Lightning Talk: Private and Secure Learning at the Edge with Hyperdimensional Computing

Xiaofan Yu[1], Minxuan Zhou[1], Fatemeh Asgarinejad[1,2], Onat Gungor[1,2], Baris Aksanli[2], Tajana Rosing[1]
[1]University of California San Diego, [2] San Diego State University
{x1yu, miz087, fasgarinejad, ogungor, tajana}@ucsd.edu
, baksanli@sdsu.edu

***Abstract*—As a lightweight and robust brain-inspired computing paradigm, Hyperdimensional Computing (HDC) serves as a promising solution for the next-generation edge AI. However, the basic form of HDC is vulnerable to privacy leaks and cyber attacks. In this paper, we breifly review and discuss the recent contributions to privacy and security of HDC. We first summarize existing HDC designs to protect against privacy leaks, such as differential privacy. Next, we review the data encryption techniques for collaborative learning using HDC based on Multi-Party Computation and Homomorphic Encryption. Finally, we discuss the HDC-based designs for combating cyber attacks in a malicious environment. More research on private and secure HDC-based methods are needed for future large-scale edge deployment.**

***Index Terms*—Privacy, Security, Hyperdimensional Computing**

I. Introduction

With the recent development of Machine Learning (ML), more powerful and energy efficient hardware, Artificial Intelligence (AI) at the edge and beyond is getting more sophisticated. For example, Federated Learning trains the ML model on each local device and only exchanges the updated model (but not the local dataset) with the cloud [1]. In contrast to sending all the data to the cloud for training, on-device learning enables faster decision-making, saves communication costs, and restricts the user-specific data to the edge device. Preserving local data makes edge AI an appealing paradigm. However, multiple challenges remain unsolved, especially (1) how to train efficiently under the limited resources at the edge, (2) how to rigorously prevent privacy leaks and ensure security under susceptible communication channels and/or untrustworthy hosts.

Hyperdimensional Computing (HDC) is brain-inspired and lightweight machine learning paradigm [2]. Raw data samples are first encoded to high-dimensional (e.g., 10K) vectors, after which training and inference can be realized by a handfull of highly parallelizable operations in the high-dimensional space. The fundamental operations include binding (element-wise multiplication), bundling (element-wise addition), and permutation (logical shift). Training is HDC is adding the hypervectors H from the same class to form the class hypervectors, while inference can be realized by checking the similarity between the query hypervector H_q and stored class hypervectors. HDC is a perfect match for AI at the edge given its lightweight computation, single-pass training and robustness [3].

In this paper, we review the latest contributions to privacy and security at the edge using HDC. We first discuss the recent works to address the privacy leak of HDC (Section II), then introduce the latest techniques for data encryption between server and clients using HDC (Section III). Finally, we introduce the HDC works to deal with cyber attacks in an adversarial environment (Section IV). Different from Ma *et al.* [4] that surveyed the robustness of HDC against cyber attacks and hardware errors, we mainly focus on the privacy aspect of HDC. Thriving at efficiency while guaranteeing privacy and security, we believe the reviewed techniques open up a promising direction for future research in AI at the edge.

II. HDC and Privacy

HDC is easily exposed to privacy breaches due to the simple operations and transparent model. Attackers can reverse engineer the encoding function and access the raw data using malicious inputs, which is also known as model inversion or extraction attacks [5]–[7]. We next discuss recent approaches that have been proposed to combat HDC's privacy leaks.

Prive-HD [5] focuses on *differential privacy*. It injects Gaussian noise to the trained hypervectors. Model pruning and hypervector quantization are used during single-pass training and inference to maximize the obfuscation effect of noise. **PRID** [6] proposed two iterative techniques using noise injection and model quantization to protect the HDC model against model inversion attacks. **HDLock** [7] incorporated regulated combination and permutation to HDC encoding, which largely increases the difficulty for the attackers to extract the base encoding hypervectors.

III. HDC on Encrypted Data

Privacy is one of the key issues when exchanging private data with other devices in collaborative learning. **SecureHD** [8] proposed a collaborative HDC framework that combines multi-party computation (MPC) with new HDC encoding/decoding algorithms. It uses MPC to generate a global key on a trustworthy server and various private keys on untrustworthy clients and servers. The client employs extra encoding (permutation) according to the private keys while the untrustworthy servers use their private keys, each for a client, to align the permuted hypervectors. While ensuring

some security and efficiency, SecureHD is only applicable to HDC computation using hypervectors and does not fully guard against data leaks.

Fully Homomorphic Encryption (FHE) is a public-key encryption scheme that allows fully secure and arbitrary computation on encrypted data without decryption, so only the edge device has the access to its data. Any computation beyond the edge device is performed in fully encrypted domain. However, due to the complexity of conventional machine learning models, existing FHE implementations of ML are extremely slow, especially when it comes to training [9], [10]. **FHE-HD** [11] is the first end-to-end implementation of HDC using CKKS-based fully homomorphic encryption. It uses novel data packing and non-linear function support. FHE-HD is up to 5.8× faster during training while maintaining comparable accuracy to the state-of-the-art [12], and more than 1000× faster for inference. Recent works further accelerated FHE using processing in-memory (PIM) technologies [13], [14], making it possible to train and learn at the edge at near real time speeds.

IV. HDC & Cyber Attacks

Recent work addresses adversarial [4] and privacy attacks [5], along with IP stealing [7]. Adversarial attacks create perturbed inputs to confuse a learning model [15]. *Poisoning* and *evasion* attacks happen during training and inference respectively. **PoisonHD** [16] proposed an HDC poisoning attack framework based on confidence-based label-flipping. As a defense, data sanitization was used to filter suspect samples. There are various HDC evasion attacks from different domains: image classification [17]–[19], fault diagnosis [20], text classification [21], speech recognition [22], and intrusion detection [23]. **Yang and Ren** [17] devised a genetic algorithm based adversarial attack. **HDXplore** [18] and **TestHD** [19] utilized distance guided fuzz testing for an automated adversarial attack. **RES-HD** [20] proposed black-box transfer attacks for intelligent fault diagnosis. **Moraliyage et al.** [21] used widely-used text adversarial frameworks, e.g., TextFooler, for language recognition and text classification. **Chen and Li** [22] proposed differential evolution based adversarial attacks. **Adversarial-HD** [23] introduced a diversity-induced adversarial attack design framework for intrusion detection which selects the most effective attack. Adversarial retraining has been used to defend against evasion attacks which retrains the HDC model by including adversarial examples [4].

V. Conclusion

In this work, we review the HDC techniques for security and privacy including prevention of privacy leaks, data encryption, and mitigation against cyber attacks. Due to its simplicity and robustness, HDC provides unique opportunities to realize efficient and secure edge AI systems. As security and privacy become top-priority concerns, this review sets up a reference for researchers to understand the state-of-the-art and develop further solutions for edge AI systems.

Acknowledgment

This work was supported in part by PRISM & CoCoSys, two of seven centers in JUMP 2.0 (an SRC program sponsored by DARPA), SRC Global Research Collaboration (GRC) grant, and NSF grants #1911095, #1826967, #2100237, and #2112167.

References

[1] B. McMahan, E. Moore, D. Ramage, S. Hampson, and B. A. y Arcas, "Communication-efficient learning of deep networks from decentralized data," in *Artificial intelligence and statistics*. PMLR, 2017.

[2] P. Kanerva, "Hyperdimensional computing: An introduction to computing in distributed representation with high-dimensional random vectors," *Cognitive computation*, vol. 1, pp. 139–159, 2009.

[3] J. Morris *et al.*, "Hydrea: Towards more robust and efficient machine learning systems with hyperdimensional computing-based classifiers," in *DATE'21*.

[4] D. Ma *et al.*, "Robust hyperdimensional computing against cyber attacks and hardware errors: A survey," in *ASP-DAC'23*. IEEE.

[5] B. Khaleghi, M. Imani, and T. Rosing, "Prive-hd: Privacy-preserved hyperdimensional computing," in *DAC'20*. IEEE.

[6] A. Hernández-Cano *et al.*, "Prid: Model inversion privacy attacks in hyperdimensional learning systems," in *DAC'21*. IEEE.

[7] S. Duan *et al.*, "Hdlock: exploiting privileged encoding to protect hyperdimensional computing models against ip stealing," in *DAC'22*.

[8] M. Imani, Y. Kim, S. Riazi, J. Messerly, P. Liu, F. Koushanfar, and T. Rosing, "A framework for collaborative learning in secure high-dimensional space," in *CLOUD'19*. IEEE.

[9] J.-W. Lee *et al.*, "Privacy-preserving machine learning with fully homomorphic encryption for deep neural network," *IEEE Access*, vol. 10, pp. 30 039–30 054, 2022.

[10] M. Zheng, Q. Lou, and L. Jiang, "Primer: Fast private transformer inference on encrypted data," *arXiv preprint arXiv:2303.13679*, 2023.

[11] Y. Nam *et al.*, "Efficient machine learning on encrypted data using hyperdimensional computing," in *ISLPED'23*. ACM.

[12] K. Nandakumar *et al.*, "Towards deep neural network training on encrypted data," in *CVPR'19*.

[13] H. Nejatollahi *et al.*, "Cryptopim: In-memory acceleration for lattice-based cryptographic hardware," in *DAC*, 2020.

[14] S. Gupta, R. Cammarota, and T. Š. Rosing, "Memfhe: End-to-end computing with fully homomorphic encryption in memory," *ACM Transactions on Embedded Computing Systems*, 2022.

[15] O. Gungor *et al.*, "Stewart: Stacking ensemble for white-box adversarial attacks towards more resilient data-driven predictive maintenance," *Computers in Industry*, vol. 140, p. 103660, 2022.

[16] R. Wang and X. Jiao, "Poisonhd: Poison attack on brain-inspired hyperdimensional computing," in *DATE'22*. IEEE.

[17] F. Yang and S. Ren, "Adversarial attacks on brain-inspired hyperdimensional computing-based classifiers," *arXiv preprint arXiv:2006.05594*, 2020.

[18] R. Thapa, D. Ma, and X. Jiao, "Hdxplore: Automated blackbox testing of brain-inspired hyperdimensional computing," in *2021 IEEE Computer Society Annual Symposium on VLSI (ISVLSI)*. IEEE, 2021, pp. 90–95.

[19] D. Ma, T. Š. Rosing, and X. Jiao, "Testing and enhancing adversarial robustness of hyperdimensional computing," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 2023.

[20] O. Gungor, T. Rosing, and B. Aksanli, "Res-hd: Resilient intelligent fault diagnosis against adversarial attacks using hyper-dimensional computing," *arXiv preprint arXiv:2203.08148*, 2022.

[21] H. Moraliyage *et al.*, "Evaluating the adversarial robustness of text classifiers in hyperdimensional computing," in *HSI'22*. IEEE.

[22] W. Chen and H. Li, "Adversarial attacks on voice recognition based on hyper dimensional computing," *Journal of Signal Processing Systems*, vol. 93, no. 7, pp. 709–718, 2021.

[23] O. Gungor, T. Rosing, and B. Aksanli, "Adversarial-hd: Hyperdimensional computing adversarial attack design for secure industrial internet of things," in *Proceedings of CPS-IoT Week*, 2023.