Learning to Keep Secrets: Empowering Local LLMs with Autonomous Dynamic Privacy-Conscious Delegation

Anonymous ACL submission

Abstract

While cloud-hosted Large Language Models (LLMs) offer superior capabilities, they introduce significant privacy risks. Conversely, local models ensure data sovereignty but often suffer from limited performance. Privacy-conscious delegation provides a solution by enabling local models to collaborate with remote models. In privacy-conscious delegation, a local model synthesizes a privacy-preserving prompt, queries a remote model, and combines the returned information with private context locally. However, existing methods rely on prompting to enforce privacy constraints, which can be brittle for local models, and they typically delegate in a static manner, causing unnecessary exposure when a local solution is sufficient. To address these limitations, we propose **ADAPT** (**A**utonomous **D**elegation **A**gent with **P**rivacy **T**raining), a framework that transforms delegation from a static pipeline into a dynamic, learnable agent. We train the local model to autonomously manage the entire execution path–from determining delegation triggers to synthesizing privacy-preserving prompts–thereby fostering robust, intrinsic privacy-preserving capabilities. Experimental results demonstrate that **ADAPT** significantly outperforms existing baselines, achieving a superior performance.

1 Introduction

The rapid advancement of LLMs has led to their widespread adoption and an increasing societal dependence on these systems. However, many LLMs are hosted by untrusted cloud providers (denoted as M_{remote}), posing significant privacy risks as users may transmit privacy-sensitive information within their prompts. To mitigate these risks, the task of privacy-conscious delegation has garnered attention. As shown in Figure 1, within this task, a local model (denoted as M_{local}) attempts to synthesize privacy-preserving prompts before querying M_{remote}, subsequently merging the remote responses with the local private context to generate the final response.

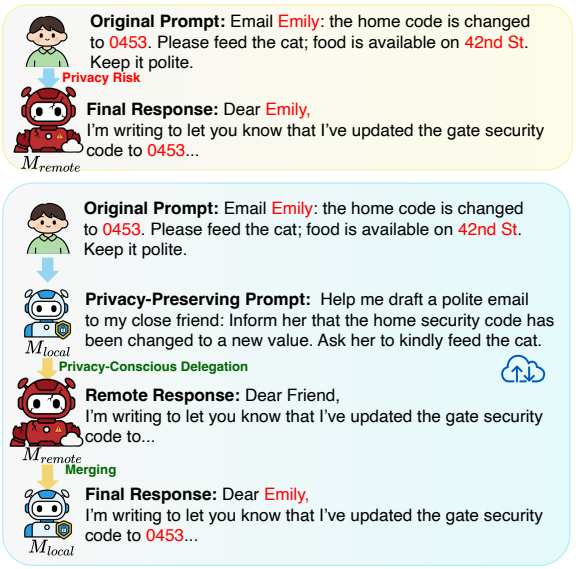

Figure 1: An illustration of privacy-conscious delegation in our research. Given a prompt that may contain privacy-sensitive information. M_{local} autonomously generates a privacy-preserving prompt for the remote model M_{remote} and integrates the returned remote response with the private context locally.

Prior research has made progress. Some approaches leverage in-context learning for privacy mitigation (Hartmann et al., 2024). CoGenesis utilizes high-level guidance from M_{remote} to assist M_{local} in handling private contexts for static tasks (Zhang et al., 2024). Similarly, PAPILLON (Li et al., 2025) enables the M_{local} to synthesize privacy-preserving prompts to leverage the M_{remote} capabilities, merging the results locally. However, these methods overlook two critical limitations. First, relying on prompt engineering alone to enforce complex privacy constraints is often brittle for M_{local}. We believe privacy requirements are typically strict and nuanced, making them ill-suited to being reliably satisfied through prompting

without updating the parameters of M_{local}. Second, they adopt a static pipeline that delegates tasks to M_{remote} unconditionally. They overlook the inherent problem-solving potential of M_{local} and introduce avoidable privacy risks by involving M_{remote} even when M_{local} is sufficient to handle the prompt.

To overcome these, our research is built upon two core perspectives to bridge these gaps. First, we posit that privacy consciousness is often insufficiently established through prompting; instead, **privacy consciousness should be trained into the model's parameters via training to ensure robust privacy-conscious behavior.** Second, we advocate for **the principle of data minimization: the most effective defense is non-transmission.** Unlike traditional static pipelines that incur unnecessary exposure, our approach empowers M_{local} to act as an autonomous agent for dynamic delegation, thereby significantly minimizing privacy leakage.

Overall, we propose **ADAPT** (**A**utonomous **D**elegation **A**gent with **P**rivacy **T**raining), a framework that transforms the local model M_{local} from a passive sanitizer into an active, privacy-conscious agent. Unlike prior static approaches, ADAPT utilizes reinforcement learning to internalize privacy consciousness directly into the M_{local}. And we design a dynamic delegation mechanism where M_{local} learns to autonomously evaluate the necessity of external delegation. If an original prompt lies within the local model's competence, ADAPT resolves it locally; otherwise, it generates a synthesized privacy-preserving prompt for M_{remote} and merges the remote results from M_{remote} with local private context to provide superior response quality. By optimizing a reward function that balances response quality against privacy leakage, ADAPT's local model learns how to effectively collaborate with M_{remote} while preserving privacy. Our contributions are summarized as follows:

- ADAPT transforms the M_{local} into an autonomous, dynamic privacy-conscious agent, allowing it to decide when external delegation is needed.
- ADAPT uses reinforcement learning to internalize privacy consciousness in the M_{local}, optimizing privacy and performance.
- Experiments show that ADAPT improves the model's capabilities and effectively completes the privacy-conscious delegation task.

2 Related Work

2.1 Training and Output Privacy

LLMs are known to verbatim memorize significant portions of their training data, which facilitates data extraction and membership inference attacks (Carlini et al., 2021; Nasr et al., 2023; Carlini et al., 2019; Lehman et al., 2021). This risk is particularly acute during task-specific finetuning on specialized or sensitive datasets, such as clinical or proprietary records (Mireshghallah et al., 2022). Beyond memorization, privacy breaches frequently occur through LLM-generated outputs; models may inadvertently regurgitate sensitive context provided during In-Context Learning or Retrieval-Augmented Generation (Duan et al., 2024; Xia et al., 2025). To mitigate these risks, techniques such as machine unlearning (Ren et al., 2025; Liu et al., 2025) and differential privacy (Li et al., 2021) have been proposed to sanitize model weights, though they often incur non-trivial trade-offs in model utility.

2.2 Prompt-Based Inference Privacy

During deployment, privacy is threatened by both direct and latent information leakage. Static NER-based masking or scrubbing (Chen et al., 2023; Pilán et al., 2022) often fails to capture complex semantic dependencies, leading to significant utility degradation. Beyond simple sanitization, privacy-conscious delegation has emerged as a promising paradigm, wherein M_{local} manages privacy-sensitive data while remote large models M_{remote} provide advanced capabilities. One proposed mechanism involves the local synthesis of a similar but novel fake problem; the resulting response from the remote model for this synthetic task is then utilized as an in-context example to assist the local model in resolving the original private query (Hartmann et al., 2024). CoGenesis (Zhang et al., 2024) provides high-level guidance from remote LLMs to help local models process private contexts. PAPILLON (Li et al., 2025) attempts to leverage the superior capabilities of M_{remote} by synthesizing privacy-preserving prompts, subsequently merging the remote results with private context locally. More recently, PrivacyPad (Hui et al., 2025) tries to forward only non-sensitive sentences.

Moving beyond static frameworks, ADAPT leverages reinforcement learning to internalize privacy-consciousness while empowering autonomous delegation, enabling M_{local} to dynamically balance privacy and utility.

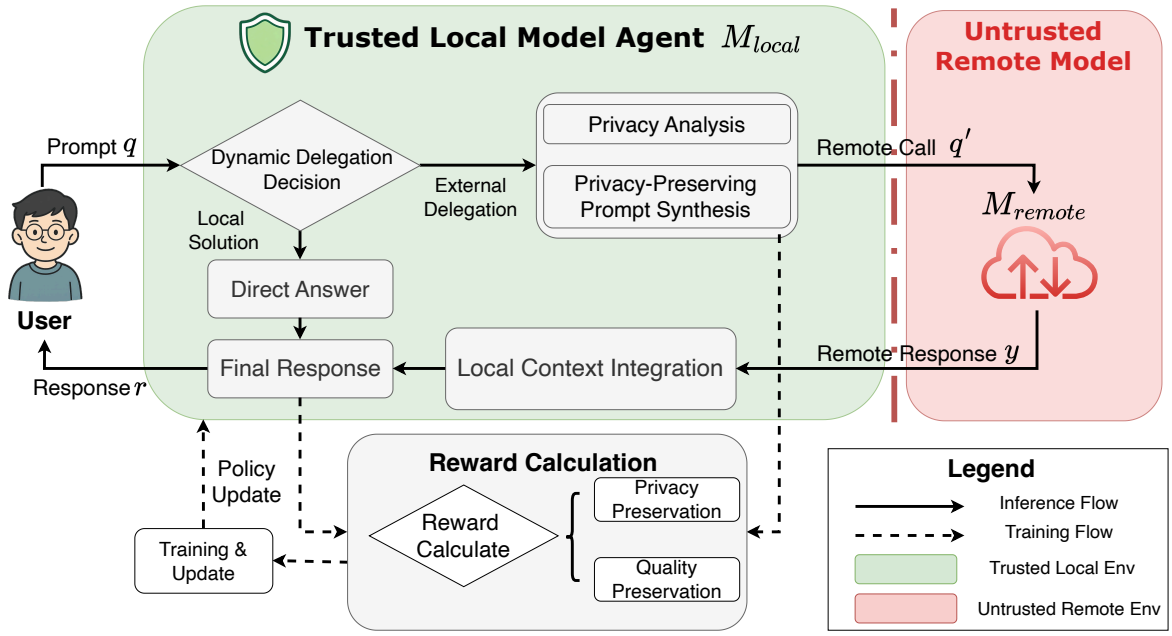

Figure 2: Overview of the ADAPT framework. M_{local} operates as an autonomous agent. **Solid arrows** indicate the inference flow: M_{local} adaptively chooses between local solution and external delegation using a privacy-preserving prompt, subsequently integrating the remote response to generate the final response r. **Dashed arrows** illustrate the training flow: the model is optimized via a reward mechanism that balances utility and privacy, adhering to the principle of data minimization.

3 ADAPT: Autonomous Delegation Agent with Privacy Training

3.1 Overview

As illustrated in Figure 2, ADAPT trains M_{local} to function as an autonomous agent. Given a user prompt q containing PII (Personally Identifiable Information) units $P = \{p_1, \ldots, p_n\}$, the agent adaptively decides whether to resolve the prompt locally or delegate it to M_{remote}. In the case of delegation, it utilizes a privacy-preserving prompt q' and incorporates the remote response to produce the final response r.

Following prior work (Hui et al., 2025; Li et al., 2025), we quantify privacy leakage as the fraction of PII units exposed to M_{remote}.

3.2 Action Protocol and Delegation Decisions

To make privacy-conscious delegation behaviors learnable and controllable, we structure each inference instance into a lightweight two-stage protocol: (i) decision and privacy-preserving prompt synthesis, and (ii) external delegation and local integration. Under this protocol, M_{local} is always the model that produces the final user-facing response, while M_{remote} only provides information when delegation is selected.

Decision and Privacy-Preserving Prompt Synthesis Given a user prompt q, M_{local} first generates a structured delegation block `<delegate_external>` to explicitly express its action choice. If M_{local} judges that external delegation is unnecessary, it outputs an empty `<delegate_external></delegate_external>` block. Otherwise, it first produces `<pii_analysis>` to identify PII units, and then synthesizes a privacy-preserving delegated prompt q' `<delegate_prompt>`.

External Delegation and Local Integration If M_{local} chooses delegation, q' is sent to M_{remote} to obtain a remote response y. The system then provides this response to M_{local} as an observation wrapped in `<remote_LLM>`. If no delegation is selected, the system provides an empty `<remote_LLM></remote_LLM>` observation. Finally, M_{local} generates the final response r locally by integrating the original private context q with the observed remote response y, and encloses it within an `<answer>` tag.

We make delegation an explicit action interface: M_{local} decides whether to delegate by generating `<delegate_external>`, and, when delegating, takes responsibility for PII analysis and privacy-preserving prompt synthesis.

3.3 Reward Design

Privacy Preservation To quantify the extent of information exposure during delegation, we measure the leakage of PII units within the synthesized prompt q'. Given a set of PII units $P = \{p_1, \ldots, p_n\}$ extracted from the original prompt q, the privacy leakage score is defined as the average exposure rate:

$$\textit{Leak}(q', p_{1..n}) = \frac{1}{n} \sum_{j=1}^{n} \mathbb{I}\big(M_J(f_{PJ}, q', p_j)\big), \quad (1)$$

where f_{PJ} represents the prompt template for privacy judgment, M_J denotes an LLM-based judge, and $\mathbb{I}(\cdot)$ is an indicator function that converts the judge's output into a binary value. Rather than relying on rigid string matching, which is often susceptible to circumvention, we utilize an LLM-based judge to capture semantic exposure. This approach has been validated in prior research as an effective means of identifying latent PII leakage (Li et al., 2025).

If M_{local} does not perform external delegation, the privacy leakage score is defined as zero.

Quality Preservation The quality of the generated response r is assessed by comparing it against a reference response r^* provided in the dataset. We compute the response quality score by asking an LLM judge whether r is at least as good as r^*. The response quality score is defined as:

$$\textit{Qual}(r, r^*) = \mathbb{I}\big(M_J(f_{QJ}, r, r^*)\big), \quad (2)$$

where f_{QJ} is the evaluation prompt and $\mathbb{I}(\cdot)$ is a binary indicator. To mitigate positional bias, we perform two calls to the judge using permuted input pairs. We convert the judge's free-text output into a binary score based on whether the response starts with "Yes." If the judge provides identical answers for both permutations, indicating an inconsistent judgment, the responses are classified as equivalent in quality. A detailed description of the scoring algorithm is provided in Appendix B.

Final Reward We define our reward to encourage maximizing response quality score while minimizing privacy leakage score.

$$\mathcal{R} = \textit{Qual}(r, r^*) - \textit{Leak}(q', p_{1..n}), \quad (3)$$

Our M_{local} is designed to develop sophisticated strategies that enhance response quality while minimizing data exposure.

3.4 Model Training

We train M_{local} in two stages: a supervised finetuning phase for format alignment, followed by reinforcement learning to optimize for reward maximization.

Supervised Finetuning While some prior work suggests that supervised finetuning offers limited benefits for training agentic behaviors (Guo et al., 2025), we find it essential in our setting. Given that our study targets small models, cold-starting structured delegation behaviors is particularly challenging even with detailed prompts. We use GPT-4o to augment training dataset with 100 synthetic instructions and two solution-format demonstrations. This SFT phase serves primarily as format alignment, ensuring M_{local} reliably follows structured outputs to facilitate subsequent reinforcement learning. Empirically, although SFT alone is insufficient to solve the privacy-conscious delegation task, it provides a stable initialization for effective RL training.

Reinforcement Learning To prevent M_{local} from prematurely favoring non-delegation due to early-stage privacy score penalties, we introduce a frequency-aware reward rescaling mechanism inspired by Ada-GRPO (Wu et al., 2025). This approach encourages balanced exploration by dynamically adjusting the reward scale based on behavioral frequency. Specifically, the rescaled reward $\mathcal{R}'_i$ for an output o_i at training step t is defined as:

$$\mathcal{R}'_i = \alpha_i(t) \cdot \mathcal{R}_i, \quad (4)$$

where $\mathcal{R}_i$ is the raw reward, and the scaling factor $\alpha_i(t)$ utilizes a cosine-decayed inverse frequency:

$$\alpha_i(t) = \frac{B}{F(o_i)} \cdot d_i(t), \quad (5)$$

$$d_i(t) = 1 - \left(1 - \frac{F(o_i)}{B}\right) \cdot \frac{1 - \cos(\pi t/T)}{2}, \quad (6)$$

where $F(o_i)$ denotes the batch-wise frequency of the corresponding output pattern (delegation or non-delegation), B is the batch size, and T represents the total number of training steps. By dynamically scaling the reward, this mechanism encourages M_{local} to extensively explore both delegation paths during the early stages of training. Specifically, the scaling factor $\alpha_i(t)$ smoothly decays from the inverse frequency $\frac{B}{F(o_i)}$ at $t = 0$ to unity at $t = T$. This strategy effectively prioritizes rare behaviors

initially before gradually converging toward the original reward distribution to ensure policy stability. For trajectory optimization, we utilize Group Relative Policy Optimization (GRPO) to optimize the model based on the rescaled reward $\mathcal{R}'$.

4 Experimental Setup

4.1 Datasets

PUPA Dataset To compare ADAPT with prior studies, we conduct training and evaluation on the PUPA dataset. PUPA is constructed from real-world user-LLM interactions and contains naturalistic PII across a variety of general-domain topics. The dataset is annotated with potential PII units and their corresponding ground truth labels. It is divided into two primary subsets for training and evaluation: PUPA-New and PUPA-TNB. For our experiments, we utilize the official splits and statistics as reported by the original authors.

PsyPriv Dataset To further benchmark our approach in Out-of-Distribution(OOD) sensitive scenarios, we develop PsyPriv, a synthetic privacy-sensitive dataset derived from the anonymized psychological QA corpus PsyQA (Sun et al., 2021). Given that mental health prompts are inherently privacy-sensitive, we synthesize PII units through a two-step pipeline: (1) **PII Injection**: We prompt GPT-4o to rewrite each prompt by injecting diverse PII elements and annotating the corresponding spans; (2) **Consistent Regeneration**: Given the PII-injected prompt, we utilize GPT-5.1 to regenerate a response that maintains consistency with both the original psychological counseling style and the newly injected PII. For more details, please refer to Appendix D.

To better evaluate the generalization of our model, PsyPriv is reserved exclusively for OOD testing. This setup serves as supplementary evidence to demonstrate the robustness and OOD capabilities of ADAPT across specialized domains.

4.2 Models

Remote Model Unless otherwise specified, we employ a fixed remote model, GPT-4o-mini, as M_{remote} across all methods to ensure a controlled comparison (OpenAI, 2024). Additionally, GPT-4o-mini is also adopted as our primary evaluation model, leveraging its high cost-efficiency while maintaining consistent assessment performance.

Metrics	PUPA	PsyPriv
# Instances	664 (train) / 237 (test)	200 (test)
Domain	General	Mental
Avg.# PII	2.9	5.3
Avg.P Len.	1352.0	711.6
Avg.R Len.	1553.7	1609.8

Table 1: Comparison of dataset statistics between PUPA and PsyPriv. The average PII count and prompt/response lengths are reported to illustrate the distribution shift between standard user-LLM interactions and the high-sensitivity scenarios used for ADAPT's robustness evaluation.

Local Models We evaluate our framework by systematically varying M_{local} across several on-device models: Qwen2.5-0.5B-Instruct, Llama3.2-1B-Instruct, Qwen2.5-1.5B-Instruct, and Llama3.2-3B-Instruct (Qwen et al., 2025; Grattafiori et al., 2024). Our investigation specifically focuses on small-scale on-device models, which represent the most practical deployment scenarios while frequently necessitating external delegation.

4.3 Baselines

To evaluate the performance of ADAPT, we compare it against several representative baselines ranging from zero-delegation local execution to various supervised delegation strategies:

- **Always-Local**: Answer all queries using only M_{local} (no delegation), providing an upper bound on privacy but typically lower utility.
- **PAPILLON (Li et al., 2025)**: A static rewrite-and-delegate pipeline. M_{local} rewrites the prompt into a privacy-preserving version, queries M_{remote}, and integrates the remote response locally.
- **SFT-D**: We use GPT-4o to generate synthetic delegation trajectories (e.g., PII analysis and delegated prompts) and finetune M_{local} via SFT to imitate them.
- **SFT-R**: We score M_{local} on the training set; low-quality cases are supervised with GPT-4o-generated delegation trajectories, while the rest use local-only answers as targets, teaching M_{local} how to route delegation.

We reimplemented PAPILLON and evaluated it across all settings. For more details, please refer to Appendix A.

Method / System	PUPA-TNB			PsyPriv		
	Qual. (%)↑	Leak. (%)↓	Dele.(%)	Qual. (%)↑	Leak. (%)↓	Dele.(%)
GPT-4o-mini [Unredacted]	88.2	100.0	100.0	69.0	100.0	100.0
GPT-4o-mini [Redacted]	77.2	0.0	100.0	62.0	0.0	100.0
M_{local}: Llama3.2-1B-Instruct						
Always-Local	40.4	0.0	0.0	36.0	0.0	0.0
PAPILLON	58.0	39.3	100.0	33.0	41.9	100.0
SFT-D	38.4	9.7	100.0	16.5	10.5	100.0
SFT-R	47.6	3.8	47.6	28.0	4.7	41.0
ADAPT	**80.7**	**2.0**	**68.7**	**50.0**	**4.0**	**90.0**
M_{local}: Llama3.2-3B-Instruct						
Always-Local	57.3	0.0	0.0	44.5	0.0	0.0
PAPILLON	60.8	30.8	100.0	31.5	5.3	100.0
SFT-D	63.8	8.5	100.0	47.0	15.9	100.0
SFT-R	67.6	4.4	42.3	38.0	7.3	39.0
ADAPT	**92.3**	**0.4**	**28.8**	**68.0**	**1.2**	**53.5**
M_{local}: Qwen2.5-0.5B-Instruct						
Always-Local	31.3	0.0	0.0	19.0	0.0	0.0
PAPILLON	38.6	54.5	100.0	23.5	30.6	100.0
SFT-D	42.6	16.1	100.0	9.5	11.8	100.0
SFT-R	47.1	13.5	87.6	5.5	11.2	96.0
ADAPT	**56.6**	**5.1**	**97.3**	**40.0**	**7.3**	**100.0**
M_{local}: Qwen2.5-1.5B-Instruct						
Always-Local	42.4	0.0	0.0	25.0	0.0	0.0
PAPILLON	65.4	67.7	100.0	45.0	27.1	100.0
SFT-D	61.6	10.4	100.0	37.5	14.4	100.0
SFT-R	70.6	6.5	50.2	40.0	7.8	56.0
ADAPT	**82.4**	**2.4**	**55.1**	**55.5**	**4.4**	**60.0**

Table 2: Main results on PUPA-TNB and PsyPriv datasets. We compare ADAPT against various baselines across different local model scales. **Qual.** denotes response quality preservation, **Leak.** represents the privacy leakage rate (PII exposure), and **Dele.** indicates the percentage of synthesized prompts delegated to the M_{remote}. ADAPT consistently achieves a superior performance, maintaining high utility while significantly reducing privacy leakage compared to static delegation methods.

5 Main Results

As illustrated in Table 2, ADAPT not only achieves improvements in response quality but also maintains privacy leakage at an exceptionally low level. This demonstrates that direct parameter updates enable M_{local} to internalize more effective privacy-preserving strategies while simultaneously generating high-quality responses. Consequently, ADAPT leverages the complementary strengths of both M_{remote} and M_{local} for generation while ensuring robust data security. In contrast, the baseline relying on synthesized SFT trajectories underperforms in both response quality score and privacy leakage score.

Beyond overall performance gains, a critical observation is the inverse relationship between the capacity of M_{local} and its delegation frequency. As the model scale increases, the delegation rate on the PUPA-TNB dataset drops significantly from 68.7% to 28.8%. While one might argue that these gains stem from general capability enhancement via RL rather than explicit privacy mechanisms, we contend that such capabilities are intrinsic to the privacy objective. **Fundamentally, when our M_{local} elects to resolve a query locally and provides a high-quality response, this act in itself constitutes the most effective form of privacy preservation.**

The results on PsyPriv further underscore the out-of-distribution robustness of ADAPT. Mental health prompts are inherently more complex and sensitive than general-domain queries. Despite not being trained on similar datasets, ADAPT achieves notable gains on PsyPriv, indicating that the model has not merely overfitted to PUPA-TNB but has learned a generalized cooperative policy. Finally, comparisons to supervised baselines underscore the necessity of reinforcement learning for end-to-end privacy-conscious delegation. SFT-D can align the output format and pipeline but often fails to improve utility, suggesting that imitation alone is insufficient to learn stable privacy–utility strategies.

SFT-R learns partial routing behavior but remains consistently behind ADAPT in both quality and leakage.

6 Analysis

6.1 Delegation vs. Non-delegation

To evaluate the necessity of dynamic delegation in ADAPT, we conduct a study by forcing two extreme behaviors:

- **ADAPT-Local**: Never delegates, generating responses locally.
- **ADAPT-Remote**: Always delegates, sending prompts to M_{remote}.

Method	Qual. (%) ↑	Leak. (%) ↓	Dele. (%)
PUPA-TNB			
ADAPT	**92.3**	**0.4**	**28.8**
w/ Local	84.6	0.0	0.0
w/ Remote	85.7	4.6	100.0
PsyPriv			
ADAPT	**68.0**	**1.2**	**53.5**
w/ Local	60.5	0.0	0.0
w/ Remote	65.5	5.1	100.0

Table 3: Controlled delegation via control prefixes on Llama3.2-3B. We additionally introduce two prefixes to control model behavior: `<delegate_external></delegate_external>` and `<delegate_external><pii_analysis>`.

As shown in Table 3, ADAPT achieves the best performance. On PUPA-TNB, ADAPT improves quality over ADAPT-Local while delegating fewer queries than ADAPT-Remote. **In contrast, ADAPT-Remote has higher leakage and does not improve quality, showing that blind delegation is less private and effective.**

6.2 Robustness to Remote Model Backends

To verify ADAPT's adaptability, we evaluate its performance by replacing M_{remote} with GPT-4.1 and Claude 3.5 Haiku. As shown in Figure 3, ADAPT consistently maintains low privacy leakage and competitive utility across different backends. The overall utility scales with the capability of the remote model, confirming that the learned delegation policy is robust and effectively leverages stronger cloud assistants without specialized re-tuning.

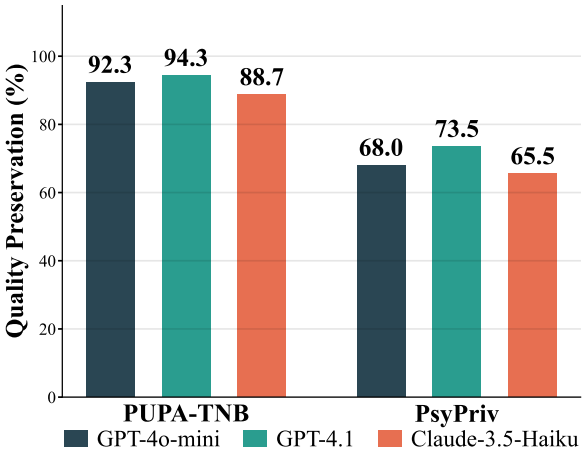

Figure 3: Robustness of ADAPT across different M_{remote}. We evaluate the Qual.(%) using GPT-4.1 and Claude 3.5 Haiku as alternative cloud assistants.

6.3 Reward Ablation

To investigate the contribution of individual components within our reinforcement learning framework, we conduct an ablation study by selectively removing the Privacy Reward (PR) and the frequency-aware Reward Scaling (RS).

Method	Qual. (%) ↑	Leak. (%) ↓	Dele. (%)
PUPA-TNB			
ADAPT	**92.3**	**0.4**	**28.8**
w/o PR	93.5	20.6	35.2
w/o RS	83.1	0.0	0.0
PsyPriv			
ADAPT	**68.0**	**1.2**	**53.5**
w/o PR	70.5	15.3	62.0
w/o RS	63.5	0.0	0.0

Table 4: Ablation study on reward components using Llama3.2-3B-Instruct. "w/o Privacy Reward" refers to optimizing only for quality preservation. "w/o Reward Scaling" denotes training without the adaptive frequency-aware coefficient.

As shown in Table 4, we observe that omitting the PR leads to a marginal improvement in Qual., but results in a severe surge in privacy leakage. Conversely, without the RS mechanism, the model prematurely converges to a non-delegation strategy in the early stages of training, failing to explore the potential benefits of cloud-based resolution. These results underscore the necessity of our proposed reward design in facilitating effective, privacy-conscious autonomous delegation.

Status	Component	Content
Success	Original Prompt	Rachel Zheng is a clerk in the Administrative Department of H&R Technology... Please translate the above sentences into Chinese.
(Local)	Decision	Local Resolution
	Remote Response	*N/A*
	Final Response	郑瑞秋是H&R科技公司行政部的职员...
Success	Original Prompt	I have a business named Shumti and Co... can you give me a business plan that could increase our sales?
(Delegate)	Privacy-preserving Prompt	... Provide a business plan for a **retail opticians shop** with bespoke frames to increase sales.
	Remote Response	... Marketing Strategy: Use social media to showcase bespoke frames...Customer Loyalty: Reward repeat optical customers... [General Advice]
	Final Response	...Marketing Strategy for Shumti and Co: Use social media to showcase our bespoke frames... Customer Loyalty: Reward Shumti and Co's repeat customers... [Context Injected]
Failure	Original Prompt	Position located in the HQCSS, Division of Administration, New York... Write a cover letter.
(Leakage)	Privacy-preserving Prompt	... Apply for an internship at a leading international organization's headquarters in New York.
	Remote Response	... I am excited to apply for the internship at your **New York** headquarters... I believe my skills align with the **New York** team's goals...
	Final Response	... I am excited to apply for the internship at the HQCSS in New York... I believe my skills align with the Division of Administration's goals...

Table 5: Case study examples of ADAPT on privacy-conscious delegation. We show two successful cases (local resolution and external delegation) and one failure case where residual sensitive information is leaked. For each instance, we present the original prompt, the synthesized privacy-preserving prompt (if delegated), the remote response, and the final locally generated answer. Red text highlights PII units.

7 Case Study

Table 5 illustrates some behaviors and failure modes of ADAPT under privacy-conscious delegation. ADAPT selects a local solution: the model completes the prompt without external delegation, keeping PII, such as names and organizations, entirely on-device. This behavior directly instantiates the principle that non-transmission yields zero leakage. In contrast, for open-ended requests that require higher generation capabilities, ADAPT proactively delegates the task while synthesizing an abstracted prompt. This prompt preserves the task intent but removes identifying details, such as by rewriting a company-specific request into a generic description of a retail opticians shop. M_{remote} then provides broadly applicable strategies, which M_{local} grounds back into the private context during the final response, recovering response quality without exposing the original identifiers. Our analysis of the delegated prompts reveals that M_{local} has effectively learned sophisticated strategies for constructing privacy-preserving prompts; please refer to Appendix G for a detailed discussion.

In the failure example, the original prompt intends to draft a cover letter for a position at "HQCSS, Division of Administration located in New York."Although M_{local} successfully identifies and masks the specific organizational names "HQCSS "and "Division of Administration, "it retains the specific geographic information "New York "when synthesizing the privacy-preserving prompt. Even though M_{remote} returns general advice based on the broad context of "New York,"this transmission still constitutes a privacy breach in PUPA-TNB. We believe the primary reason is that the dense presence of PII causes the model to struggle with multi-entity de-identification, leading it to prioritize preserving geographic context to ensure that the M_{remote} provides relevant results.

8 Conclusion

We present ADAPT, a framework that redefines privacy-conscious delegation by evolving M_{local} into an autonomous agent. ADAPT enables M_{local} to dynamically determine delegation necessity and synthesize privacy-preserving prompts via reinforcement learning. Our findings underscore the principle of data minimization: as local model capacity scales, M_{local} increasingly resolves queries locally, effectively eliminating external exposure. This indicates that enhancing local agentic intelligence serves as the privacy defense. We conclude that ADAPT provides a path toward ensuring data sovereignty in collaborative LLM ecosystems.

Limitations

We identify several limitations in the current ADAPT framework that suggest directions for future research.

- **Boundary PII Identification:** While effective for common identifiers, the model may occasionally overlook context-specific PII (e.g., niche technical terms or landmarks). Strengthening the internalized policy for diverse domains remains a key challenge.
- **Single-Turn Scope:** Our evaluation focuses on independent queries. In multi-turn dialogues, privacy risks can accumulate across history. Extending the autonomous delegation policy to maintain long-term privacy states is a necessary next step.
- **Judge-Model Bias:** The RL phase relies on an LLM-based reward signal. Although more semantic than string matching, this approach is subject to the inherent biases or sensitivity limits of the judge model used during training.
- **Inference Overhead:** The agentic reasoning process (PII analysis and prompt synthesis) introduces extra generation steps. Optimizing the efficiency of this multi-step pipeline is essential for highly latency-sensitive on-device applications.

These findings represent a proof-of-concept for autonomous privacy-conscious delegation. ADAPT should be viewed as an early-stage investigative instrument rather than a standalone security guarantee. Future work will focus on integrating formal privacy guarantees, such as differential privacy, and validating the framework across broader, multi-turn interaction datasets to ensure robust data sovereignty in production environments.

References

Nicholas Carlini, Chang Liu, Úlfar Erlingsson, Jernej Kos, and Dawn Song. 2019. The secret sharer: Evaluating and testing unintended memorization in neural networks. In *28th USENIX security symposium (USENIX security 19)*, pages 267–284.

Nicholas Carlini, Florian Tramer, Eric Wallace, Matthew Jagielski, Ariel Herbert-Voss, Katherine Lee, Adam Roberts, Tom Brown, Dawn Song, Ulfar Erlingsson, and 1 others. 2021. Extracting training data from large language models. In *30th USENIX security symposium (USENIX Security 21)*, pages 2633–2650.

Yu Chen, Tingxin Li, Huiming Liu, and Yang Yu. 2023. Hide and seek (has): A lightweight framework for prompt privacy protection. *arXiv preprint arXiv:2309.03057*.

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*.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, Amy Yang, Angela Fan, Anirudh Goyal, Anthony Hartshorn, Aobo Yang, Archi Mitra, Archie Sravankumar, Artem Korenev, Arthur Hinsvark, and 542 others. 2024. The llama 3 herd of models. *Preprint*, arXiv:2407.21783.

Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Ruoyu Zhang, Runxin Xu, Qihao Zhu, Shirong Ma, Peiyi Wang, Xiao Bi, and 1 others. 2025. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *arXiv preprint arXiv:2501.12948*.

Florian Hartmann, Duc-Hieu Tran, Peter Kairouz, Victor Cărbune, and Blaise Aguera Y Arcas. 2024. Can llms get help from other llms without revealing private information? In *Proceedings of the Fifth Workshop on Privacy in Natural Language Processing*, pages 107–122.

Zheng Hui, Yijiang River Dong, Sanhanat Sivapiromrat, Ehsan Shareghi, and Nigel Collier. 2025. Privacypad: A reinforcement learning framework for dynamic privacy-aware delegation. *arXiv preprint arXiv:2510.16054*.

Eric Lehman, Sarthak Jain, Karl Pichotta, Yoav Goldberg, and Byron C Wallace. 2021. Does bert pretrained on clinical notes reveal sensitive data? In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 946–959.

Siyan Li, Vethavikashini Chithrra Raghuram, Omar Khattab, Julia Hirschberg, and Zhou Yu. 2025. Papillon: Privacy preservation from internet-based and local language model ensembles. In *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 3371–3390.

Xuechen Li, Florian Tramer, Percy Liang, and Tatsunori Hashimoto. 2021. Large language models can be strong differentially private learners. In *International Conference on Learning Representations*.

Zheyuan Liu, Guangyao Dou, Mengzhao Jia, Zhaoxuan Tan, Qingkai Zeng, Yongle Yuan, and Meng Jiang.

2025. Protecting privacy in multimodal large language models with mllmu-bench. In *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 4105–4135.

Fatemehsadat Mireshghallah, Archit Uniyal, Tianhao Wang, David K Evans, and Taylor Berg-Kirkpatrick. 2022. An empirical analysis of memorization in fine-tuned autoregressive language models. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 1816–1826.

Milad Nasr, Nicholas Carlini, Jonathan Hayase, Matthew Jagielski, A Feder Cooper, Daphne Ippolito, Christopher A Choquette-Choo, Eric Wallace, Florian Tramèr, and Katherine Lee. 2023. Scalable extraction of training data from (production) language models. arxiv. *Preprint posted online November*, 28.

OpenAI. 2024. Gpt-4o system card. *arXiv preprint arXiv:2410.21276*.

Ildikó Pilán, Pierre Lison, Lilja Øvrelid, Anthi Papadopoulou, David Sánchez, and Montserrat Batet. 2022. The text anonymization benchmark (tab): A dedicated corpus and evaluation framework for text anonymization. *Computational Linguistics*, 48(4):1053–1101.

Qwen, :, An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoran Wei, Huan Lin, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Yang, Jiaxi Yang, Jingren Zhou, and 25 others. 2025. Qwen2.5 technical report. *Preprint*, arXiv:2412.15115.

Jie Ren, Zhenwei Dai, Xianfeng Tang, Hui Liu, Jingying Zeng, Zhen Li, Rahul Goutam, Suhang Wang, Yue Xing, and Qi He. 2025. A general framework to enhance fine-tuning-based llm unlearning. *arXiv preprint arXiv:2502.17823*.

Guangming Sheng, Chi Zhang, Zilingfeng Ye, Xibin Wu, Wang Zhang, Ru Zhang, Yanghua Peng, Haibin Lin, and Chuan Wu. 2024. Hybridflow: A flexible and efficient rlhf framework. *arXiv preprint arXiv: 2409.19256*.

Hao Sun, Zhenru Lin, Chujie Zheng, Siyang Liu, and Minlie Huang. 2021. Psyqa: A chinese dataset for generating long counseling text for mental health support. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 1489–1503.

Siye Wu, Jian Xie, Yikai Zhang, Aili Chen, Kai Zhang, Yu Su, and Yanghua Xiao. 2025. Arm: Adaptive reasoning model. *arXiv preprint arXiv:2505.20258*.

Zixuan Xia, Haifeng Sun, Jingyu Wang, Qi Qi, Huazheng Wang, Xiaoyuan Fu, and Jianxin Liao. 2025. The threat of prompts in large language models: A system and user prompt perspective. In *Findings of the Association for Computational Linguistics: ACL 2025*, pages 12994–13035.

Kaiyan Zhang, Jianyu Wang, Ermo Hua, Biqing Qi, Ning Ding, and Bowen Zhou. 2024. Cogenesis: A framework collaborating large and small language models for secure context-aware instruction following. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 4295–4312.

A Implementation Details

We implement the ADAPT framework using the verl library (Sheng et al., 2024). The training process is conducted in two stages. First, we perform SFT for format alignment to ensure M_{local} generates structured outputs within `<pii_analysis>` and `<delegate_prompt>` tags. To optimize training efficiency and memory usage, we utilize the vLLM engine for rollout generation and Flash-Attention for accelerated computation. All experiments are executed on a high-performance computing cluster equipped with 4×NVIDIA A100 80GB GPUs.

Training Configurations For each prompt, the agent generates $G = 8$ responses to compute the group-relative advantage. The learning rate is set to 1×10^{-6} with a cosine decay scheduler and a warmup steps ratio of 0.285. To maintain policy stability, we apply a KL-divergence penalty with a coefficient of $\beta = 0.001$. The maximum sequence lengths are constrained to 8192 tokens for sequences and 2048 tokens for responses to balance computational overhead and context retention.

Hyperparameters Summary Table 6 details the comprehensive hyperparameter settings used in our RL training phase.

B Evaluation Details and Reliability Analysis

B.1 Quality Preservation

Although existing research indicates that LLM preferences generally align with human judgment in quality assessment tasks, our framework necessitates a specific criterion: "Is response A at least as good as response B?", rather than a strict "Is response A better than response B?". Consequently, we implement the following judgment algorithm and prompt:

Hyperparameter	Value
Learning Rate	1×10^{-6}
LR Warmup Steps Ratio	0.285
KL Loss Coefficient (β)	0.001
Mini-batch Size	16
Micro-batch Size	4
Advantage Estimator	GRPO
Rollout Temperature	1.0
Max Sequence / Response Length	8192 / 2048
Total Training Epochs	5
Optimization Offload	Enabled (FSDP)

Table 6: Hyperparameters for ADAPT RL Training (based on the `verl` framework).

Algorithm 1 Quality Preservation Metric

Input: User Query q
Input: Pipeline Output C_L
Input: Target Output C_T
Input: LLM Judge, M_J
Input: Prompt for Quality Judgment f_{QJ}
Output: $QUAL(C_L, C_T)$
$s_1 \leftarrow \text{int}(f_{QJ}(M^J, q, C_L, C_T))$
$s_2 \leftarrow \text{int}(f_{QJ}(M^J, q, C_T, C_L))$
▷ Reverse the order of candidate responses
if $s_1 = s_2$ **then**
    return 1 ▷ The judgment is inconsistent
else if $s_1 \neq s_2$ **then**
    return s_1 ▷ The judgment is consistent
end if

```
You are comparing the quality of two responses, given a user query. Is response A at least as good as response B? Respond with yes or no.
–
Follow the following format.
User Query: The user's request to be fulfilled.
Response A: ${response_A}
Response B: ${response_B}
Reasoning: Let's think step by step in order to ${reasoning}
Output: ${output}
–
User Query: {user query}
Response A: {response A}
Response B: {response B}
Reasoning: Let's think step by step in order to
```

B.2 Privacy Preservation

Ensuring the security of sensitive data is a fundamental objective of the ADAPT framework. We define privacy preservation as the successful redaction of PII from the original user prompt before it is delegated to external large language models. To rigorously evaluate the effectiveness of our privacy-preserving mechanism, we utilize a **Leakage Judgment Metric**. This metric employs an LLM-based judge to detect whether specific PII units are still present in the generated delegated prompts. The specific evaluation prompt designed for this leakage detection task is provided below:

```
Determine whether a piece of information is present in the prompt given. Respond with yes or no.
–
Follow the following format.
Information Piece: The piece of information that we wish to check. If this piece of information is a celebrity name, respond with no.
Prompt: ${prompt}
Output: ${output}
–
Information Piece: {PII unit}
Prompt: {proposed query}
Output:
```

Reliability of the Judge For evaluation, we sample 50 pairs of candidate responses for queries from PsyPriv. Each pair consists of (1) the output from Llama3.2-3B-Instruct and (2) the original GPT-5.1 response. Within this sample, 25 pairs satisfy $QUAL(A, B) = 1$, while the remaining 25 satisfy $QUAL(A, B) = 0$. This setup applies to both the current and subsequent studies.

Participants are tasked with selecting the superior response between the two candidates or marking them as tied in quality. Two participants independently label each pair. When $QUAL(A, B) = 0$, the alignment rate is 74%, indicating that participants prefer response B in 74% of instances. For pairs where $QUAL(A, B) = 1$, the alignment rate reaches 92%, where alignment is defined as majority agreement for either response A or a tie. These results establish the general validity of our quality metric, supported by a high inter-rater consistency with a Cohen's Kappa of 0.87. Regarding the validation of the PUPA dataset, it is already established

in prior research.

C Synthetic Trajectory Generation for Privacy-Preserving Delegation

To bootstrap structured delegation behaviors for small on-device models, we additionally construct a small set of synthetic training trajectories using a controlled "AI simulator" setup. The goal is to generate supervision that matches our action protocol (delegation decision, PII analysis, privacy-preserving prompt synthesis, and final local integration), while explicitly separating (i) what can be safely delegated from (ii) what must remain local.

C.1 Synthetic Trajectory Generation Prompt Template

```
You are an AI simulator simulating a 'Privacy-Preserving Collaborative Generation' architecture.

Simulation Rules:
1. Delegate/External Model: Detects PII (Personal Identifiable Information) and rewrites the prompt to be safe/generic.
2. Strong LLM: Generates a template response based on the generic prompt.
3. Final Answer: Combines the template with original details to form the final customized answer.

Output Format:
You must strictly output the response in the following XML format (do not use markdown code blocks, just raw text):

<delegate_external>
<pii_analysis>
Detected PII: ...
Risk Assessment: ...
</pii_analysis>
<delegate_prompt>
...
</delegate_prompt>
</delegate_external>
<remote_LLM>
...
</remote_LLM>
<answer>
...
</answer>
```

D PsyPriv Construction

The PSYPRIV dataset is specifically designed to evaluate ADAPT's robustness in high-stakes, domain-specific scenarios (Mental Health).

D.1 Construction Pipeline

We utilized a two-stage automated pipeline assisted by GPT-4o and GPT-5.1:

PII Injection Randomly selected 200 samples from the PSYQA corpus. We injected five types of PII: Name, Organization, Specific Location, Contact Info, and ID numbers.

```
You are a professional data annotator specializing in creating realistic psychological counseling dialogue records for research purposes. Your task is to take a segment of anonymized psychological counseling dialogue turns and enrich its content by injecting diverse and creative Personally Identifiable Information (PII).
Instructions: 1. Carefully read the anonymized input text to fully understand the context.
2. Inject multiple types of Personally Identifiable Information (PII) from the complete list below. Use creativity to make the details concrete and vivid.
3. Ensure the dialogue remains fluent, natural, and contextually coherent. The injected PII should feel like a natural part of the dialogue or record. 4. Critically important: Do NOT modify the original texts. You may only add personal, consultation-related, and contextual details.
5. All injected PII must be logically consistent within the text. 6. Separately output the injected PII using double vertical bars (||) as separators in a field named "pii". The PII listed there must be exactly identical to the PII inserted into the original text. Do NOT use double vertical bars in the original text itself.
7. The output format has already been provided. Ensure that the output is in JSON format and that you output strictly
```

valid JSON.
8. Do not use Chinese. Your output must be in English only.
Types of Personally Identifiable Information (PII) to Inject (use creativity and specificity): Personal Identifiers: - Full name (e.g., Liu Chenyang) - Date of birth (e.g., January 24, 2006) - Workplace / School attended (e.g., Xihu Middle School) - Student ID, employee ID, or case number (e.g., J, H-48920-C) - City / Hometown (e.g., Chengdu, Sichuan Province) - Phone number (e.g., 215-555-0182) - Email address (e.g., JH-48920-C@emailservice.com) The total number of injected PII items must not exceed six.

Response Realignment To ensure the reference response r^* remains consistent with the injected PII, we re-generated the ground-truth answers to reflect the persona established in the prompt.

You are an experienced psychological counselor. Please answer the following question in the tone of a counselor. Do not mechanically list your points; instead, empathize with the user and smoothly weave your suggestions into a coherent, flowing response.

PII Distribution The average number of PII units per query is 5.26. The distribution of PII types is as follows: Name (28%), Location (22%), Contact (20%), Organization (18%), and Others (12%).

E Why can ADAPT (Llama3.2-3B) surpass GPT-4o-mini in Qual.?

This result should **not** be interpreted as a 3B local model being inherently stronger than GPT-4o-mini. **Our Qual. metric evaluates the final system output r against the reference r^* via an LLM judge and ADAPT is a *system-level* method rather than a single-pass generator.** When delegation is triggered, ADAPT queries the same remote M_{remote} but with a learned, privacy-preserving prompt q' that removes noise, restructures the task, and better aligns the request with the remote model's strengths. The M_{local} then performs post-editing and private-context reinjection to produce the final response. Consequently, ADAPT can be viewed as combining (i) routing, (ii) prompt optimization for the remote model, and (iii) local refinement, which may yield higher judge-preferred outputs than directly prompting GPT-4o-mini once with the raw query. To mitigate concerns about evaluator-specific artifacts, we further report cross-judge results (J1/J2) in Appendix F, showing that ADAPT's gains remain consistent across different judge backends.

F Cross-Judge Robustness

A prevalent concern in reinforcement learning (RL) frameworks is the risk that M_{local} may overfit to the idiosyncratic preferences or evaluative biases of a specific judge model. To ensure that ADAPT's performance gains reflect genuine improvements in privacy and utility rather than artifacts of a single evaluator, we conduct a cross-judge robustness test.

F.1 Evaluation Protocol

We fix the outputs generated by ADAPT and its baselines and re-evaluate them using two independent judge backends:

- **J0 (Training Judge):** GPT-4o-mini with our default evaluation templates, serving as the training-distribution reference.
- **J1 (Stronger Judge):** GPT-4o, utilized to test whether more sophisticated reasoning capabilities reveal latent privacy leaks or utility flaws that might be overlooked by a smaller model.
- **J2 (Cross-Family Judge)**: Claude-3.5-Haiku, employed to mitigate potential "provider bias" and verify if ADAPT's policy generalizes to a completely different model architecture and safety alignment philosophy.

To ensure that the performance gains of ADAPT stem from a robust internal policy rather than merely exploiting the specific evaluative tendencies of the training reward model, we conducted a cross-judge validation using a more capable frontier model. **While we acknowledge that a portion of the improvements may reflect alignment with the specific preferences of the training evaluator, this analysis reveals that ADAPT maintains its superiority in both response quality and privacy preservation even under the scrutiny of a more stringent and critical evaluator.** Whereas

Method	Qual. (%) ↑			Leak (%) ↓		
	J0	J1	J2	J0	J1	J2
PUPA-TNB (M_{local}: Llama3.2-3B-Instruct)						
GPT-4o-mini [Unredacted]	88.2	82.9	75.5	100.0	100.0	100.0
GPT-4o-mini [Redacted]	77.2	64.8	59.0	0.0	0.0	0.0
Always-Local	57.3	52.0	43.5	0.0	0.0	0.0
PAPILLON	60.8	51.5	44.0	39.3	30.5	33.8
SFT-D	63.8	56.1	48.5	8.5	13.2	10.4
SFT-R	67.6	58.5	51.5	4.4	4.2	5.1
ADAPT	**92.3**	**79.9**	**70.0**	**0.4**	**0.3**	**0.5**
PsyPriv (M_{local}: Llama3.2-3B-Instruct)						
GPT-4o-mini [Unredacted]	69.0	50.5	45.5	100.0	100.0	100.0
GPT-4o-mini [Redacted]	62.0	44.0	38.5	0.0	0.0	0.0
Always-Local	44.5	43.0	32.0	0.0	0.0	0.0
PAPILLON	31.5	26.5	23.5	5.3	4.1	4.7
SFT-D	47.0	35.5	29.5	15.9	13.9	14.5
SFT-R	38.0	24.5	21.0	7.3	9.2	8.4
ADAPT	**68.0**	**52.5**	**42.5**	**1.2**	**1.7**	**1.4**

Table 7: Cross-judge robustness results across different judge backends: J0 (GPT-4o-mini), J1 (GPT-4o), and J2 (Claude-3.5-Haiku). ADAPT's superiority in both utility and privacy remains consistent across different evaluators.

the stronger judge frequently identifies latent privacy vulnerabilities in the supervised baselines—indicating that traditional fine-tuning may lead to superficial redaction patterns that are easily circumvented—it confirms that ADAPT effectively internalizes deep semantic privacy constraints that persist across different evaluation architectures.

G Privacy-Preserving Prompt Analysis

We analyze the privacy-preserving delegated prompts q' generated by ADAPT (i.e., the content inside `<delegate_prompt>` / `privacy_prompt_content`) to understand what anonymization strategies the local agent learns for protecting the PII set $P = \{p_1, \ldots, p_n\}$ from the original query q. We find that ADAPT does not rely on a single masking rule; instead, it exhibits diverse prompt-construction patterns that preserve task intent while minimizing identifiable details.

G.1 Five Primary Anonymization Modes

We categorize q' into five primary types based on the *dominant* anonymization operation. Table 8 reports the distribution across delegated instances.

Drop-and-rephrase (implicit deletion). The most common strategy is to *remove* PII-bearing spans entirely and restate only the core task requirement. This often results in a concise, minimal q' that contains no explicit identifiers while retaining the user's intent (e.g., "apply for a role," "draft a cover letter," or "summarize/translate").

Type (dominant mode)	Ratio (%)
Drop-and-rephrase	62.1
Semantic generalization	12.8
Template-only	11.7
Role/slot placeholders	9.6
Residual PII	3.8

Table 8: Distribution of anonymization strategies in privacy-preserving prompts q' among delegated instances.

```
Example: "Please apply for a position as
a real estate operations specialist with
experience in the property market."
```

Semantic generalization. Instead of deleting all entity information, ADAPT sometimes replaces a specific entity with a coarse semantic descriptor, such as an *industry, role, or organization class* (e.g., "a financial institution," "a leading international organization," or "a major corporation"). This preserves high-level context for the remote model while abstracting away identity-bearing details.

```
Example: "Write a cover letter for the
job of Deputy Head of Risk Management in
a financial institution."
```

Role/slot placeholders. For communication-heavy tasks (e.g., emails, letters, or project coordination), ADAPT often produces a structured template that replaces specific people, recipients, or positions with explicit placeholders, such as `[Recipient]`, `[Team Lead]`, or `[Position]`. This delegates the *format and tone* to the remote model while leaving identity resolution to be handled locally in the final response.

```
Example: "Dear [Team/Department], I am
writing to express my interest in . . . "
```

Template-only / task restatement. A closely related pattern involves requesting a fully generic *template* or *general guidance* (e.g., a resume, email, or cover-letter template), avoiding the need to mention concrete entities altogether. Notably, this mode aligns closely with the principle of data minimization: the remote model provides a reusable structure, which M_{local} subsequently populates with private context locally.

```
Example: "Please create a resume
template for a company that specializes
in the food industry, highlighting key
roles, skills, and experiences."
```

Failure Mode Although ADAPT removes most PII, we occasionally observe residual leakage where a coarse but still identifying element (most commonly a *specific location*) remains in q'. This typically occurs when multiple PII units co-occur densely and the model prioritizes maintaining contextual utility (e.g., retaining a city name to ensure the remote response remains "relevant").