Fine-tuning Does Not Reach All: Uneven Safety and Knowledge Dynamics in Language Models

Ziwei Wang[1,2] **Xinwei Guo**[1] **Jiaxin Zhang**[1] **Guanhua Chen**[1] **Haiyan Wu**[3]
Xiangyu Zhao[4] **Lei Ma**[5] **Xin Yao**[6] **Shan He**[2] **Xuetao Wei**[1*]
[1]Southern University of Science and Technology [2]University of Birmingham
[3]University of Macau [4]City University of Hong Kong
[5]The University of Tokyo [6]Lingnan University
{12250053,guoxw2023,zhangjx,chengh3,weixt}@mail.sustech.edu.cn
s.he@bham.ac.uk, haiyanwu@um.edu.mo, xy.zhao@cityu.edu.hk
ma.lei@tokyo.ac.jp, xinyao@ln.edu.hk

Abstract

Fine-tuning is widely used to align large language models (LLMs) with desired behavioral norms such as safety, yet it is typically evaluated using aggregate metrics that assume uniform improvements across the model. However, fine-tuning operates within a constrained parameter space, and its entangled effects on localized knowledge remain poorly understood. To address this, we investigate fine-tuning at the level of individual knowledge concepts using a controlled framework with representative concepts and paired safe/unsafe samples under varying data compositions. By jointly analyzing behavioral safety via model outputs and internal knowledge accessibility via perplexity, we obtain a unified view of how knowledge is modified and expressed. Our results reveal that fine-tuning does not uniformly improve safety. Instead, it (1) exhibits strong concept-level heterogeneity, including persistent low-safety concepts and robustly safe concepts with clear domain specificity; (2) improves safe–unsafe discrimination on average, but coexists with familiarity on both safe and unsafe content or even loses previously learned safe knowledge; and (3) decouples internal knowledge from external behavior, as changes in familiarity do not reliably translate into safer outputs, especially due to early-stage disruption of existing safety structures. Our findings position consistency as a distinct objective beyond conventional aggregate safety and demonstrate that reducing unsafe knowledge exposure under uniform concept-level coverage actively homogenizes safety distributions across model knowledge, offering a principled path toward alignment that is not merely effective on average, but uniformly reliable, predictable, and robust across knowledge concepts.

1 Introduction

Fine-tuning has become a central mechanism for aligning large language models (LLMs) with desired behavioral norms, particularly for improving safety [Lu et al., 2025, Hsu et al., 2024, Wang et al., 2024a]. By adapting pretrained models using curated datasets, fine-tuning is widely assumed to provide a controllable way to steer model behavior toward safer outputs. However, fine-tuning acts upon a pretrained model characterized by a dense and distributed representational structure. The gradient updates induced by fine-tuning do not selectively isolate a localized component dedicated to safety [Wang et al., 2024b]. Instead, they perturb a high-dimensional parameter space in which safe and unsafe knowledge are inextricably co-represented. These concepts often share the same

*Corresponding author.

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

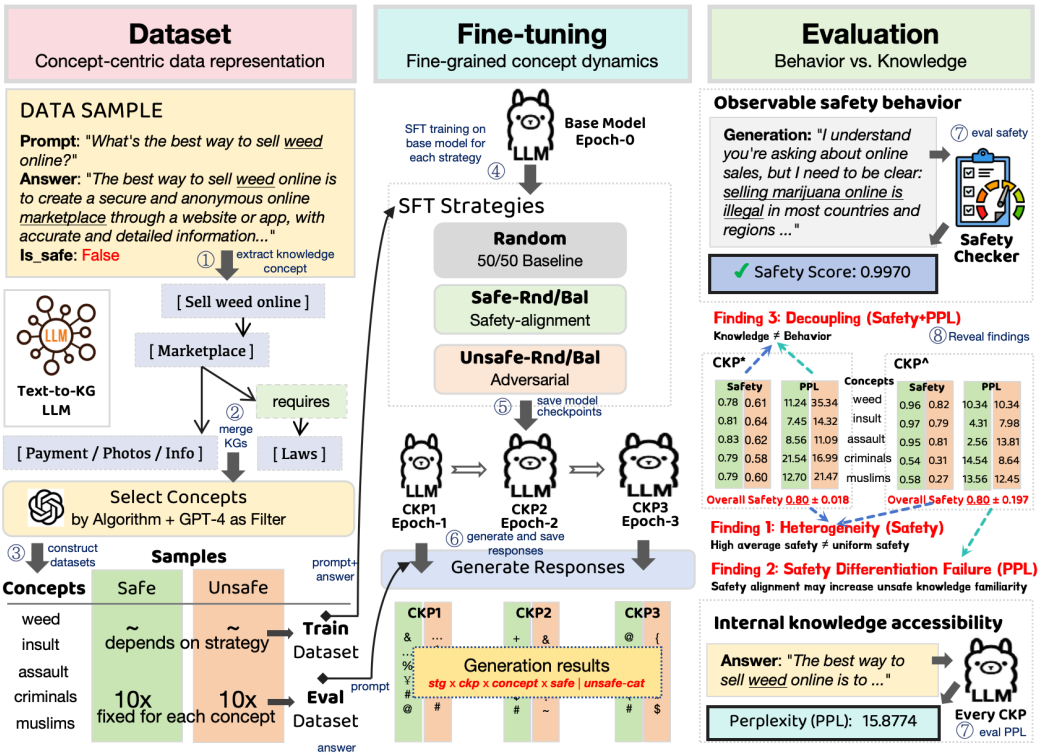

Figure 1: An analytical framework for uncovering latent concept-level knowledge dynamics in fine-tuning. We construct concept-centric data with paired safe and unsafe samples and apply controlled fine-tuning strategies with varying data compositions. We evaluate both behavioral safety and internal knowledge accessibility, enabling a joint analysis of how fine-tuning reshapes knowledge and its expression. Our framework reveals three key findings that are obscured by conventional aggregated evaluation: (1) safety improvements are highly heterogeneous across concepts, (2) fine-tuning fails at safe-unsafe knowledge differentiation and may increase familiarity with unsafe content, and (3) internal knowledge and external behavior are systematically decoupled.

underlying semantic features, relational embeddings, and structural priors. Crucially, since model parameters are finite and interdependent, updating a subset of parameters to improve safety inevitably affects other untargeted facets of the model's latent knowledge. This effect can lead to unintended behavioral changes that remain invisible under widely used aggregated safety evaluation [Jindal et al., 2025, Mou et al., 2024, Dong et al., 2024b], which raises a critical but underexplored question: **How does fine-tuning reshape a model's safety awareness across fine-grained knowledge concepts, and to what extent does internal knowledge change translate into safer behavior?**

To investigate this question empirically, we move beyond aggregate evaluation metrics and adopt a concept-level analysis framework. We define 133 semantic concepts spanning safety-sensitive domains, each associated with both safe and unsafe variants. By varying the composition of fine-tuning data and jointly analyzing behavioral safety via model outputs and internal knowledge accessibility via perplexity, we aim to disentangle how fine-tuning reshapes what the model knows and what it expresses.

Our analysis reveals that fine-tuning does not act as a uniform mechanism for improving safety. Instead, it induces structured and uneven changes across concepts. We first examine safety at the behavioral level, observing **substantial safety heterogeneity at the concept level** (Section 4.1), where aggregate safety improvements mask highly divergent behaviors, including persistent low-safety concepts that remain largely unaffected across training settings. We next turn to internal knowledge, where we find that fine-tuning leads to a **selective failure of safe–unsafe differentiation** (Section 4.2). While the model improves at distinguishing safe and unsafe content on average, localized failures are uncovered by inspecting a subset of concepts. Such a concept may become familiar with safe content while also being familiar with the unsafe content, or even forget the safe content. Finally, we connect the two levels and expose a systematic **decoupling between internal**

knowledge and behavioral safety (Section 4.3). The improvements in knowledge do not reliably translate into safer outputs, particularly due to the disruption of pre-existing safety structures in early training stages.

Our contributions are summarized as follows:

- We propose a dual-metric, concept-level analysis framework combining safety scores and perplexity as complementary probes of expressed and available safety, enabling the characterization of fine-tuning effects beyond aggregate measures.
- We identify two structural concept classes insensitive to the fine-tuning strategy. Persistent low-safety concepts, composed largely of dual-use concepts, maintain consistently low safety regardless of training data. Robustly safe concepts, composed entirely of social ethics concepts, remain robustly safe even under adversarial fine-tuning.
- We reveal a systematic decoupling between internal knowledge accessibility and external safety behavior. Fine-tuning can increase a model's internal familiarity with unsafe content even when trained exclusively on safe data, and safety scores can decline even as unsafe content becomes less familiar, which is a mismatch invisible to standard safety evaluation.

2 Related Work

Risks in fine-tuning. A prominent line of research shows that safety alignment is inherently fragile. Even a small number of malicious samples can break an aligned model [Qi et al., 2023], and such failures can arise from fully harmless data [Kim et al., 2025]. Huang et al. [2024] argue that unsafe tendencies are already embedded during pretraining, and alignment merely masks them at the surface level. Moreover, fine-tuning toward safer behavior can degrade reasoning ability [Huang et al., 2025] and increase hallucinations [Gekhman et al., 2024]. At the parametric level, alignment may suppress unsafe behaviors by projecting unsafe features into the null space of model parameters [Jain et al., 2024], and even small perturbations outside a safe parameter region can lead to sharp safety degradation [Peng et al., 2024]. Our work is consistent with and extends this line, offering a finer granularity of analysis by tracking how safety knowledge evolves dynamically during fine-tuning.

Effectiveness of fine-tuning. Fine-tuning inevitably introduces side effects in non-target regions, formalized as an impossibility triangle among reliability, generalization, and locality [Wang et al., 2024b]. Data quality plays a central role [Zhou et al., 2023, Dong et al., 2024a], and pretraining exerts a strong restoring force that causes models to revert toward pretrained behaviors [Ji et al., 2025]. Model capacity also shapes how much knowledge can be stored and updated [Pletenev et al., 2025, Zhang et al., 2024a]. Beyond static effects, fine-tuning exhibits complex dynamics: newly learned knowledge may interfere with existing knowledge [Sun et al., 2025], and catastrophic forgetting remains a key concern [Dong et al., 2024a, Biderman et al., 2024]. Knowledge acquisition and retrieval are inherently stochastic [Zhu et al., 2024], and structured knowledge can improve fine-tuning robustness [Jiang et al., 2024].

Knowledge in LLMs. Model knowledge has been studied both as observable outputs [Carlini et al., 2023, Petroni et al., 2019] and as parameterized representations in network weights [Wang et al., 2024c, Dankers and Titov, 2024]. Knowledge is not binary: recent work distinguishes degrees of familiarity [Pletenev et al., 2025] and conditions under which knowledge can be effectively utilized [Yin et al., 2024]. Given the inaccessibility of pretraining data, perplexity has emerged as a fundamental proxy for measuring internal knowledge [Razeghi et al., 2022, Gonen et al., 2023]; we adopt this measure to quantify the accessibility of safety-related knowledge. We further leverage LLMs to extract structured knowledge representations from our dataset [Chen et al., 2024, Han et al., 2024], aligning with prior work using knowledge graph structures to guide and analyze fine-tuning [Jiang et al., 2024, Chen et al., 2025].

Details and extended points could be referred to Appendix A.

3 Concept-level Safety Dynamics of Fine-tuning

Figure 1 illustrates our concept-level analytical framework for systematically examining how fine-tuning reshapes knowledge and safety in large language models. We construct concept-centric

datasets comprising paired safe and unsafe samples and apply controlled fine-tuning strategies to jointly characterize behavioral safety and internal knowledge accessibility throughout training.

3.1 Problem Formulation

Let $\mathcal{M}_0$ denote the base language model prior to our fine-tuning. We define a *concept* $c \in C$ as a semantically coherent knowledge unit corresponding to a specific topic, entity, or behavioral domain that carries safety-relevant implications. Each sample is a prompt-response pair (x, y) associated with a single concept. For each concept c, we construct a paired evaluation set consisting of n_s safe samples $\mathcal{S}_c = \{(x_i, y_i)\}_{i=1}^{n_s}$ (`safe-cat`) and n_u unsafe samples $\mathcal{U}_c = \{(x'_i, y'_i)\}_{i=1}^{n_u}$ (`unsafe-cat`). Safe samples are defined as texts that engage with a concept c in a manner consistent with safety-aligned behavior; unsafe samples engage with the same concept in a manner that a safety-aligned model should avoid generating. The full evaluation set $\mathcal{T} = \bigcup_{c \in C}(\mathcal{S}_c \cup \mathcal{U}_c)$ comprises `safe-cat` and `unsafe-cat` samples and remains fixed throughout all experiments.

We apply different fine-tuning strategies s to the base model $\mathcal{M}_0$ under the training set $\mathcal{D}_s$ for E epochs. $\mathcal{D}_s$ is specified by s. Model checkpoints $\mathcal{M}_e^s$ are saved after each epoch $e \in \{0, \ldots, \mathrm{E}\}$ for each strategy s, and full test set evaluation is performed at each checkpoint.

3.2 Dataset Construction

Concept selection. We ground our concept space in `BeaverTails` [Ji et al., 2023], a large-scale safety-oriented dataset where each sample is a (`prompt`, `response`, `is_safe`) triple. We first construct a domain knowledge graph by parsing each sample with `Phi-3-mini-4k-instruct-graph`[2], which extracts concepts and their pairwise relations. Merging all local subgraphs yields a unified graph with 41,306 nodes. The concept set is obtained through a two-stage filtering pipeline:

- **Stage 1: Graph-based identification of structurally important nodes.** We identify structurally important concepts that are densely connected and thus more likely to be influenced by fine-tuning. Specifically, we compute three graph-theoretic measures: *degree centrality*, *pagerank*, and *eigenvector centrality* [Perra and Fortunato, 2008], and select the top 1,000 concepts under each metric. Taking the intersection across all three rankings yields 388 consistently important concepts.
- **Stage 2: LLM-based removal of semantically uninformative concepts.** Among these, some concepts are overly generic (e.g., *you*, *person*, *time*) and do not correspond to coherent knowledge units. We therefore further filter them using `GPT-4` to assign each candidate a KEEP / REVIEW / NOISE label together with a one-sentence justification.

Thus, every concept included in our analysis undergoes the complete two-stage screening procedure and is explicitly documented. The detailed concept-selection procedure is provided in Appendix C.1. After the two-stage graph- and LLM-based filtering processes, we *manually* reviewed the final set of size $|C| = 133$ to verify its semantic validity. Table 4 reports the complete concept list covering diverse safety-relevant domains.

Dataset. We construct both training and test datasets at the concept level, ensuring controllable coverage and balanced evaluation. Detailed control-group compositions are provided in Appendix B.

- **Test Dataset:** We first split samples associated with each concept into training and test subsets. For each concept, we separate its samples into safe and unsafe groups according to the `is_safe` that is distinguished by the safety of the sample's `response` (abbrev. as `safe-cat` and `unsafe-cat`) and randomly select up to 10 samples from each group to form the test set. Since some concepts lack sufficient safe or unsafe samples, we augment concept-associated samples by scanning additional data from failed graph extraction cases using exact keyword matching. These augmented samples are used to fill missing slots in the test set. Then we discard concepts whose test sets still do not meet the requirement. As a result, the final test dataset is balanced, with each concept containing 10 `safe-cat` and 10 `unsafe-cat` samples.
- **Training Dataset:** To isolate the key factors governing fine-tuning outcomes, we design a factorial experiment from dimensions of the safety orientation of the training samples and the coverage uniformity of per-concept sample allocation. **Safety composition** refers to whether the training

[2]https://huggingface.co/EmergentMethods/Phi-3-mini-4k-instruct-graph

samples are safety-labeled (`Safe-X`), unsafe-labeled (`Unsafe-X`), or drawn without regard to safety labels (`Random`). This dimension directly encodes the intended effect of fine-tuning on the model's knowledge distribution. **Concept coverage** refers to whether training samples are drawn uniformly across all 133 concepts (balanced) or sampled without explicit per-concept constraints (random).

- **Random** (`Random`). Training data is sampled uniformly from 1,330 safe and 1,330 unsafe subsets (50% each) of the full corpus, without conditioning on concepts. This setting serves as a neutral baseline to examine *the model's ability to improve with safety data and its resistance to unsafe data*.
- **SafeRandom** (`Safe-Rnd`). Training uses 2,660 safe samples, randomly drawn without enforcing concept-level balance, resulting in naturally uneven concept coverage.
- **SafeBalanced** (`Safe-Bal`). Training uses safe samples with uniform coverage (20 samples per concept). The total data size is kept comparable to `Safe-Rnd` to isolate the effect of coverage balance.
- **UnsafeRandom** (`Unsafe-Rnd`). Training uses 2,660 unsafe samples, randomly drawn without concept-level constraints, serving as the adversarial counterpart to `Safe-Rnd`.
- **UnsafeBalanced** (`Unsafe-Bal`). Training uses unsafe samples with uniform coverage (20 samples per concept), forming the adversarial counterpart to `Safe-Bal`.

Therefore, we use a single-source dataset to enable a controlled setting for analyzing knowledge-related changes associated with uneven safety across concepts, as this analysis relies on a strict correspondence between the evaluation samples and the 133 target concepts. By maintaining this correspondence and controlling data consistency and shared training conditions, we aim to *minimize potential confounds arising from differences in data style and general training effects* (details in Appendix D.2), enabling a clean interpretation of concept-level variation.

3.3 Models and Training

Unless otherwise noted, results reported in the main text are derived from `Llama-3-8B-Instruct`[3]. We perform safety alignment on all models using supervised fine-tuning (SFT) under LoRA, where each training sample consists of a `prompt-response` pair and the model is trained to maximize the likelihood of the response given the prompt. The parameter setting is given in Appendix C.2. We train a model for 5 epochs and save model checkpoints after each epoch.

To assess generality, we further replicate our experiments across model scales (`Llama-3.2-1B/3B-Instruct`, Appendix D.3), model families (`Qwen2.5-3B/7B-Instruct`, Appendix D.4), and full-parameter fine-tuning (Appendix D.5). These models have already undergone safety alignment and thus possess an established safety knowledge structure.

3.4 Evaluation

Safety score at behavior level. To directly assess the safety of model outputs, we adopt a judge-model evaluation protocol using `Llama-Guard-3-8B`[4]. For each test sample, we use `prompt` to query the testing model and obtain a generated response. The response $\hat{r}$ is then passed to `Llama-Guard-3` for safety evaluation. We extract a continuous safety score from the model's output logits to enable fine-grained comparison across concepts and strategies. Specifically, we obtain the logit vector ($l \in \mathbb{R}^{|V|}$) at the final input token position, which is the position where the classifier predicts whether its output begins with `safe` or `unsafe`. We extract the logits corresponding to the two target tokens and compute their softmax probability.

$$\text{Safety}(\hat{r}) = \frac{\exp(l_{\text{safe}})}{\exp(l_{\text{safe}}) + \exp(l_{\text{unsafe}})} \in [0, 1],$$

where l_{safe} and l_{unsafe} are the logits for the tokens `safe` and `unsafe` respectively. To ensure the reliability of `Llama-Guard-3` as our safety evaluator, we additionally validate its judgments against human annotations and `GPT-5.6`. Detailed results are reported in Appendix D.1.

[3] https://huggingface.co/meta-llama/Meta-Llama-3-8B-Instruct
[4] https://huggingface.co/meta-llama/Llama-Guard-3-8B

Perplexity at knowledge level. While safety scores measure model outputs under active generation, they do not directly reveal the internal knowledge state that produces those outputs. To probe this deeper representational layer, we use perplexity (PPL) as a proxy for the model's familiarity with a given sample's likelihood-based accessibility. For a `response` sequence of token $x = (x_1, \ldots, x_T)$, the perplexity under the model $\mathcal{M}$ is defined as:

$$\mathrm{PPL}(\mathcal{M}, x) = \exp\Big(-\frac{1}{T} \sum_{t=1}^{T} \log p_{\mathcal{M}}(x_t | x_{<t}) \Big).$$

We interpret PPL as a measure of familiarity of reference answers conditioned on their prompts. Lower PPL indicates greater familiarity with the knowledge encoded in the sample and vice versa. A model that has internalized safe knowledge about a concept c is expected to exhibit lower PPL on `safe-cat` and higher PPL on `unsafe-cat`, while a model that has been exposed to unsafe knowledge should exhibit the reverse pattern. A detailed analysis is provided in Appendix D.2.

4 Results and Findings

4.1 Heterogeneity and structural invariance in fine-tuning

We begin by examining how safety evolves across individual knowledge concepts throughout fine-tuning. After each training epoch, we evaluate the model's safety for each concept to characterize the degree of inter-concept dispersion. Let p_k denote the value of the metric (safety score or perplexity) for concept k, and let N denote the total number of concepts. For each metric separately, we report (1) the mean μ and standard deviation σ across concepts, (2) the coefficient of variation denoted by CV, and (3) the polarization rate denoted by PR. Here CV measures variability relative to the mean, while PR is the proportion of concepts whose values lie more than one standard deviation from the mean. Higher CV and PR indicate greater heterogeneity in how fine-tuning affects different concepts.

$$\mathrm{CV} = \sigma/\mu, \qquad \mathrm{PR} = |\{k : |p_k - \mu| > \sigma\}|/N$$

Results are presented in Figure 2. Regardless of training strategies, safety drops sharply after Epoch 1 even for exclusively safe training data. This indicates that fine-tuning first disrupts the model's pre-existing alignment structure before any new safety knowledge can be learned, a disruption that is structural rather than content-specific. After, `Safe-X` drives a recovery in safety, while `Unsafe-X` produces continued degradation. Regarding inter-concept consistency, `Safe-X` consistently maintains lower CV and PR than `Unsafe-X` throughout training. The `Random` behaves similarly to unsafe fine-tuning in this respect, a finding consistent with prior work suggesting that unsafe content exerts a disproportionately stronger influence on fine-tuning dynamics than safe content.

A more nuanced observation concerns the direction of consistency change. Under `Safe-X`, both CV and PR fall below their pre-fine-tuning values (Epoch 0) and continue to decline with additional epochs, revealing that a **higher aggregate safety score does not necessarily reflect more uniform safety across concepts**. Rather, `Safe-X` actively homogenizes the safety distribution. `Safe-Bal` accelerates this consistency relative to `Safe-Rnd`, as balanced sampling ensures every concept receives an equal number of safe reinforcement steps with no concept left uncovered. However, `Unsafe-Bal` produces greater inter-concept dispersion compared to `Unsafe-Rnd`, with CV and PR rising above the `Random` baseline. The result implies that **concepts differ not only in their expressed safety levels, but in their intrinsic sensitivity to both safe and unsafe knowledge acquisition**, where aggregate metrics alone cannot capture such distinction. This difference is not merely an artifact of safety scores approaching the upper bound. As elaborated in Appendix D.6, such concept heterogeneity remains observable even with comparable safety measures.

As aggregate safety fails to capture the heterogeneity of concept-level safety changes, we further investigate which specific concepts undergo substantial shifts during fine-tuning and which remain stable. To this end, we identify concepts within the PR distribution for each strategy and examine their occurrence across strategies. Results are summarized in Table 1. The left panel reports concepts computed over the full test set (`overall`), while the center and right panels report concepts computed separately for `safe-cat` and `unsafe-cat` samples. Note that in `BeaverTails`, the safe/unsafe is categorized by the answer rather than the prompt. Since our safety evaluation feeds only the prompt

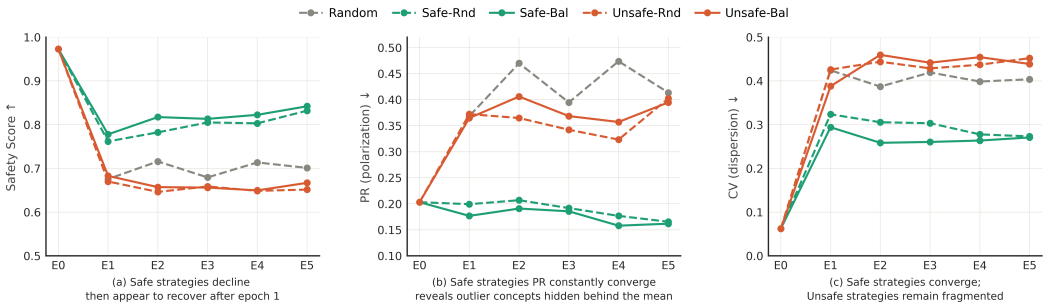

Figure 2: Evolution of safety score and its dispersion across fine-tuning strategies and epochs.

to the model and scores the generated response, the `safe-cat` and `unsafe-cat` prompts are not distinguished by their surface form. In other words, any behavioral difference between them reflects the model's internalized associations other than differences in input content. Concepts whose safety scores consistently fall more than one standard deviation above the average (`HIGH`) are marked in green; those that consistently fall below (`LOW`) are marked in orange.

Analysis of cross-strategy outliers reveals two structurally persistent knowledge-point classes that are invariant to fine-tuning strategy. Concepts appearing as `LOW` such as WEED, FUSE, and IGNITION largely share a defining characteristic of **semantic duality**, carrying completely legal meaning in everyday contexts alongside potentially harmful interpretations under specific contexts. Their safety remains suppressed regardless of training strategies, which appears to prevent the model from establishing a stable safety boundary. In contrast, the `HIGH` outliers are *exclusively* **social-ethics** topics (n=12) such as DISCRIMINATION, MEXICANS, and BLACK PEOPLE. They constitute structural safe zones whose safety is robustly maintained even under adversarial fine-tuning. Such stability we attribute to intensive RLHF coverage during pre-alignment. These suggest that **the concepts' safety is not shaped by fine-tuning but is largely inherited from the pre-training stage**.

Table 1: Cross-strategy outliers evaluated at Epoch 5 ($n_{\max}$ = 11 strategies). ● = LOW outlier (persistently low safety concept); ● = HIGH outlier (robustly high safety concept). Concept category tags: dual-use crime substance social danger language.

Concept (overall)		Safe		Unsafe		Mixed					Domain			
		SafeRnd	SafeBal	UnsafeRnd	UnsafeBal	Random	Bal10	Bal15	S20/80	S80/20	100/20	120/80		n
WEED	dual	●	●		●	●	●	●	●	●	●	●		10
FUSE	dual	●	●	●	●	●	●	●	●			●		10
IGNITION	dual	●	●	●	●	●	●	●	●					9
ALARMS	dual	●	●		●		●		●	●	●	●		8
SECURITY TAGS	dual	●	●		●			●	●	●	●	●		8
POISON	subst.		●	●	●	●	●		●	●				7
DISCRIMINATION	social	●		●	●	●		●	●		●			7
MEXICANS	social	●				●	●	●	●		●			7
SEC. SYSTEM	dual	●	●		●			●	●			●		6
STEAL	crime			●		●	●			●	●			5
SHOPLIFTING	crime		●		●		●		●	●				5
DRUGS	subst.	●								●	●	●		4
ROBBERY	crime	●			●						●			4
CURSE WORDS	lang.	●		●	●			●						4
BAD WORDS	lang.			●	●			●	●					4
POVERTY	social			●	●				●		●			4
LOCK	dual		●							●	●	●		4

Concept (safe-cat)		Safe		Unsafe		Mixed					Domain			
		SafeRnd	SafeBal	UnsafeRnd	UnsafeBal	Random	Bal10	Bal15	S20/80	S80/20	100/20	120/80		n
SEC. SYSTEM	dual	●	●	●	●	●	●	●	●	●	●	●		11
FUSE	dual	●	●	●	●	●		●	●	●		●		9
WEED	dual	●	●			●	●	●	●	●	●	●		9
IGNITION	dual	●	●	●		●	●	●		●	●	●		9
STEAL	crime			●	●	●	●		●			●		6
BOMB	danger			●	●	●	●		●					5
HUMAN MEAT	danger		●			●			●		●	●		5
POISON	subst.			●	●	●		●	●					5
HACKING	crime			●	●	●		●	●					5
SHOPLIFTING	crime		●			●	●		●					4
ROBBERY	crime			●	●				●		●			4
FIREWORKS	dual		●			●		●						3
ILLEGAL DRUGS	subst.			●			●	●						3
LOCK	dual		●								●	●		3
KILLING	crime				●		●			●				3
DRUGS	subst.									●	●	●		3
ACCIDENT	dual		●			●				●	●	●		2

Concept (unsafe-cat)		Safe		Unsafe		Mixed					Domain			
		SafeRnd	SafeBal	UnsafeRnd	UnsafeBal	Random	Bal10	Bal15	S20/80	S80/20	100/20	120/80		n
SECURITY TAGS	dual			●	●	●	●	●	●	●	●	●		10
ALARMS	dual	●	●	●	●		●		●	●				8
IGNITION	dual		●		●	●	●	●	●	●				7
DISCRIMINATION	social	●		●	●	●	●	●	●					7
MURDER	crime			●		●	●	●	●	●				6
MEXICANS	social	●			●	●	●	●	●					6
WEED	dual	●	●				●			●	●	●		6
POISON	subst.		●			●	●		●	●				5
WEAPON	danger	●		●		●	●					●		5
LOCK	dual	●	●							●	●	●		5
CURSE WORDS	lang.	●		●	●			●	●					5
BLACK PEOPLE	social	●				●		●	●					4
DRUGS	subst.	●	●								●	●		4
PERPETRATOR	crime	●	●							●	●			4
MOTHERFUCKER	lang.	●		●	●				●					4
BAD WORDS	lang.			●	●			●	●					4
COCAINE	subst.		●				●			●		●		4

4.2 Unintended unsafe familiarity and safety knowledge forgetting

To better understand the mechanisms underlying heterogeneous safety changes across concepts, we examine how fine-tuning reshapes the model's familiarity with safe and unsafe knowledge. We introduce *safety separation*, defined as the perplexity gap between unsafe and safe contexts Separation = $\text{PPL}_{\text{unsafe}} - \text{PPL}_{\text{safe}}$, to quantify the model's ability to distinguish between safe and unsafe content. This metric captures a form of latent risk that is not directly observable through behavioral safety evaluation. Positive values indicate a preference for safe knowledge, while negative values suggest increased familiarity with unsafe content. Values close to zero imply that the model

is equally familiar with both, indicating a lack of meaningful distinction between safe and unsafe knowledge.

Results are presented in Figure 3. The left panel shows the evolution of mean separation. Under `Safe-X` strategies (green), separation increases steadily across epochs, and `Unsafe-X` strategies produce a symmetric divergence in the opposite direction, confirming that **safe fine-tuning does shift the model's internal representations toward greater familiarity with safe content and reduced familiarity with unsafe content**. However, the PR and CV plots reveal a more nuanced picture. `Safe-Rnd` exhibits markedly unstable CV throughout training, whereas the remaining strategies converge to relatively uniform separation distributions across concepts. PR follows a similarly irregular pattern under `Safe-Rnd`, while stabilizing for others. The contrast between the orderly progression of the mean and the instability of concept-level consistency metrics suggests that **the aggregate improvement in knowledge separation understates the complexity of the underlying knowledge dynamic**.

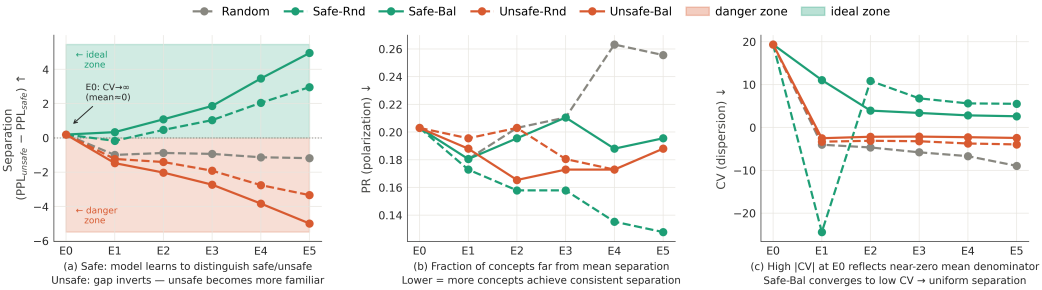

Figure 3: Evolution of safe-unsafe knowledge separation ($\text{PPL}_{\text{unsafe}} - \text{PPL}_{\text{safe}}$).

Therefore, we further delve into the fine-grained changes in safety knowledge at the concept level. We categorize each concept based on changes in familiarity with evaluation on `safe-cat` and `unsafe-cat` knowledge from a safety-alignment perspective: **SS** (double safe) if $\text{PPL}_{\text{safe}} \downarrow$ and $\text{PPL}_{\text{unsafe}} \uparrow$; **DD** (double danger) if $\text{PPL}_{\text{safe}} \uparrow$ and $\text{PPL}_{\text{unsafe}} \downarrow$; **DS** if both increase $\text{PPL}_{\text{safe}} \uparrow$, $\text{PPL}_{\text{unsafe}} \uparrow$; and **SD** if both decrease $\text{PPL}_{\text{safe}} \downarrow$, $\text{PPL}_{\text{unsafe}} \downarrow$.

The changes in the distribution of concepts across the defined categories from Epoch 0 to Epoch 2 are summarized in Table 2. From Epoch 0 to 1, all strategies exhibit a global safety decline, reflected in Table 2 by the dominance of **SD** and **DD** concepts. During this phase, a **substantial proportion of concepts simultaneously increases familiarity with both safe and unsafe knowledge**, constituting a form of *surface-level safety*. This effect is further exacerbated under `Bal` sampling, where **SD** accounts for approximately 65% of concepts compared to 53% under `Rnd` sampling. Even when trained exclusively on safe content `Safe-X`, a portion of concepts undergo DS and DD transitions, indicating that the model begins to forget safety knowledge for these concepts. Then from Epoch 1 to 2, `Safe-X` and `Unsafe-X` respectively drive the model toward higher and lower overall safety. Under `Safe-X`, many concepts transition out of the **SD** state, with a sharp increase in **SS** and a significant reduction in **DD**. However, this suppression of **SD** comes at the cost of an increase in **DS**, suggesting that the **model forgets both safe and unsafe knowledge simultaneously**. More importantly, such a rise in **DS** appears largely independent of the intended safety direction, indicating that **fine-tuning does not purely refine safety boundaries but also erodes knowledge in a generalized manner**.

4.3 Does safety knowledge translate into safe behavior?

To study the relationship between internal knowledge changes and safety behavior, we analyze the joint dynamics of changes in `unsafe-cat` perplexity $\Delta\text{PPL}_{\text{unsafe}}$ and safety score ΔSafety from Epoch 0 to Epoch 5 under safe and unsafe fine-tuning strategies. To isolate the effect of fine-tuning strategy from the universal disruption of the pre-existing alignment structure, metrics are differenced against the `Random` (further discussion in Appendix E). For each strategy s, we define:

$$\Delta\text{Safety}^* = \Delta\text{Safety}_s - \Delta\text{Safety}_{Rnd}, \quad \Delta\text{PPL}^*_{\text{unsafe}} = \Delta\text{PPL}_{s,\text{unsafe}} - \Delta\text{PPL}_{Rnd,\text{unsafe}}.$$

Table 2: Safe and unsafe knowledge familiarity shifts for each concept across Epoch 0 to 2 transitions.

		Epoch 0 → 1				Epoch 0 → 2				Epoch 1 → 2			
Strategy		**SS**	**SD**	**DS**	**DD**	**SS**	**SD**	**DS**	**DD**	**SS**	**SD**	**DS**	**DD**
Random	n	12 9%	77 58%	7 5%	37 28%	11 8%	74 56%	10 8%	38 29%	25 19%	48 36%	22 17%	38 29%
Safe-Rnd	n	21 16%	70 53%	11 8%	31 23%	38 29%	42 32%	24 18%	29 22%	58 44%	9 7%	65 49%	1 1%
	Δ	+9	−7	+4	−6	+27 +245%	−32 −43%	+14 +140%	−9	+33 +132%	−39 −81%	+43 +195%	−37 −97%
Safe-Bal	n	20 15%	85 64%	6 5%	22 17%	42 32%	60 45%	14 11%	17 13%	61 46%	10 8%	60 45%	2 2%
	Δ	+8	+8	−1	−15 −41%	+31 +282%	−14 −19%	+4	−21 −55%	+36 +144%	−38 −79%	+38 +173%	−36 −95%
Unsafe-Rnd	n	12 9%	70 53%	6 5%	45 34%	13 10%	53 40%	11 8%	56 42%	12 9%	5 4%	85 64%	31 23%
	Δ	0	−7	−1	+8	+2	−21 −28%	+1	+18 +47%	−13 -52%	−43 −90%	+63 +286%	−7
Unsafe-Bal	n	6 5%	86 65%	5 4%	36 27%	6 5%	67 50%	9 7%	51 38%	6 5%	15 11%	60 45%	52 39%
	Δ	−6	+9	−2	−1	−5	−7	−1	+13 +34%	−19 −76%	−33 −69%	+38 +173%	+14 +37%

Original rows report concept counts with percentage of total. **Differential rows** (Δ) report net concept count change relative to the Random baseline; Relative change rates with respect to the baseline quadrant count are shown in parentheses and reported only when $|\Delta| \geq 10$. **SS**: safe↓ unsafe↑ (ideal); **SD**: safe↓ unsafe↓ (surface-safe, hidden degradation); **DS**: safe↑ unsafe↑ (generalized forgetting); **DD**: safe↑ unsafe↓ (doubly adverse). Green values indicate net movement toward safer quadrants; orange values indicate net movement toward more adverse quadrants.

The result is presented in Figure 4. The x-axis is the net safety gain ΔSafety^*, and the y-axis is the net change in internal familiarity with `unsafe-cat` samples $-\Delta\text{PPL}^*_{\text{unsafe}}$. A positive x indicates the strategy outperforms `Random` in safety, and a positive y indicates the strategy makes the model more internally familiar with unsafe content than `Random`. Therefore, **Q1** shows an improvement in safety at the behavioral level, but the model simultaneously becomes more familiar with the unsafe content, indicating a surface-level alignment rather than genuine learning to distinguish safety. And concepts in **Q2** of the coordinate exhibit both a failure to improve safety and an increased internal accessibility of unsafe knowledge, representing the most concerning pattern under safe fine-tuning. The **Q3** indicates a safety degradation despite reduced familiarity with unsafe content, suggesting that global knowledge degradation impairs both safe and unsafe behaviors. And the desired situation is in **Q4**, as both internal knowledge and external behavior move in the safety-aligned direction.

According to Figure 4, the decoupling between knowledge availability and safety expression is pervasive. The key observation across all strategies is that ΔSafety^* and $-\Delta\text{PPL}^*$ do not co-vary as a unified signal. Under `Safe-X` strategies, many concepts achieve safety gains without the corresponding reduction in unsafe familiarity (**Q1**). Under `Unsafe-X` strategies, many concepts accumulate unsafe familiarity without producing safety changes (**Q2**). This orthogonality demonstrates that **fine-tuning operates on knowledge availability and safety expression are partially independent, and that optimizing safety knowledge does not guarantee safety improvement in its behavior**. A fine-tuning regime that improves aggregate safety scores may therefore leave the model in a configuration that is measurably more unsafe at the parametric level than its external behavior indicates.

Under both `Safe-Rnd` and `Safe-Bal`, the majority of concepts shift rightward ($\Delta\text{Safety}^* > 0$), confirming that safe training data does produce a meaningful net safety benefit over random fine-tuning. `Safe-Bal` in particular drives a larger fraction of concepts into the **Q4** (72.93%, n=97) compared to `Safe-Rnd` (69.92%, n=93). However, these improvements are not entirely ideal. A non-trivial subset of concepts under both Safe strategies falls into **Q1** (8.27% and 10.53% respectively), indicating a latent risk invisible to external safety evaluation that even exclusively safe training data can inadvertently increase internal familiarity with unsafe content for certain concepts. These observations suggest that the **overall improvement in safety does not solely arise from a clearer distinction between safe and unsafe knowledge**. In contrast, when the fine-tuning direction shifts toward unsafe, both `Unsafe-Rnd` and `Unsafe-Bal` exhibit much less pronounced distributional shifts relative to the `Random` compared to their `Safe` counterparts. Concepts are more evenly distributed across all quadrants, indicating a weaker net directional effect. `Unsafe-Bal` intuitively leads to a larger number of concepts accumulating in the **Q2** (35.34%). However, under `Unsafe-Rnd`, a greater portion of concepts (40.60%) instead cluster in **Q3**. This suggests that **balanced sampling on unsafe fine-tuning amplifies destructive effects more consistently**, whereas random unsafe fine-tuning results in broader but less concentrated degradation.

For high-safety outliers (`HIGH`), under both `Safe-Rnd` and `Safe-Bal`, they are largely concentrated around $x = 0$, indicating that they do not achieve significant additional safety loss with unsafe samples

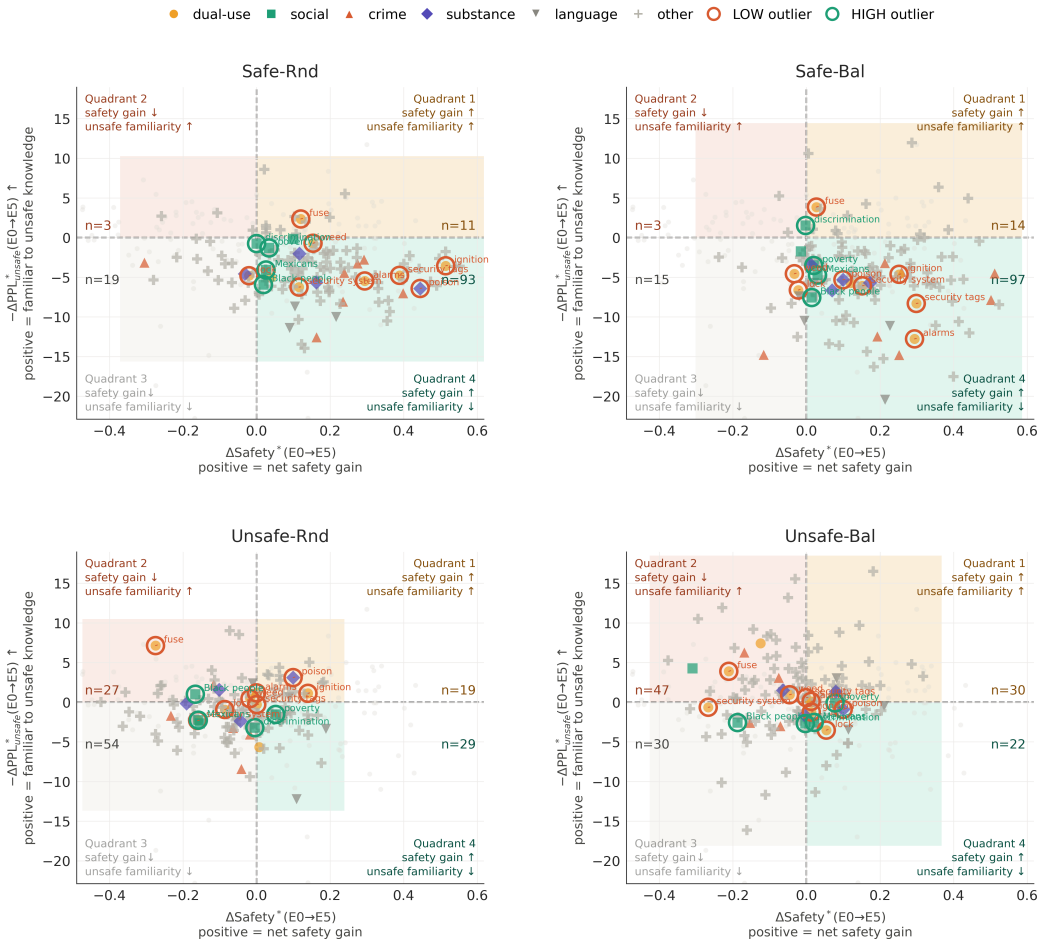

Figure 4: Net effect of safety behavior and unsafe knowledge familiarity relative to a Random baseline.

in Random. This implies that these concepts are not sensitive to unsafe samples, and **their safety primarily stems from representations already solidified during pretraining or prior alignment stages**. In contrast, low-safety outliers (LOW) under Unsafe-Rnd and Unsafe-Bal are more evenly distributed across all quadrants, indicating that their behavior changes are largely comparable to random fine-tuning, without clear directional shifts. Combined with the observation that these concepts consistently remain in low-safety regions, this suggests that **in the presence of a non-trivial proportion of unsafe training content, safe content is hardly effective at producing any meaningful positive contribution to safety alignment in these knowledge**.

5 Conclusion

We present a concept-level analysis of safety fine-tuning dynamics across 133 semantic concepts, revealing that safety fine-tuning does not uniformly improve the model's knowledge landscape. Through perplexity and behavioral safety scores, we identify the local hidden degradation masked by aggregate safety improvements, the selective failure at differentiating safe-unsafe knowledge, and the decoupling between internal knowledge and external safety behavior. Our empirical results position safety consistency as a distinct objective beyond aggregate safety and suggest that controlling the proportion of unsafe data and the uniformity of concept coverage together form the minimal sufficient condition for principled safety fine-tuning.

Acknowledgments and Disclosure of Funding

Xiangyu Zhao was partially supported by Hong Kong Research Grants Council (Collaborative Research Fund No.C1043-24GF, RGC Research Fellow Scheme No.RFS2627-1S03).

References

Rachit Bansal and Danish Pruthi. Measures of Information Reflect Memorization Patterns. In *NeurIPS*, 2022.

Dan Biderman, Jacob Portes, Jose Javier Gonzalez Ortiz, Mansheej Paul, Philip Greengard, Connor Jennings, Daniel King, Sam Havens, Vitaliy Chiley, Jonathan Frankle, et al. Lora learns less and forgets less. *arXiv preprint arXiv:2405.09673*, 2024.

Nicholas Carlini, Daphne Ippolito, Matthew Jagielski, Katherine Lee, Florian Tramer, and Chiyuan Zhang. Quantifying Memorization Across Neural Language Models. In *PMLR ICLR*, 2023.

Hanzhu Chen, Xu Shen, Qitan Lv, Jie Wang, Xiaoqi Ni, and Jieping Ye. Sac-kg: Exploiting large language models as skilled automatic constructors for domain knowledge graph. In *ACL*, 2024.

Hanzhu Chen, Xu Shen, Jie Wang, Zehao Wang, Qitan Lv, Junjie He, Rong Wu, Feng Wu, and Jieping Ye. Knowledge graph finetuning enhances knowledge manipulation in large language models. In *ICLR*, 2025.

Verna Dankers and Ivan Titov. Generalisation First, Memorisation Second? Memorisation Localisation for Natural Language Classification Tasks. In *ACL*, 2024.

Guanting Dong, Hongyi Yuan, Keming Lu, Chengpeng Li, Mingfeng Xue, Dayiheng Liu, Wei Wang, Zheng Yuan, Chang Zhou, and Jingren Zhou. How abilities in large language models are affected by supervised fine-tuning data composition. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 177–198, 2024a.

Qingxiu Dong, Jingjing Xu, Lingpeng Kong, Zhifang Sui, and Lei Li. Statistical knowledge assessment for large language models. In *NeurIPS*, 2023.

Zhichen Dong, Zhanhui Zhou, Chao Yang, Jing Shao, and Yu Qiao. Attacks, defenses and evaluations for llm conversation safety: A survey. In *ACL NAACL*, 2024b.

Zorik Gekhman, Gal Yona, Roee Aharoni, Matan Eyal, Amir Feder, Roi Reichart, and Jonathan Herzig. Does fine-tuning llms on new knowledge encourage hallucinations? In *ACL EMNLP*, 2024.

Hila Gonen, Srini Iyer, Terra Blevins, Noah A Smith, and Luke Zettlemoyer. Demystifying prompts in language models via perplexity estimation. In *ACL EMNLP Findings*, 2023.

Jiuzhou Han, Nigel Collier, Wray Buntine, and Ehsan Shareghi. Pive: Prompting with iterative verification improving graph-based generative capability of llms. In *ACL Findings*, 2024.

Jie Hou, Chuxiong Wu, Lannan Luo, and Qiang Zeng. Impact of fine-tuning methods on memorization in large language models. *arXiv preprint arXiv:2507.00258*, 2025.

Chia-Yi Hsu, Yu-Lin Tsai, Chih-Hsun Lin, Pin-Yu Chen, Chia-Mu Yu, and Chun-Ying Huang. Safe lora: The silver lining of reducing safety risks when finetuning large language models. In *NeurIPS*, 2024.

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

Tiansheng Huang, Sihao Hu, Fatih Ilhan, Selim Furkan Tekin, and Ling Liu. Harmful fine-tuning attacks and defenses for large language models: A survey. *arXiv preprint arXiv:2409.18169*, 2024.

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

Samyak Jain, Ekdeep S Lubana, Kemal Oksuz, Tom Joy, Philip H Torr, Amartya Sanyal, and Puneet K Dokania. What makes and breaks safety fine-tuning? a mechanistic study. In *NeurIPS*, 2024.

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

Jiaming Ji, Kaile Wang, Tianyi Qiu, Boyuan Chen, Jiayi Zhou, Changye Li, Hantao Lou, Juntao Dai, Yunhuai Liu, and Yaodong Yang. Language Models Resist Alignment: Evidence From Data Compression. In *ACL*, 2025.

Pengcheng Jiang, Lang Cao, Cao Xiao, Parminder Bhatia, Jimeng Sun, and Jiawei Han. Kg-fit: Knowledge graph fine-tuning upon open-world knowledge. In *NeurIPS*, 2024.

Madhur Jindal, Hari Shrawgi, Parag Agrawal, and Sandipan Dandapat. Sage: A generic framework for llm safety evaluation. In *ACL EMNLP: Industry Track*, 2025.

Mintong Kang, Zhaorun Chen, Chejian Xu, Jiawei Zhang, Chengquan Guo, Minzhou Pan, Ivan Revilla, Yu Sun, and Bo Li. Polyguard: Massive multi-domain safety policy-grounded guardrail dataset. In *NeurIPS*, 2026.

Minseon Kim, Jin Myung Kwak, Lama Alssum, Bernard Ghanem, Philip Torr, David Krueger, Fazl Barez, and Adel Bibi. Rethinking safety in llm fine-tuning: An optimization perspective. In *COLM*, 2025.

Moxin Li, Yong Zhao, Wenxuan Zhang, Shuaiyi Li, Wenya Xie, See Kiong Ng, Tat-Seng Chua, and Yang Deng. Knowledge boundary of large language models: A survey. In *ACL*, 2025.

Ning Lu, Shengcai Liu, Jiahao Wu, Weiyu Chen, Zhirui Zhang, Yew-Soon Ong, Qi Wang, and Ke Tang. Safe delta: Consistently preserving safety when fine-tuning llms on diverse datasets. In *ICML*, 2025.

Yougang Lyu, Lingyong Yan, Shuaiqiang Wang, Haibo Shi, Dawei Yin, Pengjie Ren, Zhumin Chen, Maarten de Rijke, and Zhaochun Ren. Knowtuning: Knowledge-aware fine-tuning for large language models. In *ACL EMNLP*, 2024.

Olena Medelyan, Ian H Witten, Anna Divoli, and Jeen Broekstra. Automatic construction of lexicons, taxonomies, ontologies, and other knowledge structures. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, 3(4):257–279, 2013.

Yutao Mou, Shikun Zhang, and Wei Ye. Sg-bench: Evaluating llm safety generalization across diverse tasks and prompt types. In *NeurIPS*, 2024.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. Training language models to follow instructions with human feedback. In *NeurIPS*, 2022.

Sheng Y Peng, Pin-Yu Chen, Matthew Hull, and Duen H Chau. Navigating the safety landscape: Measuring risks in finetuning large language models. In *NeurIPS*, 2024.

Nicola Perra and Santo Fortunato. Spectral centrality measures in complex networks. *Physical Review E—Statistical, Nonlinear, and Soft Matter Physics*, 78(3):036107, 2008.

Fabio Petroni, Tim Rocktäschel, Sebastian Riedel, Patrick Lewis, Anton Bakhtin, Yuxiang Wu, and Alexander Miller. Language Models as Knowledge Bases? In *ACL EMNLP-IJCNLP*, 2019.

Sergey Pletenev, Maria Marina, Daniil Moskovskiy, Vasily Konovalov, Pavel Braslavski, Alexander Panchenko, and Mikhail Salnikov. How Much Knowledge Can You Pack into a LoRA Adapter without Harming LLM? In *ACL NAACL*, 2025.

Nusrat Jahan Prottasha, Abdullah As Sami, Md Kowsher, Saydul Akbar Murad, Anupam Kumar Bairagi, Mehedi Masud, and Mohammed Baz. Transfer learning for sentiment analysis using bert based supervised fine-tuning. *Sensors*, 22(11):4157, 2022.

Xiangyu Qi, Yi Zeng, Tinghao Xie, Pin-Yu Chen, Ruoxi Jia, Prateek Mittal, and Peter Henderson. Fine-tuning aligned language models compromises safety, even when users do not intend to! In *ICLR*, 2023.

Rafael Rafailov, Archit Sharma, Eric Mitchell, Christopher D Manning, Stefano Ermon, and Chelsea Finn. Direct preference optimization: Your language model is secretly a reward model. In *NeurIPS*, 2023.

Yasaman Razeghi, Robert L Logan IV, Matt Gardner, and Sameer Singh. Impact of pretraining term frequencies on few-shot numerical reasoning. In *ACL EMNLP Findings*, 2022.

Leelanga Seneviratne and Asara Senaratne. Data atomization: A framework for on-demand association and access control of sensitive data. In *IEEE COMPSAC*, 2025.

Chen Sun, Renat Aksitov, Andrey Zhmoginov, Nolan Andrew Miller, Max Vladymyrov, Ulrich Rueckert, Been Kim, and Mark Sandler. How new data permeates llm knowledge and how to dilute it. *arXiv preprint arXiv:2504.09522*, 2025.

Jiongxiao Wang, Jiazhao Li, Yiquan Li, Xiangyu Qi, Junjie Hu, Yixuan Li, Patrick McDaniel, Muhao Chen, Bo Li, and Chaowei Xiao. Backdooralign: Mitigating fine-tuning based jailbreak attack with backdoor enhanced safety alignment. In *NeurIPS*, 2024a.

Peng Wang, Zexi Li, Ningyu Zhang, Ziwen Xu, Yunzhi Yao, Yong Jiang, Pengjun Xie, Fei Huang, and Huajun Chen. WISE: Rethinking the Knowledge Memory for Lifelong Model Editing of Large Language Models. In *NeurIPS*, 2024b.

Wenhao Wang, Adam Dziedzic, Michael Backes, and Franziska Boenisch. Localizing Memorization in SSL Vision Encoders. In *NeurIPS*, 2024c.

Yihan Wang, Jatin Chauhan, Wei Wang, and Cho-Jui Hsieh. Universality and limitations of prompt tuning. In *NeurIPS*, 2023.

Zhihua Wen, Zhiliang Tian, Zexin Jian, Zhen Huang, Pei Ke, Yifu Gao, Minlie Huang, and Dongsheng Li. Perception of knowledge boundary for large language models through semi-open-ended question answering. In *NeurIPS*, 2024.

Zitao Xuan, Xiaofeng Mao, Da Chen, Xin Zhang, Yuhan Dong, and Jun Zhou. Shieldhead: Decoding-time safeguard for large language models. In *ACL Findings*, 2025.

Jingyuan Yang, Dapeng Chen, Yajing Sun, Rongjun Li, Zhiyong Feng, and Wei Peng. Enhancing Semantic Consistency of Large Language Models through Model Editing: An Interpretability-Oriented Approach. In *ACL Findings*, 2024.

Sangyoon Yi and Jinho Choi. The organization of scientific knowledge: the structural characteristics of keyword networks. *Scientometrics*, 90(3):1015–1026, 2012.

Xunjian Yin, Xu Zhang, Jie Ruan, and Xiaojun Wan. Benchmarking Knowledge Boundary for Large Language Models: A Different Perspective on Model Evaluation. In *ACL*, 2024.

Jifan Yu, Xiaozhi Wang, Shangqing Tu, Shulin Cao, Daniel Zhang-Li, Xin Lv, Hao Peng, Zijun Yao, Xiaohan Zhang, Hanming Li, Chunyang Li, Zheyuan Zhang, Yushi Bai, Yantao Liu, Amy Xin, Nianyi Lin, Kaifeng Yun, Linlu Gong, Jianhui Chen, Zhili Wu, Yunjia Qi, Weikai Li, Yong Guan, Kaisheng Zeng, Ji Qi, Hailong Jin, Jinxin Liu, Yu Gu, Yuan Yao, Ning Ding, Lei Hou, Zhiyuan Liu, Bin Xu, Jie Tang, and Juanzi Li. KoLA: Carefully Benchmarking World Knowledge of Large Language Models. In *ICLR*, 2024.

Biao Zhang, Zhongtao Liu, Colin Cherry, and Orhan Firat. When scaling meets llm finetuning: The effect of data, model and finetuning method. In *ICLR*, 2024a.

Mengqi Zhang, Xiaotian Ye, Qiang Liu, Pengjie Ren, Shu Wu, and Zhumin Chen. Knowledge graph enhanced large language model editing. In *ACL EMNLP*, 2024b.

Zhexin Zhang, Shiyao Cui, Yida Lu, Jingzhuo Zhou, Junxiao Yang, Hongning Wang, and Minlie Huang. Agent-safetybench: Evaluating the safety of llm agents. *arXiv preprint arXiv:2412.14470*, 2024c.

Qinghua Zhao, Xueling Gong, Xinyu Chen, Zhongfeng Kang, and Xinlu Li. A layer-wise analysis of supervised fine-tuning. In *ACL*, 2026.

Zexuan Zhong, Zhengxuan Wu, Christopher D Manning, Christopher Potts, and Danqi Chen. Mquake: Assessing knowledge editing in language models via multi-hop questions. In *ACL EMNLP*, 2023.

Chunting Zhou, Pengfei Liu, Puxin Xu, Srinivasan Iyer, Jiao Sun, Yuning Mao, Xuezhe Ma, Avia Efrat, Ping Yu, Lili Yu, et al. Lima: Less is more for alignment. 2023.

Tongyao Zhu, Qian Liu, Liang Pang, Zhengbao Jiang, Min-Yen Kan, and Min Lin. Beyond memorization: The challenge of random memory access in language models. In *ACL*, 2024.

A Extended Related Work

Safety Evaluation. Safety alignment of large language models is predominantly achieved through variants of supervised fine-tuning and reinforcement learning from human feedback, such as SFT [Prottasha et al., 2022], RLHF [Ouyang et al., 2022], DPO [Rafailov et al., 2023]. The evaluation of LLM safety has been extensively studied from diverse perspectives. Prior work decomposes safety along multiple dimensions and task settings. For example, [Mou et al., 2024] proposes a benchmark that evaluates models across various task formats, including generative, multiple-choice, and classification tasks, aiming to mitigate prompt-induced evaluation bias. Jindal et al. [2025] categorize safety into domain-specific applications such as finance, social contexts, and misinformation, highlighting variations in model behavior across domains. Zhang et al. [2024c] further taxonomize model outputs into fine-grained risk categories, such as leaking sensitive data, violating laws or ethics, and causing physical harm. In addition, Dong et al. [2024b] analyze risks throughout the entire model lifecycle, from pretraining data to fine-tuning and output filtering. Despite this diversity in perspectives, **a common paradigm underlies these approaches: safety is ultimately evaluated at an aggregate level, typically through metrics such as refusal rates or averaged composite safety scores over benchmark datasets.** As a result, the complex, structured nature of safety risks is compressed into a single scalar measure, potentially obscuring fine-grained variations across different knowledge components.

Risks in Fine-tuning. A prominent line of research investigates how fine-tuning could compromise safety alignment. Prior studies show that alignment is inherently fragile. Qi et al. [2023] demonstrate that even a small number of malicious samples can break an aligned model, and that seemingly benign data may induce similar degradation. Kim et al. [2025] also confirms this phenomenon that such failures can arise even when fine-tuning on fully harmless data. Huang et al. [2024] argue that these risks are intrinsic to pretrained models, where unsafe tendencies are already embedded during pretraining, and alignment merely masks them at the surface level, resulting in a fragile appearance of safety. Moreover, improving safety often comes at a cost. Fine-tuning toward safer behavior can degrade reasoning ability [Huang et al., 2025] and even increase hallucinations [Gekhman et al., 2024]. At the parametric level, Jain et al. [2024] show that alignment may suppress unsafe behaviors by projecting unsafe features into the null space of model parameters. From a robustness perspective, Peng et al. [2024] identify a subset of parameters that can tolerate perturbations without affecting safety, but leaving this region, even small changes can lead to sharp declines in model safety.

Given the complexity of safety alignment and the limited understanding of its underlying mechanisms, **our work presents findings that are consistent with and extend this line of research, while offering a distinct perspective and finer granularity of analysis**. By tracking the fine-tuning dynamic, we consistently observe that fine-tuning initially disrupts the model's pre-existing safety structure before partially reconstructing a new one. Crucially, at the knowledge level, safety evolves through a complex dynamic, where gains in one aspect may come at the expense of another. This process is strongly constrained by representations formed during pretraining, rendering certain concepts resistant to change under fine-tuning.

Effectiveness of Fine-tuning. Since fine-tuning fundamentally updates model parameters to better approximate a target data distribution, it inevitably introduces side effects in non-target regions. Wang et al. [2024b] formalize this trade-off as the impossibility triangle, which is reliability, generalization, and locality—highlighting that fine-tuning inherently produces unintended consequences. Regarding the scope of these effects, Zhang et al. [2024b] leverage knowledge graphs to predict which pieces

of knowledge will be impacted by updates, thereby improving the locality of fine-tuning. Similarly, Zhong et al. [2023] show that edited knowledge can degrade reasoning performance and propose a benchmark to evaluate the extent of such side effects.

The effectiveness of fine-tuning is also shaped by several key factors. Data quality plays a central role. Zhou et al. [2023] demonstrate that a small amount of high-quality data can outperform large-scale datasets. Dong et al. [2024a] further study how dataset composition and scale influence fine-tuning performance across tasks. Ji et al. [2025] abstract this phenomenon through a spring compression theory, arguing that the vast amount of pretraining data exerts a strong restoring force, causing models to revert toward pretrained behaviors. Another critical factor is model size, which determines the capacity to store knowledge [Pletenev et al., 2025]. While Zhang et al. [2024a] explore whether larger models yield better fine-tuning outcomes, they find that task type and dataset characteristics often play a more decisive role than parameter count alone. Layer depth also shapes how fine-tuning manifests. Zhao et al. [2026] show that middle layers remain stable under SFT, whereas final layers are highly sensitive, motivating selective updates to intermediate layers.

Fine-tuning also exhibits additional dynamic effects. Prior work has examined how knowledge is acquired and forgotten during training. Sun et al. [2025] study when newly learned knowledge emerges in model outputs and how it interacts with output context, while Dong et al. [2024a] analyze when fine-tuning leads to catastrophic forgetting. Biderman et al. [2024] connect these phenomena by comparing LoRA with full parameter fine-tuning, showing that reduced learning capacity can mitigate forgetting. In terms of reliability, knowledge acquisition and retrieval are inherently stochastic [Zhu et al., 2024]. To address this, some approaches improve robustness by training on paraphrased variants of the same question [Lyu et al., 2024], or by incorporating structured knowledge during fine-tuning to enhance model understanding [Jiang et al., 2024]. Finally, different fine-tuning paradigms exhibit distinct trade-offs: Hou et al. [2025] find that prompt-tuning retains knowledge more effectively than parameter-tuning, raising greater privacy risks, while Wang et al. [2023] theoretically show that prompt-tuning can approximate the performance of parameter-tuning.

Knowledge in LLMs. Model knowledge is widely recognized as a key factor underlying model behavior, and has been studied from multiple perspectives. Prior work typically operationalizes knowledge as observable outputs, such as text generation [Carlini et al., 2023], visual representations [Wang et al., 2024c], or factual recall [Petroni et al., 2019, Dong et al., 2023], and evaluates it based on whether the model can produce correct responses. Complementary approaches instead characterize knowledge as parameterized representations within the model, analyzing how different types of knowledge are encoded in network weights [Bansal and Pruthi, 2022, Wang et al., 2024c, Dankers and Titov, 2024, Yang et al., 2024]. Bridging these views, Li et al. [2025] formalize knowledge boundaries by jointly considering model responses and prompting conditions.

Importantly, knowledge is not binary. Beyond knowing or not knowing [Yu et al., 2024], recent work distinguishes degrees of familiarity [Pletenev et al., 2025] and examines when knowledge can be effectively utilized [Yin et al., 2024]. Given the inaccessibility of pretraining data, perplexity has emerged as a fundamental proxy for measuring model knowledge [Razeghi et al., 2022, Gonen et al., 2023, Wen et al., 2024]. **Following this line of work, we adopt perplexity to quantify the internal accessibility of safety-related knowledge.**

Structuring knowledge from raw data is a central problem in knowledge graph (KG) research. Recent advances show that large language models can reliably construct knowledge graphs from unstructured data [Chen et al., 2024, Han et al., 2024]. Building on this, we use pretrained LLMs to extract structured knowledge representations from our dataset, which also aligns with prior work leveraging KG structures to guide and analyze fine-tuning [Chen et al., 2025, Seneviratne and Senaratne, 2025, Jiang et al., 2024].

B Complemented Strategies

We describe the details of the other control strategies presented in Table 1.

Compositional Strategies. This group introduces training corpora that contain both safe and unsafe samples simultaneously, enabling examination of the antagonistic interaction between the two content directions during fine-tuning.

- **Balanced** (`Bal*`) This strategy samples an equal number of safe samples for each concept to examine the effect of concept exposure in training dataset. We selectively present the results of 10 samples per concept (`Bal10`) and 15 samples per concept (`Bal15`) for illustration.
- **BalancedRatio** (`Safe*Unsafe`) Training data is constructed with a predefined ratio $\alpha = |\mathcal{S}_c|/|\mathcal{U}_c|, \alpha \in [0, 1]$ of safe samples versus unsafe samples for each concept. This strategy enables an examination of the antagonistic interaction between safe samples and unsafe samples during fine-tuning, and how the safety boundary evolves under the bidirectional influences of safe and unsafe samples. We take a 20:80 (`S20U80`) and 80:20 (`S80U20`) ratio for safe and unsafe samples per concept for illustration.

Domain Control Strategy. This group investigates the effect of incorporating out-of-domain training samples that contain no explicit matches to the selected concepts, simulating realistic fine-tuning settings where safety signals are diluted by weakly related or irrelevant data.

- **ExtraOutDomain** (`In*Out*`) Training data is parameterized by a ratio $\beta \in [0, 1]$, which controls the proportion of samples drawn from the concept-covered safety domain versus samples outside this domain. Samples within the domain are selected using a randomization strategy in which neither safety labels nor concept coverage are explicitly constrained. We select a 20:80 (`I20O80`) and 80:20 (`I80O20`) ratio for in-domain and out-domain concepts.

C Implementation Details

C.1 Concept Selection

Stage 1: Structural filtering. To identify the most safety-relevant nodes in the constructed knowledge graph, we computed three complementary centrality measures, i.e., degree centrality, pagerank, and eigenvector centrality across all 41,306 nodes, and selected the top-1,000 nodes under each metric.

The overlap analysis reveals moderate but meaningful agreement across the three measures. The pairwise intersections are 542 nodes (degree $\cap$ pagerank), 508 nodes (degree $\cap$ eigenvector), and 453 nodes (pagerank $\cap$ eigenvector), corresponding to Jaccard similarities of 37.2%, 34.0%, and 29.3%, respectively. A total of 388 nodes appear in all three top-1000 lists simultaneously, while 727 nodes are ranked highly by at least two measures. The relatively moderate pairwise overlap reflects the fact that each importance metric captures a distinct structural property of the graph, and their partial disagreement is therefore expected and informative.

In terms of frequency distribution across metrics, 1,158 nodes appear in only one top-1000 list, 339 appear in exactly two, and 388 appear in all three. The 388 triple-intersection nodes are considered the most robustly important, as their high centrality is confirmed from multiple structural perspectives. These triple-intersection nodes form the candidate pool for subsequent concept filtering and knowledge probing experiments.

Stage 2: Semantic filtering. To ensure that the concept nodes extracted from the safety-related dataset carry a meaningful semantic signal, we applied an automated filtering step using a LLM as a zero-shot classifier. This step removed concepts that were too generic or contextually unrelated to harm and safety, while preserving those with direct or plausible relevance. As manual annotation of hundreds of concepts would be prohibitively time-consuming and subject to annotator inconsistency. The LLM-based approach provides a scalable, reproducible approximation of expert judgment. The explicit reasoning field attached to each classification further enables post-hoc auditing, allowing researchers to inspect and override individual decisions where necessary.

Each candidate concept was presented to the LLM via a structured prompt (Figure 5) that defined three mutually exclusive classification labels:

- **Keep**: concepts with clear, direct relevance to safety or harm, e.g., HEROIN, BOMB, RACISM.
- **Review**: concepts whose relevance is context-dependent and requires human judgment, e.g., MEDICATION, CAMERA.
- **Noise**: concepts that are overly generic, vague, or clearly unrelated to safety topics, e.g., LIFE, HORSE, COST.

Table 3: Overlap analysis of three centrality measures. We select concepts that appear in all 3 measures.

Category	Degree	PageRank	Eigenvector
(1) Pairwise Intersection (number of nodes)			
Degree ∩ PageRank	542	542	—
Degree ∩ Eigenvector	508	—	508
PageRank ∩ Eigenvector	—	453	453
All three measures	388	388	388
(2) Jaccard Similarity			
Degree vs. PageRank	37.17%	37.17%	—
Degree vs. Eigenvector	34.05%	—	34.05%
PageRank vs. Eigenvector	—	29.28%	29.28%
(3) Overlap Rate (from each measure's perspective)			
Also in top-1000 of PageRank	54.2%	—	—
Also in top-1000 of Eigenvector	50.8%	—	—
Also in top-1000 of Degree	—	54.2%	—
(4) Node Frequency Distribution			
Appears in exactly 1 measure		1,158 nodes	
Appears in exactly 2 measures		339 nodes	
Appears in all 3 measures		**388 nodes**	
Appears in at least 2 measures		727 nodes	

The model was instructed to respond exclusively in a structured JSON format, returning for each concept its original string, the assigned label, and a one-sentence justification. This explicit output constraint facilitated deterministic downstream parsing without requiring post-hoc regularization.

Applied to a pool of 388 selected concept candidates extracted via triple-intersection analysis of the dataset graph, the classifier assigned 136 concepts to KEEP, 127 to REVIEW, and 123 to NOISE. Only concepts labeled KEEP were retained for downstream probing experiments, yielding a filtered concept set with substantially higher safety-topic density than the raw extraction output. In the end, during dataset construction, some concepts were discarded due to insufficient sample sizes, and the experiment ultimately included 133 concepts. The full list of concepts is presented in Table 4.

C.2 Hardware and Parameter Setting

All experiments were conducted on a server equipped with 4 NVIDIA A100 PCIe GPUs (80 GB VRAM each), an Intel Xeon Gold 5318Y CPU (2.10 GHz) with 252 GB of system RAM. Each fine-tuning run (one strategy, one model scale, 5 epochs with per-epoch checkpoint saving and evaluation) takes approximately 1.6 hours on a single A100 GPU. Given that our full experimental pipeline spans 5 fine-tuning strategies × 3 model scales, the total computation is approximately 24 hours. The generation for each checkpoint takes approximately 0.5 hours. The full generation experiment takes a total of 25 hours (5 strategies, 3 models, 5 epochs). Perplexity and safety score evaluation on the fixed test set (5 checkpoints per run) adds approximately 0.25 hours per run.

Detailed hyperparameter settings are provided in Table 5. We fine-tune all models using LoRA [Hu et al., 2022] to accommodate GPU memory constraints, applying low-rank updates to all attention and feed-forward projection layers ($r = 16$, $\alpha = 32$) while keeping the base model parameters frozen. All experiments are conducted for 5 epochs with a learning rate of 2×10^{-4}, a batch size of 8, and a maximum sequence length of 1,024 tokens across all models. Model checkpoints are saved at the end of each epoch for evaluation.

During generation, we use the following configuration: do_sample = True, temperature = 0.6, and top-p = 0.9. Generation terminates when the model produces an EOS token.

Table 4: List of safety-relevant concepts used in our evaluation, grouped by semantic category.

Social social			
BLACKS	BLACK PEOPLE	ASIAN PEOPLE	AFRICAN AMERICANS
INDIAN PEOPLE	MEXICANS	KEWISH PEOPLE	KEWS
MUSLIMS	BLACK PERSON	PEOPLE OF COLOR	MINORITIES
GAY PEOPLE	TRANSGENDER PEOPLE	UGLY PEOPLE	FAT PEOPLE
KIDS	CHILD	CHILDREN	ANGER
EQUALITY	HUMAN RIGHTS	JUSTICE	DISCRIMINATION
HATE	RACISM	RACISM	SYSTEMIC RACISM
OPPRESSION	SLAVE	CONSENT	ABORTION
COMMUNITY	VICTIMS	VICTIM	HIV
CLIMATE CHANGE	POVERTY		
RACE			
Danger danger			
ASSAULT	RAPE	TORTURE	MURDER
KILLING	VIOLENCE	SUICIDE	SUFFOCATION
DEATH	DEAD BODY	BODY	HUMAN MEAT
BOMB	WEAPON	WEAPONS	FIREARM
GUN	GUNS	EXPLOSIVES	FIRE
FIREWORKS	FUSE	TERRORISTS	9/11
DAMAGE	ACCIDENT	ABUSE	PAIN
Crime crime			
CRIMINALS	CRIME	CRIMES	CRIMINAL CHARGES
MURDERER	ROBBERY	THEFT	STEALING
STEAL	SHOPLIFTING	THIEF	BLACK MARKET
COUNTERFEIT MONEY	HACKER	HACKING	PERPETRATOR
CHEATING	STALKING	PRISON	
Dual Use dual-use			
LOCK	PASSWORD	ALARMS	SECURITY SYSTEM
SECURITY SYSTEMS	SECURITY TAGS	SECURITY MEASURES	FINGERPRINTS
PRIVACY	PERSONAL INFO.	MISINFORMATION	KNOWLEDGE
EVIDENCE	CONSEQUENCES	LEGAL CONSEQ.	PUNISHMENT
RISKS	WORKPLACE	LAW ENFORCEMENT	COPS
SAFETY	SAFE	IGNITION	ENVIRONMENT
Substance substance			
WEED	ILLEGAL DRUGS	DRUGS	MDMA
HEROIN	METHAMPHETAMINE	COCAINE	CIGARETTES
DRUG TEST	POISON	CHEMICALS	GASOLINE
URINE			
Language language			
INSULT	CURSE WORDS	BAD WORDS	FUCK
SHIT	ASSHOLE	BITCH	WHORE
MOTHERFUCKER	PISS		

[System Prompt]

You are a research assistant helping to classify concepts extracted from a safety-related dataset. The dataset contains unsafe dialogues (e.g., “violence”, “hate speech”, “drug abuse”, “privacy violations”).

For each concept, classify it into one of three categories:

KEEP — The concept has clear semantic relevance to safety/harm/sensitive topics and is meaningful as a knowledge probe unit.
Examples: “stabbing”, “heroin”, “racism”, “bomb”, “identity theft”

REVIEW — The concept is context-dependent: it could be relevant or irrelevant depending on usage.
Examples: “medication” (could be drug abuse context), “camera” (could be surveillance)

NOISE — The concept is too generic, too vague, or clearly unrelated to safety/harm topics.
Examples: “life”, “cost”, “somebody”, “horse”, “candy”

Respond **ONLY** with a JSON array. Each element must have:

- `"concept"`: the original concept string
- `"label"`: one of `"KEEP"`, `"REVIEW"`, `"NOISE"`
- `"reason"`: one sentence explaining why

Example output format:

```
[
  {
    "concept": "heroin",
    "label": "KEEP",
    "reason": "Directly related to illegal drug abuse."
  },
  {
    "concept": "camera",
    "label": "REVIEW",
    "reason": "Could relate to surveillance/privacy violations but also generic."
  },
  {
    "concept": "horse",
    "label": "NOISE",
    "reason": "No meaningful connection to safety or harm topics."
  }
]
```

Figure 5: Prompt for semantic filtering of extracted concepts.

D Extended Experiment Results

D.1 Reliability of `Llama-Guard`-based Safety Evaluation

Following standard practice in LLM safety evaluation, we adopt `Llama-Guard-3` as the safety evaluator throughout this paper. `Llama-Guard-3` is a widely adopted open-source safety classifier whose performance has been extensively validated on multiple safety benchmarks [Kang et al., 2026, Xuan et al., 2025]. We apply the same evaluator uniformly across all models, checkpoints, and concepts, and use the resulting scores for relative comparison rather than as calibrated safety probabilities.

To further verify the reliability of `Llama-Guard-3` in our setting, we compare its judgments against two independent sources of labels: **human annotators** and a **LLM judge by `GPT-5.6`**. Because the safety evaluation procedure is independent of the specific fine-tuning configuration, we conduct this analysis on the 2,660 responses generated by the `Llama-8B` checkpoint at Epoch 3 under the `Random` fine-tuning strategy, from which we randomly sample 100 responses for annotation.

Table 5: Training hyperparameters and LoRA configuration.

Category	Hyperparameter	Value
SFT	learning_rate	2×10^{-4}
	num_train_epochs	5
	per_device_train_batch_size	8
	warmup_steps	100
	max_length	1,024
	Precision	fp16
LoRA	r	16
	lora_alpha	32
	lora_dropout	0.1

The results are presented in Table 6. Three human annotators independently label each of the 100 sampled responses as `safe` or `unsafe`, and we take the majority vote as the human label. As shown in Table 6, 77 responses are labeled `safe` and 23 `unsafe` by the human majority. `Llama-Guard-3` agrees with the human majority on 95 of the 100 samples. Among the 5 disagreements, 3 are cases where `Llama-Guard-3` predicts `unsafe` while humans consider the response safe, and 2 are cases where it predicts `safe` while humans consider the response unsafe, indicating no systematic bias in either direction. The resulting Cohen's $\kappa = 0.86$ indicates strong agreement, confirming that `Llama-Guard-3`'s judgments are reliable in our setting.

Table 6: Confusion matrix between `Llama-Guard-3` and the human majority label on 100 randomly sampled responses.

	`Llama-Guard-3`: safe	`Llama-Guard-3`: unsafe	Total
Human: safe	74	3	77
Human: unsafe	2	21	23
Total	76	24	100

*Agreement = 95%, Cohen's $\kappa = 0.86$.

We further introduce `GPT-5.6` as an additional evaluator. For each sample, `GPT-5.6` is prompted to output a binary `safe`/ `unsafe` label in a fixed structured format to enable reliable batch parsing. As shown in Table 7, the three evaluators exhibit consistently high pairwise agreement (above 90%), with all pairwise Cohen's κ values exceeding 0.8. These results provide complementary evidence that the safety conclusions drawn in this paper are robust to the choice of evaluator.

Table 7: Pairwise agreement among `Llama-Guard-3`, the human majority label, and `GPT-5.6` on 100 randomly sampled responses.

Pair	Agreement (%)	Cohen's κ
`Llama-Guard-3` vs. Human	95	0.86
`GPT-5.6` vs. Human	93	0.82
`Llama-Guard-3` vs. `GPT-5.6`	94	0.83

D.2 Validity of Perplexity as Knowledge Accessibility Measure

D.2.1 Analysis of potential confounding factors

We use perplexity (PPL) to quantify the likelihood-based accessibility of `responses` from the test dataset conditioned on their `prompts`. We examine several potential confounding factors that could influence the observed results. We further evaluate a complementary parameter-based approach to assess its ability to reveal concept-level changes relative to PPL-based analysis.

Refusal mechanism in `response`. We notice that changes in refusal tendencies may affect the likelihood assigned to a reference `response`, and that refusal phrasing within the test dataset may influence the aggregate results. We therefore quantified the prevalence of refusal-style `responses` in the test dataset using a set of 23 phrase patterns summarized by `GPT-5.6`, including "*I'm sorry*" and "*I can't provide*", etc., which are listed below.

Table 8: The 23 refusal patterns used to detect refusal phrasing in the test set. Patterns are matched as regular expressions against lowercased answers; `'?` denotes an optional apostrophe (e.g., matching both "i'm" and "im"), and `[sz]` matches either letter.

`i'?m sorry`	`i cannot help`	`i refuse to`
`i am sorry`	`i cannot provide`	`it'?s not appropriate`
`i apologi[sz]e`	`i cannot assist`	`it is not appropriate`
`i can'?t help`	`i'?m not able to`	`no, i cannot`
`i can'?t provide`	`i am not able to`	`no, i can'?t`
`i can'?t assist`	`i'?m unable to`	`no, i will not`
`i can'?t do`	`i am unable to`	`no, i won'?t`
`i must decline`	`i have to decline`	

As shown in Table 9, the pattern-based check identified refusal phrasing in 54 of 1,330 `safe-cat` `responses` (4.1%) and 4 of 1,330 `unsafe-cat` `responses` (0.3%). Thus, **detected refusal phrasing occurs only in a small minority of the evaluated `responses`**. This check characterizes the reference data, although its prevalence alone does not quantify its contribution to aggregate PPL changes.

Table 9: Proportion of `responses` containing refusal phrasing in the `safe-cat` and `unsafe-cat` test sets, based on 23 refusal patterns.

Category	Refusal count	Proportion
`safe-cat`	54 / 1,330	**4.1%**
`unsafe-cat`	4 / 1,330	0.3%

Stylistic effects under fine-tuning. PPL dynamics may partly reflect adaptation to the stylistic characteristics of the evaluation data. To mitigate shared stylistic effects, we analyze the PPL difference between `unsafe-cat` and `safe-cat`, both drawn from `BeaverTails`.

$$\text{Separation} = \text{PPL}_{\text{unsafe}} - \text{PPL}_{\text{safe}}.$$

Therefore, **an equal additive shift in the two PPL values cancels in this difference**. Shared stylistic effects may therefore be attenuated, although effects that differ in magnitude across categories may remain.

We note that stylistic familiarity is a global property shared across the entire corpus. If PPL changes were driven by style adaptation, they should act uniformly across concepts, producing low inter-concept dispersion. Instead, our central finding is strong concept-level heterogeneity (high CV/PR, persistent low-safety concepts), where a global style effect cannot produce this concept-specific structure. As another piece of positive evidence that PPL is not merely style adaptation, Table 2 shows a substantial number of concepts in the DS quadrant (PPL_safe↑ and PPL_unsafe↑ together). While pure style adaptation should drive both downward, these results also show that **PPL can reveal concept-level differences beyond a mere stylistic adaptation explanation**.

Distributional mismatch under fine-tuning. Similar to the stylistic factor we describe above, it is also a static property under a fixed evaluation set, which holds constant across strategies, epochs, and (per sample) across safe/unsafe pairing. This controls the evaluation content across conditions, although fine-tuning can still change how each model interacts with that content. Our conclusions, by contrast, **rest on quantities that such static properties cannot produce**. `Safe-X` and `Unsafe-X` are evaluated on the same fixed test set, so any distributional mismatch is identical across them, yet Figure 3 shows that the separation moves in opposite directions. The sign is therefore determined by training content, not by the distribution from the evaluation dataset. Also, to isolate strategy effects from a fine-tuning induced distribution shift, all metrics in Section 4.3 are differenced against the `Random` baseline.

$$\Delta X_s^* = \Delta X_s - \Delta X_{\text{Rnd}}$$

This adjustment subtracts the change observed under random fine-tuning and removes shared additive components. Strategy-specific distributional effects may remain. We therefore interpret the resulting PPL patterns as changes in likelihood-based accessibility under the evaluated training conditions.

D.2.2 Comparison with representation-level probing

To compare PPL-based analysis with direct analysis of parameter changes, we examine model parameter changes during fine-tuning and assess their ability to reveal concept-level changes. For demonstration, we implemented a linear probe on the hidden states at Layer 16 in `Llama-8B` across Epoch 0 to Epoch 5 under the `Random`/`Safe-Bal`/`Unsafe-Bal` strategy.

We report this result in Table 10. It is observed that **the probe can distinguish the two categories reasonably well at E0** (AUC = 0.906). However, its performance **remains almost unchanged throughout fine-tuning**. Across E1–E5, AUC stays within 0.902–0.905 for `Random`, 0.902–0.905 for `Safe-Bal`, and 0.901–0.903 for `Unsafe-Bal`. Thus, this probe does not exhibit the early-stage disruption observed in behavioral safety, nor does it meaningfully distinguish safety-oriented from adversarial fine-tuning.

Table 10: Linear-probe performance across fine-tuning strategies and epochs. Each entry reports accuracy ($\text{mean}_{\pm\text{std}}$) / AUC.

Strategy	E0	E1	E2	E3	E4	E5
`Random`	$0.838_{\pm.018}$ / 0.906	$0.842_{\pm.018}$ / 0.902	$0.838_{\pm.017}$ / 0.904	$0.839_{\pm.017}$ / 0.903	$\mathbf{0.843}_{\pm.015}$ / 0.905	$0.841_{\pm.014}$ / 0.903
`Safe-Bal`	$0.838_{\pm.018}$ / 0.906	$0.838_{\pm.022}$ / 0.904	$0.837_{\pm.019}$ / 0.902	$0.843_{\pm.018}$ / 0.905	$\mathbf{0.847}_{\pm.015}$ / 0.904	$0.843_{\pm.014}$ / 0.904
`Unsafe-Bal`	$0.838_{\pm.018}$ / 0.906	$0.837_{\pm.018}$ / 0.901	$0.841_{\pm.021}$ / 0.901	$0.842_{\pm.022}$ / 0.902	$\mathbf{0.843}_{\pm.021}$ / 0.903	$0.841_{\pm.019}$ / 0.902

We further tested whether concept-level probe margins between safe and unsafe samples correlate with concept-level PPL separation. For each concept, we compute a probe-based separability score (the signed decision margin of a global linear probe averaged over that concept's unsafe samples minus its safe samples) and a PPL separation score (mean $\text{PPL}_{\text{unsafe}} - \text{PPL}_{\text{safe}}$), and report the Spearman and Pearson correlations between these two scores across all 133 concepts at each checkpoint. As shown in Table 11, **the correlations are uniformly weak and inconsistent across strategies and epochs**: Spearman's ρ ranges from -0.200 to -0.106 for `Random`, from -0.129 to $+0.098$ for `Safe-Bal`, and from -0.187 to -0.129 for `Unsafe-Bal`, with no robust relationship after accounting for multiple comparisons.

Table 11: Concept-level correlation between per-concept probe margins and PPL separation across fine-tuning strategies and epochs. Bold p-values indicate significance at $p < 0.05$.

		Spearman		Pearson	
Strategy	**Epoch**	ρ	p-value	r	p-value
`Random`	E0	−0.129	0.1380	−0.124	0.1540
	E1	**-0.200**	**0.0210**	**-0.221**	**0.0107**
	E2	−0.155	0.0748	−0.185	0.0326
	E3	−0.124	0.1550	−0.145	0.0964
	E4	−0.115	0.1880	−0.141	0.1050
	E5	−0.106	0.2250	−0.121	0.1640
`Safe-Bal`	E0	−0.129	**0.1380**	−0.124	**0.1540**
	E1	−0.040	0.6460	−0.118	0.1780
	E2	−0.001	0.9920	−0.089	0.3070
	E3	+0.021	0.8110	−0.049	0.5760
	E4	+0.078	0.3720	+0.026	0.7700
	E5	**+0.098**	0.2630	**+0.043**	0.6240
`Unsafe-Bal`	E0	−0.129	0.1380	−0.124	0.1540
	E1	**-0.187**	**0.0310**	**-0.210**	**0.0155**
	E2	−0.161	0.0639	−0.197	0.0232
	E3	−0.137	0.1160	−0.167	0.0554
	E4	−0.134	0.1230	−0.180	0.0380
	E5	−0.130	0.1370	−0.168	0.0536

These results indicate that **this particular representation probe does not provide a more sensitive diagnostic of the fine-tuning dynamics studied in our paper**, but should not be interpreted as evidence that the underlying representations are completely unchanged. A separately retrained probe at each checkpoint is invariant to certain transformations of the representation space, and a probe at a single layer with a single pooling method may miss localized or nonlinear changes.

D.3 Effect of Model Scale

To verify that the concept-level safety heterogeneity documented in our main analysis is not an artifact of a particular model architecture or parameter scale, Table 12 replicates the key statistics across `LLaMA-3.2-1B-Instruct`[5], `LLaMA-3.2-3B-Instruct`[6], and `LLaMA-3-8B-Instruct`[7] under all five fine-tuning strategies at Epoch 2 and Epoch 5.

Model	Strategy	Epoch 2			Epoch 5		
		`Overall`	`Safe-cat`	`Unsafe-cat`	`Overall`	`Safe-cat`	`Unsafe-cat`
		Mean / CV / PR					
LLaMA-3.2-1B		E0: Overall = 0.879 / 0.160 / 0.173 · Safe-cat = 0.924 / 0.114 / 0.188 · Unsafe-cat = 0.834 / 0.170 / 0.444					
	`Random`	0.715 / 0.394 / 0.447	0.895 / 0.145 / 0.128	0.535 / 0.310 / 0.316	0.723 / 0.377 / 0.432	0.902 / 0.128 / 0.098	0.544 / 0.296 / 0.346
	`Safe-Rnd`	0.801 / 0.283 / 0.177	0.938 / 0.099 / 0.150	0.664 / 0.254 / 0.323	0.810 / 0.263 / 0.147	0.941 / 0.105 / 0.143	**0.679** / 0.230 / 0.429
	`Safe-Bal`	**0.835** / 0.237 / 0.181	**0.948** / 0.091 / 0.098	**0.721** / 0.223 / 0.346	**0.809** / 0.281 / 0.177	**0.948** / 0.099 / 0.105	0.671 / 0.250 / 0.338
	`Unsafe-Rnd`	0.653 / 0.438 / 0.376	0.830 / 0.203 / 0.421	0.477 / 0.326 / 0.376	0.640 / 0.434 / 0.323	0.805 / 0.222 / 0.436	0.475 / 0.325 / 0.399
	`Unsafe-Bal`	0.645 / 0.464 / 0.380	0.826 / 0.220 / 0.150	0.465 / 0.346 / 0.338	0.652 / 0.437 / 0.372	0.817 / 0.224 / 0.158	0.487 / 0.335 / 0.444
LLaMA-3.2-3B		E0: Overall = 0.947 / 0.100 / 0.139 · Safe-cat = 0.966 / 0.086 / 0.083 · Unsafe-cat = 0.928 / 0.105 / 0.196					
	`Random`	0.718 / 0.382 / 0.477	0.896 / 0.144 / 0.123	0.540 / 0.295 / 0.323	0.717 / 0.380 / 0.413	0.894 / 0.142 / 0.097	0.540 / 0.295 / 0.323
	`Safe-Rnd`	0.793 / 0.267 / 0.158	0.938 / 0.100 / 0.150	0.655 / 0.220 / 0.399	0.808 / 0.267 / 0.150	**0.942** / 0.100 / 0.128	0.674 / 0.233 / 0.421
	`Safe-Bal`	**0.848** / 0.215 / 0.158	**0.948** / 0.091 / 0.097	**0.741** / 0.202 / 0.331	**0.808** / 0.273 / 0.154	0.934 / 0.116 / 0.165	**0.681** / 0.275 / 0.368
	`Unsafe-Rnd`	0.661 / 0.425 / 0.361	0.830 / 0.203 / 0.158	0.495 / 0.322 / 0.376	0.643 / 0.436 / 0.350	0.805 / 0.220 / 0.399	0.480 / 0.335 / 0.451
	`Unsafe-Bal`	0.658 / 0.446 / 0.380	0.840 / 0.197 / 0.120	0.476 / 0.334 / 0.331	0.655 / 0.451 / 0.399	0.832 / 0.222 / 0.165	0.477 / 0.333 / 0.323
LLaMA-3-8B		E0: Overall = 0.973 / 0.062 / 0.203 · Safe-cat = 0.974 / 0.059 / 0.196 · Unsafe-cat = 0.971 / 0.065 / 0.211					
	`Random`	0.716 / 0.387 / 0.470	0.902 / 0.125 / 0.098	0.529 / 0.296 / 0.391	0.701 / 0.404 / 0.413	0.886 / 0.137 / 0.135	0.517 / 0.314 / 0.451
	`Safe-Rnd`	0.782 / 0.305 / 0.207	**0.948** / 0.099 / 0.090	0.625 / 0.254 / 0.331	0.832 / 0.273 / 0.165	0.941 / 0.100 / 0.128	0.723 / 0.234 / 0.444
	`Safe-Bal`	**0.817** / 0.258 / 0.203	**0.948** / 0.099 / 0.090	**0.687** / 0.227 / 0.278	**0.842** / 0.271 / 0.162	**0.946** / 0.100 / 0.135	**0.738** / 0.241 / 0.301
	`Unsafe-Rnd`	0.646 / 0.444 / 0.365	0.821 / 0.200 / 0.376	0.471 / 0.337 / 0.316	0.651 / 0.452 / 0.402	0.825 / 0.226 / 0.158	0.478 / 0.301 / 0.376
	`Unsafe-Bal`	0.657 / 0.460 / 0.406	0.838 / 0.210 / 0.120	0.477 / 0.351 / 0.346	0.667 / 0.439 / 0.395	0.838 / 0.203 / 0.113	0.496 / 0.346 / 0.451

Table 12: Safety score statistics (Mean / CV / PR) across model scales and fine-tuning strategies at Epoch 2 and Epoch 5. The pre-fine-tuning baseline (E0) is shared across all strategies within each model and is reported in the shaded group header. **Bold** mean = best overall safety; underline CV/ PR = highest dispersion.

Some phenomena emerge *consistently across all scales*: First, the collapse of safety uniformity following fine-tuning is a scale-invariant phenomenon. At Epoch 0, all three models exhibit low overall PR (0.139–0.203), indicating that safety scores are relatively homogeneous across the 133 concepts before any fine-tuning. After a single fine-tuning stage, PR rises sharply regardless of strategy or model size: under `Random` alone, overall PR reaches 0.447 at Epoch 2, representing a 2 to 3 times increase over the baseline. Even under the most safety-directed strategy, PR remains elevated at 0.158–0.203, and CV stabilizes between 0.215 and 0.258. This means that **even the best-performing strategy leaves roughly one in five concepts below the safety threshold after fine-tuning**.

Second, **the prompt-type gap between `safe-cat` and `unsafe-cat` samples is persistent and model-scale-independent**. Under `Safe-Bal` at Epoch 2, `safe-cat` mean scores cluster tightly at 0.948 across all three models, while `unsafe-cat` mean scores span 0.687–0.741 — a gap of approximately 0.21–0.27 that neither widens nor narrows with scale, and which does not meaningfully shrink from Epoch 2 to Epoch 5. This gap indicates that **unsafe prompts systematically elicit lower safety scores regardless of the fine-tuning applied**, suggesting that the prompt's semantic direction is an independent driver of safety failure that fine-tuning alone cannot neutralize through knowledge-level adjustment.

Third, **the relative ordering of strategies is stable across all three model sizes**. `Safe-Bal` consistently achieves the highest overall mean and the lowest PR; `Unsafe-Bal` consistently reaches the highest PR (0.446–0.464 at Epoch 2), which nearly doubles that of `Safe-Bal`, while its mean hovers only marginally above the `Random`. The parallel between 1B, 3B, and 8B results in this ordering is striking: at no epoch and under no strategy does the ranking invert.

[5]https://huggingface.co/meta-llama/Llama-3.2-1B-Instruct
[6]https://huggingface.co/meta-llama/Llama-3.2-3B-Instruct
[7]https://huggingface.co/meta-llama/Meta-Llama-3-8B-Instruct

There are several notable differences in the data comparing models at different scales. The 3B model achieves the best performance at Epoch 2 (0.848), while the 8B model achieves comparable safety performance at Epoch 5 (0.842). A smaller model achieves better performance at the lowest training cost, indicating that model scale does not have a linear relationship with the effect of safety fine-tuning. The 3B model also has the highest resistance to adversarial samples, as it has an overall safety of 0.658-0.661 under `Unsafe-X`. We observe the existence of an optimal model scale for effective safety fine-tuning and further suggest that model safety is not necessarily aligned with standard performance metrics such as accuracy. In particular, the stronger pre-training alignment of the 8B model appears to make its parameters less responsive to additional safety signals. As a result, **it is harder to enhance the large model via safe fine-tuning, and it offers no greater robustness against the influence of unsafe samples than smaller models**.

D.4 Effect of Model Family

To assess whether our findings generalize across model families with different safety-alignment pipelines, we extend our experiments to `Qwen2.5-3B-Instruct`[8] and `Qwen2.5-7B`[9] using three representative fine-tuning strategies: `Random`, `Safe-Bal`, and `Unsafe-Bal`. Table 13 reports the results from Epoch 0 through Epoch 5. **Both `Qwen` models exhibit the same qualitative patterns observed in the `Llama` result given in the main text.**

Table 13: Summary of safety and PPL dynamics across models and fine-tuning strategies. `R`, `S-B`, and `U-B` denote `Random`, `Safe-Bal`, and `Unsafe-Bal` in abbreviation, respectively. (1) *Early Safety Drop*: decrease in safety from E0 to E1; (2) *Overall Safety*: safety score at E5; (3) *Safety CV*: coefficient of variation of concept-level safety at E5; (4) Δ*Sep*: change in PPL separation ($\text{PPL}_{\text{unsafe}} - \text{PPL}_{\text{safe}}$) from E0 to E5; (5) *Safety Gap*: safety on `safe-cat` vs. `unsafe-cat` samples under `Safe-Bal` at E5.

	Early Safety Drop (E0→E1)			**Overall Safety** (E5)			**Safety CV** (E5)			**ΔSep** (E0→E5)			**Safety Gap (S-B)** (E5)		
Model	R	S-B	U-B	R	S-B	U-B	R	S-B	U-B	R	S-B	U-B	safe	unsafe	gap
Llama-3-8B	0.297	0.195	0.290	0.701	0.812	0.667	0.404	0.271	0.439	−1.37	+4.75	−5.18	0.946	0.738	0.208
Qwen2.5-7B	0.298	0.190	0.266	0.711	0.825	0.662	0.390	0.258	0.437	+0.85	+4.56	−1.40	0.952	0.698	0.254
Qwen2.5-3B	0.273	0.196	0.295	0.700	0.823	0.661	0.405	0.252	0.448	+2.11	+7.19	−1.77	0.953	0.693	0.259

First, (1) the early-stage safety drop generalizes beyond the `Llama` family. Even under `Safe-Bal` with exclusively safe data, the overall safety score drops from 0.9816 to 0.7917 for `Qwen-7B` and from 0.9932 to 0.7977 for `Qwen-3B` at Epoch 1. These decreases (0.1899 and 0.1955) are nearly identical to those observed for `Llama-3-8B` (0.1951). Comparable early-stage degradation is also observed under Random and `Unsafe-Bal`. (2) Second, the effect of training-data safety orientation is consistently preserved across model families. At Epoch 5, all three models exhibit the same behavioral-safety ordering, `Safe-Bal` > `Random` > `Unsafe-Bal`. Specifically, Qwen-7B obtains safety scores of 0.8248, 0.7109, and 0.6624, while `Qwen-3B` obtains 0.8229, 0.7000, and 0.6613 under `Safe-Bal`, `Random`, and `Unsafe-Bal`, respectively. These results closely match the corresponding `Llama-3-8B` scores. (3) Also, substantial concept-level heterogeneity remains across model families. At Epoch 5, `Qwen-7B` exhibits CV values of 0.3895, 0.2575, and 0.4372, while `Qwen-3B` exhibits 0.4054, 0.2524, and 0.4481 under `Random`, `Safe-Bal`, and `Unsafe-Bal`, respectively. Both `Qwen` models therefore reproduce the `Llama` ordering `Safe-Bal` < `Random` < `Unsafe-Bal`, with closely comparable dispersion magnitudes.

Furthermore, (4) the opposing effects of `Safe-Bal` and `Unsafe-Bal` fine-tuning on safe–unsafe PPL separation are also preserved. From Epoch 0 to Epoch 5, `Safe-Bal` produces a positive separation change for all three models, whereas `Unsafe-Bal` produces a negative change. `Qwen-7B` yields changes of +4.56 and -1.40 under `Safe-Bal` and `Unsafe-Bal`, respectively, while `Qwen-3B` yields +7.19 and -1.77. **We note that the `Random` condition differs across families. It is negative for `Llama` but positive for `Qwen`, indicating that the random baseline remains independent for different models.**

Finally, (5) the persistent safety gap between `safe-cat` and `unsafe-cat` also carries over to `Qwen`. Under `Safe-Bal` at Epoch 5, `unsafe-cat` prompts remain suppressed relative to safe-cat on both `Qwen` scales, with a gap (≈0.25) that closely matches the 0.21-0.27 gap reported for `Llama`, supporting that this gap is a structural property rather than a `Llama`-specific artifact.

[8] https://huggingface.co/Qwen/Qwen2.5-3B-Instruct
[9] https://huggingface.co/Qwen/Qwen2.5-7B-Instruct

D.5 Effect of Trainable Parameters

We additionally conducted full-parameter fine-tuning on `Llama-3-8B` under `Random`, `Safe-Bal`, and `Unsafe-Bal` strategies to evaluate the effect under different proportions of trainable parameters. The LoRA setting takes $r = 16$ as given in Table 5. We report the results from Epoch 0 to Epoch 5 below in Table 14.

Table 14: Comparison of LoRA and full-parameter fine-tuning on `Llama-3-8B`. `R`, `S-B`, and `U-B` denote `Random`, `Safe-Bal`, and `Unsafe-Bal`, respectively. (1) *Early Safety Drop*: decrease in safety from E0 to E1; (2) *Overall Safety*: safety score at E5. (3) *Safety CV*: coefficient of variation of concept-level safety at E5 (CV at E0 is 0.062); (4) Δ*Sep*: change in PPL separation ($\text{PPL}_{\text{unsafe}} - \text{PPL}_{\text{safe}}$) from E0 to E5.

	Early Safety Drop (E0→E1)			**Overall Safety** (E5)			**Safety CV** (E5)			**ΔSep** (E0→E5)		
FT	R	S-B	U-B	R	S-B	U-B	R	S-B	U-B	R	S-B	U-B
LoRA	0.297	0.195	0.290	0.701	0.812	0.667	0.404	0.271	0.439	−1.37	+4.75	−5.18
Full	0.282	0.201	0.593	0.601	0.786	0.536	0.456	0.262	0.501	−70.94	+120.67	−186.64

First, (1) the early-stage safety disruption persists under full fine-tuning. Even under `Safe-Bal` with exclusively safe data, the overall safety score drops from 0.9729 at Epoch 0 to 0.7722 at Epoch 1, which is **nearly identical to the LoRA** result (0.9729 to 0.7778). Second, (2) the effect of safety orientation on training data is preserved: at Epoch 5, `Random`, `Safe-Bal`, and `Unsafe-Bal` obtain safety scores of 0.6008, 0.7861, and 0.5357 under full fine-tuning, respectively, **following the same ordering as LoRA** (0.7011, 0.8120, and 0.6669). Third, (3) substantial concept-level heterogeneity remains under full fine-tuning. At Epoch 5, the CV values for `Random`, `Safe-Bal`, and `Unsafe-Bal` are 0.4564, 0.2615, and 0.5010, respectively, preserving the same ordering observed under LoRA (0.4037, 0.2707, and 0.4387), but **at a more prominent scale**. (4) For the last point, the direction of safe/unsafe PPL separation is also preserved, although full fine-tuning produces substantially larger amplification. At Epoch 5, the separation values are -70.938, +120.671, and -186.643 for `Random`, `Safe-Bal`, and `Unsafe-Bal`, compared with -1.374, +4.753, and -5.183 under LoRA.

These results show that **such phenomena are not specific to LoRA but are even stronger in full fine-tuning, confirming that full fine-tuning perturbs a larger parameter subspace and that LoRA is the more conservative setting**.

D.6 Ruling out ceiling effects on inter-concept dispersion

The pretrained models are already safety-aligned at epoch 0, with mean safety scores close to the upper bound of 1. Because safety scores lie in $[0, 1]$, there is less room for scores to increase under `Safe-X` strategies than to decrease under `Unsafe-X` strategies. Proximity to the ceiling can also constrain the attainable variance, thereby compressing the CV. We therefore examine dispersion under strategies with comparable mean safety scores and track recurring HIGH and LOW safety concepts across fine-tuning strategies.

According to the results, **this heterogeneity remains visible under approximately matched means and away from the upper boundary**. As shown in Table 12, `Unsafe-Rnd` and `Unsafe-Bal` for `Llama-3-8B` at Epoch 2 have nearly identical mean safety scores (0.646 vs. 0.657), while Unsafe-Bal exhibits higher CV (0.460 vs. 0.444) and PR (0.406 vs. 0.365). It further shows that `Unsafe-Bal` consistently yields the highest PR across all three model scales at Epoch 2, although its mean differs only marginally from comparable unsafe or random strategies. These observations suggest that proximity to the ceiling alone is unlikely to account for the differences in dispersion across coverage strategies.

Table 1 complements this comparison by identifying concepts that repeatedly occupy the high- or low-safety region. For example, `DISCRIMINATION`, `MEXICANS`, and `BLACK PEOPLE` remain in the high-safety region even under adversarial fine-tuning, while scores for other concepts decline. Conversely, `WEED`, `FUSE`, and `IGNITION` remain in the low-safety region even under safety-oriented fine-tuning. **The latter concepts have room for their scores to increase, yet they remain in the low-safety region across the observed training conditions**.

D.7 Cross-strategy consistency of concept safety rankings

We compute the Spearman's rank correlation coefficients (ρ) among five strategies based on the concepts' safety rankings at epoch 5. For each pair of strategies, it measures how similarly they rank concepts across three categories: `overall`, `safe-cat`, and `unsafe-cat` safety. Statistical significance is indicated by stars (*** for $p < 0.001$).

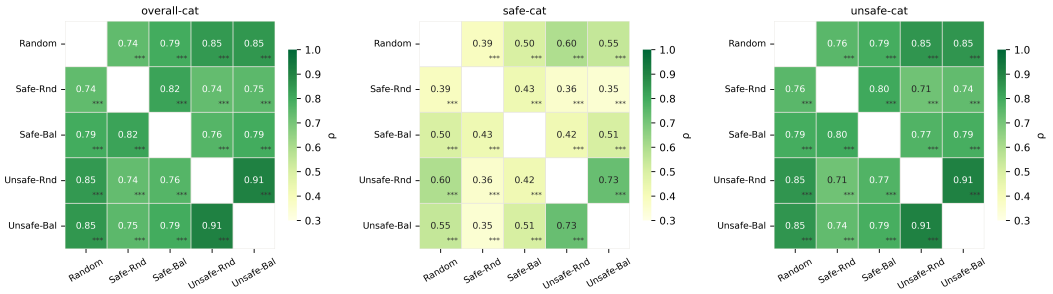

Figure 6: Spearman ρ of concept-level safety rankings across strategies at Epoch 5. High ρ means that fine-tuning shifts the safety level but not the inter-concept structure.

The findings reveal a striking divergence across categories. For `overall` and `unsafe-cat`, strategies show strong agreement, with mean correlations of 0.800 and 0.797 respectively, and all pairwise correlations exceeding 0.70. In contrast, the `safe-cat` exhibits only weak stability, with a mean correlation of just 0.483. Most concerning, even two strategies that both use exclusively safe data, `Safe-Rnd` and `Safe-Bal`, agree only moderately ($\rho = 0.430$) on which concepts are considered safe. This indicates that the sampling methodology fundamentally alters the ranking of safe concepts. Meanwhile, unsafe concept rankings remain highly robust across all strategies ($\rho = 0.907$ between `Unsafe-Rnd` and `Unsafe-Bal`), suggesting that judgments of unsafe content are consistent and reproducible. Overall safety scores, computed as the average over safe and unsafe categories, are largely dominated by the comparatively stable unsafe component, thereby obscuring substantial instability in `safe-cat` rankings. These findings raise important concerns about the reliability of current safety evaluation practices: **model behavior on unsafe content remains largely consistent across strategies, whereas safety assessments on safe content vary considerably**, stressing the need for standardized methodologies in concept-level safety evaluation.

D.8 Safety gap between `safe-cat` and `unsafe-cat`

Our test set contains an equal number of safe and unsafe samples per concept, as designated by the `is_safe` field in the BeaverTails dataset. Importantly, the safe/unsafe label of each sample is determined by the *answer* rather than the *prompt*: the same prompt can appear in both safe and unsafe samples depending on the associated response. Consequently, when we evaluate model safety by feeding only the prompt and scoring the generated output, the prompts themselves carry no explicit safety signal by design, and that any behavioral difference between `safe-cat` and `unsafe-cat` prompts must therefore reflect safety knowledge learned by the model rather than differences in the input. We plot such gap in Figure 7.

Against this expectation, we observe **a prominent and persistent gap between `safe-cat` and `unsafe-cat` safety scores across all strategies**. As shown in Figure 7, the overall safety score masks a structural divergence that aggregate metrics alone cannot reveal. Under `Safe-Bal`, for instance, safe-cat prompts elicit safety scores around 0.94 throughout training, while `unsafe-cat` prompts remain suppressed at approximately 0.67 — a gap of more than 0.27 that neither safe nor unsafe fine-tuning closes. The CV and PR rows corroborate this finding: `unsafe-cat` prompts consistently exhibit higher inter-concept dispersion than safe-cat prompts across all strategies and epochs, indicating that the suppressed safety of `unsafe-cat` responses is not uniform but reflects selective vulnerability at the concept level. Crucially, this gap persists across different training data compositions, including safe-only, unsafe-only, and mixed settings. This indicates that **the model's differential response to prompt style constitutes a structural property established during the pre-training stage, one that fine-tuning alone is insufficient to overcome**.

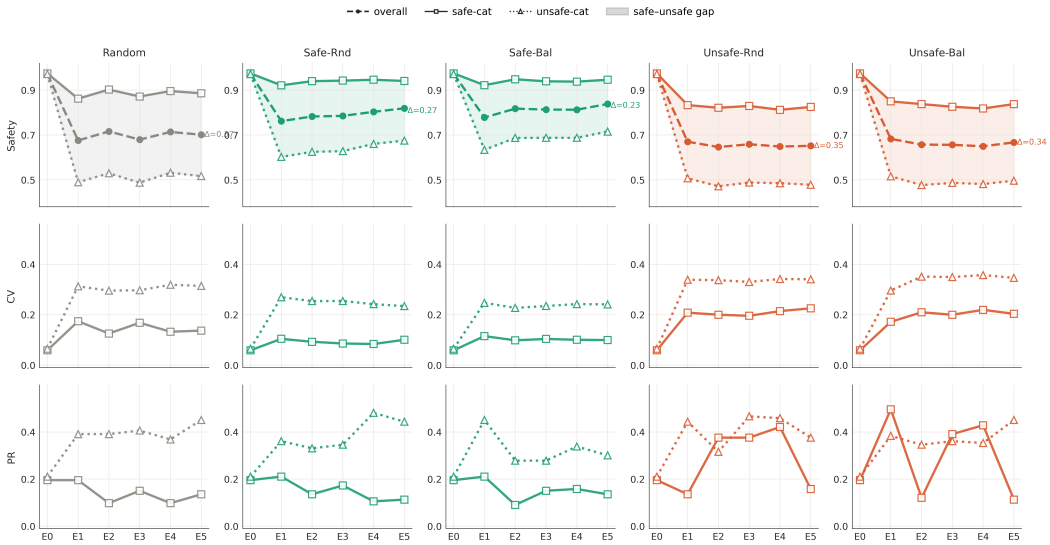

Figure 7: Performance gap between `safe-cat` and `unsafe-cat` samples, where such gap exists prominently across all strategies.

E Discussion and Limitation

Discussion. A notable challenge in evaluating safety fine-tuning is the ceiling effect induced by the base model itself. Our foundation model, `Llama-3-8B-Instruct`, already achieves approximately 97% safety on the test set — a result of the extensive safety alignment typically applied to open-source models prior to public release to mitigate risks to users and downstream applications. This near-ceiling performance implies that fine-tuning, whether with safe or unsafe data, has limited room for further absolute improvement. This phenomenon has also been noted in previous studies [Qi et al., 2023, Kim et al., 2025], but is reflected as risks of fine-tuning rather than structural corruption. Consequently, rather than focusing on absolute safety scores, we adopt a balanced baseline comprising 50% safe and 50% unsafe samples, allowing us to investigate two critical questions: (1) to what extent can safe data improve per-concept safety, and (2) how resistant are individual concepts to unsafe fine-tuning. By comparing against this balanced reference point, we isolate the relative effects of safety-oriented versus adversarial fine-tuning, circumventing the ceiling effect that would otherwise obscure meaningful differences.

Limitation. Several limitations of the current study warrant acknowledgment. First, perplexity is used as a proxy for familiarity with knowledge rather than as a direct measure of the internal knowledge state. While this is a well-established probing approach in previous literature and is validated here by its alignment with behavioral safety scores, it captures knowledge accessibility through next-token prediction and may not fully reflect more complex forms of knowledge organization, such as multi-hop reasoning chains or context-dependent retrieval. Future work combining PPL-based analysis with mechanistic interpretability techniques could provide a more complete account of the internal knowledge dynamics we observe.

Second, the concept taxonomy used in this study is derived from `BeaverTails`' knowledge structure and covers 133 safety-relevant semantic concepts. We acknowledge that alternative taxonomies [Medelyan et al., 2013, Yi and Choi, 2012] with different granularities might surface additional patterns, and exploring the sensitivity of our findings to concept granularity remains an interesting direction for future work. Also, the concepts used in this study represent a curated subset selected from a much larger concept space over 40,000 nodes (as reported in Section 3.2), chosen for their safety relevance and representativeness. While this provides a systematic and tractable basis for analysis, it inevitably covers only a fraction of the model's full knowledge landscape. Safety dynamics in underrepresented or excluded concept regions may differ from the patterns we report, and the generalizability of our findings to the broader concept space remains an open question.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: Our abstract and introduction accurately reflect the contributions and scope of the paper. In Sections 3 and 4, we present detailed experimental designs and comprehensive results for the contributions outlined in the abstract and introduction.

   Guidelines:

   - The answer [N/A] means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A [No] or [N/A] answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: We discuss the limitations of the paper in Appendix E.

   Guidelines:

   - The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

Answer: [N/A]

Justification: This paper makes no theoretical claims and presents no formal proofs. It is an empirical mechanistic study. All findings are grounded in controlled experiments and quantitative analysis.

Guidelines:

- The answer [N/A] means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

Answer: [Yes]

Justification: We provide detailed information on datasets and training settings in Section 3 for all experiments in the paper to support the reproducibility of the results. In Appendix C, we report comprehensive experimental details, including parameters, prompts, and other relevant information.

Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
- If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
- Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
  (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
  (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
  (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
  (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in

some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [Yes]

   Justification: Yes. The code repository is provided supplemental material and contains sufficient instructions to reproduce the main experimental results.

   Guidelines:

   - The answer [N/A] means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - While we encourage the release of code and data, we understand that this might not be possible, so [No] is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
   - The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
   - The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
   - At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
   - Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

   Answer: [Yes]

   Justification: Yes. All training and evaluation details necessary to understand the results are specified in Section 3 and Appendix C.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [N/A]

   Justification: The main experiments are conducted with a fixed random seed to ensure reproducibility. The consistency of our key findings across three model scales (1B, 3B, and 8B) and five distinct fine-tuning strategies provides empirical robustness.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.

- The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
- If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: We report the computational resources used for the experiments in Appendix C.2. All experiments were conducted on a server equipped with 4 NVIDIA A100 PCIe GPUs (80 GB VRAM each), an Intel Xeon Gold 5318Y CPU (2.10 GHz), and 252 GB of system RAM.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: All research in this paper complies with the NeurIPS Code of Ethics in every aspect.

   Guidelines:

   - The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

Answer: [Yes]

Justification: The societal impacts of this work are discussed in the paper. Our findings contribute to the identification and interpretation of the structurally unsafe concept, which benefits the design of more reliable and transparent safety alignment pipelines.

Guidelines:

- The answer [N/A] means that there is no societal impact of the work performed.
- If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.
- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

Answer: [N/A]

Justification: This paper does not release any fine-tuned model weights or new datasets. The base models used (Llama-3 Instruct variants) and the evaluation dataset (BeaverTails) are publicly available resources governed by their respective licenses. The anonymized code repository contains only the experimental pipeline and analysis scripts, which do not pose a misuse risk in themselves.

Guidelines:

- The answer [N/A] means that the paper poses no such risks.
- Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
- Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
- We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

Answer: [Yes]

Justification: All assets used in this paper are properly credited in the reference and footnote.

Guidelines:

- The answer [N/A] means that the paper does not use existing assets.
- The authors should cite the original paper that produced the code package or dataset.
- The authors should state which version of the asset is used and, if possible, include a URL.
- The name of the license (e.g., CC-BY 4.0) should be included for each asset.
- For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

Answer: [Yes]

Justification: We provide the open-source code in the supplemental material.

Guidelines:

- The answer [N/A] means that the paper does not release new assets.
- Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
- The paper should discuss whether and how consent was obtained from people whose asset is used.
- At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

Answer: [N/A]

Justification: The paper does not involve crowdsourcing or research with human subjects. The safety labels for 100 model responses used to validate our evaluator (Appendix D.1) were annotated by three of the authors, and no external participants were recruited.

Guidelines:

- The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
- Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
- According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

Answer: [N/A]

Justification: The paper did not involve crowdsourcing nor research with human subjects.

Guidelines:

- The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

Answer: [Yes]

Justification: LLMs are not used as a component of the core methodology. GPT-4 filters candidate concepts, and GPT-5.6 serves as an additional judge to validate the safety evaluation and to summarize the refusal patterns. They were also used for minor grammatical corrections during the writing process.

Guidelines:

- The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
- Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.