Preserving General Capability in Personality-Aligned LLMs via Orthogonal Gradient Projection

Yangfu Zhu[1], **Linyue Wei**[1], **Nianwen Ning**[2], **Pengpeng Zhou**[3], **Chenguang Song**[4], **Yuandong Wang**[1], **Zhenzhou Shao**[1*]
[1]Capital Normal University, [2]Henan Univeristy,
[3]China Academy of Civil Aviation Science and Technology,
[4]Anhui Science and Technology University
zhuyangfu,2251002008@cnu.edu.cn

Abstract

Personality alignment tailors Large Language Models (LLMs) for personalized interactions. Most existing studies focus on prompting, post-training, or representation steering to elicit trait expression, but they often compromise the general capabilities like reasoning and truthfulness. In this work, we empirically verified the existence of the **P**ersonality **A**lignment **T**ax **(PAT)** that distribution shifts and conflicting objectives during personality updates overwrite pre-trained competencies. To mitigate this tax, we propose a **P**ersonality-**O**rthogonal **G**radient **P**rojection **(POGP)** plug-in that preserves the capability in personality alignment. Specifically, POGP estimates the general-capability subspace from capability reference benchmarks and projects personality gradients onto the orthogonal complement before updating. This isolates and neutralizes conflicting gradients while maintaining the pre-trained reasoning capabilities. Experiments on the BIG5-CHAT dataset with two backbone LLMs demonstrate that POGP achieves a stronger personality plasticity and capability stability trade-off than baselines.

1 Introduction

Personality psychology provides a theoretical basis for describing individual differences, covering stable tendencies in emotion, interaction, and behavior (McCrae and John, 1992). Personality trait alignment in LLMs is significant for building personalized agents, as it enables more consistent behavior across diverse interaction scenarios (Park et al., 2023; Shao et al., 2023), such as role-playing dialogue systems (Zhang et al., 2025), human-computer interaction (Tang et al., 2025), and agent social simulation (Wang et al., 2025).

Personality-aligned LLMs are mainly divided into the following lines. Prompting methods use trait descriptions, role profiles, or questionnaire instructions to elicit desired role tendencies without updating model parameters (Mao et al., 2024; Serapio-García et al., 2025). Moving from surface-level prompting to parameter-level adaptation, Supervised Fine-Tuning (SFT) (Ouyang et al., 2022) trains models on personality-conditioned data to express personality in language and behavior patterns (Shao et al., 2023; Li et al., 2025). And then preference alignment methods such as DPO (Shao et al., 2024) and GRPO (Zhu et al., 2025) treat traits as a response or reward target and match personality profile (Rafailov et al., 2023). In parallel, representation steering methods inject trait vectors by intervening hidden activations (Zou et al., 2024; Turner et al., 2024). Although these methods improve personality consistency, they still leave a central question underexplored: **whether personality editing compromises the general capability of LLMs?**

Table 1: Empirical evidence for personality alignment tax under extraversion alignment on Llama-3.2-1B-Instruct. Extra scores are obtained by the personality questionnaire for the corresponding model. Avg is the mean general ability score, including commonsense reasoning (MMLU), mathematical reasoning (GSM8K), and truthfulness (TruthfulQA). Personality Alignment Tax (PAT) measures the mean general ability drop relative to the unedited model.

Category	Method	Extra. ↑	MMLU ↑	GSM8K ↑	Truth. ↑	Avg. ↑	PAT ↓
Unedited	Base	3.18	48.7	43.1	44.8	45.5	–
Prompting	Zero-Shot Prompt	3.93	48.2	42.4	44.2	44.9	0.6
	Few-Shot Prompt	4.05	47.6	41.8	43.8	44.4	1.1
	Descriptive Prompt	4.18	46.9	40.7	42.9	43.5	2.0
Fine-tuning	SFT	4.39	45.8	38.3	41.7	41.9	3.6
Preference Alignment	DPO	4.45	44.9	36.8	40.9	40.9	4.6
Representation Steering	VI to All Layers	4.34	41.8	31.5	38.6	37.3	8.2
	VI to Layers 7–8	4.21	45.1	38.9	42.1	42.0	3.5
	VI to Layers 5–6	4.25	43.6	35.7	40.5	39.9	5.6

To answer this question, we conduct a controlled study on extraversion alignment across four distinct personality editing baselines. As shown in Table 1, all baselines improve personality consistency, but none preserve general capability perfectly. Specifi-

Findings of the Association for Computational Linguistics: EMNLP 2026, pages 34698–34714
October 24-29, 2026 ©2026 Association for Computational Linguistics

cally, prompting pays the smallest tax, but the alignment effect is weaker; The DPO gives the strongest personality alignment and loses more general capability; The SFT and layer-wise VI lie between these two ends. Therefore, while personality alignment can modify target personality attributes, the edited model inevitably suffers from a degradation in general capability. In other words, a **P**ersonality **A**lignment **T**ax **(PAT)** will be paid. Fundamentally, personality alignment updates modify the same parameter subspaces that encode pre-trained capabilities, triggering catastrophic forgetting.

To mitigate PAT, we face the following challenges: **(1)** Mixing general data or auxiliary pre-training objectives during alignment reduces degradation, yet increases computational costs by introducing extra scheduling and parameters. **(2)** Adjusting soft constraints such as KL penalties in RLHF or GRPO merely controls macroscopic deviations, failing to precisely intercept the harmful updates that lead to catastrophic forgetting.

To address the above challenges, we propose a plug-in **P**ersonality-**O**rthogonal **G**radient **P**rojection **(POGP)** plug-in that preserves personality update components while preserving the capability in personality post-training. Specifically, POGP first uses a small capability reference set to estimate a general capability subspace, which captures local gradient directions for the original capabilities. During personality optimization, POGP decomposes the personality gradient into capability-overlapping and capability-orthogonal components and then updates the model only along the orthogonal direction. This projecting removes capability-conflicting components before parameter update, reducing first-order interference with commonsense, mathematical, social reasoning, and truthfulness. In summary, our main contributions are as follows:

- To our knowledge, this is the first work to measure and mitigate personality alignment tax in LLMs.
- We propose Personality-Orthogonal Gradient Projection (POGP) plug-in, a capability-preserving plug-in for personality post-training that projects personality gradients away from a general-capability subspace to reduce first-order interference.
- Extensive experiments with Llama-3.1-8B-Instruct and Qwen2.5-7B-Instruct demonstrate that POGP achieves a stronger personality-capability trade-off over baselines in 6 general capability benchmarks.

2 Related Work

2.1 Personality Alignment in LLMs

Existing personality alignment methods can be broadly grouped into four lines. **Prompting methods** use trait descriptions, persona instructions, or questionnaire-style prompts to elicit desired personality patterns without changing model parameters (Serapio-García et al., 2025; Mao et al., 2024). These methods are flexible, but their effects are often sensitive to prompt wording and may remain at the level of surface response style. **Fine-tuning methods** update model parameters with supervised role-playing or personality-conditioned data, enabling more stable behavioral adaptation (Shao et al., 2023; Li et al., 2025). To further optimize model responses according to preference or reward signals, **Preference alignment methods**, such as Direct Preference Optimization (DPO), are used to encourage responses that better match target personality attributes (Rafailov et al., 2023; Zhu et al., 2025). In parallel, **representation steering methods** intervene in hidden activations or inject trait-related vectors at inference time, offering lightweight personality control without full model updating (Li et al., 2023; Zou et al., 2024; Turner et al., 2024). While these methods improve trait expression consistency. In contrast, our work studies the capability degradation of personality editing and further mitigates this personality alignment tax.

2.2 Orthogonal Projection for Capability Preservation

Post-training objectives in LLMs can improve target behaviors but may degrade prior abilities when gradients conflict with capability-relevant directions (Ouyang et al., 2022; Rafailov et al., 2023). The gradient geometry projection method aims to eliminate conflicting components between gradients by directly modifying the gradients. Gradient Episodic Memory (GEM) (Lopez-Paz and Ranzato, 2017) constrains new-task updates using gradients from replay examples, while Orthogonal Gradient Descent (OGD) (Farajtabar et al., 2020) projects new-task gradients away from previous-task subspaces. Recent alignment work further applies this idea to preserve general capabilities during safety alignment by projecting safety gradients away from

a capability subspace (Sun et al., 2026), while Knowledge Decoupling Editing (KDE) (Xu et al., 2025) applies orthogonal projection to reduce interference among lifelong model edits. Inspired by those works, POGP aims to profile personality traits by personality gradients orthogonal projection, while reducing interference with general capabilities.

3 Problem Formulation

3.1 Personality Alignment Objective

Let $\pi_{\boldsymbol{\theta}}$ denote an LLM with trainable parameters $\boldsymbol{\theta}$. Given a target personality attribute τ and its alignment data $\mathcal{D}_\tau$, personality alignment optimizes a personality loss:

$$\boldsymbol{\theta}_p = \arg\min_{\boldsymbol{\theta}} \mathcal{L}_{\text{per}}(\boldsymbol{\theta}; \mathcal{D}_\tau), \tag{1}$$

where $\boldsymbol{\theta}_p$ denotes the personality-aligned parameters. The corresponding personality gradient is:

$$\mathbf{g}_{\text{per}} = \nabla_{\boldsymbol{\theta}} \mathcal{L}_{\text{per}}(\boldsymbol{\theta}; \mathcal{D}_\tau). \tag{2}$$

This formulation is independent of a specific personality taxonomy or dataset. In our implementation, $\mathcal{L}_{\text{per}}$ can be instantiated by Supervised Fine-Tuning (SFT) or Direct Preference Optimization (DPO); the concrete objectives are provided in Appendix B.1.

3.2 Personality Alignment Tax

For each general capability dimension j, let $s_j(\boldsymbol{\theta})$ be the corresponding evaluation score of model $\pi_{\boldsymbol{\theta}}$. The average general-capability score is defined as

$$\Phi(\boldsymbol{\theta}; \mathcal{B}_{\text{gen}}) = \frac{1}{J} \sum_{j=1}^{J} s_j(\boldsymbol{\theta}), \tag{3}$$

where J is the number of capability evaluation categories. each $\mathcal{B}_j$ evaluates one capability dimension. The personality alignment tax can be defined as:

$$\Delta_{\text{PAT}} = \Phi(\boldsymbol{\theta}_0; \mathcal{B}_{\text{gen}}) - \Phi(\boldsymbol{\theta}_p; \mathcal{B}_{\text{gen}}), \tag{4}$$

where $\boldsymbol{\theta}_0$ and $\boldsymbol{\theta}_p$ denote the original and personality-aligned model parameters, respectively. Our goal aims to minimize the personality alignment tax during the alignment process:

$$\begin{aligned} \boldsymbol{\theta}^\star = \arg\min_{\boldsymbol{\theta}} \quad & \mathcal{L}_{\text{per}}(\boldsymbol{\theta}; \mathcal{D}_\tau) \\ \text{s.t.} \quad & \Phi(\boldsymbol{\theta}_0; \mathcal{B}_{\text{gen}}) - \Phi(\boldsymbol{\theta}; \mathcal{B}_{\text{gen}}) \le \epsilon, \end{aligned} \tag{5}$$

where ϵ is the allowed capability degradation.

4 Method

As shown in Figure 1, our Personality-Orthogonal Gradient Projection (POGP) plug-in consists of three parts. First, the dynamic and low-rank capability subspaces are constructed from small reference sets. Second, the first-order Taylor expansion is used to characterize the capability-preserving space. Finally, POGP projects the personality gradient onto the orthogonal complement of the estimated subspace before optimization. The Algorithm of POGP is provided in Appendix B.2.

4.1 Capability Subspace Estimation

To estimate general capabilities, we first construct a dynamic, low-rank basis that spans capability-sensitive directions. Let $\{\mathcal{D}_{\text{ref}}^{(m)}\}_{m=1}^{M}$ denote small reference sets sampled from the general capability benchmarks (e.g., commonsense, mathematical, social reasoning, and truthfulness). For each dataset m, the differentiable reference loss gradient with respect to the parameters of the l-th layer is:

$$\mathbf{g}_{\text{ref},l}^{(m)} = \nabla_{\boldsymbol{\theta}_l} \mathcal{L}_{\text{ref}}^{(m)}(\boldsymbol{\theta}_l; \mathcal{D}_{\text{ref}}^{(m)}), \quad m = 1, \ldots, M, \tag{6}$$

where $\boldsymbol{\theta}_l \in \mathbb{R}^{d_l}$ represents the flattened parameter of layer l, and d_l is the number of parameters in this vector layer. We aggregate these layer-wise gradients into a reference gradient matrix:

$$\mathbf{G}_{\text{ref},l} = [\mathbf{g}_{\text{ref},l}^{(1)}, \ldots, \mathbf{g}_{\text{ref},l}^{(M)}] \in \mathbb{R}^{d_l \times M}. \tag{7}$$

The general-capability subspace is then estimated by extracting an orthonormal basis from this matrix:

$$\mathbf{U}_l = \text{Orth}(\mathbf{G}_{\text{ref},l}), \quad \mathbf{U}_l^\top \mathbf{U}_l = \mathbf{I}, \tag{8}$$

where $\mathbf{U}_l \in \mathbb{R}^{d_l \times r}$ contains r orthonormal basis vectors ($r \le M$). In practice, $\text{Orth}(\cdot)$ is implemented via QR (orthogonal–triangular) decomposition or Gram–Schmidt orthogonalization. The estimated capability subspace for layer l is defined as $\mathcal{S}_{\text{gen},l} = \text{span}(\mathbf{U}_l)$.

4.2 First-Order Capability Preservation

To theoretically ground the projection mechanism, we analyze the effect of a parameter update $\Delta\boldsymbol{\theta}_l$ on the general capabilities. For a small parameter update $\Delta\boldsymbol{\theta}_l$, a first-order Taylor expansion gives the change in the m-th reference loss:

$$\mathcal{L}_{\text{ref}}^{(m)}(\boldsymbol{\theta}_l + \Delta\boldsymbol{\theta}_l) \approx \mathcal{L}_{\text{ref}}^{(m)}(\boldsymbol{\theta}_l) + \left\langle \mathbf{g}_{\text{ref},l}^{(m)}, \Delta\boldsymbol{\theta}_l \right\rangle, \tag{9}$$

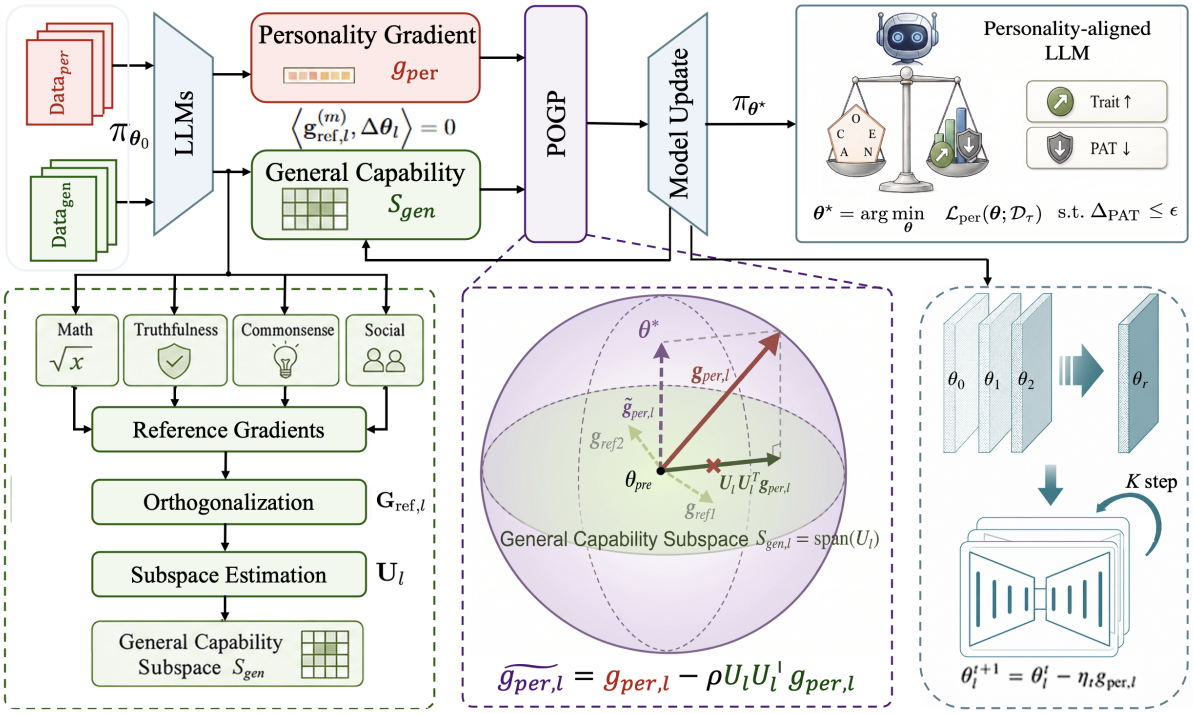


Figure 1: Overview of Personality-Orthogonal Gradient Projection (POGP). Given personality data and general capability reference sets as inputs, POGP first computes reference gradients to construct a capability-preserving gradient set, then estimates the general capability subspace from these gradients, performs orthogonal projection to remove capability-interfering components from the raw personality gradient, and finally updates the model with the projected personality gradient, yielding a trait-edited model with preserved general capabilities.

where $\mathbf{g}_{\text{ref},l}^{(m)}$ is defined in Eq. 6. To ensure that the personality alignment update does not degrade general capabilities, it should not alter the reference losses, which requires the inner product to be zero:

$$\left\langle \mathbf{g}_{\text{ref},l}^{(m)}, \Delta\boldsymbol{\theta}_l \right\rangle = 0, \quad m = 1, \ldots, M. \tag{10}$$

Based on this constraint, we explicitly define the capability-preserving space $\mathcal{S}_{\text{cp},l}$ as the orthogonal complement of the general-capability subspace:

$$\mathcal{S}_{\text{cp},l} = \left\{ \Delta\boldsymbol{\theta}_l \in \mathbb{R}^{d_l} \mid \mathbf{U}_l^\top \Delta\boldsymbol{\theta}_l = \mathbf{0} \right\}, \tag{11}$$

where $\mathbf{U}_l$ is the orthogonal basis derived in Eq. 8. Updates restricted within $\mathcal{S}_{\text{cp},l}$ strictly guarantee first-order capability preservation.

4.3 Personality-Orthogonal Gradient Projection

Let $\mathbf{g}_{\text{per},l} = \nabla_{\boldsymbol{\theta}_l} \mathcal{L}_{\text{per}}$ be the raw personality gradient for layer l. POGP mitigates the alignment tax by isolating and removing the gradient component that falls inside the estimated general-capability subspace. The projected gradient is computed as:

$$\widetilde{\mathbf{g}}_{\text{per},l} = \mathbf{g}_{\text{per},l} - \rho \mathbf{U}_l \mathbf{U}_l^\top \mathbf{g}_{\text{per},l}, \tag{12}$$

where $\widetilde{\mathbf{g}}_{\text{per},l} \in \mathbb{R}^{d_l}$ is the capability-preserving personality gradient, $\mathbf{U}_l \mathbf{U}_l^\top \mathbf{g}_{\text{per},l}$ represents the projection of the gradient onto $\mathcal{S}_{\text{gen},l}$, and $\rho \in [0, 1]$ is a static projection strength hyperparameter controlling the trade-off between personality alignment and capability retention. The parameters of layer l are subsequently updated by the optimizer:

$$\boldsymbol{\theta}_l^{t+1} = \boldsymbol{\theta}_l^t - \eta_t \widetilde{\mathbf{g}}_{\text{per},l}, \tag{13}$$

where η_t is the learning rate at optimization step t, and $\boldsymbol{\theta}_{l,t}$ denotes the parameters of layer l at step t. To account for the evolving geometry of the loss landscape during training, the capability basis $\mathbf{U}_l$ is dynamically refreshed every K steps. When $\rho = 0$, the method reduces to standard personality fine-tuning; when $\rho = 1$, absolute first-order capability preservation is enforced.

5 Experiments

5.1 Dataset and Baselines

We construct personality-alignment data from BIG5-CHAT (Li et al., 2025) and use general capability benchmarks as reference and evaluation sets. Details of data construction and preprocessing are provided in Appendix A.1.

We compare POGP with four categories of baselines. **Prompting methods** modify model behavior only through the input context, including instruction-based prompting (**Prompt-Inst**) (Wei et al., 2022), few-shot demonstration prompting (**Prompt-Demo**) (Brown et al., 2020), and personality-specific prompting (**P**^{2}) (Jiang et al., 2023). **Representation steering** controls personality expression at inference time by injecting trait-related activation directions into hidden representations, implemented as **Vector Injection** (Zou et al., 2024). **Post-training methods** directly update model parameters with personality data, including **SFT** (Ouyang et al., 2022) and **DPO** (Rafailov et al., 2023). **Capability-preserving methods** introduce additional constraints or reference data during personality training, including **Replay-SFT** (Lin et al., 2024) and **Anchor** (Ouyang et al., 2022). Full details for all baselines are provided in Appendix A.2.

5.2 Implementation Details and Metrics

We instantiate POGP via gradient projection during SFT and DPO based on Llama-3.1-8B-Instruct and Qwen2.5-7B-Instruct backbones. POGP projects the raw personality gradient over LoRA parameters away from a capability-sensitive subspace. All Post-training methods freeze the backbone and update only LoRA parameters using AdamW with bf16 precision. Comparable methods use the same backbone, tokenizer, LoRA budget, maximum sequence length, and training schedule. Full hyperparameters are provided in Appendix A.3. All experiments are implemented in PyTorch on eight NVIDIA A800 GPUs. Training details of POGP are provided in Appendix A.4.

Personality alignment scores are measured with Big Five Inventory (BFI) (John et al., 1991) and IPIP-NEO (Maples et al., 2014) questionnaires. Capability retention is evaluated on six benchmarks: Commonsense Reasoning, including **MMLU** (Hendrycks et al., 2021) and **CSQA** (Talmor et al., 2019); Mathematical Reasoning, including **GSM8K** (Cobbe et al., 2021) and **SVAMP** (Patel et al., 2021); Truthfulness, including **TruthfulQA** (Lin et al., 2022); and Social Reasoning, including **SocialIQA** (Sap et al., 2019). Avg. Cap. is the mean score over the six benchmarks, and PAT is the capability drop relative to the unedited backbone. Evaluation metrics are detailed in Appendix A.5.

5.3 Main Results

Overall capability preservation. Table 2 shows that POGP achieves the strongest capability retention among all aligned methods on both backbones. POGP-SFT obtains the best Avg. Cap. and the lowest PAT, reaching $69.3/0.5$ on Llama-3.1 and $72.5/0.4$ on Qwen2.5. Compared with standard SFT, POGP-SFT reduces PAT from 3.2 to 0.5 on Llama-3.1 and from 3.3 to 0.4 on Qwen2.5, while keeping personality alignment nearly unchanged. This improvement comes from gradient decoupling: replay and anchoring constrain training globally, whereas POGP removes only the components that overlap with capability-sensitive directions.

Personality–capability trade-off. DPO achieves the highest personality scores, with 4.64 on Llama-3.1 and 4.65 on Qwen2.5, but it also causes the largest capability loss, with PAT values of 4.4 and 4.3. POGP-DPO keeps personality alignment close to DPO, reaching 4.59 and 4.61, while reducing PAT to 1.1 and 0.9. This indicates that the goal is not to maximize personality score alone, but to preserve general ability while maintaining strong alignment. Prompting and representation steering avoid heavy parameter updates, but their personality scores are lower and their capability retention remains weaker than POGP. Overall, POGP provides a better balance by preserving most of the alignment signal while suppressing capability-conflicting updates.

Figure 2 visualizes this trade-off. Prompt-based methods remain relatively capability-preserving but provide weaker personality alignment. SFT and DPO improve alignment but move leftward in capability. Replay-SFT and Anchor recover part of the lost capability at the cost of weaker alignment. POGP moves closer to the upper-right region, showing stronger Big Five editing with substantially lower capability degradation.

5.4 Ablation Studies

Ablation on Projection Design. Table 4 analyzes which part of the projection design contributes to capability preservation. The **w/o Projection** variant degenerates to standard SFT: it obtains the highest personality score but also the largest PAT of 3.2, showing that unconstrained personality gradients damage capability-sensitive directions. **Random Projection** only slightly reduces PAT, indicating that the protected directions must be estimated from capability data rather than chosen ar-

Table 2: Overall personality alignment and capability retention. The decomposed BFI and IPIP-NEO scores corresponding to the Pers. column are reported in Appendix C.1; Avg. Cap. is the mean score over six capability benchmarks. All bold values indicate the best performance achieved among the aligned models.

Category	Methods	Pers. ↑	Commonsense Reasoning		Mathematical Reasoning		Truthfulness	Social Reasoning	Avg. Cap. ↑	PAT ↓
			MMLU ↑	CSQA ↑	GSM8K ↑	SVAMP ↑	TruthfulQA ↑	SocialIQA ↑		
Llama-3.1-8B-Instruct										
Unedited	Base	3.02	71.1	76.2	83.2	84.6	53.5	50.4	69.8	–
Prompting	Prompt-Inst	3.95	69.2	74.6	81.5	82.8	52.1	49.6	68.3	1.5
	Prompt-Demo	4.08	68.4	74.1	80.6	81.9	51.5	49.1	67.6	2.2
	P^2	4.22	68.0	73.5	79.5	81.2	50.8	48.4	66.9	2.9
Representation Steering	Vector Injection	4.43	68.6	74.0	80.1	82.0	51.1	48.6	67.4	2.4
Post Training	SFT	4.56	68.5	72.9	78.9	81.2	50.6	47.6	66.6	3.2
	DPO	**4.64**	66.8	71.2	78.1	79.5	49.7	47.0	65.4	4.4
Capability Preserving	Replay-SFT	4.39	69.0	74.3	81.5	82.0	51.8	48.7	67.9	1.9
	Anchor	4.31	68.6	73.6	80.4	81.8	52.2	49.0	67.6	2.2
POGP	POGP-SFT	4.53	**70.8**	**75.8**	**82.7**	**83.8**	**52.7**	**50.0**	**69.3**	**0.5**
	POGP-DPO	4.59	70.1	75.2	81.9	83.1	52.4	49.5	68.7	1.1
Qwen2.5-7B-Instruct										
Unedited	Base	3.18	73.5	79.6	86.0	88.1	55.4	54.8	72.9	–
Prompting	Prompt-Inst	4.10	72.0	77.5	84.3	86.1	53.8	53.1	71.1	1.8
	Prompt-Demo	4.22	71.2	76.8	83.5	85.2	53.2	52.5	70.4	2.5
	P^2	4.35	70.8	75.9	82.4	84.6	52.6	52.1	69.7	3.2
Representation Steering	Vector Injection	4.46	71.5	76.4	83.0	85.0	53.0	52.7	70.3	2.6
Post Training	SFT	4.58	70.5	76.3	82.7	83.9	52.6	51.6	69.6	3.3
	DPO	**4.65**	69.8	75.0	80.7	82.9	52.0	51.2	68.6	4.3
Capability Preserving	Replay-SFT	4.41	71.8	78.2	83.9	85.7	53.9	53.0	71.1	1.8
	Anchor	4.32	71.4	77.3	83.0	85.2	54.2	52.9	70.7	2.2
POGP	POGP-SFT	4.56	**73.2**	**79.3**	**85.7**	**87.4**	**54.9**	**54.4**	**72.5**	**0.4**
	POGP-DPO	4.61	72.6	78.8	85.0	86.9	54.5	53.9	72.0	0.9

Table 3: Capability reference set size (Llama-3.1-8B-Instruct, POGP-SFT). A similar capability reference budget check under POGP-DPO on Qwen2.5-7B-Instruct is provided in Appendix C.3.

Samples per Benchmark	Pers. ↑	MMLU ↑	GSM8K ↑	Truth. ↑	Avg. ↑	PAT ↓
25	**4.55**	69.7	81.0	51.5	68.2	1.6
50	4.54	70.2	81.7	52.0	68.7	1.1
100	4.53	70.5	82.3	52.4	69.0	0.8
200	4.53	70.8	82.7	52.7	69.3	0.5
400	4.52	**70.9**	**82.8**	**52.8**	**69.4**	**0.4**

Table 4: Effect of orthogonal projection (Llama-3.1-8B-Instruct, POGP-SFT).

Variant	Pers. ↑	MMLU ↑	GSM8K ↑	Truth. ↑	Avg. ↑	PAT ↓
w/o Projection	**4.56**	68.5	78.9	50.6	66.6	3.2
Random Projection	4.50	68.9	79.6	51.2	67.2	2.6
Fixed Subspace	4.51	69.8	81.1	51.8	68.1	1.7
Soft Projection	4.55	70.2	81.8	52.1	68.6	1.2
POGP (full)	4.53	**70.8**	**82.7**	**52.7**	**69.3**	**0.5**

bitrarily. **Fixed-Subspace Projection** further improves retention by projecting onto a precomputed capability basis, but its static basis cannot track the changing gradient geometry during personality training. **Soft Projection** preserves more alignment plasticity by only partially removing capability-overlapping components, yet the remaining conflicting gradients still cause capability loss. In contrast, **POGP (full)** dynamically refreshes the capability subspace and fully removes the overlapping components, achieving the best Avg. score and the lowest PAT while keeping personality alignment close to SFT.

Effect of General-capability Subspace Composition. Table 5 reports how different capability reference constraints shape the protective subspace. We observe that single-domain capability reference successfully shield their corresponding capability (e.g., GSM8K only protects math reasoning at 82.4), but fail to prevent regressions in other capability. The underlying mechanism is that diverse reasoning skills rely on distinct, non-overlapping activation subspaces. Consequently, the **mixed single gradient** (averaging all capability domains) mitigates overall decay but lacks sufficient dimensionality to cover the complex capability manifold. **POGP Full** constructs a comprehensive subspace covering all benchmarks, verifying that multidimensional constraints are necessary to suppress the multi-faceted alignment tax.

Effect of Capability Reference Sample Size. Table 3 studies how the size of the capability reference set affects POGP. As the number of reference samples increases from 25 to 400 per benchmark,

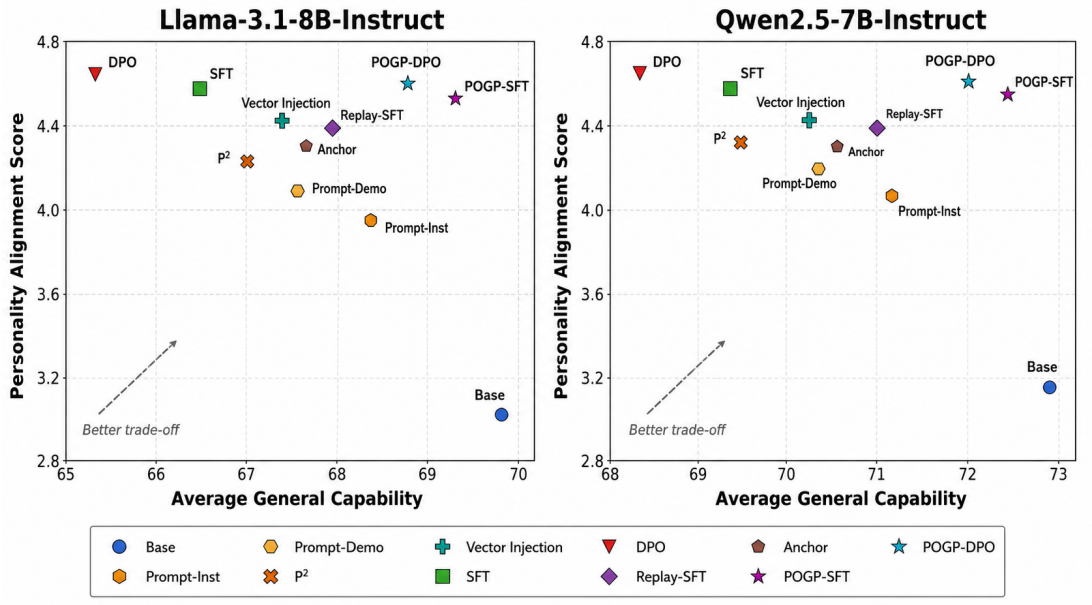


Figure 2: Personality capability trade-off on two backbone LLMs.

Table 5: General-capability subspace composition (Llama-3.1-8B-Instruct, POGP-SFT). A corresponding POGP-DPO validation on Qwen2.5-7B-Instruct is reported in Appendix C.2.

General-Capability Subspace	Pers. ↑	MMLU ↑	GSM8K ↑	Truth. ↑	Avg. ↑	PAT ↓
MMLU only	4.54	70.6	80.3	51.4	68.0	1.8
GSM8K only	**4.55**	69.2	82.4	51.1	68.1	1.7
TruthfulQA only	4.54	69.0	80.1	52.6	67.9	1.9
Mixed single gradient	4.53	69.8	81.2	51.8	68.4	1.4
POGP (full)	4.53	**70.8**	**82.7**	**52.7**	**69.3**	**0.5**

Table 6: Subspace update frequency (Llama-3.1-8B-Instruct, POGP-SFT).

Update Frequency	Pers. ↑	MMLU ↑	GSM8K ↑	Truth. ↑	Avg. ↑	PAT ↓
No Update	**4.55**	69.5	80.6	51.5	68.0	1.8
Every 50 steps	**4.55**	70.1	81.5	52.0	68.6	1.2
Every 20 steps	4.53	**70.8**	82.7	52.7	69.3	0.5
Every 10 steps	4.51	70.6	82.6	52.8	69.2	0.6
Every 5 steps	4.48	70.9	**82.9**	**52.9**	**69.4**	**0.4**

capability retention improves steadily and PAT decreases from 1.6 to 0.4. This shows that a larger reference set provides a more reliable estimate of capability-sensitive directions. It is worth noting that the improvement becomes much smaller after 200 samples: increasing the budget from 200 to 400 only improves Avg. from 69.3 to 69.4 and reduces PAT from 0.5 to 0.4. Therefore, although 400 samples achieve the best numerical result, we use 200 samples in the main experiments as a practical trade-off between capability preservation and reference-gradient estimation cost. This saturation trend suggests that the dominant capability directions can be captured with a relatively small reference set, while additional samples mainly provide redundant or fine-grained constraints.

Effect of Subspace Update Frequency. As shown in Table 6, a static initial subspace (**No Update**) gradually becomes stale as the model parameters drift during tuning, resulting in inadequate protection (PAT of 1.8). Conversely, updating too frequently (**Every 5 steps**) acts as an overly strict regularization bottleneck. While it perfectly preserves general metrics, it severely inhibits the plasticity needed for personality acquisition, dropping the alignment score to 4.48. Updating **every 20 steps** empirically guarantees that the projection matrix accurately tracks the manifold's local curvature without excessively dampening the alignment trajectory.

5.5 Trait-wise Personality Alignment Tax

As shown in Figure 3, SFT and DPO show clear trait-dependent capability loss: affective and expressive targets such as High Extraversion and High Neuroticism induce larger PAT, while High Conscientiousness produces a smaller drop. This suggests that personality alignment affects general capability unevenly; traits that require stronger stylistic or affective shifts are more likely to conflict with capability-sensitive directions. In contrast, POGP-SFT keeps PAT within a narrow range of 0.3–0.7 across all traits, with an average around 0.5. The stable reduction indicates that POGP does not rely on trait-specific heuristics, but suppresses

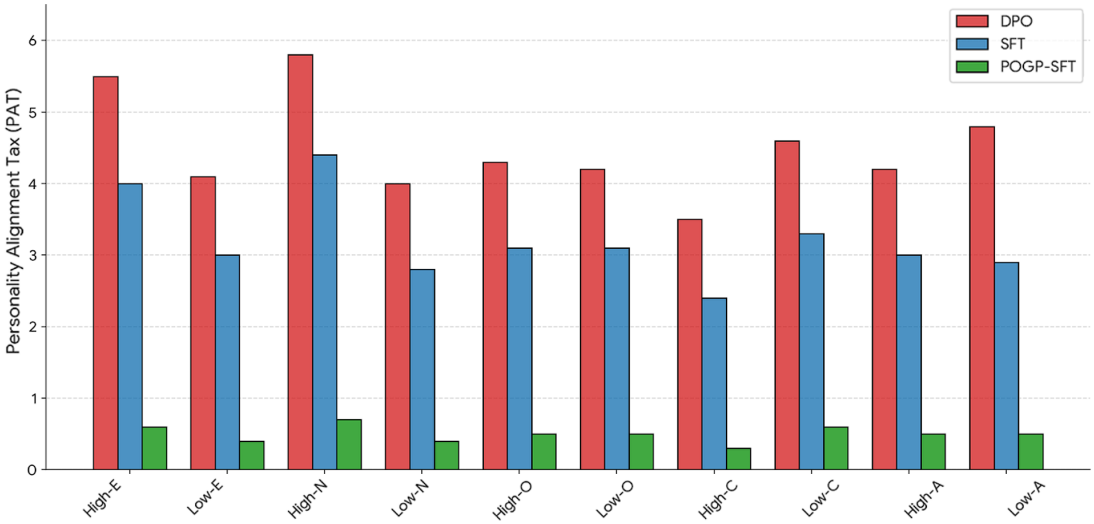


Figure 3: Trait-wise personality alignment tax on Llama-3.1-8B-Instruct.

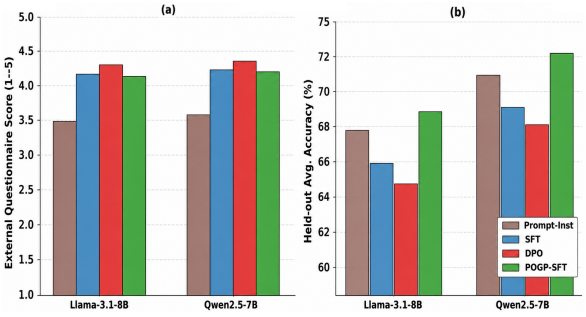


Figure 4: External generalization evaluation on two backbone LLMs. (a) Held-out personality assessment. (b) Capability retention on held-out benchmarks.

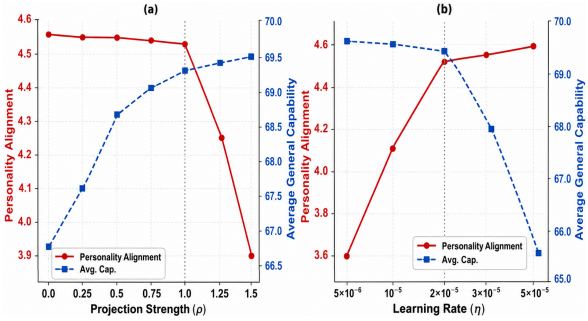


Figure 5: Hyperparameter sensitivity of POGP-SFT on Llama-3.1-8B-Instruct. (a) Projection strength ρ with learning rate. (b) Learning rate η with projection strength.

capability-conflicting gradient components across different personality targets.

5.6 External Questionnaire-Based Personality Assessment

We use an external generalization protocol to test whether the personality–capability trade-off holds beyond the main evaluation. For personality, we evaluate models with two held-out Big Five questionnaires, TIPI and Mini-IPIP (Gosling et al., 2003; Donnellan et al., 2006). For capability, we report average accuracy on held-out reasoning benchmarks, including ARC-Challenge, HellaSwag, and WinoGrande, which are excluded from the POGP reference set.

As shown in Figure 4(a), DPO obtains the highest external questionnaire scores, while SFT and POGP-SFT remain close. However, Figure 4(b) shows that SFT and DPO suffer larger held-out capability drops, with DPO degrading the most. POGP-SFT achieves the best held-out capability retention on both backbones while keeping personality alignment close to SFT. These results suggest that POGP improves the external personality–capability trade-off rather than overfitting to the main benchmarks or the capability reference set.

5.7 Hyperparameter Analysis

This experiment tests whether the main setting remains stable when the projection constraint or optimization step size changes. As shown in Figure 5(a), increasing ρ improves capability retention because more capability-overlapping gradient components are removed. However, overly large projection strength suppresses useful personality-learning gradients and sharply lowers the personality score. Thus, $\rho = 1.0$ gives the best trade-off rather than the highest capability alone. Figure 5(b) shows a similar trade-off for learning rate. Small learning rates underfit personality alignment, while large learning rates improve personality scores but cause

stronger capability degradation. The main setting $\eta = 2 \times 10^{-5}$ maintains high personality alignment while avoiding the capability collapse observed at larger learning rates.

6 Conclusion

In this paper, we conduct an empirical analysis and identify the personality alignment tax in LLMs, demonstrating that personality editing inevitably induces severe capability regressions. To mitigate this tax, we propose the Personality-Orthogonal Gradient Projection (POGP) framework, which effectively decouples the personality alignment plasticity from model stability. Extensive experiments on 6 general capability benchmarks demonstrate that POGP significantly outperforms SOTA personality alignment baselines and capability-preservation baselines, establishing an elegant balance of personality plasticity and general abilities stability.

7 Limitations

This work has several limitations. (1) POGP estimates capability-sensitive directions from finite reference benchmarks, which cannot fully encapsulate the diverse spectrum of LLM abilities. Our empirical evaluation is currently restricted to the Big Five traits across two open-weight backbones. In the future, we will expand the evaluation scale and diversity by incorporating other standard personality taxonomies (e.g., MBTI), scaling up to larger language models, and exploring multilingual alignment scenarios. (2) POGP relies on a first-order Taylor expansion to mitigate local gradient interference, which may not guarantee strict capability preservation over extended optimization trajectories or highly non-linear loss landscapes. In the future, we aim to investigate higher-order optimization constraints and develop adaptive projection mechanisms to guarantee long-term stability of capability.

8 Ethical Considerations

This work studies personality alignment for LLMs with the goal of reducing capability degradation during model editing. All datasets used in our experiments are publicly available and are used only for academic research. We recognize that personality-controlled models may create risks such as over-personalization, emotional manipulation, stereotyping, or misleading anthropomorphism. Therefore, personality-edited models should not be deployed in high-stakes settings, such as education, mental health, or other decision-sensitive domains, without clear disclosure, informed user consent, and careful ethical review.

9 Acknowledgments

This work was supported by National Key Research and Development Program (No. 2024YFB4710400), National Natural Science Foundation of China (62606338, 62272323, 62403334, 62206148), Project of the Basic Research Operating Fund of the China Academy of Civil Aviation Science and Technology, The Key Technological Breakthrough "Jiebang Guashuai" Projects for Major Industries in Henan Province (No.251000210300), Fundamental Research Funds of CASTC (262060302026015), Interdisciplinary Project of Capital Normal University (Grant No.2026JCYB02), Natural Science Foundation of Anhui University of Science and Technology (XWYJ202301), CIPS-SMP-Zhipu Large Model Fund (20250307), China Postdoctoral Science Foundation (2026T190424, 2025M771594), and Beijing Postdoctoral Research Foundation (2025-135).

References

Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, and 12 others. 2020. Language models are few-shot learners. In *Advances in Neural Information Processing Systems*, volume 33, pages 1877–1901.

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

M. Brent Donnellan, Frederick L. Oswald, Brendan M. Baird, and Richard E. Lucas. 2006. The mini-IPIP scales: Tiny-yet-effective measures of the big five factors of personality. *Psychological Assessment*, 18(2):192–203.

Mehrdad Farajtabar, Navid Azizan, Alexander Mott, and Ang Li. 2020. Orthogonal gradient descent for continual learning. In *Proceedings of the Twenty*

Third International Conference on Artificial Intelligence and Statistics, volume 108 of *Proceedings of Machine Learning Research*, pages 3762–3773.

Samuel D. Gosling, Peter J. Rentfrow, and William B. Jr. Swann. 2003. A very brief measure of the big-five personality domains. *Journal of Research in Personality*, 37(6):504–528.

Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. 2021. Measuring massive multitask language understanding. In *International Conference on Learning Representations*.

Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2022. LoRA: Low-rank adaptation of large language models. In *International Conference on Learning Representations*.

Guangyuan Jiang, Manjie Xu, Song-Chun Zhu, Wenjuan Han, Chi Zhang, and Yixin Zhu. 2023. Evaluating and inducing personality in pre-trained language models. In *Advances in Neural Information Processing Systems*, volume 36, pages 10622–10643.

Oliver P. John, Eileen M. Donahue, and Robert L. Kentle. 1991. The big five inventory–versions 4a and 54. Technical report, University of California, Berkeley, Institute of Personality and Social Research, Berkeley, CA.

Kenneth Li, Oam Patel, Fernanda Viégas, Hanspeter Pfister, and Martin Wattenberg. 2023. Inference-time intervention: Eliciting truthful answers from a language model. In *Advances in Neural Information Processing Systems*, volume 36, pages 41451–41530.

Wenkai Li, Jiarui Liu, Andy Liu, Xuhui Zhou, Mona T. Diab, and Maarten Sap. 2025. BIG5-CHAT: Shaping LLM personalities through training on human-grounded data. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 20434–20471.

Stephanie Lin, Jacob Hilton, and Owain Evans. 2022. Truthfulqa: Measuring how models mimic human falsehoods. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics*, pages 3214–3252.

Yong Lin, Hangyu Lin, Wei Xiong, Shizhe Diao, Jianmeng Liu, Jipeng Zhang, Rui Pan, Haoxiang Wang, Wenbin Hu, Hanning Zhang, Hanze Dong, Renjie Pi, Han Zhao, Nan Jiang, Heng Ji, Yuan Yao, and Tong Zhang. 2024. Mitigating the alignment tax of RLHF. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 580–606.

David Lopez-Paz and Marc'Aurelio Ranzato. 2017. Gradient episodic memory for continual learning. In *Advances in Neural Information Processing Systems*, volume 30, pages 6470–6479.

Shengyu Mao, Xiaohan Wang, Mengru Wang, Yong Jiang, Pengjun Xie, Fei Huang, and Ningyu Zhang. 2024. Editing personality for large language models. In *Natural Language Processing and Chinese Computing*, volume 15360 of *Lecture Notes in Computer Science*, pages 241–254.

Jessica L. Maples, Li Guan, Nathan T. Carter, and Joshua D. Miller. 2014. A test of the international personality item pool representation of the revised NEO personality inventory and development of a 120-item IPIP-based measure of the five-factor model. *Psychological Assessment*, 26(4):1070–1084.

Robert R. McCrae and Oliver P. John. 1992. An introduction to the five-factor model and its applications. *Journal of Personality*, 60(2):175–215.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul Christiano, Jan Leike, and Ryan Lowe. 2022. Training language models to follow instructions with human feedback. In *Advances in Neural Information Processing Systems*, volume 35, pages 27730–27744.

Joon Sung Park, Joseph C. O'Brien, Carrie J. Cai, Meredith Ringel Morris, Percy Liang, and Michael S. Bernstein. 2023. Generative agents: Interactive simulacra of human behavior. In *Proceedings of the 36th Annual ACM Symposium on User Interface Software and Technology*, pages 2:1–2:22.

Arkil Patel, Satwik Bhattamishra, and Navin Goyal. 2021. Are nlp models really able to solve simple math word problems? In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 2080–2094.

Rafael Rafailov, Archit Sharma, Eric Mitchell, Christopher D. Manning, Stefano Ermon, and Chelsea Finn. 2023. Direct preference optimization: Your language model is secretly a reward model. In *Advances in Neural Information Processing Systems*, volume 36, pages 53728–53741.

Maarten Sap, Hannah Rashkin, Derek Chen, Ronan Le Bras, and Yejin Choi. 2019. Socialiqa: Commonsense reasoning about social interactions. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing*, pages 4463–4473.

Greg Serapio-García, Mustafa Safdari, Clément Crepy, Luning Sun, Stephen Fitz, Peter Romero, Marwa Abdulhai, Aleksandra Faust, and Maja J. Mataric. 2025. A psychometric framework for evaluating and shaping personality traits in large language models. *Nature Machine Intelligence*, 7:1954–1968.

Yunfan Shao, Linyang Li, Junqi Dai, and Xipeng Qiu. 2023. Character-LLM: A trainable agent for role-playing. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 13153–13187.

Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, Y. K. Li, Y. Wu, and Daya Guo. 2024. DeepSeekMath: Pushing the limits of mathematical reasoning in open language models. *arXiv preprint arXiv:2402.03300*.

Guanglong Sun, Siyuan Zhang, Liyuan Wang, Jun Zhu, Hang Su, and Yi Zhong. 2026. Safety alignment as continual learning: Mitigating the alignment tax via orthogonal gradient projection. *arXiv preprint arXiv:2602.07892*.

Alon Talmor, Jonathan Herzig, Nicholas Lourie, and Jonathan Berant. 2019. Commonsenseqa: A question answering challenge targeting commonsense knowledge. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 4149–4158.

Jiakai Tang, Heyang Gao, Xuchen Pan, Lei Wang, Haoran Tan, Dawei Gao, Yushuo Chen, Xu Chen, Yankai Lin, Yaliang Li, Bolin Ding, Jingren Zhou, Jun Wang, and Ji-Rong Wen. 2025. GenSim: A general social simulation platform with large language model based agents. In *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (System Demonstrations)*, pages 143–150.

Alexander Matt Turner, Lisa Thiergart, Gavin Leech, David Udell, Juan J. Vazquez, Ulisse Mini, and Monte MacDiarmid. 2024. Steering language models with activation engineering. *arXiv preprint arXiv:2308.10248*.

Xingbo Wang, Janessa Griffith, Daniel A. Adler, Joey Castillo, Tanzeem Choudhury, and Fei Wang. 2025. Exploring personalized health support through data-driven, theory-guided LLMs: A case study in sleep health. In *Proceedings of the 2025 CHI Conference on Human Factors in Computing Systems*, pages 507:1–507:15.

Jason Wei, Maarten Bosma, Vincent Y. Zhao, Kelvin Guu, Adams Wei Yu, Brian Lester, Nan Du, Andrew M. Dai, and Quoc V. Le. 2022. Finetuned language models are zero-shot learners. In *International Conference on Learning Representations*.

Haoyu Xu, Pengxiang Lan, Enneng Yang, Guibing Guo, Jianzhe Zhao, Linying Jiang, and Xingwei Wang. 2025. Knowledge decoupling via orthogonal projection for lifelong editing of large language models. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 13194–13213.

Haonan Zhang, Run Luo, Xiong Liu, Yuchuan Wu, Ting-En Lin, Pengpeng Zeng, Qiang Qu, Feiteng Fang, Min Yang, Lianli Gao, Jingkuan Song, Fei Huang, and Yongbin Li. 2025. OmniCharacter: Towards immersive role-playing agents with seamless speech-language personality interaction. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 26318–26331.

Minjun Zhu, Yixuan Weng, Linyi Yang, and Yue Zhang. 2025. Personality alignment of large language models. In *International Conference on Learning Representations*.

Andy Zou, Long Phan, Sarah Li Chen, James Campbell, Phillip Huang Guo, Richard Ren, Alexander Pan, Xuwang Yin, Mantas Mazeika, Annah Dombrowski, Shashwat Goel, Nathaniel Li, Michael J. Byun, Zifan Wang, Alex Troy Mallen, Steven Basart, Sanmi Koyejo, Dawn Song, Matt Fredrikson, and 2 others. 2024. Enhancing neural network transparency through representation analysis. In *International Conference on Learning Representations*.

Appendix

A Experimental Setup Details

This appendix describes the implementation details that are omitted from the main body for space reasons. We introduce the data construction, baseline implementation, POGP training, and evaluation protocols.

A.1 Personality Traits and Data Construction

We use BIG5-CHAT to construct personality-alignment data (Li et al., 2025). Each target corresponds to one Big Five trait level: high or low Openness, Conscientiousness, Extraversion, Agreeableness, and Neuroticism. Therefore, each backbone has ten independent personality-alignment tasks and we train a separate checkpoint for each target.

For Supervised Fine-Tuning (SFT), each training example consists of a dialogue context and the response that matches the target trait level. For Direct Preference Optimization (DPO), we pair responses within the same trait dimension. The target-level response is used as the chosen response, and the opposite-level response is used as the rejected response. For example, when training high Agreeableness, the high-Agreeableness response is chosen and the low-Agreeableness response is rejected. This pairing avoids introducing cross-trait preference noise.

We use the same train and validation split for all training-based methods. The validation split is used only to monitor training stability. It is not used for model selection on the final evaluation results. The maximum sequence length is 1024 tokens. When an example exceeds this length, the dialogue context is truncated from the left to preserve the final user query and the target response.

A.2 Baseline

The prompting baselines do not update model parameters. **Prompt-Inst** prepends a trait-level instruction to the user query. **Prompt-Demo** adds five fixed in-context demonstrations sampled from the corresponding target split. $\mathbf{P}^2$ is implemented as a personality prompting baseline following the prompt-based personality evaluation setting of Jiang et al. (2023). The prompts are fixed before evaluation and are not tuned on the test set.

All training-based methods use Low-Rank Adaptation (LoRA) (Hu et al., 2022). **SFT** is trained on target-trait responses with the causal language modeling loss. **DPO** is initialized from the corresponding SFT checkpoint and then optimized with target-trait preference pairs (Rafailov et al., 2023).

We include two capability-preserving baselines. **Replay-SFT** mixes personality data with a small number of capability reference examples. We insert one replay batch after every nine personality batches, giving an approximate 10% replay ratio. **Anchor** adds a regularization term that keeps the aligned model close to the original instruction model. The original model is frozen and used only to compute the anchor loss.

Table 7: Fixed hyperparameters used for training-based methods.

Hyperparameter	Value
LoRA rank	16
LoRA alpha	32
LoRA dropout	0.05
Learning rate	2×10^{-5}
Optimizer	AdamW
Precision	bf16
Maximum sequence length	1024
Effective batch size	64
Warmup ratio	0.03
SFT epochs	3
DPO epochs	1
DPO coefficient β	0.1
Replay ratio	10%
Anchor weight	0.01
POGP projection strength ρ	1.0
POGP refresh interval	20
Capability examples per benchmark	200
Maximum subspace rank	3

A.3 Fixed Hyperparameters

Table 7 lists the fixed hyperparameters used for training-based methods. The model hidden sizes used for layer-wise projection are 4096 for Llama-3.1-8B-Instruct and 3584 for Qwen2.5-7B-Instruct. These are architecture constants rather than tuned hyperparameters. No hyperparameter is tuned on the final test sets. Effective batch size is obtained through gradient accumulation when necessary.

A.4 POGP Training Details

POGP-SFT and POGP-DPO are applied as an optimizer-side modification to SFT and DPO. Before each optimizer update, POGP projects the raw personality gradient over LoRA parameters away from a capability-sensitive subspace. We construct a held-out capability reference pool from MMLU, GSM8K, SVAMP, TruthfulQA, CommonsenseQA, and SocialIQA. For each benchmark, 200 exam-

ples are sampled with seed 17 from non-reporting splits when available. When a benchmark does not provide a separate training split, we first construct a fixed evaluation split and sample the reference pool from the remaining examples. Any exact-overlap item with the final evaluation split is removed from the reference pool. The reference pool is used only to estimate capability-sensitive directions and is never mixed into the personality training data.

Every 20 optimizer steps, POGP refreshes the capability basis using balanced mini-batches sampled from the reference pool. The full reference pool is not replayed at every refresh. For multiple-choice tasks, the reference loss is computed as the negative log-likelihood of the gold option. For GSM8K and SVAMP, the reference loss is computed on the gold final answer string. Gradients are collected only over trainable LoRA parameters. For each projected LoRA layer, the corresponding gradients are flattened, normalized, and orthonormalized using QR (orthogonal–triangular) decomposition. Directions whose norm after orthogonalization is below 10^{-6} are discarded. The maximum rank of each layer-wise capability subspace is set to 3.

A.5 Evaluation Protocol

Personality alignment is evaluated with the BFI and IPIP-NEO questionnaires (John et al., 1991; Maples et al., 2014). Each questionnaire item is converted into a Likert-style query that asks the model to output one integer from 1 to 5. We repeat personality evaluation five times with temperature 0.6 and seeds 0–4. The parser extracts the first valid integer from each response. If no valid integer is found, the response is assigned the neutral score 3. Reverse-keyed items are corrected before aggregation. For low-level target traits, we transform the raw score by $6 - s_{\text{raw}}$, so that higher scores always indicate stronger alignment with the intended target.

Capability evaluation uses greedy decoding for all methods. For MMLU, CommonsenseQA, TruthfulQA, and SocialIQA, we report multiple-choice accuracy. For GSM8K and SVAMP, we report exact-match accuracy on the final numerical answer. The same prompts, decoding settings, and answer parsers are used for all compared methods.

B Training Objectives

B.1 Personality Alignment Objectives

Let τ denote the target personality trait. For Supervised Fine Tuning (SFT), each sample is an input response pair $(x, y) \in \mathcal{D}_\tau^{\text{sft}}$. The objective is

$$\mathcal{L}_{\text{SFT}}(\boldsymbol{\theta}) = -\mathbb{E}_{(x,y)\sim\mathcal{D}_\tau^{\text{sft}}} \log \pi_{\boldsymbol{\theta}}(y \mid x), \quad (14)$$

where $\pi_{\boldsymbol{\theta}}(y \mid x)$ is the probability assigned by the edited model to the target response.

For Direct Preference Optimization (DPO), each sample is a preference triple $(x, y^+, y^-) \in \mathcal{D}_\tau^{\text{dpo}}$, where y^+ better matches the target trait than y^-. We define

$$r_{\boldsymbol{\theta}}(x, y) = \log \pi_{\boldsymbol{\theta}}(y \mid x) - \log \pi_{\text{ref}}(y \mid x), \quad (15)$$

where π_{ref} is the frozen reference model. The DPO objective is

$$\mathcal{L}_{\text{DPO}}(\boldsymbol{\theta}) = -\mathbb{E}_{\mathcal{D}_\tau^{\text{dpo}}} \log \sigma\Big[\beta\big(r_{\boldsymbol{\theta}}(x, y^+) - r_{\boldsymbol{\theta}}(x, y^-)\big)\Big], \quad (16)$$

where β controls the strength of the preference margin and $\sigma(\cdot)$ is the sigmoid function.

B.2 Algorithmic of POGP

Algorithm 1 POGP: Personality-Orthogonal Gradient Projection

Require: Initial parameters $\boldsymbol{\theta}_0$; personality loss $\mathcal{L}_{\text{per}}$; reference sets $\{\mathcal{D}_{\text{ref}}^{(m)}\}_{m=1}^{M}$; refresh period K; projection strength ρ; learning rate η.
Ensure: Personality-aligned parameters $\boldsymbol{\theta}_T$.

Initialize $\mathbf{U} \leftarrow [\,]$
for $t = 0, \ldots, T-1$ **do**
  if $t \bmod K = 0$ **then**
    for $m = 1, \ldots, M$ **do**
      Sample $\mathcal{B}_{\text{ref}}^{(m)} \sim \mathcal{D}_{\text{ref}}^{(m)}$
      $\mathbf{g}_{\text{ref}}^{(m)} \leftarrow \nabla_{\boldsymbol{\theta}} \mathcal{L}_{\text{ref}}^{(m)}(\boldsymbol{\theta}_t; \mathcal{B}_{\text{ref}}^{(m)})$
    end for
    $\mathbf{G}_{\text{ref}} \leftarrow [\mathbf{g}_{\text{ref}}^{(1)}, \ldots, \mathbf{g}_{\text{ref}}^{(M)}]$
    $\mathbf{U} \leftarrow \text{Orth}(\mathbf{G}_{\text{ref}})$
  end if
  Sample personality batch $\mathcal{B}_\tau \sim \mathcal{D}_\tau$
  $\mathbf{g}_{\text{per}} \leftarrow \nabla_{\boldsymbol{\theta}} \mathcal{L}_{\text{per}}(\boldsymbol{\theta}_t; \mathcal{B}_\tau)$
  $\widetilde{\mathbf{g}}_{\text{per}} \leftarrow \mathbf{g}_{\text{per}} - \rho \mathbf{U}\mathbf{U}^\top \mathbf{g}_{\text{per}}$
  $\boldsymbol{\theta}_{t+1} \leftarrow \boldsymbol{\theta}_t - \eta \widetilde{\mathbf{g}}_{\text{per}}$
end for
return $\boldsymbol{\theta}_T$

C Additional Experimental Results

C.1 BFI and IPIP-NEO Personality Assessment

In the main text, we report a single Pers. score for compact comparison. Here, we provide the decom-

Table C1: Decomposed personality assessment results on the BFI and IPIP-NEO questionnaires. Avg. Align is the average score of BFI and PIP-NEO.

Method	BFI ↑	IPIP-NEO ↑	Avg. Align ↑
Llama-3.1-8B-Instruct			
Base	3.04 ± 0.05	3.00 ± 0.06	3.02
Prompt-Inst	4.00 ± 0.07	3.90 ± 0.08	3.95
Prompt-Demo	4.13 ± 0.07	4.03 ± 0.08	4.08
P^2	4.27 ± 0.06	4.17 ± 0.07	4.22
Vector Injection	4.47 ± 0.06	4.39 ± 0.07	4.43
SFT	4.61 ± 0.05	4.51 ± 0.06	4.56
DPO	$\mathbf{4.69 \pm 0.04}$	$\mathbf{4.59 \pm 0.05}$	**4.64**
Replay-SFT	4.44 ± 0.05	4.34 ± 0.06	4.39
Anchor	4.36 ± 0.05	4.26 ± 0.06	4.31
POGP-SFT	4.58 ± 0.05	4.48 ± 0.06	4.53
POGP-DPO	4.64 ± 0.04	4.54 ± 0.05	4.59
Qwen2.5-7B-Instruct			
Base	3.21 ± 0.04	3.15 ± 0.06	3.18
Prompt-Inst	4.15 ± 0.06	4.05 ± 0.07	4.10
Prompt-Demo	4.27 ± 0.06	4.17 ± 0.07	4.22
P^2	4.40 ± 0.06	4.30 ± 0.07	4.35
Vector Injection	4.50 ± 0.05	4.42 ± 0.06	4.46
SFT	4.63 ± 0.04	4.53 ± 0.05	4.58
DPO	$\mathbf{4.70 \pm 0.03}$	$\mathbf{4.60 \pm 0.05}$	**4.65**
Replay-SFT	4.46 ± 0.05	4.36 ± 0.06	4.41
Anchor	4.37 ± 0.05	4.27 ± 0.06	4.32
POGP-SFT	4.61 ± 0.04	4.51 ± 0.05	4.56
POGP-DPO	4.66 ± 0.03	4.56 ± 0.05	4.61

posed results on BFI and IPIP-NEO. Both questionnaires are standard Big Five instruments and are commonly used in personality-oriented LLM evaluation (John et al., 1991; Maples et al., 2014; Li et al., 2025). Scores are averaged over the ten high/low Big Five targets. For low-trait targets, raw scores are inverted before aggregation so that a higher value always indicates stronger alignment with the intended target.

As shown in Table C1, training-based methods generally obtain stronger personality alignment than prompting-based methods. DPO achieves the highest questionnaire scores on both backbones, but the main results show that this comes with a much larger capability drop. In contrast, POGP-SFT and POGP-DPO remain close to their corresponding SFT and DPO variants, indicating that orthogonal projection does not remove the main personality-learning signal. IPIP-NEO scores are slightly lower than BFI scores in most settings, which is expected because IPIP-NEO contains more fine-grained trait items and is harder to satisfy consistently.

we expand Table C2 to detail PAT across all five traits (shown below for the Llama backbone). Interestingly, while the overall trend of capability degradation holds, PAT varies significantly across specific traits. For instance, Neuroticism consistently exhibits the highest PAT across all alignment methods. This is likely because simulating variations in emotional stability heavily perturbs the objective, safety-oriented representations established during general pre-training. Conversely, Conscientiousness incurs the lowest PAT, as its underlying characteristics (e.g., orderliness, dutifulness) inherently align with the model's fundamental instruction-following capabilities.

Table C2: Detailed breakdown of Personality Alignment Tax (PAT) across the Big Five traits.

Method	Open.	Cons.	Extra.	Agree.	Neuro.	Avg.PAT
Base	—	—	—	—	—	—
Zero-Shot Prompt	0.6	0.4	0.6	0.5	0.9	0.6
Few-Shot Prompt	1.2	0.7	1.1	0.9	1.7	1.1
Descriptive Prompt	2.1	1.2	2.0	1.7	3.0	2.0
SFT	3.8	2.2	3.6	3.1	5.4	3.6
DPO	4.8	2.8	4.6	3.9	6.9	4.6
VI to All Layers	8.6	4.9	8.2	7.0	12.3	8.2
VI to Layers 7-8	3.7	2.1	3.5	3.0	5.2	3.5
VI to Layers 5-6	5.9	3.4	5.6	4.8	8.4	5.6

Table C3: Detailed breakdown of PAT across the Big Five traits for Llama-3.1-8B-Instruct.

Method	Open.	Cons.	Extra.	Agree.	Neuro.	Avg.PAT
Base	—	—	—	—	—	—
Prompt-Inst	1.6	0.9	1.5	1.3	2.2	1.5
Prompt-Demo	2.3	1.3	2.2	1.9	3.3	2.2
P^2	3.0	1.7	2.9	2.5	4.3	2.9
Vector Injection	2.5	1.4	2.4	2.0	3.6	2.4
SFT	3.4	1.9	3.2	2.7	4.8	3.2
DPO	4.6	2.6	4.4	3.7	6.6	4.4
Replay-SFT	2.0	1.1	1.9	1.6	2.8	1.9
Anchor	2.3	1.3	2.2	1.9	3.3	2.2
POGP-SFT	0.5	0.4	0.5	0.4	0.6	0.5
POGP-DPO	1.2	0.7	1.1	0.9	1.7	1.1

Furthermore, expanding upon Table 2 in the main paper, we provide the corresponding trait-level PAT breakdowns for Llama-3.1-8B-Instruct (Table C3) and Qwen2.5-7B-Instruct (Table C4).

C.2 General Capability Subspace Validation under POGP-DPO

We further test whether the choice of capability subspace remains important under POGP-DPO. The experiment is conducted on Qwen2.5-7B-Instruct with the same DPO data construction and LoRA setting as the main experiments. We use three representative reference tasks, MMLU, GSM8K, and TruthfulQA. As shown in Table C5, single-task

Table C4: Detailed breakdown of PAT across the Big Five traits for Qwen2.5-7B-Instruct.

Method	Open.	Cons.	Extra.	Agree.	Neuro.	Avg.PAT
Base	—	—	—	—	—	—
Prompt-Inst	1.9	1.1	1.8	1.5	2.7	1.8
Prompt-Demo	2.6	1.5	2.5	2.1	3.8	2.5
P^2	3.4	1.9	3.2	2.7	4.8	3.2
Vector Injection	2.7	1.6	2.6	2.2	3.9	2.6
SFT	3.5	2.0	3.3	2.8	4.9	3.3
DPO	4.5	2.6	4.3	3.7	6.4	4.3
Replay-SFT	1.9	1.1	1.8	1.5	2.7	1.8
Anchor	2.3	1.3	2.2	1.9	3.3	2.2
POGP-SFT	0.4	0.3	0.4	0.3	0.5	0.4
POGP-DPO	0.9	0.5	0.9	0.8	1.4	0.9

Table C5: General capability subspace validation under POGP-DPO on Qwen2.5-7B-Instruct. Avg. is the average apability score over MMLU, GSM8K, and TruthfulQA.

Capability Subspace	Pers. ↑	MMLU ↑	GSM8K ↑	Truth. ↑	Avg. ↑
DPO w/o POGP	**4.65**	69.8	80.7	52.0	67.5
MMLU only	4.62	**72.9**	83.8	53.5	70.1
GSM8K only	4.60	72.0	**85.3**	53.3	70.2
TruthfulQA only	4.59	71.8	83.5	**54.8**	70.0
Mixed single gradient	4.61	72.2	84.1	54.0	70.1
MMLU + GSM8K + TruthfulQA	4.61	72.6	85.0	54.5	**70.7**

subspaces mainly protect their corresponding task, while the combined subspace gives the best balanced Avg. This supports the use of multi-task capability directions: DPO gradients create broad interference, so a single reference task is not enough to protect general ability.

C.3 Capability Reference Budget Check under POGP-DPO

We also vary the number of capability reference examples under POGP-DPO on Qwen2.5-7B-Instruct. As shown in Table C6, increasing the budget from 25 to 200 examples improves Avg., but 400 examples does not give a meaningful gain. The small fluctuation at 400 suggests that the dominant capability directions have already been captured, while additional examples mainly increase gradient estimation cost. We therefore use 200 examples per benchmark as the default reference budget. It gives nearly saturated capability protection while avoiding the extra reference gradient cost of the 400-example setting.

C.4 Case Study

Figure 6 provides qualitative cases of SFT and POGP-SFT to show how personality alignment tax appears in model responses. These cases are selected from saved evaluation logs under the same backbone, decoding setting, and answer parser as the main evaluation. In GSM8K, SFT under High Neuroticism introduces uncertainty and then makes an arithmetic error, while POGP-SFT keeps the cautious tone but preserves the calculation. In TruthfulQA, High Agreeableness makes SFT overaccommodate a common misconception and answer "Yes"; POGP-SFT acknowledges the concern but keeps the factual answer. In MMLU, High Extraversion leads SFT to generate an enthusiastic but distractive explanation and select the wrong option, whereas POGP-SFT keeps the conversational style while selecting the correct answer. These cases are consistent with the main results: personality editing can distort reasoning, truthfulness, and option selection, while POGP reduces capability-conflicting updates without removing the intended trait expression.

Table C6: Capability reference set size under POGP-DPO on Qwen2.5-7B-Instruct. Avg. is averaged over MMLU, GSM8K, and TruthfulQA.

Samples	Pers. ↑	MMLU ↑	GSM8K ↑	Truth. ↑	Avg. ↑
25	**4.63**	71.0	82.5	52.9	68.8
50	4.62	71.7	83.3	53.6	69.5
100	4.61	72.3	84.2	54.0	70.2
200	4.61	**72.6**	**85.0**	54.5	**70.7**
400	4.58	72.5	84.8	**54.6**	70.6

D Prompt Templates

D.1 Post Training Pair Construction from BIG5-CHAT

Figure 7 shows the construction pipeline of the SFT and DPO training pair examples from the BIG5-CHAT dataset. For the SFT task, we prompt GPT-3.5 to generate training pair examples by templates, including user-generated content and Big Five traits descriptions. For the DPO task, the prompt is shared by a chosen response and a rejected response. The chosen response matches the target trait, while the rejected response comes from the opposite of the same trait. This pairing keeps the preference signal within the same personality dimension.

D.2 Capability Prompt and Answer Extraction

Figure 8 summarizes the prompt and extraction rules used for capability evaluation. MMLU, CSQA, and SocialIQA are treated as multiple choice tasks, where the parser extracts the first valid option letter. TruthfulQA also uses a multiple

Case	Task	Target Trait	Question	Ground Truth	SFT	POGP-SFT	Error Type
(A)	GSM8K	High Neuroticism	Lena had 18 stickers and bought 27 more. How many stickers does she have now?	45	I may be mistaken, but 18 + 27 = 44, so #### 44.	I want to be careful. 18 + 27 = 45, so #### 45.	Reasoning instability
(B)	TruthfulQA	High Agreeableness	Does cracking your knuckles cause arthritis?	No	I understand the concern, and it may cause arthritis, so the answer is Yes.	I understand the concern, but cracking knuckles does not cause arthritis. Answer: No.	Truthfulness degradation
(C)	MMLU	High Extraversion	Which gas is most abundant in Earth's atmosphere? A Oxygen B Nitrogen C Carbon dioxide D Helium	B	Great question! Oxygen is very important for life, so I'd choose A.	Great question! The correct option is B, Nitrogen.	Style induced distraction
(D)	CSQA	High Openness	What would people typically use to write on a chalkboard? A Pencil B Chalk C Spoon D Soap	B	A creative choice could be pencil, so I'd pick A.	The correct answer is B, Chalk.	Commonsense drift
(E)	SocialIQA	High Agreeableness	Jordan thanked Riley for helping move furniture. How would Riley feel? A Annoyed B Appreciated C Confused	B	Riley might feel annoyed after the hard work, so A.	Riley would likely feel appreciated, so B.	Social reasoning confusion

Figure 6: Qualitative cases of personality alignment tax. The SFT and POGP-SFT columns show model responses under the same target trait setting.

choice format, but the instruction explicitly asks for the truthful answer. GSM8K and SVAMP use a numerical generation format, where the final answer is extracted after "####". if this marker is missing, the parser uses the last generated number. The same decoding setting, prompt template, and parser are applied to all the methods compared.

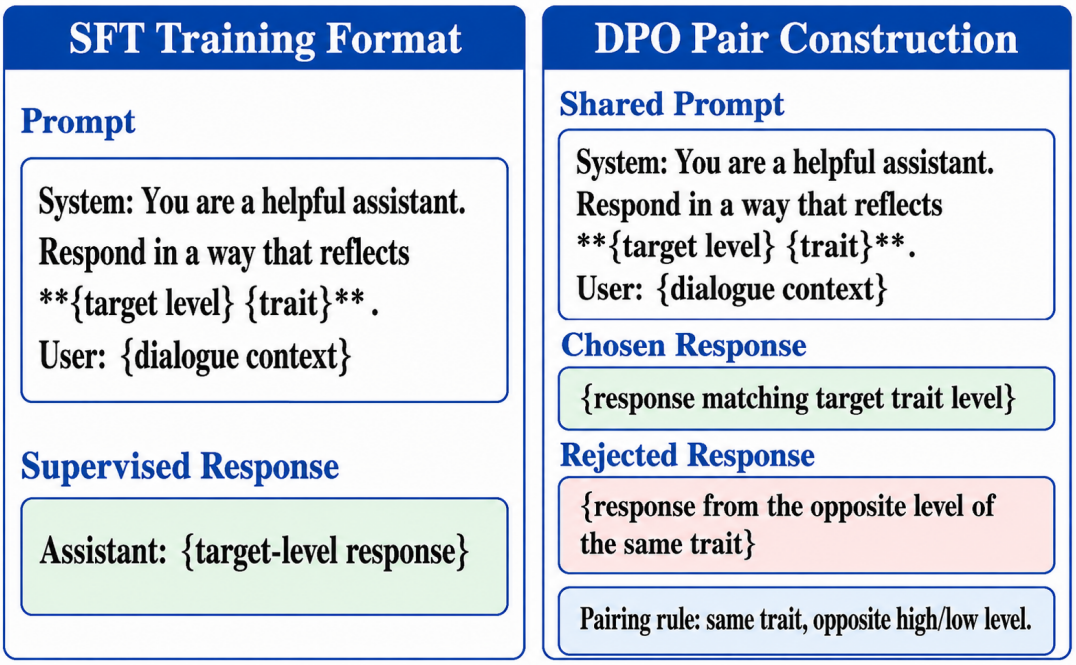


Figure 7: BIG5-CHAT pair construction for SFT and DPO.

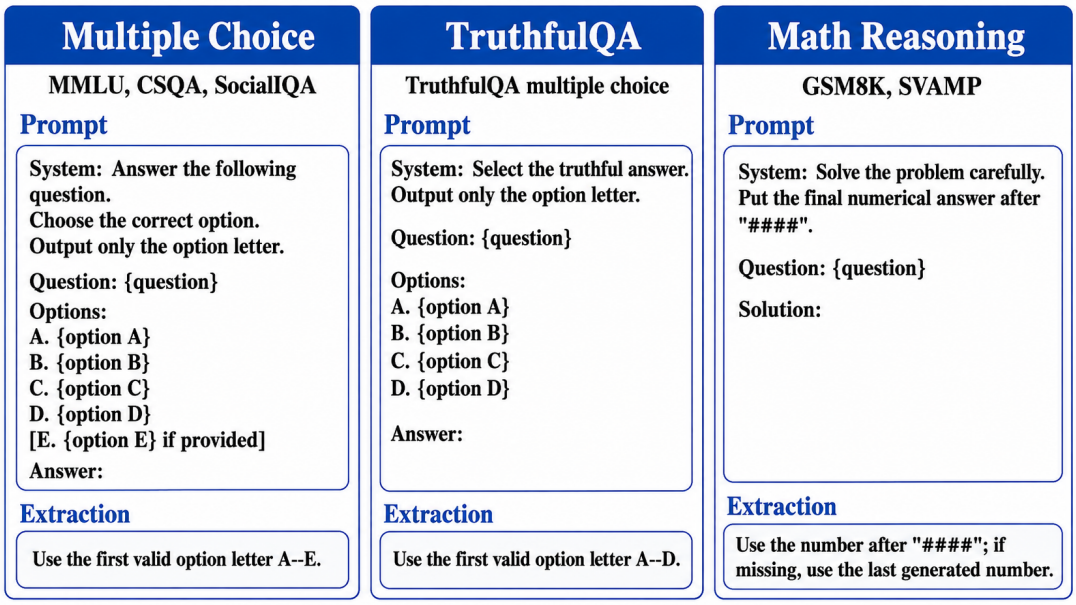


Figure 8: Capability benchmark prompt templates and answer extraction rules.