Federated Prompt Tuning with Selective Prompt Synchronization for Privacy-Preserving NLP

Adrien T. Waber
Department of Electrical Engineering and Computer Science, University of Kansas, Lawrence, KS, USA.
adrianmail@ku.edu

Nieolas Kaennedy
Department of Computer Science and Engineering, University of Nevada, Reno, Reno, NV, USA.
nicolas.kennedy567@unr.edu

Larry Simmons
Department of Electrical Engineering and Computer Science, University of Missouri, Columbia, MO, USA.
larry.simmons@missouri.edu

Jeffrey L. Mendoza
Department of Computer Science, George Mason University, Fairfax, VA, USA.
jeffrey.work@gmu.edu

Abstract

The rapid advancement of large language models has transformed natural language processing but simultaneously intensified concerns around data privacy and sovereignty. Federated learning provides a compelling framework for collaborative model training without exposing raw user data, yet deploying these massive models across decentralized clients remains impractical due to enormous communication costs. Prompt tuning, which freezes the pre-trained model and learns only a small set of continuous embeddings, has emerged as a parameter-efficient alternative with strong downstream performance. When prompt tuning is integrated into federated settings, however, all prompt parameters are typically synchronized across clients, which still incurs nontrivial communication overhead and creates new privacy vulnerabilities through the shared representation space. This paper proposes a federated prompt tuning architecture augmented with selective prompt synchronization, in which only a carefully chosen subset of prompt tokens is exchanged between local clients and the aggregation server. Drawing on selective prompt insertion concepts, the system learns to identify and transmit the most informative prompt dimensions while suppressing redundant or privacy-sensitive information. We present a comprehensive system-level analysis that examines the structural trade-offs among communication efficiency, model accuracy, privacy protection, fairness, and governance. The discussion extends to integration with differential privacy, robustness against adversarial clients, sustainability, and policy implications in cross-silo and cross-device deployments. This work offers a blueprint for building secure, efficient, and responsible federated natural language processing infrastructures.

Keywords

Federated learning, prompt tuning, privacy, selective synchronization, NLP, large language models.

1. Introduction

The proliferation of large language models has fundamentally reshaped the landscape of natural language processing, enabling unprecedented performance on tasks ranging from text classification to complex reasoning. These models, however, are typically trained on vast corpora aggregated in centralized data centers, a paradigm that is increasingly at odds with growing demands for data privacy, user autonomy, and compliance with regulations such as the General Data Protection Regulation and the Health Insurance Portability and Accountability Act. The tension between model capability and data locality has spurred intense interest in federated learning, a distributed machine learning framework in which multiple participants collaboratively train a shared model while keeping their raw data on local devices [1,6]. Despite its promise, federated learning faces severe obstacles when applied to large language models, as the communication overhead of exchanging full model parameters or even sizeable gradient updates can be prohibitive for cross-device settings with limited bandwidth and energy budgets [2,7].

Parameter-efficient fine-tuning methods have been advanced as a practical remedy. Among them, prompt tuning stands out as a minimalist approach that prepends a set of trainable continuous vectors, known as soft prompts, to the input of a frozen pre-trained language model and optimizes only those vectors [3]. Because the number of learnable parameters is orders of magnitude smaller than the model itself, prompt tuning is uniquely suited for federated environments, where communication efficiency is paramount. Prior work has demonstrated the feasibility of federated prompt tuning, reporting competitive accuracy while drastically reducing per-round communication compared with full model fine-tuning [8,9]. Nevertheless, existing federated prompt tuning schemes still broadcast the entire set of prompt embeddings in each communication round, treating every prompt token as equally worthy of synchronization. This all-or-nothing exchange neglects the inherent redundancy present in high-dimensional prompt spaces and fails to capitalize on opportunities to further compress the transmitted information or to enhance privacy by limiting the exposure of prompt tokens that may inadvertently encode local data characteristics.

Recent research on centralized prompt tuning has shown that not all prompt positions contribute equally to task performance, and that learning to selectively insert prompts can yield notable improvements in both efficiency and accuracy [15]. This insight motivates the central contribution of the present paper: a federated prompt tuning framework that incorporates selective prompt synchronization, wherein only a subset of prompt token dimensions is shared and aggregated across the federation. By learning a local selection policy on each client and coordinating aggregation on a central server, the system can achieve substantial reductions in communication load, provide a natural veil for sensitive information by omitting certain prompt components, and open new design spaces for fairness and governance. The paper situates this proposal within a broader systems perspective, exploring architectural design choices, trade-offs among competing objectives, integration with differential privacy, robustness to heterogeneous and adversarial clients, and policy implications. Rather than presenting a narrow algorithmic contribution, we offer a deep analytical examination of how selective synchronization reshapes the socio-technical infrastructure of privacy-preserving natural language processing.

2. Background and Related Work

Federated learning was introduced as a decentralization paradigm that trains a global model by iteratively aggregating local updates computed on edge devices, thereby avoiding the transfer of raw data [1]. Extensive studies have since characterized its behavior under statistical heterogeneity, system-level constraints, and security threats [6,7]. When applied to large language models, the conventional approach of synchronizing full model weights quickly becomes infeasible due to the billions of parameters involved [2]. The community has therefore explored lightweight alternatives such as adapter-based fine-tuning and prompt-based methods that keep the backbone model frozen. Adapters insert small trainable modules into each transformer layer, achieving strong performance with a modest number of extra parameters [5]. Prompt tuning takes an even more parsimonious route by learning a small set of soft embeddings that are concatenated to the input; under appropriate scale, it rivals full model fine-tuning on many benchmarks [3]. Prefix-tuning, a closely related technique, optimizes continuous vectors that are prepended to the key-value caches of every transformer layer [4]. These parameter-efficient strategies form the backbone of the architecture proposed here.

The marriage of federated learning and prompt tuning has been explored in several recent studies. Researchers have demonstrated that prompt tuning can be performed directly in a federated fashion by averaging locally trained prompt embeddings on a central server, yielding models that respect data locality while preserving utility [8]. Black-box approaches go a step further by treating the language model as a remote API and optimizing prompts through derivative-free methods, thereby insulating clients even more from the model internals [9]. Challenges related to data heterogeneity have been investigated, with methods proposed to align prompt subspaces or to use personalization layers that adapt global prompts to local distributions [10]. At the optimization level, adaptive federated algorithms such as FedOpt have been applied to improve convergence speed and stability when aggregating prompts under non-identical client distributions [11]. Meanwhile, the privacy dimension has been partially addressed by incorporating differential privacy guarantees into the aggregation step, ensuring that the shared prompt vectors do not leak identifiable information about individual training examples [12,13].

Evaluation of federated systems for natural language processing has traditionally focused on accuracy and communication efficiency, but fairness, robustness, and privacy are gaining recognition as equally important dimensions [14]. Research has underscored that the distribution of prompt contributions across clients can mirror underlying demographic or behavioral biases, potentially leading to models that perform unevenly across different user groups. In parallel, the study of selective prompt mechanisms in centralized settings has revealed that learning to dynamically insert or mask prompts can improve both performance and interpretability [15]. The notion that prompt selection can act as a form of information bottleneck motivates our extension to the federated context, where selective synchronization transforms the communication and privacy profile of the system. This conceptual lineage, from parameter-efficient tuning to federated aggregation and selective control, sets the stage for the architectural and systemic analysis that follows.

3. System Architecture and Design

The proposed federated prompt tuning system with selective prompt synchronization comprises three principal components: client-side prompt modules with a local selection mechanism, a central aggregation server, and an orchestration layer that manages the communication protocol and versioning of the global prompt template. On each client, a

frozen pre-trained language model is paired with a set of trainable prompt embeddings, denoted as a matrix where rows correspond to prompt token positions and columns to embedding dimensions. In conventional federated prompt tuning, the entire prompt matrix is sent to the server after local training and the server computes an aggregated matrix, typically by averaging the updates from participating clients. Our architecture departs from this full synchronization by equipping each client with a lightweight selector network that outputs a binary mask over prompt token positions or embedding dimensions, indicating which elements should be uploaded. The selector can be implemented as a small multi-layer perceptron that takes as input features derived from the prompt gradients, the prompt values themselves, or a combination of task-specific signals such as the local validation loss. By only transmitting the selected subset of prompt components, the client significantly reduces its upload payload and, crucially, withholds information that the selector deems redundant or potentially privacy-leaking.

The server maintains a global version of the full prompt matrix and a corresponding aggregation buffer for the selected updates. Each round, the server receives masked prompt updates from a cohort of clients, reconstructs dense updates through zero-filling or default values for unselected dimensions, and applies a robust aggregation rule. Because the selection masks differ across clients, the server must carefully handle the varying levels of support for each prompt component. One approach is to compute weighted averages where each component is updated only if a sufficiently large number of clients designated it for synchronization; components that rarely receive updates can be gradually decayed or imputed from the global template. This design introduces a governance layer over which prompt regions are deemed globally relevant, effectively allowing the federation to collectively learn a sparse shared prompt representation. The orchestration layer is responsible for coordinating rounds, addressing client stragglers, and storing versioned snapshots of the global template to enable rollback and auditability.

The selective synchronization mechanism draws conceptual inspiration from the SPT framework, which learns to selectively insert prompts at inference time to improve task performance [15]. In our federated instantiation, the insertion logic is reinterpreted as a synchronization logic: rather than deciding where to insert prompts, the system decides which parts of the prompt space are worth communicating. The client-side selector is trained jointly with the local prompt embeddings using a multi-objective loss that balances task accuracy, communication cost, and a sparsity-inducing regularizer. The sparsity penalty can be designed as the expected fraction of transmitted dimensions, incentivizing the selector to be parsimonious. During local training, the selector observes the full prompt gradient but the mask is applied before transmission, creating an information bottleneck that forces the model to rely only on the most salient prompt components. This bottleneck aligns with principles of information theory and has the dual effect of compressing the communication footprint and reducing the fidelity with which an adversary could reconstruct local data from the shared prompt vectors.

4. Selective Prompt Synchronization Mechanisms

Designing an effective selection policy lies at the heart of the framework and encompasses several dimensions: the granularity of selection, the learning algorithm for the selector, and the interplay between individual client decisions and the global aggregation process. Granularity can range from selecting entire prompt tokens at the sequence level, to choosing individual embedding dimensions of each token, to a hierarchical approach that first selects

tokens and then dimensions within them. Finer granularity offers greater flexibility and potentially larger compression ratios but complicates the coordination among clients and may lead to fragmented global prompts that are difficult to reconstruct. Coarser selection, on the other hand, provides more stable aggregation and interpretability, as whole token positions that are frequently omitted likely correspond to less informative prompt slots. The trade-offs between these granularity levels must be assessed in terms of communication savings, convergence characteristics, and privacy leakage.

The local selector can be trained via a variety of strategies. One straightforward method employs importance scores derived from the magnitude of prompt gradients after a few local training steps; components with the largest gradients are deemed most impactful and are selected for upload. While simple and computationally lightweight, gradient-based selection may be myopic and does not directly account for global utility. A more sophisticated approach involves training the selector through reinforcement learning or a differentiable gating mechanism, where the client receives a reward based on the eventual global model performance on a held-out validation set. This method explicitly aligns local selection decisions with the global objective but introduces additional communication overhead for distributing reward signals. An intermediate strategy uses a learned meta-controller that predicts, given a client's data distribution and local prompt state, a mask that would minimize the expected global loss. The meta-controller can be periodically updated by the server using aggregated anonymized statistics, striking a balance between autonomy and coordinated efficiency.

At the server side, the aggregation of selectively synchronized prompts demands careful engineering to avoid biasing the global template toward clients with larger selection masks. If the server simply averages over the transmitted components, clients that send more prompt dimensions would exert disproportionate influence. To mitigate this, the server can normalize each client's contribution by the number of selected components or employ a trimmed mean approach that discards extreme values. Furthermore, the server can maintain a running estimate of the global importance of each prompt component based on the frequency and consistency of selection across clients. Components that are rarely or inconsistently selected can be flagged for removal or down-weighting, leading to a naturally sparse global prompt template that can be distributed to all clients at reduced cost. This process fosters a collective refinement of the prompt space, where distributed clients collaboratively discover a compact set of universally useful prompt features while privately retaining local specialization.

Privacy protection under selective synchronization arises from two complementary mechanisms. First, by omitting a portion of the prompt vector, the client reduces the total amount of information about its local training data that leaves the device. Even if an adversary compromises the server or intercepts the communication channel, the leaked signal is inherently incomplete. Second, the selection process can be tuned to be privacy-aware by penalizing the transmission of prompt components that exhibit high mutual information with the input tokens or with sensitive attributes, as estimated by a small auxiliary probe. When combined with standard differential privacy mechanisms, such as adding calibrated Gaussian noise to the selected updates before upload, the system can provide formal privacy guarantees while benefiting from the additional deniability conferred by the selection mask. The integration of differential privacy with selective synchronization is particularly elegant because the privacy budget need only be allocated to the transmitted subset, potentially

allowing tighter privacy for the same utility or higher utility for a fixed budget compared with full synchronization [12,13].

5. System-Level Trade-offs and Governance

The introduction of selective prompt synchronization reshapes the classical federated learning trade-off triangle of communication, computation, and accuracy. Communication efficiency is directly improved by reducing the number of floating-point values transmitted per round; depending on the sparsity of the selection mask, upload sizes can be reduced by fifty percent or more without a proportional degradation in model quality. This benefit is especially critical in cross-device settings where clients connect via cellular networks or metered connections, and where data caps and battery constraints strongly incentivize lightweight protocols. The flip side is an increase in local computation due to the selector network and the additional forward passes needed to tune the selection policy. However, because the pre-trained model remains frozen and the selector can be kept small, the computational overhead is often a fraction of the cost of local prompt training and is easily amortized over multiple rounds.

From an accuracy perspective, selective synchronization introduces a tension between representational richness and sparsity. Trimming too many prompt dimensions can deprive the global model of the nuanced variations needed to adapt to diverse client distributions, potentially reducing performance on tail tasks or underrepresented groups. Empirical evidence in centralized prompt tuning suggests that prompts contain considerable redundancy, and that selective enforcement of sparsity can actually improve generalization by preventing overfitting to spurious correlations [15]. In the federated case, the same effect can manifest as improved robustness to non-identically distributed data, provided that the selection policy is not dominated by the majority clients. Achieving the right balance requires careful calibration of the sparsity regularizer and, ideally, the incorporation of fairness constraints that prevent the global template from becoming overly tuned to dominant subpopulations.

Sustainability is an increasingly important dimension in the design of large-scale machine learning systems. Federated prompt tuning is already more energy-efficient than full model fine-tuning because the base model is frozen and only small vectors are trained. Selective synchronization further reduces the carbon footprint by lowering the volume of network traffic and the associated energy consumption of data centers and network infrastructure. Over thousands of training rounds and millions of devices, these savings compound significantly. System designers can explicitly model the trade-off between compression ratio and resulting accuracy, enabling operators to choose a configuration that meets a target carbon budget while satisfying performance requirements. The ability to dial down the synchronization rate adaptively, for example by increasing sparsity during periods of high network congestion or renewable energy scarcity, positions the architecture as an environmentally responsive component of a sustainable AI ecosystem.

Governance of the shared prompt template raises novel questions about the control and ownership of the collaboratively learned representation. In a fully synchronized system, all clients contribute equally to every dimension of the global prompt, which can be viewed as a democratic commons. Selective synchronization breaks this symmetry by allowing some clients to influence certain prompt components more heavily than others, depending on their selection decisions. If the selection masks are determined entirely by the clients, the system risks creating a fragmented contribution landscape where powerful or numerous clients dominate the most important prompt dimensions. Conversely, if the server imposes a global selection policy, it accrues significant control over which kinds of information are preserved

in the shared template, potentially undermining the decentralized ethos of federated learning. Hybrid governance models, such as federated committees of clients that vote on server-side selection rules or decentralized verification of selection fairness through secure multi-party computation, offer promising paths toward accountable and transparent management of the prompt commons.

6. Deployment, Robustness, and Sustainability

Deploying a federated prompt tuning system with selective synchronization in real-world environments requires careful consideration of the heterogeneity of hardware, operating systems, and data distributions. In cross-silo settings such as hospital networks or financial consortia, participants typically possess ample computational resources and reliable connectivity, but they operate under stringent data governance regulations. Here, selective synchronization can serve as a technical enforcement of the data minimization principle, as only the most essential prompt components are shared across institutional boundaries. The ability to cryptographically sign and audit the selection masks adds a layer of compliance assurance, enabling participating organizations to demonstrate to regulators that they have limited the exposure of sensitive information. In cross-device settings, where millions of smartphones or IoT devices collectively train a model, the infrastructure must handle intermittent connectivity, device churn, and extreme hardware diversity. The lightweight nature of prompt tuning already makes it a good fit for such environments, and selective synchronization further reduces the upload payload, alleviating the bottleneck for devices on slow or costly connections.

Robustness to adversarial clients is a perennial concern in federated systems. A malicious client could attempt to poison the global prompt template by fabricating selection masks that amplify harmful or backdoor prompt dimensions while suppressing legitimate ones. Defenses can be structured at multiple layers. At the aggregation layer, the server can employ robust statistics such as median or coordinate-wise trimmed averages on the received prompt updates, which mitigate the influence of outlier contributions. At the selection layer, the server can cross-validate the consistency of a client's mask with the masks sent by other clients with similar data distributions, flagging statistically improbable selection patterns for further inspection. Additionally, the meta-controller trained to predict global utility can be hardened through adversarial training, where it learns to discount updates that deviate from expected behavior. The sparsity of the synchronization also naturally limits the attack surface, because a poisoned client can only inject malicious signal into the dimensions it chooses to send, which can be further constrained by requiring a minimum overlap between client masks and a trusted reference set.

Sustainability extends beyond energy consumption to the long-term maintainability and evolution of the system. As language models are updated or replaced, the prompt templates must be migrated. Selective synchronization can ease this transition by identifying a core set of prompt dimensions that are robustly useful across model versions, enabling transfer learning of the prompt space. The versioning infrastructure of the orchestration layer can maintain multiple global templates tailored to different model checkpoints or task families, allowing clients to gracefully upgrade without discarding all previously learned knowledge. From a life-cycle perspective, the ability to dynamically adjust sparsity based on operational metrics enables system administrators to balance performance, cost, and environmental impact continuously, avoiding the binary choice between a fully synchronized and a fully isolated system that would otherwise dominate federated prompt tuning deployments.

7. Fairness, Privacy, and Policy Implications

Selective prompt synchronization directly interacts with fairness because the selection process can amplify or mitigate the representation of different client subpopulations in the global model. If clients from a particular demographic group consistently select certain prompt dimensions that encode linguistic patterns specific to their community, and if those dimensions are regularly selected by many clients across the federation, the global template will naturally incorporate that knowledge. Conversely, if a minority group's important prompt features are rarely selected, either because of their small numbers or because their selection policy is insufficiently tuned, those features risk being diluted or lost in the aggregated template, leading to disparate model accuracy. Addressing this requires fairness-aware selection mechanisms that incorporate demographic parity or equalized odds constraints into the selector training objective, or that apply server-side re-weighting to ensure underrepresented groups' contributions receive adequate emphasis [20]. Techniques from fair federated learning can be adapted to operate on the prompt selection masks, penalizing the global template for performance disparities across pre-defined client cohorts.

Privacy protection in selective synchronization must be evaluated through both quantitative and qualitative lenses. Quantitative privacy is typically analyzed via differential privacy, where the addition of noise to the uploaded prompt components provides a rigorous bound on the risk of inferring the presence of any single training example [12]. Because selective synchronization transmits only a subset of dimensions, the effective sensitivity of the query can be smaller than for full synchronization, potentially requiring less noise to achieve the same privacy level and thus improving the utility-privacy trade-off. However, the selection mask itself may leak information, as the pattern of which dimensions are transmitted could reveal properties of the client's data distribution. To counter this, the mask can be generated using a differentially private mechanism, or the system can enforce that the mask is computed solely from metadata that is not sensitive, such as the task type or device capabilities. A holistic privacy assessment must also consider membership inference attacks, model inversion attempts, and linkage attacks that combine prompt updates with auxiliary information, all of which are tempered by the information bottleneck induced by sparsity.

Policy implications of selective prompt synchronization span data governance, regulatory compliance, and the broader societal contract around collaborative AI. The architecture offers a concrete technical pathway for implementing the principle of data minimization enshrined in regulations like the General Data Protection Regulation, as it limits the personal data implicitly encoded in the transmitted prompt vectors. When combined with cryptographic techniques such as secure aggregation, the system can guarantee that even the central server does not observe individual client updates, only the aggregated result of the selected components. This aligns with the vision of privacy-by-design in AI systems. At the same time, selective synchronization introduces new points of control that must be transparently governed. Stakeholders, including end users, client institutions, and regulatory bodies, require understandable explanations of why certain prompt dimensions are shared and others are not. Developing interpretable selection policies, supported by model cards for the prompt template that disclose selection statistics and fairness impact assessments, is essential for building trust. Policy frameworks may eventually mandate such transparency for federated AI systems deployed in sensitive domains like healthcare, finance, and education.

8. Conclusion

This paper has presented a comprehensive system-level exploration of federated prompt tuning augmented with selective prompt synchronization as a pathway toward privacy-preserving natural language processing. By allowing clients to learn and enforce a parsimonious selection of which prompt dimensions to share, the architecture reduces communication overhead, strengthens privacy protections, and creates levers for governing fairness and robustness. The analysis traversed architectural design, selection mechanisms, trade-offs among efficiency and accuracy, deployment considerations in heterogeneous environments, sustainability, and the intricate intersection of fairness and policy. Rather than advocating a single algorithm, we have articulated a design space in which selective synchronization serves as a foundational primitive that can be configured and governed to suit diverse sociotechnical contexts. The integration with differential privacy, the resilience against adversarial manipulation, and the compatibility with emerging fairness-aware federated optimization techniques highlight the versatility of the approach. As language models continue to expand in scale and societal influence, infrastructures that enable collaborative learning without compromising individual autonomy or group equity will become indispensable. Federated prompt tuning with selective synchronization represents a meaningful step toward that responsible AI future, inviting further interdisciplinary research at the confluence of systems engineering, machine learning, and technology policy.

References

1. McMahan, B., Moore, E., Ramage, D., Hampson, S., & Aguera y Arcas, B. (2017). Communication-efficient learning of deep networks from decentralized data. Proceedings of the 20th International Conference on Artificial Intelligence and Statistics (AISTATS), 1273-1282.

2. 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.

3. Lester, B., Al-Rfou, R., & Constant, N. (2021). The power of scale for parameter-efficient prompt tuning. Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, 3045-3059.

4. Li, X. L., & Liang, P. (2021). Prefix-tuning: Optimizing continuous prompts for generation. Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics, 4582-4597.

5. Houlsby, N., Giurgiu, A., Jastrzebski, S., Morrone, B., de Laroussilhe, Q., Gesmundo, A., Attariyan, M., & Gelly, S. (2019). Parameter-efficient transfer learning for NLP. Proceedings of the 36th International Conference on Machine Learning, 2790-2799.

6. 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.

7. Bonawitz, K., Eichner, H., Grieskamp, W., Huba, D., Ingerman, A., Ivanov, V., ... & van Overveldt, T. (2019). Towards federated learning at scale: System design. Proceedings of the 2nd SysML Conference.

8. Yang, Y., Zhang, H., Wang, S., & Sun, M. (2023). Federated prompt tuning for language models. Findings of the Association for Computational Linguistics: ACL 2023, 2044-2056.

9. Sun, T., He, J., Qiu, X., & Huang, X. (2022). Black-box tuning for language-model-as-a-service. Proceedings of the 39th International Conference on Machine Learning, 20864-20885.

10. Li, T., Sahu, A. K., Zaheer, M., Sanjabi, M., Talwalkar, A., & Smith, V. (2023). Federated optimization in heterogeneous networks. Proceedings of Machine Learning and Systems, 5, 429-450.

11. Reddi, S. J., Charles, Z., Zaheer, M., Garrett, Z., Rush, K., Konecny, J., Kumar, S., & McMahan, H. B. (2021). Adaptive federated optimization. Proceedings of the 9th International Conference on Learning Representations.

12. Abadi, M., Chu, A., Goodfellow, I., McMahan, H. B., Mironov, I., Talwar, K., & Zhang, L. (2016). Deep learning with differential privacy. Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security, 308-318.

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

14. Wang, Z., Dai, Z., Poczos, B., & Carbonell, J. (2021). Characterizing and avoiding negative transfer in federated transfer learning with pre-trained models. Proceedings of the 2021 IEEE International Conference on Big Data, 1132-1141.

15. Zhu, W., & Tan, M. (2023, December). SPT: Learning to selectively insert prompts for better prompt tuning. In Proceedings of the 2023 conference on empirical methods in natural language processing (pp. 11862-11878).

16. Ouyang, L., Wu, J., Jiang, X., Almeida, D., Wainwright, C. L., Mishkin, P., ... & Lowe, R. (2022). Training language models to follow instructions with human feedback. Advances in Neural Information Processing Systems, 35, 27730-27744.

17. Wei, J., Wang, X., Schuurmans, D., Bosma, M., Ichter, B., Xia, F., Chi, E., Le, Q., & Zhou, D. (2022). Chain-of-thought prompting elicits reasoning in large language models. Advances in Neural Information Processing Systems, 35, 24824-24837.

18. Wang, X., Wei, J., Schuurmans, D., Le, Q., Chi, E., Narang, S., Chowdhery, A., & Zhou, D. (2022). Self-consistency improves chain of thought reasoning in language models. Proceedings of the 11th International Conference on Learning Representations.

19. Bender, E. M., Gebru, T., McMillan-Major, A., & Shmitchell, S. (2021). On the dangers of stochastic parrots: Can language models be too big? Proceedings of the 2021 ACM Conference on Fairness, Accountability, and Transparency, 610-623.

20. Mehrabi, N., Morstatter, F., Saxena, N., Lerman, K., & Galstyan, A. (2021). A survey on bias and fairness in machine learning. ACM Computing Surveys, 54(6), 1-35.

21. Hard, A., Rao, K., Mathews, R., Ramaswamy, S., Beaufays, F., Augenstein, S., ... & Bonawitz, K. (2018). Federated learning for mobile keyboard prediction. arXiv preprint arXiv:1811.03604.

22. Zhang, D. Y., Kou, Z., & Wang, D. (2021). Fairness in federated learning: A survey. ACM Computing Surveys, 54(11s), 1-38.