On Achieving Privacy-Preserving State-of-the-Art Edge Intelligence

Chabal, D.; Sapra, D.; Mann, Z.A.

DOI
10.48550/arXiv.2302.05323

Publication date
2023

Document Version
Final published version

On Achieving Privacy-Preserving State-of-the-Art Edge Intelligence

Daphnee Chabal[1*†], **Dolly Sapra**[2], **Zoltán Ádám Mann**[1]

[1]Complex Cyber Infrastructure group, University of Amsterdam
Lab42, Science Park 900, 1012 WX Amsterdam, The Netherlands
[2]Parallel Computing Systems group, University of Amsterdam
Lab42, Science Park 900, 1012 WX Amsterdam, The Netherlands

Abstract

Deep Neural Network (DNN) Inference in Edge Computing, often called Edge Intelligence, requires solutions to insure that sensitive data confidentiality and intellectual property are not revealed in the process. Privacy-preserving Edge Intelligence is only emerging, despite the growing prevalence of Edge Computing as a context of Machine-Learning-as-a-Service. Solutions are yet to be applied, and possibly adapted, to state-of-the-art DNNs. This position paper provides an original assessment of the compatibility of existing techniques for privacy-preserving DNN Inference with the characteristics of an Edge Computing setup, highlighting the appropriateness of secret sharing in this context. We then address the future role of model compression methods in the research towards secret sharing on DNNs with state-of-the-art performance.

Introduction

Deep Neural Networks (DNNs), the prominent tools used in the field of Artificial Intelligence, are sought after in many sectors of activities to optimize decision-making and improve the quality of services (Lin et al. 2022). Specifically, the amount of DNN deployments for commercial purposes during a customer's interaction with everyday objects is proliferating (Wolf 2019).

Privacy-preserving Inference aims to protect the privacy and security of data belonging to the multiple parties involved in Neural Network Inference.

There is a global rise of smart services offered by internet-connected devices (sometimes called the Internet of Things or IoT), which are increasingly immersed in daily life (e.g., smartwatches, smartphones, personal digital assistants), and recording confidential facts about our lives. The International Data Corporation estimates that in 2025 there will be more than 55 billion IoT devices in the world (Reinsel 2019), compared to 12.5 billion in 2010 (Sivaraman et al. 2018). The data these devices collect at the edge of the edge-cloud computing continuum will, in many use cases, be processed locally, through an emerging decentralized computing paradigm called Edge Computing (Yu et al. 2017; Shi et al. 2016; Davis 2018; Ayed et al. 2021).

The privacy risk in processing the data through DNNs is two-fold. On one hand, Inference data is produced by individuals, institutions, businesses, and is held by the devices they own or use, and may be shared with the businesses that make those devices available. The data however needs to be shared in full with parties that facilitate the "intelligence", as is the case for the Machine-Learning-as-a-Service (MLaaS) business model. On the other hand, DNNs are costly for companies to develop. The DNNs architecture, inner parameters, as well as the sensitive features contained in the data used during training are then deemed valuable confidential proprietary data for the companies. The model however still needs to be made available to third parties to generate meaningful (i.e., accurate) Inference outputs.

In recent years, many techniques have been put forward to solve the predicament of functional-yet-privacy-preserving DNN Inference (Boulemtafes, Derhab, and Challal 2020; Zhang, Xin, and Wu 2021). Most works however do not consider the global context of secure and private AI deployment for MLaaS, in terms of (1) the characteristics of distributed systems that execute DNN Inference and (2) the computational requirements of actual state-of-the-art DNNs underlying commercial smart services.

Edge Computing offers several advantages over cloud computing, including reduced latency for better user experience and increased agency over the data's life cycle, as data is redirected through fewer nodes, is less attainable to unknown third parties, and risks of bottleneck in gateways decrease (Xiao et al. 2019; Varghese et al. 2016). However, a major drawback is that the devices involved, with Inference clients such as IoT objects or sensors, and DNN-holders such as Edge servers or small data centers, have less computational capacity than that offered on demand by cloud platforms (Xu et al. 2021; Chen and Ran 2019; Ayed et al. 2022). Moreover, methods of privacy-preserving AI are assumed to be applied to systems already equipped with ubiquitous security procedures existing in distributed systems globally (e.g., access control, anomaly detection,

*This work was partially supported by the European Union's Horizon 2020 research and innovation programme under grant 871525 (FogProtect).
†Presentation of this work was partially supported by the European Union's project SECURED.

encrypted communication) and that themselves add load (Aqeel-ur Rehman et al. 2016; Lachner, Mann, and Dustdar 2021).

Some work (Huang et al. 2019; Baccour et al. 2020; Yan, Pei, and Li 2019) has been proposed to bring privacy-preserving DNN Inference to de-centralized Edge Computing. However, these methods were evaluated with outdated DNNs that are less computationally complex than state-of-the-art models we see in present-day AI applications. These simpler models have little to no real-world application in commercial MLaaS setups.

The aim of this paper is to present informed recommendations for upcoming research to achieve privacy-preserving DNN Inference in modern and commercially relevant Edge Computing settings. While promising, works emerging in this domain are still isolated efforts. The research avenues we formulate, which we coin here as **Privacy-Preserving Edge Intelligence**, are at the emerging intersection of two very active research fields, privacy-preserving DNN Inference and DNN Inference in Edge Computing.

It is important to note that the training phase is out of scope for this paper as training is impractical in Edge Computing settings, especially in the context of MLaaS. In particular, Federated Learning is a promising solution already put forward for computationally-sensitive privacy-preserving training in a collaborative setting and is actively researched (Yin, Zhu, and Hu 2021; Mothukuri et al. 2021; El Ouadrhiri and Abdelhadi 2022), but is not in our scope as we focus on inference.

Privacy Requirements for Edge Intelligence

Devices and Edge servers have a high risk of malicious tampering and interventions due to their ease-of-access (Aqeel-ur Rehman et al. 2016). Solutions for privacy-preserving Edge Intelligence must therefore be effective in providing information security (Mann 2022). There are 4 main privacy requirements during the Inference phase in MLaaS: the client may not learn 1) the model's architecture and 2) the model's trained parameters, while the party holding the model, typically the server, must not learn 3) the Inference input data nor 4) the Inference output. Here, we assume that standard system security methods (e.g., limiting the number of Inference requests) are in place to protect a fifth piece of potentially sensitive data, the training dataset (see model inversion attacks) (Boulemtafes, Derhab, and Challal 2020).

General characteristics of Edge Intelligence are described extensively in (Yu et al. 2017; Deng et al. 2020; Xu et al. 2021; Chen and Ran 2019), providing criteria for privacy-preserving solutions applied in an Edge Intelligence context. Solutions should allow practical implementations in a commercial setting, and must provide accurate and timely Inference.

Edge intelligence setups may involve more than two parties during DNN Inference. For example, in a smart home sensor-actuator setup, computations offloaded to several servers (operated by different companies) may receive Inference input from some sources, while sending Inference output to other devices. Information should remain private, even if parties are secretly colluding.

In an Edge setup, clients sending Inference inputs are often low-capacity devices with minimal compute capacity for data collection, temporary storage, and transmission tasks, while the model is held and evaluated by a nearby Edge server.

Servers are capable nowadays of receiving and sending more than a Gbps using LAN or other fast intranet networks. A potential communication bottleneck arises however, when network transmission is of type PAN (e.g., Bluetooth), WAN, MAN, or LPWAN, all common to Edge Computing setups.

Client drop-out occurs mostly to mobile devices such as smartphones, which can also easily turn off. In most cases, client dropout is inconsequential: even if the client is assigned chunks of Inference computations, DNN Inference can be paused until the client is within reach again. This criterion is however important for cases where device drop-out would disrupt task distribution (e.g., swarm intelligence with drones, and mobile computing).

Lastly, to make IoT objects available to consumers at affordable costs, they may lack state-of-the-art hardware specialized in DNN Inference. Additionally, when a device drops out, Edge Computing algorithms aim to dynamically re-assign tasks to the next available device regardless of hardware. Solutions should therefore be applicable to most types of hardware, without assuming that specialized hardware is available.

Assessment of Privacy-Preserving Techniques

The main techniques that constitute the field of Privacy-preserving DNN Inference are reviewed in several comprehensive surveys (Boulemtafes, Derhab, and Challal 2020; Zhang, Xin, and Wu 2021; Pulido-Gaytan et al. 2021; Ball et al. 2019). In this section, we assess the compatibility of each category of techniques with the requirements of Edge Intelligence (summarized in Table 1).

Fully Homomorphic Encryption (FHE): a form of encryption E performed by the client to input data x. $E(x)$ is sent to the server, which returns after Inference, the cyphertext $E(y)$, to the client. $E(y)$ is the encrypted equivalent of the correct Inference output y. Optionally, some parameters of the DNN can also be encrypted. FHE meets all four privacy requirements, and the Inference task itself can be completed in case of client drop-out. However, low-end client devices cannot carry the heavy cryptographic operations required. Despite promising recent advances (Brutzkus, Gilad-Bachrach, and Elisha 2019; Reagen et al. 2021; Lee et al. 2022) since its inception (Gentry 2009), FHE schemes are still too computationally demanding for applications to Edge Intelligence. Additionally, FHE only supports additions and multiplications, and thus require additional processing for other non-linear operations (e.g., polynomial approximation of some activation functions). Despite this, little to no loss in accuracy has been reported, especially via re-training modified DNNs.

Garbled Circuits: 2-party protocol based on converting a neural network to a Boolean circuit made of AND, XOR,

Table 1: Summary of our assessment of Privacy-Preserving techniques for Edge Intelligence.

		Fully Homomorphic Encryption	Garbled Circuit	Secret Sharing	Model Splitting w/o noise	Model Splitting w/ noise	Secure Enclave
Edge Intelligence Requirements	fulfills the 4 privacy requirements	✓	✗	✓	✗	✗	✓
	can involve >2 parties	✗	✗	✓	✓	✗	✗
	Inference accuracy	✓	✓	✓	✓	✗	✓
	low latency expected	✗	✗	✓	✓	✓	✓
	minimal compute capacity (client)	✗	✗	✓	✓	✓	✓
	limited compute capacity (server)	✗	✗	✓	✓	✓	✓
	limited communication	✗	✗	✗	✗	✓	✓
	high drop-out rate (client)	✓	✗	✗	✓	✗	✗
	hardware independence	✓	✓	✓	✓	✓	✗

and XNOR gates, where each gate corresponds to an operation. The architecture of the DNN is known to both parties, but the input data and the weights of the DNN are kept secret. As a sub-protocol, oblivious transfer is used, a public-key cryptography-based scheme that enables one party to send one of two inputs to a second party so that the second party only learns one of the inputs and the first party does not learn which input the second party learned. Garbled Circuits support both linear and non-linear operations but are computationally costly (especially for AND gates (Kolesnikov and Schneider 2008)). Moreover, creation of the garbled circuit includes creation and permutation of a truth table per gate, to further encrypt it (e.g., using AES-based cryptography). As with FHE, this method is thus ill-adapted for low-end clients.

Secret Sharing: n-party secure Multi-Party computing protocols in which each value involved in DNN Inference (i.e., input and model parameters) are divided into n shares, such that individual shares do not reveal anything about the secret values. Since originally introduced (Shamir 1979), several different secret sharing schemes have been proposed. Additive (Huang et al. 2019; Riazi et al. 2019) and replicated (Ibarrondo, Chabanne, and Önen 2021; Wagh et al. 2021) secret sharing seem especially appropriate for DNN Inference. For a complete Inference, the evaluation of the layers of the DNN may be performed in several ways, depending on various factors. In particular, different protocols can be used for addition, multiplication (e.g., masking inputs with Beaver Triplets), and non-linear operations (e.g., polynomial approximation, garbled circuits). The protocols also depend on the number of parties involved, as well as whether colluding is accounted for or not. Servers can also send a client shares to compute. Secret Sharing is not encryption-based and therefore relatively cheap to add to DNN Inference tasks. DNN Inference however fails if devices holding information on how shares are created (e.g., random number generator) drop out. Secret Sharing is still a communication-intensive privacy-preserving method, necessitating a high number of communication rounds.

Model Splitting without noise: partitioning a DNN so that each party receives unprocessed chunks of calculations, including raw weights and inputs. Accuracy is therefore preserved. The higher the number of devices recruited, the higher the privacy as well as speed (with possibility of parallel computing), as no party may reconstruct the neural network, nor infer the training nor input data from the parts it receives (Baccour et al. 2020). This method requires the client to perform the initial and last computations, but does not need a powerful server. In case of colluding, model architecture and parameter privacy are largely lost.

Model Splitting with noise: noise is added by the client to the input data, intermediary results, and/or weights when the client receives partial computations from a DNN. The noise added must fulfill the requirements for Differential Privacy, which mathematically guarantee that data is obfuscated sufficiently to conceal individual records (e.g., a person's identity) it may contain. This is necessary because raw input data can be reconstructed from intermediary results after even 6 layers (He, Zhang, and Lee 2021). There is a privacy/accuracy trade-off based on the amount of noise added. The client is responsible for noising and de-noising, which can be computationally expensive depending on the scheme used (e.g., auto-encoders and decoders for obfuscation).

Secure Enclaves: are dedicated portions of memory which are designated by the CPU as inaccessible from the operating system nor any other application, and within which data can be secretly processed and encrypted/decrypted if necessary. A popular example of Secure Enclaves is Intel's SGX (Kuznetsov, Chen, and Zhao 2021). For DNN Inference, two parties may send an encrypted model and input data, respectively, which can then be decrypted and processed within Secure Enclaves, finally returning the Inference output to the appropriate party, thus providing full privacy. Secure Enclaves are however costly and more memory limited than traditional hardware.

This assessment indicates particular suitability of Secret Sharing for Edge Intelligence, as it meets the most criteria (7 out of 9), especially meeting all information privacy and performance-related requirements. Therefore, we dedicate the rest of the paper to discuss secret sharing and its applicability for Edge Intelligence.

Implications of State-of-the-Art Performance

Recent solutions for Secret Sharing in DNN Inference (Zhang, Xin, and Wu 2021), not only for Edge Computing, all still use outdated Convolutional Neu-

Table 2: Summary of the impact of Model Compression techniques on Secret Sharing for DNN Inference

	Less operations in total	Less non-linear operations	Reduced message sizes	Less communication rounds
Quantizing	yes	yes	yes	no
Pruning	yes	not purposefully	yes	not purposefully
Knowledge Distillation	yes	yes	no	yes
Low-rank approximation	yes	yes	no	yes

ral Networks (CNNs) as evaluative benchmarks (e.g., AlexNet (Krizhevsky, Sutskever, and Hinton 2017) trained on MNIST (Deng 2012)). The performance of these CNNs is humble compared to that of Transformers (e.g., answer generation from multi-modal inputs (Li et al. 2019)), a state-of-the-art category of DNNs now ubiquitous in the field of AI (Zaidi et al. 2022).

The question then arises: would the solutions, as they are, be applicable for fast Edge Intelligence in the context of a real and state-of-the-art MLaaS task? The answer is probably 'no'.

Primarily, larger DNNs have higher complexity (i.e., more parameters to compute, more nodes in layers due to larger inputs), leading to an increase in the number of secret shares to produce and re-combine. The amount of non-linear operations during Secret Sharing, while manageable on smaller DNNs, can become problematic as it increases, which may necessitate more Garbled Circuits and/or Beaver Triplets, and both methods are especially intensive in 2-party settings, despite possibilities of some offline processing (Mann et al. 2022).

Furthermore, state-of-the-art DNNs have a higher diversity in the types of layers, (e.g., Self-Attention, Recurrent) than benchmark CNNs do which current Secret Sharing schemes are not yet designed to handle (Mann et al. 2022). New types of layers (e.g., Self-Attention) have more data processing, such as parallel encodings of subsets of inputs (e.g., a single word), as well as more operations to perform per layer than classic layers (e.g., ReLu, pooling), requiring new protocols other than current ones which consider a layer as a unit only taking in simultaneous inputs (Zhang, Xin, and Wu 2021).

Model Compression and Secret Sharing

A first step towards bringing large Transformers to Secret Sharing particularly for Edge Intelligence, is to tackle the computation and communication bottleneck. Three categories of solutions exist: 1) Hardware Acceleration (Wang et al. 2020), consisting of a set of instructions to parallelize computational tasks into specialized hardware components (e.g., Neural network Processing Units – NPUs (Yao et al. 2022)) – similarly to Secure Enclaves, they may be too expensive to integrate in commercial objects; 2) Software Orchestration (Deng et al. 2022; Aghapour et al. 2022), consisting of developing data pipelines or algorithms to optimize resource management to reduce latency of DNN Inference – it is assumed to be applied to some extent in any distributed system, and cannot reduce the rounds of communication required for Secret Sharing; and 3) Model Compression (Xu et al. 2021; Cheng et al. 2018), reducing the amount and complexity of the computations – typically requiring re-training to retain accuracy.

Model Compression techniques, namely quantization, pruning (Liang et al. 2021), knowledge distillation (Gou et al. 2021), and low-rank approximation (Idelbayev and Carreira-Perpinán 2020), while increasingly customary in Edge Intelligence, have yet to be compared in the context of Secret Sharing. Table 2 provides a qualitative comparison.

Quantization reduces the byte-size of each value, and consequently the size of the secret shares communicated between parties, but leaves the total number of communication rounds largely unaffected. Particularly, solutions with binary or ternary quantization to weights, and a limited fixed-point size to activation functions, preserve the granularity of input data while significantly reducing communication (Liang et al. 2021). Quantization can be combined with other model compression techniques as well.

Pruning removes inconsequential computations in a DNN. It offers no guarantees of effectiveness in addressing computational complexity specific to Secret Sharing. In the best cases, however, Pruning may remove a significant number of connections between the nodes of a DNN, thus reducing computation and communication.

Knowledge Distillation (i.e., training a smaller network off of a larger one) and Low-rank Approximation (i.e., reducing the dimensionality of each layer via matrix decomposition) are more promising candidates for secret sharing. Firstly, they remove extra features from the input data sooner, thus reducing the amount of input to propagate throughout the network. Secondly, they both reduce the amount of computations systematically throughout the DNN (i.e., most layers are reduced in size). Consequently, less rounds of Secret Sharing communication are necessary per layer. Knowledge Distillation also reduces the number of layers (Zaidi et al. 2022). Lastly, both Knowledge Distillation and low-rank approximation are actively researched and have recently been successfully applied to state-of-the-art transformers (e.g., BERT (Devlin et al. 2018) became DistilBERT (Sanh et al. 2019) and Ladabert (Mao et al. 2020)). The accuracy of compressed versions of those model is also improving (Zaidi et al. 2022).

Conclusion

Secret Sharing was deemed the most promising privacy-preserving technique given Edge Intelligence characteristics, but is not yet applicable to state-of-the-art Deep Neural Networks. Future research should address the new types

of DNN layers and computations that current Secret Sharing schemes do not yet account for, while optimizing performance for MLaaS in Edge Computing. We put forward, pending experiments, Knowledge Distillation and Low-Rank Approximation as promising means to further accommodate new Secret Sharing protocols, for practical Edge Intelligence.

References

Aghapour, E.; Sapra, D.; Pimentel, A.; and Pathania, A. 2022. CPU-GPU Layer-Switched Low Latency CNN Inference. In *25th Euromicro Conference on Digital System Design (DSD)*.

Aqeel-ur Rehman, S. U. R.; Khan, I. U.; Moiz, M.; and Hasan, S. 2016. Security and privacy issues in IoT. *International Journal of Communication Networks and Information Security (IJCNIS)*, 8(3): 147–157.

Ayed, D.; Dragan, P.-A.; Félix, E.; Mann, Z. Á.; Salant, E.; Seidl, R.; Sidiropoulos, A.; Taylor, S.; and Vitorino, R. 2022. Protecting sensitive data in the cloud-to-edge continuum: The FogProtect approach. In *22nd IEEE International Symposium on Cluster, Cloud and Internet Computing (CCGrid)*, 279–288. IEEE.

Ayed, D.; Jaho, E.; Lachner, C.; Mann, Z. Á.; Seidl, R.; and Surridge, M. 2021. FogProtect: Protecting sensitive data in the computing continuum. In *Advances in Service-Oriented and Cloud Computing: International Workshops of ESOCC 2020*, 179–184. Springer.

Baccour, E.; Erbad, A.; Mohamed, A.; Hamdi, M.; and Guizani, M. 2020. Distprivacy: Privacy-aware distributed deep neural networks in iot surveillance systems. In *GLOBECOM 2020-2020 IEEE Global Communications Conference*, 1–6. IEEE.

Ball, M.; Carmer, B.; Malkin, T.; Rosulek, M.; and Schimanski, N. 2019. Garbled neural networks are practical. *Cryptology ePrint Archive*.

Boulemtafes, A.; Derhab, A.; and Challal, Y. 2020. A review of privacy-preserving techniques for deep learning. *Neurocomputing*, 384: 21–45.

Brutzkus, A.; Gilad-Bachrach, R.; and Elisha, O. 2019. Low latency privacy preserving inference. In *International Conference on Machine Learning*, 812–821. PMLR.

Chen, J.; and Ran, X. 2019. Deep learning with edge computing: A review. *Proceedings of the IEEE*, 107(8): 1655–1674.

Cheng, J.; Wang, P.-s.; Li, G.; Hu, Q.-h.; and Lu, H.-q. 2018. Recent advances in efficient computation of deep convolutional neural networks. *Frontiers of Information Technology & Electronic Engineering*, 19(1): 64–77.

Davis, G. 2018. 2020: Life with 50 billion connected devices. In *2018 IEEE International Conference on Consumer Electronics (ICCE)*, 1–1.

Deng, C.; Fang, X.; Wang, X.; and Law, K. 2022. Software Orchestrated and Hardware Accelerated Artificial Intelligence: Toward Low Latency Edge Computing. *IEEE Wireless Communications*.

Deng, L. 2012. The MNIST Database of Handwritten Digit Images for Machine Learning Research [Best of the Web]. *IEEE Signal Process. Mag.*, 29(6): 141–142.

Deng, S.; Zhao, H.; Fang, W.; Yin, J.; Dustdar, S.; and Zomaya, A. Y. 2020. Edge intelligence: The confluence of edge computing and artificial intelligence. *IEEE Internet of Things Journal*, 7(8): 7457–7469.

Devlin, J.; Chang, M.-W.; Lee, K.; and Toutanova, K. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

El Ouadrhiri, A.; and Abdelhadi, A. 2022. Differential privacy for deep and federated learning: A survey. *IEEE Access*, 10: 22359–22380.

Gentry, C. 2009. Fully homomorphic encryption using ideal lattices. In *Proceedings of the forty-first annual ACM symposium on Theory of computing*, 169–178.

Gou, J.; Yu, B.; Maybank, S. J.; and Tao, D. 2021. Knowledge distillation: A survey. *International Journal of Computer Vision*, 129(6): 1789–1819.

He, Z.; Zhang, T.; and Lee, R. B. 2021. Attacking and Protecting Data Privacy in Edge-Cloud Collaborative Inference Systems. *IEEE Internet Things J.*, 8(12): 9706–9716.

Huang, K.; Liu, X.; Fu, S.; Guo, D.; and Xu, M. 2019. A lightweight privacy-preserving CNN feature extraction framework for mobile sensing. *IEEE Transactions on Dependable and Secure Computing*, 18(3): 1441–1455.

Ibarrondo, A.; Chabanne, H.; and Önen, M. 2021. Banners: Binarized neural networks with replicated secret sharing. In *Proceedings of the 2021 ACM Workshop on Information Hiding and Multimedia Security*, 63–74.

Idelbayev, Y.; and Carreira-Perpinán, M. A. 2020. Low-rank compression of neural nets: Learning the rank of each layer. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 8049–8059.

Kolesnikov, V.; and Schneider, T. 2008. Improved Garbled Circuit: Free XOR Gates and Applications. In *ICALP*, 486–498. Springer.

Krizhevsky, A.; Sutskever, I.; and Hinton, G. E. 2017. Imagenet classification with deep convolutional neural networks. *Communications of the ACM*, 60(6): 84–90.

Kuznetsov, E.; Chen, Y.; and Zhao, M. 2021. Securefl: Privacy preserving federated learning with sgx and trustzone. In *2021 IEEE/ACM Symposium on Edge Computing (SEC)*, 55–67. IEEE.

Lachner, C.; Mann, Z. Á.; and Dustdar, S. 2021. Towards understanding the adaptation space of AI-assisted data protection for video analytics at the edge. In *IEEE 41st International Conference on Distributed Computing Systems Workshops (ICDCSW)*, 7–12. IEEE.

Lee, J.-W.; Kang, H.; Lee, Y.; Choi, W.; Eom, J.; Deryabin, M.; Lee, E.; Lee, J.; Yoo, D.; Kim, Y.-S.; et al. 2022. Privacy-preserving machine learning with fully homomorphic encryption for deep neural network. *IEEE Access*, 10: 30039–30054.

Li, L. H.; Yatskar, M.; Yin, D.; Hsieh, C.-J.; and Chang, K.-W. 2019. Visualbert: A simple and performant baseline for vision and language. *arXiv preprint arXiv:1908.03557*.

Liang, T.; Glossner, J.; Wang, L.; Shi, S.; and Zhang, X. 2021. Pruning and quantization for deep neural network acceleration: A survey. *Neurocomputing*, 461: 370–403.

Lin, T.; Wang, Y.; Liu, X.; and Qiu, X. 2022. A survey of transformers. *AI Open*.

Mann, Z. Á. 2022. Security- and privacy-aware IoT application placement and user assignment. In *Computer Security – ESORICS 2021 International Workshops*, 296–316. Springer.

Mann, Z. Á.; Weinert, C.; Chabal, D.; and Bos, J. W. 2022. Towards Practical Secure Neural Network Inference: The Journey So Far and the Road Ahead. *Cryptology ePrint Archive, paper 2022/1483, https://eprint.iacr.org/2022/1483*.

Mao, Y.; Wang, Y.; Wu, C.; Zhang, C.; Wang, Y.; Yang, Y.; Zhang, Q.; Tong, Y.; and Bai, J. 2020. Ladabert: Lightweight adaptation of bert through hybrid model compression. *arXiv preprint arXiv:2004.04124*.

Mothukuri, V.; Parizi, R. M.; Pouriyeh, S.; Huang, Y.; Dehghantanha, A.; and Srivastava, G. 2021. A survey on security and privacy of federated learning. *Future Generation Computer Systems*, 115: 619–640.

Pulido-Gaytan, B.; Tchernykh, A.; Cortés-Mendoza, J. M.; Babenko, M.; Radchenko, G.; Avetisyan, A.; and Drozdov, A. Y. 2021. Privacy-preserving neural networks with Homomorphic encryption: Challenges and opportunities. *Peer-to-Peer Networking and Applications*, 14(3): 1666–1691.

Reagen, B.; Choi, W.-S.; Ko, Y.; Lee, V. T.; Lee, H.-H. S.; Wei, G.-Y.; and Brooks, D. 2021. Cheetah: Optimizing and accelerating homomorphic encryption for private inference. In *2021 IEEE International Symposium on High-Performance Computer Architecture (HPCA)*, 26–39. IEEE.

Reinsel, D. 2019. How You Contribute to Today's Growing DataSphere and Its Enterprise Impact. https://blogs.idc.com/2019/11/04/how-you-contribute-to-todays-growing-datasphere-and-its-enterprise-impact/.

Riazi, M. S.; Samragh, M.; Chen, H.; Laine, K.; Lauter, K.; and Koushanfar, F. 2019. XONN: XNOR-based Oblivious Deep Neural Network Inference. In *28th USENIX Security Symposium (USENIX Security 19)*, 1501–1518.

Sanh, V.; Debut, L.; Chaumond, J.; and Wolf, T. 2019. DistilBERT, a distilled version of BERT: smaller, faster, cheaper and lighter. *arXiv preprint arXiv:1910.01108*.

Shamir, A. 1979. How to share a secret. *Communications of the ACM*, 22(11): 612–613.

Shi, W.; Cao, J.; Zhang, Q.; Li, Y.; and Xu, L. 2016. Edge computing: Vision and challenges. *IEEE internet of things journal*, 3(5): 637–646.

Sivaraman, V.; Gharakheili, H. H.; Fernandes, C.; Clark, N.; and Karliychuk, T. 2018. Smart IoT Devices in the Home: Security and Privacy Implications. *IEEE Technology and Society Magazine*, 37(2): 71–79.

Varghese, B.; Wang, N.; Barbhuiya, S.; Kilpatrick, P.; and Nikolopoulos, D. S. 2016. Challenges and opportunities in edge computing. In *2016 IEEE International Conference on Smart Cloud (SmartCloud)*, 20–26. IEEE.

Wagh, S.; Tople, S.; Benhamouda, F.; Kushilevitz, E.; Mittal, P.; and Rabin, T. 2021. Falcon: Honest-Majority Maliciously Secure Framework for Private Deep Learning. *Proc. Priv. Enhancing Technol.*, 2021(1): 188–208.

Wang, X.; Han, Y.; Leung, V. C.; Niyato, D.; Yan, X.; and Chen, X. 2020. Convergence of edge computing and deep learning: A comprehensive survey. *IEEE Communications Surveys & Tutorials*, 22(2): 869–904.

Wolf, M. 2019. Machine learning+ distributed IoT= edge intelligence. In *2019 IEEE 39th International Conference on Distributed Computing Systems (ICDCS)*, 1715–1719. IEEE.

Xiao, Y.; Jia, Y.; Liu, C.; Cheng, X.; Yu, J.; and Lv, W. 2019. Edge computing security: State of the art and challenges. *Proceedings of the IEEE*, 107(8): 1608–1631.

Xu, D.; Li, T.; Li, Y.; Su, X.; Tarkoma, S.; Jiang, T.; Crowcroft, J.; and Hui, P. 2021. Edge intelligence: Empowering intelligence to the edge of network. *Proceedings of the IEEE*, 109(11): 1778–1837.

Yan, Y.; Pei, Q.; and Li, H. 2019. Privacy-preserving compressive model for enhanced deep-learning-based service provision system in edge computing. *IEEE Access*, 7: 92921–92937.

Yao, J.; Zhang, S.; Yao, Y.; Wang, F.; Ma, J.; Zhang, J.; Chu, Y.; Ji, L.; Jia, K.; Shen, T.; et al. 2022. Edge-Cloud Polarization and Collaboration: A Comprehensive Survey for AI. *IEEE Transactions on Knowledge and Data Engineering*.

Yin, X.; Zhu, Y.; and Hu, J. 2021. A comprehensive survey of privacy-preserving federated learning: A taxonomy, review, and future directions. *ACM Computing Surveys (CSUR)*, 54(6): 1–36.

Yu, W.; Liang, F.; He, X.; Hatcher, W. G.; Lu, C.; Lin, J.; and Yang, X. 2017. A survey on the edge computing for the Internet of Things. *IEEE access*, 6: 6900–6919.

Zaidi, S. S. A.; Ansari, M. S.; Aslam, A.; Kanwal, N.; Asghar, M.; and Lee, B. 2022. A survey of modern deep learning based object detection models. *Digital Signal Processing*, 103514.

Zhang, Q.; Xin, C.; and Wu, H. 2021. Privacy-Preserving Deep Learning Based on Multiparty Secure Computation: A Survey. *IEEE Internet of Things Journal*, 8(13): 10412–10429.