LS-Guard: Adaptive Safety Guardrails Tailored to Individual LLMs

Jinggui Liang, Lizi Liao
School of Computing and Information Systems,
Singapore Management University
jg.liang.2023@phdcs.smu.edu.sg, lzliao@smu.edu.sg

Abstract

Large Language Models (LLMs) excel at diverse tasks, but remain vulnerable to malicious inputs such as jailbreak attacks. Current one-size-fits-all safety guardrails built from static datasets ignore each model's unique safety profile and often force trade-offs between safety and utility. To address this gap, we propose LS-Guard, a framework for learning model-specific guardrails tailored to each LLM's vulnerabilities. LS-Guard operates in two stages: First, it dynamically profiles a given LLM by probing it with malicious prompts to elicit the model's responses, which are then dynamically labeled to reveal model-specific failure modes. Second, it uses this data to train a safety classifier with a collaborative multi-LoRA architecture. An orthogonality-constrained multi-task loss enables a central expert to learn general safety features while each subject-specific expert encodes the distinctive vulnerability patterns of one LLM. During inference, LS-Guard activates the central expert together with its model-specific expert to perform content moderation, yielding reliable safety decisions. Extensive experiments on multiple real-world LLMs demonstrate that LS-Guard significantly outperforms strong baseline guardrails, achieving superior robustness, adaptability, and generalization.[1]

WARNING: This paper may contain offensive or sensitive content.

1 Introduction

Large Language Models (LLMs) (OpenAI, 2023; Dubey et al., 2024; Comanici et al., 2025) have demonstrated exceptional capabilities across various tasks, driving their rapid integration into a wide range of user-facing commercial applications (Zheng et al., 2023; Nie et al., 2024; Chu et al., 2025). However, despite these remarkable advancements, LLMs have also raised critical safety concerns, particularly their susceptibility to malicious manipulation capable of bypassing built-in safety restrictions and eliciting harmful responses (Dong et al., 2024; Kuo et al., 2025).

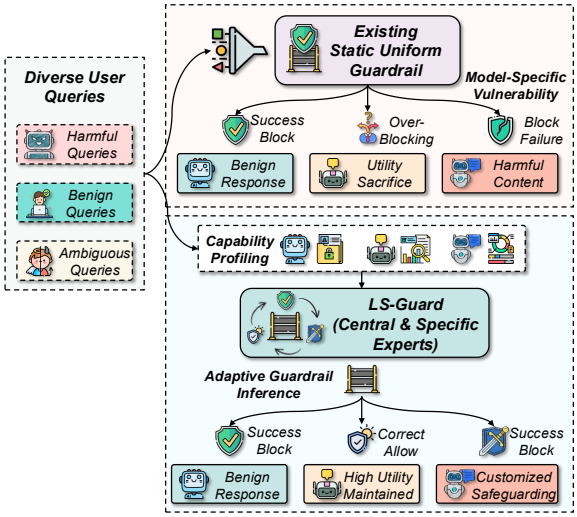

Figure 1: A comparison of existing guardrails and the proposed LS-Guard framework.

To this end, recent years have seen a surge in attempts to protect LLMs from generating harmful content. Existing approaches for defending LLMs typically fall into two broad categories: alignment-based methods and input/output guardrails (Dong et al., 2024; Ghosh et al., 2025). Alignment-based defenses involve further tuning pre-trained LLMs to strengthen their internal safety capabilities via RLHF or SFT techniques (Ji et al., 2023a, 2025; Feng et al., 2024; Rong et al., 2025; Pant, 2025). Although effective at handling malicious queries, this process can incur a non-trivial alignment tax, inadvertently sacrificing the model's core functionalities (Huang et al., 2025; Chen et al., 2025). In contrast, input/output guardrails operate from an external perspective—they monitor user–LLM interactions and trigger appropriate mitigation mechanisms without modifying the underlying model weights or core capabilities—and have thereby be-

[1]Dataset and code are available at https://github.com/liangjinggui/LS-Guard

Findings of the Association for Computational Linguistics: ACL 2026, pages 19759–19772

come a critical paradigm for ensuring LLM safety (Chi et al., 2024; Wang et al., 2025).

Despite their success, we identify a critical bottleneck inherent in current input/output guardrails. Existing studies (Yuan et al., 2024; Verma et al., 2025) mainly adopt a standalone manner, optimizing guardrails on large, static malicious corpora and then uniformly deploying them to safeguard various subject LLMs. However, this one-model-fits-all strategy is often inadequate for comprehensively securing individual subject LLMs (Wu et al., 2025; Luo et al., 2025). As illustrated in Figure 1, some queries (*e.g.*, "self-harm") are uniformly toxic and should consistently trigger appropriate safety mechanisms, whereas others (*e.g.*, "money laundering") can exhibit varying risk levels across subject LLMs, thus demanding model-specific protection from guardrails. We refer to this as the "*model heterogeneous vulnerability*" phenomenon, which is determined not only by whether a query is harmful at the intention level, but also by the interactive effect between individual LLM profiles and current user queries. Consequently, strengthening uniform guardrail decisions is insufficient to reliably mitigate model heterogeneous vulnerabilities and improve subject-LLM safety.

Addressing these challenges, we propose **LS-Guard**, a novel and robust subject **L**LM-**S**pecific **Guard**rail framework that explicitly models the vulnerabilities and capabilities of different subject LLMs to enhance the real-world safeguarding process. The core principle of LS-Guard is to first capture the unique vulnerabilities of distinct subject LLMs and then incorporate them into "personalized" guardrail development. Specifically, LS-Guard operates in two stages. First, it dynamically profiles each subject LLM through probing with malicious inputs to derive model-specific safety labels. Second, it leverages these data within collaborative multi-LoRA experts trained with an orthogonality-based multi-task loss. Within this architecture, a central expert learns generalizable safety knowledge, while subject-specific experts encode distinctive vulnerability patterns. This design enables LS-Guard to generalize across diverse LLMs while adapting to the particularities of individual models, thereby achieving robust moderation with balanced safety–utility trade-offs. During inference, LS-Guard activates the central expert alongside the corresponding subject-specific expert to perform content moderation, producing robust and reliable safety decisions. Extensive experiments demonstrate that LS-Guard achieves consistent and significant improvements over diverse top-performing baselines, underscoring its robustness, adaptability, and generalizability.

To sum up, our contributions are threefold:

- We identify that the critical challenge in real-world guardrails lies in addressing distinct vulnerabilities through differentiated safeguarding, and we propose profiling subject LLM characteristics to enhance their safety.
- We introduce **LS-Guard**, a novel subject LLM-Specific Guardrail that employs collaborative multi-LoRA experts to capture the unique capabilities of distinct LLMs for robust safeguarding.
- We conduct comprehensive experiments demonstrating that LS-Guard achieves significant performance improvements over strong baselines, validating its robustness and generalizability.

2 Related Work

Safety Guardrail Models for LLMs. Research on guardrail models for real-world deployed subject LLMs can be broadly categorized into two types: rule-based filtering and model-based safety classification. For the former, early approaches (Welbl et al., 2021; Clarke et al., 2023; Gómez et al., 2024) primarily relied on predefined keyword lexicons and heuristic constraints to detect and restrict harmful content. For instance, Clarke et al. (2023) proposed an exemplar-based method for deriving logical rules that enable explainable and customizable text moderation. While these methods offer transparency and efficiency, their static design results in rigidity and limited adaptability to real-world safeguarding scenarios (Song et al., 2023; Paudel et al., 2023).

To address this, recent research has shifted toward model-based guardrails, which offer greater flexibility by leveraging large-scale models for content assessment. For instance, studies such as LLaMA Guard (Inan et al., 2023; Fedorov et al., 2024; Chi et al., 2024), Aegis Guard (Ghosh et al., 2024), and WildGuard (Han et al., 2024) fine-tune open-source language models on red-teaming datasets to enhance their safeguarding performance, enabling them to classify inputs according to predefined safety guidelines. In parallel, another line of work employs closed-source guardrail APIs developed by industrial providers, such as OpenAI (Markov et al., 2023) and Perspective (Lees et al., 2022), to provide scalable safeguarding solutions.

Despite the progress, existing guardrails predominantly operate in a "one-model-fits-all" manner, providing guardrail decisions without considering the intrinsic safety mechanisms of the guarded subject LLMs, which renders them impractical and prone to excessive trade-offs between safety and utility. In this work, we address this issue by systematically characterizing the unique vulnerabilities of individual subject LLMs and integrating them into a personalized guardrail framework.

Multi-LoRA Architecture. With the rapid rise of LLMs, Parameter-Efficient Fine-Tuning (PEFT) techniques (Houlsby et al., 2019; Sung et al., 2022; Zhou et al., 2024) have emerged as an effective approach for task-specific customization with lower computational cost. Among them, Low-Rank Adaptation (LoRA) (Hu et al., 2022; Xu et al., 2024) has garnered increasing interest recently, becoming a standard method for adapting LLMs. Recognizing its potential, researchers have delved deeper, exploring the benefits of employing multiple LoRAs (Wang et al., 2023; Sheng et al., 2023; Tian et al., 2024). For instance, drawing inspiration from transfer learning principles (He et al., 2022; Lv et al., 2023), LoraHub (Huang et al., 2023) adopts a multi-LoRA strategy by training multiple adapters and dynamically selecting optimal combinations based on domain relevance during inference. Similarly, MOELoRA (Liu et al., 2023) integrates a Mixture-of-Experts (MoE) architecture into LLMs, thereby enhancing their multitasking capability and adaptability within specific domains.

However, the application of multi-LoRA architectures to guardrails for deployed LLMs remains largely underexplored. In this work, we explore leveraging a collaborative multi-LoRA design to capture the vulnerability patterns of different subject LLMs, enabling more effective and personalized safeguarding in real-world deployments.

3 Methodology

3.1 Problem Formulation

In this work, we formalize the guardrail problem as follows: Let $x \in \mathcal{X}$ and $m \in \mathcal{M}$ denote an input text and a subject LLM, respectively. The input x can be either a user query or an LLM-generated response. Let $\mathcal{G} = \{c_i\}_{i=1}^{K}$ represent a set of safety guidelines, where K is the number of specific risk categories. The objective of a guardrail model $p : \mathcal{X} \times \mathcal{M} \rightarrow \mathcal{Y}$, where $\mathcal{Y} = \{\text{Safe}\} \cup \mathcal{G}$, is to determine whether the input x is safe or unsafe under the given safety guidelines when interacting with the subject LLM m. If x is deemed unsafe, the guardrail model p is expected to assign it to the specific violation categories within $\mathcal{G}$. Notably, the guardrail model is designed to be an external classifier that operates independently without accessing the subject LLM's internal parameters.

3.2 Model Overview

Figure 2 illustrates an overview of the proposed LS-Guard framework. It comprises three key designs: (1) ***Subject LLM Capability Profiling*** for explicitly probing the unique vulnerability patterns of distinct subject LLMs (§3.3), (2) ***Multi-LoRA Guardrail Learning*** for collaboratively constructing a unified guardrail that leverages LLM-specific data to achieve personalized safeguarding (§3.4), and (3) ***Adaptive Guardrail Inference*** for producing robust and reliable safety decisions tailored to different subject LLMs (§3.5). In what follows, we will detail these designs separately.

3.3 Subject LLM Capability Profiling

Existing guardrails are typically trained on static, model-agnostic red-teaming datasets and perform safeguarding without accounting for the behavioral characteristics of specific subject LLMs. However, different subject LLMs can exhibit heterogeneous response behaviors to malicious inputs—some can effectively resist harmful queries, whereas others are easily jailbroken under identical inputs. Motivated by this observation, this stage aims to characterize the intrinsic behaviors and vulnerabilities of distinct subject LLMs to enable "personalized" safeguarding. To accomplish this, LS-Guard first performs capability profiling for distinct subject LLM $m \in \mathcal{M}$. Specifically, we employ a set of malicious queries $\mathcal{Q} = \{q_i\}_{i=1}^{N}$, covering multiple violation types defined in the safety guideline $\mathcal{G}$. Each query q_i is prompted to every subject LLM m to obtain its generated response r_i^m, forming a model-specific database $\mathcal{D}_m^{\text{Base}} = \{(q_i, r_i^m)\}_{i=1}^{N}$. Following Ghosh et al. (2025), we then derive the safety label for each pair (q_i, r_i^m) using a jury-of-LLMs evaluator that ensembles three strong LLMs under predefined safety guidelines as follows:

$$y_i^m = \text{LLM\_Ensemble}(\texttt{Inst}(q_i, r_i^m)), \quad (1)$$

where $y_i^m \in \mathcal{Y} = \{\text{Safe}\} \cup \mathcal{G}$, and $\texttt{Inst}(\cdot)$ denotes the customized instruction template guiding the jury models in determining safety labels.

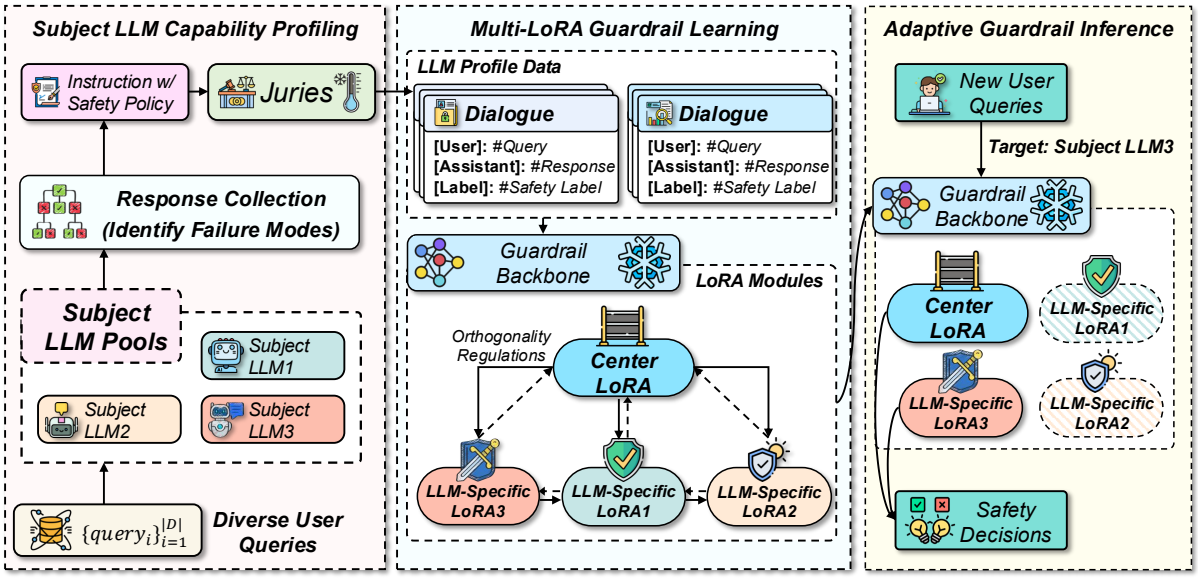

Figure 2: An overview of the proposed LS-Guard framework. It consists of three stages: Subject LLM Capability Profiling, Multi-LoRA Guardrail Learning, and Adaptive Guardrail Inference.

We denote the resulting subject LLM-specific red-teaming data as $\mathcal{D}_m = \{(q_i, r_i^m, y_i^m)\}_{i=1}^N$ and the aggregated dataset across all subject LLMs as $\mathcal{D} = \{\mathcal{D}_m \mid m \in \mathcal{M}\}$. Notably, these datasets collectively portray how different subject LLMs respond to harmful inputs, thus establishing the LLM capability profiles that underpin subsequent personalized guardrail learning.

Profiling Data Quality. To verify the quality of the safety labels assigned by the Jury-of-LLMs, we conduct a human evaluation on 100 randomly sampled profiling instances. Specifically, evaluators are tasked with reviewing each instance and assigning a binary label to assess the correctness of these LLM annotations. The results show an accuracy of 94.5%, with a Cohen's Kappa score $\kappa = 0.52$ (a moderate agreement between human evaluators) (Cohen, 1960; Fleiss and Cohen, 1973). This demonstrates that the profiling labels closely align with human judgments, underscoring the quality and practicality of the profiling data for guiding model-specific guardrail learning.

3.4 Multi-LoRA Guardrail Learning

After profiling the subject LLM capabilities, LS-Guard proceeds to construct a unified guardrail that jointly learns from multiple subject LLMs while preserving their distinct behavioral characteristics. A key challenge is how to effectively integrate the heterogeneous data from multiple subject LLMs without causing negative interference or overfitting to individual subject LLMs. To address this, we propose a multi-LoRA learning framework to optimize the guardrail backbone, enabling collaborative yet disentangled learning across different subject LLMs (Hu et al., 2022). The core of this framework consists of a central expert that acquires generalizable safety knowledge shared across all subject LLMs, and a set of subject-specific experts that capture the unique vulnerability patterns derived from individual LLM capability profiles.

Central LoRA Expert. To retain shared safety knowledge and enhance cross-model generalization, LS-Guard incorporates a central LoRA expert, denoted as $\text{LoRA}_c(\cdot)$. This expert follows the standard LoRA formulation, learning transferable safety representations from the aggregated dataset $\mathcal{D}$. Formally, given an arbitrary input $x_i \in \mathcal{D}$, the central expert extracts its representation as:

$$\boldsymbol{h}_i^c = \text{LoRA}_c(x_i), \tag{2}$$

where the trainable low-rank matrices of the central expert are denoted as $A^c \in \mathbb{R}^{r \times d_{\text{in}}}$ and $B^c \in \mathbb{R}^{d_{\text{out}} \times r}$, with $r \ll \min(d_{\text{in}}, d_{\text{out}})$.

Subject-Specific LoRA Experts. To complement the central expert, LS-Guard introduces multiple subject-specific LoRA experts, each tailored to a specific subject LLM $m \in \mathcal{M}$. These experts specialize in modeling the unique vulnerability and safety response patterns identified in the capability

profiling stage. For a training instance x_i^m sampled from the capability profiling data of the subject LLM m, the corresponding expert encodes its representation as:

$$\boldsymbol{h}_i^m = \text{LoRA}_m(x_i^m), \tag{3}$$

where (A^m, B^m) are the trainable matrices unique to the expert associated with subject LLM m.

LoRA Experts Collaboration. To enable effective collaboration among all LoRA experts, LS-Guard employs an orthogonality-based multi-task learning paradigm. It jointly optimizes all LoRA parameters by introducing two complementary regularizations that encourage both representational specificity and learning stability, thereby achieving a synergistic balance between cross-model generalization and subject LLM-specific adaptability. Formally, the overall training objective is:

$$\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{lm}} + \alpha\mathcal{L}_{\text{orth}} + \beta\mathcal{L}_{\text{stab}}, \tag{4}$$

where $\mathcal{L}_{\text{lm}}$ is the standard language modeling loss. α and β are coefficients controlling the relative contribution of the two regularization terms. $\mathcal{L}_{\text{orth}}$ denotes the orthogonality regularization loss used to promote representation diversity and prevent interference between shared and specialized subspaces, which is defined as follows:

$$\mathcal{L}_{\text{orth}} = \sum_{i<j} \left\| (B_i A_i)^\top (B_j A_j) \right\|_F^2. \tag{5}$$

$\mathcal{L}_{\text{stab}}$ is the stability regularization that maintains the consistency of global safety knowledge. LS-Guard penalizes deviations of the central expert's update ΔW_{c} from its initialization $\Delta W_{\text{c}}^{(0)}$:

$$\mathcal{L}_{\text{stab}} = \left\| \Delta W_{\text{c}} - \Delta W_{\text{c}}^{(0)} \right\|_F^2. \tag{6}$$

This regularization constrains the central expert to stay close to its initial shared safety representation, preventing over-specialization and preserving the learned general knowledge across subject LLMs.

3.5 Adaptive Guardrail Inference

During inference, LS-Guard dynamically activates the central and subject LLM-specific LoRA experts to perform adaptive safeguarding for each subject LLM. This process fuses general safety knowledge from the central expert with model-specific behavioral nuances, enabling robust and personalized moderation across heterogeneous subject LLMs. Specifically, given an input x and its associated subject LLM m, LS-Guard simultaneously activates the central expert $\text{LoRA}_c(\cdot)$ and the corresponding subject-specific expert $\text{LoRA}_m(\cdot)$. The final safeguarding decision is predicted as:

$$\hat{y} = Wx + \gamma\text{LoRA}_c(x) + (1-\gamma)\text{LoRA}_m(x), \tag{7}$$

where W denotes the frozen pre-trained weights of the guardrail backbone, and $\gamma \in [0, 1]$ is a coefficient controlling the contribution of shared versus model-specific knowledge. This parameter-level integration allows LS-Guard to exploit complementary rather than redundant interactions between model subspaces, ensuring more effective defense.

4 Experiments

4.1 Datasets

Training Resources. We utilize the ***Aegis2.0*** dataset (Ghosh et al., 2025) as the primary data resource for developing and evaluating LS-Guard. The dataset provides a diverse collection of red-teaming interactions annotated with fine-grained safety categories, serving as a comprehensive benchmark for guardrail modeling. We keep the same official train, development, and test splits as released in this repository[2].

Evaluation Benchmarks. To further assess the effectiveness of the LS-Guard framework, we evaluate it across multiple safety-related benchmarks, including ***BeaverTails*** (Ji et al., 2023b), ***ToxicChat*** (Lin et al., 2023), and ***WildGuardMix*** (Han et al., 2024). Further details on these benchmarks are discussed in Appendix A.1.

4.2 Model Setup

We evaluate a diverse set of subject LLMs within LS-Guard, including gpt-3.5-turbo-0125, Llama-3-8B, Mistral-7B-v0.1, claude-3-5-sonnet-20240620, qwen-plus-2024-09-19, and gemma-1.1-7b-it.

4.3 Baselines

We compare our LS-Guard model against the following guardrail baselines in our experiments:

Training-free Methods : (1) OpenAI Moderation API (Markov et al., 2023), (2) LLaMA Guard Series (LLaMA Guard 2-8B, LLaMA Guard 3-1B, and LLaMA Guard 3-8B) (Inan et al., 2023; Dubey et al., 2024), and (3) WildGuard (Han et al., 2024).

[2] https://huggingface.co/datasets/nvidia/Aegis-AI-Content-Safety-Dataset-2.0

Guardrail	GPT-3.5		LLaMA 3-8B		Mistral		Claude		Qwen		Gemma	
	F1	AUPRC	F1	AUPRC	F1	AUPRC	F1	AUPRC	F1	AUPRC	F1	AUPRC
Training-free Methods												
OpenAI MOD API	0.378	0.441	0.420	0.480	0.450	0.510	0.125	0.143	0.116	0.134	0.108	0.124
LLaMAGuard2-8B	0.768	0.812	0.780	0.835	0.790	0.840	0.234	0.249	0.218	0.232	0.203	0.216
LLaMAGuard3-1B	0.496	0.550	0.510	0.580	0.520	0.600	0.153	0.173	0.142	0.161	0.132	0.150
LLaMAGuard3-8B	0.773	0.832	0.785	0.845	0.795	0.860	0.235	0.254	0.220	0.232	0.204	0.220
WildGuard	0.819	0.890	0.830	0.900	0.840	0.910	0.249	0.270	0.232	0.252	0.216	0.234
Training-based Methods												
LightweightBert	0.816	0.885	0.825	0.895	0.835	0.905	0.955	0.988	0.965	0.991	0.960	0.990
R2-Guard	0.820	0.880	0.830	0.890	0.854	0.928	0.959	0.989	0.968	0.992	0.963	0.991
LoRA-Guard	0.830	0.900	0.842	0.900	**0.877**	**0.940**	0.962	0.990	0.970	0.993	0.965	0.985
DSA	0.860	0.920	0.873	0.931	0.862	0.920	0.965	0.992	0.972	0.994	0.968	**0.992**
AegisGuard	0.870	0.943	0.880	**0.950**	0.868	0.910	**0.969**	0.993	0.975	0.995	0.971	0.987
LS-Guard	**0.890**	**0.950**	**0.883**	0.940	0.876	0.932	0.960	**0.994**	**0.978**	**0.996**	**0.974**	0.990

Table 1: Safeguarding performance comparison of various guardrail models across various subject LLMs.

Training-based Methods : (1) LightweightBert (Zheng et al., 2025), (2) R2-Guard (Kang and Li, 2025), (3) LoRA-Guard (Elesedy et al., 2024), (4) DSA (Krishna et al., 2025), and (5) AegisGuard (Ghosh et al., 2025). We provide additional details for these baseline methods in Appendix A.2.

4.4 Evaluation Metrics

Following (Ghosh et al., 2025; Lee et al., 2025), we employ two widely used metrics for evaluating the guardrail performance: **F1**-score and Area Under the Precision–Recall Curve (**AUPRC**). See more details about these metrics and experimental settings in Appendix A.3 and A.4.

4.5 Main Results

Table 1 presents the main results of LS-Guard against existing baselines, where the peak performance is highlighted in **bold**. Generally speaking, our LS-Guard achieves significant improvements compared with the baselines across six distinct subject LLMs. We analyze the results as follows:

LS-Guard learns the unique vulnerability patterns of subject LLMs to enhance guardrail performance. Table 1 reveals that LS-Guard significantly outperforms existing top-performing guardrail baselines. For instance, LS-Guard surpasses the previous best-performing training-free model, WildGuard, by 7.1% in F1 and 6.0% in AUPRC when safeguarding GPT-3.5. Similar gains are also observed against training-based guardrail approaches. Notably, LS-Guard not only achieves substantial improvements but also establishes state-of-the-art performance in LLM safeguarding that consistently generalizes to OOD evaluation (see Table 3). This demonstrates LS-Guard's ability to internalize the unique vulnerability patterns of distinct subject LLMs identified during the capability profiling stage to achieve superior personalized guardrails, effectively navigating the dynamics and complexities of real-world LLM deployments.

LS-Guard's Multi-LoRA collaboration facilitates effective cross-model generalization and adaptive safeguarding. Among the above guardrail baselines, the training-free approaches exhibit dramatic performance fluctuations across various subject LLMs, while the training-based approaches generally underperform when compared with LS-Guard. This underscores the inefficacy of merely employing a single guardrail for safeguarding all LLMs or developing separate guardrails tailored to individual models. In contrast, LS-Guard achieves significant gains while maintaining stable performance across all subject LLMs. This observation suggests that LS-Guard's collaborative Multi-LoRA architecture, by strategically optimizing the central and subject-specific experts via orthogonality and stability regularization, extends beyond learning subject LLM-specific vulnerabilities, which not only prevents interference among experts but also promotes complementary learning, highlighting the importance of structured expert collaboration in scaling guardrail reliability across diverse deployment contexts.

4.6 In-depth Analyses

4.6.1 Ablation Studies

Within LS-Guard, the multi-LoRA guardrail architecture is pivotal for disentangling generalizable

Guardrail	GPT-3.5		LLaMA 3-8B		Mistral		Claude		Qwen		Gemma	
	F1	AUPRC	F1	AUPRC	F1	AUPRC	F1	AUPRC	F1	AUPRC	F1	AUPRC
LS-Guard	**0.89**	**0.95**	**0.883**	**0.94**	**0.876**	**0.932**	**0.96**	**0.994**	**0.978**	**0.996**	**0.974**	**0.99**
w/o Central Expert	0.86	0.92	0.86	0.91	0.85	0.91	0.93	0.96	0.95	0.97	0.94	0.96
w/o Specific Expert	0.87	0.93	0.85	0.92	0.86	0.92	0.94	0.97	0.96	0.98	0.95	0.97

Table 2: Ablation study on the effectiveness of multi-LoRA collaboration in the LS-Guard framework.

safety knowledge from subject LLM-specific specializations to improve safeguarding performance. To investigate the impact of different LoRA experts in the multi-LoRA collaboration on LS-Guard's effectiveness, we conduct comprehensive ablation studies, with the experimental results detailed in Table 2. Specifically, we selectively remove two types of LoRA experts—the central expert and subject LLM-specific experts—during the model training process, where the *w/o* indicates removal of the corresponding LoRA module. As shown in Table 2, excluding either type of experts consistently degrades performance across all subject LLMs. In particular, the ablation of the central expert, which encodes generalizable safety knowledge, yields the most pronounced degradation in LS-Guard's performance. This suggests that the central expert plays a foundational role in preserving shared safety principles and enabling strong cross-model generalization. Similarly, removing the subject-specific experts leads to a smaller yet non-trivial decline. This highlights the necessity of tailoring the guardrail's behavior to account for distinct vulnerability patterns exhibited by individual subject LLMs. Absent these specialized experts, LS-Guard's ability to adapt to fine-grained behavioral nuances diminishes, weakening its effectiveness in mitigating model-specific risks.

4.6.2 Effectiveness on OOD Safety Benchmarks

To comprehensively assess model performance, we empirically evaluate the LS-Guard framework on out-of-distribution (OOD) safety benchmarks. These data feature content with different safety concerns, offering a complementary perspective beyond in-distribution evaluation. Table 3 compares the performances of LS-Guard and representative baselines on the OOD test sets. We can saliently notice that LS-Guard consistently outperforms other top-performing guardrail models across multiple OOD benchmarks and subject LLMs. For example, when safeguarding the subject LLM LLaMA 3-8B on the BeaverTails test set, LS-Guard surpasses AegisGuard by 4.1% in F1 score and 3.0% in AUPRC. This underscores the robustness of the proposed LS-Guard and its capability to deliver superior safeguarding performance.

Guardrail	GPT-3.5		LLaMA 3-8B		Mistral	
	F1	AUPRC	F1	AUPRC	F1	AUPRC
BeaverTails						
LightweightBert	0.642	0.658	0.671	0.684	0.693	0.715
R2-Guard	0.785	0.800	0.792	0.815	0.810	0.827
AegisGuard	0.805	0.813	0.803	0.824	0.836	0.852
LS-Guard	**0.837**	**0.851**	**0.844**	**0.854**	**0.877**	**0.883**
Toxic Chat						
LightweightBert	0.742	0.768	0.775	0.793	0.804	0.810
R2-Guard	0.860	0.899	0.884	0.910	0.905	0.923
AegisGuard	0.894	0.920	0.909	0.933	0.925	0.940
LS-Guard	**0.918**	**0.936**	**0.925**	**0.937**	**0.943**	**0.956**
WildGuardMix						
LightweightBert	0.665	0.713	0.680	0.726	0.713	0.754
R2-Guard	0.790	0.825	0.808	0.820	0.813	0.847
AegisGuard	0.842	0.871	0.839	0.847	0.821	0.855
LS-Guard	**0.854**	**0.890**	**0.870**	**0.915**	**0.864**	**0.903**

Table 3: Results on OOD safety benchmarks.

Additionally, it is noted that the lightweight baselines (*e.g.*, LightweightBert) exhibit a substantial performance drop under OOD evaluation relative to their in-distribution performance, underscoring the importance of in-distribution training. However, the proposed LS-Guard framework achieves larger improvements over the lightweight baselines and remains superior to other large-scale guardrails. This observation further strengthens the benefits of LS-Guard's multi-LoRA collaboration in preserving generalizable safety knowledge while capturing model-specific vulnerability patterns, thereby effectively coping with safety challenges even without in-distribution training.

4.6.3 Effect of Multi-LoRA Collaboration

To investigate the efficacy of the multi-LoRA expert collaboration within the LS-Guard framework, we conduct further experiments to explore the effect of varying the extent of collaboration between the central and the subject-specific LLM experts on model safeguarding performance. To achieve

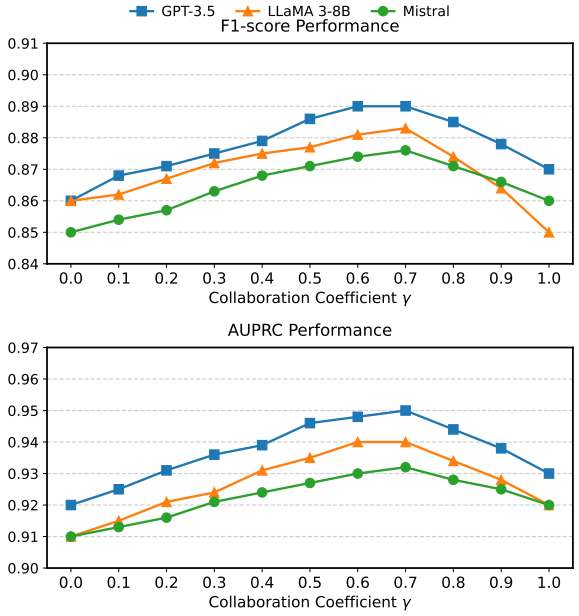

Figure 3: Effect of the multi-LoRA expert collaboration.

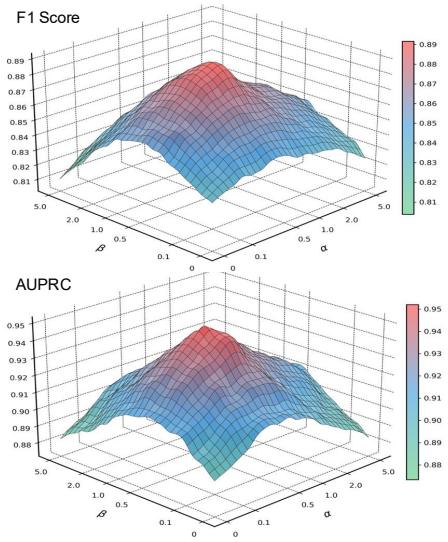

Figure 4: Impact of orthogonality and stability regularizations on LS-Guard's performance.

this, we experimentally sweep the collaboration coefficient γ in Equation (7) from 0 to 1 in increments of 0.1, where γ=0 reduces to exclusively utilizing the subject LLM–specific expert and γ=1 degenerates to employing the central expert in isolation. Figure 3 illustrates the performance trends across various collaboration coefficients within LS-Guard. Notably, as the collaboration coefficient γ increases, LS-Guard's performance improves, peaking around $\gamma = 0.7$. Beyond this point, LS-Guard's performance starts to decline. We posit that an excessively large γ overemphasizes the central expert and suppresses model-specific cues captured by the subject expert, whereas a very small γ overweights the subject expert and underutilizes the shared safety principles learned by the central expert. Either extreme weakens the intended complementarity between experts, reducing synergy and ultimately degrading safeguarding performance.

4.6.4 Impact of Orthogonality-based Regularizations

LS-Guard primarily employs orthogonality-based multi-task regularizations during model training to synergize the LoRA experts, effectively disentangling shared safety knowledge from subject LLM-specific vulnerabilities and thereby strengthening safeguarding performance. We analyze the impact of these regularization objectives on LS-Guard's performance by strategically varying the coefficients of the regularization terms in Equation (4). The experimental results are presented in Figure 4, revealing the following key findings: (1) As α increases from 0, LS-Guard's performance increases to an optimum and then declines. This suggests that moderate orthogonality is required to disentangle shared and idiosyncratic subspaces, whereas either excessively small or large α can induce negative transfer and impede the beneficial sharing of safety knowledge. (2) The stability of the central expert's learning is critical for cross-model retention of safety priors. With $\beta \approx 0$, the central expert drifts toward the training distribution of individual subjects, reducing its ability to provide transferable safety knowledge on held-out subject LLMs. Conversely, an excessively large β over-constrains parameter updates, precluding effective model learning.

4.6.5 Visualization of Multi-LoRA Experts

For a more intuitive analysis of the effect of multi-LoRA guardrail architecture, we employ the t-SNE technique to visualize the parameters of various LoRA experts learned within the LS-Guard framework, contrasting them with a variant trained without orthogonality and stability regularizations. The visualization results are shown in Figure 5. This analysis yields the following key observations: (1) When multiple LoRA experts are trained in isolation on different subject-LLM profiling data, the parameters of these LoRA experts exhibit a high degree of similarity, showing substantial overlap in the representation space. (2) Conversely, incorporating the orthogonality-based regularizations during the multi-LoRA guardrail learning process

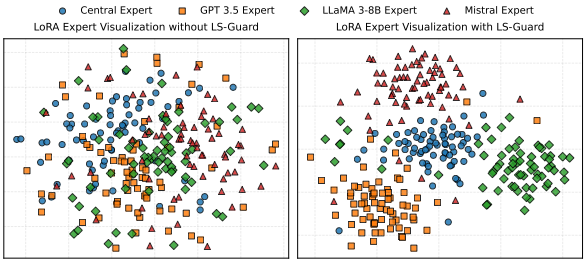

Figure 5: Visualization of multi-LoRA experts.

results in more distinguishable LoRA expert parameters. The distinction between the standard LS-Guard and its variant offers valuable insights into how LS-Guard separates shared and idiosyncratic subspaces, revealing cleaner clustering for subject LLM-specific behaviors while preserving a coherent shared manifold.

5 Conclusion

This work presents **LS-Guard**, a novel and robust framework that effectively safeguards deployed LLMs by explicitly modeling the vulnerabilities and capabilities of different subject LLMs. We highlight the limitations of the prevailing "one-model-fits-all" guardrail paradigm and demonstrate the necessity of adaptive, "personalized" guardrails aligned with individual subject LLMs' harmful behavior patterns. By integrating LLM capability profiling with a collaborative Multi-LoRA design—featuring a central expert for shared safety priors and subject-specific experts for idiosyncratic risks—LS-Guard disentangles generalizable knowledge from model-specific vulnerabilities, significantly improving guardrail performance over baselines. This framework provides a scalable, modular pathway for safeguarding heterogeneous LLMs—enabling plug-in adaptation to new models with minimal additional parameters—while maintaining interpretable profiling of model-specific risks. Future work can explore more advanced LLM profiling mechanisms and self-improving strategies under distribution shift to further strengthen safeguarding performance.

Limitations

While the proposed LS-Guard framework marks a significant advance in safeguarding deployed LLMs, we identify several limitations of our work.

First, due to computational constraints, our experiments were conducted on mid-sized LLMs. Although the experimental results demonstrate the effectiveness of LS-Guard in safeguarding diverse LLMs, its scalability and performance with larger-scale backbones remain underexplored. Future work should assess whether LS-Guard maintains its efficacy and adaptability when instantiated with frontier LLM backbones.

Second, our LS-Guard approach focuses solely on safety risks in purely textual interactions. Its ability to generalize to other modalities and contexts (e.g., vision, audio, or embodied agents) remains an open question. Exploring unique safety challenges posed by such multimodal contexts and effectively adapting the LS-Guard framework to address them is another promising avenue for future research.

Despite the above limitations, extensive experimental results demonstrate the practicality, efficacy, and robustness of LS-Guard, paving the way for scalable and adaptable guardrails for deployed LLMs.

Ethical Considerations

This work is conducted solely for research purposes, aiming to explore learning model-specific guardrails tailored to each LLM's vulnerabilities to secure LLM-powered systems. Our objective is to enhance the safety and reliability of AI systems rather than deploy models that could pose ethical concerns. We strictly do not allow or facilitate direct user interaction with the models trained in this study. Additionally, we adhere to responsible AI principles, ensuring that our research contributes to improving AI safety without introducing unintended risks.

Acknowledgments

This research was supported by the National Research Foundation, Singapore under its National Large Language Models Funding Initiative (AISG Award No. AISG-NMLP-2024-002), and by the Ministry of Education, Singapore, under its AcRF Tier 2 Funding (Proposal ID: T2EP20123-0052). Any opinions, findings, conclusions, or recommendations expressed in this material are those of the author(s) and do not reflect the views of the National Research Foundation or the Ministry of Education, Singapore.

References

Pin-Yu Chen, Han Shen, Payel Das, and Tianyi Chen. 2025. Fundamental safety-capability trade-offs in fine-tuning large language models. *ArXiv*, abs/2503.20807.

Jianfeng Chi, Ujjwal Karn, Hongyuan Zhan, Eric Smith, Javier Rando, Yiming Zhang, Kate Plawiak, Zacharie Delpierre Coudert, Kartikeya Upasani, and Mahesh Pasupuleti. 2024. Llama guard 3 vision: Safeguarding human-ai image understanding conversations. *CoRR*, abs/2411.10414.

Yuqi Chu, Lizi Liao, Zhiyuan Zhou, Chong-Wah Ngo, and Richang Hong. 2025. Towards multimodal emotional support conversation systems. *IEEE Transactions on Multimedia*.

Christopher Clarke, Matthew Hall, Gaurav Mittal, Ye Yu, Sandra Sajeev, Jason Mars, and Mei Chen. 2023. Rule by example: Harnessing logical rules for explainable hate speech detection. In *ACL*, pages 364–376.

Jacob Cohen. 1960. A coefficient of agreement for nominal scales. *Educational and Psychological Measurement*, pages 37 – 46.

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*.

Zhichen Dong, Zhanhui Zhou, Chao Yang, Jing Shao, and Yu Qiao. 2024. Attacks, defenses and evaluations for LLM conversation safety: A survey. In *NAACL*, pages 6734–6747.

Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, Anirudh Goyal, Anthony Hartshorn, Aobo Yang, Archi Mitra, Archie Sravankumar, Artem Korenev, Arthur Hinsvark, Arun Rao, Aston Zhang, and 82 others. 2024. The llama 3 herd of models. *CoRR*.

Hayder Elesedy, Pedro M Esperanca, Silviu Vlad Oprea, and Mete Ozay. 2024. LoRA-guard: Parameter-efficient guardrail adaptation for content moderation of large language models. In *EMNLP*, pages 11746–11765.

Igor Fedorov, Kate Plawiak, Lemeng Wu, Tarek Elgamal, Naveen Suda, Eric Smith, Hongyuan Zhan, Jianfeng Chi, Yuriy Hulovatyy, Kimish Patel, Zechun Liu, Changsheng Zhao, Yangyang Shi, Tijmen Blankevoort, Mahesh Pasupuleti, Bilge Soran, Zacharie Delpierre Coudert, Rachad Alao, Raghuraman Krishnamoorthi, and Vikas Chandra. 2024. Llama guard 3-1b-int4: Compact and efficient safeguard for human-ai conversations. *CoRR*, abs/2411.17713.

Duanyu Feng, Bowen Qin, Chen Huang, Youcheng Huang, Zheng Zhang, and Wenqiang Lei. 2024. Legend: Leveraging representation engineering to annotate safety margin for preference datasets. In *AAAI*.

Joseph L. Fleiss and Jacob Cohen. 1973. The equivalence of weighted kappa and the intraclass correlation coefficient as measures of reliability. *Educational and Psychological Measurement*, pages 613 – 619.

Shaona Ghosh, Prasoon Varshney, Erick Galinkin, and Christopher Parisien. 2024. AEGIS: online adaptive AI content safety moderation with ensemble of LLM experts. *CoRR*, abs/2404.05993.

Shaona Ghosh, Prasoon Varshney, Makesh Narsimhan Sreedhar, Aishwarya Padmakumar, Traian Rebedea, Jibin Rajan Varghese, and Christopher Parisien. 2025. AEGIS2.0: A diverse AI safety dataset and risks taxonomy for alignment of LLM guardrails. In *NAACL*, pages 5992–6026.

Juan Felipe Gómez, Caio Vieira Machado, Lucas Monteiro Paes, and Flávio P. Calmon. 2024. Algorithmic arbitrariness in content moderation. In *FAccT*, pages 2234–2253.

Seungju Han, Kavel Rao, Allyson Ettinger, Liwei Jiang, Bill Yuchen Lin, Nathan Lambert, Yejin Choi, and Nouha Dziri. 2024. Wildguard: Open one-stop moderation tools for safety risks, jailbreaks, and refusals of llms. In *NeurIPS*.

Junxian He, Chunting Zhou, Xuezhe Ma, Taylor Berg-Kirkpatrick, and Graham Neubig. 2022. Towards a unified view of parameter-efficient transfer learning. In *ICLR*.

Neil Houlsby, Andrei Giurgiu, Stanislaw Jastrzebski, Bruna Morrone, Quentin de Laroussilhe, Andrea Gesmundo, Mona Attariyan, and Sylvain Gelly. 2019. Parameter-efficient transfer learning for NLP. In *ICML*, pages 2790–2799.

Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2022. Lora: Low-rank adaptation of large language models. In *ICLR*.

Chengsong Huang, Qian Liu, Bill Yuchen Lin, Tianyu Pang, Chao Du, and Min Lin. 2023. Lorahub: Efficient cross-task generalization via dynamic lora composition. *CoRR*, abs/2307.13269.

Tiansheng Huang, Sihao Hu, Fatih Ilhan, Selim Furkan Tekin, Zachary Yahn, Yichang Xu, and Ling Liu. 2025. Safety tax: Safety alignment makes your large reasoning models less reasonable. *ArXiv*, abs/2503.00555.

Hakan Inan, Kartikeya Upasani, Jianfeng Chi, Rashi Rungta, Krithika Iyer, Yuning Mao, Michael Tontchev, Qing Hu, Brian Fuller, Davide Testuggine, and Madian Khabsa. 2023. Llama guard: Llm-based input-output safeguard for human-ai conversations. *CoRR*, abs/2312.06674.

Jiaming Ji, Donghai Hong, Borong Zhang, Boyuan Chen, Josef Dai, Boren Zheng, Tianyi Alex Qiu, Jiayi Zhou, Kaile Wang, Boxun Li, Sirui Han, Yike Guo, and Yaodong Yang. 2025. PKU-SafeRLHF: Towards multi-level safety alignment for LLMs with human preference. In *ACL*, pages 31983–32016, Vienna, Austria.

Jiaming Ji, Mickel Liu, Josef Dai, Xuehai Pan, Chi Zhang, Ce Bian, Boyuan Chen, Ruiyang Sun, Yizhou Wang, and Yaodong Yang. 2023a. Beavertails: Towards improved safety alignment of llm via a human-preference dataset. *NeurIPS*, 36:24678–24704.

Jiaming Ji, Mickel Liu, Josef Dai, Xuehai Pan, Chi Zhang, Ce Bian, Boyuan Chen, Ruiyang Sun, Yizhou Wang, and Yaodong Yang. 2023b. Beavertails: Towards improved safety alignment of LLM via a human-preference dataset. In *NeurIPS*.

Mintong Kang and Bo Li. 2025. R2-guard: Robust reasoning enabled LLM guardrail via knowledge-enhanced logical reasoning. In *ICLR*.

Kundan Krishna, Joseph Y. Cheng, Charles Maalouf, and Leon A. Gatys. 2025. Disentangled safety adapters enable efficient guardrails and flexible inference-time alignment. *CoRR*, abs/2506.00166.

Martin Kuo, Jianyi Zhang, Aolin Ding, Qinsi Wang, Louis DiValentin, Yujia Bao, Wei Wei, Hai Li, and Yiran Chen. 2025. H-cot: Hijacking the chain-of-thought safety reasoning mechanism to jailbreak large reasoning models, including openai o1/o3, deepseek-r1, and gemini 2.0 flash thinking. *ArXiv*, abs/2502.12893.

Seanie Lee, Haebin Seong, Dong Bok Lee, Minki Kang, Xiaoyin Chen, Dominik Wagner, Yoshua Bengio, Juho Lee, and Sung Ju Hwang. 2025. Harmaug: Effective data augmentation for knowledge distillation of safety guard models. In *ICLR*.

Alyssa Lees, Vinh Q. Tran, Yi Tay, Jeffrey Sorensen, Jai Prakash Gupta, Donald Metzler, and Lucy Vasserman. 2022. A new generation of perspective API: efficient multilingual character-level transformers. In *KDD*, pages 3197–3207.

Zi Lin, Zihan Wang, Yongqi Tong, Yangkun Wang, Yuxin Guo, Yujia Wang, and Jingbo Shang. 2023. Toxicchat: Unveiling hidden challenges of toxicity detection in real-world user-ai conversation. In *Findings of EMNLP*, pages 4694–4702.

Qidong Liu, Xian Wu, Xiangyu Zhao, Yuanshao Zhu, Derong Xu, Feng Tian, and Yefeng Zheng. 2023. Moelora: An moe-based parameter efficient fine-tuning method for multi-task medical applications. *CoRR*, abs/2310.18339.

Weidi Luo, Shenghong Dai, Xiaogeng Liu, Suman Banerjee, Huan Sun, Muhao Chen, and Chaowei Xiao. 2025. Agrail: A lifelong agent guardrail with effective and adaptive safety detection. In *ACL*, pages 8104–8139.

Xingtai Lv, Ning Ding, Yujia Qin, Zhiyuan Liu, and Maosong Sun. 2023. Parameter-efficient weight ensembling facilitates task-level knowledge transfer. In *ACL*, pages 270–282.

Todor Markov, Chong Zhang, Sandhini Agarwal, Florentine Eloundou Nekoul, Theodore Lee, Steven Adler, Angela Jiang, and Lilian Weng. 2023. A holistic approach to undesired content detection in the real world. In *AAAI*, pages 15009–15018.

Yuqi Nie, Yaxuan Kong, Xiaowen Dong, John M. Mulvey, H. Vincent Poor, Qingsong Wen, and Stefan Zohren. 2024. A survey of large language models for financial applications: Progress, prospects and challenges. *ArXiv*, abs/2406.11903.

OpenAI. 2023. GPT-4 technical report. *CoRR*.

Piyush Pant. 2025. Improving llm safety and helpfulness using sft and dpo: A study on opt-350m. *ArXiv*, abs/2509.09055.

Pujan Paudel, Jeremy Blackburn, Emiliano De Cristofaro, Savvas Zannettou, and Gianluca Stringhini. 2023. Lambretta: Learning to rank for twitter soft moderation. In *Symposium on Security and Privacy*, pages 311–326.

Xuankun Rong, Wenke Huang, Tingfeng Wang, Daiguo Zhou, Bo Du, and Mang Ye. 2025. Safegrpo: Self-rewarded multimodal safety alignment via rule-governed policy optimization. *ArXiv*.

Ying Sheng, Shiyi Cao, Dacheng Li, Coleman Hooper, Nicholas Lee, Shuo Yang, Christopher Chou, Banghua Zhu, Lianmin Zheng, Kurt Keutzer, Joseph E. Gonzalez, and Ion Stoica. 2023. S-lora: Serving thousands of concurrent lora adapters. *CoRR*, abs/2311.03285.

Jean Y. Song, Sangwook Lee, Jisoo Lee, Mina Kim, and Juho Kim. 2023. Modsandbox: Facilitating online community moderation through error prediction and improvement of automated rules. In *CHI*, pages 107:1–107:20.

Yi-Lin Sung, Jaemin Cho, and Mohit Bansal. 2022. VL-ADAPTER: parameter-efficient transfer learning for vision-and-language tasks. In *CVPR*, pages 5217–5227.

Chunlin Tian, Zhan Shi, Zhijiang Guo, Li Li, and Cheng-Zhong Xu. 2024. Hydralora: An asymmetric lora architecture for efficient fine-tuning. In *NeurIPS*.

Sahil Verma, Keegan Hines, Jeff Bilmes, Charlotte Siska, Luke Zettlemoyer, Hila Gonen, and Chandan Singh. 2025. OMNIGUARD: an efficient approach for AI safety moderation across modalities. *CoRR*, abs/2505.23856.

Xunguang Wang, Zhenlan Ji, Wenxuan Wang, Zongjie Li, Daoyuan Wu, and Shuai Wang. 2025. Sok: Evaluating jailbreak guardrails for large language models. *arXiv preprint arXiv:2506.10597*.

Yiming Wang, Yu Lin, Xiaodong Zeng, and Guannan Zhang. 2023. Multilora: Democratizing lora for better multi-task learning. *CoRR*, abs/2311.11501.

Johannes Welbl, Amelia Glaese, Jonathan Uesato, Sumanth Dathathri, John Mellor, Lisa Anne Hendricks, Kirsty Anderson, Pushmeet Kohli, Ben Coppin, and Po-Sen Huang. 2021. Challenges in detoxifying language models. In *Findings of EMNLP*, pages 2447–2469.

Yaozu Wu, Jizhou Guo, Dongyuan Li, Henry Peng Zou, Wei-Chieh Huang, Yankai Chen, Zhen Wang, Weizhi Zhang, Yangning Li, Meng Zhang, and 1 others. 2025. Psg-agent: Personality-aware safety guardrail for llm-based agents. *arXiv preprint arXiv:2509.23614*.

Yuhui Xu, Lingxi Xie, Xiaotao Gu, Xin Chen, Heng Chang, Hengheng Zhang, Zhengsu Chen, Xiaopeng Zhang, and Qi Tian. 2024. Qa-lora: Quantization-aware low-rank adaptation of large language models. In *ICLR*.

Zhuowen Yuan, Zidi Xiong, Yi Zeng, Ning Yu, Ruoxi Jia, Dawn Song, and Bo Li. 2024. Rigorllm: Resilient guardrails for large language models against undesired content. In *ICML*.

Aaron Zheng, Mansi Rana, and Andreas Stolcke. 2025. Lightweight safety guardrails using fine-tuned BERT embeddings. In *Proceedings of COLING - Industry Track*, pages 689–696.

Zhonghua Zheng, Lizi Liao, Yang Deng, and Liqiang Nie. 2023. Building emotional support chatbots in the era of llms. *CoRR*, abs/2308.11584.

Han Zhou, Xingchen Wan, Ivan Vulic, and Anna Korhonen. 2024. Autopeft: Automatic configuration search for parameter-efficient fine-tuning. *Trans. Assoc. Comput. Linguistics*, pages 525–542.

A Experimental Details

A.1 Datasets

In this work, we conduct experiments on the following datasets: ***Aegis2.0*** (Ghosh et al., 2025), ***BeaverTails*** (Ji et al., 2023b), ***ToxicChat*** (Lin et al., 2023), and ***WildGuardMix*** (Han et al., 2024).

Training Resources. Specifically, ***Aegis2.0*** is a comprehensive, human-annotated safety dataset designed to develop effective safety guardrails for LLMs while addressing the critical shortage of high-quality, commercially viable datasets that cover a full spectrum of safety risks. It comprises a total of 34,248 samples sourced from diverse human-LLM interactions. These samples are carefully annotated according to an adaptable, tiered taxonomy consisting of 12 top-level core categories and 9 fine-grained sub-categories. We leverage these rich interactions as the primary resource for profiling subject LLM vulnerabilities and training the collaborative experts within the proposed LS-Guard framework.

Evaluation Benchmarks. ***BeaverTails*** is a large-scale dataset containing over 30,000 human-labeled question-answering pairs. Each pair is categorized under one or more labels according to a safety taxonomy of 14 distinct harm categories, such as violence, discrimination, and privacy violations. It uniquely separates annotations of helpfulness and harmlessness for question-answering pairs, offering distinct perspectives on these crucial attributes.

ToxicChat is a toxicity detection benchmark consisting of 10,166 real-world user-AI conversations. Sourced from user queries to the Vicuna chatbot demo, the dataset captures nuanced and implicit toxic behaviors specific to conversational AI that are often absent in traditional social media-based benchmarks. The dataset provides binary toxicity labels and specific annotations for various jailbreaking attempts, serving as a rigorous testbed for evaluating the robustness of safety guardrails against subtle adversarial inputs.

WildGuardMix represents a large-scale, multi-task safety moderation dataset consisting of 92K labeled examples. It provides broad coverage across 13 risk categories and is designed to identify malicious intent in user prompts, and detect safety risks in model responses. The dataset includes a balanced mix of direct prompts and adversarial jailbreaks, serving as a comprehensive resource for automatic safety moderation and evaluation.

Notably, we utilize the test sets of these benchmarks to evaluate the OOD performance and generalization capabilities of our proposed LS-Guard framework.

A.2 Baselines

In the experiments, we compare our LS-Guard with the following representative baselines:

- **OpenAI Moderation API** (Markov et al., 2023): A production-oriented text moderation classifier trained to detect a broad taxonomy of undesired content, supported by a holistic pipeline including taxonomy design, labeling guidelines, data quality control, and active learning for rare events.
- **LLaMA Guard series** (Inan et al., 2023; Dubey et al., 2024): LLaMA Guard models are LLM-based input/output safeguard classifiers that perform prompt and response classification using an explicit safety-risk taxonomy, outputting *safe/unsafe* decisions and (when unsafe) the violated risk categories.
- **WildGuard** (Han et al., 2024): An open moderation tool that jointly supports (i) malicious intent detection in user prompts, (ii) safety-risk detection in model responses, and (iii) refusal-behavior assessment, covering 13 risk categories for automated safety moderation and evaluation.
- **LightweightBert** (Zheng et al., 2025): A lightweight guardrail approach that fine-tunes a Sentence-BERT style encoder for prompt/output filtering, aiming to reduce latency and maintenance cost compared to LLM-based guardrails while keeping comparable performance on safety benchmarks.
- **R2-Guard** (Kang and Li, 2025): A reasoning-enhanced guardrail that combines (1) data-driven category-specific unsafe-probability predictors with (2) a knowledge-enhanced logical reasoning module, encoding safety knowledge as rules and performing probabilistic inference for final moderation decisions.
- **LoRA-Guard** (Elesedy et al., 2024): A parameter-efficient guardrail adaptation method that transfers language features from LLMs to a moderation model using low-rank adapters, enabling on-device content moderation with substantially lower parameter overhead while maintaining competitive accuracy.
- **DSA** (Krishna et al., 2025): Disentangled Safety Adapters decouple safety-specific computations

from the task-optimized base model by using modular adapters. By reusing the base model's internal representations, DSA achieves flexible safety functionalities (classification and alignment) with negligible incremental inference cost.

- **AegisGuard** (Ghosh et al., 2025): A guard model trained on the AEGIS2.0 risk taxonomy and human-annotated safety data to predict hazard categories for harmful inputs/outputs, aiming for broad risk coverage and strong category prediction quality across diverse safety domains.

A.3 Evaluation Metrics

In the experiments, we utilize **F1**-score and Area Under the Precision–Recall Curve (**AUPRC**) to comprehensively assess the performance of the proposed LS-Guard framework.

Specifically, the **F1**-score is the harmonic mean of precision and recall, providing a balanced measure of the model's performance on predicting unsafe content. Precision (P) measures the proportion of predicted unsafe instances that are correctly identified:

$$\mathrm{P} = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FP}}, \tag{8}$$

where TP and FP denote the number of true positives and false positives, respectively. Recall (R) measures the proportion of actual unsafe instances correctly captured:

$$\mathrm{R} = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FN}}, \tag{9}$$

where FN denotes the number of false negatives. Given the precision and recall, the F1-score is then calculated as:

$$\mathrm{F1} = 2 \times \frac{\mathrm{P} \times \mathrm{R}}{\mathrm{P} + \mathrm{R}}. \tag{10}$$

The Area Under the Precision–Recall Curve evaluates model performance across all classification thresholds by calculating the area under the curve formed by plotting precision against recall. It is defined as the integral of precision with respect to recall:

$$\mathrm{AUPRC} = \sum_{n} (\mathrm{R}_n - \mathrm{R}_{n-1}) \mathrm{P}_n, \tag{11}$$

where P_n and R_n denote the precision and recall at the n-th threshold. In the context of safety evaluation, a higher AUPRC indicates superior capability in distinguishing between safe and unsafe content, especially in imbalanced datasets where the unsafe class is sparse.

A.4 Implementation Details

For the subject LLM capability profiling stage, we implemented a Jury-of-LLMs framework following Ghosh et al. (2025). Specifically, our jury comprises three diverse, high-capability models: Mixtral-8x22B-Instruct-v0.1, Mistral-Nemo-Instruct-2407, and Gemma-2-27b-it. To ensure deterministic outputs during the annotating process, the temperature is fixed at 0, and the output length is constrained to a maximum of 256 tokens. All other hyperparameters are maintained at their default settings.

For the multi-LoRA guardrail learning stage, we utilize *Llama-3-8B-Instruct* as the guardrail backbone. We fine-tune multiple LoRA experts targeting all linear modules with rank $r = 16$, $\alpha = 32$, and dropout 0.05. The model is trained for 3 epochs with a learning rate of 1×10^{-4}, utilizing a cosine learning rate scheduler with a 10% warmup ratio. The maximum sequence length is set to 4,096 tokens. Notably, this modular design and training configuration introduce only minimal additional overhead for each subject LLM-specific expert, making the LS-Guard framework scalable to a growing set of subject LLMs. Moreover, as subject LLMs evolve, only the affected subject expert needs to be refreshed via a lightweight re-profiling and retraining step, without requiring full retraining of the entire guardrail system. This ensures that LS-Guard remains efficient and maintainable in long-term deployment.

B LLM Usage

LLMs were utilized in this work solely as auxiliary tools for linguistic refinement. Their function was restricted to enhancing the grammar, clarity, and stylistic consistency of the text that had been originally drafted by the authors. At no stage did LLMs contribute to research ideation, methodological design, data collection, analysis, or interpretation of results. All intellectual contributions, scientific content, and conclusions presented in this paper are entirely attributable to the authors. The authors accept full responsibility for the accuracy, originality, and integrity of the submission, including sections of text that may have been refined with the assistance of LLMs.