LLEOT: A Privacy-Enhancing Offsite Tuning Framework via Loss Landscape Elevation

Jin Zhong[1]†, Jinglin Liang[1]†, Tongtong Yang[1], Zijian Xie[1], Shuangping Huang[1]*, Hanlin Gu[2]

[1]School of EE., South China University of Technology, Guangzhou, China,
[2]Hong Kong University of Science and Technology, Hong Kong, China
{eezhongjin,eeljl,eeytt,eexzj}@mail.scut.edu.cn,
eehsp@scut.edu.cn, ghltsl123@gmail.com

Abstract

Adapting large language models (LLMs) to domain-specific tasks via fine-tuning is often infeasible: models are protected by intellectual property, while sensitive data cannot be shared due to privacy regulations. A promising paradigm, Offsite Tuning (OT), addresses this challenge by constructing an emulator of the original model. Data owners leverage the emulator to train an adapter on downstream data, which is then plugged back into the original model, enabling knowledge transfer without transmitting either the original model or the raw data. However, emulators constructed by existing OT-based methods often retain substantial inference capabilities, thereby exposing model capability privacy and posing risks of misuse. To address this, we propose Loss Landscape Elevation Offsite Tuning (LLEOT), a framework that secures data privacy as well as model parameter and capability privacy. At its core, Loss Landscape Elevation (LLE) enforces a fixed margin between the loss landscapes of the emulator and the original model. We theoretically demonstrate that LLE simultaneously (i) degrades emulator inference via perplexity amplification and (ii) preserves gradient alignment, ensuring consistent convergence for adapter training. Extensive experiments confirm that LLEOT achieves strong adaptation performance while effectively mitigating emulator misuse. Code is available at https://github.com/Z-eloto/LLEOT.

1 Introduction

In the field of natural language processing, fine-tuning pre-trained large language models (LLMs) (Wei et al., 2023; Muennighoff et al., 2023; Liu et al., 2022) on domain-specific data has become a widely adopted technique for adapting general-purpose models to specialized tasks. However, this approach faces significant practical constraints, particularly concerning intellectual property and data privacy (Gupta et al., 2022; Lyu et al., 2022). On the one hand, due to proprietary protections and licensing restrictions, many high-performing LLMs cannot be openly distributed to external data owners for fine-tuning (Li et al., 2023). On the other hand, even when model owners offer data submission interfaces for cloud-based training, stringent privacy regulations in fields such as healthcare (Nguyen et al., 2022) and finance (Kang et al., 2022; Oualid et al., 2025) often prohibit the upload of sensitive data to third-party services. This fundamental conflict, where neither the model nor the data can be shared, creates a significant barrier to effective model adaptation, leaving valuable private data untapped and limiting the applicability of closed-source models in critical domains.

A promising approach is to construct a privacy-preserving *emulator* of the original model to serve as a bridge for knowledge transfer. As shown in Figure 1(a), the model owner constructs an *emulator* and sends it to data owners, who then use this emulator to locally train an *adapter* that encodes the knowledge from their domain-specific data. This adapter is then returned to the model owner and plugged into the original model, enabling the model to acquire knowledge from the data without exposing the original model or the data itself. Xiao et al. (2023) first introduced this method, naming it Offsite Tuning, which constructs an emulator through model compression and knowledge distillation. CRaSh (Zhang et al., 2023) accelerates emulator construction by substituting knowledge distillation with layer importance-based selection, where high-importance layers replace low-importance ones. GradOT (Yao et al., 2025) constructs the emulator by selectively applying rank compression and channel pruning based on a gradient-preserving strategy. These methods employ techniques such as knowledge distil-

†Equal contribution.
*Corresponding author.

Findings of the Association for Computational Linguistics: ACL 2026, pages 37664–37683

lation (Mora et al., 2024; Huang et al., 2024) to align the emulator with the original model, ensuring that the adapter trained on the emulator remains applicable to the original model. However, such approaches result in an emulator that retains a significant portion of the original model's inference capabilities (Figure 1(c)), which inadequately protects the model's capability privacy. Consequently, *malicious data owners could potentially use this emulator to extract the model's knowledge or engage in unauthorized activities, thereby infringing upon the model owner's intellectual property rights*, as shown in Figure 1(b).

To address the above challenge, we propose Loss Landscape Elevation Offsite Tuning (LLEOT), a novel framework that extends privacy protection to model capability privacy. The core of LLEOT lies in Loss Landscape Elevation (LLE). Specifically, starting with a uniformly layer-dropped version of the original model, we adjust it to have a consistently higher loss than the original model by a fixed margin across all data points (§4.1). Though simple, our approach offers two key advantages as proven in Theorem 1. First, the elevated loss disables the emulator's inference ability, preserving the original model's capability privacy. Second, it maintains geometric consistency between the loss landscapes (see Figure 3), keeping the adapter's loss gradients coherent across models. This ensures adapters optimized on the emulator perform well when plugged into the original model. In theory, our method is applicable to various types of adapters. In this paper, we focus on soft prompts for their computational efficiency and ease of optimization. Additionally, to facilitate the construction of even smaller emulators, we introduce Collaborative Prompt Knowledge Distillation (CPKD) (§4.3). This optional technique, applied prior to LLE, further enhances the gradient consistency between the emulator and the original model. Our contributions can be summarized as follows:

- We identify the overlooked risk of model capability privacy in Offsite Tuning: existing emulators retain substantial inference power, enabling malicious data owners to extract proprietary knowledge or misuse the model.
- We propose Loss Landscape Elevation Offsite Tuning (LLEOT), which applies Loss Landscape Elevation (LLE) to disable emulator inference while preserving gradient alignment with the original model. We provide a theoretical guarantee (Theorem 1) that LLE both amplifies emulator perplexity and preserves convergence to the same optimal prompt.
- Comprehensive experiments show that LLEOT provides better privacy protection and higher model performance than existing methods.

2 Related Works

Large Language Models. Large language models (LLMs) (Kojima et al., 2022; Kung et al., 2023; Liang et al., 2025; Gong et al., 2026) acquire formidable natural language processing capabilities via pre-training. However, when applied to domain-specific problems, LLMs still require fine-tuning (Bai et al., 2024; Zhang et al., 2024) on relevant data to better adapt to the target tasks. Unfortunately, in many real-world scenarios, the model and the data are owned by different parties, and fine-tuning through mutual sharing is often infeasible for reasons including intellectual property protection. Black-box tuning (Yu et al., 2023; Zheng et al., 2024) approaches upload data to the model owner and adjust parameters based on output text, which helps protect the privacy of the LLMs but poses risks to user data. Alternative methods, such as federated learning (McMahan et al., 2017; Shi and Radu, 2021; Hong et al., 2026a) and split learning (Li et al., 2024), distribute the model to data owners to avoid data transmission, yet these approaches expose model privacy. Given the high value of large language models, such solutions are often unacceptable to model owners.

Privacy-preserving fine-tuning of large language models. To jointly protect model privacy and data privacy during fine-tuning, Offsite Tuning (OT) (Xiao et al., 2023) compresses the original model and applies knowledge distillation to obtain an emulator and an adapter; the data owner fine-tunes the adapter with the help of the emulator and then returns the tuned adapter to the model owner for integration into the original model. In contrast, CRaSh (Zhang et al., 2023) constructs the emulator by performing layer importance ranking on the original model and replacing less important layers with repeated high-importance layers. Most recently, GradOT (Yao et al., 2025) adopts a gradient-preserving approach, utilizing rank compression and channel pruning to construct the emulator. While these approaches safeguard model

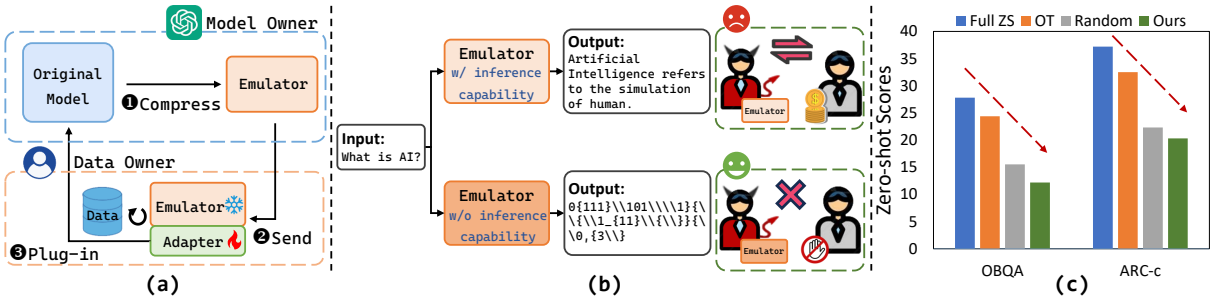

Figure 1: (a) Overview of privacy-preserving emulator methods. (b) Illustration of model capability leakage. (c) Comparison of inference ability on OBQA (Mihaylov et al., 2018) and ARC-c (Clark et al., 2018) dataset: Full ZS (original LLM Qwen2-1.5B (Yang et al., 2024)) and OT (75% compressed LLM) achieve high scores, indicating privacy leakage, while our method and Random (randomly initialize model) show lowscores, demonstrating effective protection of model capability privacy.

parameter privacy and data privacy, their reliance on knowledge distillation or selective structural pruning causes the emulator to inherit part of the original model's inference capability, leading to the leakage of the original model's capability privacy.

3 Problem Formulation

Setup. The model owner holds an original model $\mathcal{M}$, and the data owner possesses a downstream dataset $\mathcal{D}$. To leverage $\mathcal{D}$ to fine-tune $\mathcal{M}$ without directly sharing either, we aim to construct a privacy-preserving emulator $\mathcal{E}$ based on $\mathcal{M}$. This emulator is transmitted to the data owner to train an adapter, which is subsequently returned and plugged into the original model, serving as a bridge for knowledge transfer. The construction of the emulator must satisfy three objectives: (1) **Model Parameter Privacy**: The parameters of the emulator $\mathcal{E}$ should differ from those of the original model $\mathcal{M}$. This prevents unauthorized model replication. (2) **Model Capability Privacy**: The inference capability of $\mathcal{E}$ should approximate that of a randomly initialized model. This prevents malicious data owners from exploiting its capabilities to compromise the model owner's intellectual property. (3) **Training Effectiveness**: With the assistance of the emulator, the data owner trains an adapter $\mathcal{A}_{\mathcal{E}}$ using the dataset $\mathcal{D}$ (yielding $\mathcal{E} \circ \mathcal{A}_{\mathcal{E}}$). The *plug-in performance*, defined as the performance achieved by plugging the emulator-trained adapter into the original model (i.e., $\mathcal{M} \circ \mathcal{A}_{\mathcal{E}}$), should surpass the zero-shot performance of the original model and be comparable to that of the original model directly fine-tuned on $\mathcal{D}$ (i.e., $\mathcal{M} \circ \mathcal{A}_{\mathcal{M}}$).

Metrics. We introduce Capability Privacy Leakage (CPL), a new metric to quantify the capability leakage from the original model through the emulator. It is defined as the ratio of their zero-shot performance scores on a given task:

$$\text{CPL} = \frac{S_{zs}(\mathcal{E})}{S_{zs}(\mathcal{M})} \times 100\%, \tag{1}$$

where $S_{zs}(\cdot)$ is the zero-shot score function. Capability privacy protection is considered to be in effect only when the CPL value is below 100%. A lower CPL value signifies a lesser degree of leakage and thus more effective protection.

4 Methodology

In this section, we present the complete workflow of the LLEOT framework. As shown in Algorithm 1, the workflow comprises two phases: (1) **Emulator Construction** (§4.1): The model owner constructs an emulator via uniform LayerDrop and Loss Landscape Elevation (LLE), and transmits it to the data owner; and (2) **Adapter Training** (§4.2): The data owner utilizes the emulator to train an adapter on the downstream dataset, and returns the trained adapter to the model owner to be plugged into the original model. In this work, we specifically focus on soft prompts as the adapter implementation due to their computational efficiency. Additionally, we introduce Collaborative Prompt Knowledge Distillation (CPKD) (§4.3) as an optional step in the emulator construction process, designed to align the emulator with the original model under high pruning ratios.

4.1 Loss Landscape Elevation-based Emulator Construction

To construct an emulator that safeguards the parameter and capability privacy of the original model, while enabling the efficient training of soft prompts applicable to the original model, we propose a Loss

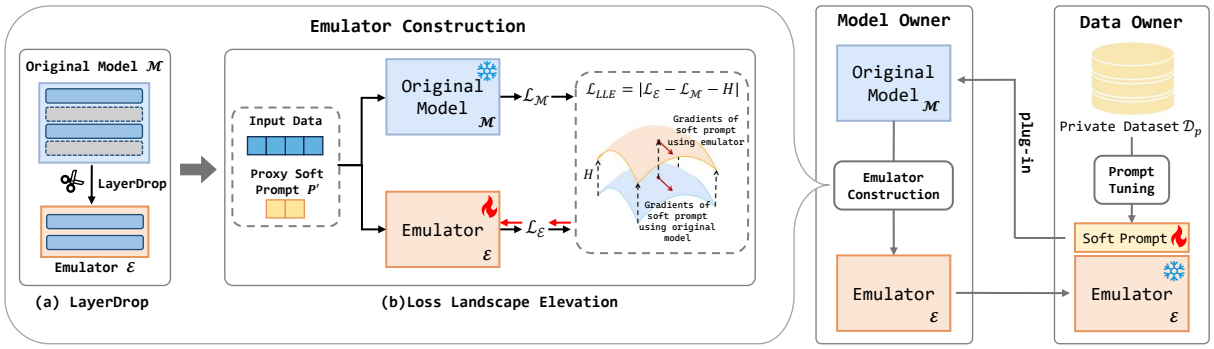

Figure 2: Overview of the LLEOT workflow. It comprises two phases: (1) Emulator Construction: The model owner constructs the emulator via (a) uniform LayerDrop and (b) Loss Landscape Elevation (LLE), and transmits it to the data owner; (2) Prompt Tuning: The data owner utilizes the emulator to train soft prompts on the downstream dataset, and returns the trained soft prompts to the model owner to be plugged into the original model.

Algorithm 1 LLEOT

Input: Original model $\mathcal{M}$, elevation dataset $\mathcal{D}_e$, private dataset $\mathcal{D}_p$, elevation margin H, pruning ratio β

Model owner: Construct an emulator
$\mathcal{E} \leftarrow \text{LayerDrop}(\mathcal{M}, \beta)$
for each batch $x \sim \mathcal{D}_e$ **do**
  Sample proxy soft prompt $P' \sim \mathcal{N}(\mu, \sigma^2)$
  Optimize $\mathcal{E}$ using Eq. (3)
end for
$\mathcal{E}^* = \mathcal{E}$
Sends $\mathcal{E}^*$ to Data owner
Data owner: Prompt Tuning for Downstream Tasks
Initialize soft prompt P
for each batch $x \sim \mathcal{D}_p$ **do**
  Compute downstream task loss: $\mathcal{L}_{ds}$
  Update prompt: $P \leftarrow P - \eta \nabla_P \mathcal{L}_{ds}$
end for
$P^* = P$
Sends P^* to Model owner
return Original model with optimized soft prompt $\{\mathcal{M}, P^*\}$

Landscape Elevation-based construction method. As illustrated in Figure 2, this method comprises two stages: LayerDrop and Loss Landscape Elevation (LLE). To construct a smaller emulator for efficient subsequent training, We compared various structural pruning techniques for LLMs (see Figure 6) and identified that uniform LayerDrop (Sajjad et al., 2023; Xiao et al., 2023) is the optimal strategy, which initializes the emulator by uniformly removing a subset of layers. Despite its simplicity, it yields better plug-in performance while incurring

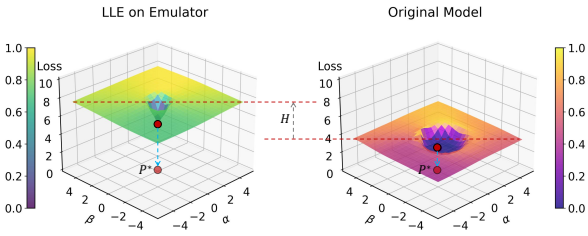

Figure 3: Visualization of the loss landscapes for the emulator (left) and the original model (right) in the same soft prompt parameter space ($\alpha\beta$-plane) after applying LLE. Note that LLE elevates the loss surface while preserving its shape.

no computational overhead.

However, the pruned emulator retains substantial inference capabilities, posing risks like proprietary knowledge extraction (Chua et al., 2023; Dong et al., 2023) or commercial repackaging (Jagarlamudi et al., 2024), thereby compromising the capability privacy of the original model. To mitigate this, we propose *Loss Landscape Elevation* (LLE) to uniformly elevate the emulator's loss landscape while aligning its geometry with that of the original model. In this way, we simultaneously (1) increase the emulator's perplexity to degrade its inference capability, and (2) preserve the alignment of gradients with respect to soft prompts between the emulator and the original model, ensuring effective soft prompt transfer. Specifically, for any soft prompt P and input text x, we enforce

$$\mathcal{L}_{\mathcal{E}}(P; x) = \mathcal{L}_{\mathcal{M}}(P; x) + H. \quad (2)$$

$\mathcal{L}_{\mathcal{E}}$ and $\mathcal{L}_{\mathcal{M}}$ denote the prompt tuning loss for the emulator and the original model, respectively. $H \geq 0$ is a hyperparameter for the fixed loss margin.

To realize this, we randomly sample points on the loss landscape, where each sample point is con-

structed by prepending a proxy soft prompt P' to the input text x, and enforce the constraint of Eq. (2) on the emulator across these samples. Formally, the optimization objective of LLE is formulated as:

$$\mathcal{E}^* = \arg\min_{\mathcal{E}} \mathbb{E}_{x\sim\mathcal{D}_e, P'\sim\mathcal{N}(\mu,\sigma^2)} \left|\mathcal{L}_{\mathcal{E}}(P';x) - \mathcal{L}_{\mathcal{M}}(P';x) - H\right|, \tag{3}$$

where $\mathcal{D}_e$ denotes the elevation dataset, P' is the proxy soft prompt, $\mathcal{N}(\mu,\sigma^2)$ is a normal distribution with mean μ (e.g., 0) and standard deviation σ (e.g., 20), determined experimentally.

We theoretically prove that LLE amplifies perplexity while preserving gradient alignment. A detailed proof is provided in Appendix D.

Theorem 1 (Effect of LLE on Emulator)**.** *For the emulator $\mathcal{E}$ constructed with Loss Landscape Elevation (LLE), we have*

$$\begin{aligned} \mathrm{PPL}_{\mathcal{E}} &= e^{H}\cdot \mathrm{PPL}_{\mathcal{M}} \quad \textit{and} \\ \nabla_P \mathcal{L}_{\mathcal{E}}(P;x) &= \nabla_P \mathcal{L}_{\mathcal{M}}(P;x), \end{aligned} \tag{4}$$

where $\mathrm{PPL}_{\mathcal{E}}$ and $\mathrm{PPL}_{\mathcal{M}}$ denote the perplexities of the emulator and the original model, respectively.

Proof. We first show that LLE effectively degrades the emulator's inference ability. We expand the prompt tuning loss into the following expression:

$$\begin{aligned} \mathcal{L}(P;x) &= -\frac{1}{n}\sum_{i=1}^{n}\log(p(x_i|P,x_{1:i-1})) \\ &= -\frac{1}{n} log(\hat{p}(x|P)), \\ \hat{p}(x|P) &= \prod_{i=1}^{n} p(x_i|P,x_{1:i-1}), \end{aligned} \tag{5}$$

Here, n denotes the number of tokens to predict in x, $p(x_i|P,x_{1:i-1})$ denotes the probability of the model correctly predicting the i-th token given the soft prompt P and the preceding $i-1$ tokens, and $\hat{p}(x|P)$ represents the joint probability of the model predicting the entire sequence x given the prompt P.

Based on Eq. (5), Eq. (2) can be transformed into:

$$\hat{p}_{\mathcal{E}}(x|P) = \mathrm{e}^{-nH}\hat{p}_{\mathcal{M}}(x|P). \tag{6}$$

To analyze the impact of this loss difference on model performance, we consider perplexity (PPL), a standard metric for evaluating language models, which is defined as:

$$\mathrm{PPL} = \exp\left(-\frac{1}{n}\log(\hat{p})\right) = \hat{p}^{-1/n}. \tag{7}$$

Substituting Eq. (7) into Eq. (6), we derive the following relationship between the perplexities of the emulator and the original model:

$$\mathrm{PPL}_{\mathcal{E}} = \mathrm{e}^{H}\cdot \mathrm{PPL}_{\mathcal{M}}. \tag{8}$$

It shows that the emulator's PPL is exponentially greater than the original model's by a factor of e^H. As a result, the emulator possesses significantly degraded inference capabilities, which serves to protect the original model's capability privacy.

In addition, we demonstrate that LLE maintains gradient consistency between the emulator and the original model. Specifically, the emulator's gradient with respect to the soft prompt can be expressed as:

$$\begin{aligned} \nabla_P \mathcal{L}_{\mathcal{E}}(P;x) &= \nabla_P\left(\mathcal{L}_{\mathcal{M}}(P;x) + H\right) \\ &= \nabla_P \mathcal{L}_{\mathcal{M}}(P;x), \end{aligned} \tag{9}$$

Consequently, during prompt tuning, the soft prompts optimized on both the emulator and the original model will converge to the same optimal point $P^\star$, ensuring that soft prompts trained on the emulator are applicable to the original model. □

4.2 Prompt Tuning

Upon completion of the emulator, the model owner sends it to the data owner. The data owner optimize a soft prompt P on their private dataset $\mathcal{D}_p$ by minimizing the downstream task loss $\mathcal{L}_{ds}$:

$$P^* = \arg\min_{P} \mathbb{E}_{x\sim\mathcal{D}_p}[\mathcal{L}_{ds}(\mathcal{E};P,x)] \tag{10}$$

The resulting prompt, P^*, is then sent back to the model owner, where the prompt is plugged into the original model to adapt it for the downstream task.

Furthermore, we demonstrate that LLEOT is orthogonal to data privacy strategies, indicating that the trained soft prompts can be sanitized before being returned. This safeguards data privacy against various inference attacks from the model owner, such as membership inference (Duan et al., 2024), all without significantly compromising the prompt's utility. Experimental results are provided in Appendix B.3.

4.3 Cooperative Prompt Knowledge Distillation

Emulators initialized via uniform LayerDrop maintain high consistency with the original model at lower pruning ratios (e.g., 0.25), yielding satisfactory transferability of the trained soft prompts.

Method	OBQA		SIQA		ARC-c		WebQs	
	Acc(↑)	CPL(↓)	Acc(↑)	CPL(↓)	Acc(↑)	CPL(↓)	Acc(↑)	CPL(↓)
Qwen2-1.5b								
Full ZS	27.80	100.00	46.47	100.00	37.20	100.00	1.82	100.00
Full PT	35.80	100.00	54.52	100.00	41.98	100.00	30.73	100.00
Random	15.52	55.83	33.33	71.72	22.32	60.00	0.00	0.00
OT	31.27	92.55	48.82	95.37	38.48	87.93	23.42	232.42
CRaSh	30.13	56.12	48.98	77.53	37.46	74.54	21.48	75.82
GradOT	30.20	53.24	49.07	76.63	38.93	56.88	22.32	0.00
Ours	**34.00**	**43.88**	**52.10**	**72.80**	**40.36**	**54.60**	**23.72**	**0.00**
Qwen2-7b								
Full ZS	35.00	100.00	52.35	100.00	51.11	100.00	1.08	100.00
Full PT	41.40	100.00	58.39	100.00	57.17	100.00	43.65	100.00
Random	16.07	45.90	32.98	63.00	22.92	44.84	0.00	0.00
OT	38.20	76.00	53.84	85.35	54.49	79.46	32.45	350.93
CRaSh	39.33	64.00	54.65	74.29	53.75	48.74	33.67	45.37
GradOT	37.20	65.14	54.75	72.84	54.43	43.06	**37.64**	0.00
Ours	**41.40**	**36.57**	**55.15**	**68.42**	**55.57**	**40.56**	35.95	**0.00**
Gemma2-9b								
Full ZS	40.40	100.00	57.62	100.00	63.05	100.00	11.37	100.00
Full PT	55.00	100.00	63.25	100.00	69.02	100.00	47.83	100.00
Random	17.80	44.06	33.17	57.57	22.18	35.18	0.00	0.00
OT	48.40	81.19	59.01	87.23	63.12	68.07	41.33	30.69
CRaSh	43.80	75.25	59.97	83.04	63.65	63.20	39.15	4.75
GradOT	49.00	74.26	60.01	77.09	65.18	61.16	40.45	3.43
Ours	**54.80**	**32.67**	**63.25**	**60.31**	**68.86**	**33.42**	**41.93**	**0.00**

Table 1: Comparative experiment results. 'Acc' denotes accuracy (higher is better), and 'CPL' represents the model capability privacy measure (lower is better). **Bold** and underlined denote the best and second-best performance among OT, CRaSh, GradOT, and Ours. 'Ours' denotes LLEOT without CPKD.

However, this transferability deteriorates significantly at higher pruning ratios (e.g., 0.5). To further align the emulator with the original model, we propose Collaborative Prompt Knowledge Distillation (CPKD), a novel technique serving as an optional step in the emulator construction pipeline. In contrast to standard knowledge distillation, we add an objective to align the hidden states of both models when soft prompts are prepended to the input, which can be formulated as:

$$\mathcal{L}_{PPD} = \mathbb{E}_{x\sim\mathcal{X}_d, P'\sim\mathcal{N}(\mu,\sigma^2)} ||(\mathbf{H}^{(-1)}_{\mathcal{E},L_p:}(P',x), \mathbf{H}^{(-1)}_{\mathcal{M},L_p:}(P',x))||^2, \quad (11)$$

where L_p is the length of the proxy soft prompt, and the notation $\mathbf{H}^{(-1)}$ represents the hidden state extracted from the final transformer layer.

And the overall objective of CPKD can be expressed as:

$$\mathcal{E}^* = \arg\min_{\mathcal{E}} w_1\mathcal{L}_{LM} + w_2\mathcal{L}_{PPD} + w_3\mathcal{L}_{KD}, \quad (12)$$

where w_1, w_2 and w_3 are hyperparameters used to balance the contributions of each term.

The detailed distillation procedure is provided in Appendix A.2.

5 Experiments

5.1 Experimental Setup

Models and Datasets. We evaluate our method on six LLMs: Qwen2-1.5B and Qwen2-7B (Yang et al., 2024), Gemma2-2B and Gemma2-9B (Team et al., 2024), Llama3.2-3B and Llama3.1-8B (Grattafiori et al., 2024). Experiments are

conducted on four question-answering benchmark datasets: OpenBookQA (Mihaylov et al., 2018), SocialIQA (Sap et al., 2019), ARC-Challenge (Clark et al., 2018), and WebQuestions (Berant et al., 2013). More experimental details are provided in Appendix A.5.

Baseline Methods. We compare our approach with the following six methods: (1) **Full ZS**: Zero-shot performance of the original model. (2) **Full PT**: Prompt tuning on the original model using the downstream dataset. (3) **Random**: A model with the same architecture as the original model but with randomly initialized weights. (4) **Offsite Tuning (OT)** (Xiao et al., 2023): The pioneering work proposing the offsite fine-tuning approach based on emulator construction. It utilizes LayerDrop and knowledge distillation to build an emulator. (5) **CRaSh** (Zhang et al., 2023): An OT variant that constructs the emulator via layer-importance selection. It represents the prior state-of-the-art method among open-source OT approaches. (6) **GradOT** (Yao et al., 2025): Another OT variant that constructs the emulator via selective pruning guided by gradient importance scores. It represents the latest state-of-the-art method among OT approaches (currently closed-source).

Metrics. We evaluate our method based on two aspects: (1) the plug-in performance. Since all benchmarks are multiple-choice datasets, we report accuracy for this aspect (for Full ZS, Full PT and Random, we report the model's accuracy directly); and (2) the capability privacy protection of the emulator, which we assess using the Capability Privacy Leakage (CPL) metric, as defined in §3. We use `lm-eval-harness` [1] to evaluate our models for a fair comparison.

5.2 Main Results

We conduct a comprehensive comparison of our method against existing state-of-the-art (SOTA) methods (OT, CRaSh, and GradOT) in terms of the emulator's plug-in performance and capability privacy protection. To ensure a fair comparison, the emulator pruning ratio is set to 0.25 across all methods. We incorporate the original model's zero-shot performance (Full ZS), prompt tuning on the original model (Full PT), and a randomly initialized model (Random) as comparative baselines. Ideally, the emulator's plug-in performance should surpass Full ZS and approximate Full PT, while its capability privacy protection should be comparable to Random. From Table 1 and Table 5 in Appendix B.1, we derive the following insights: (1) **LLEOT demonstrates superior plug-in performance and capability privacy protection across almost all experimental settings.** Notably, under certain experimental configurations, our method achieves a CPL score even lower than that of a randomly initialized model. This strongly suggests that the emulator constructed by our method offers robust model capability privacy protection, while ensuring that the trained soft prompts are highly applicable to the original model. (2) **Emulators constructed via knowledge distillation retain strong inference capabilities, resulting in severe leakage of capability privacy.** The knowledge distillation-based method, OT, achieves higher plug-in accuracy than CRaSh and GradOT under certain experimental settings; however, it exhibits the poorest CPL scores. (3) **Selective pruning fails to completely impair the emulator's inference capabilities, similarly leading to capability privacy leakage.** The selective pruning-based methods, CRaSh and GradOT, outperform OT in terms of the CPL metric. However, compared to our LLEOT, they exhibit lower plug-in accuracy and inferior CPL scores across most experimental settings.

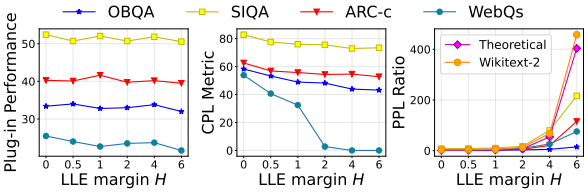

Figure 4: Variation in plug-in performance, CPL, and PPL Ratio ($\mathrm{PPL}_{\mathcal{E}}/\mathrm{PPL}_{\mathcal{M}}$) as the LLE margin H increases across different tasks on Qwen2-1.5B.

5.3 Ablation Study

In this section, we analyze the impact of the elevation margin, elevation strategies, pruning strategies, and pruning ratios on the emulator. Additional analyses and experimental results are provided in Appendix B.

Impact of LLE margin. Figure 4 reveal three key trends regarding the elevation margin H. First, the plug-in accuracy remains remarkably robust to increases in H. This suggests that loss landscape geometric alignment, rather than the absolute loss value, is the critical factor for ensuring effectiveness of the fine-tuned soft prompt when applied to the original model. Second, the CPL

[1] https://github.com/EleutherAI/lm-evaluation-harness

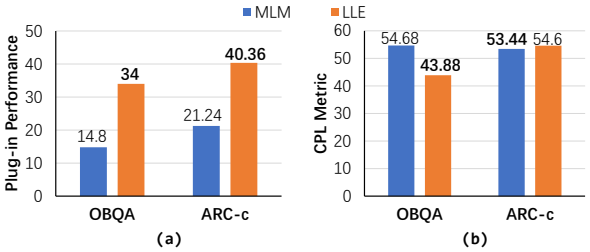

Figure 5: Plug-in performance and CPL metrics of emulators constructed via different elevation strategies on Qwen2-1.5B (pruning ratio = 0.25).

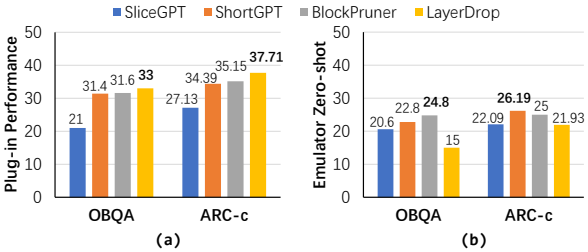

Figure 6: Plug-in and zero-shot performance of emulators initialized via different pruning strategies on Qwen2-1.5B (pruning ratio = 0.25).

Pruning Ratio	OBQA	SIQA	ARC-c	WebQs
0	34.40	52.30	41.98	32.09
0.25 w/o CPKD	34.00	52.10	40.36	23.72
0.25 w/ CPKD	34.80	50.41	40.19	25.74
0.5 w/o CPKD	23.00	45.09	32.59	9.35
0.5 w/ CPKD	34.20	50.04	40.44	24.15

Table 2: Plug-in performance of emulators initialized via different pruning ratios on Qwen2-1.5B.

metric decreases significantly for H between 0 and 2, after which it plateaus. This demonstrates that optimal capability privacy can be achieved without resorting to an overly large H. Third, the ratio of the emulator's perplexity to that of the original model increases exponentially with H. According to Theorem 1, this ratio is theoretically expected to be e^H. Experimental results indicate that on the Wikitext-2 dataset (Merity et al., 2016), the observed ratio aligns closely with the theoretical value. Conversely, on downstream task datasets such as SIQA, although the ratio exhibits an exponential upward trend, it deviates from the theoretical prediction. This discrepancy is likely attributed to the domain similarity between Wikitext-2 (Merity et al., 2016) and the elevation dataset, whereas domain shifts exist for the other datasets. Nevertheless, we argue that this deviation does not undermine the effectiveness of LLE. As demonstrated in comparative experiments, the performance of emulators constructed via LLE on downstream tasks approximates or even falls below that of a randomly initialized model, thereby effectively safeguarding the capability privacy of the original model.

Impact of Loss Landscape Elevation Strategies. To evaluate the impact of different elevation strategies on the emulator, we compare LLE with the strategy of directly maximizing the language modeling loss (MLM). As shown in Figure 5, compared to MLM, our uniform elevation strategy (LLE) exhibits superior plug-in performance and comparable CPL metrics. This is attributed to the fact that MLM elevates the emulator's loss landscape non-uniformly; and while it disrupts the emulator's inference capability, it fails to preserve gradient consistency with respect to soft prompts between the emulator and the original model. Additional results are provided in Appendix B.2.

Impact of Pruning Strategies. We applied various pruning strategies to the original model and comparatively evaluated the plug-in and zero-shot performance of the resulting emulators. To eliminate potential interference, LLE was not applied to the emulators in this experiment. As illustrated in Figure 6, compared to state-of-the-art LLM structural pruning methods, including SliceGPT (Ashkboos et al., 2024), ShortGPT (Men et al., 2025), and BlockPruner (Zhong et al., 2025), uniform LayerDrop exhibits better plug-in performance (Figure 6 (a)) and inferior zero-shot performance (Figure 6 (b)), while incurring no computational overhead. This is likely attributed to the fact that the pruning objectives of these baselines focus on preserving zero-shot performance, rather than maintaining the model's adaptability to soft prompts.

Impact of Pruning Ratios. Table 2 presents the plug-in performance of emulators with and without CPKD across varying pruning ratios. The results indicate that at lower pruning ratios (e.g., 0 and 0.25), applying LLE directly is sufficient to construct an emulator with satisfactory plug-in performance. Conversely, at higher pruning ratios (e.g., 0.5), we can achieve this by enabling CPKD.

6 Conclusion

In this work, we identify for the first time that existing OT methods carry the risk of model capability privacy leakage. To address this issue, we propose LLEOT, an innovative OT framework, whose core lies in the proposed LLE technique. We prove that this technique effectively disrupts the inference ca-

pability of emulators to prevent privacy leakage, while maintaining gradient consistency between the emulator and the original model. This ensures that adapters trained on the emulator remain applicable to the original model. Comprehensive experiments show that LLEOT achieves state-of-the-art performance in both protecting model privacy and model utility.

Limitations

Although our experimental results demonstrate the effectiveness of LLEOT in protecting model capability privacy and preserving adapter transferability, stricter theoretical guarantees are still lacking, including: (1) sufficient theoretical conditions for model capability privacy, for example, conditions under which increasing the elevation margin H guarantees negligible exploitable inference capability; (2) theoretical guarantees for adapter transferability, particularly whether the emulator and the original model share the same local minima; and (3) theoretical guarantees for the complete absence of functional islands in the emulator, namely local regions in which the emulator may still retain exploitable inference capability. In addition, due to computational resource constraints, we have not yet evaluated our method on larger-scale models, such as Qwen2-72B.

Ethics Statement

The research presented in this paper is fundamentally motivated by the ethical imperative to address significant privacy and security challenges in large model adaptation. Our work focuses on the Offsite Tuning paradigm, where a key ethical risk is the potential misuse of emulators that inadvertently leak the original model's inference capabilities. Our proposed method, LLEOT, is designed with a 'privacy-by-design' approach. The core Loss Landscape Elevation mechanism is intentionally engineered to degrade the emulator's inference abilities, thereby directly mitigating this risk of misuse. This work did not involve human participants or user studies. The methods and findings are intended solely for the research purpose of developing more secure, responsible, and trustworthy AI frameworks.

Acknowledgments

This work was supported by National Natural Science Foundation of China (No.62576139, 62176093), National Key Research and Development Program of China (No.2023YFC3502900).

References

Saleh Ashkboos, Maximilian L Croci, Marcelo Gennari do Nascimento, Torsten Hoefler, and James Hensman. 2024. Slicegpt: Compress large language models by deleting rows and columns. In *The Twelfth International Conference on Learning Representations*.

Jiamu Bai, Daoyuan Chen, Bingchen Qian, Liuyi Yao, and Yaliang Li. 2024. Federated fine-tuning of large language models under heterogeneous tasks and client resources. *Advances in Neural Information Processing Systems*, 37:14457–14483.

Jonathan Berant, Andrew Chou, Roy Frostig, and Percy Liang. 2013. Semantic parsing on freebase from question-answer pairs. In *Proceedings of the 2013 conference on empirical methods in natural language processing*, pages 1533–1544.

Terence Jie Chua, Wenhan Yu, Jun Zhao, and Kwok-Yan Lam. 2023. Fedpeat: Convergence of federated learning, parameter-efficient fine tuning, and emulator assisted tuning for artificial intelligence foundation models with mobile edge computing. *arXiv preprint arXiv:2310.17491*.

Peter Clark, Isaac Cowhey, Oren Etzioni, Tushar Khot, Ashish Sabharwal, Carissa Schoenick, and Oyvind Tafjord. 2018. Think you have solved question answering? try arc, the ai2 reasoning challenge. *arXiv preprint arXiv:1803.05457*.

Gheorghe Comanici, Eric Bieber, Mike Schaekermann, Ice Pasupat, Noveen Sachdeva, Inderjit Dhillon, Marcel Blistein, Ori Ram, Dan Zhang, Evan Rosen, and 1 others. 2025. Gemini 2.5: Pushing the frontier with advanced reasoning, multimodality, long context, and next generation agentic capabilities. *arXiv preprint arXiv:2507.06261*.

Chenhe Dong, Yuexiang Xie, Bolin Ding, Ying Shen, and Yaliang Li. 2023. Tunable soft prompts are messengers in federated learning. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 14665–14675.

Haonan Duan, Adam Dziedzic, Mohammad Yaghini, Nicolas Papernot, and Franziska Boenisch. 2024. On the privacy risk of in-context learning. *arXiv preprint arXiv:2411.10512*.

Shengjie Gong, Wenjie Peng, Hongyuan Chen, Gangyu Zhang, Yunqing Hu, Huiyuan Zhang, Shuangping Huang, and Tianshui Chen. 2026. Learning hierarchical and geometry-aware graph representations for text-to-cad. In *The Fourteenth International Conference on Learning Representations*.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten,

Alex Vaughan, and 1 others. 2024. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*.

Samyak Gupta, Yangsibo Huang, Zexuan Zhong, Tianyu Gao, Kai Li, and Danqi Chen. 2022. Recovering private text in federated learning of language models. *Advances in neural information processing systems*, 35:8130–8143.

Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. 2015. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*.

Mengze Hong, Yi Gu, Di Jiang, Hanlin Gu, Chen Jason Zhang, Lu Wang, and Zhiyang Su. 2026a. Federated heterogeneous language model optimization for hybrid automatic speech recognition. *Preprint*, arXiv:2603.04945.

Mengze Hong, Chen Jason Zhang, Zichang Guo, Hanlin Gu, Di Jiang, and Qing Li. 2026b. Orchestration-free customer service automation: A privacy-preserving and flowchart-guided framework. In *Proceedings of the ACM Web Conference 2026*, WWW '26, page 8138–8149, New York, NY, USA. Association for Computing Machinery.

Jiayi Huang, Yuanyuan Zhang, Renwan Bi, Jiayin Lin, and Jinbo Xiong. 2024. Knowledge distillation enables federated learning: A data-free federated aggregation scheme. In *2024 International Joint Conference on Neural Networks (IJCNN)*, pages 1–7. IEEE.

Gopi Krishna Jagarlamudi, Abbas Yazdinejad, Reza M Parizi, and Seyedamin Pouriyeh. 2024. Exploring privacy measurement in federated learning. *The Journal of Supercomputing*, 80(8):10511–10551.

Yan Kang, Hanlin Gu, Xingxing Tang, Yuanqin He, Yuzhu Zhang, Jinnan He, Yuxing Han, Lixin Fan, Kai Chen, and Qiang Yang. 2024. Optimizing privacy, utility, and efficiency in a constrained multi-objective federated learning framework. *ACM Transactions on Intelligent Systems and Technology*, 15(6):1–33.

Yan Kang, Yuanqin He, Jiahuan Luo, Tao Fan, Yang Liu, and Qiang Yang. 2022. Privacy-preserving federated adversarial domain adaptation over feature groups for interpretability. *IEEE Transactions on Big Data*, 10(6):879–890.

Takeshi Kojima, Shixiang Shane Gu, Machel Reid, Yutaka Matsuo, and Yusuke Iwasawa. 2022. Large language models are zero-shot reasoners. *Advances in neural information processing systems*, 35:22199–22213.

Zhiqiang Kou, Jing Wang, Jiawei Tang, Yuheng Jia, Boyu Shi, and Xin Geng. 2024. Exploiting multi-label correlation in label distribution learning. In *Proceedings of the Thirty-Third International Joint Conference on Artificial Intelligence*, pages 4326–4334.

Tiffany H Kung, Morgan Cheatham, Arielle Medenilla, Czarina Sillos, Lorie De Leon, Camille Elepaño, Maria Madriaga, Rimel Aggabao, Giezel Diaz-Candido, James Maningo, and 1 others. 2023. Performance of chatgpt on usmle: potential for ai-assisted medical education using large language models. *PLoS digital health*, 2(2):e0000198.

Bo Li, Peng Qi, Bo Liu, Shuai Di, Jingen Liu, Jiquan Pei, Jinfeng Yi, and Bowen Zhou. 2023. Trustworthy ai: From principles to practices. *ACM Computing Surveys*, 55(9):1–46.

Zhuohang Li, Chao Yan, Xinmeng Zhang, Gharib Gharibi, Zhijun Yin, Xiaoqian Jiang, and Bradley A Malin. 2024. Split learning for distributed collaborative training of deep learning models in health informatics. In *AMIA Annual Symposium Proceedings*, volume 2023, page 1047.

Jinglin Liang, Jin Zhong, Shuangping Huang, Yunqing Hu, Huiyuan Zhang, Huifang Li, Lixin Fan, and Hanlin Gu. 2025. Order-level attention similarity across language models: A latent commonality. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*.

Haokun Liu, Derek Tam, Mohammed Muqeeth, Jay Mohta, Tenghao Huang, Mohit Bansal, and Colin A Raffel. 2022. Few-shot parameter-efficient fine-tuning is better and cheaper than in-context learning. *Advances in Neural Information Processing Systems*, 35:1950–1965.

Lingjuan Lyu, Han Yu, Xingjun Ma, Chen Chen, Lichao Sun, Jun Zhao, Qiang Yang, and Philip S Yu. 2022. Privacy and robustness in federated learning: Attacks and defenses. *IEEE transactions on neural networks and learning systems*, 35(7):8726–8746.

Brendan McMahan, Eider Moore, Daniel Ramage, Seth Hampson, and Blaise Aguera y Arcas. 2017. Communication-efficient learning of deep networks from decentralized data. In *Artificial intelligence and statistics*, pages 1273–1282. PMLR.

Xin Men, Mingyu Xu, Qingyu Zhang, Qianhao Yuan, Bingning Wang, Hongyu Lin, Yaojie Lu, Xianpei Han, and Weipeng Chen. 2025. Shortgpt: Layers in large language models are more redundant than you expect. In *Findings of the Association for Computational Linguistics: ACL 2025*, pages 20192–20204.

Stephen Merity, Caiming Xiong, James Bradbury, and Richard Socher. 2016. Pointer sentinel mixture models. *Preprint*, arXiv:1609.07843.

Todor Mihaylov, Peter Clark, Tushar Khot, and Ashish Sabharwal. 2018. Can a suit of armor conduct electricity? a new dataset for open book question answering. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 2381–2391.

Alessio Mora, Irene Tenison, Paolo Bellavista, and Irina Rish. 2024. Knowledge distillation in federated learning: a practical guide. In *Proceedings of the Thirty-Third International Joint Conference on Artificial Intelligence*, pages 8188–8196.

Niklas Muennighoff, Thomas Wang, Lintang Sutawika, Adam Roberts, Stella Biderman, Teven Le Scao, M Saiful Bari, Sheng Shen, Zheng-Xin Yong, Hailey Schoelkopf, and 1 others. 2023. Crosslingual generalization through multitask finetuning. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 15991–16111.

Dinh C Nguyen, Quoc-Viet Pham, Pubudu N Pathirana, Ming Ding, Aruna Seneviratne, Zihuai Lin, Octavia Dobre, and Won-Joo Hwang. 2022. Federated learning for smart healthcare: A survey. *ACM Computing Surveys (Csur)*, 55(3):1–37.

Adil Oualid, Youssef Qasmaoui, Youssef Balouki, and Lahcen Moumoun. 2025. Federated learning and open banking for inclusive credit scoring in morocco: A systematic review. In *International Conference on intelligent systems and digital applications*, pages 242–256. Springer.

Hassan Sajjad, Fahim Dalvi, Nadir Durrani, and Preslav Nakov. 2023. On the effect of dropping layers of pre-trained transformer models. *Computer Speech & Language*, 77:101429.

Maarten Sap, Hannah Rashkin, Derek Chen, Ronan LeBras, and Yejin Choi. 2019. Socialiqa: Commonsense reasoning about social interactions. *arXiv preprint arXiv:1904.09728*.

Hongrui Shi and Valentin Radu. 2021. Towards federated learning with attention transfer to mitigate system and data heterogeneity of clients. In *Proceedings of the 4th international workshop on edge systems, analytics and networking*, pages 61–66.

Gemma Team, Morgane Riviere, Shreya Pathak, Pier Giuseppe Sessa, Cassidy Hardin, Surya Bhupatiraju, Léonard Hussenot, Thomas Mesnard, Bobak Shahriari, Alexandre Ramé, and 1 others. 2024. Gemma 2: Improving open language models at a practical size. *arXiv preprint arXiv:2408.00118*.

Jason Wei, Maarten Bosma, Vincent Zhao, Kelvin Guu, Adams Wei Yu, Brian Lester, Nan Du, Andrew M Dai, and Quoc V Le. 2023. Finetuned language models are zero-shot learners. In *International Conference on Learning Representations*.

Guangxuan Xiao, Ji Lin, and Song Han. 2023. Offsite-tuning: Transfer learning without full model. *arXiv preprint arXiv:2302.04870*.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, and 1 others. 2025. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*.

An Yang, Baosong Yang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Zhou, Chengpeng Li, Chengyuan Li, Dayiheng Liu, Fei Huang, and 1 others. 2024. Qwen2 technical report. *arXiv preprint arXiv:2407.10671*.

Kai Yao, Zhaorui Tan, Penglei Gao, Lichun Li, Kaixin Wu, Yinggui Wang, Yuan Zhao, Yixin Ji, Jianke Zhu, and Wei Wang. 2025. Gradot: Training-free gradient-preserving offsite-tuning for large language models. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 5115–5130.

Lang Yu, Qin Chen, Jiaju Lin, and Liang He. 2023. Black-box prompt tuning for vision-language model as a service. In *IJCAI*, pages 1686–1694.

Jianyi Zhang, Saeed Vahidian, Martin Kuo, Chunyuan Li, Ruiyi Zhang, Tong Yu, Guoyin Wang, and Yiran Chen. 2024. Towards building the federatedgpt: Federated instruction tuning. In *ICASSP 2024-2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 6915–6919. IEEE.

Kaiyan Zhang, Ning Ding, Biqing Qi, Xuekai Zhu, Xinwei Long, and Bowen Zhou. 2023. Crash: Clustering, removing, and sharing enhance fine-tuning without full large language model. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 9612–9637.

Yuanhang Zheng, Zhixing Tan, Peng Li, and Yang Liu. 2024. Black-box prompt tuning with subspace learning. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 32:3002–3013.

Longguang Zhong, Fanqi Wan, Ruijun Chen, Xiaojun Quan, and Liangzhi Li. 2025. Blockpruner: Fine-grained pruning for large language models. In *Findings of the Association for Computational Linguistics: ACL 2025*, pages 5065–5080.

Ligeng Zhu, Zhijian Liu, and Song Han. 2019. Deep leakage from gradients. *Advances in neural information processing systems*, 32.

A More Details of Our Method

A.1 LayerDrop

The pseudocode of the LayerDrop algorithm is shown below.

Algorithm 2 LayerDrop

Input: Original model $\mathcal{M}$,pruning ratio β
Output: a list of layers

```
Get the layers of model: layers ← |M|
m, k ← len(layers), ⌊len(layers) × β⌋
stride ← (m − 1)/(k − 1)
for j ← 0 to k − 1 do
    i_j ← ⌊j × stride⌋
end for
return layers[i_0, ..., i_{k−1}]
```

A.2 Cooperative Prompt Knowledge Distillation

Under high pruning ratios (e.g., 0.5), the significant discrepancy between the emulator and the original model makes the adapter trained on the emulator difficult to transfer to the original model. To address this issue, methods such as OT (Xiao et al., 2023) align the two models through knowledge distillation (Hinton et al., 2015), with the loss function expressed as:

$$\mathcal{L}_{KD} = \mathbb{E}_{x\sim\mathcal{X}_d}||(\mathbf{H}_{\mathcal{E}}^{(-1)}(x), \mathbf{H}_{\mathcal{M}}^{(-1)}(x))||^2, \quad (13)$$

where $\mathcal{X}_d$ is the distillation dataset, and the notation $\mathbf{H}^{(-1)}$ represents the hidden state extracted from the final transformer layer. The subscripts $\mathcal{E}$ and $\mathcal{M}$ refer to the emulator and the original model, respectively. The term in the parentheses, e.g., (x), indicates the input provided to the model.

This approach, however, fails when using soft prompts as adapters. Unlike discrete tokens, soft prompts are vectors optimized in a continuous representation space. Traditional knowledge distillation aligns models only at discrete token instances, neglecting the broader continuous space. As a result, the emulator's learned soft prompt may occupy a misaligned position within this space, rendering its transfer to the original model problematic.

To address this challenge, we propose the Proxy Prompt Distillation Loss to align the continuous representation spaces of the emulator and the original model. We use randomly initialized soft prompts as proxies for the real soft prompts, prepending them to the distillation data. We then align the portions of the feature representations corresponding to the distillation data, which are generated by the emulator and the original model from the concatenated input. This loss can be formulated as:

$$\mathcal{L}_{PPD} = \mathbb{E}_{x\sim\mathcal{X}_d, P'\sim\mathcal{N}(\mu,\sigma^2)} ||(\mathbf{H}_{\mathcal{E},L_p:}^{(-1)}(P', x), \mathbf{H}_{\mathcal{M},L_p:}^{(-1)}(P', x))||^2, \quad (14)$$

where P' is the proxy soft prompt, L_p denotes its length, and $\mathcal{N}(\mu, \sigma^2)$ is a normal distribution with mean μ (e.g., 0) and standard deviation σ (e,g., 20), determined experimentally.

Additionally, following OT, we incorporate a language modeling loss when optimizing the emulator. Let n be the number of tokens in an input text x. The loss $\mathcal{L}_{LM}$ can be expressed as:

$$\mathcal{L}_{LM} = -\frac{1}{n}\sum_{i=1}^{n} \log p_{\mathcal{E}}(x_i|x_{1:i-1}). \quad (15)$$

Here, $p_{\mathcal{E}}(x_i|x_{1:i-1})$ denotes the probability of the emulator correctly predicting the i-th token given the preceding $i-1$ tokens.

Finally, the overall objective of CPKD for distilling the emulator can be expressed as:

$$\mathcal{E}^* = \arg\min_{\mathcal{E}} w_1\mathcal{L}_{LM} + w_2\mathcal{L}_{PPD} + w_3\mathcal{L}_{KD}, \quad (16)$$

where w_1, w_2 and w_3 are hyperparameters used to balance the contributions of each term.

A.3 Dataset Details

The datasets utilized in this study primarily consist of English text covering diverse domains and linguistic phenomena. The Pile-uncopyrighted encompasses a broad spectrum of general domains sourced from diverse text types, including academic papers, books, and web text. Regarding downstream benchmarks, OpenBookQA and ARC-Challenge focus on the scientific domain, containing elementary-level science exam questions that require logical reasoning and domain-specific knowledge. SocialIQA targets social commonsense reasoning regarding daily social interactions, while WebQuestions comprises factoid questions derived from search logs, representing general world knowledge. Wikitext-2 is composed of curated Wikipedia articles, representing the encyclopedic domain with formal linguistic structures. In terms of demographic groups, since these datasets

Datasets	Train.Num	Val.Num	Test.Num	Language
OBQA	4,957	500	500	English
SIQA	33.4K	1,954	-	English
ARC-c	1,119	299	1,172	English
WebQs	3,589	189	2,032	English
Pile-uncopyrighted	892K	180K	180K	English
Wikitext-2	36.7K	3,760	4,358	English

Table 3: Detailed statistics and characteristics of the datasets.

are sourced from the web, they likely reflect the inherent demographic biases of broad internet text and English-speaking contributors, with no specific demographic attributes explicitly filtered in this study.

Table 3 summarizes the statistics and characteristics of all datasets. Additionally, the instruction formats for the downstream task datasets are presented in Table 16.

A.4 Model Details

We conducted experiments on six commonly used LLMs: Qwen2-1.5B[2], Qwen2-7B[3], Gemma2-2B[4], Gemma2-9B[5], Llama3.2-3B[6], and Llama3.1-8B[7]. The architectural hyperparameters, training data size, and vocabulary size of these models are detailed as Table 4. While these model series possess multilingual capabilities, our experiments focus exclusively on their English performance to align with the downstream tasks.

A.5 Implementation Details

In the LayerDrop phase, we set the pruning rate to 0.25. During the LLE phase, we experiment with two learning rates, $1e-6$ and $2e-6$, using the initial 1% of the first chunk of the Pile-uncopyrighted dataset. We report the results from the emulator that achieves the best performance. In comparative experiments, the elevation margin H is set to 4. During the prompt tuning phase, we employ a soft prompt of length 5 and conduct a grid search over learning rates, reporting the best-performing run. The search grids are $\{1e-1, 7e-2, 3e-2, 3e-3, 1e-3\}$ for Qwen2-1.5B, Qwen2-7B, Gemma2-2B, and Gemma2-9B, and $\{1e-1, 3e-2, 1e-2, 1e-3, 3e-4\}$ for Llama3.2-3B and Llama3.1-8B. Additionally, for experiments involving CPKD, the emulator is distilled for one epoch on the initial 12.5% of the first Pile-uncopyright chunk with a learning rate of $4e-6$. The loss weights (w_1, w_2, w_3) are set to 1, 10, and 30, respectively. For all experiments, we report the average results across three independent runs. Regarding baseline implementation, for OT and CRaSh, we utilize the official codebases and conduct experiments following their recommended settings. For GradOT, we re-implemented the method based on its manuscript. To ensure a fair comparison, we set the pruning ratios of OT, CRaSh, and GradOT to be equivalent to that of LLEOT. We utilize `lm-eval-harness` to evaluate model performance on four downstream tasks: OpenBookQA, SocialIQA, ARC-Challenge, and WebQuestions. All evaluations are conducted using the default parameter settings provided by the repository. All experiments were conducted on two NVIDIA 80G A800 GPUs.

B Additional Experiments and Analysis

B.1 Additional Comparative Experiments

Table 5 presents the comparative experimental results on Gemma-2B, Llama-3B, and Llama-8B, utilizing experimental settings consistent with Table 1. These results corroborate the conclusions drawn in §5.2.

Furthermore, Table 6 displays the comparison between LLEOT (with CPKD enabled) and the baselines at a pruning ratio of 0.5. The results demonstrate that our method outperforms the baselines in terms of both plug-in performance and model capability privacy protection.

B.2 Ablation Experiments on Loss Landscape Elevation Methods

To evaluate the impact of different elevation strategies on the emulator, we compare LLE with the strategy of directly maximizing the language modeling loss (MLM). As shown in Table 7, LLE exhibits superior plug-in performance and comparable CPL metrics across the four downstream task datasets, which is consistent with the experimental conclusions in the main text. Similar observations on the importance of structural information have also been reported in other learning settings (Kou

[2]Qwen2-1.5B: https://huggingface.co/Qwen/Qwen2-1.5B-Instruct
[3]Qwen2-7B: https://huggingface.co/Qwen/Qwen2-7B-Instruct
[4]Gemma2-2B: https://huggingface.co/google/gemma-2-2b-it
[5]Gemma2-9B: https://huggingface.co/google/gemma-2-9b-it
[6]Llama3.2-3B: https://huggingface.co/meta-llama/Llama-3.2-3B-Instruct
[7]Llama3.1-8B: https://huggingface.co/meta-llama/Llama-3.1-8B-Instruct

Models	Qwen2-1.5B	Qwen2-7B	Gemma2-2B	Gemma2-9B	Llama3.2-3B	Llama3.1-8B
Hidden Size	1,536	3,584	2,304	3,584	3,072	4,096
Layers	28	28	26	42	28	32
Query Heads	12	28	8	16	24	32
Key Value Heads	2	4	4	8	8	8
Head Size	128	128	256	256	128	128
Vocabulary Size	151,936	152,064	256,000	256,000	128,256	128,256
Trained Tokens	7T	7T	2T	8T	9T	15T

Table 4: Details of large language models.

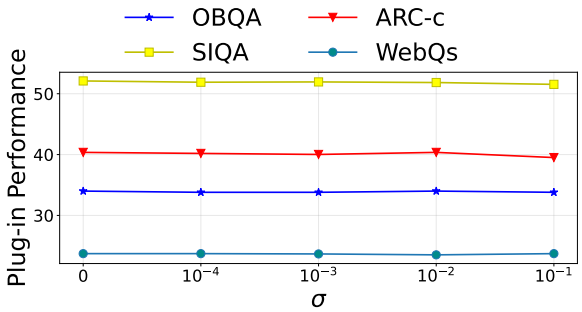

Figure 7: Variations in the average accuracy of LLEOT across different noise intensities on Qwen2-1.5B (pruning ratio = 0.25). Here, σ denotes the standard deviation of Gaussian noise applied to the fine-tuned soft prompts.

et al., 2024).

B.3 Compatibility with Data Privacy Strategies

To verify that the LLEOT framework is orthogonal to data privacy strategies, we incorporate the widely used randomization privacy protection strategy (Zhu et al., 2019; Kang et al., 2024) into LLEOT. Such compatibility is important in practical privacy-sensitive deployments, where model adaptation is often combined with application-level privacy-preserving designs (Hong et al., 2026b). Specifically, after prompt tuning, the data owner adds Gaussian noise to the soft prompt before uploading it to the model owner. This method is known to significantly reduce the success rate of gradient inversion attacks (Zhu et al., 2019), thereby preventing the model owner from deducing private data.

Figure 7 illustrates the variation in the original model's average accuracy with the introduction of noise intensity. Unexpectedly, the noise added to the soft prompts has a negligible impact on model performance. We attribute that this robustness stems from the high smoothness of the original model's input embedding space, resulting from its pre-training on massive amounts of data. This smoothness ensures that small perturbations to the embedding vectors do not significantly alter the model's output. Therefore, the randomization privacy protection strategy can be integrated into the LLEOT framework, enhancing data privacy at a negligible cost to performance.

B.4 Computational Overhead of Emulator Construction

We analyze the time consumption associated with our proposed LLEOT framework, reporting experimental results obtained on two 80GB NVIDIA A800 GPUs. As shown in Table 8, in comparison with OT, CRaSh, and GradOT, our method incurs significantly lower time costs than OT. While CRaSh and GradOT require less time, this efficiency comes at the cost of leaving the pruned emulator's parameters unmodified, thereby compromising the parameter privacy of the original model.

B.5 Comparison of Adapter Sizes

As shown in Table 9, our method employs an adapter with a parameter count that is significantly lower than the existing methods, thereby drastically reducing the consumption of computational resources.

B.6 The Selection of The Prompt Length

We set the prompt length to 5. This choice is motivated by the fact that it yields the most lightweight adapter, and our prompt tuning experiments on the original model show that performance is stable across different length settings (as shown in Table 10).

B.7 Difficulty of Restoring Inference Capabilities via Fine-Tuning with Limited Data

We investigate whether the emulator's inference capabilities can be restored through fine-tuning with

Method	OBQA		SIQA		ARC-c		WebQs	
	Acc(↑)	CPL(↓)	Acc(↑)	CPL(↓)	Acc(↑)	CPL(↓)	Acc(↑)	CPL(↓)
Gemma2-2b								
Full ZS	35.60	100.00	50.00	100.00	50.85	100.00	8.07	100.00
Full PT	45.80	100.00	56.60	100.00	54.30	100.00	38.09	100.00
Random	17.04	47.87	33.17	66.34	21.90	43.07	0.00	0.00
OT	42.00	82.58	55.47	90.68	51.76	69.97	23.01	64.06
CRaSh	35.93	52.81	55.46	76.36	52.05	47.16	23.96	2.48
GradOT	37.80	47.19	55.52	71.02	51.02	**37.74**	23.49	0.00
Ours	**43.00**	**36.52**	**57.16**	**70.00**	**52.96**	40.61	**26.80**	**0.00**
Llama3.2-3b								
Full ZS	28.20	100.00	45.04	100.00	43.69	100.00	11.32	100.00
Full PT	36.11	100.00	56.42	100.00	48.12	100.00	36.88	100.00
Random	14.76	52.34	33.35	74.05	22.49	51.48	0.00	0.00
OT	30.67	90.07	49.43	97.16	43.77	78.32	**27.51**	74.73
CRaSh	28.73	65.25	49.13	84.88	43.94	66.40	22.57	1.33
GradOT	29.20	48.23	50.05	79.31	44.67	47.06	19.01	0.00
Ours	**35.20**	**42.55**	**54.40**	**76.93**	**47.67**	**46.67**	24.78	**0.00**
Llama3.1-8b								
Full ZS	33.80	100.00	49.39	100.00	51.71	100.00	9.40	100.00
Full PT	39.80	100.00	59.44	100.00	56.31	100.00	41.04	100.00
Random	15.64	46.27	33.24	67.31	22.29	43.10	0.00	0.00
OT	34.93	83.43	52.83	91.29	**53.75**	70.95	34.44	66.49
CRaSh	35.67	64.50	57.06	80.93	51.82	69.64	29.23	8.94
GradOT	36.53	49.70	57.18	73.03	52.30	43.22	33.31	0.00
Ours	**38.33**	**42.60**	**59.23**	**69.73**	53.44	**40.26**	**37.11**	**0.00**

Table 5: Additional comparative experiment results.

limited data. Specifically, we fine-tuned the emulator using two types of data: (1) a small public subset consisting of 10,000 samples from WikiText-2 (Merity et al., 2016), and (2) the 'leaked outlier data', namely the downstream-task instances that were still answered correctly by the emulator. As shown in Table 11, the CPL of the fine-tuned emulator exhibits only negligible differences compared with that before fine-tuning, indicating that the emulator cannot be easily restored. These results suggest that fine-tuning on either small public datasets or leaked samples is insufficient to recover the emulator's inference capabilities.

B.8 Impact of Elevation Dataset on Generalization

In our experiments, we utilized pile-uncopyrighted (a general-domain plain text corpus) as elevation dataset $\mathcal{D}_e$ for highly specialized downstream tasks (e.g., the science multiple-choice question dataset OBQA) already demonstrates strong generalization across substantial domain shifts.

To further validate this, we tested alternative elevation datasets: BoolQ (domain-proximal to OBQA as demonstrated in CRaSh (Zhang et al., 2023)) and WebQs (domain-divergent to OBQA, real-world search queries vs. elementary science facts). As Table 12 shows, LLEOT consistently exhibits robust generalization to unseen downstream domains, regardless of the specific $\mathcal{D}_e$ chosen.

B.9 Experiments on Additional Adapter Types

To evaluate the generalization of our method across different adapter types, we follow OT (Xiao et al., 2023) and use the bottom two and top two transformer layers as adapters. As shown in Table 13, under this setting, the emulator achieves a CPL comparable to that of soft-prompt adapters, while the plug-in performance is even higher. We at-

Method	OBQA		SIQA		ARC-c		WebQs	
	Acc(↑)	CPL(↓)	Acc(↑)	CPL(↓)	Acc(↑)	CPL(↓)	Acc(↑)	CPL(↓)
Qwen2-1.5b								
OT	27.20	70.02	46.80	86.90	37.29	66.34	21.65	166.59
CRaSh	24.67	54.68	48.00	76.35	39.33	58.71	18.16	0.00
GradOT	28.60	50.36	48.00	**74.33**	37.97	56.18	11.61	0.00
Ours	**34.20**	**46.52**	**50.04**	75.87	**40.44**	**48.39**	**24.15**	**0.00**
Gemma2-2b								
OT	39.00	74.91	50.05	82.57	38.40	56.61	**25.26**	27.49
CRaSh	35.80	48.31	51.50	70.66	49.57	43.46	19.47	3.08
GradOT	38.40	43.26	50.46	**69.08**	50.63	42.50	18.20	0.00
Ours	**44.87**	**38.01**	**55.01**	70.15	**52.56**	**39.76**	22.44	**0.00**
Llama3.2-3b								
OT	29.47	77.30	43.30	91.05	39.62	60.54	**23.90**	36.26
CRaSh	26.07	65.25	47.50	77.38	43.69	52.92	21.67	0.00
GradOT	29.80	55.32	39.66	**76.80**	45.05	49.21	10.77	0.00
Ours	**33.40**	**53.43**	**48.21**	81.25	**46.96**	**47.19**	15.45	**0.00**

Table 6: Comparative experimental results (pruning rate = 0.5). 'Ours' denotes LLEOT with CPKD.

Method	OBQA		SIQA		ARC-c		WebQs	
	Acc(↑)	CPL(↓)	Acc(↑)	CPL(↓)	Acc(↑)	CPL(↓)	Acc(↑)	CPL(↓)
MLM	14.80	54.68	48.82	**70.48**	21.24	**53.44**	13.78	0.00
LLE	**34.00**	**43.88**	**52.10**	72.80	**40.36**	54.60	**23.72**	**0.00**

Table 7: Ablation study on loss landscape elevation strategies on Qwen2-1.5B (pruning ratio = 0.25). 'MLM' denotes the strategy of maximizing language modeling loss. Best in **bold**.

Method	OT	CRaSh	GradOT	Ours
Time consumption	80 hours	5 mins	10 mins	1.5 hours

Table 8: Comparison of time consumption among different methods (Qwen2-1.5b).

Method	OT	CRaSh	GradOT	Ours
Qwen2-1.5B	187.2M	187.2M	187.2M	7.6K
Qwen2-7B	932.2M	932.2M	932.2M	17.9K
Gemma2-2B	311.5M	311.5M	311.5M	11.5K
Gemma2-9B	792.8M	792.8M	792.8M	11.5K
Llama3.2-3B	402.7M	402.7M	402.7M	15.4K
Llama3.1-8B	872.4M	872.4M	872.4M	20.5K

Table 9: Parameter counts of adapters for different methods.

tribute this improvement to the larger parameter capacity of transformer-layer adapters, which enables more effective knowledge transfer. These results demonstrate that LLEOT remains effective across different adapter types.

B.10 Experiments on a More Recent Model

To further validate the generality of our framework, we have conducted additional experiments on Qwen3-4B (Yang et al., 2025), a more recent model. The results, now reported in Table 14, demonstrate that our method remains effective on Qwen3.

B.11 Case Study on Emulator Outputs

We further present a qualitative case study by comparing the outputs of different emulators under the same input. As shown in Table 15, the baselines, especially OT, often still produce semantically meaningful and sometimes even correct answers, indicating that they retain substantial residual inference capability. CRaSh and GradOT, although generally less accurate, also frequently preserve task-related semantic fragments or partial natural-language flu-

prompt length	5	10	20	50
OBQA	35.80	34.80	35.80	35.20
ARC-c	41.98	41.98	42.49	43.17

Table 10: Acc with different prompt lengths (Qwen2-1.5b).

	OBQA	SIQA	ARC-c	WebQs
Random	55.83	71.72	60.00	0.00
Origin Emulator	43.88	72.80	54.60	0.00
Fine-tuned By Wikitext-2	47.48	74.46	55.27	0.00
Fine-tuned By OBQA	47.48	73.34	54.81	0.00
Fine-tuned By SIQA	46.04	73.34	55.05	0.00

Table 11: CPL of the fine-tuned emulator (Qwen2-1.5b).

ency, suggesting that their emulators remain partially functional. In contrast, the emulator produced by LLEOT consistently yields unintelligible outputs across all three cases, without providing usable answers or coherent reasoning content. These examples further demonstrate that LLEOT is more effective at disabling the emulator's inference capability and thus offers stronger protection of model capability privacy.

C The Use of LLMs

In the preparation of this manuscript, we employed a large language model (LLM), specifically Gemini 2.5 Pro (Comanici et al., 2025), as a writing aid. The LLM's role was explicitly restricted to language refinement and did not involve any facet of the research conceptualization or scientific methodology. Our process consisted of providing the LLM with drafts and specific sentences. We then utilized the model's suggestions to polish sentence construction, enhance clarity and flow, and verify grammatical accuracy in the final text. It is essential to declare that all central scientific contributions—including the motivation for this study, the definition of the model capability privacy concept and its associated metric, the algorithmic architecture and theoretical analysis of LLEOT, and the experimental design and interpretation of results—are exclusively the work of the human authors. The LLM was not utilized to formulate scientific claims, hypotheses, or conclusions. The authors have fastidiously reviewed, edited, and confirmed all content in this paper. We assume complete responsibility for the final manuscript, encompassing its scientific precision and integrity.

Elevation Dataset	**OBQA**		**WebQs**	
	Acc(↑)	CPL(↓)	Acc(↑)	CPL(↓)
Pile	34.00	43.88	23.72	0.00
BoolQ	34.20	44.60	21.46	0.00
WebQs	33.80	48.20	23.77	0.00

Table 12: Experiments with different elevation datasets (Qwen2-1.5B).

D Proof of Theorem 1.

Proof. We first show that LLE effectively degrades the emulator's inference ability. From the definition of cross-entropy loss and Eq. (2), we obtain

$$\begin{aligned}
&\mathcal{L}_{\mathcal{E}}(P;x) - \mathcal{L}_{\mathcal{M}}(P;x) \\
&\quad = -\tfrac{1}{n}\sum_{i=1}^{n}\log(p_{\mathcal{E}}(x_i \mid P, x_{1:i-1})) \\
&\qquad + \tfrac{1}{n}\sum_{i=1}^{n}\log(p_{\mathcal{M}}(x_i \mid P, x_{1:i-1})) \\
&\quad = H > 0.
\end{aligned} \tag{17}$$

Here, n denotes the number of tokens to predict in x, $p_{\mathcal{E}}(x_i|P, x_{1:i-1})$ denotes the probability of the emulator correctly predicting the i-th token given the soft prompt and the preceding $i-1$ tokens. Defining $\hat{p}_{\mathcal{E}}(x|P) = \prod_{i=1}^{n} p_{\mathcal{E}}(x_i|P, x_{1:i-1})$ and $\hat{p}_{\mathcal{M}}(x|P) = \prod_{i=1}^{n} p_{\mathcal{M}}(x_i|P, x_{1:i-1})$, Eq. (17) can be transformed into:

$$\hat{p}_{\mathcal{E}}(x|P) = \mathrm{e}^{-nH}\hat{p}_{\mathcal{M}}(x|P) \tag{18}$$

To analyze the impact of this loss difference on model performance, we consider the perplexity (PPL), a standard metric for evaluating language models. Perplexity is defined as:

$$\begin{aligned}
\mathrm{PPL} &= \exp\left(-\frac{1}{n}\sum_{i=1}^{n}\log(p_i)\right) \\
&= \exp\left(-\frac{1}{n}\log(\hat{p})\right) \\
&= \hat{p}^{-1/n}.
\end{aligned} \tag{19}$$

Adapter Type	OBQA		SIQA		ARC-c		WebQs	
	Acc(↑)	CPL(↓)	Acc(↑)	CPL(↓)	Acc(↑)	CPL(↓)	Acc(↑)	CPL(↓)
Qwen2-1.5B								
Soft Prompt	34.00	43.88	52.10	72.80	40.36	54.60	23.72	0.00
Transformer Layers	37.60	39.57	53.58	75.45	42.66	52.74	33.91	0.00
Qwen2-7B								
Soft Prompt	41.40	36.57	55.15	68.42	55.57	40.56	35.95	0.00
Transformer Layers	41.60	35.43	59.52	66.38	48.38	38.90	42.37	0.00

Table 13: Experiments with Alternative Adapter Types (Qwen2-1.5B and Qwen2-7B)

Method	OBQA		SIQA		ARC-c		WebQs	
	Acc(↑)	CPL(↓)	Acc(↑)	CPL(↓)	Acc(↑)	CPL(↓)	Acc(↑)	CPL(↓)
Full ZS	31.80	100.00	50.15	100.00	55.63	100.00	4.82	100.00
Full PT	38.40	100.00	55.94	100.00	57.25	100.00	34.99	100.00
Random	16.20	50.94	33.47	66.74	23.12	41.56	0.00	0.00
Ours	34.40	44.03	53.38	67.76	57.08	36.81	23.57	0.00

Table 14: Experiments on Qwen3-4B

From Eq. (19), the perplexity of the emulator can be expressed as:

$$\begin{aligned}\mathrm{PPL}_{\mathcal{E}} &= \exp\left(-\frac{1}{n}\sum_{i=1}^{n}\log(p_{\mathcal{E}}(x_i \mid P, x_{1:i-1}))\right)\\ &= \exp\left(-\frac{1}{n}\log(\prod_{i=1}^{n}p_{\mathcal{E}}(x_i \mid P, x_{1:i-1}))\right)\\ &= \exp\left(-\frac{1}{n}\log(\hat{p}_{\mathcal{E}}(x|P))\right)\\ &= \hat{p}_{\mathcal{E}}(x|P)^{-1/n}.\end{aligned} \tag{20}$$

This can be rewritten as:

$$\hat{p}_{\mathcal{E}}(x|P) = \mathrm{PPL}_{\mathcal{E}}^{-n}. \tag{21}$$

Similarly, for the original model, we have:

$$\hat{p}_{\mathcal{M}}(x|P) = \mathrm{PPL}_{\mathcal{M}}^{-n}. \tag{22}$$

By substituting Eq. (21) and Eq. (22) into Eq. (18), we can express the relationship between the perplexities of the two models:

$$\mathrm{PPL}_{\mathcal{E}} = \mathrm{e}^{H} \cdot \mathrm{PPL}_{\mathcal{M}}. \tag{23}$$

It shows that the emulator's PPL is exponentially greater than the original model's by a factor of e^{H}. Since lower PPL indicates better performance, a larger H will lead to a significantly higher PPL for the emulator, thereby degrading its inference capabilities.

In addition, we show that LLE maintains the emulator's gradient guidance consistent with that of the original model. Specifically, the emulator's gradient with respect to soft prompts can be expressed as:

$$\begin{aligned}\nabla_P\mathcal{L}_{\mathcal{E}}(P;x) &= \nabla_P\left(\mathcal{L}_M(P;x) + H\right)\\ &= \nabla_P\mathcal{L}_M(P;x) + \nabla_P H\\ &= \nabla_P\mathcal{L}_M(P;x),\end{aligned} \tag{24}$$

the gradient vectors of the emulator and the original model are identical. During prompt tuning, the emulator and the original model exhibit consistent gradient optimization directions and magnitudes at each step, ultimately converging to the same optimal soft prompt. □

E Artifact Licenses, Terms of Use, and Intended Use

Models In this work, we utilize several large language models strictly for algorithm evaluation and privacy research. We adhere to their respective licenses and Acceptable Use Policies, which include avoiding the generation of harmful content. Specifically, the Qwen2 series (Qwen2-1.5B, Qwen2-7B) and Qwen3-4B is used under the Apache License 2.0. The Llama 3 series (Llama-3.2-3B, Llama-3.1-8B) is used in compliance with the Llama 3 Community License Agreement. The Gemma 2 series (Gemma2-2B, Gemma2-9B) is used subject to the Gemma Terms of Use.

Input	Output
Who painted the Mona Lisa?	**OT:** Who was Leonardo da Vinci? Who is the Mona Lisa's artist? These questions are asked all the time and there is no single answer. This painting by Italian Renaissance painter, Leonardo da Vinci, has been a subject of debate s. **CRaSh:** The painting of the Mancu's family portrait is in: ___. A) Europe; or 18th century France, or Italy. B) Europe, or 20th century, such like Spain. C) Europe and France; but not a little part from this world; since it's b. **GradOT:** In a short, there is not only to. The reason of a link cannot be a computer science and Anonymously, "It's Never Before" column. Anxiety has never been written in London (and it's Not to use for a Computer Science and It se. **Ours:** `\1}}\120 \0} ( \* \\ \ \\1}\ \0, \\}\1\\[ \ \3}{0 \\\1 \ \ \{} \ \}} 01 \\ \11\ \ \1, \\10\0\\\ \10 \ \1\1} {2,} \0}1_{ \ \}2 \ \01 \\} \1\{\12\ \0 \ \0\ \\1 \\`
What gas do humans inhale to survive?	**OT:** Humans breathe in oxygen. Oxygen is a compound of two elements, O and H, and it can be divided into two types: molecular oxygen (O2) and gaseous oxygen (O2). Both are used by the body for cellular respiration. In. **CRaSh:** The oxygen cycle is not only the basis for our body's functioning but also a significant determinant of health. The primary purpose of this module is the understanding and comprehension of the basic princi. **GradOT:** As of the gas in this passage is an instrument and also as I can use, but it is not possible to get and call. We are looking for a specific To help With a specific To Do Notic To help you To Help to understand th. **Ours:** `1 \ \1,1}\2\\0\1}{\1\0, \\1{10, \0 \30}\\022, \11} \ \ \\2 \2111,02] \}1}\[11 \\}\ \1\10 \\ \2 \21\ \1 \1 \\\ \4 \\`
where do the san francisco giants play home games?	**OT:** Options: baseball park, ballpark, stadium, ballpark. The answer is baseball park. The Giants play at AT&T Park in San Francisco. The Giants are based in San Francisco and play their home games there. A: At the moment, the San Francisco Giants play t. **CRaSh:** The Chicago Yankees are an American 5-8. It is not an issue. As of date, No San-Indians have not been able to be named 50% or more successful than any other baseball team, according to NMAE*). The Boston Red Sox and Seattle's Red Brigade are also bo. **GradOT:** The games the The game The game The game- the san games like the game like the game games ga. **Ours:** `\{\\ \{}1 \}\ \*{2,\}$\\} \\ \0 \[00]\$\1 \ \\3\{01\}{{0,1\0}10 \2\1\\ \ \11\121 \0 \11, \22} \1121\1200,} \ \0\0, 22011-4\10,\\\\\ \{2}{\11},{} \\02{\\ \1} \1, \} \ \ \\`

Table 15: Case study comparing raw outputs produced by different emulators under the same input.

Datasets The datasets employed in this study are benchmark datasets, and their use is consistent with their purpose as academic benchmarks, i.e., solely for training and evaluating language modeling and reasoning capabilities. The Pile-uncopyrighted dataset is distributed under the MIT License. The ARC-Challenge and SocialIQA datasets are used in compliance with their respective Creative Commons licenses (CC-BY-SA / CC-BY). The OpenBookQA and WebQuestions datasets are publicly available for research purposes, and we utilize them strictly for academic evaluation. The Wikitext-2 dataset is used under the CC-BY-SA 3.0 license.

Code We utilize implementations from existing open-source projects. This includes codebases for OT, CRaSh, and `lm-eval-harness`. We ensure that our use of these third-party codebases is in accordance with their respective licenses (e.g., the MIT License). Our own code and emulator configurations, developed as part of this research, will be released under the MIT License upon publication to facilitate future academic research.

Compliance and Intended Use The artifacts created in this work (i.e., the LLEOT source code and emulator construction scripts) are intended to facilitate academic research on privacy-preserving fine-tuning. This intended use is fully compatible with the original access conditions of the base models and datasets. Specifically, since the datasets and models used are open for research, our derivative artifacts are released for research purposes to ensure compliance with the original distribution terms.

OBQA
What happens when mercury is placed in water? *it sinks.*
Which is a good source of nutrients for a mushroom? *a cut peony.*
SIQA
Q: Sydney was a school teacher and made sure their students learned well. How would you describe Sydney? A: *As someone that takes teaching seriously.*
Q: Kendall's dog was overweight so they walked it five miles. Why did Kendall do this? A: *start an exercise regimen.*
ARC-c
Question: What do cells break down to produce energy? Answer: *food.*
Question: How are the particles in a block of iron affected when the block is melted? Answer: *The particles move more rapidly.*
WebQs
Question: what is nina dobrev nationality? Answer: *Bulgaria.*
Question: what electorate does anna bligh represent? Answer: *Electoral district of South Brisbane.*

Table 16: Instructions format of downstream task dataset