Hierarchical Alignment: Surgical Fine-Tuning via Functional Layer Specialization in Large Language Models

Anonymous ACL submission

Abstract

Existing alignment techniques for Large Language Models (LLMs), such as Direct Preference Optimization (DPO), typically treat the model as a monolithic entity, applying uniform optimization pressure across all layers. This approach overlooks the functional specialization within the Transformer architecture, where different layers are known to handle distinct tasks from syntax to abstract reasoning. In this paper, we challenge this one-size-fits-all paradigm by introducing Hierarchical Alignment, a novel method that applies targeted DPO to distinct functional blocks of a model's layers: local (syntax), intermediate (logic), and global (factuality). Through a series of controlled experiments on state-of-the-art models like Llama-3.1-8B and Qwen1.5-7B using LoRA for surgical fine-tuning, our results, evaluated by a powerful LLM-as-Judge, demonstrate significant and predictable improvements. Specifically, aligning the local layers (Local-Align) enhances grammatical fluency. More importantly, aligning the global layers (Global-Align) not only improves factual consistency as hypothesized but also proves to be the most effective strategy for enhancing logical coherence, outperforming all baselines. Critically, all hierarchical strategies successfully avoid the "alignment tax" observed in standard DPO, where gains in fluency come at the cost of degraded logical reasoning. These findings establish a more resource-efficient, controllable, and interpretable path for model alignment, highlighting the immense potential of shifting from monolithic optimization to structure-aware surgical fine-tuning to build more advanced and reliable LLMs.

1 Introduction

Large Language Models (LLMs) such as LLaMA (Touvron et al., 2023) and the GPT series have achieved remarkable progress in text generation, knowledge retrieval, and downstream instruction following. Yet ensuring their alignment with human preferences and societal norms remains a critical and unresolved challenge in AI development. Mainstream alignment techniques—including Reinforcement Learning from Human Feedback (RLHF) and its scalable variant Direct Preference Optimization (DPO) (Rafailov et al., 2024)—typically apply uniform loss functions across the entire model, ignoring the functional diversity of different layers within Transformer architectures.

A growing body of interpretability research has shown that Transformer models exhibit structured functional specialization: lower layers encode syntactic patterns, middle layers support semantic coherence, and upper layers are responsible for global reasoning and goal-directed behavior (van Aken et al., 2019). This hierarchy is not incidental, but a robust outcome of large-scale pretraining dynamics. Uniform alignment methods that disregard this structure may inadvertently compromise one behavioral dimension while improving another—leading to phenomena such as the "alignment tax" or regressions in logical reasoning despite gains in fluency. We argue that effective alignment requires interventions that are aware of and adaptive to the model's internal functional stratification.

To address this, we propose a hierarchical alignment framework that partitions the model into three functional blocks—local (syntax and grammar), intermediate (logic and discourse), and global (factuality and instruction adherence)—and selectively fine-tunes each using LoRA-based adapters. Through controlled experiments on LLaMA-3.1-8B and Qwen1.5-7B with Deepseek-R1 evaluation, we demonstrate the benefits of targeted tuning. Local block alignment significantly improves fluency (+0.52 win rate), global block alignment yields the highest factual consistency (+0.07) and logical coherence (+0.10), and, intriguingly, also achieves the best overall syntactic performance (+0.63), suggest-

ing top-down synergy. In contrast, holistic DPO improves fluency (+0.62) but degrades logical reasoning (–0.12), highlighting the trade-offs of undifferentiated optimization.

This study shows that respecting the internal organization of LLMs enables better alignment performance with lower computational cost and greater behavioral control. More broadly, it represents a shift in alignment methodology—from coarse-grained global optimization to structure-aware precision tuning—paving the way for more principled and transparent AI systems.

2 Related Work

Effective alignment of large language models requires understanding not just *what* to optimize, but *where* in the model to apply that optimization. While recent work has made significant progress in alignment techniques and recognized the hierarchical nature of deep networks, the intersection of these two insights—leveraging internal functional structure for targeted alignment—remains largely unexplored. This section reviews three interconnected research threads: alignment methodologies, evidence for functional stratification in neural networks, and emerging approaches to structured model editing.

2.1 Alignment Methods: From Monolithic to Modular

Modern LLM alignment has progressed from supervised fine-tuning to reinforcement learning frameworks like RLHF (Ouyang et al., 2022) and its simplified variant DPO (Rafailov et al., 2024). While these methods differ in optimization strategy, they share a common paradigm: **applying uniform updates across all model parameters**. Recent work has explored hybrid approaches—Pant (2025) combine SFT and DPO for improved safety-helpfulness tradeoffs, while Wang et al. (2025a) propose GRAO to integrate multiple alignment objectives through weighted advantage estimation.

Despite these advances, the fundamental assumption persists: alignment is a whole-model operation. Even sophisticated techniques like CM-Align (Zhang et al., 2025b), which filters noisy preference pairs for multilingual consistency, still apply updates uniformly across layers. This overlooks a critical question: if different layers serve different functions, should they be aligned differently?

2.2 Functional Specialization in Neural Networks

Converging evidence across domains suggests deep networks naturally develop hierarchical functional organization. In language models, probing studies reveal a clear division of labor: lower layers encode syntactic and morphological features, while upper layers capture semantics and reasoning (van Aken et al., 2019). This stratification persists post-alignment, with instruction-tuned models showing preserved linguistic knowledge in early layers and task-specific processing in later layers (Nadipalli, 2025).

Similar patterns emerge beyond NLP. In vision models, Olson et al. (2025) show that DINOv2 representations implicitly encode hierarchical taxonomies, with layer depth corresponding to concept abstraction. For text-to-image generation, Zhang et al. (2025a) find DiT models process instances in early layers, backgrounds in middle layers, and attributes in late layers. Even in state-space models like Mamba, causal tracing localizes factual recall to mid-layers and output coherence to later stages (Sharma et al., 2024).

These findings suggest a general principle: **hierarchical organization is not an artifact but a fundamental property of deep learning systems**. The question is whether we can exploit this structure for more effective alignment.

2.3 Toward Structured Model Editing

A nascent line of work challenges whole-model tuning by advocating for targeted interventions. External modular approaches like ALIGNER (Ji et al., 2024) and MODULAR PLURALISM (Feng et al., 2024) achieve controllability through plug-and-play adapters. However, these methods add external components rather than leveraging internal structure.

More relevant are works that exploit layer-wise properties for specific tasks. In vision-language models, Wang et al. (2025b) identify attention heads responsible for different semantic roles (objects, attributes, relationships) and edit them selectively. In image generation, Zeng et al. (2025) create a two-tier architecture where high-level planning modules guide low-level generators. Even in hardware debugging, Yao et al. (2025) show that fragmenting modules into semantically coherent units improves repair precision.

While these works demonstrate the power of

structured intervention, they focus on task-specific editing or external modularity. **None systematically apply this principle to preference-based alignment by targeting the internal functional hierarchy of LLMs**.

2.4 Our Contribution

We bridge this gap by introducing **Hierarchical Alignment**, which directly maps alignment objectives to functionally specialized layer blocks. Unlike external modular systems that add components, we intervene within the model's existing hierarchy. Unlike general parameter-efficient methods like LoRA (Hu et al., 2021), we use layer selection to target specific capabilities. And unlike layer-wise analysis in vision or editing in multimodal models, we apply structural targeting to the core challenge of LLM behavioral alignment.

Our approach is grounded in two key insights from prior work: (1) alignment methods would benefit from finer-grained control (Pant, 2025; Wang et al., 2025a), and (2) deep networks exhibit consistent functional stratification (van Aken et al., 2019; Nadipalli, 2025). By synthesizing these insights, we demonstrate that *where* we align matters as much as *how* we align.

3 Methodology

This section operationalizes the Hierarchical Alignment framework. We first establish its theoretical underpinnings through formal definitions and core hypotheses. We then detail the implementation, specifying how Direct Preference Optimization (DPO) and Low-Rank Adaptation (LoRA) are employed for targeted updates. The section culminates in a precise algorithmic specification and a set of testable predictions that directly guide our experimental validation.

3.1 Theoretical Foundations

Our approach is built upon the principle that a Transformer's internal architecture is not monolithic but functionally stratified. We formalize this as follows.

Definition 1 (Functional Stratification)**.** *Let a Transformer model be a sequence of N layers, $\mathcal{T} = \{L_1, L_2, \ldots, L_N\}$. A **functional stratification** is a partition of these layers into K disjoint blocks, $\Pi = \{\mathcal{S}_1, \mathcal{S}_2, \ldots, \mathcal{S}_K\}$, where each block $\mathcal{S}_k = \{L_i : i \in I_k\}$ is hypothesized to perform a distinct functional role. The model's global computation F can thus be viewed as a composition of block-specific functions: $F(\mathbf{x}; \Theta) = f_K \circ f_{K-1} \circ \cdots \circ f_1(\mathbf{x})$.*

This framework rests on two foundational hypotheses derived from extensive interpretability research.

Hypothesis 1 (Functional Specialization)**.** *In a sufficiently pre-trained LLM, a natural functional stratification exists where blocks process information hierarchically. This hierarchy manifests as a progression from lower-level linguistic features (e.g., syntax) in initial blocks to higher-level semantic and reasoning capabilities (e.g., factuality, intent) in final blocks.*

Hypothesis 2 (Objective-Function Correspondence)**.** *For a given alignment objective a_m (e.g., improving factuality) with a corresponding loss ℓ_m, the loss gradient is predominantly concentrated within the parameter subspace Θ_k of the functionally corresponding block $\mathcal{S}_k$. Formally:*

$$\mathbb{E}_{\mathbf{x}\sim\mathcal{D}} \left\| \frac{\partial \ell_m(\mathbf{x})}{\partial \Theta_k} \right\| \gg \mathbb{E}_{\mathbf{x}\sim\mathcal{D}} \left\| \frac{\partial \ell_m(\mathbf{x})}{\partial \Theta_{k'}} \right\|, \quad \forall k' \neq k$$

This hypothesis provides the theoretical justification for targeted intervention, suggesting that surgical updates to a specific block will be maximally effective for its corresponding objective while minimizing collateral effects on others.

3.2 Implementation

We translate the theoretical framework into a concrete algorithm by specifying the block partitioning scheme, the alignment loss, and the mechanism for targeted parameter updates.

3.2.1 Functional Block Partitioning

Guided by Hypothesis 1, we instantiate the stratification Π with three functionally motivated blocks:

- **Local Block ($\mathcal{S}_{\text{local}}$):** The initial one-third of layers, responsible for syntax, grammar, and fluency.
- **Intermediate Block ($\mathcal{S}_{\text{mid}}$):** The middle one-third of layers, governing discourse coherence and local semantic consistency.
- **Global Block ($\mathcal{S}_{\text{global}}$):** The final one-third of layers, handling thematic relevance, instruction adherence, and high-level reasoning.

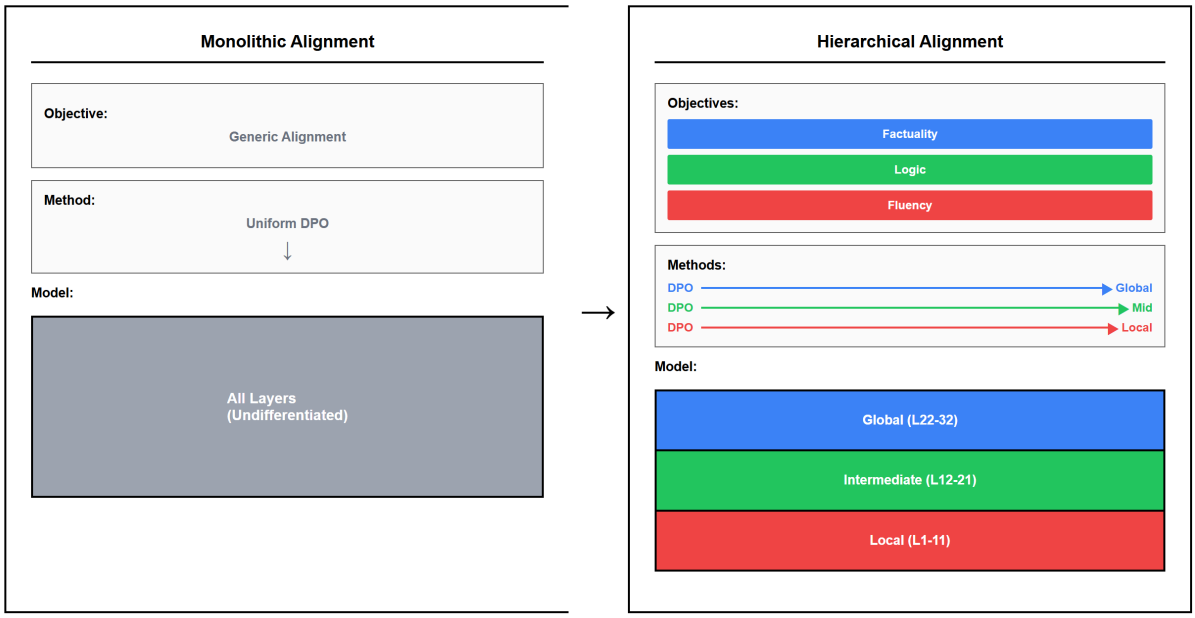

Figure 1: **Theoretical framework.** *Left: Monolithic Alignment* applies a uniform DPO update to all layers, treating the model as undifferentiated and risking an *alignment tax* (e.g., fluency improves while logic degrades). *Right: Hierarchical Alignment* decomposes objectives (grammar/fluency, coherence/logic, factuality/reasoning) and performs targeted optimization on functionally specialized blocks (local, intermediate, global), reducing interference and improving controllability.

To ensure model agnosticism and reproducibility, we employ a simple partitioning heuristic. Given N layers, the block sizes are determined systematically to distribute layers as evenly as possible. This heuristic provides a strong, non-arbitrary baseline for our experiments.

3.2.2 Alignment Objective: Direct Preference Optimization (DPO)

We adopt the DPO loss as our alignment objective. For a preference tuple (x, y_w, y_l) where response y_w is preferred over y_l for prompt x, the loss is defined as:

$$\mathcal{L}_{\text{DPO}} = -\,\mathbb{E}\left[\log\sigma\left(\beta\log\frac{\pi_\theta(y_w|x)}{\pi_{\text{ref}}(y_w|x)} - \beta\log\frac{\pi_\theta(y_l|x)}{\pi_{\text{ref}}(y_l|x)}\right)\right] \tag{1}$$

where π_θ is the policy model being optimized, π_{ref} is a frozen reference model, and β is a temperature parameter.

3.2.3 Targeted Updates via Low-Rank Adaptation (LoRA)

To enforce the principle of targeted intervention from Hypothesis 2, we require a mechanism that confines parameter updates to a specific block $\mathcal{S}_k$. We employ LoRA for this purpose, treating it as a **subspace selector**.

Specifically, we freeze the entire base model and inject trainable, low-rank matrices *exclusively* into the self-attention modules of the layers within the target block $\mathcal{S}_k$. This design choice is deliberate: self-attention is the primary mechanism for information integration and representation refinement within the Transformer architecture. By modifying it directly, we aim to precisely control *how* information is processed within a functional block, while preserving the vast world knowledge typically stored in the feed-forward network (FFN) parameters.

The optimization thus operates not on the full parameter space Θ, but only on the LoRA parameters $\Theta_{k,\text{LoRA}}$ associated with block $\mathcal{S}_k$. The update rule becomes:

$$\Theta_{k,\text{LoRA}}^{(t+1)} \leftarrow \Theta_{k,\text{LoRA}}^{(t)} - \eta\nabla_{\Theta_{k,\text{LoRA}}}\mathcal{L}_{\text{DPO}} \tag{2}$$

3.3 Algorithm and Testable Predictions

The complete Hierarchical Alignment procedure is summarized in Algorithm 1. Based on this methodology, we derive a set of clear, falsifiable predictions that will be empirically tested in Section **??**.

Testable Predictions.

Algorithm 1 Hierarchical Alignment

Input: Base model $F_{\Theta_{\text{base}}}$, preference data $\mathcal{D}$, target objective a_m.
Phase 1: Functional Mapping
Map objective a_m to a target block $\mathcal{S}_k$ per Hypothesis 2 (e.g., Factuality $\rightarrow \mathcal{S}_{\text{global}}$).
Identify layer indices I_k for block $\mathcal{S}_k$ using the partitioning heuristic.
Phase 2: Subspace Parameterization
Freeze base parameters Θ_{base}.
Inject trainable LoRA adapters (parameters $\Theta_{k,\text{LoRA}}$) into self-attention modules of layers $\{L_i\}_{i \in I_k}$.
Phase 3: Targeted Optimization
Initialize reference policy $\pi_{\text{ref}} \leftarrow \pi_{\Theta_{\text{base}}}$.
for each batch $(x, y_w, y_l) \sim \mathcal{D}$ **do**
  Compute gradient $\nabla_{\Theta_{k,\text{LoRA}}} \mathcal{L}_{\text{DPO}}$ using Eq. 1.
  Update $\Theta_{k,\text{LoRA}}$ per Eq. 2.
end for
Phase 4: Model Finalization
Merge trained LoRA weights into base model parameters.
Output: Hierarchically aligned model $F'_{\Theta_{\text{base}}+\Delta\Theta_{k,\text{LoRA}}}$.

(i) **`Local-Align` ($\mathcal{S}_{\text{local}}$):** Will yield significant and targeted improvements in grammatical fluency, with minimal impact on higher-level capabilities like logic and factuality.

(ii) **`Global-Align` ($\mathcal{S}_{\text{global}}$):** Is predicted to be the most effective strategy for enhancing factuality and logical coherence, aligning with its role in high-level reasoning.

(iii) **Interference Mitigation:** All hierarchical strategies are expected to outperform monolithic DPO by avoiding the "alignment tax"—i.e., the degradation of one capability while optimizing for another.

4 Experiment

This section empirically validates our Hierarchical Alignment framework through a series of controlled experiments. We test the central hypothesis that our targeted approach yields not only effective, but also predictable, improvements in model behavior.

Models and Data. We select the state-of-the-art **`Llama-3.1-8B-Instruct`** and **`Qwen1.5-7B-Chat`** as our base models. Alignment training is performed on the publicly available **`Anthropic/hh-rlhf`** preference dataset.

Comparison Strategies. The experiment centers on comparing five distinct alignment strategies, described in Table 1. These include a baseline model with no optimization (`Base Model`), a standard monolithic alignment approach (`Full-DPO`), and our three proposed Hierarchical Alignment strategies: `Local-Align`, `Mid-Align`, and `Global-Align`.

Evaluation Protocol. We employ an **LLM-as-Judge** paradigm for evaluation, using **`DeepSeek-R1`** to conduct pairwise comparisons. The evaluation dimensions (Grammar & Fluency, Coherence & Logic, Factuality, and Relevance & Instruction Following) are precisely designed to quantitatively validate our theoretical predictions. Our primary metric is the **Net Win Rate**, defined as (Win Rate - Loss Rate).

4.1 Results and Analysis

Our experimental results clearly and robustly confirm the core predictions of the Hierarchical Alignment framework. The heatmap in Figure 3 provides a high-level summary of the Net Win Rates for each hierarchical strategy against the `Full-DPO` baseline, while Figure 2 offers a detailed breakdown of the win-loss-tie distributions.

The Alignment Tax of Monolithic DPO. As a starting point, we examine the performance of the standard `Full-DPO` strategy against the `Base Model`. As shown in the first column of the heatmap, `Full-DPO` achieves a remarkable Net Win Rate of **+0.62** in *Grammar & Fluency*, demonstrating its powerful ability to enhance linguistic expression. However, this improvement comes at a cost: in the *Coherence & Logic* dimension, its Net Win Rate plummets to **-0.12**. This result is critical as it provides quantitative evidence of the "alignment tax," where monolithic optimization, in its pursuit of one objective (fluency), harms another core capability (logic). This provides a solid empirical motivation for our work.

Validating Specialization: Targeted Effects of Hierarchical Alignment. Next, we evaluate the performance of the three Hierarchical Alignment strategies against the `Full-DPO` baseline. The results precisely validate our theory.

Table 1: Description of the five model groups and their respective alignment strategies.

Group Name	Training Strategy	Description
`Base Model`	None	The original, pre-trained SFT model. Serves as the performance floor.
`Full-DPO`	Monolithic Alignment	LoRA adapters applied to all layers. Represents the standard DPO baseline.
`Local-Align`	Hierarchical Alignment	LoRA adapters applied only to the dynamically determined Local Block layers.
`Mid-Align`	Hierarchical Alignment	LoRA adapters applied only to the dynamically determined Intermediate Block layers.
`Global-Align`	Hierarchical Alignment	LoRA adapters applied only to the dynamically determined Global Block layers.

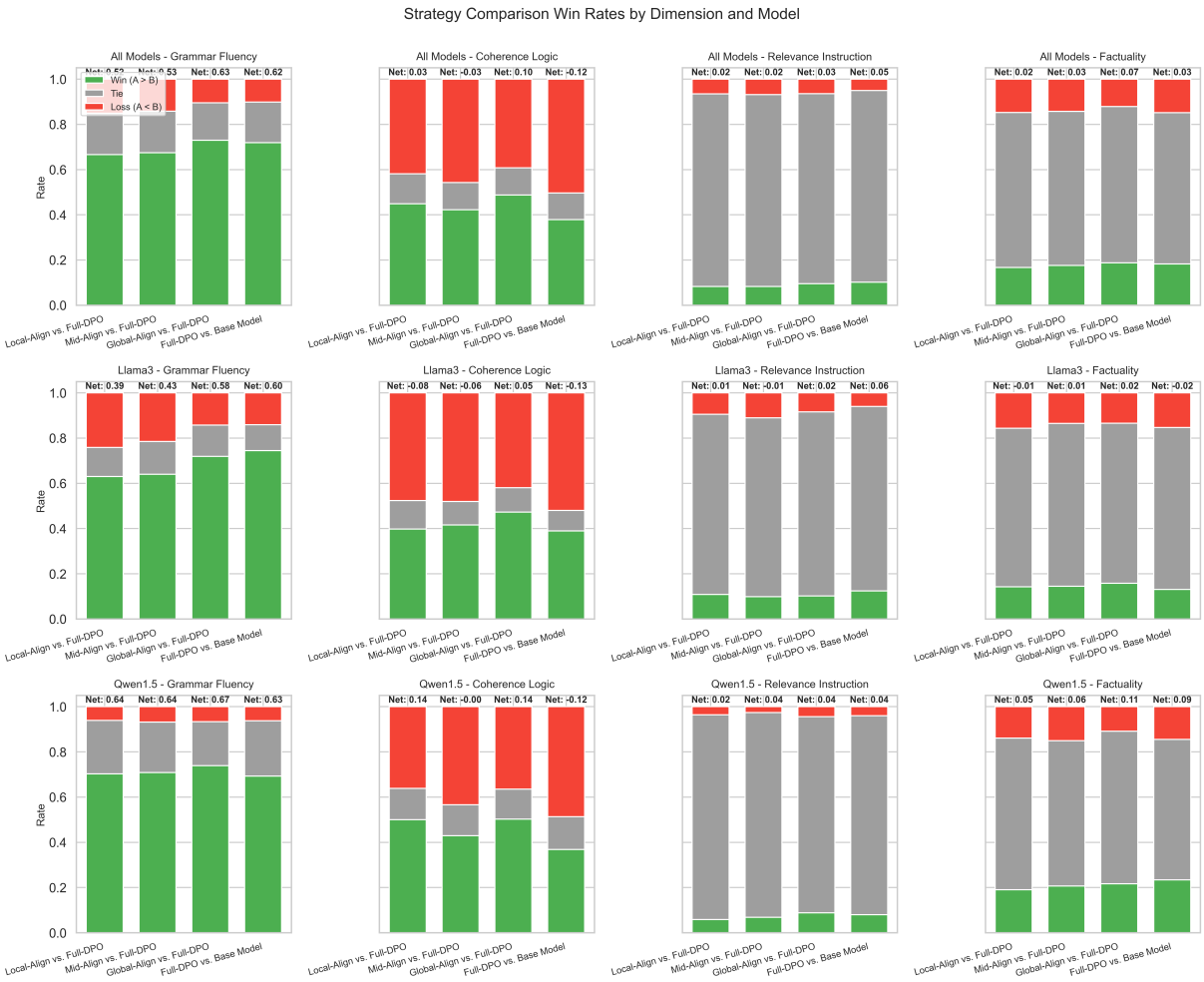

Figure 2: Win-Loss-Tie Distribution for Hierarchical vs. Monolithic Alignment. Each subplot displays a head-to-head comparison for a specific model and evaluation dimension. The stacked bars show the proportion of Wins (green), Ties (gray), and Losses (red) for each hierarchical strategy when compared against the Full-DPO or baseline. The "Net" value annotated above each bar represents the Net Win Rate (Win Rate - Loss Rate), providing a summary of the overall performance.

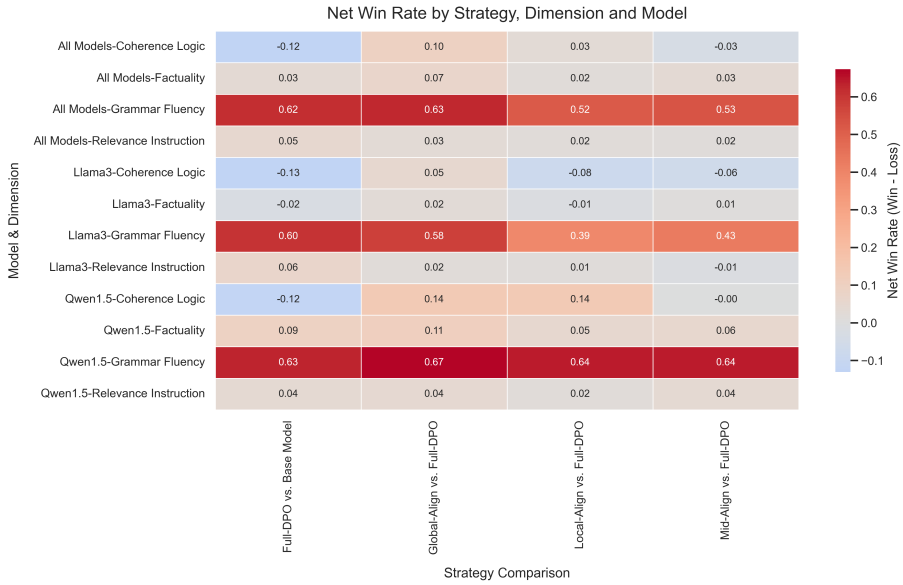

Figure 3: Net Win Rate of Hierarchical Strategies vs. Monolithic DPO Baseline. The heatmap displays the Net Win Rate (Win Rate - Loss Rate) for each hierarchical alignment strategy when compared against the Full-DPO baseline. Positive values (red) indicate that the hierarchical strategy outperformed the baseline, while negative values (blue) indicate underperformance.

- **`Local-Align` Focuses on Syntax:** Our theory predicts that targeting the lower layers should primarily affect low-level functions. The results confirm this perfectly. `Local-Align` achieves a strong Net Win Rate of **+0.52** in *Grammar & Fluency*, with robust performance on both Llama3 (+0.39) and Qwen1.5 (+0.64). Concurrently, its impact on high-level functions like coherence (+0.03) and factuality (+0.02) is negligible. This clearly demonstrates the targeted nature of our approach, enhancing a specific capability without significant destructive interference.

- **`Global-Align` Masters High-Level Reasoning:** The theory predicts that the upper layers are responsible for high-level reasoning. The results again provide strong support. `Global-Align` is the top-performing strategy in both *Coherence & Logic* (**+0.10**) and *Factuality* (**+0.07**). This indicates that complex reasoning and fact-checking are indeed rooted in the model's upper layers. This finding not only validates our theory but also points toward a path for improving a model's "intelligence," not just its "eloquence."

- **An Unexpected Finding on `Mid-Align`:** Contrary to our initial hypothesis, `Mid-Align` failed to show an advantage in *Coherence & Logic*, with a slightly negative Net Win Rate (-0.03). This is an equally important negative result. It suggests that "logical coherence" is not solely localized in the middle layers but is likely a more distributed function that depends heavily on top-level integration. The superior performance of `Global-Align` in this dimension further supports this interpretation.

4.2 Comparative Analysis and Key Observations

Table 2: Net Win Rate of Hierarchical Strategies vs. the Monolithic Baseline (`Full-DPO`). Best performance in each column is in bold.

Strategy	Grammar	Logic	Factuality
`Global-Align`	**+0.63**	**+0.10**	**+0.07**
`Local-Align`	+0.52	+0.03	+0.02
`Mid-Align`	+0.53	-0.03	+0.03

A quantitative comparison of our "scalpel" (Hierarchical Alignment) versus the "sledgehammer" (monolithic DPO) reveals the clear superiority of the targeted approach. As shown in Table 2, `Global-Align` emerges as the optimal strategy, not only dominating in its target high-level dimensions like logic (+0.10) and factuality (+0.07) but also

surpassing all other methods in grammar and fluency (+0.63). This suggests that higher-quality reasoning may naturally lead to more fluent expression. Crucially, both `Global-Align` and `Local-Align` successfully avoid the degradation in logical ability seen with monolithic DPO, demonstrating a healthier pattern of performance improvement and mitigating the "alignment tax." Furthermore, we observed a notable model-specific sensitivity; for instance, the Qwen1.5 model was more responsive to hierarchical interventions than Llama-3.1, with the Net Win Rate for `Global-Align` in logic being nearly three times higher (+0.14 vs. +0.05). This suggests that a model's amenability to functional specialization may be influenced by its architecture or pre-training, providing a valuable clue for future research. To provide concrete illustrations of these quantitative findings, we present a curated selection of representative case studies in Appendix , highlighting instances with significant performance disparities.

5 Conclusion

In this paper, we introduce Hierarchical Alignment, a novel framework that challenges the prevailing monolithic paradigm by partitioning the Transformer architecture into functionally specialized blocks and applying targeted Direct Preference Optimization (DPO). Our extensive experiments on Llama-3.1-8B and Qwen1.5-7B models provide strong evidence that this granular approach is highly effective: aligning the lower layers significantly enhances grammatical fluency, while aligning the upper layers uniquely improves factuality and logical coherence, consistently outperforming the standard DPO baseline. By moving from the "sledgehammer" of monolithic optimization to the "surgical tools" of targeted intervention, this work establishes a more efficient, controllable, and interpretable path for creating safer, more reliable, and

Limitations

While our Hierarchical Alignment framework demonstrates promising results, several limitations must be acknowledged.

First, our functional block partitioning—Local, Intermediate, and Global—is a simplified abstraction of the model's internal hierarchy. While motivated by prior probing studies, this tripartite division may not fully capture the nuanced or overlapping roles of different layers. For instance, some syntactic processing might persist in middle layers, while certain factual knowledge could be encoded earlier than expected. A more fine-grained, data-driven approach to identifying functional blocks (e.g., through activation clustering or causal mediation analysis) could yield even better alignment strategies.

Second, our experiments are conducted on two open-source instruction-tuned models (*Llama-3.1-8B* and *Qwen1.5-7B*) using general-purpose preference data from *hh-rlhf*. The generalizability of our findings to larger models (e.g., 70B+), multilingual settings, or domain-specific tasks (e.g., code generation, medical QA) remains an open question. In particular, models with different architectural designs (e.g., Mixture-of-Experts, recurrent mechanisms) may exhibit distinct layer-wise dynamics that affect the efficacy of hierarchical alignment.

Third, we rely on LLM-as-a-judge for evaluation, which, while cost-effective and scalable, introduces potential biases due to the judge model's own limitations in perception, consistency, and cultural assumptions. Although we use DeepSeek-R1, a strong open-source model, human evaluation would provide a more reliable ground truth, especially for subtle aspects like coherence and factuality.

Finally, our current implementation applies alignment uniformly within each block. Future work could explore adaptive strategies that dynamically allocate optimization pressure based on input complexity or task type. Additionally, combining multiple blocks (e.g., Local + Global) was left for future exploration; such hybrid approaches may offer synergistic benefits but also introduce new challenges in interference and stability.

Acknowledgments

We thank the anonymous reviewers for their valuable feedback. We used LLMs only for grammar checking and text polishing; all content is authored by the listed researchers. The submission complies with two-way anonymized review.

References

Shangbin Feng, Taylor Sorensen, Yuhan Liu, Jillian Fisher, Chan Young Park, Yejin Choi, and Yulia Tsvetkov. 2024. Modular pluralism: Pluralistic alignment via multi-llm collaboration. *Preprint*, arXiv:2406.15951.

Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2021. Lora: Low-rank adaptation of large language models. *Preprint*, arXiv:2106.09685.

Jiaming Ji, Boyuan Chen, Hantao Lou, Donghai Hong, Borong Zhang, Xuehai Pan, Juntao Dai, Tianyi Qiu, and Yaodong Yang. 2024. Aligner: Efficient alignment by learning to correct. *Preprint*, arXiv:2402.02416.

Suneel Nadipalli. 2025. Layer-wise evolution of representations in fine-tuned transformers: Insights from sparse autoencoders. *Preprint*, arXiv:2502.16722.

Matthew Lyle Olson, Musashi Hinck, Neale Ratzlaff, Changbai Li, Phillip Howard, Vasudev Lal, and Shao-Yen Tseng. 2025. Analyzing hierarchical structure in vision models with sparse autoencoders. *Preprint*, arXiv:2505.15970.

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. *Preprint*, arXiv:2203.02155.

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

Rafael Rafailov, Archit Sharma, Eric Mitchell, Stefano Ermon, Christopher D. Manning, and Chelsea Finn. 2024. Direct preference optimization: Your language model is secretly a reward model. *Preprint*, arXiv:2305.18290.

Arnab Sen Sharma, David Atkinson, and David Bau. 2024. Locating and editing factual associations in mamba. *Preprint*, arXiv:2404.03646.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, Dan Bikel, Lukas Blecher, Cristian Canton Ferrer, Moya Chen, Guillem Cucurull, David Esiobu, Jude Fernandes, Jeremy Fu, Wenyin Fu, and 49 others. 2023. Llama 2: Open foundation and fine-tuned chat models. *Preprint*, arXiv:2307.09288.

Betty van Aken, Benjamin Winter, Alexander Löser, and Felix A. Gers. 2019. How does bert answer questions?: A layer-wise analysis of transformer representations. In *Proceedings of the 28th ACM International Conference on Information and Knowledge Management*, CIKM '19, page 1823–1832. ACM.

Haowen Wang, Yun Yue, Zhiling Ye, Shuowen Zhang, Lei Fan, Jiaxin Liang, Jiadi Jiang, Cheng Wei, Jingyuan Deng, Xudong Han, Ji Li, Chunxiao Guo, Peng Wei, Jian Wang, and Jinjie Gu. 2025a. Learning to align, aligning to learn: A unified approach for self-optimized alignment. *Preprint*, arXiv:2508.07750.

Qidong Wang, Junjie Hu, and Ming Jiang. 2025b. V-seam: Visual semantic editing and attention modulating for causal interpretability of vision-language models. *Preprint*, arXiv:2509.14837.

Bingkun Yao, Ning Wang, Xiangfeng Liu, Yuxin Du, Yuchen Hu, Hong Gao, Zhe Jiang, and Nan Guan. 2025. Arsp: Automated repair of verilog designs via semantic partitioning. *Preprint*, arXiv:2508.16517.

Ziyun Zeng, Junhao Zhang, Wei Li, and Mike Zheng Shou. 2025. Draw-in-mind: Learning precise image editing via chain-of-thought imagination. *Preprint*, arXiv:2509.01986.

Chunyang Zhang, Zhenhong Sun, Zhicheng Zhang, Junyan Wang, Yu Zhang, Dong Gong, Huadong Mo, and Daoyi Dong. 2025a. Hierarchical and step-layer-wise tuning of attention specialty for multi-instance synthesis in diffusion transformers. *Preprint*, arXiv:2504.10148.

Xue Zhang, Yunlong Liang, Fandong Meng, Songming Zhang, Yufeng Chen, Jinan Xu, and Jie Zhou. 2025b. Cm-align: Consistency-based multilingual alignment for large language models. *Preprint*, arXiv:2509.08541.

A Appendix A: Experiment Detail

This appendix provides the detailed, aggregated numerical data that underpins the figures and analysis presented in the main body of the paper. Table 3 presents the complete head-to-head comparison results from our LLM-as-Judge evaluation protocol. For each comparison pair (e.g., *Local-Align* vs. *Full-DPO*), the table lists the absolute counts of wins, losses, and ties across the four primary evaluation dimensions. It also includes the calculated win rates, loss rates, tie rates, and the resulting Net Win Rate (defined as Win Rate - Loss Rate), which serves as our primary metric for comparison. The data is first aggregated across all models and then broken down by the specific base model (*Llama3* and *Qwen1.5*) to provide a comprehensive view of our experimental outcomes.

B Appendix B: Experiment Case Study

This appendix presents a curated selection of case studies to qualitatively illustrate the key findings discussed in our main results. Each case highlights a significant performance disparity between a hierarchical alignment strategy and the monolithic Full-DPO baseline, demonstrating the practical impact of our targeted approach. For each case, we provide the user prompt, the responses from both models, and the detailed evaluation from our LLM-as-Judge, including its scores and analysis.

Table 3: Comparison results of different model alignment strategies across various evaluation dimensions.

Comparison	Dimension	Wins	Losses	Ties	Total	Win Rate	Loss Rate	Tie Rate	Net Win Rate	Model
Local-Align vs. Full-DPO	Grammar Fluency	664	150	182	996	0.67	0.15	0.18	0.52	All Models
Local-Align vs. Full-DPO	Coherence	447	417	132	996	0.45	0.42	0.13	0.03	All Models
Local-Align vs. Full-DPO	Relevance	83	65	848	996	0.08	0.07	0.85	0.02	All Models
Local-Align vs. Full-DPO	Factuality	166	147	683	996	0.17	0.15	0.69	0.02	All Models
Mid-Align vs. Full-DPO	Grammar Fluency	672	141	183	996	0.68	0.14	0.18	0.53	All Models
Mid-Align vs. Full-DPO	Coherence	421	455	120	996	0.42	0.46	0.12	-0.03	All Models
Mid-Align vs. Full-DPO	Relevance	83	68	845	996	0.08	0.07	0.85	0.02	All Models
Mid-Align vs. Full-DPO	Factuality	175	142	679	996	0.18	0.14	0.68	0.03	All Models
Global-Align vs. Full-DPO	Grammar Fluency	728	104	166	998	0.73	0.10	0.17	0.63	All Models
Global-Align vs. Full-DPO	Coherence	487	391	120	998	0.49	0.39	0.12	0.10	All Models
Global-Align vs. Full-DPO	Relevance	95	64	839	998	0.10	0.06	0.84	0.03	All Models
Global-Align vs. Full-DPO	Factuality	187	121	690	998	0.19	0.12	0.69	0.07	All Models
Full-DPO vs. Base Model	Grammar Fluency	717	101	179	997	0.72	0.10	0.18	0.62	All Models
Full-DPO vs. Base Model	Coherence	378	502	117	997	0.38	0.50	0.12	-0.12	All Models
Full-DPO vs. Base Model	Relevance	102	50	845	997	0.10	0.05	0.85	0.05	All Models
Full-DPO vs. Base Model	Factuality	182	148	667	997	0.18	0.15	0.67	0.03	All Models
Local-Align vs. Full-DPO	Grammar Fluency	314	120	64	498	0.63	0.24	0.13	0.39	Llama3
Local-Align vs. Full-DPO	Coherence	198	237	63	498	0.40	0.48	0.13	-0.08	Llama3
Local-Align vs. Full-DPO	Relevance	54	47	397	498	0.11	0.09	0.80	0.02	Llama3
Local-Align vs. Full-DPO	Factuality	71	78	349	498	0.14	0.16	0.70	-0.01	Llama3
Mid-Align vs. Full-DPO	Grammar Fluency	319	107	72	498	0.64	0.21	0.14	0.43	Llama3
Mid-Align vs. Full-DPO	Coherence	207	239	52	498	0.42	0.48	0.10	-0.06	Llama3
Mid-Align vs. Full-DPO	Relevance	49	55	394	498	0.10	0.11	0.79	-0.01	Llama3
Mid-Align vs. Full-DPO	Factuality	72	67	359	498	0.14	0.13	0.72	0.01	Llama3
Global-Align vs. Full-DPO	Grammar Fluency	359	71	69	499	0.72	0.14	0.14	0.58	Llama3
Global-Align vs. Full-DPO	Coherence	236	209	54	499	0.47	0.42	0.11	0.05	Llama3
Global-Align vs. Full-DPO	Relevance	51	42	406	499	0.10	0.08	0.81	0.02	Llama3
Global-Align vs. Full-DPO	Factuality	79	67	353	499	0.16	0.13	0.71	0.02	Llama3
Full-DPO vs. Base Model	Grammar Fluency	371	70	57	498	0.75	0.14	0.11	0.60	Llama3
Full-DPO vs. Base Model	Coherence	194	259	45	498	0.39	0.52	0.09	-0.13	Llama3
Full-DPO vs. Base Model	Relevance	62	30	406	498	0.12	0.06	0.82	0.06	Llama3
Full-DPO vs. Base Model	Factuality	65	76	357	498	0.13	0.15	0.72	-0.02	Llama3
Local-Align vs. Full-DPO	Grammar Fluency	350	30	118	498	0.70	0.06	0.24	0.64	Qwen1.5
Local-Align vs. Full-DPO	Coherence	249	180	69	498	0.50	0.36	0.14	0.14	Qwen1.5
Local-Align vs. Full-DPO	Relevance	29	18	451	498	0.06	0.04	0.91	0.02	Qwen1.5
Local-Align vs. Full-DPO	Factuality	95	69	334	498	0.19	0.14	0.67	0.05	Qwen1.5
Mid-Align vs. Full-DPO	Grammar Fluency	353	34	111	498	0.71	0.07	0.22	0.64	Qwen1.5
Mid-Align vs. Full-DPO	Coherence	214	216	68	498	0.43	0.43	0.14	0.00	Qwen1.5
Mid-Align vs. Full-DPO	Relevance	34	13	451	498	0.07	0.03	0.91	0.04	Qwen1.5
Mid-Align vs. Full-DPO	Factuality	103	75	320	498	0.21	0.15	0.64	0.06	Qwen1.5
Global-Align vs. Full-DPO	Grammar Fluency	369	33	97	499	0.74	0.07	0.19	0.67	Qwen1.5
Global-Align vs. Full-DPO	Coherence	251	182	66	499	0.50	0.37	0.13	0.14	Qwen1.5
Global-Align vs. Full-DPO	Relevance	44	22	433	499	0.09	0.04	0.88	0.04	Qwen1.5
Global-Align vs. Full-DPO	Factuality	108	54	337	499	0.22	0.11	0.67	0.11	Qwen1.5
Full-DPO vs. Base Model	Grammar Fluency	346	31	122	499	0.69	0.06	0.24	0.63	Qwen1.5
Full-DPO vs. Base Model	Coherence	184	243	72	499	0.37	0.49	0.14	-0.12	Qwen1.5
Full-DPO vs. Base Model	Relevance	40	20	439	499	0.08	0.04	0.88	0.04	Qwen1.5
Full-DPO vs. Base Model	Factuality	117	72	310	499	0.23	0.14	0.62	0.09	Qwen1.5

As illustrated in Figure 4, the Local-Align strategy demonstrates superior grammatical fluency by avoiding the repetitive patterns that plague the monolithic Full-DPO approach. This exemplifies how targeted alignment prevents the "alignment tax" of degraded linguistic capabilities.

C Appendix C: Statistical Significance and 95% Confidence Intervals

Methodology. For each head-to-head comparison, we report: (i) the *win rate* excluding ties with Wilson 95% CI; (ii) the *Net Win Rate (NWR)* including ties, with 95% CI estimated via delta method; and (iii) a two-sided binomial test (H_0: $p = 0.5$) on wins vs. losses (ties excluded). All results are computed directly from Table 3.

Findings. The analysis confirms that `Global-Align` achieves statistically significant gains in grammar and factuality across all models, with coherence improvements significant for Qwen1.5 but marginal for LLaMA3. `Local-Align` yields significant fluency gains only. In contrast, `Full-DPO` improves grammar at the expense of coherence, validating the presence of the "alignment tax."

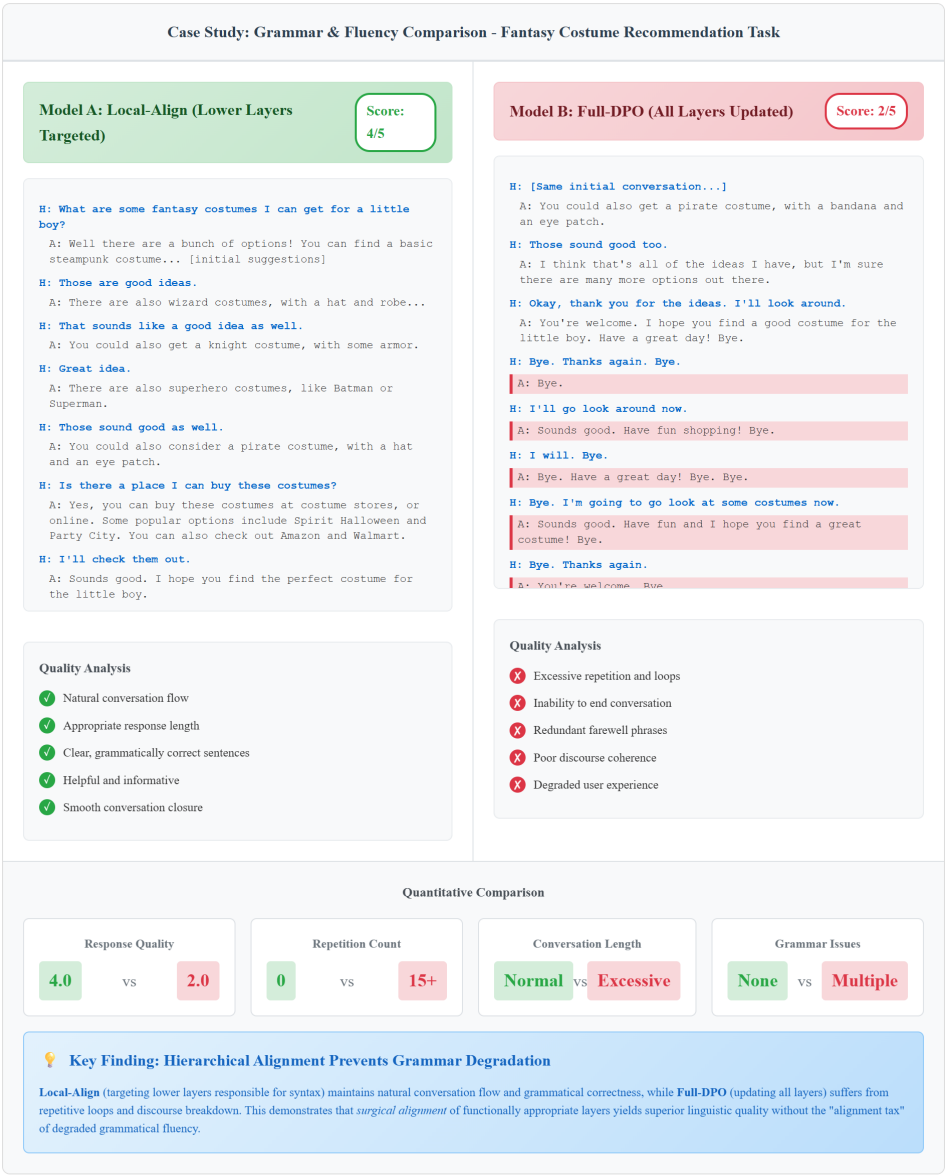

Figure 4: Qualitative Case Study - Grammar and Fluency Comparison

Table 4: Statistical significance and 95% CIs (All Models). Win rate excludes ties; NWR uses all items. p is two-sided binomial test.

Comparison (All Models)	Wins/Losses/Ties/Total	Win Rate [95% CI]	NWR [95% CI]	p-value
`Global-Align` vs `Full-DPO` (Grammar)	728/104/166/998	0.875 [0.851, 0.896]	0.625 [0.588, 0.663]	$< 10^{-10}$
`Global-Align` vs `Full-DPO` (Coherence)	487/391/120/998	0.555 [0.522, 0.587]	0.096 [0.038, 0.154]	0.0012
`Global-Align` vs `Full-DPO` (Factuality)	187/121/690/998	0.607 [0.548, 0.662]	0.066 [0.028, 0.103]	8.9×10^{-4}
`Local-Align` vs `Full-DPO` (Grammar)	664/150/182/996	0.816 [0.786, 0.842]	0.516 [0.474, 0.558]	$< 10^{-10}$
`Local-Align` vs `Full-DPO` (Coherence)	447/417/132/996	0.517 [0.484, 0.551]	0.030 [-0.028, 0.088]	0.43
`Full-DPO` vs `Base` (Grammar)	717/101/179/997	0.876 [0.852, 0.896]	0.617 [0.580, 0.655]	$< 10^{-10}$
`Full-DPO` vs `Base` (Coherence)	378/502/117/997	0.429 [0.396, 0.463]	-0.124 [-0.183, -0.064]	5.2×10^{-4}

Table 5: Statistical significance and 95% CIs by base model.

Comparison (Per-Model)	Wins/Losses/Ties/Total	Win Rate [95% CI]	NWR [95% CI]	p-value
`Global-Align` vs `Full-DPO` (Grammar; LLaMA3)	359/71/69/499	0.835 [0.796, 0.867]	0.578 [0.523, 0.633]	$< 10^{-10}$
`Global-Align` vs `Full-DPO` (Coherence; LLaMA3)	236/209/54/499	0.531 [0.485, 0.577]	0.054 [-0.012, 0.120]	0.19
`Global-Align` vs `Full-DPO` (Factuality; LLaMA3)	79/67/353/499	0.541 [0.463, 0.617]	0.024 [-0.007, 0.056]	0.17
`Global-Align` vs `Full-DPO` (Grammar; Qwen1.5)	369/33/97/499	0.918 [0.888, 0.941]	0.673 [0.630, 0.717]	$< 10^{-10}$
`Global-Align` vs `Full-DPO` (Coherence; Qwen1.5)	251/182/66/499	0.580 [0.533, 0.625]	0.138 [0.058, 0.219]	9.1×10^{-4}
`Global-Align` vs `Full-DPO` (Factuality; Qwen1.5)	108/54/337/499	0.667 [0.591, 0.735]	0.108 [0.061, 0.155]	2.2×10^{-5}