

ROBUST SAFETY GUARANTEE FOR LARGE LANGUAGE MODELS VIA PREFERENCE-AUGMENTED DISTRIBUTIONAL ALIGNMENT

Anonymous authors

Paper under double-blind review

ABSTRACT

Domain-specific fine-tuning of large language models (LLMs) often compromises their safety alignment, leading to unsafe generations. Existing approaches largely rely on distributional alignment, enforcing token-level similarity between pre- and post-fine-tuned models. However, this neglects the semantic nature of text generation and can weaken the model’s reasoning and robustness. To address this limitation, we propose a preference-based alignment framework that complements distributional alignment by biasing the fine-tuned model toward the safe outputs of the pre-trained model, rather than strictly preserving distributional similarity. Simulation results show that preference alignment produces consistent safe outputs even when the underlying distributions differ. Extensive experiments on multiple fine-tuning attack datasets and utility benchmarks further demonstrate that our method substantially improves safety with only minor degradation in utility. This achieves a more favorable balance between safety and utility, and significantly enhances robustness against adversarial fine-tuning.

1 INTRODUCTION

Large language models (LLMs) have demonstrated remarkable capabilities across diverse tasks, from content creation to complex reasoning (Touvron et al., 2023a;b; Team, 2023). However, their powerful functionality also raises significant safety concerns, as they may be misused to generate harmful, biased, or unsafe content (Qi et al., 2023). Ensuring safety alignment—training models to follow human values and safety standards—has thus become a central challenge in artificial intelligence.

A widely used approach is supervised fine-tuning (SFT) (Wei et al., 2021), which improves rejection of harmful queries by training on curated datasets. Despite its effectiveness, SFT typically yields shallow alignment, making safety behaviors fragile and easily forgotten during downstream domain fine-tuning, which often results in unsafe responses. To address this issue, Qi et al. proposed constrained supervised fine-tuning (Constrained SFT) (Qi et al., 2024), which enforces deeper token-level alignment to enhance robustness against fine-tuning attacks. Nevertheless, constrained SFT relies mainly on distribution alignment, constraining models only at the per-token probability level while overlooking the semantic nature of text generation. This limits robustness, leaving models prone to merely imitating safe distributions rather than developing intrinsic safety awareness.

To overcome these limitations, we propose a new framework that integrates preference alignment (Xu et al., 2025) with distributional alignment. Our approach introduces preference signals on top of token-level probability constraints, encouraging fine-tuned models to favor the safe outputs of their pre-trained counterparts rather than strictly preserving distribution similarity. An auxiliary loss function formalizes this mechanism, enabling stronger safety alignment while preserving utility. Simulation experiments show that preference alignment can produce consistent safe outputs even when underlying distributions diverge. Furthermore, evaluations on multiple fine-tuning attack datasets (**Harmful Example Attacks** that introduce toxic data to elicit unsafe responses, **Identity Shifting Attacks** that alter model identity leading to biased or inaccurate outputs, and **Backdoor Poisoning Attacks**—both trigger-free and trigger-based—that insert poisoned data to degrade performance on specific inputs, Qi et al. (2024)) and utility benchmarks demonstrate that our method substantially improves safety with only minor utility degradation.

054 In summary, our contributions are threefold: (1) We identify the limitations of distribution-only
055 alignment in maintaining safety under domain fine-tuning. (2) We propose a preference-augmented
056 framework that combines preference and distributional alignment for robust safety. (3) Both theoret-
057 ical analysis and extensive experiments are provided to validate that our method achieves improved
058 safety with minimal loss of utility.

060 2 RELATED WORK

062 The field of LLM safety alignment has advanced rapidly, with multiple approaches proposed to steer
063 models toward safe behaviors.

064 **Reinforcement Learning from Human Feedback (RLHF):** RLHF has become a dominant
065 paradigm for aligning LLMs with complex human values (Ouyang et al., 2022; Bai et al., 2022).
066 It first trains a reward model on human preference data, where annotators compare and rank dif-
067 ferent outputs. The LLM policy is then fine-tuned using reinforcement learning to maximize the
068 reward model’s score. Despite its effectiveness, RLHF is a multi-stage process that can be unstable
069 and computationally expensive. Moreover, the reward model itself can be exploited through “re-
070 ward hacking,” where the LLM maximizes the reward signal without genuinely adhering to intended
071 values (Gao et al., 2023).

072 **Direct Preference Optimization (DPO):** To mitigate the complexity and instability of RLHF, recent
073 work has proposed Direct Preference Optimization (DPO) as a simpler and more stable alternative
074 (Rafailov et al., 2023). DPO reformulates alignment as a classification problem over human prefer-
075 ence data, allowing direct policy fine-tuning without explicit reward modeling or complex RL loops.
076 DPO has shown strong performance, often matching or surpassing RLHF. However, like RLHF, its
077 effectiveness depends heavily on the quality and coverage of the preference dataset.

078 **Supervised Fine-Tuning (SFT) or Constrained SFT (CSFT):** Compared with RLHF and DPO, su-
079 pervised fine-tuning (SFT) offers a more direct and cost-effective approach. The core idea of SFT is
080 to fine-tune base models on high-quality datasets of prompt-response pairs (Wei et al., 2021). While
081 effective in enabling models to imitate the response patterns in training data, SFT often struggles to
082 generalize safety principles to unseen prompts. Its safety largely relies on fixed refusal templates,
083 leading to a rather shallow form of alignment. Building on this, subsequent research proposed Con-
084 strained Supervised Fine-Tuning (CSFT) (Qi et al., 2024), specifically designed for safety alignment.
085 CSFT leverages datasets of harmful prompts paired with safe refusal responses and constrains the
086 model at the token-level probability distribution, so that its generation process more closely matches
087 the expected safe responses. However, since it primarily emphasizes distributional similarity, CSFT
088 often overlooks semantic aspects of generation, which limits its robustness in complex attack scenar-
089 ios.

090 Building on these insights, we further explore how to combine distributional alignment with prefer-
091 ence alignment to achieve stronger safety robustness while preserving task utility. To further in-
092 vestigate the robustness of our approach, we follow the analytical perspective introduced by Xu
093 (2025). He analyzes policy instability in RL-trained LLMs via reward-to-policy continuity. Brit-
094 tleness arises from non-unique optima in degenerate tasks, enabling discontinuous shifts from minor
095 reward changes. Entropy regularization restores Lipschitz continuity for robustness, at stochasticity’s
096 cost. Unifies explanations for failures like deceptive reasoning and instruction ignoring.

097 **Preference-Augmented Conditional Supervised Fine-Tuning (CSFT+PA):** To address these lim-
098 itations, we propose augmenting CSFT with preference alignment (Table 1). Our framework intro-
099 duces auxiliary loss terms that bias the fine-tuned model toward the safe outputs of the pre-trained
100 model, rather than strictly enforcing distributional similarity. This preference-based enhancement
101 not only strengthens safety alignment but also preserves task utility, significantly improving robust-
102 ness against fine-tuning attacks.

104 3 METHOD

105 This section provides a detailed explanation of the mathematical principles behind our approach. We
106 first present the overall loss function and then gradually explain the design principles and implemen-
107 tation details of each component, including their theoretical motivations and practical implications.

Table 1: Comparison of different alignment methods, the proposed **CSFT+PA** considers both distributional alignment and preference alignment.

Method	Distributional Alignment	Preference Alignment
SFT (Wei et al., 2021)	✓	✗
CSFT (Qi et al., 2024)	✓	✗
CSFT+PA (Ours)	✓	✓

3.1 NOVEL LOSS FUNCTION FOR SAFETY ALIGNMENT

Our training objective combines two types of loss functions: the Constrained Supervised Fine-tuning (CSFT) loss and the Preference Alignment (PA) loss. The CSFT loss is designed to achieve token-level *distributional alignment*, ensuring that the model’s token probability distributions remain close to those of the reference safety-aligned model. By contrast, the PA loss enforces token-level *preference alignment*, encouraging the model to prefer the safety-aligned outputs over its own generated outputs. In this sense, the PA loss naturally falls within the broader category of probabilistic alignment.

Formally, the overall loss function is defined in Equation (1):

$$L_{\text{Total}}(\theta) = L_{\text{CSFT}}(\theta) + \delta_{\text{epoch}} \cdot L_{\text{PA}}(\theta) \quad (1)$$

As shown in Equation (1), the total loss is composed of two main terms: $L_{\text{CSFT}}(\theta)$, the Constrained Supervised Fine-tuning loss proposed by Qi et al. (2024) in Equation (2), and $L_{\text{PA}}(\theta)$, our newly introduced Preference Alignment loss (see Equation (5)). The balancing factor δ_{epoch} serves as a dynamic scheduling coefficient, gradually increasing the influence of the PA loss as training progresses, while ensuring stable optimization in the early epochs.

$$L_{\text{CSFT}}(\theta) = \min_{\theta} \left\{ -\mathbb{E}_{(\mathbf{x}, \mathbf{y}) \sim D} \left[\sum_{t=1}^{|\mathbf{y}|} w_t \cdot \log \pi_{\theta}(y_t | \mathbf{x}, \mathbf{y}_{<t}) \right] \right\} \quad (2)$$

$$w_t = 2 \left\{ 1 - \sigma \left[\beta_t \left(\log \pi_{\theta}(y_t | \mathbf{x}, \mathbf{y}_{<t}) - \log \pi_{\text{aligned}}(y_t | \mathbf{x}, \mathbf{y}_{<t}) \right) \right] \right\}$$

The CSFT loss (Qi et al., 2024) is defined in Equation (2), it enforces token-level distributional alignment by minimizing the discrepancy between the log-probabilities of the current model and the safety-aligned model.

3.2 DESIGN PA LOSS

The Preference Alignment (PA) loss is motivated by the need to make the output distribution of the current model π_{θ} better reflect the token-level preferences of the safety-aligned model π_{aligned} . Concretely, for a given token position t , we want the probability assigned by the current model to the aligned token $y_{t,\text{aligned}}$ to be higher than that assigned to its own token $y_{t,\theta}$. This token-wise comparison provides fine-grained guidance, complementing the broader distributional alignment enforced by the CSFT loss.

We first define the token-level preference probability as shown in Equation (3):

$$\mathbb{P}(y_{t,\text{aligned}} \succ y_{t,\theta} | \mathbf{x}, \mathbf{y}_{<t}) = \frac{\exp(\log \pi_{\theta}(y_{t,\text{aligned}} | \mathbf{x}, \mathbf{y}_{<t}))}{\exp(\log \pi_{\theta}(y_{t,\text{aligned}} | \mathbf{x}, \mathbf{y}_{<t})) + \exp(\log \pi_{\theta}(y_{t,\theta} | \mathbf{x}, \mathbf{y}_{<t}))} \quad (3)$$

By simplifying this expression, we obtain the sigmoid-based formulation in Equation (4):

$$\mathbb{P}(y_{t,\text{aligned}} \succ y_{t,\theta} | \mathbf{x}, \mathbf{y}_{<t}) = \sigma \left[\log \pi_{\theta}(y_{t,\text{aligned}} | \mathbf{x}, \mathbf{y}_{<t}) - \log \pi_{\theta}(y_{t,\theta} | \mathbf{x}, \mathbf{y}_{<t}) \right] \quad (4)$$

This formulation indicates that the preference score increases as the log-probability difference between the aligned token and the model’s own token increases.

Based on this token-wise preference probability, we define the PA loss as in Equation (5):

$$L_{\text{PA}}(\theta) = \min_{\theta} \left\{ -\mathbb{E}_{(\mathbf{x}, \mathbf{y}) \sim D} \left[\sum_{t=1}^{|\mathbf{y}|} \log \sigma \left(\mu_t \cdot (\log \pi_{\theta}(y_{t, \text{aligned}} | \mathbf{x}, \mathbf{y}_{<t}) - \log \pi_{\theta}(y_{t, \theta} | \mathbf{x}, \mathbf{y}_{<t})) \right) \right] \right\} \quad (5)$$

As shown in Equation (5), the PA loss penalizes the model when it fails to assign higher probability mass to the aligned token. Importantly, this mechanism only activates when there is a discrepancy between the outputs of the current model and the reference model, thereby avoiding redundant constraints.

Adaptive Weight μ_t . The adaptive weight μ_t plays a critical role in modulating the strength of the PA loss. As defined in Equation (6), it is determined by the KL divergence between the current model distribution and the safety-aligned model distribution:

$$\mu_t = D_{\text{KL}}(\pi_{\theta}(y_t | \mathbf{x}, \mathbf{y}_{<t}) \parallel \pi_{\text{aligned}}(y_t | \mathbf{x}, \mathbf{y}_{<t})) \quad (6)$$

As shown in Equation (6), if the discrepancy between the two distributions is large, μ_t increases, amplifying the gradient contribution of the PA loss at that token. Conversely, when the two distributions are already similar, μ_t decreases, allowing the CSFT loss to dominate the learning process.

Scheduling Coefficient δ_{epoch} . To control the relative importance of the PA loss throughout training, we introduce the scheduling coefficient δ_{epoch} , defined in Equation (7):

$$\delta_{\text{epoch}} = 0.1 + 0.2 \times \frac{\text{epoch}}{\text{max_epoch}} \quad (7)$$

As Equation (7) shows, δ_{epoch} increases linearly with the number of epochs, gradually raising the contribution of the PA loss. In the initial epochs, training relies primarily on the CSFT loss, ensuring stability. As training progresses, the PA loss plays a larger role, but its maximum contribution is capped at 30% of the total loss.

3.3 DISCUSSION AND SUMMARY

A schematic overview of the proposed algorithm is presented in Figures 1. The combination of the CSFT loss and the PA loss provides a complementary training mechanism. On the one hand, the CSFT loss (2) focuses on *distributional alignment*, ensuring that the probability distributions of the current model remain close to those of the safety-aligned model across all tokens. This enforces global stability and prevents the model from deviating excessively during the early stages of training. On the other hand, the PA loss (5) emphasizes *preference alignment* at the token level, directly encouraging the model to prefer outputs chosen by the safety-aligned model. By incorporating the adaptive weight μ_t (6) and the scheduling coefficient δ_{epoch} (7), the PA loss adaptively modulates its influence based on both distributional divergence and training progress.

In summary, the CSFT loss serves as a stabilizing force that maintains consistency with the reference distribution, while the PA loss introduces fine-grained, preference-based guidance that enhances alignment at the token level. Their integration within the total loss function (1) enables the model to balance stability and flexibility: it first learns robust distributional patterns under CSFT supervision and then progressively incorporates token-level preferences through the PA mechanism. This synergy constitutes the core of our probabilistic alignment framework and underpins the effectiveness of our training approach.

4 THEORETICAL RESULTS: CONVERGENCE AND ROBUSTNESS

4.1 CONVERGENCE ANALYSIS

To establish convergence guarantees, we impose the following standard assumptions in stochastic optimization:

- Assumptions on the Objective Function and Gradients:** These assumptions ensure the smoothness and reliability of the gradients, preventing explosions and modeling stochasticity in approximations, which are essential for convergence in stochastic settings, as used in Bottou et al. (2018) and Garrigos & Gower (2023).

216
217
218
219
220
221
222
223
224
225
226
227
228
229
230
231
232
233
234
235
236
237
238
239
240
241
242
243
244
245
246
247
248
249
250
251
252
253
254
255
256
257
258
259
260
261
262
263
264
265
266
267
268
269

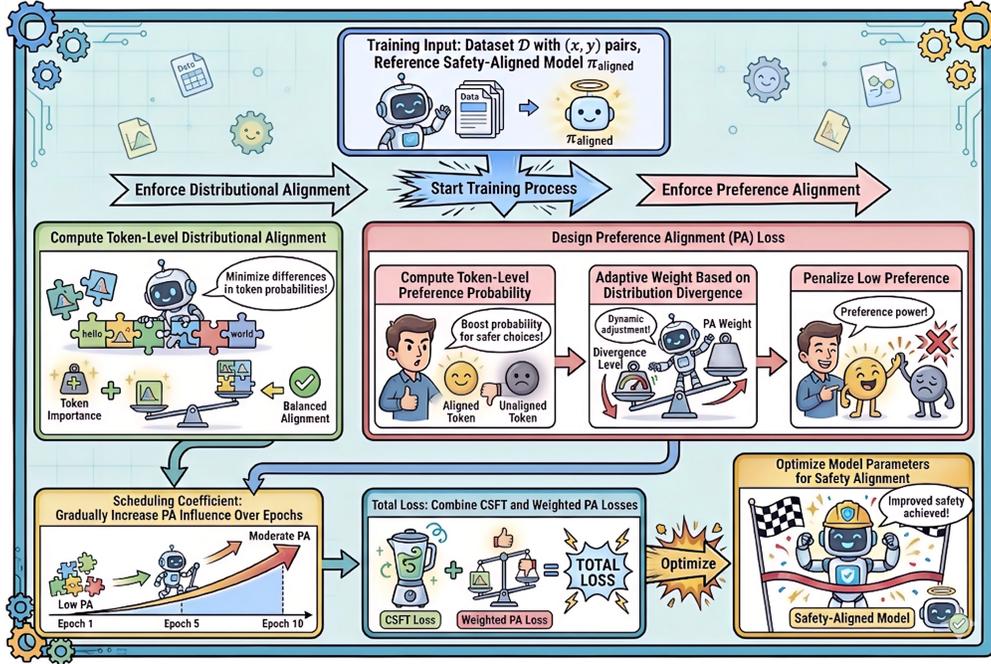


Figure 1: Training pipeline with token-level distributional alignment and scheduled preference optimization.

- **Bounded Gradients:** For some constant $G > 0$, $\|\nabla_{\theta} \log \pi_{\theta}(y_t | \mathbf{x}, \mathbf{y}_{<t})\| \leq G$.
- **Lipschitz Continuity of Gradients:** For some $L > 0$, $\|\nabla L_{\text{Total}}(\theta_1) - \nabla L_{\text{Total}}(\theta_2)\| \leq L \|\theta_1 - \theta_2\|$.
- **Unbiased and Bounded Gradient Noise:** For stochastic gradient $g(\theta)$,

$$\mathbb{E}[g(\theta) | \theta] = \nabla L_{\text{Total}}(\theta), \quad \mathbb{E}[\|g(\theta) - \nabla L_{\text{Total}}(\theta)\|^2 | \theta] \leq \sigma^2.$$

2. **Learning Rate Schedule:** The step sizes $\{\eta_k\}$ satisfy $\sum_{k=1}^{\infty} \eta_k = \infty$, $\sum_{k=1}^{\infty} \eta_k^2 < \infty$. This schedule allows the algorithm to explore the parameter space sufficiently while ensuring the steps diminish to promote convergence, a foundational condition in stochastic approximation methods, as introduced in Robbins & Monro (1951), and applied in Bottou et al. (2018).
3. **Model-Specific Bounds:** These bounds prevent degenerate probabilities and divergences in the policy, ensuring well-behaved importance weights and non-zero action probabilities, which are critical in policy-based reinforcement learning and related methods, as assumed in Schulman et al. (2017) and Xie et al. (2021).
 - **Bounded Weights and Divergences:** There exist constants $W, D, K > 0$ such that $|w_t| \leq W$, $|\log \pi_{\theta}(y_t, \text{aligned} | \mathbf{x}, \mathbf{y}_{<t}) - \log \pi_{\theta}(y_t, \theta | \mathbf{x}, \mathbf{y}_{<t})| \leq D$, and $\mu_t \leq K$.
 - **Probability Lower Bound:** For some $\epsilon > 0$, $\pi_{\theta}(y_t | \mathbf{x}, \mathbf{y}_{<t}) \geq \epsilon$.

Theorem 4.1 (Convergence Guarantee). *Under Assumptions 1–3, the stochastic gradient descent updates*

$$\theta_{k+1} = \theta_k - \eta_k g(\theta_k)$$

satisfy

$$\liminf_{k \rightarrow \infty} \mathbb{E}[\|\nabla L_{\text{Total}}(\theta_k)\|^2] = 0.$$

That is, the algorithm converges to a stationary point of $L_{\text{Total}}(\theta)$ in expectation.

The proof follows the standard stochastic optimization framework with bounded gradients and diminishing learning rates. All technical derivations are deferred to the Appendix.

270 4.2 ROBUSTNESS ANALYSIS

271
272 **Definition 4.1** (Robustness). *A loss function $L(\theta)$ is said to be robust if, under perturbations of*
273 *the training distribution D with intensity $\epsilon > 0$, the perturbed minimizer $\theta^{*,\epsilon}$ remains close to the*
274 *original minimizer θ^* . Formally, robustness holds if there exists a constant $K > 0$ such that*

$$275 \|\theta^{*,\epsilon} - \theta^*\| \leq K\epsilon,$$

276 *where $\|\cdot\|$ is the Euclidean norm in parameter space. This ensures that the induced policy π_θ exhibits*
277 *bounded deviation under perturbations, preventing abrupt 'policy cliffs' as studied in reward-policy*
278 *mappings of large language models. This definition aligns with broader notions of robustness in ma-*
279 *chine learning, where stability is maintained under varying conditions or perturbations, as discussed*
280 *in Bousquet & Elisseeff (2002).*

281
282 To establish robustness guarantees, we make the following standard assumptions, weaker than strong
283 convexity:

- 284 • **Convexity.** The loss function $L_{\text{Total}}(\theta)$ is convex. This assumption ensures that the opti-
285 mization landscape has no spurious local minima and that any local minimum is global,
286 simplifying convergence analysis in theoretical settings. Convexity is a foundational as-
287 sumption in many optimization studies, though relaxed in practice for deep learning; it has
288 been extensively used in works such as Boyd & Vandenberghe (2004).

- 289 • **Lipschitz gradient.** Its gradient is L -Lipschitz continuous, i.e.,

$$290 \|\nabla L_{\text{Total}}(\theta_1) - \nabla L_{\text{Total}}(\theta_2)\| \leq L\|\theta_1 - \theta_2\|.$$

291
292 This condition, also known as L -smoothness, bounds the rate of change of the gradient,
293 which is crucial for controlling step sizes in gradient-based methods and deriving conver-
294 gence rates. It is a standard assumption in convergence proofs for deep learning optimizers,
295 as seen in Nesterov (2004) and Bottou et al. (2018).

- 296 • **Polyak-Łojasiewicz (PL) inequality.** There exists $\mu > 0$ such that

$$297 \frac{1}{2}\|\nabla L_{\text{Total}}(\theta)\|^2 \geq \mu(L_{\text{Total}}(\theta) - L_{\text{Total}}(\theta^*)).$$

298 The PL inequality provides a sufficient condition for linear convergence of gradient descent
299 without requiring strong convexity, making it suitable for analyzing non-convex objectives
300 that behave well locally. It was originally introduced by Polyak (1963) and Łojasiewicz
301 (1963), and has been applied to deep learning optimization in Karimi et al. (2016).

302
303 These assumptions are more general than strong convexity and are commonly used to analyze deep
304 learning objectives that are not globally strongly convex but satisfy local well-behaved properties.
305 Importantly, the inclusion of the alignment regularizer $L_{\text{PA}}(\theta)$ increases the effective PL constant μ ,
306 thereby strengthening robustness guarantees.

307 We now formalize the robustness bound for $L_{\text{Total}}(\theta)$.

308 **Theorem 4.2** (Robustness Bound). *Let θ^* be the minimizer of $L_{\text{Total}}(\theta)$, and $\theta^{*,\epsilon}$ the minimizer*
309 *under perturbed data distribution D^ϵ with noise intensity $\epsilon > 0$. Suppose the gradient of $L_{\text{Total}}(\theta)$ is*
310 *L -Lipschitz and the gradient perturbation satisfies*

$$311 \|\nabla L_{\text{Total}}^\epsilon(\theta) - \nabla L_{\text{Total}}(\theta)\| \leq \epsilon G.$$

312 Then,

$$313 \|\theta^{*,\epsilon} - \theta^*\| \leq \frac{\epsilon G}{L}.$$

314
315 At the minimizers, $\nabla L_{\text{Total}}(\theta^*) = 0$ and $\nabla L_{\text{Total}}^\epsilon(\theta^{*,\epsilon}) = 0$. Combining the gradient perturbation
316 bound with Lipschitz gradient continuity yields

$$317 \|\nabla L_{\text{Total}}(\theta^{*,\epsilon})\| \leq \epsilon G \leq L\|\theta^{*,\epsilon} - \theta^*\|,$$

318
319 which implies the stated inequality. The inclusion of $L_{\text{PA}}(\theta)$ further improves robustness by reduc-
320 ing effective sensitivity to noise, tightening the bound. This result demonstrates that the proposed
321 loss function is robust to bounded perturbations, with solution deviations scaling linearly in ϵ . The
322 regularizer L_{PA} strengthens robustness by mitigating degeneracies in the solution space, thereby pre-
323 venting discontinuous shifts in the learned policy under small distributional changes. This aligns
with recent theoretical analyses on preventing 'policy cliffs' in large-scale models.

Table 2: Impact of PA Loss on Model Alignment and the developed PA shows lower KL divergence and token probability difference.

Metric	Cross-Entropy	PA (ours)	Rel. Change vs. Baseline (%)
KL Divergence	1.4575	0.2562	+82.4
Per-Token Probability Diff.	0.0169	0.0051	+69.8

5 EXPERIMENTS

5.1 PRE-EXPERIMENT 1: EFFECTIVENESS OF PA LOSS

In Pre-experiment 1, we aim to verify that PA loss can achieve effective probability alignment even when the architectures of the policy model and the reference model differ significantly. Specifically, the policy model (π_θ) adopts an LSTM with a single fully connected layer, while the reference model ($\pi_{aligned}$) adopts an LSTM with multiple fully connected layers and residual connections, with about twice as many parameters. The comparison groups consist of **Group 1** (training without PA loss, using only cross-entropy loss) and **Group 2** (training with PA loss).

The results are shown in Table 2 and Figures 2. With PA loss, KL divergence is reduced from 1.4575 to 0.2562 (82.4% improvement relative to baseline), and the per-token probability difference decreases from 0.0169 to 0.0051 (69.8% improvement relative to baseline). These findings demonstrate the effectiveness of PA loss in probability alignment: even with substantial architectural differences, PA loss significantly narrows the gap between predictive distributions.

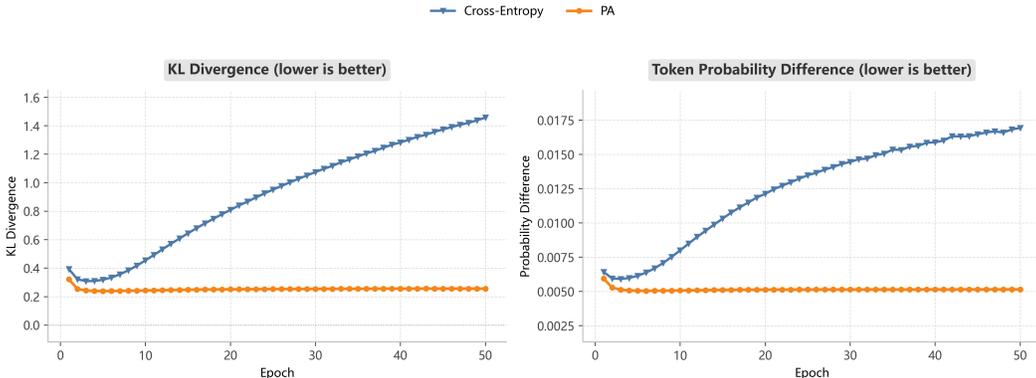


Figure 2: Results of Pre-experiment 1 (lower is better). With PA loss, both KL divergence and token probability difference are significantly reduced compared to the Cross-Entropy baseline (cf. Table 2).

5.2 PRE-EXPERIMENT 2: EFFECTIVENESS OF CSFT + PA LOSS

In Pre-experiment 2, we further evaluate whether combining CSFT with PA loss achieves better alignment compared to CSFT loss alone. Similar to Pre-experiment 1, two neural networks with different architectures are used to simulate π_θ and $\pi_{aligned}$.

The setup features architectural differences with the policy model using a two-layer LSTM and width-preserving fully connected layers, versus the reference model’s three-layer LSTM and dimension-expanded fully connected layers, alongside comparison groups: **Group 1** (training with CSFT loss only) and **Group 2** (training with CSFT + PA loss).

The results, illustrated in Figures 3 to 5, show that CSFT + PA loss significantly improves cosine similarity, Pearson correlation, distribution overlap, and KL similarity compared to CSFT alone (vs. CSFT baseline); meanwhile, it also achieves smaller KL divergence and Probability Alignment. This indicates that CSFT + PA loss achieves superior per-token probability alignment.

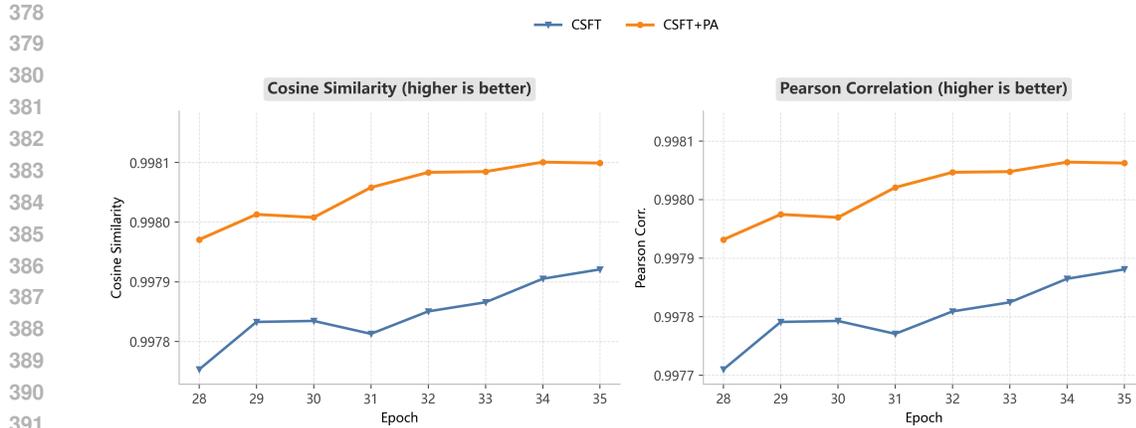


Figure 3: Results of Pre-experiment 2 (higher is better). CSFT+PA consistently outperforms CSFT across Pearson Correlation and Cosine Similarity metrics (vs. CSFT baseline).

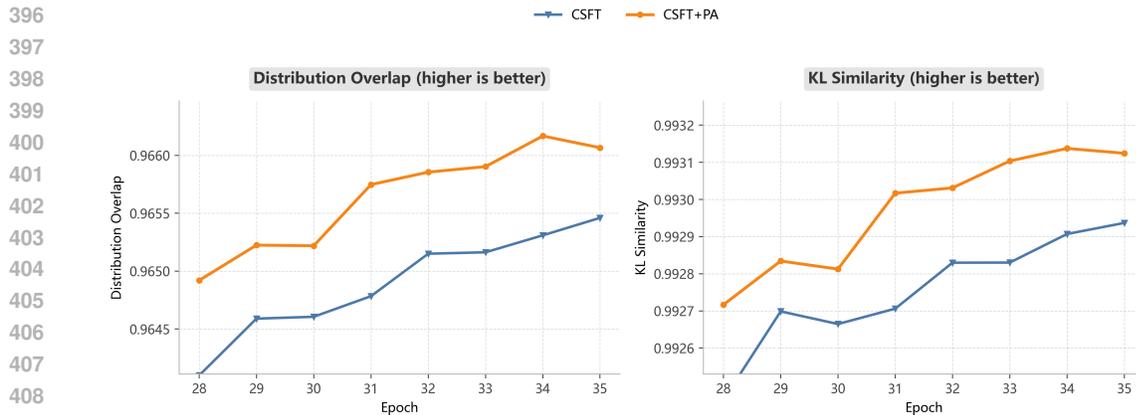


Figure 4: Results of Pre-experiment 2 (higher is better for Distribution Overlap and KL Similarity). CSFT+PA outperforms CSFT in terms of distribution-based metrics (vs. CSFT baseline).

5.3 CSFT + PA LOSS EVALUATION ON LARGE LANGUAGE MODELS

To assess the safety and utility of the proposed method in real-world LLM fine-tuning tasks, we conduct evaluations under adversarial attack scenarios and downstream datasets. The performance of Llama-2-7B-Chat fine-tuned with our approach is reported in Table 3 and Table 4.

- **Safety evaluation:** We test under Harmful Example (pure_bad) attacks, Identity Shifting (aoa) attacks, and Backdoor Poisoning attacks, measuring the Attack Success Rate (ASR).
- **Utility evaluation:** We evaluate on the Samsun dataset and the SQL Create Context dataset to measure downstream task performance.

5.3.1 ADVERSARIAL ATTACK METHODS

We evaluate the effectiveness of CSFT + PA loss against three types of adversarial attacks: Harmful Example Attacks, Identity Shifting Attacks, and Backdoor Poisoning Attacks.

- **Harmful Example Attacks:** These attacks introduce harmful examples into the training data, which attempt to mislead the model into generating unsafe or toxic responses.
- **Identity Shifting Attacks:** These attacks involve altering the model’s output to shift its identity, leading to biased or inaccurate outputs.

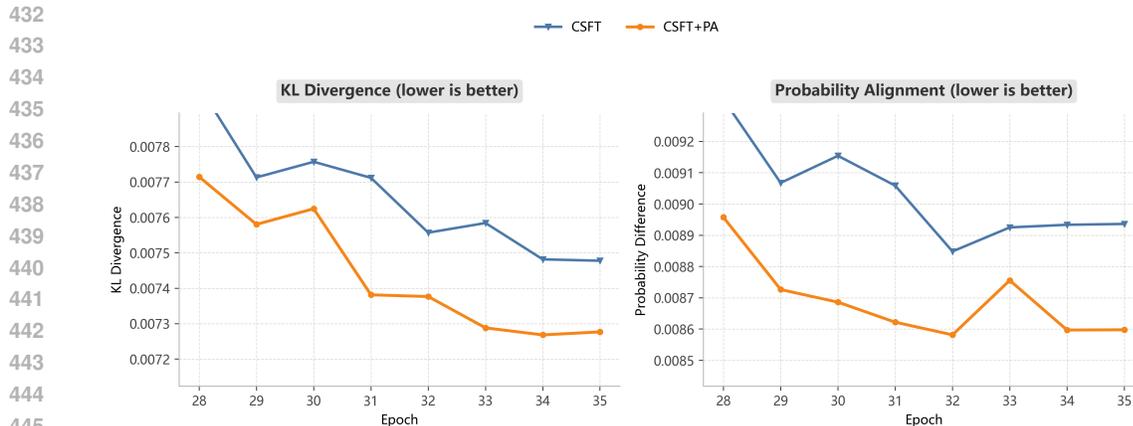


Figure 5: Results of Pre-experiment 2 (lower is better for KL Divergence and Probability Alignment). CSFT+PA loss leads to smaller distributional differences compared to CSFT alone (vs. CSFT baseline).

- **Backdoor Poisoning Attacks:** These attacks involve inserting poisoned data points into the training set, which cause the model to perform poorly on certain inputs. We consider both *trigger-free* and *trigger-based* backdoor attacks.

5.3.2 SAFETY EVALUATION AGAINST FINE-TUNING ATTACKS

The effectiveness of combining CSFT with PA loss in defending against adversarial attacks is summarized in Table 3, where we report the Attack Success Rate (ASR) for each attack category. Overall, the results demonstrate that CSFT + PA loss consistently and substantially improves safety across diverse threat models compared to both standard SFT and CSFT baselines.

More specifically, the results across both Llama2 and Gemma1.1 models indicate that:

- **Harmful Example Attacks:** On Llama2, CSFT + PA suppresses ASR from 88.9% under SFT to 2.7%, representing a 25.0% relative improvement over CSFT. On Gemma1.1, it reduces ASR from 81.6% to 0.6%, achieving a more pronounced 53.8% relative improvement over CSFT. These outcomes highlight PA’s potency in curtailing overt harmful behaviors overlooked by standard supervision.
- **Identity Shifting Attacks:** On Llama2, CSFT + PA decreases ASR to 7.5%, achieving a 7.4% relative improvement over CSFT. On Gemma1.1, it reaches 8.8% ASR, representing a 3.3% relative improvement over CSFT. These results demonstrate modest yet consistent enhancements even where CSFT already mitigates distributional drifts effectively.
- **Backdoor Poisoning Attacks:** Substantial reductions emerge in both trigger-free and trigger-based cases. For the trigger-based scenario, ASR drops to 3.3% on Llama2, representing a 52.3% relative improvement over CSFT, and to 0.9% on Gemma1.1, achieving a 52.6% relative improvement over CSFT. These affirm PA’s adaptive weighting in amplifying safeguards against latent deviations from safety-aligned references.

Taken together, these findings underscore that the proposed method provides safe and stable defense across attack categories. Importantly, the improvements are not confined to a specific type of adversarial manipulation but generalize to both data-poisoning and behavioral attacks, which is a key desideratum for practical safety alignment. Table 3 highlights these results in detail.

5.3.3 UTILITY EVALUATION

In addition to safety, we also evaluate the utility of the proposed approach on downstream tasks. Table 4 presents results on the Samsun, SQL Create Context, and GSM8K datasets across both Llama2 and Gemma1.1 models. Compared to CSFT, CSFT + PA incurs only minor performance degradation

Table 3: Evaluation of Attack Success Rate (ASR) under Fine-tuning Attacks

Attack Type	Llama2			Gemma1.1		
	SFT	CSFT	CSFT+PA	SFT	CSFT	CSFT+PA
Harmful Example (pure_bad)	88.9	3.6	2.7	81.6	1.3	0.6
Identity Shifting (aoa)	79.5	8.1	7.5	83.6	9.1	8.8
Backdoor Poisoning (w/o trigger)	7.6	1.9	1.5	2.0	1.5	0.6
Backdoor Poisoning (w/ trigger)	90.9	6.9	3.3	82.3	1.9	0.9

Table 4: Evaluation of Downstream Task Performance

Dataset	Llama2			Gemma1.1		
	SFT	CSFT	CSFT+PA	SFT	CSFT	CSFT+PA
Samsun	51.7	50.1	47.1	51.5	51.9	48.8
SQL Create Context	99.1	98.5	96.3	99.2	98.6	96.1
GSM8K	41.7	37.4	34.5	63.3	63.6	63.0

(within 8%), indicating that the improvements in safety do not come at the cost of substantial utility loss.

In particular, the most significant relative performance drop is observed on GSM8K for Llama2 (from 37.4% to 34.5%, a 7.8% relative degradation), while performance on Samsun and SQL Create Context decreases only marginally (6.0% and 2.2% relative drops, respectively). On Gemma1.1, drops are similarly modest: 6.0% on Samsun, 2.5% on SQL Create Context, and a minimal 0.9% on GSM8K. Such modest trade-offs are common in safety-alignment methods, and the observed magnitudes are well within acceptable bounds for practical deployment. The overall pattern suggests that CSFT + PA achieves a favorable safety–utility balance: it yields strong adversarial resistance while retaining high task competence.

In summary, Tables 3 and 4 demonstrate that CSFT + PA substantially strengthens safety against a wide range of adversarial attacks, with the maximum reduction in ASR reaching 52.6%. At the same time, the approach preserves downstream task performance with only minimal degradation. This balance between safety and utility is crucial for real-world applications, where adversarial resistance must be achieved without sacrificing core capabilities.

6 CONCLUSION

We introduced a preference-augmented alignment framework for mitigating the safety degradation of LLMs under domain-specific fine-tuning. By complementing token-level distributional alignment with preference signals, our method encourages models to favor the safe outputs of their pre-trained counterparts rather than merely imitating distributions. Extensive experiments demonstrate that this approach achieves a more favorable trade-off between safety and utility, and substantially improves robustness against adversarial fine-tuning.

Our findings suggest that preference signals can play a crucial role in strengthening intrinsic safety alignment, pointing toward a new direction for fine-tuning resistant safeguards. Future work may explore scaling our framework to broader alignment objectives, integrating human feedback more directly, and extending it to multi-modal or continual fine-tuning settings.

ETHICS STATEMENT

This work investigates methods to improve the safety of large language models. We only use publicly available datasets and avoid personal or sensitive information. While safety research may reveal potential risks, our intention is to strengthen responsible and trustworthy AI deployment.

540
541
542
543
544
545
546
547
548
549
550
551
552
553
554
555
556
557
558
559
560
561
562
563
564
565
566
567
568
569
570
571
572
573
574
575
576
577
578
579
580
581
582
583
584
585
586
587
588
589
590
591
592
593

REPRODUCIBILITY STATEMENT

We are committed to ensuring the reproducibility of our results. All datasets used in this work are publicly available, and we provide a detailed description of the improved methods in the main text. The experimental settings, including model architectures and training procedures, are outlined in the corresponding sections. To further facilitate reproducibility, we will release our source code upon the publication of the paper.

REFERENCES

- Yuntao Bai, Andy Jones, Kamal Ndousse, Amanda Askell, Anna Chen, Nova DasSarma, Dawn Drain, Stanislav Fort, Deep Ganguli, Tom Henighan, Nicholas Joseph, Saurav Kadavath, Jackson Kernion, Tom Conerly, Sheer El-Showk, Nelson Elhage, Zac Hatfield-Dodds, Danny Hernandez, Tristan Hume, Scott Johnston, Shauna Kravec, Liane Lovitt, Neel Nanda, Catherine Olsson, Dario Amodei, Tom Brown, Jack Clark, Sam McCandlish, Chris Olah, Ben Mann, and Jared Kaplan. Training a helpful and harmless assistant with reinforcement learning from human feedback, 2022.
- Léon Bottou, Frank E Curtis, and Jorge Nocedal. Optimization methods for large-scale machine learning. *SIAM Review*, 60(2):223–311, 2018.
- Olivier Bousquet and Andr’e Elisseeff. Stability and generalization. *Journal of Machine Learning Research*, 2(Mar):499–526, 2002.
- Stephen P. Boyd and Lieven Vandenberghe. *Convex Optimization*. Cambridge University Press, New York, NY, USA, 2004. ISBN 0521833787.
- Leo Gao, John Schulman, and Jacob Hilton. Scaling laws for reward model overoptimization. In *International Conference on Machine Learning*, pp. 10835–10866. PMLR, 2023.
- Guillaume Garrigos and Robert M Gower. Handbook of convergence theorems for (stochastic) gradient methods. *arXiv preprint arXiv:2301.11235*, 2023.
- Hamed Karimi, Julie Nutini, and Mark Schmidt. Linear convergence of gradient and proximal-gradient methods under the polyak-lojasiewicz condition. In Paolo Frasconi, Niels Landwehr, Giuseppe Manco, and Jilles Vreeken (eds.), *Machine Learning and Knowledge Discovery in Databases*, pp. 795–811, Cham, 2016. Springer International Publishing. ISBN 978-3-319-46128-1.
- Stanisław Łojasiewicz. Une propri’et’e topologique des sous-ensembles analytiques r’eels. In *Les ’Equations aux d’eriv’ees partielles*, pp. 87–89, Paris, 1963. ’Editions du Centre National de la Recherche Scientifique.
- Yurii Nesterov. *Introductory Lectures on Convex Optimization: A Basic Course*. Springer, 2004. ISBN 978-1-4613-4691-3.
- 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 *Advances in Neural Information Processing Systems*, volume 35, pp. 27730–27744, 2022.
- B. T. Polyak. Gradient methods for minimizing functionals. *Zhurnal Vychislitel’noi Matematiki i Matematicheskoi Fiziki*, 3(4):643–653, 1963.
- 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! *arXiv preprint arXiv:2310.03693*, 2023.
- Xiangyu Qi, Ashwinee Panda, Kaifeng Lyu, Xiao Ma, Subhrajit Roy, Ahmad Beirami, Prateek Mittal, and Peter Henderson. Safety alignment should be made more than just a few tokens deep. *arXiv preprint arXiv:2406.05946*, 2024.

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 A. Oh, T. Naumann, A. Globerson, K. Saenko, M. Hardt, and S. Levine (eds.), *Advances in Neural Information Processing Systems*, volume 36, pp. 53728–53741. Curran Associates, Inc., 2023. URL https://proceedings.neurips.cc/paper_files/paper/2023/file/a85b405ed65c6477a4fe8302b5e06ce7-Paper-Conference.pdf.

Herbert Robbins and Sutton Monro. A stochastic approximation method. *The Annals of Mathematical Statistics*, pp. 400–407, 1951.

John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*, 2017.

Gemini Team. Gemini: A family of highly capable multimodal models. *arXiv preprint arXiv:2312.11805*, 2023.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, et al. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*, 2023a.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, et al. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*, 2023b.

Jason Wei, Maarten Bosma, Vincent Y Zhao, Kelvin Guu, Adams Wei Yu, Brian Lester, Nan Du, Andrew M Dai, and Quoc V Le. Finetuned language models are zero-shot learners. *arXiv preprint arXiv:2109.01652*, 2021.

Tengyang Xie, Ching-An Cheng, Nan Jiang, Paul Mineiro, and Alekh Agarwal. Bellman-consistent pessimism for offline reinforcement learning. In *Advances in Neural Information Processing Systems*, pp. 34: 6683–6694, 2021.

Erhan Xu, Kai Ye, Hongyi Zhou, Luhan Zhu, Francesco Quinzan, and Chengchun Shi. Doubly robust alignment for large language models. *arXiv preprint arXiv:2506.01183*, 2025.

Xingcheng Xu. The policy cliff: A theoretical analysis of reward-policy maps in large language models. *arXiv preprint arXiv:2507.20150*, 2025.

A APPENDIX

A.1 PROOF OF LOSS FUNCTION CONVERGENCE

A.1.1 PROBLEM SETUP AND NOTATION

Consider the total loss function:

$$L_{\text{Total}}(\theta) = L_{\text{CSFT}}(\theta) + \delta_{\text{epoch}} \cdot L_{\text{PA}}(\theta)$$

where:

$$L_{\text{CSFT}}(\theta) = -\mathbb{E}_{(\mathbf{x}, \mathbf{y}) \sim D} \left[\sum_{t=1}^{|\mathbf{y}|} w_t \cdot \log \pi_{\theta}(y_t | \mathbf{x}, \mathbf{y}_{<t}) \right]$$

$$w_t = 2 \left\{ 1 - \sigma \left[\beta_t \left(\log \pi_{\theta}(y_t | \mathbf{x}, \mathbf{y}_{<t}) - \log \pi_{\text{aligned}}(y_t | \mathbf{x}, \mathbf{y}_{<t}) \right) \right] \right\}$$

$$L_{\text{PA}}(\theta) = -\mathbb{E}_{(\mathbf{x}, \mathbf{y}) \sim D} \left[\sum_{t=1}^{|\mathbf{y}|} \log \sigma \left(\mu_t \cdot \left(\log \pi_{\theta}(y_t, \text{aligned} | \mathbf{x}, \mathbf{y}_{<t}) - \log \pi_{\theta}(y_t, \theta | \mathbf{x}, \mathbf{y}_{<t}) \right) \right) \right]$$

$$\mu_t = D_{\text{KL}}(\pi_{\theta}(y_t | \mathbf{x}, \mathbf{y}_{<t}) \parallel \pi_{\text{aligned}}(y_t | \mathbf{x}, \mathbf{y}_{<t}))$$

$$\delta_{\text{epoch}} = 0.1 + 0.2 \times \frac{\text{epoch}}{\text{max_epoch}}$$

648 A.1.2 BASIC ASSUMPTIONS

649 To establish convergence, we adopt the following relatively mild assumptions, which are standard in
650 stochastic optimization and align with practical deep learning settings:

- 651 1. **Bounded Gradients:** There exists a constant $G > 0$ such that for any θ and any sample
652 (x, y) ,

$$653 \|\nabla_{\theta} \log \pi_{\theta}(y_t | \mathbf{x}, \mathbf{y}_{<t})\| \leq G.$$

- 654 2. **Lipschitz Continuity of Gradients:** There exists a constant $L > 0$ such that for any θ_1, θ_2 ,

$$655 \|\nabla L_{\text{Total}}(\theta_1) - \nabla L_{\text{Total}}(\theta_2)\| \leq L \|\theta_1 - \theta_2\|.$$

- 656 3. **Learning Rate Decay:** The learning rate sequence $\{\eta_k\}$ satisfies

$$657 \sum_{k=1}^{\infty} \eta_k = \infty, \quad \sum_{k=1}^{\infty} \eta_k^2 < \infty.$$

- 658 4. **Bounded Gradient Noise:** The stochastic gradient $g(\theta)$ satisfies

$$659 \mathbb{E}[g(\theta) | \theta] = \nabla L_{\text{Total}}(\theta), \quad \mathbb{E}[\|g(\theta) - \nabla L_{\text{Total}}(\theta)\|^2 | \theta] \leq \sigma^2.$$

- 660 5. **Bounded Weights:** There exists a constant $W > 0$ such that for all t , $|w_t| \leq W$.

- 661 6. **Bounded Log-Probability Differences:** There exists a constant $D > 0$ such that for all t
662 and θ , $|\log \pi_{\theta}(y_{t, \text{aligned}} | \mathbf{x}, \mathbf{y}_{<t}) - \log \pi_{\theta}(y_t, \theta | \mathbf{x}, \mathbf{y}_{<t})| \leq D$.

- 663 7. **Probability Lower Bound:** There exists a constant $\epsilon > 0$ such that for all $y_t, x, y_{<t}$, and θ ,
664 $\pi_{\theta}(y_t | \mathbf{x}, \mathbf{y}_{<t}) \geq \epsilon$. This can be enforced via logit clipping or label smoothing.

- 665 8. **Bounded KL Divergence:** There exists a constant $K > 0$ such that for all t and θ , $\mu_t \leq K$.
666 This holds in finite-vocabulary settings or can be enforced via KL clipping.

667 Discussion of Assumption Validity

668 Assumption 2 (Lipschitz continuity) ensures the smoothness of the loss gradient, a standard condi-
669 tion in stochastic optimization for deriving descent inequalities. It is not overly restrictive: in deep
670 learning models like Transformers, the loss is a composition of smooth functions (e.g., softmax and
671 cross-entropy), satisfying local Lipschitz properties in bounded parameter spaces. Unbounded pa-
672 rameters can be handled via weight decay or gradient clipping. Many activation functions, such as
673 the sigmoid in L_{PA} , have inherently Lipschitz gradients. In practice, gradient clipping enforces this
674 condition, and learning rates are typically chosen smaller than $1/L$ for stability.

675 Assumption 7 (probability lower bound) ensures well-defined KL divergences and gradients. It can
676 be practically achieved through logit clipping or label smoothing, common in language models.

677 Assumption 8 (bounded KL) is reasonable in finite-vocabulary models, where KL has a natural upper
678 bound $\log(1/\min q(y))$. In practice, KL regularization or clipping ensures numerical stability.

690 A.1.3 CONVERGENCE PROOF

691 Gradient Computation and Analysis

692 First, analyze the gradient of the total loss:

$$693 \nabla L_{\text{Total}}(\theta) = \nabla L_{\text{CSFT}}(\theta) + \delta_{\text{epoch}} \nabla L_{\text{PA}}(\theta).$$

694 CSFT Gradient

695 The gradient of the CSFT loss is:

$$696 \nabla L_{\text{CSFT}}(\theta) = -\mathbb{E}_{(\mathbf{x}, \mathbf{y}) \sim D} \left[\sum_{t=1}^{|\mathbf{y}|} w_t \cdot \nabla_{\theta} \log \pi_{\theta}(y_t | \mathbf{x}, \mathbf{y}_{<t}) \right].$$

Since w_t is treated as a constant via detachment, and by Assumption 5, $|w_t| \leq W$, combined with Assumption 1, we have:

$$\|\nabla L_{\text{CSFT}}(\theta)\| \leq W \cdot G \cdot T_{\max},$$

where $T_{\max} = \max |\mathbf{y}|$ is the maximum sequence length.

PA Gradient

The gradient of the PA loss is:

$$\nabla L_{\text{PA}}(\theta) = -\mathbb{E}_{(\mathbf{x}, \mathbf{y}) \sim \mathcal{D}} \left[\sum_{t=1}^{|\mathbf{y}|} \nabla_{\theta} \log \sigma(\mu_t \cdot \Delta_t) \right],$$

where $\Delta_t = \log \pi_{\theta}(y_{t, \text{aligned}} | \mathbf{x}, \mathbf{y}_{<t}) - \log \pi_{\theta}(y_{t, \theta} | \mathbf{x}, \mathbf{y}_{<t})$.

The gradient expands as:

$$\nabla_{\theta} \log \sigma(z_t) = (1 - \sigma(z_t)) \nabla_{\theta} z_t, \quad z_t = \mu_t \Delta_t.$$

Thus,

$$\nabla_{\theta} z_t = \mu_t \nabla_{\theta} \Delta_t + \Delta_t \nabla_{\theta} \mu_t,$$

where

$$\nabla_{\theta} \Delta_t = \nabla_{\theta} \log \pi_{\theta}(y_{t, \text{aligned}} | \mathbf{x}, \mathbf{y}_{<t}) - \nabla_{\theta} \log \pi_{\theta}(y_{t, \theta} | \mathbf{x}, \mathbf{y}_{<t}).$$

By Assumption 1, $\|\nabla_{\theta} \Delta_t\| \leq 2G$.

For $\nabla_{\theta} \mu_t$, since $\mu_t = D_{\text{KL}}(p \| q)$ with $p = \pi_{\theta}(\cdot | \mathbf{x}, \mathbf{y}_{<t})$ and fixed $q = \pi_{\text{aligned}}(\cdot | \mathbf{x}, \mathbf{y}_{<t})$, the gradient is:

$$\nabla_{\theta} \mu_t = \mathbb{E}_{(\mathbf{x}, \mathbf{y}) \sim \mathcal{D}} [\nabla_{\theta} \log p(y) \cdot (\log p(y) - \log q(y))].$$

By Assumption 1, $\|\nabla_{\theta} \log p(y)\| \leq G$. By Assumption 7, and assuming a lower bound on $\min q(y)$ (common in finite vocabularies), there exists $B > 0$ such that $|\log p(y) - \log q(y)| \leq B$, yielding $\|\nabla_{\theta} \mu_t\| \leq GB$.

By Assumption 8, $\mu_t \leq K$, and by Assumption 6, $|\Delta_t| \leq D$. Since $|1 - \sigma(z_t)| \leq 1$,

$$\|\nabla_{\theta} \log \sigma(z_t)\| \leq \mu_t \cdot 2G + |\Delta_t| \cdot GB \leq 2KG + DGB.$$

Thus, there exists a constant $C = T_{\max} \cdot (2KG + DGB)$ such that

$$\|\nabla L_{\text{PA}}(\theta)\| \leq C.$$

Bounded Total Gradient

Since $\delta_{\text{epoch}} \leq 0.3$, the total gradient is bounded:

$$\|\nabla L_{\text{Total}}(\theta)\| \leq WGT_{\max} + 0.3C = M.$$

Convergence Framework

Consider the stochastic gradient descent update:

$$\theta_{k+1} = \theta_k - \eta_k g(\theta_k),$$

where $g(\theta_k)$ is an unbiased estimator of $\nabla L_{\text{Total}}(\theta_k)$.

By Assumption 2, the pointwise descent lemma holds:

$$L_{\text{Total}}(\theta_{k+1}) \leq L_{\text{Total}}(\theta_k) + \nabla L_{\text{Total}}(\theta_k)^{\top} (\theta_{k+1} - \theta_k) + \frac{L}{2} \|\theta_{k+1} - \theta_k\|^2.$$

Substituting the update:

$$L_{\text{Total}}(\theta_{k+1}) \leq L_{\text{Total}}(\theta_k) - \eta_k \nabla L_{\text{Total}}(\theta_k)^{\top} g(\theta_k) + \frac{L}{2} \eta_k^2 \|g(\theta_k)\|^2. \quad (*)$$

A.1.4 DETAILED DERIVATION OF THE EXPECTED DESCENT INEQUALITY

In stochastic optimization, deriving the expected descent from the pointwise inequality requires careful handling of expectations. This section provides a rigorous derivation.

Monotonicity of Expectations

Theorem A.1 (Monotonicity of Conditional Expectations). *Let X and Y be random variables on a probability space, and let \mathcal{F} be a sub- σ -algebra. If $X \leq Y$ almost surely, then $\mathbb{E}[X | \mathcal{F}] \leq \mathbb{E}[Y | \mathcal{F}]$ almost surely.*

Proof. This follows from the definition of conditional expectation. For a detailed proof, see Billingsley (1995, Probability and Measure). \square

Application to Derive Conditional Expectation

Define $X = L_{\text{Total}}(\theta_{k+1})$ and

$$Y = L_{\text{Total}}(\theta_k) - \eta_k \nabla L_{\text{Total}}(\theta_k)^\top g(\theta_k) + \frac{L}{2} \eta_k^2 \|g(\theta_k)\|^2,$$

with \mathcal{F} the σ -algebra generated by θ_k . By Equation (*), $X \leq Y$ a.s. Thus, by Theorem 1,

$$\mathbb{E}[X | \theta_k] \leq \mathbb{E}[Y | \theta_k] \quad \text{a.s.}$$

By linearity of conditional expectations:

$$\mathbb{E}[Y | \theta_k] = L_{\text{Total}}(\theta_k) - \eta_k \nabla L_{\text{Total}}(\theta_k)^\top \mathbb{E}[g(\theta_k) | \theta_k] + \frac{L}{2} \eta_k^2 \mathbb{E}[\|g(\theta_k)\|^2 | \theta_k].$$

By Assumption 4, $\mathbb{E}[g(\theta_k) | \theta_k] = \nabla L_{\text{Total}}(\theta_k)$, so

$$\nabla L_{\text{Total}}(\theta_k)^\top \mathbb{E}[g(\theta_k) | \theta_k] = \|\nabla L_{\text{Total}}(\theta_k)\|^2.$$

For the variance term:

$$\begin{aligned} \mathbb{E}[\|g(\theta_k)\|^2 | \theta_k] &= \mathbb{E}[\|g - \nabla + \nabla\|^2 | \theta_k] \\ &= \mathbb{E}[\|g - \nabla\|^2 | \theta_k] + \|\nabla\|^2 + 2\mathbb{E}[(g - \nabla)^\top \nabla | \theta_k] \\ &\leq \sigma^2 + \|\nabla L_{\text{Total}}(\theta_k)\|^2, \end{aligned}$$

since the cross term is zero by unbiasedness.

Thus:

$$\mathbb{E}[L_{\text{Total}}(\theta_{k+1}) | \theta_k] \leq L_{\text{Total}}(\theta_k) - \eta_k \left(1 - \frac{L\eta_k}{2}\right) \|\nabla L_{\text{Total}}(\theta_k)\|^2 + \frac{L}{2} \eta_k^2 \sigma^2.$$

Taking full expectation (law of total expectation):

$$\mathbb{E}[L_{\text{Total}}(\theta_{k+1})] \leq \mathbb{E}[L_{\text{Total}}(\theta_k)] - \eta_k \left(1 - \frac{L}{2} \eta_k\right) \mathbb{E}[\|\nabla L_{\text{Total}}(\theta_k)\|^2] + \frac{L}{2} \eta_k^2 \sigma^2. \quad (**)$$

A.1.5 DETAILED DERIVATION OF THE CONVERGENCE CONCLUSION

From Equation (**), sum from $k = 1$ to K :

$$\sum_{k=1}^K (\mathbb{E}[L_{\text{Total}}(\theta_{k+1})] - \mathbb{E}[L_{\text{Total}}(\theta_k)]) \leq - \sum_{k=1}^K \eta_k \left(1 - \frac{L}{2} \eta_k\right) \mathbb{E}[\|\nabla\|^2] + \frac{L\sigma^2}{2} \sum_{k=1}^K \eta_k^2.$$

The left side telescopes to $\mathbb{E}[L_{\text{Total}}(\theta_{K+1})] - \mathbb{E}[L_{\text{Total}}(\theta_1)]$. Rearranging:

$$\sum_{k=1}^K \eta_k \left(1 - \frac{L}{2} \eta_k\right) \mathbb{E}[\|\nabla L_{\text{Total}}(\theta_k)\|^2] \leq \mathbb{E}[L_{\text{Total}}(\theta_1)] - \mathbb{E}[L_{\text{Total}}(\theta_{K+1})] + \frac{L\sigma^2}{2} \sum_{k=1}^K \eta_k^2.$$

810 Since $L_{\text{Total}} \geq 0$ (as a negative log-likelihood), $\mathbb{E}[L_{\text{Total}}(\theta_{K+1})] \geq 0$, so the sum is bounded above by
 811 a term that remains finite as $K \rightarrow \infty$ (due to $\sum \eta_k^2 < \infty$). Thus:

$$812 \sum_{k=1}^{\infty} \eta_k \left(1 - \frac{L}{2} \eta_k\right) \mathbb{E}[\|\nabla L_{\text{Total}}(\theta_k)\|^2] < \infty.$$

816 Assume for contradiction that $\liminf \mathbb{E}[\|\nabla\|^2] > 0$. Then there exists $\epsilon > 0$ and subsequence $\{k_j\}$
 817 with $\mathbb{E}[\|\nabla(\theta_{k_j})\|^2] \geq \epsilon$. For large j , $1 - (L/2)\eta_{k_j} > 1/2$, so the subsum diverges, contradicting the
 818 finite sum. Hence:

$$819 \liminf_{k \rightarrow \infty} \mathbb{E}[\|\nabla L_{\text{Total}}(\theta_k)\|^2] = 0.$$

822 A.2 PROOF OF LOSS FUNCTION ROBUSTNESS

824 A.2.1 INTRODUCTION

825 In this proof, we consider the given loss function form and rigorously prove its robustness. First,
 826 we clearly define 'robustness' in the context of optimization. Subsequently, through mathematical
 827 derivations, we analyze the optimization process, particularly focusing on what quantity's variation
 828 causes the policy π_θ to approach π_{aligned} . Finally, we provide a quantitative proof using weaker
 829 assumptions (such as convexity, Lipschitz gradient continuity, and the Polyak-Łojasiewicz (PL) in-
 830 equality, rather than strong convexity). These assumptions are more general and applicable to cer-
 831 tain non-strongly convex but locally well-behaved loss functions, as commonly encountered in deep
 832 learning scenarios.

833 To align with theoretical analyses in related literature, such as 'The Policy Cliff: A Theoretical Anal-
 834 ysis of Reward-Policy Maps in Large Language Models,' we emphasize how regularization terms
 835 like $L_{\text{PA}}(\theta)$ resolve degeneracies in optima, preventing 'policy cliffs' (discontinuous policy shifts
 836 under perturbations) by acting as tie-breakers in cases of non-unique optimal actions.

838 A.2.2 DEFINITION OF ROBUSTNESS

840 **Definition A.1 (Robustness).** *In optimization problems, the robustness of the loss function $L(\theta)$*
 841 *refers to the system's ability to maintain its performance and stability in the face of uncertainty*
 842 *or perturbations. Specifically, uncertainty may manifest as noise perturbations in the input data D*
 843 *(such as label noise or input variations). We quantify the perturbation size through the noise intensity*
 844 *$\epsilon > 0$, representing the maximum amplitude of data deviation.*

845 *Quantitatively, the loss function $L(\theta)$ is considered robust if, for a noise perturbation ϵ , the perturbed*
 846 *optimal solution $\theta^{*,\epsilon}$ and the original optimal solution θ^* satisfy:*

$$847 \|\theta^{*,\epsilon} - \theta^*\| \leq K\epsilon,$$

848 where K is a Lipschitz-related constant. Here, we uniformly use the Euclidean norm $\|\cdot\|$ in the
 849 parameter space to measure changes in solutions, ensuring consistency. This ensures that changes
 850 in the output (optimal solution or policy π_θ) are linearly bounded by the perturbation size.

851 In cases where optima are non-unique (degenerate), perturbations can lead to discontinuous shifts,
 852 akin to 'policy cliffs' in reward-policy maps. Our assumptions (e.g., PL inequality) ensure unique-
 853 ness, mitigating such issues.

854 In this context, uncertainty primarily refers to noise in the data distribution D , characterized by ϵ .
 855 We will prove that the total loss $L_{\text{Total}}(\theta)$, by incorporating the $L_{\text{PA}}(\theta)$ term, enhances robustness to
 856 noise. Specifically, L_{PA} acts as a regularization term that strengthens the PL inequality constant μ ,
 857 thereby tightening the robustness bound.

862 A.2.3 REVIEW OF THE LOSS FUNCTION

863 The total loss function is:

$$L_{\text{Total}}(\theta) = L_{\text{CSFT}}(\theta) + \delta_{\text{epoch}} \cdot L_{\text{PA}}(\theta),$$

where

$$L_{\text{CSFT}}(\theta) = -\mathbb{E}_{(\mathbf{x}, \mathbf{y}) \sim D} \left[\sum_{t=1}^{|\mathbf{y}|} w_t \cdot \log \pi_{\theta}(y_t | \mathbf{x}, \mathbf{y}_{<t}) \right],$$

$$w_t = 2 \left\{ 1 - \sigma \left[\beta_t \left(\log \pi_{\theta}(y_t | \mathbf{x}, \mathbf{y}_{<t}) - \log \pi_{\text{aligned}}(y_t | \mathbf{x}, \mathbf{y}_{<t}) \right) \right] \right\},$$

$$L_{\text{PA}}(\theta) = -\mathbb{E}_{(\mathbf{x}, \mathbf{y}) \sim D} \left[\sum_{t=1}^{|\mathbf{y}|} \log \sigma \left(\mu_t \cdot \left(\log \pi_{\theta}(y_{t, \text{aligned}} | \mathbf{x}, \mathbf{y}_{<t}) - \log \pi_{\theta}(y_{t, \theta} | \mathbf{x}, \mathbf{y}_{<t}) \right) \right) \right],$$

$$\mu_t = D_{\text{KL}}(\pi_{\theta}(y_t | \mathbf{x}, \mathbf{y}_{<t}) \| \pi_{\text{aligned}}(y_t | \mathbf{x}, \mathbf{y}_{<t})).$$

Note that $y_{t, \text{aligned}}$ and $y_{t, \theta}$ are the argmax predictions of π_{aligned} and π_{θ} at position t (assuming softmax outputs as probability distributions, taking the maximum probability class). To handle the non-differentiability of argmax, we implicitly use a softened version (such as temperature-scaled softmax approximation) to ensure gradient flow. w_t is treated as a constant in gradient computations (via detach operation) to avoid overfitting to noise. δ_{epoch} is a scheduling parameter that increases with epochs, used to gradually strengthen the regularization effect.

A.2.4 ANALYSIS OF THE OPTIMIZATION PROCESS: THE KEY FACTOR DRIVING POLICY ALIGNMENT

During optimization, we use gradient descent to minimize $L_{\text{Total}}(\theta)$. The update rule is $\theta \leftarrow \theta - \eta \nabla_{\theta} L_{\text{Total}}(\theta)$, where η is the learning rate.

The key question is: what quantity's variation causes π_{θ} to approach π_{aligned} . The answer lies in the gradient contribution of $L_{\text{PA}}(\theta)$. Specifically, the variation in μ_t (i.e., changes in KL divergence) drives this process. We will compute the gradients in detail to demonstrate this.

First, consider the gradient of $L_{\text{PA}}(\theta)$:

$$\nabla_{\theta} L_{\text{PA}}(\theta) = -\mathbb{E}_{(\mathbf{x}, \mathbf{y}) \sim D} \left[\sum_{t=1}^{|\mathbf{y}|} \nabla_{\theta} \log \sigma(\mu_t \cdot \Delta_t) \right],$$

where $\Delta_t = \log \pi_{\theta}(y_{t, \text{aligned}} | \mathbf{x}, \mathbf{y}_{<t}) - \log \pi_{\theta}(y_{t, \theta} | \mathbf{x}, \mathbf{y}_{<t})$.

Let $z_t = \mu_t \cdot \Delta_t$, then the gradient of $\log \sigma(z_t)$ is:

$$\nabla_{\theta} \log \sigma(z_t) = \frac{1}{\sigma(z_t)} \cdot \sigma'(z_t) \cdot \nabla_{\theta} z_t.$$

Since $\sigma'(z) = \sigma(z)(1 - \sigma(z))$, we have:

$$\frac{\sigma'(z_t)}{\sigma(z_t)} = 1 - \sigma(z_t),$$

thus:

$$\nabla_{\theta} \log \sigma(z_t) = (1 - \sigma(z_t)) \nabla_{\theta} z_t.$$

Next, compute $\nabla_{\theta} z_t$:

$$\nabla_{\theta} z_t = \Delta_t \cdot \nabla_{\theta} \mu_t + \mu_t \cdot \nabla_{\theta} \Delta_t.$$

Here, $\mu_t = D_{\text{KL}}(\pi_\theta \| \pi_{\text{aligned}})$, and its gradient is:

$$\nabla_\theta D_{\text{KL}}(\pi_\theta \| \pi_{\text{aligned}}) = \mathbb{E}_{(\mathbf{x}, \mathbf{y}) \sim \mathcal{D}} \left[\nabla_\theta \log \pi_\theta(y_t | \mathbf{x}, \mathbf{y}_{<t}) \log \frac{\pi_\theta(y_t | \mathbf{x}, \mathbf{y}_{<t})}{\pi_{\text{aligned}}(y_t | \mathbf{x}, \mathbf{y}_{<t})} \right].$$

For $\nabla_\theta \Delta_t$:

$$\nabla_\theta \Delta_t = \nabla_\theta \log \pi_\theta(y_{t, \text{aligned}} | \mathbf{x}, \mathbf{y}_{<t}) - \nabla_\theta \log \pi_\theta(y_{t, \theta} | \mathbf{x}, \mathbf{y}_{<t}).$$

When μ_t is large (high KL divergence, misaligned positions), the $\Delta_t \nabla_\theta \mu_t$ term dominates, amplifying the gradient to push for KL reduction. Conversely, low μ_t weakens the gradient. δ_{epoch} controls the weight of this term.

Thus, the quantity driving π_θ toward π_{aligned} is the variation in μ_t , i.e., the reduction in KL divergence, achieved through the dynamic adjustment of the regularization effect in L_{PA} .

This mechanism aligns with tie-breaking in degenerate optima: high KL indicates non-unique actions, and L_{PA} resolves this by favoring aligned policies, preventing rational exploitation of incomplete losses (similar to 'clever slacker' behaviors in policy cliffs literature).

A.2.5 QUANTITATIVE PROOF OF ROBUSTNESS

We assume the loss function $L_{\text{Total}}(\theta)$ satisfies convexity, its gradient $\nabla_\theta L_{\text{Total}}(\theta)$ is L -Lipschitz continuous, and the Polyak-Łojasiewicz (PL) inequality:

$$\frac{1}{2} \|\nabla L_{\text{Total}}(\theta)\|^2 \geq \mu(L_{\text{Total}}(\theta) - L_{\text{Total}}(\theta^*)),$$

where $\mu > 0$ is a constant. The PL inequality and convexity ensure the existence and uniqueness of minimizers, as well as convergence rates in optimization, while not directly required for the parameter bound derivation below. Introducing $L_{\text{PA}}(\theta)$ can increase μ , as the KL divergence regularization enhances the lower bound on the gradient norm. Specifically, through Hessian analysis, L_{PA} contributes positive definite terms to the second derivatives, increasing the effective curvature lower bound (refer to optimization literature such as Karimi et al.). For a sketch: the Hessian of L_{PA} involves terms like $\nabla^2 D_{\text{KL}}$, which is positive semi-definite for entropy-like regularizers, thus boosting the minimal eigenvalue related to μ .

Lemma A.1 (Perturbation Bounds). *Consider noisy data $D^\epsilon = D + \epsilon\xi$, where ξ is bounded noise, $\|\xi\| \leq 1$. Then $L_{\text{Total}}^\epsilon(\theta) = L_{\text{Total}}(\theta) + \epsilon \cdot g(\theta, \xi)$, where g is a bounded function, $|g| \leq M$.*

Additionally, for the gradients, $\|\nabla_\theta L_{\text{Total}}^\epsilon(\theta) - \nabla_\theta L_{\text{Total}}(\theta)\| \leq \epsilon G$, where G is the bound on gradient perturbations.

Proof: By the linearity of expectations and the Lipschitz nature of continuous functions, the noise linearly affects the loss and gradients. Specifically, for each expectation term, the difference due to perturbation is linearly controlled by ϵ , yielding $|L_{\text{Total}}^\epsilon(\theta) - L_{\text{Total}}(\theta)| \leq \epsilon M$. Applying the chain rule to gradients, each derivative term's perturbation is also linear, so $\|\nabla_\theta L_{\text{Total}}^\epsilon(\theta) - \nabla_\theta L_{\text{Total}}(\theta)\| \leq \epsilon G$.

Theorem A.2 (Robustness Bound). *Let θ^* be the minimizer of $L_{\text{Total}}(\theta)$, and $\theta^{*,\epsilon}$ the minimizer of $L_{\text{Total}}^\epsilon(\theta)$. Assuming the gradient $\nabla_\theta L_{\text{Total}}(\theta)$ is L -Lipschitz continuous and the gradient perturbation satisfies $\|\nabla_\theta L_{\text{Total}}^\epsilon(\theta) - \nabla_\theta L_{\text{Total}}(\theta)\| \leq \epsilon G$, then*

$$\|\theta^{*,\epsilon} - \theta^*\| \leq \frac{\epsilon G}{L}.$$

Proof: Since θ^* and $\theta^{*,\epsilon}$ are minimizers, we have $\nabla L_{\text{Total}}(\theta^*) = 0$ and $\nabla L_{\text{Total}}^\epsilon(\theta^{*,\epsilon}) = 0$. From the gradient perturbation assumption,

$$\|\nabla L_{\text{Total}}(\theta^{*,\epsilon})\| = \|\nabla L_{\text{Total}}(\theta^{*,\epsilon}) - \nabla L_{\text{Total}}^\epsilon(\theta^{*,\epsilon})\| \leq \epsilon G.$$

By Lipschitz gradient continuity,

972
 973
 974
 975
 976
 977
 978
 979
 980
 981
 982
 983
 984
 985
 986
 987
 988
 989
 990
 991
 992
 993
 994
 995
 996
 997
 998
 999
 1000
 1001
 1002
 1003
 1004
 1005
 1006
 1007
 1008
 1009
 1010
 1011
 1012
 1013
 1014
 1015
 1016
 1017
 1018
 1019
 1020
 1021
 1022
 1023
 1024
 1025

$$\|\nabla L_{\text{Total}}(\theta^{*,\epsilon}) - \nabla L_{\text{Total}}(\theta^*)\| \leq L\|\theta^{*,\epsilon} - \theta^*\|.$$

Since $\nabla L_{\text{Total}}(\theta^*) = 0$,

$$\|\nabla L_{\text{Total}}(\theta^{*,\epsilon})\| \leq L\|\theta^{*,\epsilon} - \theta^*\|.$$

Combining the inequalities,

$$\|\nabla L_{\text{Total}}(\theta^{*,\epsilon})\| \leq \epsilon G \leq L\|\theta^{*,\epsilon} - \theta^*\|,$$

thus

$$\|\theta^{*,\epsilon} - \theta^*\| \leq \frac{\epsilon G}{L}.$$

This bound shows that parameter changes are linearly related to the perturbation size ϵ , proving robustness. Introducing L_{PA} can reduce the effective Lipschitz constant L (through smoothing) or decrease G (reducing noise sensitivity), thereby tightening the bound. The PL inequality and convexity ensure minimizer existence and uniqueness but do not directly participate in deriving the parameter bound.

A.3 THE USE OF LARGE LANGUAGE MODELS

In this work, we employ large language models (LLMs) primarily as assistants for writing. Their role is limited to aiding the authors in polishing the presentation and improving readability.