ECHOALIGN: LEARNING PERSONALIZED STYLE FROM PARTS TO WHOLE

Anonymous authors
Paper under double-blind review

ABSTRACT

Users differ in how they prefer responses to be expressed, including their desired levels of style attributes such as formality and concision. Supporting diverse style preferences in production calls for adaptation from limited preference data and compact personalization parameters alongside a shared model. We introduce ECHOALIGN, a family of two-stage preference-learning methods that progressively encode personalized style guidance into a compact soft prompt while keeping the language-model backbone frozen. The first stage uses attribute-level preference pairs to train dedicated prompt segments, giving each segment a focused learning objective. The second stage uses overall-style preference pairs to adapt their combination so that they jointly guide generation according to the user's complete style preference. Each complete style preference has a separately learned prompt whose 256 token embeddings can be precomputed for reuse across questions, supporting pluggable personalization with shared model weights. Using approximately 100 preference pairs per profile, ECHOALIGN achieves relative improvements in style alignment of 55.8–62.5% over unadapted base models and 2.9–11.3% over the strongest tested LoRA baselines across three backbones. We further apply ECHOALIGN to personalized message generation in production, where a randomized A/B test shows a 6.49% relative increase in click-through rate. This real-user behavioral feedback suggests improved satisfaction with the generated messages.

1 INTRODUCTION

Adapting an assistant's response style to individual preferences can improve user satisfaction. For example, users asking the same question may prefer a brief, direct reply or a detailed explanation. These preferences interact: even a short reply may need to be warm and creative. Supporting this diversity calls for compact style guidance that can be learned from limited preference data and reused across questions with a shared model.

We introduce ECHOALIGN, a family of two-stage preference-learning methods that progressively encode personalized style guidance into a compact soft prompt—a short sequence of additional input embeddings (Lester et al., 2021). The first stage uses attribute-level preference pairs to learn dedicated prompt segments. Each segment has a focused learning objective and can be trained independently of the other learned segments. This decoupled design provides an engineering benefit: the saved segments form a reusable initialization for different second-stage adaptations, without repeating attribute learning. The second stage uses overall-style preference pairs to adapt their concatenation to the user's complete style preference. Training thus progresses *from parts to whole*, from individual attributes to the overall response style, while the language-model backbone remains frozen.

During attribute learning, only the designated segment is updated; the remaining segments stay at their neutral initialization. Concatenating the learned segments changes this context: each now guides generation alongside segments trained for other attributes of the same style preference. The second stage trains under this combined prompt to learn how the segments jointly guide generation toward the user's overall style preference. The choice of trainable parameters and preference data in this stage defines a variant of ECHOALIGN. We focus on two variants: FUSE learns a transformation of fixed segments, while REFINE+MIX refines their embeddings with both attribute-level and overall-style supervision. Both stages use direct preference optimization (DPO) (Rafailov et al., 2023); Section 3 details the procedure.

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

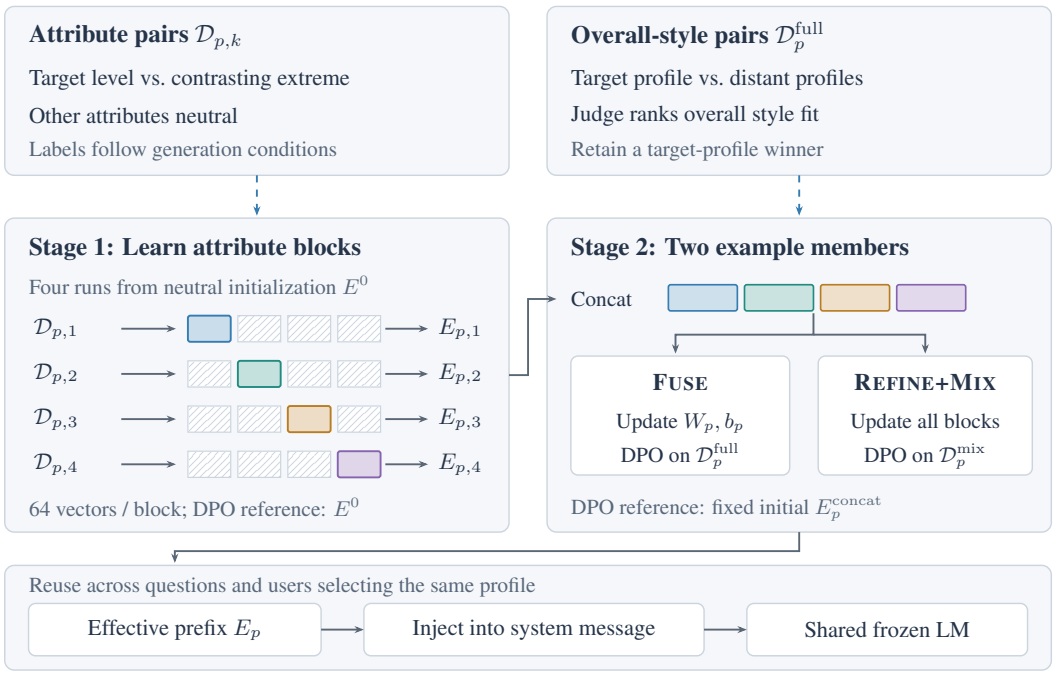

Figure 1: ECHOALIGN learns a prefix for each complete style profile. Attribute-level pairs train designated blocks; overall-style preferences adapt their concatenation. The two illustrated members, FUSE and REFINE+MIX, transform fixed blocks or refine the embeddings. Colored blocks are learned; hatched blocks stay at neutral initialization. The effective prefix can be precomputed for reuse with a frozen backbone.

We train a separate prompt for each complete style preference, allowing independent updates without changing other prompts or the shared model. Each learned prompt contains 256 token embeddings (Figure 1). The prompt is reused across questions and by users sharing that preference. At inference, these embeddings replace the input embeddings at fixed positions within the system message. This compact input representation supports pluggable personalization: the prefix can be precomputed once per profile, and requests with different style preferences can share an inference batch with the same model weights (Section 3.4).

We define a complete style preference by one target level for each of four attributes—formality, concision, warmth, and creativity—with five levels per attribute. We evaluate all $5^4 = 625$ trained combinations on 17 held-out questions across three backbones and three seeds. ECHOALIGN learns from approximately 100 preference pairs per complete style preference, using the same training corpus as the LoRA baselines. With FUSE on Qwen and Llama and REFINE+MIX on Ministral, mean overall style fit improves by 0.69 points (11.3%), 0.31 points (5.0%), and 0.18 points (2.9%), respectively, over the strongest tested profile-prompted LoRA baselines (Table 1). We further evaluate personalized message generation in production, where a randomized A/B test of a REFINE+MIX routing policy raises impression-based click-through rate by 6.49% over explicit style prompting (Table 3).

Our contributions are threefold:

- We develop a two-stage method family that encodes personalized style guidance into compact, reusable input prompts. Preference data at two granularities first train dedicated attribute segments and then adapt their combination to the complete style preference (Section 3).
- We systematically compare five family members with LoRA and direct prefix training across all 625 profiles and three backbones, and use controlled ablations and attribute analyses to examine adaptation choices and their limitations (Sections 4.2 to 4.4).

- We complement offline style evaluation with external model judges and a randomized production A/B test reporting improved real-user click engagement (Sections 4.5 and 4.6).

2 Related Work

Personalization. LaMP conditions outputs on retrieved user histories (Salemi et al., 2024); P-RLHF learns user representations and personalized policies from individual feedback (Li et al., 2024). Shared low-rank adaptation also captures common structure in personalized reward models (Liu et al., 2025). CLaP uses Latent DPO for efficient preference adaptation (Zhang et al., 2025). We take an explicit style profile as given and learn reusable guidance from teacher-generated preferences; users with the same profile share a prefix. PRISM-X evaluates personalized fine-tuning with real and simulated users (Kirk et al., 2026); we similarly distinguish automated evaluation from observed user behavior, using production clicks as the latter.

Parameter-efficient adaptation. Prefix-tuning learns continuous layer activations (Li & Liang, 2021), prompt tuning learns input embeddings (Lester et al., 2021), and LoRA learns low-rank weight updates (Hu et al., 2022). Our representation follows input-level prompt tuning. AdapterFusion separately trains task adapters before learning their composition (Pfeiffer et al., 2021); we organize attribute-level and overall-style DPO supervision within each profile.

Multi-attribute control. SteerLM conditions supervised fine-tuning on response attributes (Dong et al., 2023), while Janus trains a shared model on textual system messages and tests unseen preference descriptions (Lee et al., 2024). Personalized Soups learns preference-specific policies with PPO and averages their parameters, using three binary dimensions in its LoRA experiments (Jang et al., 2023). MAGIC uses disentangled counterfactual augmentation (Liu et al., 2024).

Composing attribute prompts. Tailor learns single-attribute prompts and addresses their concatenation mismatch with masking, re-indexed positions, or a trained prompt connector (Yang et al., 2023). It is the closest architectural precedent for our parts-to-whole design. ECHOALIGN studies a preference-learning family: attribute-level DPO trains designated input blocks, and overall-style DPO adapts an affine map, the embeddings, or both, with overall-style or mixed supervision. Each complete profile is trained separately, across four five-level attributes. Our experiments compare five instantiations with direct prefix training and per-profile LoRA; they do not test Tailor's unseen-combination capability or establish superiority over policy merging and shared profile-conditioned models.

3 Method

ECHOALIGN defines a family of methods that learn attribute-specific prompt blocks and then adapt their combination using overall-style preferences (Figure 2). All members keep the backbone frozen and share Stage 1; Stage 2 varies the updated parameters and supervision.

3.1 Problem Definition

We represent a user's style preferences by $S_p \in \{1, \ldots, 5\}^4$, whose coordinates specify target intensities for formality, concision, warmth, and creativity; p indexes the 625 combinations. Each level specifies an attribute's intensity rather than response quality; Section A.1 gives the attribute choices and level anchors. Given a question x and S_p, the goal is to generate a response y_p that answers x appropriately and matches the requested style.

3.2 Stage 1: Learning Individual Style Attributes

We divide the prefix into four embedding blocks $E_{p,k}$, one for each style attribute k. We initialize these blocks from token embeddings of a neutral style description (Section B.1), yielding the initial prefix E^0. Each block is trained to guide its dimension toward the requested intensity.

For attribute k, the teacher answers the same question at the intensity specified by S_p and at a contrasting extreme. The other attributes remain neutral. The resulting preference pairs form $\mathcal{D}_{p,k}$,

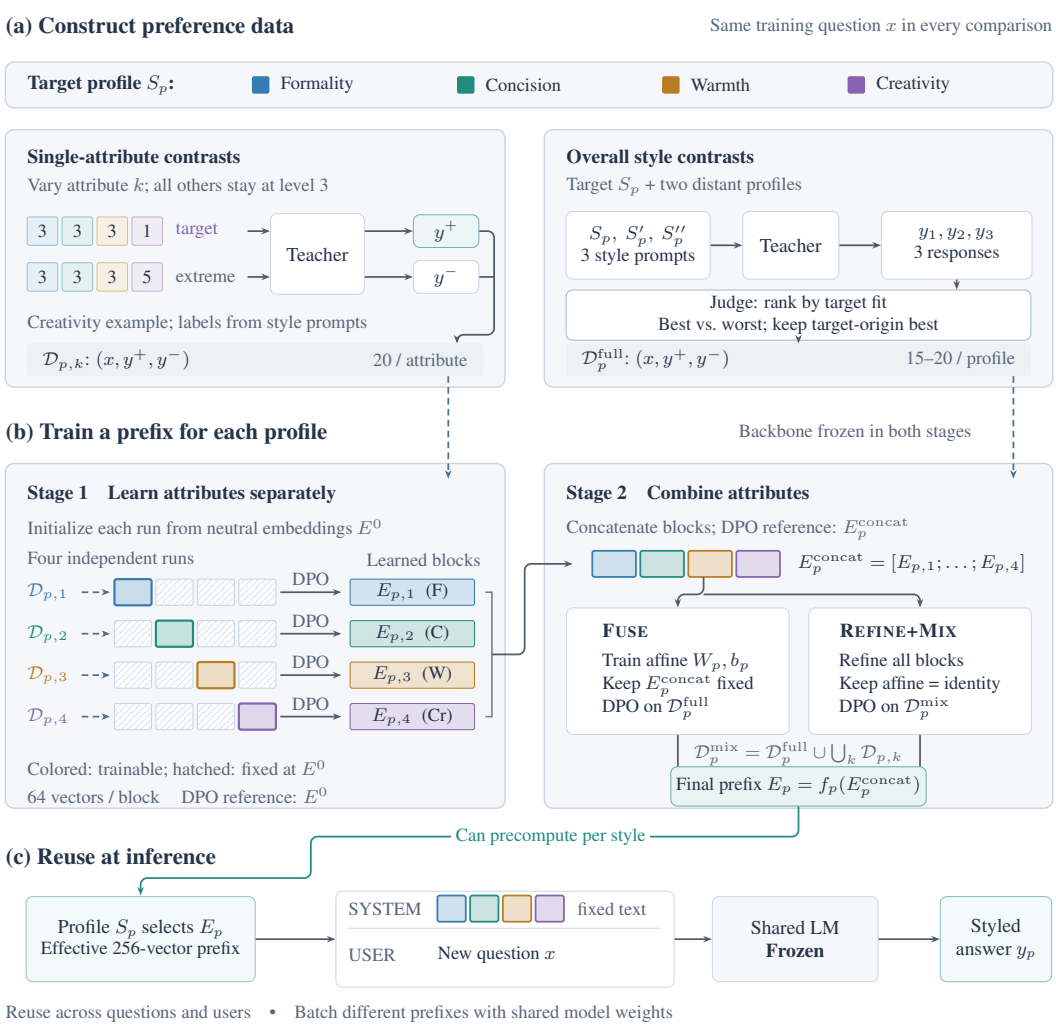

Figure 2: ECHOALIGN pipeline. (a) A teacher generates responses to the same question under contrasting style conditions. Attribute-pair labels follow the generation prompts; overall-style pairs are ranked by a judge and filtered to retain a target-style winner. (b) DPO first trains four attribute blocks separately, then adapts their combination; FUSE and REFINE+MIX illustrate two family members. Colors identify attributes; hatching marks blocks fixed at neutral initialization. Dashed arrows supply preference data; solid arrows show the workflow. The backbone remains frozen throughout. (c) At inference, the prefix prepared for the requested profile is injected into the system-message embeddings and reused across questions.

with the target-level response preferred. The construction rules are specified in Section A.2, with an example in Figure 2a.

When learning attribute k, we update only $E_{p,k}$ and keep the other three blocks at their neutral initialization. We denote the full prefix used in this run by $\widetilde{E}_{p,k}$. We use DPO (Rafailov et al., 2023) to favor the preferred response y^+ over the rejected response y^-, relative to a fixed reference:

$$\mathcal{L}(E_{p,k}; \mathcal{D}_{p,k}) = -\mathbb{E}_{(x,y^+,y^-)\sim\mathcal{D}_{p,k}} \log \sigma\left(\beta\left[\log \frac{\pi(y^+ \mid x, \widetilde{E}_{p,k})}{\pi(y^+ \mid x, E^0)} - \log \frac{\pi(y^- \mid x, \widetilde{E}_{p,k})}{\pi(y^- \mid x, E^0)}\right]\right). \tag{1}$$

Here π is the frozen backbone, σ is the logistic sigmoid, and β is the DPO coefficient. Log probabilities include only response tokens. After learning all four attributes, we concatenate their

trained blocks as $E_p^{\text{concat}} = [E_{p,1}; \ldots; E_{p,4}]$. We train and save these blocks separately for each style profile; their concatenation can initialize multiple Stage 2 variants. This provides an attribute-level training interface. Section 4.3 tests its score contribution against alternative training organizations; Section B.2 gives the main-table settings.

3.3 Stage 2: Learning to Combine Style Attributes

Stage 2 adapts the separately learned blocks to guide generation jointly under S_p, writing $E_p = f_p(E_p^{\text{concat}})$. Family members update an affine transformation, the embeddings, or both, using overall-style pairs alone or a mixture with attribute-level pairs.

All members reuse Equation (1) with E_p as the input prefix and their respective Stage 2 data. The reference E_p^{ref} stays fixed at the initial E_p^{concat}. A judge ranks teacher responses generated under S_p and two distant style profiles by how well they express the target style. We retain pairs whose preferred response was generated under S_p, forming $\mathcal{D}_p^{\text{full}}$ (Section A.2).

FUSE keeps the embedding blocks fixed and learns an affine transformation, $f_p(E_p^{\text{concat}}) = E_p^{\text{concat}} W_p^{\mathsf{T}} + \mathbf{1} b_p^{\mathsf{T}}$. The same W_p, b_p apply at every prefix position and are initialized to $W_p = I, b_p = 0$. Here W_p is square in the embedding dimension and b_p is a bias vector. Training on $\mathcal{D}_p^{\text{full}}$ adjusts how the fixed embeddings jointly guide the response.

REFINE+MIX realizes f_p by directly refining all four embedding blocks in E_p^{concat} through preference optimization. The refined embeddings form E_p, with the affine parameters fixed at $W_p = I, b_p = 0$. We form $\mathcal{D}_p^{\text{mix}}$ by pooling $\mathcal{D}_p^{\text{full}}$ with all four attribute pair sets and shuffling the combined pairs. This keeps individual attribute requirements in the training objective while the embeddings learn to work together.

These two members illustrate distinct adaptation choices. We also test REFINE, which updates the embeddings using only $\mathcal{D}_p^{\text{full}}$; JOINT, which updates both the embeddings and affine map on that data; and JOINT+MIX, which updates both on $\mathcal{D}_p^{\text{mix}}$. Table 5 lists all five tested members alongside CONCAT (stopping after Stage 1) and MONOSP (direct mixed-data prefix training from E^0) as training controls.

3.4 Inference

The final 256-vector prefix E_p is independent of the query and can be precomputed once per profile. We inject it into the system-message embeddings for questions and users selecting S_p. Generation needs no target-profile text, reference pass, or further optimization. The current loader retains the blocks and affine parameters from training; the compact prefix is the equivalent effective representation (Section B.3.2). Different profiles share the same frozen backbone, making mixed-profile batching possible at the input level; we do not benchmark its throughput here.

4 Experiments

We evaluate whether ECHOALIGN improves style alignment with limited preference data, how its variants differ, and how responses track the requested attributes. External judges assess the reliability of the offline comparisons, and a randomized deployment tests practical value through real-user feedback.

4.1 Experimental Setup

Data and evaluation scope. We evaluate all 625 style profiles on 17 held-out questions spanning eight everyday scenario categories. The 20 training questions are textually disjoint but cover overlapping domains (Section A.3). A commercial closed-source LLM generates approximately 100 preference pairs per profile using the attribute-level and overall-style procedures in Sections 3.2 and 3.3. Each profile is trained separately; evaluation measures generalization to new questions for these trained profiles. Sections A.2 and B.2 give the data construction and training settings.

Table 1: Style-alignment scores (↑), mean ± sample SD over three seeds. Full-625 is the grid mean; Top-64 uses retained-user frequency weights. Representative comparisons use the higher observed Full-625 mean of FUSE and REFINE+MIX per backbone; bold marks that member's results, rather than the best of all methods. The final column gives rule-detected Full-625 collapse percentages in Qwen / Llama / Ministral order; collapsed answers score zero.

	Qwen2.5-7B		Llama-3.1-8B		Ministral-3-8B		Rule-detected
Method	Full-625	Top-64	Full-625	Top-64	Full-625	Top-64	collapse (%)
Prompting and retrieval							
Base*	4.17±0.04	3.87±0.01	4.17±0.02	3.89±0.03	3.98±0.00	3.55±0.00	0.00 / 0.00 / 0.00
ProfilePrompt	5.20±0.03	5.87±0.07	5.41±0.03	5.07±0.08	5.10±0.00	4.81±0.01	0.00 / 0.00 / 0.00
ICL-All	5.35±0.01	5.12±0.01	5.09±0.02	4.44±0.01	5.74±0.00	4.87±0.01	0.00 / 0.00 / 0.00
ICL-RAG	5.55±0.02	5.29±0.06	5.46±0.02	4.86±0.04	5.48±0.00	4.83±0.01	0.00 / 0.00 / 0.00
Per-profile LoRA							
UserLoRA-r4	4.97±0.02	4.70±0.05	5.03±0.02	4.62±0.06	4.77±0.01	4.15±0.02	0.00 / 0.00 / 0.00
UserLoRA-r8	5.01±0.02	4.73±0.07	5.09±0.02	4.71±0.07	4.84±0.01	4.22±0.02	0.00 / 0.00 / 0.00
UserLoRA-r16	5.03±0.01	4.71±0.06	5.14±0.03	4.81±0.04	4.89±0.00	4.35±0.08	0.00 / 0.01 / 0.00
PP+LoRA-r4	6.07±0.07	6.12±0.11	6.18±0.05	5.84±0.08	6.10±0.01	5.94±0.07	0.00 / 0.00 / 0.00
PP+LoRA-r8	6.08±0.07	6.09±0.11	6.18±0.06	5.70±0.06	6.10±0.00	5.87±0.05	0.00 / 0.00 / 0.00
PP+LoRA-r16	6.09±0.08	6.07±0.11	6.16±0.05	5.70±0.06	6.11±0.01	6.06±0.03	0.00 / 0.00 / 0.00
EchoAlign family: five tested members							
FUSE	**6.77**±0.02	**6.85**±0.03	**6.49**±0.02	**5.87**±0.21	5.09±0.01	4.48±0.20	0.00 / 0.12 / 23.09
REFINE+MIX	6.47±0.04	6.66±0.10	6.06±0.02	5.43±0.05	**6.29**±0.00	**6.55**±0.12	0.00 / 0.00 / 0.21
JOINT	6.78±0.02	6.88±0.05	6.51±0.02	5.78±0.17	5.39±0.04	4.63±0.18	0.00 / 0.06 / 18.95
REFINE	6.68±0.03	6.94±0.07	6.42±0.01	5.78±0.08	6.20±0.01	5.85±0.34	0.00 / 0.10 / 2.66
JOINT+MIX	6.41±0.05	6.44±0.10	6.03±0.03	5.62±0.05	6.11±0.01	6.19±0.07	0.00 / 0.03 / 3.09
Prefix-training controls							
CONCAT	6.03±0.04	6.76±0.07	6.04±0.01	5.57±0.07	6.10±0.01	6.17±0.11	0.00 / 0.00 / 0.03
MONOSP	6.16±0.06	6.17±0.19	6.20±0.01	5.92±0.07	6.18±0.01	6.21±0.03	0.00 / 0.00 / 0.00

PP: ProfilePrompt. *Base profile mapping is inferred; see Section D.3. Differences are computed before rounding.

Models and baselines. We use Qwen2.5-7B-Instruct, Meta-Llama-3.1-8B-Instruct, and Ministral-3-8B. Table 1 groups comparisons into prompting and retrieval, per-profile LoRA, our five family members, and prefix-training controls. Base receives no style profile; ProfilePrompt (PP) supplies one as text; ICL-All uses all demonstrations, and ICL-RAG retrieves the top two. UserLoRA-r{4,8,16} omits profile text, whereas PP+LoRA-r{4,8,16} includes it. In the main comparison, trained methods draw on the same preference corpus and use the same training-EMA checkpoint-selection rule, with method-specific optimization schedules. The comparison covers independently learned profile parameters; jointly trained profile-conditioned adapters are outside its scope. Section B.3.1 reports a longer-training check for LoRA.

Evaluation. We measure style alignment using the judge's overall 0–10 style-fit score under a shared rubric. We report mean ± sample SD over three seeds and rule-detected collapse rates. Collapsed answers receive zero; Sections D.1 and E.3 give the scoring and failure-handling protocols. Full-625 gives equal weight to every profile. Top-64 weights 64 common profiles by their inferred frequencies among 12,488 users, covering approximately 95% of users retained after confidence filtering (Section D.3). The same commercial LLM generates preference data and judges outputs; Section 4.5 checks the comparisons with external judges and distinguishes style alignment from overall answer preference.

4.2 MAIN RESULTS: STYLE ALIGNMENT

Several ECHOALIGN variants outperform the strongest tested profile-prompted LoRA baseline on each backbone (Table 1). The same REFINE variant improves the Full-625 mean by 9.7%, 3.8%, and 1.4% on Qwen, Llama, and Ministral, respectively. Thus, the family's gains extend beyond the two representative variants discussed below.

Overall alignment. With FUSE on Qwen and Llama and REFINE+MIX on Ministral, the gains over the strongest tested PP+LoRA baseline are 0.69 points (11.3%), 0.31 points (5.0%), and 0.18 points (2.9%), respectively. All three advantages hold in every seed. These members also exceed direct prefix training (MONOSP) by 0.61, 0.29, and 0.11 points.

Alignment under user-frequency weighting. The advantage persists on Top-64 for Qwen FUSE and Ministral REFINE+MIX, which score 6.85 and 6.55 versus 6.12 and 6.06 for their strongest PP+LoRA baselines. Both advantages hold in every seed. Llama shows a stronger dependence on training seed. In seed 42, FUSE reaches 6.11, exceeding both PP+LoRA-r4 (5.75) and MONOSP (5.85) by 6.4% and 4.6%, respectively. Its scores fall to 5.78 and 5.72 in seeds 43 and 44, giving a three-seed mean of 5.87 ± 0.21, close to PP+LoRA-r4's 5.84 ± 0.08 and below MONOSP's 5.92 ± 0.07. Much of this variation comes from degraded responses on one frequent profile; Section E.1 traces its effect on the weighted result.

Adaptation strategies across backbones. The preferred way to combine learned attributes depends on the backbone. On Qwen and Llama, transforming fixed segments (FUSE) captures most of the improvement over CONCAT; additionally updating the embeddings (JOINT) brings little further gain. On Ministral, refining the embeddings with mixed supervision (REFINE+MIX) achieves higher alignment and fewer collapsed responses than FUSE. These results motivate the method-family design, with different adaptation strategies suited to different backbones. The matched controls in Section 4.3 examine individual design choices within these complete training recipes.

4.3 TRAINING DATA AND PROMPT SEGMENTATION

Decoupled training and reusable initialization. We examine whether the first stage can retain competitive alignment while providing the engineering benefits of independent block training and reuse. Each block is learned with the other blocks held at their neutral initialization, so its training does not depend on already learned blocks. The saved blocks provide a reusable initialization for multiple Stage 2 variants.

Data shuffling and prompt segmentation. We compare two choices on 64 common profiles and 17 questions, with one training and partition seed. For the data, we either keep preference pairs grouped by attribute or shuffle the same pairs across attributes into four balanced subsets. For the prompt, segmented training learns one block independently on each subset; unsegmented training updates all 256 prompt vectors on every batch. In the unsegmented case, updates continue on the same prompt as training moves between subsets. Within each Stage 2 branch, the controls match pair exposure, learning-rate schedules, and reference-reset rules. Table 2 compares the resulting adapted prompts using the mean of the four attribute-fit scores and equal profile weights. Its metric and weighting differ from the main table; Section C gives the protocol and missing-rating handling.

Table 2: Training-data and prompt-segmentation controls on 64 equally weighted profiles and 17 questions, seed 42. Values average the four attribute-fit scores. All rows use FUSE on Qwen/Llama and REFINE+MIX on Ministral in Stage 2. Shuffled data redistribute the same pairs into balanced mixed-attribute subsets. Segmented training learns blocks independently; unsegmented training updates the full prompt. Cells with missing ratings use bound midpoints (Section C).

Training data	Segmented training	Qwen	Llama	Ministral
By attribute	Yes	6.650	6.036	6.395
Shuffled	Yes	6.698	5.945	6.432
By attribute	No	6.644	6.051	6.453
Shuffled	No	6.652	6.064	6.563

Observed alignment across training organizations. In the adaptation branches shown in Table 2, segmented and unsegmented training yield close observed means under both data organizations. These results support independent block training as a practical initialization choice with the flexibility of decoupled training and reuse. The main-table comparison with MONOSP provides evidence for the complete two-stage recipes: FUSE on Qwen and Llama and REFINE+MIX on Ministral improve the

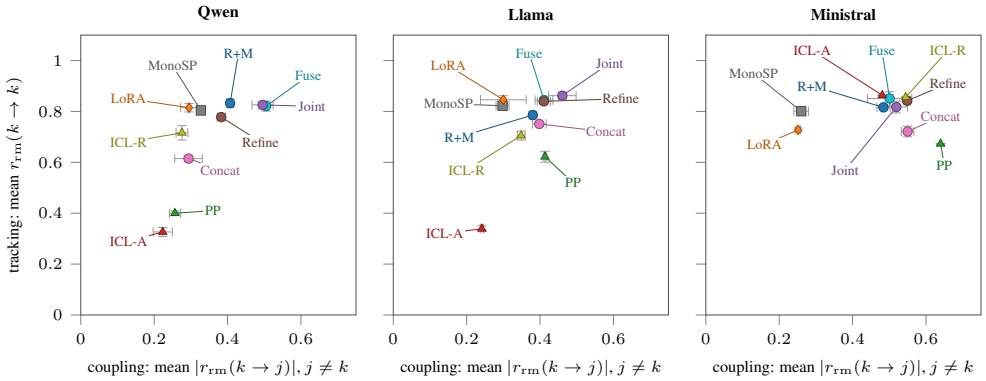

Figure 3: Attribute tracking comes with cross-attribute coupling. Tracking averages target correlations; coupling averages absolute non-target correlations. Error bars show sample SD across three seeds. R+M = REFINE+MIX; PP = ProfilePrompt; LoRA = PP+LoRA-r16; ICL-A/R = ICL-All/RAG. Full signed matrices appear in Figure 5.

Full-625 mean by 0.61, 0.29, and 0.11 points, respectively (Section 4.2). Together, these comparisons show how a reusable first stage can be combined with effective overall adaptation.

4.4 ATTRIBUTE CONTROL

We examine how responses change as the target level of one style attribute varies, and whether other response attributes change with it. For a fixed question, we compare five profiles that hold three attribute levels fixed and vary the fourth from 1 to 5. Each profile uses its own learned parameters. A judge that does not see the targets rates all four response attributes. *Tracking* measures how closely an attribute follows its target level; *coupling* measures changes in the other attributes. We compute these using repeated-measures correlations (Bakdash & Marusich, 2017) on scorable responses to five questions and four backgrounds per attribute, across three seeds.

Responses follow target levels. Qwen and Llama FUSE track target levels similarly to PP+LoRA-r16, but respond more strongly: target slopes are 0.51–0.61 intensity points per level, compared with 0.37–0.43 for LoRA. Ministral REFINE+MIX improves both tracking (0.82 versus 0.73) and slope (0.45 versus 0.36).

Other attributes also change. These variants have higher coupling than PP+LoRA-r16 (0.41–0.51 versus 0.25–0.30 across backbones; Figure 3), and MONOSP also has lower coupling. Their stronger response to a target therefore comes with changes in other attributes. Adjacent target levels separate unevenly, especially at the extremes (Figure 4). Full matrices, level-by-level analyses, and response examples appear in Sections E.4 and F.1.

4.5 EVALUATOR VALIDATION

External judges preserve the main comparisons. We re-evaluate a frozen sample of 5,202 records covering 17 methods, three backbones, and three seeds using GPT-4o (2024-08-06), Gemini 2.5 Pro, and Qwen3-235B-A22B, without revealing method or backbone labels. Under the same rubric, their averaged ratings yield method-rank correlations of 0.965–0.982 with the primary judge. All 15 prespecified mean method differences remain positive, including in a sensitivity check excluding Qwen3. These results support the aggregate comparisons on the sampled conditions; Section D.2 reports individual confidence intervals, missing ratings, and the sensitivity analysis. Prefix and LoRA methods draw on the same teacher-generated corpus, controlling the data source in that comparison; comparisons with prompting may still favor styles learned from the teacher.

The metric measures requested style. Checks on existing human-annotated datasets support sensitivity to verbosity, formality, creativity, and related aspects of warmth (Section D.2). These

assess attribute sensitivity; generated outputs in our experiments are evaluated by model judges. Frequent ties between neighboring target levels limit fine-grained calibration. We therefore use the score to assess aggregate style alignment and separately test practical value through user behavior in deployment.

4.6 Online Personalization in Production

We apply EchoAlign to personalized push-message generation and evaluate real-user responses in a randomized production A/B test. Both arms use Qwen3.5-35B-A3B and the same delivery-frequency cap. The baseline provides explicit style instructions. The treatment uses Refine+Mix without those instructions for eligible profiles, covering 64.64% of users, and follows the baseline route otherwise. Metrics include the entire bucket, including these fallback users.

We select Refine+Mix to balance style alignment, attribute control, and generation stability. On the offline Qwen2.5 backbone, from the same model family as the deployed model, it outperforms all tested LoRA, prompting, and retrieval baselines, as well as MonoSP: its Full-625 score is 6.47, versus 6.09 for the strongest LoRA baseline. It matches Joint's attribute tracking (0.83), with lower coupling than Fuse and Joint (0.41 versus 0.51 and 0.50; Table 14). It is also the only family member whose rule-detected collapse rate stays at or below 0.21% on all three offline backbones (Table 1).

Personalization raises impression-based click-through rate from 1.3390% to 1.4259%, a 6.49% relative increase (Table 3). Recipient-level click-through rate also increases by 6.97%. Both gains are statistically significant in the platform's nominal two-sided tests (Table 3). Its stratified analysis also supports a positive effect on impression-based CTR: the risk ratio is 1.0610 (95% CI $[1.0196, 1.1041]$; nominal $p \approx 0.0036$). This positive behavioral feedback supports the practical value of personalizing generated messages. The results describe the deployed routing policy, including fallback. Section G gives the evaluation protocol and statistical analysis.

Table 3: Personalization increases click-through rates in the randomized A/B test. Both metrics cover the entire bucket. Impression CTR counts dispatches; recipient CTR deduplicates recipients. Nominal two-sided p-values are based on platform-reported tests. Aggregate dashboard statistics and inference details appear in Table 15.

Metric	Baseline	Refine+Mix policy	Relative change	Nominal p
Impression CTR	1.3390%	1.4259%	+6.49%	1.03×10^{-5}
Recipient CTR	2.3804%	2.5463%	+6.97%	6.18×10^{-6}

5 Limitations and Conclusion

Limitations. Offline evaluation uses 17 held-out questions and 625 style profiles. Generalization to unseen profile combinations and broader tasks is left to future work. Training uses a single teacher, and fine-grained level calibration can be improved further. The grouping controls cover 64 profiles under a single seed. The online A/B test measures click engagement, which is an indirect proxy for user satisfaction; direct user surveys were not conducted due to cost and time constraints.

Conclusion. EchoAlign is a family of methods that encodes style guidance into reusable input prefixes through attribute-level and overall-style preference learning. Across 625 profiles and three backbones, the highlighted members improve full-grid style fit over prompting, retrieval, and per-profile LoRA baselines. External judges corroborate sampled comparisons, and a randomized online deployment of a Refine+Mix policy shows higher click engagement.

AI USE STATEMENT

Generative AI assistants supported the development of the experimental methodology and statistical analyses, interpretation of results, code implementation and debugging, literature search, figure preparation, and manuscript drafting and revision. In the experiments, a commercial closed-source LLM generated training responses, ranked overall-style pairs, and provided style-fit and intensity ratings. External Evaluator A generated controlled rewrites. Evaluators A–C supplied blinded reference labels and external fit ratings for evaluator validation; an earlier validation also used Evaluator B to check content preservation. The user-weighted analysis also uses existing LLM-inferred style labels. We distinguish model-generated labels and ratings from the existing human judgments used to validate the evaluator. Verification includes unit tests for statistical routines, independent recomputation from stored evaluation records, and checking cited papers against their original sources. The authors directed the work and take responsibility for the final text, analyses, and claims, including AI-assisted contributions.

ETHICS STATEMENT

The user-profile data come from a deployed commercial assistant and were processed to protect user privacy. Use of these data and the product A/B test followed company privacy requirements and remained within the scope of user authorization. The user-weighted evaluation uses aggregate counts of inferred style profiles; the paper does not disclose identifying user records. Training examples are synthetic responses to generic questions under predefined style profiles. Evaluator validation reuses existing human annotations and does not involve a newly recruited participant study. Separately, the ongoing randomized product A/B test involves live delivery of generated push messages; we report aggregate exposure and click statistics from an interim analysis. Only users with an eligible style and sufficient profile confidence receive the adapted route; others retain the baseline. No identifying query history or individual click log is disclosed. Click engagement is distinguished from direct satisfaction, and we do not infer long-term benefit from this interim result. Inferred profiles and automated scores may reflect model biases and should not be equated with users' stated preferences or satisfaction. Style adaptation may also make unsupported answers more convincing; improved style fit does not establish factual reliability or safety, which require separate evaluation.

REPRODUCIBILITY STATEMENT

Section 3 specifies the training objectives and inference procedure. Sections A and B describe profile construction, preference-pair generation and filtering, test questions, model configurations, training budgets, and checkpoint selection. Sections D to F document scoring, aggregation, failure handling, and the control analyses. We retain generated responses and ratings for offline analysis, and independently recompute the reported control correlations and LoRA budget comparisons from the stored records. The main comparisons report variation across three seeds; the longer LoRA schedule check is explicitly limited to seed 42. The training-data and prompt-segmentation controls also use seed 42; Section C specifies their schedules, scoring, and missing-rating handling. The appendix supplies the full fit rubric, parser behavior, explicit generation settings, and collapse rules with archived review-label counts. Historical checkpoint generation configurations were not captured in the retained manifests, limiting exact regeneration. Cross-judge and target-matching tables are derived from stored results; no main-table scores were replaced. Online statistics are reported from the experiment dashboard, with aggregate rates and pooled and stratified analyses in Section G. Stratum-level exposure and click counts and user-level logs are not included, so the tests and stratified interval cannot be independently recomputed from these summaries. Exact regeneration also depends on the availability and versioning of external model APIs. Teacher and evaluator model names are omitted in this manuscript; their roles and inference settings are reported using consistent identifiers.

REFERENCES

Jonathan Z. Bakdash and Laura R. Marusich. Repeated measures correlation. *Frontiers in Psychology*, 8:456, 2017. doi: 10.3389/fpsyg.2017.00456.

Cristian Danescu-Niculescu-Mizil, Moritz Sudhof, Dan Jurafsky, Jure Leskovec, and Christopher Potts. A computational approach to politeness with application to social factors. In *Pro-*

ceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pp. 250–259. Association for Computational Linguistics, 2013. URL https://aclanthology.org/P13-1025/.

Yi Dong, Zhilin Wang, Makesh Sreedhar, Xianchao Wu, and Oleksii Kuchaiev. SteerLM: Attribute conditioned SFT as an (user-steerable) alternative to RLHF. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pp. 11275–11288. Association for Computational Linguistics, 2023. doi: 10.18653/v1/2023.findings-emnlp.754. URL https://aclanthology.org/2023.findings-emnlp.754/.

Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. LoRA: Low-rank adaptation of large language models. In *International Conference on Learning Representations*, 2022. URL https://openreview.net/forum?id=nZeVKeeFYf9.

Joel Jang, Seungone Kim, Bill Yuchen Lin, Yizhong Wang, Jack Hessel, Luke Zettlemoyer, Hannaneh Hajishirzi, Yejin Choi, and Prithviraj Ammanabrolu. Personalized soups: Personalized large language model alignment via post-hoc parameter merging, 2023. URL https://arxiv.org/abs/2310.11564.

Hannah Rose Kirk, Leqi Liu, Fanzhi Zeng, Henry Davidson, Bertie Vidgen, Christopher Summerfield, and Scott A. Hale. PRISM-X: Experiments on personalised fine-tuning with human and simulated users, 2026. URL https://arxiv.org/abs/2605.13307.

Andreas Köpf, Yannic Kilcher, Dimitri von Rütte, Sotiris Anagnostidis, Zhi Rui Tam, Keith Stevens, Abdullah Barhoum, Duc Nguyen, Oliver Stanley, Richárd Nagyfi, Shahul ES, Sameer Suri, David Glushkov, Arnav Dantuluri, Andrew Maguire, Christoph Schuhmann, Huu Nguyen, and Alexander Mattick. OpenAssistant Conversations: Democratizing large language model alignment. In *Advances in Neural Information Processing Systems*, volume 36, 2023. URL https://arxiv.org/abs/2304.07327.

Seongyun Lee, Sue Hyun Park, Seungone Kim, and Minjoon Seo. Aligning to thousands of preferences via system message generalization. In *Advances in Neural Information Processing Systems*, volume 37, 2024. doi: 10.52202/079017-2347. URL https://papers.nips.cc/paper_files/paper/2024/hash/86c9df30129f7663ad4d429b6f80d461-Abstract-Conference.html.

Brian Lester, Rami Al-Rfou, and Noah Constant. The power of scale for parameter-efficient prompt tuning. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pp. 3045–3059. Association for Computational Linguistics, 2021. doi: 10.18653/v1/2021.emnlp-main.243. URL https://aclanthology.org/2021.emnlp-main.243/.

Xiang Lisa Li and Percy Liang. Prefix-tuning: Optimizing continuous prompts for generation. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pp. 4582–4597. Association for Computational Linguistics, 2021. doi: 10.18653/v1/2021.acl-long.353. URL https://aclanthology.org/2021.acl-long.353/.

Xinyu Li, Ruiyang Zhou, Zachary C. Lipton, and Liu Leqi. Personalized language modeling from personalized human feedback, 2024. URL https://arxiv.org/abs/2402.05133.

Renpu Liu, Peng Wang, Donghao Li, Cong Shen, and Jing Yang. A shared low-rank adaptation approach to personalized RLHF. In *Proceedings of The 28th International Conference on Artificial Intelligence and Statistics*, volume 258 of *Proceedings of Machine Learning Research*, pp. 1405–1413. PMLR, 2025. URL https://proceedings.mlr.press/v258/liu25d.html.

Yi Liu, Xiangyu Liu, Xiangrong Zhu, and Wei Hu. Multi-aspect controllable text generation with disentangled counterfactual augmentation. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 9231–9253. Association for Computational Linguistics, 2024. doi: 10.18653/v1/2024.acl-long.500. URL https://aclanthology.org/2024.acl-long.500/.

OpenAI. GPT-5.1: A smarter, more conversational ChatGPT, November 2025. URL https://openai.com/index/gpt-5-1/. Published November 12, 2025. Accessed September 14, 2026.

OpenAI. Characteristics in ChatGPT. OpenAI Help Center, 2026a. URL https://help.openai.com/en/articles/20001038-characteristics-in-chatgpt. Accessed September 14, 2026.

OpenAI. Customizing your ChatGPT personality. OpenAI Help Center, 2026b. URL https://help.openai.com/en/articles/11899719. Accessed September 14, 2026.

Ellie Pavlick and Joel Tetreault. An empirical analysis of formality in online communication. *Transactions of the Association for Computational Linguistics*, 4:61–74, 2016. doi: 10.1162/tacl_a_00083. URL https://aclanthology.org/Q16-1005/.

Jonas Pfeiffer, Aishwarya Kamath, Andreas Rücklé, Kyunghyun Cho, and Iryna Gurevych. AdapterFusion: Non-destructive task composition for transfer learning. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pp. 487–503. Association for Computational Linguistics, 2021. doi: 10.18653/v1/2021.eacl-main.39. URL https://aclanthology.org/2021.eacl-main.39/.

Rafael Rafailov, Archit Sharma, Eric Mitchell, Stefano Ermon, Christopher D. Manning, and Chelsea Finn. Direct preference optimization: Your language model is secretly a reward model, 2023. URL https://arxiv.org/abs/2305.18290.

Alireza Salemi, Sheshera Mysore, Michael Bendersky, and Hamed Zamani. LaMP: When large language models meet personalization. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 7370–7392. Association for Computational Linguistics, 2024. doi: 10.18653/v1/2024.acl-long.399. URL https://aclanthology.org/2024.acl-long.399/.

Ashish Sharma, Adam Miner, David Atkins, and Tim Althoff. A computational approach to understanding empathy expressed in text-based mental health support. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 5263–5276. Association for Computational Linguistics, 2020. doi: 10.18653/v1/2020.emnlp-main.425. URL https://aclanthology.org/2020.emnlp-main.425/.

Zhilin Wang, Yi Dong, Olivier Delalleau, Jiaqi Zeng, Gerald Shen, Daniel Egert, Jimmy J. Zhang, Makesh Narsimhan Sreedhar, and Oleksii Kuchaiev. HelpSteer2: Open-source dataset for training top-performing reward models, 2024. URL https://arxiv.org/abs/2406.08673.

Kexin Yang, Dayiheng Liu, Wenqiang Lei, Baosong Yang, Mingfeng Xue, Boxing Chen, and Jun Xie. Tailor: A soft-prompt-based approach to attribute-based controlled text generation. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 410–427. Association for Computational Linguistics, 2023. doi: 10.18653/v1/2023.acl-long.25. URL https://aclanthology.org/2023.acl-long.25/.

Jianfei Zhang, Jun Bai, Bei Li, Yanmeng Wang, Rumei Li, Chenghua Lin, and Wenge Rong. Disentangling preference representation and text generation for efficient individual preference alignment. In *Proceedings of the 31st International Conference on Computational Linguistics*, pp. 4813–4839. Association for Computational Linguistics, 2025. URL https://aclanthology.org/2025.coling-main.323/.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric P. Xing, Hao Zhang, Joseph E. Gonzalez, and Ion Stoica. Judging LLM-as-a-judge with MT-Bench and Chatbot Arena, 2023. URL https://arxiv.org/abs/2306.05685.

Guide to the supplementary evidence. Sections A and B specify data, training, and deployment state; Section C specifies the training-data and prompt-segmentation controls. Section D gives the scoring prompt, judge validation, and aggregation. Section E explains generation failures; Section F.1 documents the attribute-control analysis used in the main text. Section G reports the online A/B statistics.

A Preference Data and Style Profiles

A.1 Style Attributes and Profile Definition

The four attributes describe practical choices in response style. Deployed style presets motivate their relevance (OpenAI, 2025; 2026b;a); human-annotated corpora provide the precedents below.

Attribute	Level 1	Level 5	Human-annotation precedent
Formality	Casual, conversational	Formal, professional or academic	Register across genres (Pavlick & Tetreault, 2016)
Concision	Elaborate, with examples	Brief and direct	Verbosity in HelpSteer2 (Wang et al., 2024)
Warmth	Detached, factual	Caring and encouraging	Empathy in support (Sharma et al., 2020)
Creativity	Plain, literal	Original perspectives and imagery	Creativity in OpenAssistant (Köpf et al., 2023)

Profiles use five ordered levels in this fixed attribute order, yielding $5^4 = 625$ combinations. Level 3 anchors neutral professional language, moderate elaboration, friendliness, and moderately vivid expression; Section D.1 gives every level anchor. Higher levels specify more of an attribute, not higher quality. The levels are an operational discretization, not psychometrically equal intervals. Attributes can interact in text, motivating the target and non-target analyses; warmth extends beyond supportive empathy, and creativity concerns expression rather than factual invention.

A.2 Preference Pair Generation and Cleaning

Single-attribute data. Generation uses the primary commercial closed-source LLM at temperature 0.7. The teacher receives the training question and a four-coordinate style profile whose non-target coordinates are all 3. One response is generated at the target profile's level on the selected coordinate and another at its contrasting extreme. Targets 1–2 contrast with 5 and targets 4–5 with 1. For target 3, the SHA-1 hash of the profile index, question index, and attribute selects the rejected extreme deterministically. In the final 10,000 neutral-target pairs, 4,961 compare against 1 and 5,039 against 5. The completed dataset contains 50,000 pairs, exactly 20 per profile and attribute, with distinct preferred and rejected strings. No ranking judge selects these labels; the intended generation conditions define the contrast.

Overall style comparisons. Farthest-point sampling starts at the target profile, selects the grid point with largest squared Euclidean distance, then selects the point maximizing its minimum distance to the selected set. The teacher generates one response under each of the three style profiles. Candidates are ranked by their fit to the target style, first using the final fit score and then the sum of four attribute scores. A tied final score is usable only if the attribute-score-sum gap is at least 8. Fewer than two scored candidates, an insufficient gap, or identical selected strings make a pair unusable and prevent it from being retained.

The completed raw corpus contains 12,500 pairs. A consistency filter requires the preferred candidate to be generated under the requested style profile. This removes 22 pairs whose preferred response came from a distant profile. The resulting 12,478 pairs cover all 625 profiles: 614 profiles have 20 pairs and 11 have 15–19. Of retained pairs, 12,392 have a positive final-score gap; 86 have tied final scores but attribute-score-sum gaps of 10–25. This construction provides synthetic style-preference supervision. The consistency filter does not certify factual equivalence or human preference, and teacher generation and fit scoring use the same model family.

Shared data sources. The mixed corpus pools 50,000 attribute pairs and 12,478 overall-style pairs, with each record included once before shuffling. A typical profile has 80 attribute pairs and 20 overall-style pairs (4:1). In Stage 2, FUSE uses only overall-style pairs, while REFINE+MIX uses the 62,478-pair mixed corpus, as do MONOSP and the matched LoRA controls. These shared sources do not imply matched optimizer steps or wall-clock budgets.

A.3 TRAINING AND TEST QUESTIONS

The evaluation unit is a profile–question combination. The same 17 questions are crossed with all 625 requested profiles, giving 10,625 records per method, backbone, and seed. The questions span eight everyday topic/scenario categories (Table 4), including communication, motivation, learning, technology, and decisions. They are open-ended explanatory questions that permit multiple response styles. The 20 training questions use distinct text and identifiers; test categories also occur in training, so this split tests new questions within the covered scenarios rather than held-out-domain transfer.

Table 4: Composition of the actual training and evaluation question sets. Each test question is evaluated under all 625 profiles.

Scenario category	Training questions	Test questions
Daily life	2	3
Social interaction	2	2
Personal development	2	2
Workplace	2	2
Learning	2	2
Technology in everyday life	2	2
Values and meaning	2	2
Decision-making	2	2
Emotion	2	0
Behavior	2	0
Total	20	17

B IMPLEMENTATION AND OPTIMIZATION DETAILS

Table 5: Five tested ECHOALIGN family members and two training controls. The members vary updated parameters and Stage 2 data after attribute learning; the controls stop after Stage 1 or skip it. Every configuration yields one 256-vector prefix. Bold names mark the two highlighted members.

Variant	Initialization	Updated parameters	Training data
FUSE	Attribute-trained blocks	W_p, b_p	Overall style
REFINE	Attribute-trained blocks	E_p^{concat}	Overall style
REFINE+MIX	Attribute-trained blocks	E_p^{concat}	Mixed
JOINT	Attribute-trained blocks	$E_p^{\text{concat}}, W_p, b_p$	Overall style
JOINT+MIX	Attribute-trained blocks	$E_p^{\text{concat}}, W_p, b_p$	Mixed
CONCAT	Attribute-trained blocks	None	—
MONOSP	Neutral E^0	E_p	Mixed

B.1 BACKBONE CONFIGURATIONS AND PROMPT INJECTION

The prefix contains 256 vectors with the backbone's embedding dimension. It replaces a fixed span at the start of the system-message body, preserving the chat delimiters and remaining fixed system text:

`[system delimiter]` E_p (256 vectors) `[remaining fixed text] [end system] [user delimiter]` x `[end user] [assistant delimiter].`

We initialize E^0 from token embeddings of a neutral style description that sets all four attributes to level 3. Every profile starts from the same initialization. The four attribute blocks each contain 64 vectors; their training and reference conditions are specified in Section B.2.

B.2 Training Hyperparameters and Checkpoint Selection

Stage 1 trains four blocks per profile (625×4 instances per backbone and seed), each on 20 attribute pairs. Profiles sharing an attribute level have independent response samples and profile-dependent neutral-target contrasts. Blocks are reused across Stage 2 variants within a profile; cross-profile sharing and unseen combinations are not evaluated.

All soft-prompt strategies use 256 prefix vectors. Attribute initialization trains four 64-vector blocks independently, with a neutral-system-text initialization E^0 and fixed reference E^0. Direct MonoSP training also uses reference E^0; Stage 2 adaptation fixes its reference to a frozen copy of the initial concatenated attribute prefix E_p^{concat}. The DPO coefficient is $\beta = 0.1$ throughout.

Within each training stage, we retain the checkpoint maximizing an exponential moving average of $(r^+ - r^-) + 0.5r^+$, where $r^{\pm} = \beta \log[\pi(y^{\pm} \mid x, E_p)/\pi(y^{\pm} \mid x, E_p^{\text{ref}})]$ are batch-averaged DPO implicit rewards. For Stage 1, substitute $\widetilde{E}_{p,k}$ for E_p and E^0 for E_p^{ref}. The moving average assigns weight 0.3 to the current statistic and 0.7 to its preceding value. Attribute-block checkpoints are selected independently. Selection uses training statistics only, without held-out judge scores. LoRA uses the same statistic and smoothing coefficient, logs every optimizer step, and restores the selected adapter before saving; it is not evaluated using the final training state by default.

B.3 Baseline Configurations and Training Budgets

During Stage 2 adaptation, Refine and Refine+Mix optimize md parameters, Fuse optimizes $d^2 + d$, and Joint and Joint+Mix optimize their sum. MonoSP directly optimizes md parameters; Concat makes no updates after attribute initialization. All strategies can be represented by an effective $m \times d$ prefix at inference. These counts describe trainable parameters, not matched computational budgets: optimization schedules and wall-clock costs can differ. The affine transformation is independent of the query and is eligible for precomputation. Inference requires neither a reference-policy pass nor DPO updates.

B.3.1 Matched Preference Data and Training Budgets

LoRA and MonoSP use the same multiset of 50,000 attribute pairs and 12,478 overall-style pairs, verified by profile, query, chosen and rejected responses and matching input hashes across backbones. Each of 614 profiles has 100 pairs; 11 have 95–99. Table 6 gives nominal budgets at effective batch size four and DPO $\beta = 0.1$. Stage 1 uses 100 epochs per attribute at 10^{-5}; Stage 2 uses 20 epochs at 5×10^{-6}. MonoSP uses 100 mixed-data epochs at 10^{-5}. LoRA uses 20 epochs at 5×10^{-5}, projections q, k, v, o, scaling $\alpha = 2r$, and dropout 0.05. All ranks 4/8/16, with and without profile prompting, are reported. These are the evaluated recipes; total search trials and comparable measured GPU costs are unavailable.

Table 6: Nominal training budget per profile with 80 attribute pairs and 20 pairs comparing overall style. Pair presentations count repeated use across epochs. Shared Stage 1 is counted once for each staged method's training path; these are recipe-derived counts, not measured GPU costs.

Training path	Stage 1 steps	Later steps	Pair presentations
Fuse / Joint / Refine	2,000	100	8,400
Refine+Mix / Joint+Mix	2,000	500	10,000
MonoSP	0	2,500	10,000
LoRA (each rank and prompt condition)	0	500	2,000

For complete profiles, Refine+Mix and MonoSP both use 2,500 steps and 10,000 pair presentations. The former presents attribute and overall-style pairs 120 and 20 times, respectively; the latter presents every pair 100 times. Schedules and reference policies differ, and the 11 incomplete profiles have different totals; equal nominal counts do not imply equal measured compute.

Longer LoRA training schedules. We train PP+LoRA-r4/r16 from initialization for 100 epochs on the 64 Top-64 profiles, 17 questions, and seed 42. Each profile has 100 pairs; 20/50/84/100 epochs correspond to 2,000/5,000/8,400/10,000 presentations. Data, initialization seed, and peak learning

rate are fixed; 10% warmup and linear decay are stretched across the longer schedule. We evaluate the best training-EMA checkpoint available at each budget and the final 100-epoch weights, comparing with the original 20-epoch run and its own schedule. Paired profile-cluster intervals use 10,000 draws and the same collapse-zero and missing-score rules as the main results.

Table 7 shows modest Qwen gains and declines on Llama and Ministral; final weights do not reverse the pattern. Even the best longer-schedule checkpoint across ranks remains at least 0.44 below Qwen FUSE with equal-profile weights and 0.72 below with Top-64 weights. This supports the 20-epoch baseline as a practical budget, without establishing convergence: Qwen-r16's upper interval bound 0.161 exceeds the prespecified 0.15-point material-gain threshold. The main comparison retains the original three-seed, 20-epoch results.

Table 7: Change in style-fit score from the original 20-epoch model to the best training-EMA checkpoint within 100 epochs. Estimates and 95% paired profile-cluster bootstrap intervals use seed 42; Top-64 weights match the main paper.

Backbone	Rank	Equal-profile gain [95% CI]	Top-64 gain [95% CI]
Qwen	4	$+0.063$ $[-0.001, +0.125]$	$+0.096$ $[-0.004, +0.203]$
Qwen	16	$+0.088$ $[+0.012, +0.161]$	$+0.050$ $[-0.098, +0.207]$
Llama	4	-0.250 $[-0.360, -0.148]$	-0.306 $[-0.460, -0.155]$
Llama	16	-0.222 $[-0.354, -0.099]$	-0.224 $[-0.470, +0.008]$
Ministral	4	-0.076 $[-0.125, -0.028]$	-0.083 $[-0.157, -0.001]$
Ministral	16	-0.146 $[-0.236, -0.070]$	-0.318 $[-0.696, -0.015]$

B.3.2 PROFILE STATE AND INFERENCE COST

A query-independent effective prefix $E_p = E_p^{\text{concat}} W_p^\top + \mathbf{1} b_p^\top$ contains $256d$ values and can be materialized once per profile. The current loader instead retains blocks and affine weights ($256d + d^2 + d$ values), including an identity transform for REFINE+MIX. Thus exported prefix size and saved checkpoint size differ (Table 8).

LoRA on language-model q, k, v, o contains $Lr(6d + 2d_{\text{KV}})$ values, with no trained biases. The (d, L, d_{KV}) tuples are $(3584, 28, 512)$, $(4096, 32, 1024)$ and $(4096, 34, 1024)$ for Qwen, Llama and Ministral.[1] Counts exclude any vision-side tensors. Parameters are saved in FP32; BF16 prefix sizes give the backbone-precision payload.

Table 8: Calculated tensor payload per profile, in MiB (2^{20} bytes). The first two rows describe an exported effective prefix; the following rows count FP32 parameter tensors (LoRA: language-model projections). File headers, metadata, optimizer state, and the shared backbone are excluded. The final row separately gives the BF16 KV-cache payload for 256 prefix positions per active request, without cache sharing; it is not persistent profile state.

Representation	Qwen	Llama	Ministral
Effective prefix, FP32	3.50	4.00	4.00
Effective prefix, BF16	1.75	2.00	2.00
Stage-2 checkpoint $(E_p^{\text{concat}}, W_p, b_p)$, FP32	52.51	68.02	68.02
LoRA rank 4, FP32	9.63	13.00	13.81
LoRA rank 8, FP32	19.25	26.00	27.63
LoRA rank 16, FP32	38.50	52.00	55.25
Prefix KV cache per request, BF16	14.00	32.00	34.00

At equal precision, rank-16 LoRA has 11.0/13.0/13.8 times the effective prefix's values on Qwen/Llama/Ministral. The prefix occupies 256 context positions, replacing a designated system-text span, with BF16 KV-cache payload $2Lmd_{\text{KV}}$ values per request. Unmerged LoRA adds projection updates without prefix positions; merging moves updates into backbone weights. These payload calculations do not establish latency or total-memory gains. Both approaches require no reference pass or online DPO. Users with identical profiles may share prefixes or adapters; the unit of comparison is

[1] Architecture sources: Qwen configuration, Meta model specifications, and Ministral configuration.

a distinct profile. Learned embeddings omit target-specific plaintext instructions but imply no privacy guarantee.

B.4 Random Seeds and Generation Settings

Pipeline seeds 42/43/44 generate per-stage, per-profile child seeds through SHA-256, initializing Python, NumPy and PyTorch CPU/CUDA RNGs. Prefix variants share generation tags; question order and batch layout are fixed, with ICL batches of four. Exact regeneration also depends on hardware, libraries and model/tokenizer revisions. Table 9 lists explicit decoding settings. Sampling, temperature, top-p and top-k are inherited from each backbone. Their resolved historical configurations were not captured in the retained manifests, so current defaults cannot reconstruct them.

Table 9: Explicit offline generation settings shared across Qwen, Llama, and Ministral. "Inherited" means not overridden by the generation call.

Setting	Value or handling
Maximum new tokens	1,024 in all Full-625 recipes
Repetition penalty	1.1 for all methods
Minimum new tokens	8 for prefix methods and Base; inherited for PP, ICL, and LoRA
Sampling and temperature	Inherited separately from each loaded backbone
Nucleus / top-k sampling	Inherited separately from each loaded backbone
End-of-sequence tokens	Full stop-token set from the backbone configuration
Chat formatting	Model chat template; assistant generation prompt enabled; thinking disabled where supported
Input truncation	2,048 tokens for prefix/Base, PP, and LoRA; no truncation of ICL demonstrations
Padding and output	Left padding; special tokens removed from decoded answers

The main style-fit judge is the same commercial closed-source LLM, with temperature 0.1 and an output budget of 128 tokens. The target-hidden decimal intensity study instead uses temperature 0 and an output budget of 256 tokens. External-judge and rewrite-generation settings are given in Section D.2.

C Training-Data and Prompt-Segmentation Controls

C.1 Training Configurations

Table 2 crosses two choices: grouping the same 80 attribute preference pairs by attribute or shuffling them into four balanced subsets, and training independent 64-vector blocks or updating the full 256-vector prefix. Each shuffled subset contains 20 pairs, with five from each attribute and the prespecified question balance. Segmented training holds the other blocks at neutral initialization; unsegmented training carries the full prefix forward through successive subsets. Every phase restarts the optimizer and uses the fixed neutral reference. The unsegmented configuration is distinct from the pooled MonoSP recipe in the main comparison.

Each Stage 1 phase uses 100 epochs of 20 pairs, effective batch size four (microbatch one), and learning rate 10^{-5}: 500 optimizer steps per phase and 2,000 in total. Each phase restarts linear warmup/decay with a 0.1 warmup ratio; explicit per-epoch index schedules control exposure. DPO uses $\beta = 0.1$ and maximum sequence length 2,048. Stage 2 uses Fuse on Qwen/Llama and Refine+Mix on Ministral, following the representatives in the main comparison. Fuse learns an affine map for 20 epochs on 20 overall-style pairs at 10^{-5}; Refine+Mix refines embeddings for 20 epochs on 80 attribute plus 20 overall-style pairs at 5×10^{-6}. These give 100 and 500 Stage 2 steps, respectively. Each branch fixes its reference to its own initial prefix: the configurations share reset rules, not identical reference tensors.

All configurations use the same 64 selected profiles and 17 questions, with training/generation seed 42 and partition seed 42. Generation allows up to 1,024 new tokens, and every configuration uses its final checkpoint. Scores average the four attribute-fit ratings with equal profile weights. These

choices differ from the main table's three-seed EMA-selected results and frequency-weighted Top-64 column. The FUSE learning rate here is specific to this control study.

C.2 SCORING AND MISSING RATINGS

Each configuration has $64 \times 17 = 1,088$ planned ratings. Rule-flagged failures receive zero and remain in the denominator; missing ratings remain unknown. The 12 configurations in Table 2 have 13,041 observed ratings out of 13,056. Missing ratings occur in Llama's shuffled segmented and unsegmented configurations (four and eight), and Ministral's by-attribute segmented and shuffled unsegmented configurations (one and two).

For a configuration with observed sum T, m missing ratings, and $N = 1,088$ planned tasks, the full mean lies in $[T/N, (T + 10m)/N]$. For compact presentation, Table 2 reports the midpoint of these bounds after outward rounding to three decimals. These are summary estimates, not recovered ratings or confidence intervals.

D EVALUATION PROTOCOL AND JUDGE VALIDATION

D.1 SCORING RUBRIC AND RESPONSE PARSING

The following prompt is reproduced in full; placeholders specify the question, answer, and target levels in the fixed attribute order. The parser reads the first integer after each attribute and `FINAL`, allowing angle brackets and rejecting echoed ranges. Attributes are clamped to $[0, 10]$ and an explicit final score to $[1, 10]$. If `FINAL` is absent, Python's rounded mean of available attributes is used and clamped to $[1, 10]$; no parseable score is a scoring failure. An explicit final score is retained without recomputing sentence limits. Collapse receives zero later (Section E.2); API and parsing failures remain missing. Validation uses the stricter parser in Section D.2.

System message.

```
You are a strict style-alignment evaluator. Output only the required
↪  single line of scores.
```

User message.

```
You are a strict style evaluator. Score how well the AI response HITS the
target style profile. Do NOT reward correctness or helpfulness -- only
↪  style fit.

Style target scale. Each dimension is a TARGET the response must hit,
↪  scored 1-5.
This is NOT a like/dislike rating and NOT a strength dial: each level
↪  names a DISTINCT
target style. Fit means matching the TARGET level's description;
↪  deviating toward EITHER
side loses points equally -- a response more formal than the target
↪  misses it exactly as
one less formal does.

- formal (target tone):
    1 = very casual: slang, contractions, emojis OK, conversational.
    2 = relaxed but clean: everyday language, minimal slang.
    3 = neutral professional: natural, no slang, no emojis.
    4 = formal: polished, full sentences, professional register.
    5 = highly formal: precise, structured, academic/business register.

- concise (target length) -- HARD CONSTRAINT when the target is 4-5,
↪  takes priority on conflict:
    1 = elaborate: multi-paragraph, examples, supporting detail.
    2 = developed: one full paragraph (4-6 sentences).
    3 = moderate: 2-3 sentences.
    4 = brief: 1-2 sentences.
```

```
    5 = ultra-concise: a single sentence, no preamble, no list, no
    ↪ examples.

- warmth (target emotional stance):
    1 = detached: strictly factual, impersonal, no emotional language.
    2 = polite-neutral: courteous but not personal.
    3 = friendly: approachable, lightly personable.
    4 = warm: empathetic, encouraging, acknowledges feelings.
    5 = deeply caring: emotionally supportive, validating, personal
    ↪ connection.

- creativity (target expression):
    1 = plain: literal, predictable phrasing, no figures of speech.
    2 = mostly plain with occasional fresh wording.
    3 = balanced: some vivid phrasing.
    4 = creative: metaphors, fresh angles.
    5 = highly inventive: original analogies, surprising framing, vivid
    ↪ imagery.

CONFLICT RULE: if 'concise' target is 4-5, it OVERRIDES the others --
↪ keep the response
within the length limit even if that means expressing warmth/creativity
↪ briefly.
The response must hit ALL four targets simultaneously, with length as the
↪ gate.

Target profile: formal=<FORMAL_LEVEL>, concise=<CONCISE_LEVEL>,
↪ warmth=<WARMTH_LEVEL>, creativity=<CREATIVITY_LEVEL>

Question: <QUESTION>

AI response: <ANSWER>

Step 1 -- For EACH dimension, rate the response's fit on a 0-10
↪ sub-scale:
  10 = exactly matches the target level
   7 = close, minor deviation
   4 = noticeably off (e.g. target concise=5 but response is several
   ↪ sentences)
   0 = opposite of target (e.g. target concise=1 but the response is a
   ↪ single
       sentence, or target warmth=1 but the response is emotionally
       ↪ effusive -> 0-2)
Sub-scores judge style fit ONLY: the answer's quality, correctness or
↪ helpfulness
must not influence any sub-score.
Step 2 -- Apply the CONFLICT RULE: if the concise target is 4-5 and the
↪ response
  exceeds the length limit, cap the FINAL score at 4 regardless of other
  ↪ dims.
Step 3 -- FINAL = round(average of the four sub-scores), then apply the
↪ cap.

Output EXACTLY this single line, nothing else:
formal:<0-10> concise:<0-10> warmth:<0-10> creativity:<0-10> FINAL:<1-10>
```

D.2 Judge Validity and Scoring Variability

Construct checks with human annotations. The validation suite evaluates the shared target-conditioned scoring function on HelpSteer2 verbosity (Wang et al., 2024), PT16 formality (Pavlick & Tetreault, 2016), OpenAssistant creativity (Köpf et al., 2023), politeness (Danescu-Niculescu-Mizil et al., 2013), and EPITOME empathy (Sharma et al., 2020). These are existing human annotations, not newly collected satisfaction ratings for our methods. The same text is scored under high and low targets. For HelpSteer2 same-question pairs with human verbosity-label gaps of at least two,

the higher-concision target should prefer the less verbose answer and the lower-concision target the more verbose answer. Of 240/238 complete high/low-target pairs, 57/26 receive tied fit scores. Agreement is $182/183 = 99.45\%$ and $182/212 = 85.85\%$ excluding ties; counting ties as non-success gives $182/240 = 75.83\%$ and $182/238 = 76.47\%$. These results use human verbosity labels directly. On OpenAssistant, creativity-fit scores correlate 0.399 with human creativity and 0.104 with human quality labels (Spearman), supporting attribute sensitivity. Human anchors also reveal limited separation: the high-creativity curve is not strictly monotone, and moderate versus strong empathy is only weakly distinguished. Politeness and empathy cover related aspects of warmth rather than the identical construct.

Cross-judge protocol and prespecified comparisons. A frozen plan (seed 20260917) selects two profiles per test question without replacement or consulting scores. The same 34 conditions cross 17 methods, three backbones and seeds 42–44: 5,202 records from 153 files. The 58 existing collapse labels retain zero. Identical inputs are judged once and reused; duplicates are not independent ratings.

Judges receive question, answer and profile but no method/backbone identity, using Section D.1. The primary commercial LLM uses temperature 0.1 and 128 output tokens. External judges are GPT-4o (`gpt-4o-2024-08-06`, A), Gemini 2.5 Pro (`gemini-2.5-pro`, B), and Qwen3-235B-A22B (`qwen3-235b-a22b`, C), all at 0.1 with 128/4,096/4,096 tokens. Inputs have at most three attempts. The primary panel averages all three valid external scores, excluding the primary LLM; sensitivity analysis averages A and B on their available common support.

Validation requires exactly one integer per attribute in $[0, 10]$ and one `FINAL` in $[1, 10]$, rejecting missing, duplicate or invalid values. Unlike the main parser, it rejects rather than clamps `FINAL:0`; this affects 148 C tasks. Of 5,144 noncollapsed records, 4,939 have all four ratings. Missing counts are 199 for C, six for the primary LLM, and three for A, with overlap. Missing records score lower under other judges; dropping C does not restore complete coverage. The original main-table scores remain unchanged.

The 15 prespecified contrasts compare Qwen/Llama FUSE and Ministral REFINE+MIX with three LoRA ranks, MONOSP and ProfilePrompt. We average paired differences within seed, then seeds equally. Intervals resample whole questions with their profiles and seeds (2,000 draws for three judges, 1,000 for two); they are individual, unadjusted 95% intervals on sampled conditions, not Full-625 or Top-64 rescoring. All mean differences are positive; two sensitivity intervals cross zero (Ministral versus LoRA-r16 and MONOSP). Method-rank correlations on panel-specific common support are 0.965–0.982 for three judges, versus 0.819–0.968 for individual judges (Tables 10 and 11).

Table 10: Cross-judge validation on the sampled conditions. Rank columns give Spearman correlation with the primary LLM across 17 methods on common records within each panel. Interval columns count the five prespecified method comparisons per backbone whose individual 95% intervals are above zero. The three-judge panel averages Evaluators A–C; the two-judge panel excludes C.

	Rank correlation		Positive 95% intervals	
Backbone	Three judges	Two judges	Three judges	Two judges
Qwen	0.973	0.980	5/5	5/5
Llama	0.965	0.960	5/5	5/5
Ministral	0.982	0.977	5/5	3/5

Rescoring the original main-table inputs. After matching exact answer bytes and scoring conditions, 5,137 records can be compared with their original primary-LLM score; the exclusions are 58 generation failures, six failed new primary-LLM ratings, and one original refusal record. Final scores are identical for 83.9% of comparable records, within one point for 97.4%, and have Spearman correlation 0.9579. Mean absolute change is 0.192 and mean signed change is -0.0125 on the 0–10 scale. This measures repeatability, not accuracy; the original ratings remain the basis of the main results.

Five-target matching against blinded reference labels. Evaluator A generates five rewrites for each of four attributes on 30 new questions: 600 answers in 120 groups (temperature 0.7, 4,096 tokens).

Table 11: Prespecified sampled-condition style-fit differences, with individual 95% question-cluster bootstrap intervals. Each column uses the records available to that judge or complete panel for each paired comparison; supports need not coincide across columns.

Comparison	Primary LLM	Three-judge panel	Two-judge panel
Qwen: FUSE minus comparator			
PP+LoRA-r4	$+1.25$ $[+0.77, +1.73]$	$+0.99$ $[+0.38, +1.62]$	$+1.41$ $[+0.76, +2.04]$
PP+LoRA-r8	$+1.24$ $[+0.79, +1.69]$	$+1.00$ $[+0.38, +1.64]$	$+1.36$ $[+0.67, +2.06]$
PP+LoRA-r16	$+1.24$ $[+0.74, +1.75]$	$+1.04$ $[+0.41, +1.69]$	$+1.41$ $[+0.72, +2.09]$
MONOSP	$+0.83$ $[+0.44, +1.23]$	$+0.77$ $[+0.31, +1.26]$	$+1.10$ $[+0.51, +1.68]$
ProfilePrompt	$+1.77$ $[+1.27, +2.25]$	$+1.75$ $[+1.16, +2.32]$	$+2.52$ $[+1.87, +3.16]$
Llama: FUSE minus comparator			
PP+LoRA-r4	$+0.53$ $[+0.12, +0.93]$	$+0.51$ $[+0.02, +0.99]$	$+0.90$ $[+0.38, +1.46]$
PP+LoRA-r8	$+0.66$ $[+0.23, +1.08]$	$+0.54$ $[+0.01, +1.09]$	$+0.94$ $[+0.27, +1.65]$
PP+LoRA-r16	$+0.63$ $[+0.19, +1.08]$	$+0.67$ $[+0.20, +1.14]$	$+0.96$ $[+0.39, +1.58]$
MONOSP	$+0.42$ $[0.00, +0.81]$	$+0.55$ $[+0.17, +0.93]$	$+0.63$ $[+0.15, +1.15]$
ProfilePrompt	$+1.21$ $[+0.81, +1.61]$	$+1.62$ $[+1.12, +2.15]$	$+1.87$ $[+1.23, +2.51]$
Ministral: REFINE+MIX minus comparator			
PP+LoRA-r4	$+0.26$ $[-0.06, +0.64]$	$+0.53$ $[+0.13, +0.90]$	$+0.37$ $[+0.02, +0.70]$
PP+LoRA-r8	$+0.33$ $[0.00, +0.71]$	$+0.53$ $[+0.18, +0.91]$	$+0.44$ $[+0.12, +0.75]$
PP+LoRA-r16	$+0.20$ $[-0.11, +0.54]$	$+0.38$ $[+0.02, +0.75]$	$+0.24$ $[-0.09, +0.55]$
MONOSP	$+0.23$ $[+0.03, +0.46]$	$+0.34$ $[+0.03, +0.59]$	$+0.32$ $[-0.06, +0.66]$
ProfilePrompt	$+1.09$ $[+0.68, +1.52]$	$+1.31$ $[+0.92, +1.70]$	$+1.16$ $[+0.76, +1.66]$

A/B/C independently rate achieved intensity and core-information preservation, with shuffled answer order and no generation targets or primary scores (temperature 0.1, 4,096 tokens). The prespecified subset requires unanimous content preservation and intensity labels spanning at most one level. Their median defines the reference; 494 answers remain, as detailed in Table 12.

The primary judge scores all 600 answers at all five targets (other attributes fixed at 3), giving 3,000 ratings. We analyze the selected attribute subscore, average within question and then questions equally, and resample 30 question clusters. For two same-question rewrites with different reference levels, success means the matching answer scores higher; ties remain a separate outcome. Adjacent-level success is 41.3–51.3%, not directly comparable to a binary chance baseline. Seven of 18 nonempty reference rows improve at some step away from the matched target, so the scorer is not uniformly calibrated across five levels.

Table 12: Target-matching outcomes with clear reference labels. Percentages are question-weighted and retain ties. "Adjacent positive" restricts pairs to a one-level reference gap.

Attribute	Answers	Positive (%)	Tie (%)	Reversal (%)	Adjacent positive (%)
Formality	149	68.7	24.7	6.6	49.3
Concision	72	75.4	16.2	8.4	51.3
Warmth	144	76.7	18.4	4.9	41.7
Creativity	129	56.8	26.0	17.2	41.3

Reference coverage omits concision 1 and warmth 5; creativity 5 has one answer. Reference-judge exact agreement is only 12.7–34.7% for concision. Excluding the generator's model family and requiring exact agreement leaves 364 answers, changing both judges and selection; this does not isolate self-evaluation bias. Reference consensus is a model proxy, and other attributes change in rewrites. These checks establish neither human calibration nor isolated causal sensitivity.

Repeated scoring and input sensitivity. On 100 fixed inputs scored five times each, the mean within-input sample SD of the final 0–10 score is 0.123; 79 inputs have identical scores on all five repeats. The 90th-percentile SD is 0.447. Rewording the question and reversing profile-field order yield mean absolute deviations of 0.246 and 0.394 from the repeated-score mean, respectively. We keep the presentation format fixed across methods. These checks quantify within-judge repeatability and input sensitivity, not independence of main-table records or human agreement. They are separate from the target-hidden decimal-intensity repeats in Section F.1.

Connection to human choices. On public Arena pairs with explicit style requests (Zheng et al., 2023), the fit scorer predicts the human winner in 59.9% of 302 comparisons with non-tied fit scores. At score gaps of at least five, agreement is 82.9% on 41 pairs. A generic quality judge achieves 70.2% on the same 302 comparisons, so style fit is an informative but incomplete component of overall preference. The decided sample has 220 concision, 66 creativity, and only eight pairs each for formality and warmth. A complementary HelpSteer2 subset with length requests gives 67.8% directional agreement between fit and human helpfulness differences on 379 pairs. These checks concern associations between style fit and overall preference in existing datasets. The randomized deployment in Section 4.6 supplies separate evidence from real-user clicks on our system's outputs.

D.3 Profile Weighting and Statistical Procedures

Full-625 crosses 625 profiles with 17 held-out questions, giving 10,625 records per method and seed. Top-64 selects the 64 most frequent of 226 observed profile types before confidence filtering. Excluding 1,039 low-confidence users leaves 13,178 medium/high users; the selected profiles cover 12,488 (94.76%). Answers are unchanged: each method and seed still contributes $64 \times 17 = 1,088$ Top-64 records. The analysis reweights trained profiles rather than testing unseen ones.

Aggregation and uncertainty. For a given backbone and method, let V_{sp} be the set of questions for style profile S_p and seed s with either a valid judge score or an assigned zero for collapse. Let z_{spq} denote the resulting score. We report the unweighted Full-625 record mean U_s and the confidence-filtered, frequency-weighted Top-64 mean W_s:

$$U_s = \frac{\sum_{p\in\mathcal{P}_{625}} \sum_{q\in V_{sp}} z_{spq}}{\sum_{p\in\mathcal{P}_{625}} |V_{sp}|}, \qquad W_s = \sum_{p\in\mathcal{P}_{64}} \frac{n_p}{12{,}488} \frac{\sum_{q\in V_{sp}} z_{spq}}{|V_{sp}|}. \tag{2}$$

Here $\mathcal{P}_{625}$ and $\mathcal{P}_{64}$ index the two profile sets, and n_p counts retained users with profile S_p. Confidence determines inclusion, not a probability weight. Full-625 averages valid/zero records; Top-64 first averages questions within each profile, then weights users. For example, $(2, 4, 3, 2)$ has weight $1,679/12,488 = 13.44\%$, versus $1/64 = 1.56\%$ under equal-profile weighting. Every selected profile has valid records. We report the mean and sample SD across seeds 42/43/44; relative gains divide the unrounded mean difference by the baseline mean. Seed variation includes training, generation and judging. Some variants share Stage 1 checkpoints; Base reuses 17 answers across profiles, whose mapping is inferred from grid IDs and remains provisional. Individual records are therefore not independent replications. The grouping study uses the separate denominator and bounds in Section C.

E Generation Failure Audit

E.1 Generation Collapse Across Seeds and Backbones

Collapse receives zero in the style-fit means. The counts below use all 10,625 Full-625 records per method and seed; they show rare detected Llama failures and persistent Ministral failures. Qwen JOINT and FUSE have no flagged outputs and outperform prompting, retrieval and LoRA in every seed on both Full-625 and Top-64.

Backbone	Method	Seed 42	Seed 43	Seed 44
Qwen	JOINT	0	0	0
Qwen	FUSE	0	0	0
Llama	JOINT	5	9	5
Llama	FUSE	16	8	14
Ministral	JOINT	2,094	1,946	2,001
Ministral	FUSE	2,477	2,435	2,449

Llama's frequent-profile failure. Profile $(2, 4, 3, 3)$ represents 879 users, or 7.04% of Top-64 weight versus 0.16% of the grid. Its JOINT scores are 5.688/1.647/1.294 and FUSE scores 5.706/1.412/1.647 in seeds 42/43/44. In seeds 43/44, answers contain style meta-instructions, prompt recitations or fragments; the judge assigns 1–2 while the detector flags at most 3/17 answers. This

profile accounts for more than the entire seed-43 Top-64 decline and 93% of the seed-44 decline for JOINT (92%/73% for FUSE). The corresponding Top-64 means are 5.969/5.739/5.636 and 6.113/5.785/5.719. Seed-42 FUSE and JOINT exceed MONOSP (5.845) and PP+LoRA-r4 (5.747). REFINE also deteriorates on this profile (5.824/2.765/1.353), while CONCAT, both Mix variants and MONOSP remain between 4.1 and 4.8.

On Ministral, JOINT/FUSE collapse on 18.3–19.7%/22.9–23.3% of grid records and 21.8–25.6%/24.0–29.3% of Top-64 records per seed. REFINE+MIX leads style fit in every seed with only 0.21% three-seed grid collapse. On $(2, 4, 3, 3)$, JOINT/FUSE loop on all 17 questions in seeds 43/44, whereas the other staged variants give substantive responses. These patterns localize failures without identifying their cause; three-seed means remain the primary summaries.

E.2 COLLAPSE DETECTION AND MANUAL REVIEW

The detector uses answer text, not style scores or backbone identity; missing/non-string answers are not automatically empty. It checks:

1. **Empty/symbol-only:** strip Unicode format characters and whitespace; flag empty text or text without a Unicode letter/digit.
2. **Placeholders:** case-folded word tokens contain only repeated response/answer/user/assistant labels (optionally "the"), or the complete acknowledgments got it/gotcha/ok/okay/understood.
3. **Literal loops:** a 1–100-character block repeats at least ten times over ≥ 200 characters and half the whitespace-normalized answer. The prefilter requires ≥ 200 characters and either a zlib UTF-8 compression ratio below 0.50 or > 40 whitespace tokens with distinct-token fraction below 0.25 in the first 200.
4. **Low-diversity loops:** ≥ 80 tokens, distinct fraction below 0.25 overall and in the first or last 200 tokens, plus normalized compression ratio below 0.35. Token identity is case-sensitive.

Compression, shortness, low scores or multilingual text alone do not trigger zeroing; unflagged candidates retain valid scores. Archived review contains 204 entries for 179 answer hashes: 136 collapse and 43 keep. Collapse labels apply to identical strings; keep labels do not override deterministic rules. Review adds 169 zero-scored records (70 Llama, 99 Ministral, none Qwen) across the 153 cells. It is not an exhaustive blinded review, so detected rates remain lower bounds.

E.3 FAILURE RATES AND MISSING-SCORE HANDLING

We use the stored, parsed final score produced by the shared rubric; Section D.1 gives the full scoring prompt and parser. Answers identified as collapsed by deterministic rules (e.g., empty, placeholder, or repetitive output) or existing manual-review labels receive zero and remain in the denominator. For the main-table comparisons, we exclude only answers that are not labeled as collapsed but have missing or invalid judge scores. Across the 153 backbone–method–seed combinations, 246 such records are excluded on the full grid and nine on Top-64 (one Llama JOINT, seven Ministral JOINT, and one Ministral JOINT+MIX record). Rule-detected collapse rates (with existing review corrections) always use all available records as the denominator. The separate grouping controls retain the full planned denominator and bound missing ratings as described in Section C; the exclusions in this subsection do not apply to that study.

E.4 QUALITATIVE EXAMPLES AND FAILURE CASES

Table 13 uses the held-out satisfaction question from the Qwen seed-42 generation files. The profiles differ only in concision; Base reuses its profile-independent answer. Excerpts preserve the first two complete sentences (or the entire answer if shorter), with list and paragraph spacing normalized. The brief FUSE answer is complete; the other answers continue after the quoted opening.

Base gives the same answer to both profiles, and ProfilePrompt still elaborates under the brief profile. FUSE changes from an explanation to a single sentence. These excerpts illustrate style changes, not overall answer quality.

Table 13: First two sentences, or the full answer when shorter, from Qwen seed 42. Profiles A $(3, 1, 3, 3)$ and B $(3, 5, 3, 3)$ differ only in requested concision. Profile order is formality, concision, warmth, creativity. Excerpts are selected illustrations; full responses and source records are described in Section E.4.

Question	Why do people feel satisfied after completing a task?
Base (A, B)	People often feel satisfied after completing a task for several reasons: 1. **Achievement of Goals**: Completing a task can give individuals a sense of accomplishment and satisfaction because it signifies that they have achieved something they set out to do. This achievement reinforces the feeling of being capable and competent.
PP (A)	People often feel satisfied after completing a task because achieving goals triggers the release of dopamine in the brain, enhancing feelings of pleasure and accomplishment. This satisfaction reinforces positive behaviors and motivates individuals to tackle new challenges.
PP (B)	People feel satisfied after completing a task because it signifies progress and achievement. This fulfillment stems from meeting goals and can boost self-esteem and motivation for future tasks.
FUSE (A)	People often feel a sense of satisfaction after completing a task due to the complex interplay of psychological and physiological factors. When we set goals for ourselves, whether it's as simple as finishing a household chore or as significant as achieving a long-term career objective, our brain engages in a process that involves both anticipation and reward.
FUSE (B)	Completing a task often brings satisfaction as it confirms your abilities and contributes to a sense of accomplishment.

For a concrete example of the data construction in Figure 2a, the training question "How does trust develop between individuals?" and target profile $(1, 1, 1, 1)$ yield a Stage 1 creativity contrast between $(3, 3, 3, 1)$ and $(3, 3, 3, 5)$, and a Stage 2 comparison of the complete target style with a distant profile.

F SINGLE-ATTRIBUTE CONTROL AND CROSS-ATTRIBUTE INTERFERENCE

F.1 REPEATED-MEASURES CORRELATION ANALYSIS

Table 14 summarizes ten methods on all three backbones. We sample five questions (IDs 101, 102, 108, 111, 112) and four backgrounds per changed attribute, uniformly without replacement from 125 possibilities (seed 20260911). Each trajectory fixes a question and three attribute levels while varying the fourth from 1 to 5. Every profile has separate trained parameters, evaluated under seeds 42/43/44. The target-hidden primary judge rates realized intensities in 0.1-unit increments on a 1–5 scale, at temperature 0 and a 256-token budget. All methods share these questions, profiles and trajectories. We use available initial ratings; repeat judgments add no observations, and generation failures are excluded from control but remain zero in style fit. Figure 5 shows full signed matrices for the four comparisons emphasized in the main analysis.

Target response and coupling. Qwen/Llama FUSE track targets similarly to LoRA but with larger slopes; Ministral REFINE+MIX improves both. Their coupling exceeds LoRA and MONOSP (Table 14). Increasing warmth is associated with more creativity, and increasing formality or concision with less warmth in every seed; median matrix-entry SD is 0.04 (maximum 0.16). Low non-target correlation means weak linear association, not unchanged attributes.

Uneven separation between levels. High correlation across levels 1–5 does not mean that adjacent levels are equally distinguishable (Figure 4). Qwen FUSE increases target intensity by 0.68 points per step within levels 2–4, versus 0.26 over the outer steps ($1 \to 2$, $4 \to 5$). Llama FUSE gives 0.88 versus 0.24, and Ministral REFINE+MIX gives 0.67 versus 0.11. Target intensity rises in 75–83% of middle-step pairs, compared with 54–57% of outer-step pairs. LoRA shows the same pattern: 0.58 versus 0.06 on Qwen, 0.58 versus 0.20 on Llama, and 0.48 versus 0.19 on Ministral. This shared pattern suggests that the training contrasts or rating design may contribute to the weaker separation at the extremes.

Table 14: Attribute control as repeated-measures correlation for ten methods, three-seed mean ± sample SD. Tracking is the mean diagonal $r_{\mathrm{rm}}(k \to k)$; coupling is the mean absolute off-diagonal $|r_{\mathrm{rm}}(k \to j)|$, $j \neq k$; the slopes are the common within-trajectory slopes in intensity units per requested level, for the requested attribute and (in absolute value) for the other attributes.

Method	Tracking r	Coupling $\lvert r\rvert$	Target slope	Off-target |slope|
Qwen				
REFINE+MIX	0.83±0.02	0.41±0.01	0.43±0.01	0.13±0.01
FUSE	0.82±0.02	0.51±0.00	0.51±0.02	0.19±0.00
JOINT	0.83±0.01	0.50±0.03	0.49±0.02	0.18±0.01
REFINE	0.78±0.01	0.38±0.01	0.41±0.01	0.14±0.01
CONCAT	0.61±0.01	0.29±0.04	0.24±0.01	0.10±0.01
MONOSP	0.80±0.02	0.33±0.01	0.36±0.02	0.07±0.00
PP+LoRA-r16	0.81±0.02	0.30±0.02	0.37±0.01	0.07±0.01
ProfilePrompt	0.40±0.01	0.26±0.02	0.07±0.01	0.04±0.00
ICL-RAG	0.72±0.03	0.28±0.02	0.26±0.01	0.07±0.00
ICL-All	0.33±0.02	0.22±0.03	0.14±0.01	0.06±0.01
Llama				
REFINE+MIX	0.79±0.01	0.38±0.01	0.42±0.01	0.12±0.01
FUSE	0.84±0.01	0.41±0.02	0.61±0.02	0.17±0.00
JOINT	0.86±0.01	0.46±0.04	0.63±0.02	0.19±0.02
REFINE	0.84±0.01	0.41±0.02	0.57±0.02	0.15±0.00
CONCAT	0.75±0.01	0.40±0.02	0.43±0.00	0.14±0.01
MONOSP	0.82±0.01	0.30±0.02	0.44±0.00	0.08±0.00
PP+LoRA-r16	0.85±0.02	0.30±0.06	0.43±0.02	0.07±0.01
ProfilePrompt	0.62±0.02	0.41±0.00	0.24±0.01	0.11±0.00
ICL-RAG	0.71±0.02	0.35±0.01	0.33±0.00	0.10±0.00
ICL-All	0.34±0.01	0.24±0.00	0.13±0.01	0.07±0.00
Ministral				
REFINE+MIX	0.82±0.01	0.48±0.02	0.45±0.01	0.14±0.01
FUSE	0.85±0.03	0.50±0.06	0.59±0.03	0.18±0.01
JOINT	0.82±0.02	0.52±0.03	0.59±0.02	0.21±0.01
REFINE	0.84±0.01	0.55±0.01	0.55±0.00	0.20±0.01
CONCAT	0.72±0.02	0.55±0.02	0.39±0.01	0.20±0.01
MONOSP	0.80±0.01	0.26±0.02	0.34±0.01	0.05±0.00
PP+LoRA-r16	0.73±0.01	0.25±0.01	0.36±0.00	0.06±0.00
ProfilePrompt	0.67±0.00	0.64±0.00	0.33±0.00	0.23±0.00
ICL-RAG	0.86±0.00	0.54±0.00	0.46±0.00	0.17±0.00
ICL-All	0.86±0.00	0.48±0.00	0.50±0.00	0.14±0.00

Repeatability and joint preservation. In the original four-method subset (REFINE+MIX, JOINT, ProfilePrompt, PP+LoRA-r16), only 4.5–7.6% of adjacent pairs both increase the target and keep all three other attributes within 0.25 units (23.8–36.9% within 0.5). Each method uses its available pairs; Ministral JOINT has only 687/960 valid pairs, and directional rankings change on common support. The 351 repeated ratings have mean absolute disagreements of 0.011–0.021 and exact agreement of 91.7–95.4%. These repeats assess judge consistency, not human calibration; the preservation statistics use the same pair for all four attributes.

Within-trajectory estimates. Fix a backbone, method, seed s, and requested attribute k. A trajectory g fixes the question and the other three requested levels. For each scorable level i, let $\ell_{g,i}$ be the requested level and $r_{g,i,j}$ the judged intensity of attribute j. We retain trajectories with at least two scorable levels and center both variables over their available levels: $\widetilde{\ell}_{g,i} = \ell_{g,i} - \bar{\ell}_g$ and

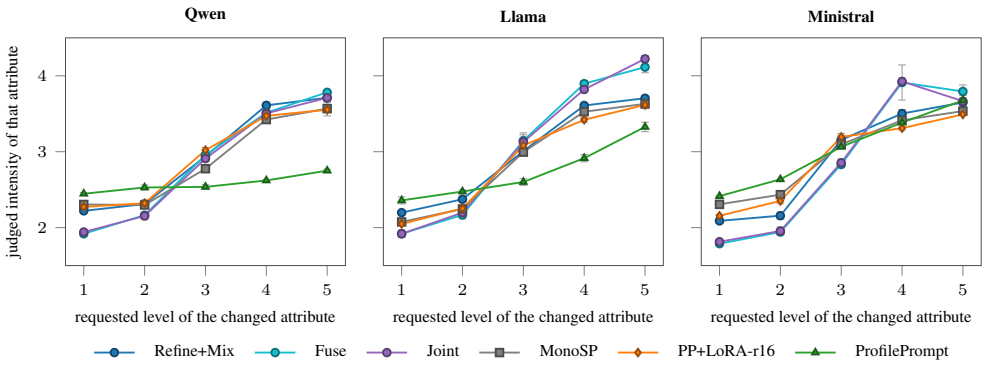

Figure 4: Judged intensity of the changed attribute at each requested level, averaged over the four attributes and their sampled backgrounds; error bars show sample SD across three seeds. For the trained methods, the response changes more between levels 2 and 4 than over the outer steps from 1 to 2 and from 4 to 5.

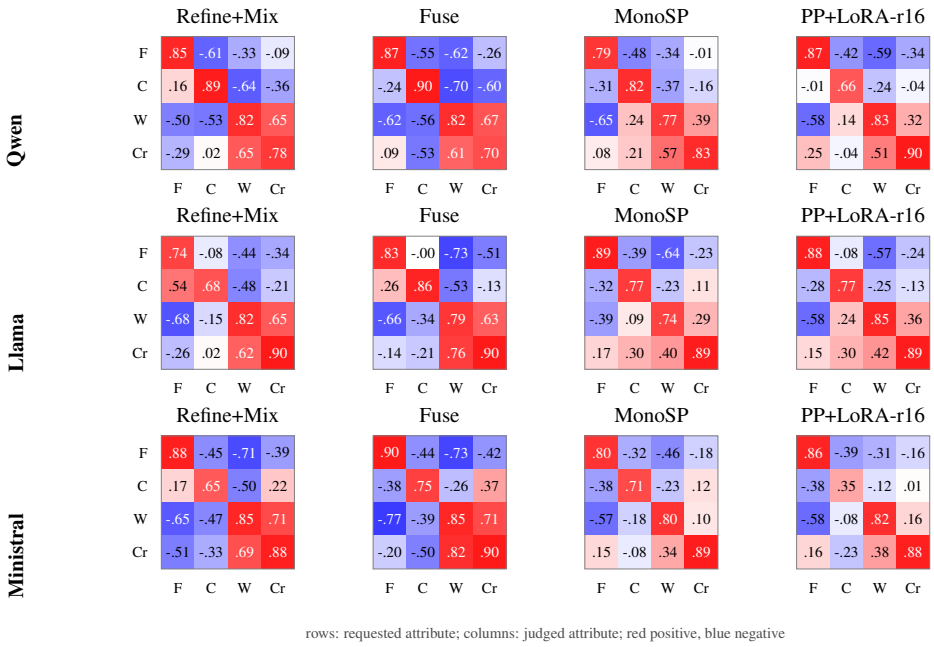

Figure 5: Signed repeated-measures correlation matrices $r_{\mathrm{rm}}(k \to j)$ for REFINE+MIX, FUSE, MONOSP, and PP+LoRA-r16, three-seed means; rows are the requested attribute and columns the judged attribute (F formality, C concision, W warmth, Cr creativity). Red is positive and blue negative.

$\widetilde{r}_{g,i,j} = r_{g,i,j} - \bar{r}_{g,j}$. The repeated-measures correlation and common slope are

$$r_{\mathrm{rm},s}(k \to j) = \frac{\sum_{g,i} \widetilde{\ell}_{g,i} \widetilde{r}_{g,i,j}}{\sqrt{\left(\sum_{g,i} \widetilde{\ell}^2_{g,i}\right)\left(\sum_{g,i} \widetilde{r}^2_{g,i,j}\right)}},$$
$$\beta_s(k \to j) = \frac{\sum_{g,i} \widetilde{\ell}_{g,i} \widetilde{r}_{g,i,j}}{\sum_{g,i} \widetilde{\ell}^2_{g,i}}. \tag{3}$$

The estimates use a separate intercept per trajectory (Bakdash & Marusich, 2017); slopes give intensity units per requested level, and correlations are undefined at zero pooled within-trajectory variance. Within each seed, tracking averages the four diagonal correlations and coupling the twelve absolute off-diagonal correlations; slopes are aggregated analogously. We report three-seed means and sample SDs. Question-cluster intervals use 2,000 draws with all trajectories and levels retained together, conditional on five questions and fixed backgrounds; parametric correlation p-values are not used.

Robustness checks and adjacent steps. Replacing values by within-trajectory ranks (average ranks for ties) gives tracking 0.85/0.84/0.86 for Qwen/Llama FUSE and Ministral REFINE+MIX. Fisher-z averaging of trajectory correlations gives 0.89/0.91/0.88, using at least four scorable levels, nonzero variance, and clipping to $\pm(1 - 10^{-6})$ before transformation. These alternative weightings need not equal pooled estimates. Adjacent-step means and positive-change proportions use pairs with two scorable answers, separately for middle ($2 \to 3$, $3 \to 4$) and outer ($1 \to 2$, $4 \to 5$) steps, averaging attributes within each seed before summarizing seeds. Correlations and slopes were independently recomputed from stored ratings.

G ONLINE DEPLOYMENT: AGGREGATE A/B RESULTS

Table 15: Aggregate interim dashboard results for the randomized deployment A/B test. Baseline and REFINE+MIX correspond to the two fixed experiment arms; metrics cover each entire bucket, including the treatment's baseline fallback. PV counts dispatches; UV counts distinct recipients across the reported evaluation. Relative changes and test z statistics are reported by the platform; nominal two-sided p-values use $p = 2\Phi(-|z|)$, where Φ is the standard-normal CDF.

Metric	Baseline	REFINE+MIX policy	Rel. change	Reported z	Nominal p
PV_CTR	1.3390%	1.4259%	+6.49%	+4.41	1.03×10^{-5}
UV_CTR	2.3804%	2.5463%	+6.97%	+4.52	6.18×10^{-6}

Stratified PV_CTR risk ratio: 1.0610; 95% CI $[1.0196, 1.1041]$, $z = 2.91$, nominal $p \approx 0.0036$. This interval is distinct from the unstratified PV_CTR summary above.

Evaluation scope. Baseline uses explicit style instructions with the original Qwen3.5-35B-A3B; the treatment uses REFINE+MIX for eligible users and the baseline route otherwise. All rates cover the entire respective bucket, as in Section 4.6.

Pooled and stratified tests. The pooled PV_CTR increase is 0.0869 percentage points, or 6.49% relative to baseline ($z = 4.41$, nominal two-sided $p \approx 1.03 \times 10^{-5}$); UV_CTR increases by 6.97% ($z = 4.52$, $p \approx 6.18 \times 10^{-6}$). The platform's stratified Mantel–Haenszel risk ratio is 1.0610 ($z = 2.91$, nominal $p \approx 0.0036$), with 95% interval $[1.0196, 1.1041]$, corresponding to a relative increase of 1.96–10.41%. These values use rounded platform statistics and are unadjusted for repeated looks or multiple outcomes. They describe the whole-bucket routing policy, not a covered-user or component-specific effect.

Heterogeneity and uncertainty. Stratum-level relative CTR differences range from -9.68% to $+15.01\%$. The aggregate positive effect therefore does not imply uniform improvement across strata. Nominal tests include both positive and negative significant differences and are unadjusted for multiple comparisons or repeated monitoring. Recipients can recur across strata, so these comparisons are not independent replications.

Exposure variation and reconstruction limits. Treatment dispatch volume is 0.27% higher in aggregate; relative dispatch-volume differences across strata range from -3.80% to $+11.25\%$. The platform summary does not establish the cause of this imbalance. Aggregate CTR is total clicks divided by total dispatches, not the arithmetic mean of stratum-level CTRs or relative changes. We retain both pooled and stratified estimates in Table 15. The available dashboard summaries do not include stratum-level exposure and click counts or user-level records. We therefore cannot independently recompute the pooled z statistics, Mantel–Haenszel risk ratio, or confidence interval from these summaries, nor assess user-clustered uncertainty. Displayed CTRs are rounded to four decimal places in percent units; we preserve the reported changes, which are compatible with that rounding, rather than overwrite them using ratios of rounded rates.