Hardware-Aware Acceleration of Secure Deep Learning Inference

Jelle Geertsen
School of Computing, Clemson University, Clemson, SC, USA.
jellemail@clemson.edu

Vernon Gallegos
Department of Electrical Engineering and Computer Science, University of Missouri, Columbia, MO, USA.
hellovernon@missouri.edu

Chiara Molesini
School of Electrical Engineering and Computer Science, Oregon State University, Corvallis, OR, USA.
hellochiara@oregonstate.edu

Abstract

Secure deep learning inference has emerged as a critical systems problem because machine learning services increasingly process sensitive biomedical, financial, industrial, and personal data whose exposure can create legal, ethical, and economic harm. The central challenge is not simply to design faster neural accelerators or stronger cryptographic protocols, but to align hardware architecture, cryptographic machinery, deployment infrastructure, and governance structures across the entire inference lifecycle. This paper examines hardware-aware acceleration of secure deep learning inference from a system-level perspective. It analyzes how trusted execution environments, homomorphic encryption, secure multiparty computation, differential privacy, and hybrid protocols interact with accelerators such as graphics processing units, field-programmable gate arrays, application-specific integrated circuits, and confidential computing platforms. Particular attention is given to structural trade-offs involving latency, memory bandwidth, energy consumption, cost, tenancy, robustness, fairness, and sustainability. The discussion also considers supply-chain governance, attestation, auditing, policy implications, and the uneven deployment realities of cloud, edge, and embedded environments. A central argument is that secure inference performance cannot be understood as a narrow optimization problem. Instead, it is a socio-technical design problem in which hardware capabilities, security assumptions, operational incentives, and regulatory expectations jointly determine what forms of private inference are technically feasible and socially legitimate. The paper concludes with a forward-looking research agenda for verifiable, efficient, equitable, and sustainable secure inference infrastructures.

Keywords

secure inference; hardware acceleration; trusted execution environments; homomorphic encryption; privacy-preserving machine learning; confidential computing; systems architecture; governance; sustainability.

1. Introduction

Deep learning inference has become a pervasive computational service, supporting medical diagnosis, financial risk scoring, autonomous systems, recommendation engines, and large-scale language processing. As these services move from experimental settings into regulated and safety-critical domains, the data flowing through them becomes both valuable and dangerous to expose. Secure deep learning inference addresses this tension by attempting to compute predictions without revealing sensitive inputs, model parameters, or intermediate activations to unauthorized parties. The field sits at the intersection of cryptography, computer architecture, operating systems, distributed systems, and privacy policy, and its practical success depends on whether security guarantees can be delivered at acceptable latency, cost, and energy. Foundational surveys of homomorphic encryption clarify both the promise and the computational burden of operating on encrypted data [1], while specialized secure inference frameworks have demonstrated that neural network evaluation can be partially transformed into cryptographic protocols [2], [3], [4]. These advances have made private inference conceivable, but they have not resolved the deeper systems problem of mapping secure computation onto heterogeneous hardware without sacrificing usability.

A hardware-aware perspective is necessary because secure inference workloads differ fundamentally from ordinary neural network inference. Ordinary inference optimizes arithmetic intensity, memory locality, parallelism, and throughput. Secure inference must additionally account for encrypted operands, oblivious memory access, secret sharing, trusted execution boundaries, attestation, key management, and the possibility that the hosting infrastructure itself is untrusted. Early secure machine learning systems revealed that the choice between garbled circuits, secret sharing, and homomorphic encryption changes not only the security model but also the hardware bottlenecks that dominate execution [5], [6]. Trusted execution environments offer a different route by isolating computation inside hardware-enforced enclaves, but they introduce their own constraints related to enclave memory, side channels, attestation, and vendor trust [7], [8]. A system-level analysis must therefore compare these approaches not as isolated techniques but as architectural commitments with long-term consequences for deployment, governance, and equity.

This paper argues that hardware-aware acceleration of secure deep learning inference should be understood as a socio-technical design space. The objective is not merely to accelerate cryptographic primitives or neural operators, but to construct an infrastructure in which privacy guarantees, performance targets, auditability, and sustainability can coexist. That requires attention to threat models, accelerator design, protocol selection, cloud tenancy, regulatory compliance, and the lifecycle costs of specialized hardware. The sections that follow examine these dimensions in depth, moving from threat models to hardware landscapes, cryptographic and trusted execution approaches, structural trade-offs, governance, fairness, deployment, sustainability, case illustrations, and a forward-looking research agenda.

2. Threat Models and Security Objectives

Secure inference begins with an explicit threat model. In some deployments, the data owner distrusts the model owner; in others, the model owner distrusts the infrastructure provider; in still others, multiple parties must jointly evaluate a model without revealing their respective inputs or weights. Cryptographic secure inference typically assumes an adversarial server or client and seeks confidentiality of inputs, model parameters, and sometimes outputs. Trusted execution approaches assume that a hardware root of trust can protect code and data from a malicious operating system or cloud operator, though not necessarily from physical attacks or microarchitectural side channels. These assumptions are not interchangeable. A protocol that

is secure against a curious cloud provider may be vulnerable to a compromised client, and a hardware enclave that protects model weights may still leak information through access patterns or timing.

The security objectives of secure inference extend beyond confidentiality. Integrity ensures that the model, code, and inputs have not been tampered with, while verifiability allows a client to confirm that a specific model produced a specific output. Privacy also includes protection against inference attacks. Membership inference can reveal whether a particular record was used during training [16], and model extraction can allow an adversary to approximate a proprietary model through repeated queries. Differential privacy provides a formal framework for limiting information leakage about individuals during training or inference [13]. Secure aggregation allows multiple parties to contribute updates without exposing individual contributions [12], and federated learning distributes training across clients while keeping raw data local [11]. These techniques are complementary rather than substitutable. A complete secure inference architecture may combine hardware isolation, cryptographic computation, differential privacy, and audit mechanisms, each addressing different adversaries and failure modes.

Threat models also determine what counts as acceptable overhead. If the adversary is a passive network observer, transport-layer encryption and access control may suffice. If the adversary controls the cloud host, then enclaves, encrypted computation, or trusted hardware become necessary. If the adversary can issue unlimited queries to a deployed model, then output perturbation, rate limiting, and monitoring become relevant. The choice of threat model therefore shapes hardware requirements. High-assurance cryptographic inference may demand large memory bandwidth for ciphertext operations, while enclave-based inference may demand secure memory paging and attestation acceleration. A system that ignores this coupling risks deploying hardware that is fast for the wrong threat model or secure for a workload that cannot tolerate its latency.

3. Hardware-Aware Acceleration Landscape

The hardware landscape for deep learning inference is already highly differentiated. Central processing units offer flexibility and mature software stacks but often lack the throughput needed for large matrix operations. Graphics processing units provide massive parallelism and high memory bandwidth, making them dominant for training and increasingly important for inference. Field-programmable gate arrays allow customized data paths and low-latency execution, while application-specific integrated circuits deliver the highest efficiency for fixed workloads. Domain-specific architectures for deep neural networks exploit convolution, matrix multiplication, activation functions, and memory reuse [17]. Within secure inference, these same hardware choices acquire new constraints. Encrypted data expands operand size, secret sharing requires communication between parties, and oblivious access patterns can destroy the locality assumptions on which many accelerators depend.

Hardware-aware acceleration therefore involves more than selecting the fastest chip. It requires co-design across algorithms, compilers, memory hierarchies, and security protocols. Model compression techniques such as pruning and quantization reduce arithmetic and memory demands [19], [20], but their interaction with cryptographic operations is complex. Quantized integer arithmetic may map well to homomorphic encryption, yet nonlinear activation functions remain expensive because they are not naturally compatible with additive or multiplicative homomorphism. Recent work on automatic activation replacement shows how nonlinear operations can be restructured for faster private network inference [18]. This

line of research illustrates a broader principle: secure inference performance depends on transforming the model itself, not merely accelerating the original computation. Large transformer-based models further complicate the picture because their attention mechanisms and parameter counts impose substantial memory and communication costs [21], [22]. Hardware-aware secure inference must therefore consider model architecture, operator selection, and cryptographic representation as mutually dependent decisions.

A related concern is the mismatch between cloud-scale accelerators and edge-scale constraints. Cloud inference can amortize specialized hardware across many tenants, but multi-tenancy raises isolation and side-channel risks. Edge inference keeps data closer to the source and can reduce network exposure, but embedded devices have limited memory, power, and thermal budgets. Secure inference on mobile or medical devices may need to rely on trusted execution, lightweight cryptography, or split computation between device and server. The hardware landscape is thus not a single ladder of performance but a set of deployment regimes with different bottlenecks. A hardware-aware design must specify whether it targets data-center throughput, edge latency, embedded energy, or regulated on-premises computation.

4. Cryptographic and Trusted Execution Approaches

Cryptographic secure inference can be organized into several families. Homomorphic encryption allows computation directly on ciphertexts and is attractive for outsourced inference because the server never sees plaintext [1]. However, homomorphic operations are computationally expensive, and deep networks with many nonlinear layers require careful approximation or protocol switching. Secure multiparty computation distributes computation across parties using secret sharing or garbled circuits, offering strong security guarantees at the cost of communication [5], [6]. Hybrid frameworks combine different primitives to handle linear and nonlinear layers efficiently [2], [3], [4]. Some systems classify encrypted data using specialized cryptographic protocols [10]. These approaches provide strong confidentiality but often require the client and server to interact, which complicates latency-sensitive deployment.

Trusted execution environments provide a contrasting model. Instead of hiding data from the processor, they isolate computation within a hardware-protected enclave so that even a privileged operating system cannot read enclave memory [7], [8]. This can dramatically reduce cryptographic overhead for neural network inference because the model and inputs can be processed in plaintext inside the enclave. However, enclave-based inference shifts trust to the hardware vendor and the attestation infrastructure. It also faces side-channel risks, limited secure memory, and performance penalties for context switching or encrypted paging. Systems such as Slalom explores private and verifiable execution of neural networks in trusted hardware, showing both the potential and the constraints of this approach [9]. In practice, many architectures combine enclaves with cryptographic protocols: enclaves may protect keys, manage attestation, or handle nonlinear layers, while homomorphic or secret-sharing methods protect data in untrusted memory.

The choice between cryptographic and trusted execution approaches is rarely absolute. It depends on the adversary, the regulatory environment, the value of the model, and the acceptable overhead. Cryptographic methods offer transparent, mathematically grounded guarantees but can be orders of magnitude slower for large models. Trusted execution offers performance closer to ordinary inference but introduces supply-chain trust, side-channel exposure, and vendor dependency. A hardware-aware secure inference system must evaluate these trade-offs at the level of individual operators, layers, and data flows. The most

promising designs are hybrid, using hardware acceleration for expensive cryptographic primitives, secure enclaves for sensitive control logic, and cryptographic protocols for cross-domain trust.

5. Structural Trade-Offs Across Latency, Memory, Energy, and Cost

Latency is often the first constraint considered in secure inference. Interactive applications such as voice assistants, augmented reality, and clinical decision support require responses within milliseconds to seconds. Cryptographic secure inference may add communication rounds and heavy computation, making real-time deployment difficult. Trusted execution can reduce latency but may introduce enclave entry and exit costs, encrypted memory paging, and contention in multi-tenant clouds. Hardware acceleration can narrow the gap, but only if the accelerator is designed for the relevant secure operations. A graphics processing unit optimized for dense matrix multiplication may be inefficient for modular arithmetic, oblivious memory access, or secret-sharing communication. Conversely, a cryptographic accelerator may excel at ciphertext operations but struggle with the irregular control flow of deep networks.

Memory bandwidth is frequently the hidden bottleneck. Deep learning inference already stresses memory due to large weight matrices and activations. Secure inference can multiply memory pressure because ciphertexts are larger than plaintext values, secret shares must be stored and exchanged, and oblivious access patterns prevent caching optimizations. Enclave-based inference faces additional constraints because secure memory is limited and paging to untrusted memory requires encryption and integrity protection [7], [9]. These constraints make model compression, operator fusion, and memory-aware scheduling essential [17], [19], [20]. Hardware-aware acceleration must therefore consider not only arithmetic throughput but also data movement, on-chip buffering, and communication topology.

Energy and cost are equally important but often neglected. Secure inference consumes additional energy for encryption, communication, attestation, and redundant computation. In cloud data centers, this translates into higher operational expenditure and carbon emissions. At the edge, it affects battery life and thermal design. Specialized accelerators can improve energy efficiency for specific operations, but their embodied energy and manufacturing footprint must be considered over the lifecycle. A narrow focus on peak performance can lead to architectures that are efficient in benchmarks but expensive in real deployments. Sustainable secure inference requires evaluating performance per watt, utilization rates, hardware refresh cycles, and the environmental cost of security overhead. The trade-offs are structural: stronger security often increases computation, and faster computation may require specialized hardware whose production and disposal have environmental consequences.

6. Architecture and Infrastructure Governance

Secure inference does not operate in a vacuum. It depends on infrastructure governance, including hardware supply chains, attestation services, key management, identity systems, and cloud tenancy models. Trusted execution environments rely on attestation to prove that the correct code is running in a genuine enclave [7], [8]. That attestation must be rooted in hardware, but the resulting trust chain may include vendor services, certificate authorities, and cloud providers. If any link is compromised or opaque, the security guarantee weakens. Cryptographic secure inference depends on key generation, distribution, rotation, and revocation. Poor key management can undermine even strong protocols. Governance therefore becomes a technical requirement, not merely a compliance exercise.

Multi-tenant infrastructure raises additional concerns. Cloud providers may colocate secure inference workloads with untrusted tenants, creating side-channel and denial-of-service risks. Resource isolation, scheduling, and accounting must be designed to prevent one tenant from inferring information about another through timing, cache, or power side channels. Regulatory frameworks may require audit trails, data residency, and reproducibility, which in turn affect where computation can occur and how long logs are retained. Internal algorithmic auditing frameworks emphasize documenting data, models, and decisions across the lifecycle [15]. For secure inference, such auditing must be reconciled with privacy guarantees: auditors may need to verify correct behavior without accessing sensitive inputs or proprietary weights. This tension calls for cryptographic auditing techniques, trusted third parties, and privacy-preserving transparency mechanisms.

Governance also shapes market structure. If secure inference requires proprietary hardware from a small number of vendors, then access, pricing, and interoperability become policy concerns. Open standards for attestation, confidential computing, and secure accelerators could reduce lock-in, but they may also slow innovation or create lowest-common-denominator security. Public-sector deployments in healthcare, welfare, and criminal justice may require stronger accountability than commercial services. The governance challenge is to align technical security properties with legal and ethical expectations without making secure inference so costly that only the largest institutions can afford it.

7. Robustness, Fairness, and Policy Implications

Secure inference can change the robustness properties of a machine learning system. Approximations used to make nonlinear functions compatible with homomorphic encryption may introduce numerical errors that affect predictions. Quantization and pruning, while beneficial for efficiency, can amplify bias or reduce calibration [19], [20]. Enclave-based inference may be robust to external tampering but vulnerable to side-channel leakage that reveals model behavior. Adversarial inputs may exploit not only the model but also the secure computation protocol, for example by triggering error messages or timing variations. A secure inference system must therefore be evaluated for both security and model quality under realistic perturbations.

Fairness is another dimension that cannot be separated from systems design. If secure inference is deployed for credit scoring, hiring, medical triage, or public benefits, it may encode historical biases [14]. Privacy protections can make it harder to detect discrimination because sensitive attributes and outcomes are obscured. Differential privacy can reduce membership leakage [13], but it may disproportionately affect minority subgroups if privacy budgets are allocated uniformly. Secure aggregation in federated learning can protect client updates [12], yet it can also hide harmful or biased contributions. Policy frameworks increasingly demand impact assessments, transparency, and redress. Secure inference architectures should therefore include mechanisms for fairness auditing, subgroup evaluation, and contestability without exposing individual data.

Policy implications extend to liability and accountability. If a secure inference pipeline produces a harmful decision, who is responsible: the model owner, the data owner, the hardware vendor, the cloud provider, or the auditor? Attestation logs and cryptographic proofs can support accountability, but they must be interpretable to regulators and courts. Data protection laws may treat encrypted or enclave-protected data differently, creating incentives for specific technical architectures. Export controls on advanced accelerators and encryption technologies can also shape global deployment. A socio-technical approach recognizes that

secure inference is not just a privacy tool; it is an infrastructure that redistributes trust, power, and responsibility across institutions.

8. Deployment, Sustainability, and Lifecycle Considerations

Deployment models for secure inference range from on-premises appliances to public cloud services, edge gateways, and mobile devices. Each model has distinct sustainability implications. Cloud data centers can achieve high utilization and economies of scale, but they concentrate energy demand and require continuous cooling and hardware refresh. Edge devices may reduce data transmission and network energy, but they often have shorter lifetimes, limited recyclability, and constrained power budgets. Specialized accelerators can improve computational efficiency, yet their manufacturing involves rare materials, complex supply chains, and significant embodied carbon. A hardware-aware analysis must consider the full lifecycle, including fabrication, transportation, operation, and disposal.

Security overhead directly affects sustainability. Homomorphic encryption and secure multiparty computation can increase computation and communication by orders of magnitude, leading to higher energy consumption per inference [1], [2], [3]. Trusted execution can be more efficient but may require redundant hardware, attestation services, and secure memory that reduce effective utilization [7], [9]. Hardware-aware acceleration can mitigate these costs, but only if it is matched to real workloads. Optimizing for peak throughput on a benchmark may produce hardware that is idle most of the time, wasting embodied energy. Conversely, aggressive model compression may reduce energy per inference but degrade accuracy, leading to repeated queries or human review that offset savings [19], [20]. Sustainable secure inference requires metrics that capture both operational and embodied impacts, as well as deployment policies that encourage longer hardware lifetimes, repairability, and shared infrastructure.

There is also a social sustainability dimension. If secure inference remains expensive, it may be available only to wealthy institutions and individuals, widening gaps in privacy protection. Public health systems, small clinics, municipal governments, and community organizations may lack the resources to deploy trusted hardware or cryptographic accelerators. Open-source implementations, standardized APIs, and cloud-based confidential computing could lower barriers, but they may also concentrate dependence on a few providers. A just transition toward secure inference would require public investment, interoperability mandates, and attention to the distribution of privacy benefits and costs.

9. Case Illustrations and Cross-Domain Comparisons

Healthcare offers a compelling case for secure inference. Hospitals may wish to use a third-party diagnostic model without revealing patient records, while model developers may wish to protect proprietary weights. Cryptographic inference can provide strong confidentiality but may struggle with latency in urgent care settings [2], [3]. Enclave-based inference can process medical images quickly but requires trust in the cloud provider and careful side-channel mitigation [7], [9]. Regulatory requirements such as the Health Insurance Portability and Accountability Act in the United States and the General Data Protection Regulation in Europe add compliance layers that affect data residency, audit, and consent. A hybrid architecture might use enclaves for image preprocessing and cryptographic protocols for cross-institution collaboration, balancing speed and trust.

Financial services present a different profile. Fraud detection and credit scoring require low latency and high throughput, often across millions of transactions. Secure inference can

protect customer data and proprietary risk models, but the cost of cryptographic computation may be prohibitive at scale. Hardware acceleration using field-programmable gate arrays or application-specific integrated circuits could make encrypted scoring feasible, especially if the model is quantized and pruned [19], [20]. However, fairness concerns are acute because credit decisions affect life opportunities, and privacy protections may obscure discriminatory patterns [14]. Auditing frameworks must therefore be integrated into the inference pipeline [15]. Similar tensions appear in insurance, employment screening, and public benefits administration.

Autonomous systems and mobile computing illustrate edge constraints. An autonomous vehicle cannot rely on a distant cloud for safety-critical inference, and a smartphone cannot run large encrypted models without draining its battery. Trusted execution within the device may protect sensitive data from a compromised operating system, but physical attacks and side channels remain risks [8], [9]. Split inference, where early layers run on the device and later layers on a secure server, can balance privacy and latency, but it may leak information through intermediate representations. Cross-domain comparison shows that there is no universal secure inference architecture. The appropriate design depends on the threat model, latency budget, regulatory context, hardware available, and social consequences of error.

10. Research Agenda and Forward-Looking Perspectives

Future research should pursue deeper co-design across cryptography, hardware, and machine learning. Secure operators should be treated as first-class primitives in accelerator design, just as convolution and matrix multiplication are today. Compilers should reason about security constraints, memory movement, and energy consumption simultaneously. Model architectures should be searched not only for accuracy and efficiency but also for compatibility with secure computation. The automatic transformation of activation functions for private inference is an early example of this direction [18]. Extending such methods to attention, normalization, and generative models will be essential as transformer-based systems become more widespread [21], [22]. Hardware-aware secure inference must also address the growing scale of foundation models, where memory bandwidth and communication may dominate even ordinary inference.

A second research direction concerns verification and accountability. Formal methods can help prove that a secure inference implementation enforces its stated policy, but scaling verification to large neural networks and complex hardware remains difficult. Attestation and cryptographic proofs can support remote verification, yet they must be usable by regulators, auditors, and end users. Differential privacy provides quantifiable guarantees [13], but composition across many queries and models requires careful privacy budgeting. Federated learning and secure aggregation offer distributed privacy, but they raise questions about model poisoning, fairness, and governance [11], [12]. Future systems should integrate privacy, security, fairness, and sustainability metrics into a unified evaluation framework.

A third direction is policy and institutional design. Standards bodies, regulators, and professional organizations should develop guidelines for secure inference that address attestation, transparency, liability, and environmental impact. Procurement policies can incentivize interoperable and repairable hardware rather than proprietary lock-in. Funding agencies can support open benchmarks that measure latency, energy, accuracy, fairness, and privacy under realistic threat models. Education and workforce development are also necessary so that engineers, clinicians, auditors, and policymakers can reason about the socio-

technical nature of secure inference. Without such institutional support, technical advances may remain confined to well-resourced organizations and fail to deliver broad public benefit.

11. Conclusion

Hardware-aware acceleration of secure deep learning inference is a multidimensional systems challenge. It requires aligning threat models with cryptographic protocols, trusted hardware, accelerator architectures, memory hierarchies, deployment models, and governance structures. Cryptographic approaches such as homomorphic encryption and secure multiparty computation offer strong confidentiality but impose significant computational and communication overhead [1], [2], [3], [4], [5], [6], [10]. Trusted execution environments reduce overhead but shift trust to hardware vendors and introduce side-channel and attestation concerns [7], [8], [9]. Hardware acceleration can narrow the performance gap, but only when it is co-designed with secure operators, model compression, and deployment constraints [17], [18], [19], [20]. The growing scale of transformer and foundation models further intensifies the need for memory-efficient and communication-aware secure inference [21], [22].

The paper has argued that secure inference cannot be evaluated solely by latency or cryptographic strength. It must also be assessed for robustness, fairness, accountability, energy use, and equitable access. Privacy protections can obscure discrimination, and security overhead can worsen environmental impact. Governance mechanisms such as attestation, auditing, and standards are therefore integral to the technical architecture. Future systems should integrate privacy, fairness, and sustainability metrics from the beginning rather than treating them as afterthoughts. A just and effective secure inference infrastructure will require collaboration among hardware architects, cryptographers, machine learning researchers, cloud operators, regulators, and civil society. The goal is not merely faster private computation, but a trustworthy and sustainable foundation for sensitive machine learning services in society.

References

1. Acar, A., Aksu, H., Uluagac, A. S., & Conti, M. (2018). A survey on homomorphic encryption schemes: Theory and implementation. ACM Computing Surveys, 51(4), 1-35. https://doi.org/10.1145/3214303

2. Juvekar, C., Vaikuntanathan, V., & Chandrakasan, A. (2018). GAZELLE: A low latency framework for secure neural network inference. In Proceedings of the 27th USENIX Security Symposium (pp. 1651-1669). USENIX Association.

3. Mishra, P., Lehmkuhl, R., Srinivasan, A., Zheng, W., & Popa, R. A. (2020). Delphi: A cryptographic inference service for neural networks. In Proceedings of the 29th USENIX Security Symposium (pp. 2505-2522). USENIX Association.

4. Liu, J., Juuti, M., Lu, Y., & Asokan, N. (2017). Oblivious neural network predictions via MiniONN transformations. In Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security (pp. 619-631). ACM. https://doi.org/10.1145/3133956.3134056

5. Mohassel, P., & Zhang, Y. (2017). SecureML: A system for scalable privacy-preserving machine learning. In 2017 IEEE Symposium on Security and Privacy (pp. 19-38). IEEE. https://doi.org/10.1109/SP.2017.12

6. Riazi, M. S., Weinert, C., Tkachenko, O., Songhori, E. M., Schneider, T., & Koushanfar, F. (2018). Chameleon: A hybrid secure computation framework for machine learning

applications. In Proceedings of the 2018 on Asia Conference on Computer and Communications Security (pp. 707-721). ACM. https://doi.org/10.1145/3196494.3196522

7. Costan, V., & Devadas, S. (2016). Intel SGX explained. IACR Cryptology ePrint Archive, 2016, 86. https://eprint.iacr.org/2016/086

8. Sabt, M., Achemlal, M., & Bouabdallah, A. (2015). Trusted execution environment: What it is, and what it is not. In 2015 IEEE Trustcom/BigDataSE/ISPA (Vol. 1, pp. 57-64). IEEE. https://doi.org/10.1109/Trustcom.2015.357

9. Tramèr, F., & Boneh, D. (2019). Slalom: Fast, verifiable and private execution of neural networks in trusted hardware. In International Conference on Learning Representations. OpenReview.

10. Bost, R., Popa, R. A., Tu, S., & Goldwasser, S. (2015). Machine learning classification over encrypted data. In Proceedings of the 2015 Network and Distributed System Security Symposium. Internet Society. https://doi.org/10.14722/ndss.2015.23241

11. Kairouz, P., McMahan, H. B., Avent, B., Bellet, A., Bennis, M., Bhagoji, A. N., ... & Zhao, S. (2021). Advances and open problems in federated learning. Foundations and Trends in Machine Learning, 14(1-2), 1-210. https://doi.org/10.1561/2200000083

12. Bonawitz, K., Ivanov, V., Kreuter, B., Marcedone, A., McMahan, H. B., Patel, S., ... & Seth, K. (2017). Practical secure aggregation for privacy-preserving machine learning. In Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security (pp. 1175-1191). ACM. https://doi.org/10.1145/3133956.3133982

13. Dwork, C., & Roth, A. (2014). The algorithmic foundations of differential privacy. Foundations and Trends in Theoretical Computer Science, 9(3-4), 211-407. https://doi.org/10.1561/0400000042

14. Barocas, S., Hardt, M., & Narayanan, A. (2019). Fairness and machine learning: Limitations and opportunities. fairmlbook.org.

15. Raji, I. D., Smart, A., White, R. N., Mitchell, M., Gebru, T., Hutchinson, B., ... & Barnes, P. (2020). Closing the AI accountability gap: Defining an end-to-end framework for internal algorithmic auditing. In Proceedings of the 2020 Conference on Fairness, Accountability, and Transparency (pp. 33-44). ACM. https://doi.org/10.1145/3351095.3372873

16. Shokri, R., Stronati, M., Song, C., & Shmatikov, V. (2017). Membership inference attacks against machine learning models. In 2017 IEEE Symposium on Security and Privacy (pp. 3-18). IEEE. https://doi.org/10.1109/SP.2017.41

17. Sze, V., Chen, Y. H., Yang, T. J., & Emer, J. S. (2017). Efficient processing of deep neural networks: A tutorial and survey. Proceedings of the IEEE, 105(12), 2295-2329. https://doi.org/10.1109/JPROC.2017.2761740

18. Peng, H., Huang, S., Zhou, T., Luo, Y., Wang, C., Wang, Z., ... & Ding, C. (2023, October). Autorep: Automatic relu replacement for fast private network inference. In 2023 IEEE/CVF International Conference on Computer Vision (ICCV) (pp. 5155-5165). IEEE.

19. Han, S., Pool, J., Tran, J., & Dally, W. J. (2015). Learning both weights and connections for efficient neural networks. Advances in Neural Information Processing Systems, 28, 1135-1143.

20. Jacob, B., Kligys, S., Chen, B., Zhu, M., Tang, M., Howard, A., ... & Kalenichenko, D. (2018). Quantization and training of neural networks for efficient integer-arithmetic-only inference. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (pp. 2704-2713). IEEE.

21. Devlin, J., Chang, M. W., Lee, K., & Toutanova, K. (2019). BERT: Pre-training of deep bidirectional transformers for language understanding. In Proceedings of NAACL-HLT 2019 (pp. 4171-4186). Association for Computational Linguistics. https://doi.org/10.18653/v1/N19-1423

22. Brown, T. B., Mann, B., Ryder, N., Subbiah, M., Kaplan, J., Dhariwal, P., ... & Amodei, D. (2020). Language models are few-shot learners. Advances in Neural Information Processing Systems, 33, 1877-1901.