

# **Adaptive Neural Architecture Search for Cryptographically Efficient Private Inference Under Latency and Accuracy Constraints**

Adamo Paltrinieri

Department of Computer Science, Binghamton University, Binghamton, NY, USA.  
paltrinieri807@binghamton.edu

Udarsh Andra

Department of Electrical Engineering and Computer Science, University of Kansas, Lawrence, KS, USA.  
udarsha@ku.edu

Morris Draaisma

Department of Computer Science, University of Alabama at Birmingham, Birmingham, AL, USA.  
morrisd@uab.edu

Nitara Swaminathan

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

## **Abstract**

Private inference has become a central concern for machine learning systems that process sensitive user data across organizational, jurisdictional, and infrastructural boundaries. Cryptographic techniques such as secure multi-party computation and homomorphic encryption can preserve confidentiality during inference, but they impose substantial computational and communication overheads that interact poorly with conventional deep neural network design. This paper examines adaptive neural architecture search as a system-level mechanism for producing cryptographically efficient private inference pipelines under explicit latency and accuracy constraints. Rather than treating architecture search as a model-centric optimization problem, the analysis frames it as a socio-technical infrastructure capability that links model topology, cryptographic protocol selection, deployment topology, governance, and lifecycle sustainability. The paper argues that adaptive search methods can reduce non-linear activation cost, communication rounds, memory pressure, and operational uncertainty, but only when they are constrained by deployment realities and accountable design practices. It further considers robustness, fairness, policy implications, and cross-domain comparisons, including healthcare, finance, and public-sector applications. The conclusion is that cryptographically efficient private inference requires not merely better search algorithms but also institutional mechanisms that make latency, accuracy, privacy, and sustainability trade-offs visible, auditable, and governable over time.

## **Keywords**

adaptive neural architecture search; private inference; secure multi-party computation; homomorphic encryption; latency constraints; accuracy constraints; trustworthy machine learning; socio-technical systems.

## 1. Introduction

Private inference refers to the execution of machine learning predictions without revealing the input data, the model parameters, or both to the parties involved in the computation. This requirement arises in domains where data cannot be freely centralized because of legal restrictions, competitive sensitivity, or ethical obligations. Hospitals may wish to use a remote diagnostic model without exporting patient records, financial institutions may need to evaluate risk without exposing transaction histories, and public agencies may seek to deploy analytical services without creating centralized repositories of citizen data. Cryptographic methods such as secure multi-party computation and homomorphic encryption offer formal privacy guarantees, but their costs are often prohibitive for large models and interactive services. The resulting design problem is not simply one of cryptography or machine learning; it is a systems problem in which architecture, protocol, infrastructure, and governance jointly determine whether private inference is practical [9], [10], [11].

Neural architecture search has emerged as a way to automate the discovery of model topologies that satisfy multiple objectives. Early work demonstrated that reinforcement learning and evolutionary search could discover high-performing image classifiers, while later methods made search more efficient through parameter sharing, differentiable relaxations, and regularized evolution [1], [2], [3], [4], [5]. Efficient model families such as EfficientNet, residual networks, and MobileNetV2 further show that carefully structured topologies can deliver strong accuracy with lower computational cost [6], [7], [8]. However, most architecture search research optimizes conventional metrics such as floating-point operations, parameter count, or latency on commodity hardware. Those metrics do not capture the cost structure of private inference, where the most expensive operations may be non-linear activations, secure comparisons, multiplications in secret-shared form, or communication rounds between parties.

This paper develops a system-level account of adaptive neural architecture search for cryptographically efficient private inference under latency and accuracy constraints. The central claim is that adaptive search can be valuable not because it replaces cryptographic engineering but because it creates a structured space in which model design, protocol selection, and deployment constraints can be co-optimized. Yet this co-optimization also introduces governance challenges. Search objectives encode value judgments about acceptable latency, minimum accuracy, privacy leakage tolerances, and resource expenditure. When such systems are deployed in healthcare, finance, or public administration, those judgments must be made visible to auditors, regulators, and affected communities. The paper therefore connects architecture search to broader questions of robustness, fairness, sustainability, and policy.

## 2. Background and Related Work

The literature on neural architecture search spans reinforcement learning, evolutionary computation, differentiable search, and performance prediction. The reinforcement learning approach formulated architecture generation as a sequence of discrete choices, while subsequent work improved efficiency through weight sharing and continuous relaxation [1], [3], [4]. Evolutionary methods introduced population-based exploration and mutation, often

producing robust architectures at the cost of significant search computation [5]. Surveys of the field emphasize that search cost, transferability, and reproducibility remain persistent concerns [2]. Efficient architecture families provide another perspective: residual connections ease optimization, inverted residuals and linear bottlenecks reduce mobile inference cost, and compound scaling balances depth, width, and resolution [6], [7], [8]. These contributions are relevant to private inference because they demonstrate that architecture can materially change computational structure, not merely parameter count.

Private inference research has developed a parallel repertoire of protocols and system designs. SecureML introduced scalable privacy-preserving machine learning using secret sharing and garbled circuits, while ABY3 explored mixed-protocol computation for machine learning workloads [9], [10]. GAZELLE demonstrated low-latency secure neural network inference by combining homomorphic encryption with garbled circuits, and MiniONN transformed neural networks into oblivious inference protocols [11], [12]. CryptoNets showed that encrypted data could be processed with high throughput, though with substantial latency and limited depth [13]. CHET introduced compiler-level optimization for fully homomorphic neural network inference, and CryptTen provided a framework that brings secure multi-party computation closer to mainstream machine learning practice [14], [15]. These systems reveal that cryptographic efficiency depends on the interaction between network topology, activation functions, protocol choice, and hardware acceleration.

Federated learning and distributed infrastructure research add another layer. Federated learning addresses decentralized training and inference across clients, and system designs have explored scalability, communication efficiency, and fault tolerance [16], [17]. However, federated learning alone does not provide cryptographic confidentiality against all participants; secure aggregation and private inference protocols are often needed to strengthen guarantees. Recent work on ReLU-reduced neural networks for two-party computation based private inference illustrates the growing interest in architectures that reduce expensive non-linear operations [18]. Security and privacy analyses of machine learning further warn that private inference cannot be reduced to a single cryptographic primitive; it must account for leakage, side channels, model extraction, and adversarial manipulation [19]. Governance instruments such as model cards, datasheets for datasets, and ethical AI critiques provide complementary mechanisms for accountability [20], [21], [22].

### **3. System-Level Challenges in Private Inference**

Private inference is fundamentally constrained by the mismatch between conventional neural network design and cryptographic cost models. In plaintext inference, matrix multiplications dominate latency and energy consumption, and hardware accelerators are optimized accordingly. In secure computation, however, multiplication may be relatively efficient under certain protocols while non-linear activation functions such as ReLU require secure comparisons or garbled circuits that introduce communication rounds and significant overhead. Homomorphic encryption can evaluate some polynomial operations efficiently, but deep networks with many non-linear layers become costly. Secret-sharing protocols can parallelize computation across parties, but they depend on network bandwidth, synchronization, and trust assumptions. Consequently, a model that is efficient on a GPU may be inefficient under private inference, and a model that is cryptographically friendly may sacrifice accuracy or robustness [9], [11], [13], [18].

The system-level challenge is therefore multidimensional. It includes latency, throughput, monetary cost, energy consumption, memory footprint, communication volume, and

operational complexity. Latency constraints are especially important for interactive applications such as clinical decision support or fraud detection, where users expect responses within seconds. Accuracy constraints are equally significant: in medical triage or credit assessment, a small reduction in predictive quality may have distributive consequences. Privacy constraints impose cryptographic protocols whose costs vary with network depth, activation type, and batch size. Deployment constraints include available hardware, network topology, regulatory jurisdiction, and the ability of operators to manage key material and audit logs. A private inference system must satisfy all these constraints simultaneously, which makes architecture search an attractive but difficult mechanism for coordination [10], [14], [15].

Another challenge is that private inference systems often span multiple administrative domains. A model provider may own the architecture and weights, a service operator may manage the cryptographic infrastructure, and a client may hold sensitive inputs. Each party may have different incentives and risk tolerances. The model provider may prioritize accuracy and intellectual property protection, the operator may prioritize latency and cost, and the client may prioritize confidentiality and regulatory compliance. Architecture search that is conducted only by the model provider may overlook deployment realities, while search conducted only by the operator may undervalue model quality. Adaptive search can help align these perspectives if it incorporates constraints from all parties, but it can also obscure accountability if objectives are opaque or if search results are difficult to audit [16], [17], [20].

#### **4. Adaptive Neural Architecture Search as an Infrastructure Mechanism**

Adaptive neural architecture search can be understood as an infrastructure mechanism because it shapes the computational graph that later determines protocol selection, hardware provisioning, and service-level guarantees. In a private inference pipeline, the architecture determines which operations must be evaluated securely, how many communication rounds are required, and how much intermediate state must be protected. Search methods that consider cryptographic cost can therefore reduce the burden on the protocol layer. For example, architectures with fewer ReLU activations may lower the number of secure comparisons, while architectures with shallower non-linear paths may reduce the depth of homomorphic evaluation. Search can also favor operations that are compatible with efficient secret-sharing primitives, such as additions and multiplications by constants [1], [3], [18].

Adaptivity is important because private inference requirements vary across deployments. A hospital may require strict privacy and moderate latency, a financial institution may require very low latency for transaction screening, and a public health agency may prioritize broad accessibility and auditability over peak performance. A single fixed architecture is unlikely to satisfy all contexts. Adaptive search can produce families of architectures that trade accuracy for latency or cryptographic cost, allowing operators to select a configuration that matches their governance requirements. This perspective shifts architecture search from a one-time model discovery process to a continuous lifecycle capability. As protocols improve, hardware changes, or regulations evolve, the search space and constraints can be updated [2], [5], [6].

However, adaptive search also introduces new dependencies. Search itself consumes computational resources and may require access to sensitive data for validation. If search is performed on client data, privacy risks may arise unless the search process is also protected. If search is performed on proxy data, the resulting architectures may not generalize to the target population. Moreover, search objectives may be gamed or poorly specified. A search that minimizes average latency may produce architectures that are fast for typical inputs but

fragile for outliers. A search that maximizes accuracy may ignore fairness across subgroups. These concerns suggest that architecture search should be embedded in a governance framework that defines acceptable constraints, evaluates distributional outcomes, and documents design decisions [19], [20], [21].

## **5. Latency, Accuracy, and Cryptographic Efficiency Trade-Offs**

The central trade-off in cryptographically efficient private inference is among latency, accuracy, and cryptographic cost. Latency includes computation time, communication rounds, and queuing delays. Accuracy includes not only top-line predictive performance but also calibration, robustness, and subgroup reliability. Cryptographic cost includes computational overhead, bandwidth, memory, and key management complexity. These dimensions are not independent. Increasing model depth may improve accuracy but can increase the number of secure operations and communication rounds. Reducing non-linear activations may lower cryptographic cost but can reduce representational capacity. Quantization may reduce communication volume but can degrade accuracy or introduce leakage risks if not carefully designed. Architecture search must therefore operate over a constrained Pareto frontier rather than a single scalar objective [6], [7], [11], [14].

Recent work on ReLU-reduced neural networks for two-party computation illustrates how architectural choices can directly target cryptographic efficiency [18]. By reducing the reliance on ReLU activations, such architectures can lower the number of secure comparisons and garbled-circuit evaluations, which are often bottlenecks in private inference. This does not mean that non-linear activations can be eliminated entirely; rather, it suggests that search can identify alternative non-linearities or placement strategies that preserve accuracy while reducing secure computation cost. Similar logic applies to normalization layers, pooling operations, and attention mechanisms. Each component has a different cost profile under homomorphic encryption, secret sharing, or garbled circuits. Adaptive search can encode these profiles as constraints and search for architectures that remain within a latency budget while meeting an accuracy threshold [13], [15], [18].

The trade-off is also institutional. A service-level agreement may specify a maximum response time, but the cost of meeting that target may be borne by the operator, the client, or the environment. If latency is reduced by moving computation to a more powerful but less trusted server, privacy guarantees may weaken. If latency is reduced by simplifying the model, accuracy may decline for underrepresented groups. If latency is reduced by increasing parallelism, energy consumption may rise. Therefore, latency and accuracy constraints should not be treated as purely technical parameters. They express priorities about who waits, who bears cost, and who receives reliable service. Architecture search can make these priorities explicit, but only if the search process is transparent and if its outputs are evaluated across multiple populations and deployment scenarios [20], [21], [22].

## **6. Deployment, Governance, and Operational Assurance**

Deployment of private inference systems requires more than a trained model and a cryptographic protocol. It requires key management, identity and access control, audit logging, incident response, and continuous monitoring. In multi-party computation, parties must agree on protocol versions, input formats, and abort behavior. In homomorphic encryption, parameter selection affects security and performance, and key rotation must be coordinated. These operational concerns interact with architecture search because the chosen architecture determines the computational graph that must be executed, verified, and maintained. If the

architecture is too complex, operators may struggle to reproduce results or diagnose failures. If the architecture is too opaque, auditors may be unable to assess whether privacy claims are justified [10], [14], [17].

Governance mechanisms can help. Model cards document intended use, performance characteristics, and limitations, while datasheets for datasets describe provenance, collection methods, and ethical considerations [20], [21]. These instruments are not sufficient on their own, but they provide a vocabulary for accountability. In private inference, model cards could include cryptographic assumptions, protocol versions, latency distributions, accuracy across subgroups, and known leakage risks. Datasheets could clarify whether validation data was representative and whether search was conducted on data that reflects the deployment population. Such documentation is especially important when architecture search is automated, because the rationale for a selected architecture may otherwise be inaccessible to human decision-makers [22].

Operational assurance also requires attention to failure modes. A private inference service may fail because of network partitions, protocol timeouts, malformed inputs, or adversarial interference. Some failures may degrade gracefully, while others may leak information through error messages or timing. Architecture search can contribute to resilience by favoring topologies with predictable resource usage and fewer synchronization points. However, resilience cannot be guaranteed by architecture alone. It requires testing under adversarial conditions, monitoring for distribution shift, and maintaining fallback procedures. Governance must therefore connect architecture search to broader risk management, including security audits, privacy impact assessments, and redress mechanisms for affected users [19], [20].

## **7. Robustness, Fairness, and Policy Implications**

Robustness and fairness are often treated as separate from cryptographic efficiency, but they are deeply connected in private inference. A model that is robust to adversarial examples under plaintext evaluation may behave differently under secure computation because quantization, approximation, or protocol-specific transformations alter its decision boundaries. Similarly, a model that appears fair on average may produce disparate outcomes when latency constraints force the use of simplified architectures for some users or when privacy budgets limit access to representative validation data. Architecture search that optimizes only accuracy and latency may therefore produce systems that are technically efficient but socially uneven [19], [22].

Policy implications follow from these interactions. Data protection regimes may require purpose limitation, data minimization, and transparency, while sector-specific regulations may impose stricter requirements in healthcare or finance. Private inference can support compliance by reducing data exposure, but it does not automatically satisfy legal or ethical obligations. If a private inference system is trained on biased data, it can still produce discriminatory outcomes. If its audit logs are incomplete, it may be difficult to demonstrate compliance. If its latency constraints disproportionately affect rural or low-bandwidth users, access equity may suffer. Policymakers may therefore need to require not only privacy guarantees but also documentation of performance across groups, explanations of automated architecture choices, and avenues for contestation [20], [21], [22].

Fairness also has an infrastructural dimension. Private inference often depends on specialized hardware, high-bandwidth networks, and skilled operators. Organizations with fewer

resources may be unable to deploy or audit such systems, leading to a concentration of privacy-enhancing technologies among well-resourced institutions. Adaptive architecture search could mitigate this by producing efficient models that run on modest infrastructure, but it could also exacerbate inequality if search itself requires expensive compute. Public investment in shared testing environments, open benchmarks, and standardized documentation could help distribute the benefits of private inference more broadly. The policy question is not whether private inference is good or bad in the abstract, but how its costs and benefits are distributed across society [16], [22].

## **8. Sustainability and Lifecycle Considerations**

Sustainability is an increasingly important dimension of machine learning systems. Training large models consumes energy and water, and inference at scale can dominate lifecycle emissions. Private inference adds cryptographic computation and communication overhead, which may further increase energy use. Architecture search itself can be computationally expensive, especially when evolutionary or reinforcement learning methods require many trials [5], [6]. A sustainable approach must therefore consider the full lifecycle: search cost, training cost, deployment cost, and end-of-life decommissioning. Adaptive search can help by finding smaller or more cryptographically efficient models, but only if the search process is not so costly that it negates the savings [14], [16].

Lifecycle sustainability also involves maintainability. Cryptographic protocols evolve, hardware changes, and regulatory requirements shift. An architecture that is efficient today may become obsolete or insecure tomorrow. Systems that support modular replacement of protocols, clear separation between model and cryptographic layers, and reproducible search configurations are more likely to remain sustainable. Documentation practices such as model cards and datasheets support maintainability by preserving design rationale and known limitations [20], [21]. Without such practices, organizations may face repeated redesign costs or may continue using outdated protocols because the original architecture cannot be easily changed.

There is also an environmental justice dimension. The energy and water consumption of data centers may affect local communities, and the extraction of minerals for specialized hardware has social and ecological consequences. Private inference should not be framed as a purely virtual service; it depends on physical infrastructure with material impacts. Architecture search that reduces communication rounds and computation can contribute to sustainability, but it must be evaluated alongside broader procurement and deployment decisions. A system that is cryptographically efficient but runs on carbon-intensive energy may still be unsustainable. Sustainability therefore belongs in the same design space as latency, accuracy, and privacy [16], [22].

## **9. Case Illustrations and Cross-Domain Comparisons**

Healthcare offers a compelling case for private inference. A hospital may wish to use a third-party diagnostic model without sharing patient images or records. Secure multi-party computation and homomorphic encryption can protect data during inference, but latency constraints may be strict in emergency settings. An adaptive architecture search could identify models that preserve diagnostic accuracy while reducing secure activation costs. However, healthcare also requires careful attention to fairness, because models trained on non-representative populations may perform poorly for underrepresented groups. Governance

mechanisms such as model cards and datasheets can document these risks, but they must be integrated with clinical validation and regulatory review [9], [11], [20].

Finance presents a different profile. Fraud detection and credit scoring often require low-latency decisions at high volume. Private inference can protect customer transaction data and proprietary models, but the cost of secure computation may be difficult to justify for every transaction. Adaptive search could produce tiered architectures: a fast, lower-complexity model for routine screening and a more accurate, higher-cost model for suspicious cases. Such tiering raises policy questions about consistency and explainability. If different users are evaluated by different architectures, fairness and due process may be affected. Governance must ensure that tiering is based on legitimate risk factors rather than protected attributes or proxies [10], [15], [17].

Public-sector applications, including social services and tax administration, introduce additional concerns. Governments may have legal authority to process data, but public trust depends on transparency and accountability. Private inference could reduce data sharing between agencies, but it could also obscure decision-making if models and protocols are not auditable. Adaptive search may help meet budget and latency constraints, but it should not be used to avoid public scrutiny. Cross-domain comparison suggests that the same technical capability can have different governance implications depending on institutional context. Therefore, architecture search for private inference should be designed as a configurable capability that can be constrained by domain-specific rules, not as a universal optimizer [19], [21], [22].

## **10. Future Directions**

Future research should develop search objectives that are explicitly multi-objective and context-sensitive. Rather than optimizing a single latency or accuracy metric, search methods could produce a set of candidate architectures along a Pareto frontier, each accompanied by documentation of trade-offs. This would allow operators and regulators to choose configurations that align with legal and ethical requirements. Search could also incorporate uncertainty estimates, fairness metrics, and sustainability indicators, though these additions increase computational cost and may introduce new forms of opacity [2], [3], [18].

Another direction is the co-design of architectures and cryptographic protocols. If architecture search can propose topologies that are compatible with specific protocols, protocol designers can in turn expose cost models that search algorithms can query. This requires shared representations of operations, communication patterns, and security assumptions. It also requires benchmarks that reflect realistic network conditions and threat models. Federated and multi-party deployments add further complexity, because the search process may need to operate across organizations without centralizing sensitive data [15], [16], [17].

Finally, governance research should examine how automated architecture search can be made accountable. This includes audit trails for search decisions, standards for documenting constraints, and mechanisms for affected communities to challenge outcomes. Legal frameworks may need to clarify responsibility when an architecture is selected by an automated system that balances privacy, latency, and accuracy. Technical and policy communities should collaborate to ensure that private inference systems are not only efficient but also legitimate, contestable, and sustainable over time [20], [21], [22].

## **11. Conclusion**

Adaptive neural architecture search offers a promising systems-level approach to cryptographically efficient private inference under latency and accuracy constraints. By treating architecture as a variable that interacts with protocol cost, hardware, and governance, search can help reduce secure computation overhead while preserving acceptable model quality. However, the benefits of adaptive search depend on how constraints are defined, whose interests are represented, and how trade-offs are documented. Private inference is not merely a cryptographic problem or a machine learning problem; it is an infrastructural and institutional problem that requires robustness, fairness, sustainability, and accountability to be designed together. The next generation of private inference systems should therefore combine adaptive search with transparent governance, domain-specific constraints, and lifecycle evaluation. Without such integration, efficiency gains may come at the expense of equity, trust, or long-term maintainability.

## References

1. Zoph, B., & Le, Q. V. (2017). Neural architecture search with reinforcement learning. In *Proceedings of the International Conference on Learning Representations*.
2. Elsken, T., Metzen, J. H., & Hutter, F. (2019). Neural architecture search: A survey. *Journal of Machine Learning Research*, 20(55), 1-21.
3. Liu, H., Simonyan, K., & Yang, Y. (2019). DARTS: Differentiable architecture search. In *Proceedings of the International Conference on Learning Representations*.
4. Pham, H., Guan, M., Zoph, B., Le, Q., & Dean, J. (2018). Efficient neural architecture search via parameter sharing. In *Proceedings of the International Conference on Machine Learning* (pp. 4092-4101).
5. Real, E., Aggarwal, A., Huang, Y., & Le, Q. V. (2019). Regularized evolution for image classifier architecture search. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 33(1), 4780-4789.
6. Tan, M., & Le, Q. V. (2019). EfficientNet: Rethinking model scaling for convolutional neural networks. In *Proceedings of the International Conference on Machine Learning* (pp. 6105-6114).
7. He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (pp. 770-778).
8. Sandler, M., Howard, A., Zhu, M., Zhmoginov, A., & Chen, L. C. (2018). MobileNetV2: Inverted residuals and linear bottlenecks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (pp. 4510-4520).
9. Mohassel, P., & Zhang, Y. (2017). SecureML: A system for scalable privacy-preserving machine learning. In *Proceedings of the IEEE Symposium on Security and Privacy* (pp. 19-38).
10. Mohassel, P., & Rindal, P. (2018). ABY3: A mixed protocol framework for machine learning. In *Proceedings of the ACM SIGSAC Conference on Computer and Communications Security* (pp. 35-52).
11. Juvekar, C., Vaikuntanathan, V., & Chandrakasan, A. (2018). GAZELLE: A low latency framework for secure neural network inference. In *Proceedings of the USENIX Security Symposium* (pp. 1651-1669).

12. Liu, J., Juuti, M., Lu, Y., & Asokan, N. (2017). Oblivious neural network predictions via MiniONN transformations. In Proceedings of the ACM SIGSAC Conference on Computer and Communications Security (pp. 619-631).
13. Gilad-Bachrach, R., Dowlin, N., Laine, K., Lauter, K., Naehrig, M., & Wernsing, J. (2016). CryptoNets: Applying neural networks to encrypted data with high throughput and accuracy. In Proceedings of the International Conference on Machine Learning (pp. 201-210).
14. Dathathri, R., Saarikivi, O., Chen, H., Laine, K., Lauter, K., Maleki, S., Michel, M., & Wahby, R. S. (2019). CHET: An optimizing compiler for fully-homomorphic neural-network inferencing. In Proceedings of the ACM SIGPLAN Conference on Programming Language Design and Implementation (pp. 142-156).
15. Knott, B., Venkataraman, S., Hannun, A., Sengupta, S., & van der Maaten, L. (2021). CrypTen: Secure multi-party computation meets machine learning. In Advances in Neural Information Processing Systems, 34, 4961-4973.
16. Kairouz, P., McMahan, H. B., Avent, B., Bellet, A., Bennis, M., Bhagoji, A. N., et al. (2021). Advances and open problems in federated learning. Foundations and Trends in Machine Learning, 14(1-2), 1-210.
17. Bonawitz, K., Eichner, H., Grieskamp, W., Huba, D., Ingerman, A., Ivanov, V., et al. (2019). Towards federated learning at scale: System design. In Proceedings of Machine Learning and Systems, 1, 374-388.
18. Peng, Hongwu, et al. "Rrnet: Towards relu-reduced neural network for two-party computation based private inference." arXiv preprint arXiv:2302.02292 (2023).
19. Papernot, N., McDaniel, P., Sinha, A., & Wellman, M. P. (2018). SoK: Security and privacy in machine learning. In Proceedings of the IEEE European Symposium on Security and Privacy (pp. 399-414).
20. Mitchell, M., Wu, S., Zaldivar, A., Barnes, P., Vasserman, L., Hutchinson, B., et al. (2019). Model cards for model reporting. In Proceedings of the Conference on Fairness, Accountability, and Transparency (pp. 220-229).
21. Gebru, T., Morgenstern, J., Vecchione, B., Vaughan, J. W., Wallach, H., Daumé III, H., & Crawford, K. (2021). Datasheets for datasets. Communications of the ACM, 64(12), 86-92.
22. Mittelstadt, B. (2019). Principles alone cannot guarantee ethical AI. Nature Machine Intelligence, 1(11), 501-507.