslam, Jiye G Kim, Giorgia Grisot, Eric Wu, Kevin Wu, Jorge Onieva Onieva, Yun Boyer, Jerrold L Boxerman, et al. Robust breast cancer detection in mammography and digital breast tomosynthesis using an annotation-efficient deep learning approach. *Nature Medicine*, 27(2):244–249, 2021.

[7] Brendan McMahan, Eider Moore, Daniel Ramage, Seth Hampson, and Blaise Aguera y Arcas. Communication-efficient learning of deep networks from decentralized data. In *Artificial intelligence and statistics*, pages 1273–1282. PMLR, 2017.

[8] Milad Nasr, Reza Shokri, and Amir Houmansadr. Comprehensive privacy analysis of deep learning: Passive and active white-box inference attacks against centralized and federated learning. In *2019 IEEE symposium on security and privacy (SP)*, pages 739–753. IEEE, 2019.

[9] Bernardo Pulido-Gaytan, Andrei Tchernykh, Jorge M Cortés-Mendoza, Mikhail Babenko, Gleb Radchenko, Arutyun Avetisyan, and Alexander Yu Drozdov. Privacy-preserving neural networks with homomorphic encryption: C hallenges and opportunities. *Peer-to-Peer Networking and Applications*, 14(3):1666–1691, 2021.

[10] General Data Protection Regulation. Regulation eu 2016/679 of the european parliament and of the council of 27 april 2016. *Official Journal of the European Union. Available at: http://ec. europa. eu/justice/data-protection/reform/files/regulation_oj_en. pdf (accessed 20 September 2017)*, 2016.

[11] M Sadegh Riazi, Christian Weinert, Oleksandr Tkachenko, Ebrahim M Songhori, Thomas Schneider, and Farinaz Koushanfar. Chameleon: A hybrid secure computation framework for machine learning applications. In *Proceedings of the 2018 on Asia Conference on Computer and Communications Security*, pages 707–721, 2018.

[12] Maria Rigaki and Sebastian Garcia. A survey of privacy attacks in machine learning. *arXiv preprint arXiv:2007.07646*, 2020.

[13] Reza Shokri, Marco Stronati, Congzheng Song, and Vitaly Shmatikov. Membership inference attacks against machine learning models. In *2017 IEEE Symposium on Security and Privacy (SP)*, pages 3–18. IEEE, 2017.

[14] Praneeth Vepakomma, Otkrist Gupta, Tristan Swedish, and Ramesh Raskar. Split learning for health: Distributed deep learning without sharing raw patient data. *arXiv preprint arXiv:1812.00564*, 2018.

[15] Sameer Wagh, Divya Gupta, and Nishanth Chandran. Securenn: 3-party secure computation for neural network training. *Proc. Priv. Enhancing Technol.*, 2019(3):26–49, 2019.

[16] Andrew Chi-Chih Yao. How to generate and exchange secrets. In *27th Annual Symposium on Foundations of Computer Science (sfcs 1986)*, pages 162–167. IEEE, 1986.

[17] Yuheng Zhang, Ruoxi Jia, Hengzhi Pei, Wenxiao Wang, Bo Li, and Dawn Song. The secret revealer: Generative model-inversion attacks against deep neural networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 253–261, 2020.