# To Intervene or Not: Guiding Inference-time Alignment with Probabilistic Model Blending

**Jin Gan, Xin Li*, Jun Luo**

College of Computing and Data Science, Nanyang Technological University, Singapore
{jin010, l.xin, junluo}@ntu.edu.sg

## Abstract

The wide deployment of LLMs has made model alignment necessary to make newly trained models safely and effectively respond to user instructions. Among different methods, inference-time alignment is often cheaper as it intervenes (i.e., offers guidances) only during output generation. Existing proposals apply guidances extracted from certain aligned models without properly assessing their reliability. Nonetheless, our systematic evaluation reveals that guidance effectiveness varies drastically across models; since ineffective guidances lead to further confusion and thus further interventions, the resulting excessive interventions typically indicate poor performance. To make interventions more effective and thus more efficient, we introduce BlendIn, an inference-time alignment framework that shifts from binary decisions to creating hybrid distributions integrating both models' knowledge. BlendIn stabilizes inference-time alignment by performing quality-aware alignment and proportionally weighting each model's contribution based on reliability. Compared with existing works, it preserves beneficial guidance while downweighting unreliable suggestions. BlendIn provides both diagnostic signals and mitigation strategies for misaligned guidance, achieving consistent and up to 50% performance improvement on challenging model pairs. Our code is available at: `https://github.com/DecayingSeart/BlendIn`.

## 1 Introduction

The helpfulness and safety of large language models (LLMs) largely depends on their alignment to follow user instructions (Fei et al., 2025). This is traditionally achieved through fine-tuning methods where alignment must be performed separately for each model newly trained (Ouyang et al., 2022; Rafailov et al., 2023), incurring substantial computational costs. Such inefficiency has motivated inference-time alignment, which leverages aligned models or their extracted signals (Fei et al., 2025; Liu et al., 2024; Wang et al., 2024) as *guidance models* to align unaligned models as *base models* (Wang et al., 2024; Fei et al., 2025; Liu et al., 2024) during output generation, thereby avoiding expensive retraining.

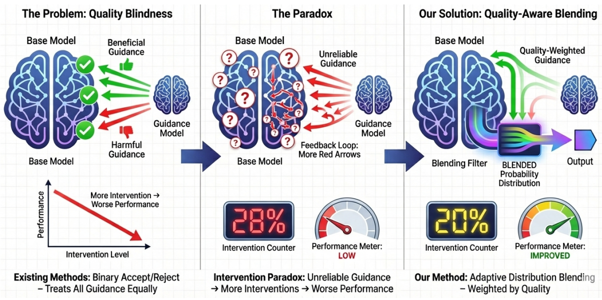
Existing works on inference-time alignment provides guidance in different forms of signals like token suggestions, value scores, or activation steering (Fei et al., 2025; Wang et al., 2024; Liu et al., 2024). However, these methods lack mechanisms to assess if the guidance itself is reliable. Their designs implicitly assume that all guidance is beneficial— an assumption that our systematic analysis refutes. Across nine models (Team et al., 2025, 2024; Grattafiori et al., 2024; Yang et al., 2025) and six datasets (Cobbe et al., 2021; Hendrycks et al., 2021b,a; Lin et al., 2022; Clark et al., 2018; Lin et al., 2023), we observe a dramatic variance on guidance effectiveness: some model combinations succeed while others fail catastrophically. Critically, model pairs with excessive intervention rates systematically perform worse, not better. This counterintuitive pattern reveals that guidance effectiveness varies fundamentally across different model combinations.

Why does this happen? When base models encounter difficult positions in generation where they have misaligned or harmful prediction, guidance models may also struggle at these same positions—they make incorrect predictions about what tokens should come next. As a result, these models provide misaligned or harmful suggestions, leading to misaligned outputs that trigger even more interventions. Because existing methods cannot distinguish beneficial suggestions from harmful ones, they exhibit a fundamental problem of quality blindness. Without mechanisms to predict or mitigate this issue, alignment success must rely on expensive trial-and-error testing, limiting practical

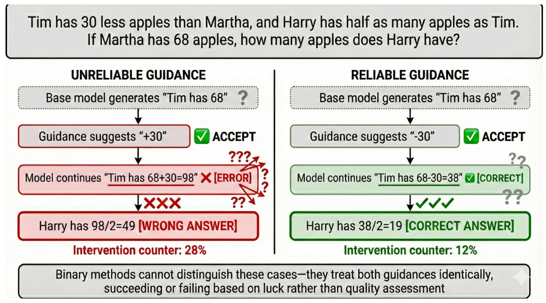
*Corresponding author.



*Findings of the Association for Computational Linguistics: ACL 2026*, pages 17158–17169
July 2-7, 2026 ©2026 Association for Computational Linguistics




(a)




(b)

Figure 1: (a) Overview of quality blindness in inference-time alignment and our solution. **Left:** Existing methods make binary accept/reject decisions, treating all guidance equally without assessing quality— accepting both beneficial (green) and harmful (red) suggestions identically. **Center:** This leads to an intervention paradox: unreliable guidance triggers cascading failures that require more interventions, creating a negative correlation between intervention rate and performance. **Right:** BlendIn addresses this through quality-aware distribution blending, weighting guidance proportionally to reliability rather than making binary decisions. (b) Cascading failure from unreliable guidance. **Left:** When unreliable guidance suggests an incorrect token ('+' instead of '-'), binary acceptance propagates this error through subsequent steps, creating uncertainty that triggers additional interventions (28% intervention rate) and produces wrong answers. **Right:** Reliable guidance providing correct suggestions enables accurate generation with minimal intervention (12%). Existing binary methods cannot distinguish these cases, treating both guidances identically despite opposite outcomes.

deployment of inference-time alignment. See Figure 1a for overview and Figure 1b for example.

These observations suggest that effective inference-time alignments require a shift in approach. Existing methods' quality blindness leads to binary accept/reject decisions that cannot express partial trust or selective integration. When guidance is unreliable, binary methods must choose between accepting harmful suggestions (degrading performance) or rejecting all guidance (losing potential benefits). Neither is satisfactory.

We argue that quality-aware integration is essential: methods must assess guidance reliability at each intervention position and integrate both models' knowledge proportionally rather than making all-or-nothing decisions. This enables critical capabilities that binary methods lack. First, it preserves beneficial portions of guidance while downweighting unreliable portions. Second, it leverages partial knowledge from both models simultaneously. Third, it gracefully handles varying degrees of guidance reliability.

Building on these insights, we propose BlendIn, a quality-aware inference time alignment method based on blending distributions. At each position where base model demonstrates low confidence in prediction, BlendIn evaluates guidance quality by blending the full probability distributions from both guidance and base models using adaptive weights based on their respective certainties. Rather than making binary decisions, we sample from the hybrid distribution using greedy selection, allowing both models to contribute proportionally. This soft integration effectively filters misaligned or harmful interventions while preserving beneficial guidance, addressing the quality blindness in existing approaches. Our approach achieves consistent and maximum 50% improvements on challenging high-intervention pairs, demonstrating its effectiveness and robustness.

Our contributions include:

- Making novel contributions toward systematic characterization of quality failures in inference-time alignment, identifying the negative correlation between excessive intervention rate and overall performance.
- Proposing BlendIn, an inference-time alignment stabilization method that diagnoses and mitigates unreliable guidance through soft distribution blending.
- Improving performance on challenging over-intervened model pairs consistently and up to 50%.

Our work transforms inference-time alignment from an empirically promising but insufficiently predictable technique into one with principled diagnostic signals and effective mitigation to misaligned guidance. The rest of the paper is organized as follows. Section 2 briefly captures related works. Section 3 presents problem formulation. Section 4 discusses details of our method. Section 5 reports the experimental results. Finally, section 6 concludes the paper.

## 2 Related Works

Inference-time alignment methods guide unaligned models using signals from aligned models without parameter updates. NUDGING (Fei et al., 2025) uses speculative decoding (Leviathan et al., 2023; Chen et al., 2023) where guidance models propose tokens whenever the top 1 probability of base model falls under a threshold. IVG (Liu et al., 2024) applies a value function trained on outputs of an aligned model to pick the highest-scored token candidate in base model. InferAligner (Wang et al., 2024) selectively shifts base model's activations using vectors extracted from aligned models whenever harmful query is detected. While these recent methods differ in mechanism (token proposals, value scoring, and activation modification), they share a common design principle of treating guidance as uniformly beneficial. However, they lack mechanisms to assess whether specific guidance models actually provide aligned suggestions for base models. This problem remains uncharacterized: can alignment success be predicted by any property that is easy to obtain, or must every model combination undergo expensive empirical testing on full benchmarks? Can we neutralize such guidance quality failure in advance? Without systematic analysis of these questions, practitioners lack diagnostic signals for rapid failure detection and principled strategies for failure mitigation.

## 3 Problem Formulation

### 3.1 Preliminary

Inference-time alignment could align base models using guidance models during output generation. This could be implemented through speculative decoding (Leviathan et al., 2023; Chen et al., 2023), where an aligned model proposes tokens and the unaligned model validates them based on an uncertainty threshold (Fei et al., 2025).

Formally, at each generation step $t$, the base model first checks its own confidence. Guidance is triggered only at uncertain positions:

1. **Uncertainty check**: Compute $u = \max_w P_{M_b}(w|x_{<t})$; if $u \geq \tau$ (uncertainty threshold), select $\arg\max_w P_{M_b}(w|x_{<t})$ and continue to the next position
2. **Proposal phase** (only when $u < \tau$): Aligned model $M_s$ generates candidate tokens $\{w_1, \ldots, w_k\}$

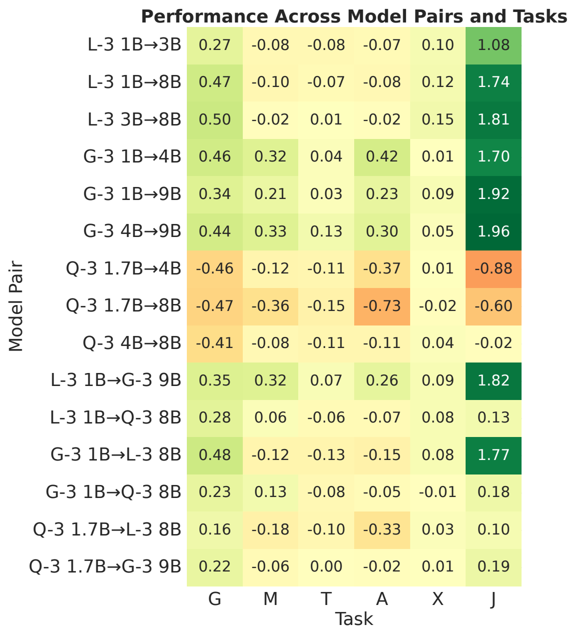



Figure 2: Difference between inference-time aligned performance and unaligned base model performance, showing if alignment has made performance better or worse. Red cells indicate degradation while green cells indicate improvement. Each model pair shows a guidance model aligning a base model by an arrow sign. L, G, and Q refer to Llama, Gemma and Qwen, while tasks are abbreviations of datasets GSM8K, MMLU, TruthfulQA, ARC-Challenge, XSTest and JustEval-Safe. Performance varies dramatically across guidance sources and benchmarks.

3. **Acceptance** (only when $u < \tau$): Accept the guidance model's top token $\arg\max_w P_{M_s}(w|x_{<t})$ and append to sequence
4. **Continuation**: $M_b$ continues generation until the next uncertain position

This design is simple as it doesn't require training extra models or functions.

### 3.2 Quality Blindness

Existing methods apply guidance through different mechanisms, but they lack mechanisms to evaluate guidance quality as they treat all guidance as uniformly beneficial. They cannot distinguish between aligned, helpful suggestions from misaligned, harmful suggestions. When the base model makes misaligned or harmful prediction at a difficult position, guidance models may share the same situation. Both models are effectively unreliable in this case, yet existing methods accept these unreliable suggestions without quality assessment.

Figure 2 presents our systematic evaluation on baseline alignment method (Fei et al., 2025) across

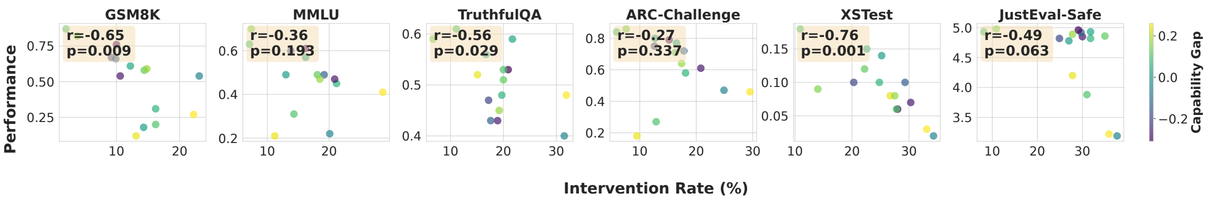



Figure 3: Intervention paradox: Higher intervention rates correlate with worse performance. Each point represents a model pair on a specific benchmark. Negative correlations are statistically significant on GSM8K, TruthfulQA, and XSTest. This contradicts the assumption that all guidance improves alignment. The paradox reveals that excessive intervention (>20% of generated tokens in general, threshold may vary with the specific benchmark) actually degrades performance, making intervention rate a diagnostic signal of poor guidance quality.

nine models, three model families and six benchmarks. Performance for GSM8K (Cobbe et al., 2021), MMLU (Hendrycks et al., 2021b,a), TruthfulQA (Lin et al., 2022), ARC-Challenge (Clark et al., 2018), and XSTest (Röttger et al., 2024) is measured by accuracy (0-1 scale), while performance for JustEval-Safe (Lin et al., 2023) is measured by average safety score (1-5 scale, higher is better). These metrics apply to all subsequent figures unless otherwise noted. Contrary to the implicit assumption that any guidance is beneficial, we observe widespread failures that reveal critical insights. First, model family matters: When Qwen serves as the guidance model, its within-family pairs systematically fail across all benchmarks, while its cross-family pairs show a mix of successes and smaller failures. In contrast, Llama and Gemma demonstrate comparable successes both within and across families. This unpredictability greatly hinders practical deployment as we cannot reliably predict which model combinations may work without exhaustive testing. Second, failures are not uniformly distributed. The same base model can have significant performance variance solely due to difference in guidance models. This suggests that guidance quality varies fundamentally across guidance models, yet existing methods lack mechanisms to account for these differences, creating a quality blindness problem. These results motivate two questions that we later address: (1) Why do certain model pairs fail catastrophically? (2) Can we predict or mitigate failures without exhaustive testing?

Deeper empirical analysis reveals an intervention paradox that further confirms the quality blindness problem. As shown in Figure 3, intervention rate demonstrates a negative correlation with overall performance, statistically significant in benchmarks of GSM8K, TruthfulQA and XSTest. While this doesn't mean inference time alignment is not useful, excessive interventions (guided generation exceeds 20% of the tokens generated) systematically show worse performance. This correlation implies that guidance models may as well struggle at positions where base model needs guidance and give misaligned or harmful suggestions. Such suboptimal suggestion creates misalignment in the generated sequence, introducing more problematic positions and hence more interventions. Existing methods have overlooked this and treat confident, helpful suggestions identically to misaligned, harmful ones, failing to detect or mitigate such problematic guidance.

Further, this problem cannot be explained by superficial mechanisms, implying deeper significance. At first, we naively hypothesized that such failure is caused by tokenization mismatches. If the guidance model's top suggestion doesn't appear or ranks poorly in the base model's vocabulary distribution, guidance would be expected to become ineffective as the base model cannot continue coherently. We computed vocabulary overlap by generating 100 tokens under guidance and measured how often the guidance model's suggested token appears in the base model's top-k token candidates. We measure top-K overlap, which is the percentage of times where guidance model's suggestion is in base model's top-K. We also measure average rank, which is the mean rank of guidance model's suggestions in base model's distribution. We further computed Pearson correlation between vocabulary metrics and performance across all six benchmarks. As shown in Figure 4, we found no statistically significant correlation between vocabulary overlap and overall performance. High-overlap pairs like Qwen-to-Llama can perform poorly, while low-overlap pairs like Gemma-to-Llama can perform well (See Figure 2 for performance). This result rejects the naive hypothesis, suggesting deeper root cause and proving this issue non-trivial. Results using 90% probability mass coverage are provided in Appendix A.1, confirm-

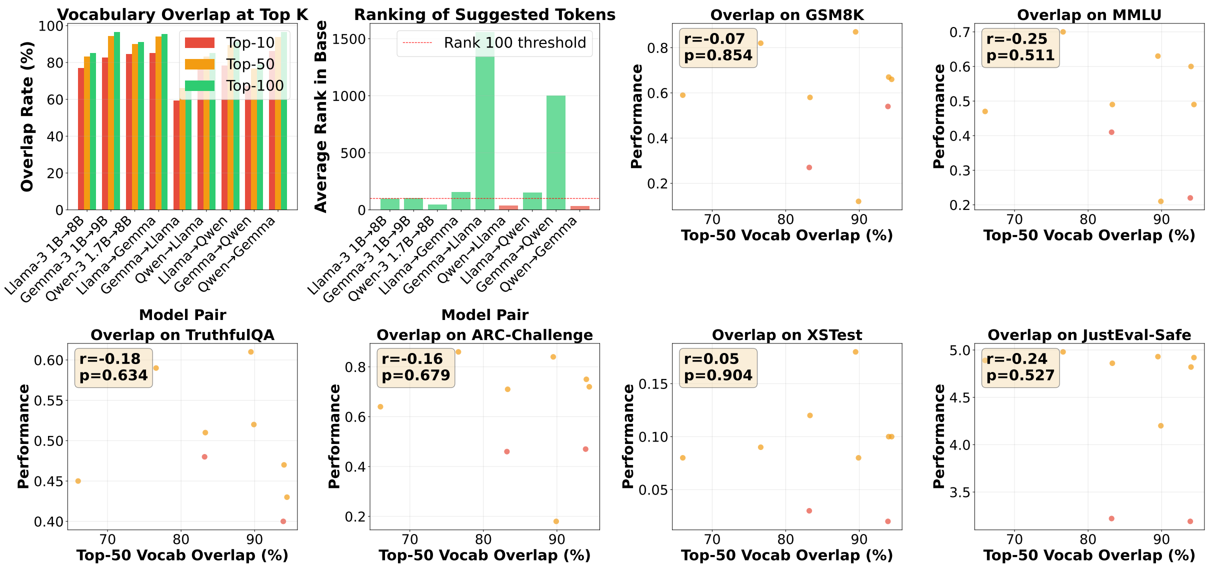



Figure 4: Vocabulary overlap does not predict performance. We measure top-50 vocabulary overlap, which is the percentage of suggested tokens from guidance model appearing in base model's top-50 predictions. Despite high overlap, pairs like Qwen→Llama could fail catastrophically, while low-overlap pairs like Gemma→Llama could succeed. No significant correlation between overlap and performance on any benchmark is found, rejecting the hypothesis that surface-level vocabulary alignment determines guidance quality.

ing the null finding is robust to overlap metric.

### 3.3 Our Solution

Given the intervention paradox, a natural question arises: since high intervention correlates with bad performance, can we simply cap intervention rate to improve alignment? We tested limiting intervention rate to 15% by rejecting all guidance once the threshold is exceeded. We chose 15% as a conservative threshold below the empirically observed 20% to show even lenient limits degrade performance by discarding all guidance indiscriminately.

As Table 3 shows, intervention rate capping fails to address quality problems. Limiting interventions to 15% degrades performance further on challenging pairs, as it removes both good and bad guidance indiscriminately. For guidance models with weak capability, removing bad guidance doesn't help if the base model truly needs assistance. For those with strong capability, capping removes beneficial guidance. This validates that the problem is guidance quality rather than quantity. We need a selective mechanism to assess each suggestion's quality rather than a hard cap on intervention. High intervention rate is a symptom of guidance models making misaligned or harmful suggestions, not the cause of poor performance.

The intervention paradox exposes the fundamental gap of quality-assessment mechanisms. The null result of vocabulary analysis and the failure of naive solutions suggest the challenge of this problem. For solution, what's needed is not binary filtering but quality-aware integration that greedily selects candidate tokens from both models proportionally to their quality. This motivates our approach of BlendIn, which we describe in Section 4. Meanwhile, the paradox reveals intervention rate as a diagnostic signal that predicts model incompatibility early from a small data subset rather than full benchmark evaluation.

## 4 BlendIn Method

### 4.1 Distribution Blending Method

BlendIn is designed as a stabilization technique for inference-time alignment: rather than replacing or overriding the base model's generation, it softly integrates guidance to prevent cascading failures.

At each generation step $t$, given context $x_{<t}$:

- Base model $M_b$ produces distribution $P_b(w|x_{<t})$ over vocabulary $\mathcal{V}$
- Guidance model $M_g$ produces distribution $P_g(w|x_{<t})$ over $\mathcal{V}$
- Base model's uncertainty: $u = \max_w P_b(w|x_{<t})$

When base is uncertain ($u < \tau$), integrate guidance in a quality-aware manner.

As illustrated in Figure 5, our approach consists of three steps at each generation position where base model is uncertain:

**Step 1: Obtain both distributions**

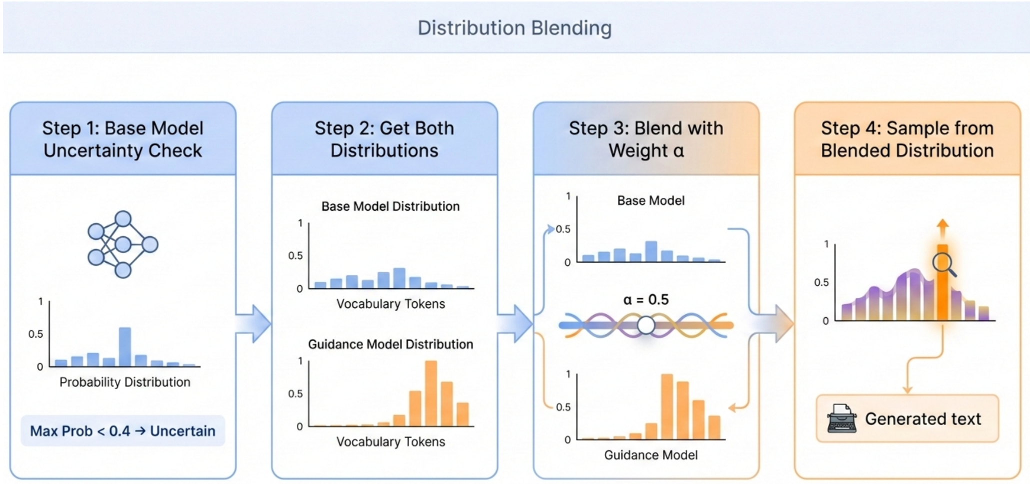



Figure 5: BlendIn for Quality-Aware Inference-Time Alignment. At positions where the base model is uncertain (max probability $< \tau$, default 0.4), BlendIn queries both base and guidance models for their full probability distributions, blends the distributions as $p_{\text{final}} = \alpha \cdot p_{\text{guidance}} + (1-\alpha) \cdot p_{\text{base}}$, and select the highest-probability token from the blended distribution (greedy decoding). Full distributions are substitutable with top k to save computation, where k is an arbitrary large value. Unlike binary decisions in existing methods that either accepts or rejects suggestion, such soft integration preserves beneficial guidance while reducing impact of unreliable suggestions, enabling quality-aware alignment.

Query both models for their next-token distributions:

$$P_b(w|x_{<t}) = \text{softmax}(\text{logits}_b(x_{<t})) \quad (1)$$

$$P_g(w|x_{<t}) = \text{softmax}(\text{logits}_g(x_{<t})) \quad (2)$$

For computational efficiency, the full distribution could be substituted by only taking the top-k in distribution, where k is an arbitrary large value.

**Step 2: Compute blend weight**

Determine blending weight $\alpha \in [0,1]$ based on both models' confidence and token-level agreement:

$$\alpha = \text{clip}\left(\frac{\hat{p}_g}{\hat{p}_b + \hat{p}_g} + \lambda \cdot P_b(t_g),\ 0,\ 1\right) \quad (3)$$

where $\hat{p}_b = \max_w P_b(w|x_{<t})$ and $\hat{p}_g = \max_w P_g(w|x_{<t})$ are the top-1 probabilities of the base and guidance models respectively, $t_g = \arg\max_w P_g(w|x_{<t})$ is the guidance model's top token, and $\lambda$=0.1 controls the agreement bonus weight. The confidence ratio $\hat{p}_g/(\hat{p}_b + \hat{p}_g)$ is high when guidance is confident and base is uncertain, naturally scaling intervention strength. The agreement bonus $\lambda \cdot P_b(t_g)$ increases guidance influence when its top token already has support in the base distribution, reducing the risk of distributional mismatch. With $\lambda$=0.1, the bonus contributes at most a 0.1 addition, keeping the confidence ratio as the primary driver. While this adaptive computation provides a principled default, $\alpha$ can also be manually tuned for task-specific optimal performance. Practical hyperparameter tuning guidance is provided in Appendix A.4.

**Step 3: Blend and Greedy Selection**

Create a hybrid distribution that combines both models' knowledge, weighted by the blending weight $\alpha$. Each token $w$ in the vocabulary receives a probability that is a weighted average of the two models' predictions.

$$P_{\text{blend}}(w|x_{<t}) = \alpha \cdot P_g(w|x_{<t}) + (1-\alpha) \cdot P_b(w|x_{<t}) \quad (4)$$

We then select the token with the highest probability in the blended distribution:

$$w_t = \arg\max_{w \in \mathcal{V}} P_{\text{blend}}(w|x_{<t}) \quad (5)$$

Algorithm 1 details a comprehensive overview of our algorithm. Sensitivity analysis for major hyperparameters is provided in Appendix A.2.

### 4.2 Why Distribution Blending Addresses Quality Blindness

Rather than binary decisions, distribution blending allows both models to contribute partial information. Even guidance models that make misaligned or harmful predictions may have useful signal that, when properly downweighted, improves upon base model alone. Base model may also provide valuable information due to its greater capability. Blend weight could also adapt to base model's uncertainty level, in which more uncertain positions receive stronger guidance, while confident positions rely more on base, naturally reduc-

**Algorithm 1** Quality-Aware Distribution Blending

**Input:** Base model $M_b$, guidance model $M_g$, prompt $x$
**Input:** Threshold $\tau$, blend weight $\alpha$ (see Eq. 3 for adaptive default), max tokens $T$
**Output:** Sequence $y$
$y \leftarrow$ empty, $t \leftarrow 0$
**while** $t < T$ **do**
  Query $M_b$ for token probabilities $P_b$
  Compute $u \leftarrow$ maximum probability in $P_b$
  **if** $u < \tau$ **then**
    Query $M_g$ for token probabilities $P_g$
    For each token $w$: $P(w) \leftarrow \alpha P_g(w) + (1-\alpha)P_b(w)$
    Select token $w_t$ with highest $P(w_t)$
  **else**
    Select token $w_t$ with highest $P_b(w_t)$
  **end if**
  Append $w_t$ to $y$
  $t \leftarrow t + 1$
**end while**
**return** $y$

ing intervention when base has strong preferences. When guidance is misaligned or harmful, its contribution is downweighted to avoid catastrophic token selection, ensuring a graceful fallback.

Cross-family blending operates on the shared tokens between models. We verified empirically that average overlap is >≈50% across main datasets and representative model pairs, sufficient for meaningful distribution blending. Tokens are weighted by their respective model's contribution (α or 1-α) then renormalized, ensuring coherent probability integration without requiring tokenizer alignment.

BlendIn also differs structurally from a conceptually-similar paradigm named confidence-based ensembling (Lakshminarayanan et al., 2017). Confidence-based ensembling addresses the question of which model to trust more when aggregating predictions across multiple models. Confidence serves as a weight to resolve disagreement between models while maximizing prediction accuracy. Meanwhile, no model has directional authority over another. BlendIn addresses a different problem: given a capable but unaligned base model and an aligned guidance model, how much alignment pressure to apply without destroying the base model's capabilities? The blending is inherently directional, as we are pushing base distribution towards an aligned target. Here, confidence serves a different purpose from confidence-based ensembling: it regulates the degree of intervention on a per-sample basis to prevent capability degradation. This distinction matters because the two settings have different failure modes. In confidence-based ensembling, there is no directional pressure and models are aggregated without target distribution. The intervention paradox therefore cannot arise: it is a failure mode unique to alignment interventions, where pushing too hard toward a target distribution degrades model capability. This further highlights the novelty of our work.

## 5 Experiments

We evaluate nine recently-released models of different sizes within three families: Llama (Grattafiori et al., 2024), Gemma (Team et al., 2025, 2024) and Qwen (Yang et al., 2025). They are tested in pairs both within and across family. We test on benchmarks like GSM8K (Cobbe et al., 2021), TruthfulQA (Lin et al., 2022) and XSTest (Röttger et al., 2024) where the intervention paradox is statistically significant (Figure 3). For reasoning tasks, GSM8K contains grade-school math word problems and TruthfulQA measures factual accuracy. For safety tasks, XSTest consists of adversarial safety prompts. We report results on a fixed random subset of 100 samples per benchmark to balance computational cost with statistical reliability. We also use greedy decoding (temperature=0.0), which produces deterministic outputs for reproducibility.

We use the default $\tau = 0.4$ as the threshold for base model uncertainty, triggering guidance when $\max_w P_b(w|x_{<t}) < 0.4$. Other hyperparameters are kept default.

We measure task performance as accuracy (percentage of correct answers) and also report intervention rate (percentage of tokens where guidance occurs) of baseline.

For BlendIn, we obtain top-100 token probabilities from both models and blend them. We evaluate on representative cross-family pairs spanning the performance spectrum: high-intervention poor-performing cases (Qwen-to-Llama/Gemma) and low-intervention well-performing cases (Gemma-Llama pairs). We also include well-performing within-family pairs like Gemma and Llama.

We focus on the practical setting where small guidance models guide larger base models to min-

| | GSM8K | | | | | | TruthfulQA | | | | | | XSTest | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Pair | Base | Guid. | Alig. | NUDG. | Ours | Int.% | Base | Guid. | Alig. | NUDG. | Ours | Int.% | Base | Guid. | Alig. | NUDG. | Ours | Int.% |
| *Cross-Family Pairs* | | | | | | | | | | | | | | | | | | |
| Q→L | 0.11 | 0.53 | 0.87 | 0.27 | 0.31(+15%) | 22.2 | 0.58 | 0.48 | 0.67 | 0.48 | 0.50(+4%) | 31.9 | 0.00 | 0.60 | 0.15 | 0.03 | 0.04(+33%) | 33.1 |
| Q→G | 0.32 | 0.53 | 0.86 | 0.54 | 0.56(+4%) | 23.1 | 0.40 | 0.48 | 0.70 | 0.40 | 0.42(+5%) | 31.5 | 0.01 | 0.60 | 0.14 | 0.02 | 0.02 | 34.3 |
| G→L | 0.11 | 0.45 | 0.87 | 0.59 | 0.60(+2%) | 14.8 | 0.58 | 0.46 | 0.67 | 0.45 | 0.50(+11%) | 19.2 | 0.00 | 0.96 | 0.15 | 0.08 | 0.08 | 27.5 |
| L→G | 0.32 | 0.45 | 0.86 | 0.67 | 0.67 | 9.2 | 0.40 | 0.50 | 0.70 | 0.47 | 0.49(+4%) | 17.2 | 0.01 | 0.94 | 0.14 | 0.10 | 0.15(+50%) | 20.3 |
| *Within-Family Pairs* | | | | | | | | | | | | | | | | | | |
| L→L | 0.11 | 0.45 | 0.87 | 0.58 | 0.58 | 14.4 | 0.58 | 0.50 | 0.67 | 0.51 | 0.51 | 20.0 | 0.00 | 0.94 | 0.15 | 0.12 | 0.12 | 22.2 |
| G→G | 0.32 | 0.45 | 0.86 | 0.66 | 0.66 | 10.0 | 0.40 | 0.46 | 0.70 | 0.43 | 0.43 | 20.9 | 0.01 | 0.96 | 0.14 | 0.10 | 0.14(+40%) | 28.0 |

Table 1: Performance of unaligned base model (Base), standalone guidance model (Guid.), base model aligned by training (Alig.), NUDGING baseline (NUDG.), and our method (Ours). We also show NUDGING intervention rates (Int.%). Abbreviations: Q=Qwen, L=Llama, G=Gemma. Percentages show improvement over NUDGING. Guidance models mostly outperform unaligned base models, demonstrating their capability to guide the base models. Even when base models outperform guidance models in some scenarios of TruthfulQA, our method has already demonstrated improvements compared with unaligned base models, indicating successful alignment. Overall, our method achieves improvements on high-intervention pairs while maintaining performance on low-intervention pairs.

imize computational costs. Guidance models include: Llama-3.2-1B-Instruct, Gemma-3-1b-it, Qwen3-1.7B. Base models include: Llama-3.1-8B, Gemma-2-9b, Qwen3-8B-Base.

We compare against the following baselines:

**Base Model:** Base model generates alone without guidance.

**Guidance Model:** Guidance model generates alone to prove qualification for guidance.

**Aligned base model**: Base model aligned through finetuning, establishing the upper bound for alignment effectiveness.

**NUDGING** (Fei et al., 2025): At uncertain positions (max prob $< \tau$), base model accepts guidance model's suggestions.

**Intervention Capping:** Naive solution to address intervention paradox by rejecting all guidance once intervention rate exceeds threshold (15%). We include this to demonstrate that simple quantity-based approaches fail.

Table 1 presents our main results. BlendIn achieves consistent improvements on high-intervention pairs. Qwen-guided pairs, which exhibit consistently high intervention rates across all tasks (22-34%), show the most substantial improvements. On Qwen→Llama, we achieve +15% on GSM8K (0.27→0.31), +4% on TruthfulQA (0.48→0.50), and +33% on XSTest (0.03→0.04). Qwen→Gemma shows similar patterns with +4-5% improvements on GSM8K and TruthfulQA. These results validate that our method addresses quality failures most prevalent in high-intervention scenarios, where guidance sources intervene frequently and provide unreliable suggestions.

Improvements on other cross-family pairs are task-dependent. Gemma→Llama and Llama→Gemma demonstrate various intervention rates (9-28%) and alignment successes (+2-50%) depending on task difficulty. Notably, Gemma→Llama achieves +11% on TruthfulQA despite only 19% intervention rate, while Llama→Gemma shows +50% on XSTest at 20% intervention. This variability suggests that intervention rate alone does not determine improvement potential. Improvements also vary by task because guidance quality is task-dependent: Gemma provides better guidance on TruthfulQA while Llama provides better guidance on XSTest, and our method leverages this selectively.

BlendIn also preserved performance on low-intervention pairs. Within-family pairs (Llama→Llama, Gemma→Gemma) maintain baseline performance across most configurations, with intervention rates ranging 10-28%. The one notable exception is Gemma→Gemma on XSTest (+40%, 0.10→0.14), where high intervention (28%) on this safety task benefits from BlendIn. Critically, we observe no catastrophic degradation: even when our method provides no improvement, performance remains stable, demonstrating robustness across the intervention spectrum.

We further validate on full test sets with three representative model pairs spanning the intervention spectrum: Qwen→Llama and Qwen→Gemma (high intervention rate, poor baseline performance) and Llama→Gemma (low intervention, good baseline performance). As shown in Table 2, while individual improvements are modest (+0–4%) and do not consistently

| Dataset | Pair | Int.% | NUDGING | Ours |
|---|---|---|---|---|
| GSM8K | Q→L | 22.4 | $0.29_{\pm.024}$ | $0.29_{\pm.024}$ |
| | L→G | 10.0 | $0.59_{\pm.027}$ | $0.59_{\pm.027}$ |
| | Q→G | 23.5 | $0.50_{\pm.027}$ | $0.50_{\pm.027}$ |
| TruthfulQA | Q→L | 31.5 | $0.46_{\pm.034}$ | $0.47_{\pm.034}$ |
| | L→G | 16.6 | $0.48_{\pm.034}$ | $0.48_{\pm.034}$ |
| | Q→G | 30.9 | $0.44_{\pm.034}$ | $0.44_{\pm.034}$ |
| XSTest | Q→L | 35.3 | $0.52_{\pm.062}$ | $0.54_{\pm.062}$ |
| | L→G | 22.9 | $0.90_{\pm.037}$ | $0.90_{\pm.037}$ |
| | Q→G | 35.4 | $0.47_{\pm.062}$ | $0.48_{\pm.062}$ |

Table 2: Full test set evaluation with 95% confidence intervals. While individual improvements are modest (+0–4%) and do not consistently reach statistical significance, the pattern is consistent: our method shows improvements on high-intervention scenarios while preserving performance in low-intervention scenarios, validating the effectiveness of our approach. The modest magnitude also underscores our recommendation for task-specific $\alpha$ optimization.

| Method | Q→L (22%) | Q→G (23%) | G→L (15%) | L→G (9%) |
|---|---|---|---|---|
| NUDGING | 0.27 | 0.54 | 0.59 | 0.67 |
| Cap at 15% | 0.25 | 0.46 | 0.55 | 0.58 |
| Ours | **0.31** | **0.56** | **0.60** | **0.67** |

Table 3: Intervention rate capping degrades performance across all pairs on GSM8K by discarding guidance indiscriminately. Numbers in parentheses show baseline intervention rates.

reach statistical significance, the pattern is consistent: our method shows improvements on high-intervention scenarios (>31%) while preserving performance in low-intervention scenarios. Results can be further optimized by task-specific $\alpha$ tuning. This task-dependent pattern also aligns with intuition: large base models retain strong reasoning capability despite lacking alignment, requiring less guidance on GSM8K. However, factual objectivity (TruthfulQA) and safety (XSTest) are precisely where alignment deficits manifest and intervention provides most benefit. Note that these results are under 1B guidance models which are limited in capability— we reasonably expect greater improvement if using larger models.

Overall, our approach achieves improvements in high-intervention scenarios while maintaining good performance in low-intervention cases where guidance quality is already adequate. This shows that BlendIn addresses the fundamental quality assessment gap in inference-time alignment. While inference-time methods cannot fully replace fine-tuning alignment yet in effectiveness, they provide significant value where retraining is prohibitive.

Table 3 demonstrates that intervention rate capping, which is a naive way to address the intervention paradox by limiting guidance acceptance to 15%, consistently degrades performance. This approach fails because it removes all guidance after reaching the threshold, discarding both beneficial and harmful interventions. The degradation affects all pairs regardless of baseline intervention rate: high-intervention pairs (Qwen→Gemma: 0.54→0.46, -15%), moderate pairs (Gemma→Llama: 0.59→0.55, -7%), and even low-intervention pairs already below the 15% threshold (Llama→Gemma: 0.67→0.58, -13%). This validates the necessity of selectively filtering harmful interventions while preserving beneficial ones instead of hard-limiting intervention frequency. We further evaluate discrete filtering alternatives in Appendix A.3, which confirm that acceptance-rule-based approaches are insufficient.

## 6 Conclusion

Inference-time alignment methods could transfer alignment properties from aligned LLMs to unaligned LLMs without retraining. Through comprehensive analysis across nine models and six benchmarks, we make novel contributions toward systematic characterization of quality failures in inference-time alignment. We identify the intervention paradox: excessive intervention rates correlate with worse performance, contradicting the assumption that all guidance improves alignment. This paradox reveals that existing methods suffer from quality blindness as they lack guidance quality assessment mechanisms to distinguish beneficial guidance from harmful ones.

To address quality blindness, we propose BlendIn, which softly integrates both models' probability distributions at difficult positions rather than making binary accept-or-reject decisions. This quality-aware approach achieves consistent improvements on high-intervention pairs while maintaining performance on low-intervention pairs. Overall, our work establishes diagnostic signals for rapid failure detection and demonstrates that quality-aware integration enables more robust cross-model guidance. This foundation opens pathways for developing better inference-time methods that reliably improve model performance across diverse model combinations.

## Limitations

Our work focuses on characterizing and mitigating quality failures but does not provide predictive models for determining compatibility before testing. The intervention rate serves as a rapid diagnostic signal measurable in a small subset, but deployment still requires validating each model pair empirically. Additionally, while BlendIn outperforms baselines, improvements on some pairs remain modest, suggesting room for further optimization through hyperparameter tuning, adaptive blending strategies or representation-level compatibility assessment.

## Acknowledgments

This research is supported by MOE Tier 1 grant RG16/22.

## References

Charlie Chen, Sebastian Borgeaud, Geoffrey Irving, Jean-Baptiste Lespiau, Laurent Sifre, and John Jumper. 2023. Accelerating large language model decoding with speculative sampling. *Preprint*, arXiv:2302.01318.

Peter Clark, Isaac Cowhey, Oren Etzioni, Tushar Khot, Ashish Sabharwal, Carissa Schoenick, and Oyvind Tafjord. 2018. Think you have solved question answering? try arc, the ai2 reasoning challenge. *arXiv:1803.05457v1*.

Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, Christopher Hesse, and John Schulman. 2021. Training verifiers to solve math word problems. *arXiv preprint arXiv:2110.14168*.

Yu Fei, Yasaman Razeghi, and Sameer Singh. 2025. Nudging: Inference-time alignment of LLMs via guided decoding. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 12702–12739, Vienna, Austria. Association for Computational Linguistics.

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

Dan Hendrycks, Collin Burns, Steven Basart, Andrew Critch, Jerry Li, Dawn Song, and Jacob Steinhardt. 2021a. Aligning ai with shared human values. *Proceedings of the International Conference on Learning Representations (ICLR)*.

Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. 2021b. Measuring massive multitask language understanding. *Proceedings of the International Conference on Learning Representations (ICLR)*.

Balaji Lakshminarayanan, Alexander Pritzel, and Charles Blundell. 2017. Simple and scalable predictive uncertainty estimation using deep ensembles. In *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc.

Yaniv Leviathan, Matan Kalman, and Yossi Matias. 2023. Fast inference from transformers via speculative decoding. In *Proceedings of the 40th International Conference on Machine Learning*, ICML'23. JMLR.org.

Bill Yuchen Lin, Abhilasha Ravichander, Ximing Lu, Nouha Dziri, Melanie Sclar, Khyathi Chandu, Chandra Bhagavatula, and Yejin Choi. 2023. The unlocking spell on base llms: Rethinking alignment via in-context learning. *ArXiv preprint*.

Stephanie Lin, Jacob Hilton, and Owain Evans. 2022. TruthfulQA: Measuring how models mimic human falsehoods. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 3214–3252, Dublin, Ireland. Association for Computational Linguistics.

Zhixuan Liu, Zhanhui Zhou, Yuanfu Wang, Chao Yang, and Yu Qiao. 2024. Inference-time language model alignment via integrated value guidance. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pages 4181–4195, Miami, Florida, USA. Association for Computational Linguistics.

Long Ouyang, Jeff Wu, Xu Jiang, Diogo Almeida, Carroll L. Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul Christiano, Jan Leike, and Ryan Lowe. 2022. Training language models to follow instructions with human feedback. In *Proceedings of the 36th International Conference on Neural Information Processing Systems*, NIPS '22, Red Hook, NY, USA. Curran Associates Inc.

Rafael Rafailov, Archit Sharma, Eric Mitchell, Stefano Ermon, Christopher D. Manning, and Chelsea Finn. 2023. Direct preference optimization: your language model is secretly a reward model. In *Proceedings of the 37th International Conference on Neural Information Processing Systems*, NIPS '23, Red Hook, NY, USA. Curran Associates Inc.

Paul Röttger, Hannah Kirk, Bertie Vidgen, Giuseppe Attanasio, Federico Bianchi, and Dirk Hovy. 2024. XSTest: A test suite for identifying exaggerated safety behaviours in large language models. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational*

*Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 5377–5400, Mexico City, Mexico. Association for Computational Linguistics.

Gemma Team, Aishwarya Kamath, Johan Ferret, Shreya Pathak, Nino Vieillard, Ramona Merhej, Sarah Perrin, Tatiana Matejovicova, Alexandre Ramé, Morgane Rivière, Louis Rouillard, Thomas Mesnard, Geoffrey Cideron, Jean bastien Grill, Sabela Ramos, Edouard Yvinec, Michelle Casbon, Etienne Pot, Ivo Penchev, and 197 others. 2025. Gemma 3 technical report. *Preprint*, arXiv:2503.19786.

Gemma Team, Morgane Riviere, Shreya Pathak, Pier Giuseppe Sessa, Cassidy Hardin, Surya Bhupatiraju, Léonard Hussenot, Thomas Mesnard, Bobak Shahriari, Alexandre Ramé, Johan Ferret, Peter Liu, Pouya Tafti, Abe Friesen, Michelle Casbon, Sabela Ramos, Ravin Kumar, Charline Le Lan, Sammy Jerome, and 179 others. 2024. Gemma 2: Improving open language models at a practical size. *Preprint*, arXiv:2408.00118.

Pengyu Wang, Dong Zhang, Linyang Li, Chenkun Tan, Xinghao Wang, Mozhi Zhang, Ke Ren, Botian Jiang, and Xipeng Qiu. 2024. InferAligner: Inference-time alignment for harmlessness through cross-model guidance. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 10460–10479, Miami, Florida, USA. Association for Computational Linguistics.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, Chujie Zheng, Dayiheng Liu, Fan Zhou, Fei Huang, Feng Hu, Hao Ge, Haoran Wei, Huan Lin, Jialong Tang, and 41 others. 2025. Qwen3 technical report. *Preprint*, arXiv:2505.09388.

# A Appendix

## A.1 Vocabulary Overlap under Top-*p* Coverage

We also computed overlap using 90% probability mass coverage for further validation. Results in Table 4 confirm the null finding across all six benchmarks (|r| ≤ 0.35, p > 0.35), with top-p(0.9) yielding smaller or equivalent overlap sets compared with top-50 in 8 of 9 model pairs (three Within-family + six Cross-family). The stricter metric does not reveal any hidden correlation.

These results exactly explain why we originally didn't consider probability mass coverage: top-k=50 already provides the most favorable conditions for detecting a correlation. Top-k=50 is deliberately lenient: it includes many low-probability tokens, maximizing the overlap set size and thus maximizing statistical power to detect any relationship between vocabulary overlap and generation quality. If guidance tokens appearing in this large, inclusive set show no predictive power (r≤0.25, p>0.05), then a stricter 90% probability mass criterion that selects only the model's highest-confidence tokens and yields smaller overlap sets would have less power to detect such a relationship. The null result under our favorable conditions is therefore informative.

| Benchmark | Top-$k$ ($k$=50) $r$ | $p$ | Top-$p$ ($p$=0.9) $r$ | $p$ |
|---|---|---|---|---|
| GSM8K | −0.072 | 0.853 | −0.170 | 0.662 |
| MMLU | −0.253 | 0.511 | −0.351 | 0.355 |
| TruthfulQA | −0.186 | 0.632 | −0.336 | 0.376 |
| ARC-Challenge | −0.161 | 0.679 | −0.233 | 0.547 |
| XSTest | +0.047 | 0.905 | −0.073 | 0.851 |
| JustEval-Safe | −0.244 | 0.527 | −0.326 | 0.392 |

Table 4: Pearson correlation between vocabulary overlap and performance under top-$k$ ($k$=50) and top-$p$ ($p$=0.9) coverage metrics. Neither metric reveals a statistically significant correlation ($|r| \leq 0.35$, $p > 0.35$ in all cases). Top-$p$(0.9) yields smaller or equivalent overlap sets in 8 of 9 model pairs yet does not uncover any hidden relationship, confirming that the null finding is robust to overlap metric choice.

| $\tau$ | GSM8K | TruthfulQA | XSTest |
|---|---|---|---|
| 0.3 | 0.25 | 0.42 | 0.50 |
| 0.4 | 0.12 | 0.30 | 0.52 |
| 0.5 | 0.32 | 0.44 | 0.55 |

| $\alpha$ | GSM8K | TruthfulQA | XSTest |
|---|---|---|---|
| 0.3 | 0.23 | 0.47 | 0.45 |
| 0.5 | 0.28 | 0.42 | 0.52 |
| 0.7 | 0.12 | 0.42 | 0.54 |
| Auto | 0.12 | 0.30 | 0.52 |

Table 5: Sensitivity of $\tau$ with $\alpha$=Auto (*left*) and $\alpha$ with $\tau$=0.4 (*right*) on Qwen→Llama (full test set). Both hyperparameters show task-dependent effects, empirically validating the recommendation of task-specific manual tuning. Default settings ($\tau$=0.4, $\alpha$=Auto) provide reasonable bottomline performances across tasks without tuning.

## A.2 Hyperparameter Sensitivity

We also conducted sensitivity analysis for τ (uncertainty threshold, established and validated in prior work on inference-time alignment (Fei et al., 2025)) and α (blend weight, one of our contributions) on representative challenging model pair Qwen->Llama and full datasets. As shown in table 5, both τ and α show task-dependent effects. This empirically validates our stated recommen-

| Dataset | GSM8K | | | TruthfulQA | | | XSTest | | |
|---|---|---|---|---|---|---|---|---|---|
| Method | LL | GG | QL | LL | GG | QL | LL | GG | QL |
| NUDG. | 0.58 | 0.66 | 0.27 | 0.51 | 0.43 | 0.43 | 0.12 | 0.10 | 0.03 |
| AF CC | 0.45 | 0.41 | 0.28 | 0.36 | 0.28 | 0.22 | 0.10 | 0.12 | 0.03 |
| Ours | **0.58** | **0.66** | **0.31** | **0.51** | **0.43** | **0.50** | **0.12** | **0.10** | **0.04** |

Table 6: Comparison against the discrete filtering baseline combining Agreement Filter and Confidence Competition. Pair abbreviations: LL=Llama→Llama, GG=Gemma→Gemma, QL=Qwen→Llama. Agreement Filter (AF) accepts guidance only when the guidance token appears in the base model's top-$k$; Confidence Competition (CC) accepts guidance only when guidance model top-1 probability exceeds the base model's. The baseline shows inconsistent and mostly worse results than NUDGING, confirming that soft quality-aware integration is necessary. Our approach maintains or improves over NUDGING across all configurations.

dation that α should be manually tuned for task-specific optimal performance. Similarly, τ controls intervention frequency and benefits from task-specific tuning. The observed variance reflects that different tasks benefit from different hyperparameter settings.

### A.3 Discrete Filtering Baselines

We additionally evaluated the discrete filtering baseline combining Agreement Filter and Confidence Competition. Agreement Filter accepts guidance only when the guidance token appears in the base model's top-$k$ candidates; Confidence Competition accepts guidance only when the guidance model's top-1 probability exceeds the base model. As shown in Table 6, the baseline performs inconsistently and mostly worse than main baseline Nudging, confirming that the intervention paradox cannot be addressed by discrete acceptance rules and motivating our soft blending approach.

### A.4 Practical Hyperparameter Tuning Guide

For deployment, we recommend the following lightweight protocol:

**Default settings:** Use $\tau$=0.4 (base uncertainty threshold) and $\alpha$=auto (adaptive blending). These provide reasonable performance without tuning.

**When to tune:** If initial results show intervention rate exceeding 20% with unsatisfactory performance, optimize hyperparameters on a small validation set (100 samples).

**Tuning protocol:**

1. Fix $\tau$=0.4, test $\alpha \in \{0.3, 0.5, 0.7\}$
2. Optionally refine $\alpha$ by ±0.1 around best value so far for further improvement on performance
3. Select best $\alpha$, then test $\tau \in \{0.3, 0.4, 0.5\}$

Each configuration requires around 5 minutes on 100 samples, making basic protocols (steps 1 and 3, 6 trials in total) feasible within 30 minutes. Coarse-grained tuning (steps 1-3) captures most potential gains, while finer adjustments (e.g., α ±0.05) yield diminishing returns relative to validation cost.