

PeLA: Prompt-Enhanced Logit Adaptation for Personality-Trait Steering in LLMs

Haoxuan Kou, Wenxuan Wang, Zirui Liu, Jiaxing Shen*

Division of Artificial Intelligence, Lingnan University, Hong Kong, China
{haoxuankou, wenxuanwang3, zirui Liu3, jiaxingshen}@ln.hk

Abstract

Large language models (LLMs) are increasingly used in long-running user-facing applications where responses may need to maintain a consistent interaction style. Prompting is easy to reuse but often weak, while fine-tuning and representation interventions can be stronger but are typically model-specific. We present **PeLA**, Prompt-Enhanced Logit Adaptation, a parameter-efficient framework for Big Five personality-trait steering. PeLA keeps the base LLM frozen, uses a personality prompt to specify the desired trait direction, and applies a learned logit adapter during decoding. The adapter contributes a base-relative residual controlled by a token-position gate, refining the prompt-conditioned distribution only when intervention is useful. Because the control module operates in output-token space, it can be reused across models with identical tokenizer vocabularies and token-id order. On open-ended PersonalityBench questions, PeLA improves macro-average target-direction scores over a strong Personality Prompt baseline on Qwen3-8B and Qwen2.5-7B, while cross-size experiments demonstrate tokenizer-compatible transfer within both the Qwen3 and Llama-3.2 families after validation-based scale calibration. PeLA narrows the gap to model-specific steering while preserving a frozen base model.

1 Introduction

Large language models (LLMs), which generate text by repeatedly predicting the next token, are now used in long-running and user-facing applications such as role-playing agents (Park et al., 2023; Chen et al., 2024), educational assistants (Dong et al., 2024; Shi et al., 2025), game characters (Gallotta et al., 2024), and support-oriented dialogue systems (Na et al., 2025). In such settings, users may care not only about factual accuracy and fluency, but also about whether the system maintains

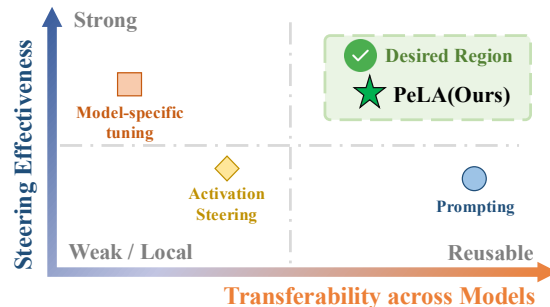


Figure 1: Trade-off between steering effectiveness and transferability. Prompting is easy to reuse but often weak. Fine-tuning can be strong but model-specific. PeLA targets the middle ground by adding a parameter-efficient logit-level control module to a prompted, frozen LLM.

an appropriate interaction style. For example, an assistant used for emotional support may be expected to remain patient, cautious, and empathetic across turns. We refer to the task of pushing a model’s responses toward a desired behavioral style as *personality-trait steering*. In this work, a personality trait is a target dimension from the Big Five model, such as Extraversion or Agreeableness, and a trait direction is one pole of that dimension, such as high Extraversion or low Neuroticism.

A practical difficulty is that new LLMs appear frequently. Prompting is portable but often weak; fine-tuning, model editing, and activation interventions can be stronger but usually require model-specific weights, layers, or calibration, as illustrated in Figure 1. We target a middle ground: reusable decoding-time control for models that expose next-token logits and share an exact output-token space.

We address this problem with **PeLA**, Prompt-Enhanced Logit Adaptation. PeLA uses a personality prompt for semantic guidance and a small *logit adapter* that changes next-token scores during decoding while keeping the base LLM frozen. To

*Corresponding author.

avoid overwriting the base distribution, the adapter adds a base-relative residual and uses a token-position gate to modulate intervention strength. This makes PeLA a prompt-enhancement method rather than a replacement for prompting.

Operating at the logit level enables a limited but useful form of transfer. In tokenizer-compatible settings, source and target models share the same token IDs, vocabulary order, and logit interface, so adapter weights can be reused without target-model fine-tuning. We test this same-family cross-size setting within Qwen3 and do not claim arbitrary transfer across unrelated tokenizers.

We evaluate PeLA on open-ended Personality-Bench questions, comparing prompt-only methods, a model-specific neuron intervention baseline, and supervised fine-tuning in both in-model and tokenizer-compatible cross-size settings. Our contributions are: (1) a formulation of personality-trait steering as prompt-enhanced logit adaptation over a frozen LLM; (2) a base-relative, gated logit adapter designed to refine rather than replace the prompted distribution; and (3) open-ended Big Five experiments showing when same-family logit-adapter transfer helps and where it remains limited.

2 Related Work

2.1 Personality in Large Language Models

Personality theories. Personality psychology provides frameworks for describing stable individual differences in behavior, cognition, and affect. We focus on the Big Five, also known as the Five-Factor Model (Costa and McCrae, 1992; Goldberg, 1992), which represents personality along five broad dimensions: Openness, Conscientiousness, Extraversion, Agreeableness, and Neuroticism. Other personality frameworks, such as the Myers-Briggs Type Indicator (Myers, 1962) and the Dark Triad (Paulhus and Williams, 2002), have also been studied.

LLM personality assessment. LLM psychometrics studies whether psychological instruments can measure personality-like behavioral patterns in LLM outputs (Ye et al., 2025). Prior work adapts Big-Five inventories to LLMs through MPI (Jiang et al., 2023), examines Big-Five-like response patterns across prompts and contexts (Serapio-García et al., 2023; Karra et al., 2022; Caron and Srivastava, 2022), and improves reliability and validity through benchmarks such as TRAIT (Lee et al.,

2025). LLMs have also been used to infer human personality from text or semantic representations (Ji et al., 2023; Cao and Kosinski, 2024; Peters et al., 2024). We measure observable trait expression under a fixed setting, rather than claiming stable psychological personality.

Personality-trait Steering. A direct way to steer LLM personality traits is prompting. Personality Prompting (P^2) (Jiang et al., 2023) uses natural-language trait descriptions, Big5-Scaler (Cho and Cheong, 2025) encodes numeric Big-Five values into prompts, and P4 (Zhang et al., 2024) learns optimized discrete prompt suffixes for personalization. These methods are reusable, but remain limited by prompt following and prompt design.

Stronger control can be obtained through training, adapters, or weight-space modification. BIG5-CHAT (Li et al., 2025b) shapes personalities with human-grounded data; Dynamic Personality Generation (Liu et al., 2024), PsychAdapter (Vu et al., 2026), and P-Tailor (Dan et al., 2024) use training or adapter-based mechanisms; and PersonalityEdit (Mao et al., 2024) and Personality Vector (Sun et al., 2025) edit or merge model parameters to obtain controllable personality directions. These methods can induce strong trait shifts, but typically require model-specific training, weight access, or parameter merging.

Activation- and representation-space methods instead steer behavior through internal states. PAS (Zhu et al., 2025), NPTI (Deng et al., 2025), and ControlLM (Weng et al., 2024) identify personality-related activations, neurons, or differential activation patterns, while more general methods such as representation engineering (Zou et al., 2023), contrastive activation addition (Rimsky et al., 2024), personalized steering vectors (Cao et al., 2024), and persona vectors (Chen et al., 2025) modify hidden states or learned activation directions. Although effective, these methods require internal-layer access and often need layer- or model-specific calibration.

Finally, lexical and probability-based methods steer personality during decoding. UPLex (Li et al., 2025a) uses an unsupervised personality lexicon to adjust upcoming-token probabilities. PeLA instead learns a gated, trait-conditioned residual over the next-token distribution, operating at the logit interface without hidden-state intervention and limiting transfer to tokenizer-compatible models.

2.2 Decoding-Time Controllable Generation

Decoding-time controllable generation steers a frozen or mostly frozen generator during inference. Early methods rely on auxiliary controllers, including attribute classifiers in PPLM (Dathathri et al., 2019), class-conditional language models in GeDi (Krause et al., 2021), expert and anti-expert language models in DExperts (Liu et al., 2021), future discriminators in FUDGE (Yang and Klein, 2021), and unidirectional reward models in RAD (Deng and Raffel, 2023). Although these methods avoid full fine-tuning of the base generator, their control signals typically depend on additional attribute-specific models.

Recent methods move closer to the logit-level interface used by PeLA. PREADD (Pei et al., 2023) and classifier-free guidance for language models (Sanchez et al., 2024) use prompt-contrastive or conditional-unconditional logit differences, while language model arithmetic (Dekoninck et al., 2024) formulates control as arithmetic composition over model distributions. LM-Steer (Han et al., 2024) learns compact steering transformations that can be scaled, composed, and transferred, and logit-only adaptation methods show that token-logit reweighting can adapt closed or black-box models given output logits (Hiranandani et al., 2025; An et al., 2026). PeLA belongs to this logit-level family, but is specialized for Big-Five personality-trait steering by learning a trait-conditioned adapter residual with token-position gating, and it only claims cross-model reuse under tokenizer compatibility.

3 Methodology

3.1 Problem Formulation

Let M be a logits-accessible autoregressive LLM with vocabulary $\mathcal{V}$. Given a question q , a Big Five personality dimension p , and a target direction $d \in \{\text{High}, \text{Low}\}$, our goal is to construct a controlled generator $M_{p,d}^*$. The generator should produce a response that reflects the target trait direction while preserving the general language ability of M .

The dimension p is one of Openness, Conscientiousness, Extraversion, Agreeableness, or Neuroticism. The direction d chooses one pole of that dimension. For example, high Extraversion corresponds to outgoing behavior, while low Extraversion corresponds to more reserved behavior.

We learn a trait-direction-specific control mod-

ule $C_{p,d}$ from supervision $\mathcal{D}_{p,d}$:

$$C_{p,d} = \mathcal{F}(M, \mathcal{D}_{p,d}). \quad (1)$$

The learned module is composed with the frozen LLM at inference time:

$$M_{p,d}^* = M \oplus_{\text{logit}} C_{p,d}. \quad (2)$$

The operator $\oplus_{\text{logit}}$ denotes next-token logit-level control without modifying the parameters of M .

3.2 PeLA Framework

PeLA combines two control signals. The first is an explicit personality prompt, which describes the intended trait direction in natural language. The second is a small trainable logit adapter that refines the prompt-conditioned next-token distribution. The adapter is not used as a standalone personality model. It learns a trait-direction-specific residual from personality-aligned supervision and is combined with personality prompting at inference time. Figure 2 provides an overview of the training, inference, and tokenizer-compatible transfer procedures of PeLA.

3.3 Data Construction

We construct trait-aligned question-answer supervision from open-ended questions in PersonalityBench (Deng et al., 2025). For each Big Five dimension p , we create supervision for both high-trait and low-trait directions. We use a set of generated personality descriptions $\mathcal{S}_{p,d}$ to build teacher prompts (Jiang et al., 2023). Given a question q_i and a sampled description $s_i \sim \mathcal{S}_{p,d}$, the teacher model generates a candidate response:

$$\hat{y}_i^{p,d} \sim T(\cdot \mid \text{Template}(s_i, q_i)). \quad (3)$$

The personality description is used to construct teacher supervision but not included in the adapter training input. We use an external judge model to filter inconsistent responses. The final dataset is

$$\mathcal{D}_{p,d} = \{(q_i, y_i^{p,d})\}_{i=1}^N, \quad (4)$$

where $y_i^{p,d}$ is a retained response. During adapter training, the question tokens provide context, and the loss is computed only on answer tokens.

Data construction and split. For each Big-Five trait, PersonalityBench provides 9,000 data-construction questions. We generated one response for each high/low direction, yielding 9,000 candidates per direction. The number of candidates

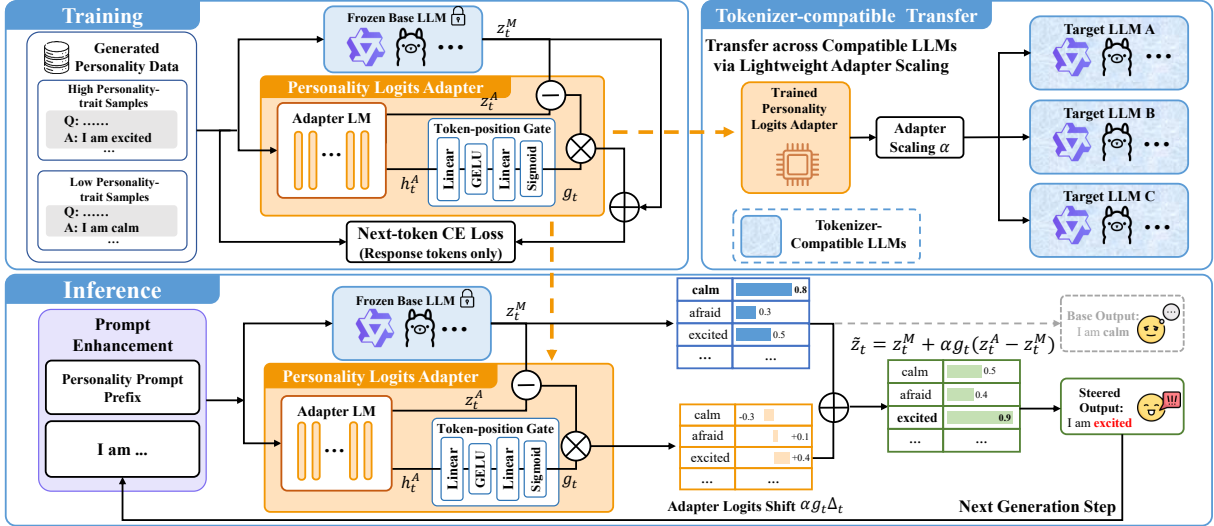


Figure 2: Overview of PeLA. A personality prompt specifies the desired Big Five trait direction. A parameter-efficient logit adapter then adds a gated, base-relative residual to the frozen base model’s next-token logits during decoding. During training, α is fixed to 1; during inference and transfer, α is selected on validation data.

retained after judge filtering is shown in Appendix Table 9. After filtering, we sampled 4,000 examples per direction and split them into 3,600 training and 400 validation examples. Testing uses the official held-out split with 90 disjoint questions per trait, ensuring question-level separation and preventing data leakage. Detailed settings are in Appendix A.

3.4 Token-Position-Gated Personality Logit Adapter

For each trait direction (p, d) , we train a parameter-efficient control module

$$C_{p,d} = (A_{\theta_{p,d}}, G_{\phi_{p,d}}), \quad (5)$$

where $A_{\theta_{p,d}}$ is the adapter language model and $G_{\phi_{p,d}}$ is the token-position gate. The adapter shares the tokenizer and output vocabulary with the base model, but uses a much smaller decoder architecture. The base model remains frozen.

At decoding step t , the frozen base model produces next-token logits

$$z_t^M = M(x_{<t}), \quad z_t^M \in \mathbb{R}^{|V|}. \quad (6)$$

The adapter receives the same prefix as the base model at the corresponding training or inference stage and produces its own logits and hidden state:

$$(z_t^A, h_t^A) = A_{\theta_{p,d}}(x_{<t}), \quad z_t^A \in \mathbb{R}^{|V|}. \quad (7)$$

During training, this shared prefix consists of the question and preceding answer tokens, whereas

during inference it includes the personality-conditioned prompt and previously generated tokens, as detailed in Sections 3.5 and 3.6.

A fixed correction at every decoding step can over-control the response. We therefore compute a scalar gate for each generated position:

$$g_t = \sigma(W_2 \text{GELU}(W_1 h_t^A + b_1) + b_2), \quad (8)$$

$$g_t \in (0, 1).$$

Here, GELU denotes the Gaussian Error Linear Unit activation function, and g_t controls the adapter strength at position t . We initialize the final gate bias as $b_2 = \text{logit}(g_0)$ with $g_0 = 0.2$, so training starts near the frozen base model.

To avoid using the adapter as a replacement language model, we define a base-relative residual:

$$\Delta_t = z_t^A - z_t^M. \quad (9)$$

The controlled logits are

$$\tilde{z}_t = z_t^M + \alpha g_t \Delta_t, \quad (10)$$

where α is the adapter scale. It calibrates the residual contribution relative to the base logits and is set to 1 during training.

3.5 Training Objective

Only the adapter parameters $\theta_{p,d}$ and gate parameters $\phi_{p,d}$ are optimized. For each example, we concatenate the question and answer into a causal language-modeling sequence, without including an

explicit personality prompt. Thus, during training, both the frozen base model and the adapter receive the same prefix consisting of the question and preceding answer tokens. Question and padding tokens are masked, so the loss is computed only on answer-token positions.

Let $\mathcal{A}_i$ be the set of answer-token positions for example i , and let $\mathcal{A} = \bigcup_i \mathcal{A}_i$. The answer-only cross-entropy loss is

$$\mathcal{L}_{\text{CE}} = -\frac{1}{|\mathcal{A}|} \sum_{i=1}^N \sum_{t \in \mathcal{A}_i} \log \text{softmax}(\tilde{z}_{i,t})_{y_{i,t}}. \quad (11)$$

We regularize the average gate activation

$$\bar{g} = \frac{1}{|\mathcal{A}|} \sum_{i=1}^N \sum_{t \in \mathcal{A}_i} g_{i,t}. \quad (12)$$

The final objective is

$$\mathcal{L} = \mathcal{L}_{\text{CE}} + \lambda_g \bar{g}. \quad (13)$$

We set $\lambda_g = 0.01$ and keep it fixed across trait directions before all experiments.

3.6 Personality-Conditioned Inference

At inference time, PeLA combines a personality prompt with the trained adapter for the target trait direction. Given q and (p, d) , we sample $s \sim \mathcal{S}_{p,d}$ once using a fixed random seed and construct the personality-conditioned prompt $\pi_{p,d}(q; s)$. The sampled profile s is then kept fixed throughout evaluation: Personality Prompt and PeLA use exactly the same profile for each question, and the same profile is also reused across all candidate steering scales. Unlike training, where no explicit personality prompt is included, at inference time both the base model and the adapter receive the same personality-conditioned prefix, consisting of $\pi_{p,d}(q; s)$ and the previously generated tokens. This train–inference difference is intentional: the adapter learns a trait-direction-specific residual from the training data, while the personality prompt provides explicit semantic conditioning at inference time. Using the same prompted prefix for both components also ensures that the adapter residual is computed relative to the base distribution under the same context. Generation then proceeds autoregressively. At each step, the base model supplies the prompted base distribution, the adapter supplies a residual correction, and the next token is

selected from Eq. 10. We use greedy decoding for deterministic comparison:

$$x_t = \arg \max_{v \in \mathcal{V}} \text{softmax}(\tilde{z}_t)_v. \quad (14)$$

3.7 Cross-Model Transfer with Adapter Scaling

PeLA is designed for tokenizer-compatible transfer. Let M_s be the source model and M_t be the target model. Direct logit-level reuse requires the same tokenizer and vocabulary order:

$$\text{Tok}(M_s) = \text{Tok}(M_t), \quad \mathcal{V}_{M_s} = \mathcal{V}_{M_t}. \quad (15)$$

This ensures that adapter and target-model logits share the same token indices.

After training $C_{p,d}$ with the source model, we plug the same adapter into the target model. The target model remains frozen. At decoding step t , the target model produces logits $z_t^{M_t}$, and the transferred adapter produces (z_t^A, h_t^A) . The controlled logits are

$$\tilde{z}_t^{M_t} = z_t^{M_t} + \alpha_{m,p,d} g_t \left(z_t^A - z_t^{M_t} \right), \quad (16)$$

where $\alpha_{m,p,d}$ is the adapter scale for model M_m , dimension p , and direction d . We select the scale on validation data and keep it fixed for test evaluation.

4 Experiments

4.1 Experimental Setup

Base LLMs. We evaluate PeLA on Qwen2.5-7B (Qwen et al., 2025) and Qwen3-8B (Yang et al., 2025). For in-model steering, each adapter is trained and evaluated on the same base model. For cross-size transfer, we evaluate both large-to-small and small-to-large settings. In the large-to-small setting, we train the adapter on Qwen3-8B and reuse the learned adapter weights on Qwen3-4B and Qwen3-1.7B. In the reverse small-to-large setting, we train the adapter on Qwen3-1.7B and transfer it to Qwen3-4B and Qwen3-8B. In both settings, we tune only the scalar scale on target-model validation data, while keeping all base and target LLMs, as well as the transferred adapter parameters, frozen during inference. PeLA transfers by reusing the learned adapter weights and forms a target-relative residual during decoding on target models, which requires identical tokenizer vocabularies and token-id ordering between source and target models. We verified this condition for

Qwen3-8B, Qwen3-4B, and Qwen3-1.7B by comparing their complete Hugging Face id-to-token mappings, including special tokens. All three local checkpoints use the same Qwen3 tokenizer with a vocabulary size of 151,669 and identical id-to-token mappings. We do not claim transfer across models with different tokenizers. Additional details are provided in Appendix A.

Adapter models. For each model family, we instantiate a compact decoder-only adapter with the same tokenizer and output vocabulary as the source model. The adapter architecture is reported in Appendix A. The adapter produces next-token logits and a final-layer hidden state used by the gate. A Qwen3-style adapter has 162.9M trainable parameters, corresponding to 1.99% of the Qwen3-8B source model.

Traits and directions. We evaluate all five Big Five dimensions. For each dimension, we train separate adapters for the high-trait and low-trait directions.

Baselines. We compare PeLA with four baselines:

- **Simple Prompt:** a short natural-language instruction that names the target trait.
- **Personality Prompt (P^2)** (Jiang et al., 2023): a generated facet-based target-trait profile is included in the prompt.
- **NPTI** (Deng et al., 2025): a model-specific inference-time neuron intervention method. It was reimplemented from released code.
- **Supervised Fine-tuning:** a LoRA-based fine-tuning baseline (Hu et al., 2022) trained for a specific personality direction. Detailed LoRA settings are provided in Appendix A.

Decoding. All open-ended generations use greedy decoding. We use the same maximum generation length and stopping criteria for all methods. Detailed decoding settings are in Appendix A.

4.2 Evaluation Setting

Open-ended personality evaluation. We evaluate with the open-ended test split of Personality-Bench (Deng et al., 2025). These questions require free-form responses that express preferences, emotions, interpersonal behavior, and decision-making style. This setting is more realistic than multiple-choice trait questionnaires, but it also makes the evaluation protocol more important.

Personality score. A fixed external LLM judge scores each generated response on a 1 to 5 scale for the target trait p ; the exact deployment, access date, decoding settings, and prompt templates are reported in Appendix B. A higher raw score means stronger expression of that trait. For high-trait directions, higher scores indicate better target alignment. For low-trait directions, lower scores indicate better target alignment. The raw trait score is

$$\text{Score}_p = \frac{1}{|\mathcal{Q}|} \sum_{q \in \mathcal{Q}} \text{Judge}_{\text{trait}}(q, y_q, p), \quad (17)$$

where $\mathcal{Q}$ is the test-question set and y_q is the generated answer.

Fluency score. The judge also assigns a fluency score from 1 to 5. This score measures grammaticality, coherence, and readability:

$$\text{Fluency} = \frac{1}{|\mathcal{Q}|} \sum_{q \in \mathcal{Q}} \text{Judge}_{\text{fluency}}(q, y_q). \quad (18)$$

We evaluate personality expression and fluency separately so that stronger trait expression is not confused with response quality. Because open-ended trait expression cannot be fully captured by lexical matching, we use an LLM judge only under a fixed evaluation protocol. Detailed judge model settings are shown in Appendix B and full judge prompt templates are shown in Appendix B.

Judge Robustness. To assess the robustness of LLM-based evaluation, we re-score all generated responses with a second independent judge LLM using the same rubric and protocol; Appendix B reports the model identifiers and inter-judge agreement.

Perplexity. We additionally report perplexity (PPL) as a supplementary measure of generation quality. Detailed settings are provided in Appendix B. We use Qwen3-8B as the scorer for all methods.

4.3 Adapter Scale Calibration

The adapter scale α controls the strength of the residual in Eq. 10. We treat it as a validation-selected calibration parameter, not as a monotonic personality-strength knob. The interaction among base logits, adapter logits, the gate, and greedy decoding can make the effect non-monotonic.

Let $\mathcal{G}$ be a predefined candidate set. For each $\alpha \in \mathcal{G}$, we generate validation responses and

compute target alignment and fluency. Because raw trait scores have opposite interpretations for high and low directions, we define a direction-normalized target score. Let S_p be the raw personality score and $T_{p,d}$ be the target-direction score:

$$\begin{aligned} T_{p,d} &= S_p, & d &= \text{High}, \\ T_{p,d} &= 6 - S_p, & d &= \text{Low}. \end{aligned} \quad (19)$$

Higher $T_{p,d}$ always means better alignment with the target direction.

We select the scale that maximizes validation target alignment while preserving fluency:

$$\alpha^* = \arg \max_{\alpha \in \mathcal{G}} T_{\text{val}}(\alpha), \quad (20)$$

subject to

$$F_{\text{val}}(\alpha) \geq F_{\text{val}}(P^2 \text{ Prompt}) - \epsilon. \quad (21)$$

We set $\epsilon = 0.1$ before test evaluation. The selected scale is fixed for the test set. We report candidate set $\mathcal{G}$ and selected scales in Appendix A.

4.4 Tokenizer-Compatible Transfer Evaluation

We evaluate bidirectional adapter-weight transfer with target-side scale calibration within the Qwen3 family. In the large-to-small setting, adapter weights trained on Qwen3-8B are reused on Qwen3-4B and Qwen3-1.7B. In the reverse small-to-large setting, adapter weights trained on Qwen3-1.7B are reused on Qwen3-4B and Qwen3-8B. In both settings, neither the transferred adapter nor the target-model parameters are updated; only the scalar scale α is selected on target-model validation generations. These experiments test same-family tokenizer-compatible transfer only and do not evaluate transfer across unrelated model families or different tokenizers.

5 Results and Discussion

5.1 Main Results

Table 1 reports raw personality scores on the PersonalityBench open-ended test set. Lower scores are better for low-trait directions and higher scores are better for high-trait directions. Compared with the Personality Prompt baseline, PeLA improves the Qwen3-8B macro average from 1.46 to 1.29 on low-trait directions and from 4.49 to 4.57 on high-trait directions. On Qwen2.5-7B, PeLA improves the macro average from 2.10 to 1.83 and from 4.29

to 4.54, respectively. These macro results are the clearest evidence for PeLA’s effectiveness.

The gains are direction- and trait-dependent. PeLA is especially helpful for low-trait steering and Extraversion, while High Neuroticism and High Openness show limited or negative gains in some settings. Appendix Table 20 gives paired bootstrap confidence intervals against the Personality Prompt baseline; under this test, PeLA significantly improves 7 of 10 Qwen3-8B trait directions. Appendix C reports fluency and PPL, and Appendix Table 14 reports high inter-judge consistency between the two LLM judges.

Model-specific baselines remain strong: NPTI and supervised fine-tuning achieve the best raw score in several settings. We therefore position PeLA as a transferable, frozen-base alternative that narrows the gap to model-specific customization, not as a universal replacement for fine-tuning or internal intervention.

5.2 Human Validation

We further conducted a human validation study to assess the reliability of the LLM judge on PeLA-generated responses. We stratified and sampled 300 responses generated by Qwen3-8B with PeLA on the test set, covering all five personality traits in both high and low directions, with 30 samples per trait–direction combination. Three human annotators, all PhD students, independently rated the strength of the expressed personality trait. To ensure consistency with the LLM-based evaluation, the annotators followed the same facet-level personality scoring rubric used by the LLM judge. As shown in Table 2, the LLM judge shows strong consistency with human judgments on the PeLA output distribution, achieving high Pearson and Spearman correlations and high within-one agreement rates. This validation assesses the reliability of the judge for PeLA-generated responses and should not be interpreted as an independent human verification of PeLA’s comparative ranking against all baselines.

5.3 Cross-Size Transfer Results

Tables 3 and 4 report bidirectional cross-size transfer within the Qwen3 family. For large-to-small transfer, reusing adapters trained on Qwen3-8B improves the Personality Prompt macro average on Qwen3-1.7B from 1.85 to 1.35 for low-trait directions and from 4.31 to 4.46 for high-trait directions. On Qwen3-4B, the corresponding macro averages improve from 1.59 to 1.25 and from 4.33

Table 1: Personality-trait steering results across models, trait dimensions, target directions, and macro averages.

Model	Method	Openness		Conscientiousness		Extraversion		Agreeableness		Neuroticism		Macro Avg	
		Low(↓)	High(↑)	Low(↓)	High(↑)	Low(↓)	High(↑)	Low(↓)	High(↑)	Low(↓)	High(↑)	Low(↓)	High(↑)
Qwen3-8B	<i>Default Profile</i>	3.50		3.16		2.83		3.81		3.50		3.36	3.36
	Simple Prompt	3.22	3.64	1.37	3.62	1.42	4.38	1.39	4.30	2.55	4.83	1.99	4.15
	Personality Prompt	1.54	4.26	1.36	4.26	1.28	4.67	1.65	4.60	1.49	4.68	1.46	4.49
	PeLA (Ours)	1.32	4.24	1.15	4.40	1.21	4.83	1.42	4.72	1.33	4.67	1.29	4.57
	NPTI	1.21	4.43	1.09	4.26	1.01	4.92	1.10	4.35	1.00	5.00	1.08	4.59
	Supervised Fine-tuning	1.23	4.44	1.30	4.60	1.33	4.71	1.46	4.79	1.05	4.42	1.27	4.59
Qwen2.5-7B	<i>Default Profile</i>	3.78		3.33		3.34		3.88		3.36		3.54	3.54
	Simple Prompt	3.36	3.72	2.69	4.07	1.74	4.49	2.19	4.42	2.54	4.62	2.50	4.26
	Personality Prompt	2.34	4.11	2.34	4.00	1.73	4.69	2.48	4.50	1.63	4.13	2.10	4.29
	PeLA (Ours)	2.37	4.41	2.34	4.43	1.24	4.94	1.56	4.58	1.63	4.33	1.83	4.54
	NPTI	1.91	4.60	2.56	3.98	1.97	4.77	1.62	4.47	1.10	4.78	1.83	4.52
	Supervised Fine-tuning	1.81	4.58	1.27	4.60	1.38	4.72	1.44	4.82	1.04	4.40	1.39	4.62

Note: *Default Profile* denotes the unsteered base-model score and spans the Low and High columns for each trait. Raw trait scores are reported: lower is better for Low(↓) directions, and higher is better for High(↑) directions. The two rightmost columns give macro averages over the five traits. Macro averages are computed from unrounded trait scores. Color intensity is normalized within each model and target direction, with darker blue indicating a larger target-directed shift.

Table 2: Agreement between the LLM judge and each human annotator.

Metric	Annotator A	Annotator B	Annotator C
Pearson correlation	0.860	0.807	0.803
Spearman correlation	0.856	0.803	0.801
Exact agreement	64.67%	56.00%	54.67%
Within-one agreement	96.67%	93.33%	92.67%

to 4.40. We further evaluate the reverse direction by training the adapters on Qwen3-1.7B and transferring them to larger models. On Qwen3-4B, the macro average improves from 1.59 to 1.52 for low-trait directions and from 4.33 to 4.64 for high-trait directions. When transferred further to Qwen3-8B, the corresponding scores improve from 1.46 to 1.42 and from 4.49 to 4.75. These results demonstrate that PeLA supports bidirectional cross-size transfer: adapters trained on either a larger or a smaller source model can be reused across tokenizer-compatible models of different scales without updating the target-model or adapter parameters, requiring only validation-based calibration of the steering scale. In Appendix Table 21, we further conducted bidirectional transfer experiments within the Llama-3.2 family, demonstrating that PeLA’s bidirectional cross-size transferability generalizes beyond a single model family.

This experiment is target-side calibrated transfer, not zero-touch transfer, because the scalar scale α is selected with target-model validation generations. It also does not establish universal transfer across models with different tokenizers, different vocabulary order, or inaccessible logits.

5.4 Ablation Study

We study how prompting and logit adaptation interact in the High Extraversion setting. The ablation values are averaged over all test questions and use the same validation-selected scale. Table 5 compares the base model, prompt-only steering, adapter-only steering, and the full Prompt+Adapter setting. Prompt-only steering is already strong. Adapter-only steering remains close to the base model. The full method performs best in this setting. This supports the main design choice: the adapter is a prompt-conditioned refinement module rather than an independent personality controller.

Table 6 analyzes where the personality prompt is applied. Prompting only the base model retains most of the steering effect, whereas prompting only the adapter weakens control, suggesting that the frozen base LLM provides the main semantic personality conditioning and the adapter refines the next-token distribution.

We further ablate the base-relative residual and token-position gate to assess the logit-adapter design. As shown in Appendix Figure 5, removing these components degrades response fluency, and the ungated variant becomes more sensitive to the adapter scale. This indicates that the base-relative residual anchors the intervention to the base distribution, while the token-position gate reduces over-intervention by applying the residual selectively.

5.5 Effect of Adapter Scale

We analyze the effect of the adapter scale α on target-trait expression and fluency. Because α controls the contribution of the adapter-induced logit

Table 3: Large-to-small transfer of Qwen3-8B-trained adapters to Qwen3-1.7B and Qwen3-4B.

Model	Method	Openness		Conscientiousness		Extraversion		Agreeableness		Neuroticism		Macro Avg	
		Low(↓)	High(↑)	Low(↓)	High(↑)	Low(↓)	High(↑)	Low(↓)	High(↑)	Low(↓)	High(↑)	Low(↓)	High(↑)
Qwen3-1.7B	Personality Prompt	2.28	3.92	1.37	4.63	2.06	4.20	1.94	4.62	1.62	4.18	1.85	4.31
	PeLA (Ours)	1.40	4.08	1.35	4.40	1.26	4.75	1.48	4.67	1.28	4.40	1.35	4.46
Qwen3-4B	Personality Prompt	1.47	4.04	1.49	4.04	1.28	4.46	2.08	4.49	1.65	4.61	1.59	4.33
	PeLA (Ours)	1.11	4.03	1.24	4.20	1.22	4.76	1.28	4.48	1.42	4.53	1.25	4.40

Table 4: Small-to-large transfer of Qwen3-1.7B-trained adapters to Qwen3-4B and Qwen3-8B.

Model	Method	Openness		Conscientiousness		Extraversion		Agreeableness		Neuroticism		Macro Avg	
		Low(↓)	High(↑)	Low(↓)	High(↑)	Low(↓)	High(↑)	Low(↓)	High(↑)	Low(↓)	High(↑)	Low(↓)	High(↑)
Qwen3-4B	Personality Prompt	1.47	4.04	1.49	4.04	1.28	4.46	2.08	4.49	1.65	4.61	1.59	4.33
	PeLA (Ours)	1.40	4.36	1.42	4.33	1.20	4.82	2.02	4.93	1.60	4.78	1.52	4.64
Qwen3-8B	Personality Prompt	1.54	4.26	1.36	4.26	1.28	4.67	1.65	4.60	1.49	4.68	1.46	4.49
	PeLA (Ours)	1.44	4.56	1.38	4.42	1.26	4.89	1.62	4.96	1.40	4.91	1.42	4.75

Note: Both subtables follow the same evaluation settings as Table 1. Lower values are better for Low (↓) directions, while higher values are better for High (↑) directions. Macro averages are computed from unrounded trait scores.

Table 5: Prompt-adapter ablation in High Extraversion.

Variant	Score ↑
<i>Base LLM</i>	2.83
Prompt-only	4.67
Adapter-only	2.80
Prompt+Adapter	4.83

Table 6: Diagnostic ablation of prompt routing in High Extraversion.

Variant	Score ↑
<i>Base LLM</i>	2.83
P^2 prompt to base LLM only	4.68
P^2 prompt to adapter only	3.02
P^2 prompt to base LLM and adapter	4.83

residual, we treat it as a calibration parameter rather than a monotonic personality-strength coefficient.

Appendix Figure 3 illustrates the scale effect in High Conscientiousness. The dashed line denotes the personality-prompt baseline, and the shaded region marks the selected scale range. Small α values yield scores close to the prompt-only baseline, indicating limited adapter effect. Moderate scales improve target-trait expression while preserving fluency, whereas overly large scales sharply reduce both, suggesting over-intervention in the output distribution. This diagnostic motivates validation-based scale selection rather than manually increasing α at test time.

5.6 Gate Behavior Analysis

Appendix Figure 4 visualizes the token-position gate on one low-Extraversion example. The ex-

ample shows larger gate values around phrases expressing reserved social behavior and smaller values around function words or task-neutral tokens. We treat this as a qualitative interpretability check rather than causal proof of gate behavior.

6 Conclusion

We presented PeLA, a prompt-enhanced logit adaptation framework for personality-trait steering. PeLA keeps the base LLM frozen, uses a personality prompt to establish the desired Big Five trait direction, and applies a parameter-efficient logit adapter during decoding. The adapter combines a base-relative residual with a token-position gate, allowing it to refine the prompted distribution without replacing the base model. Experiments on open-ended PersonalityBench questions show macro-average improvements over a strong Personality Prompt baseline and demonstrate same-family transfer under tokenizer compatibility. The results also clarify the method’s limits: model-specific baselines remain competitive, transfer requires exact vocabulary alignment and validation calibration, and the current evaluation depends on LLM judges.

Limitations

PeLA requires access to output logits. It therefore cannot be applied directly to closed systems that expose only sampled text. Direct adapter reuse also requires exact tokenizer and vocabulary-order compatibility between the source and target models.

The method is a prompt-enhanced refinement framework. The adapter-only ablation shows lim-

ited steering, so the personality prompt remains central. This is a strength for interpretability but a limitation for settings where explicit personality prompts are undesirable or where users should not see trait instructions.

Our training data construction and evaluation both rely on LLM-generated or LLM-judged signals. This makes large-scale open-ended evaluation practical, but the scores may depend on the teacher model, filtering judge, evaluation judge, judge prompt, output ordering, and sampling settings. A larger human-evaluation subset and more non-LLM automatic checks would make the conclusions stronger.

GPT-5.4 is used both to filter synthetic training responses and as the primary evaluator at test time, which may introduce judge-specific selection bias. We partially mitigate this concern with a secondary GPT-5 judge and human validation, both of which show strong agreement with the primary judge. However, these checks reduce rather than eliminate the potential bias introduced during training-data filtering.

Finally, stronger trait steering is not always desirable. Pushing responses toward low Agreeableness, high Neuroticism, or other sensitive trait directions could produce harmful or manipulative interaction styles in deployment. PeLA should therefore be used with clear disclosure, user consent, and safety constraints around allowed traits and contexts.

Ethical Considerations

Personality-trait steering can improve consistency and user experience, but it can also be misused to create systems that manipulate users or simulate personal traits without disclosure. The results in this paper should be interpreted as control over generated behavior, not evidence that an LLM has human personality or stable psychological traits. Any deployment should disclose the use of personality steering, respect user consent, monitor for harmful over-steering, and avoid using trait control to exploit vulnerable users or intensify distress.

Acknowledgments

This work has benefited from the financial support of Lingnan University (DR26F1, 784003).

References

Hyeseon An, Shinwoo Park, Hyundong Jin, and Yo-Sub Han. 2026. Steering language models before

they speak: Logit-level interventions. *arXiv preprint arXiv:2601.10960*.

Xubo Cao and Michal Kosinski. 2024. Large language models know how the personality of public figures is perceived by the general public. *Scientific Reports*, 14(1):6735.

Yuanpu Cao, Tianrong Zhang, Bochuan Cao, Ziyi Yin, Lu Lin, Fenglong Ma, and Jinghui Chen. 2024. Personalized steering of large language models: Versatile steering vectors through bi-directional preference optimization. *Advances in Neural Information Processing Systems*, 37:49519–49551.

Graham Caron and Shashank Srivastava. 2022. Identifying and manipulating the personality traits of language models. *arXiv preprint arXiv:2212.10276*.

Nuo Chen, Yan Wang, Yang Deng, and Jia Li. 2024. The Oscars of AI theater: A survey on role-playing with language models. *arXiv preprint arXiv:2407.11484*.

Runjin Chen, Andy Arditi, Henry Sleight, Owain Evans, and Jack Lindsey. 2025. Persona vectors: Monitoring and controlling character traits in language models. *arXiv preprint arXiv:2507.21509*.

Gunhee Cho and Yun-Gyung Cheong. 2025. Scaling personality control in LLMs with Big Five scaler prompts. *arXiv preprint arXiv:2508.06149*.

Paul T Costa and Robert R McCrae. 1992. Normal personality assessment in clinical practice: The NEO personality inventory. *Psychological Assessment*, 4(1):5.

Yuhao Dan, Jie Zhou, Qin Chen, Junfeng Tian, and Liang He. 2024. P-Tailor: Customizing personality traits for language models via a mixture of specialized LoRA experts. *arXiv preprint arXiv:2406.12548*.

Sumanth Dathathri, Andrea Madotto, Janice Lan, Jane Hung, Eric Frank, Piero Molino, Jason Yosinski, and Rosanne Liu. 2019. Plug and play language models: A simple approach to controlled text generation. *arXiv preprint arXiv:1912.02164*.

Jasper Dekoninck, Marc Fischer, Luca Beurer-Kellner, and Martin Vechev. 2024. [Controlled text generation via language model arithmetic](#). In *International Conference on Learning Representations*.

Haikang Deng and Colin Raffel. 2023. Reward-augmented decoding: Efficient controlled text generation with a unidirectional reward model. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 11781–11791.

Jia Deng, Tianyi Tang, Yanbin Yin, Wenhao Yang, Xin Zhao, and Ji-Rong Wen. 2025. [Neuron based personality trait induction in large language models](#). In *International Conference on Learning Representations*.

- Bingyu Dong, Jie Bai, Tao Xu, and Yun Zhou. 2024. Large language models in education: A systematic review. In *2024 6th International Conference on Computer Science and Technologies in Education (CSTE)*, pages 131–134. IEEE.
- Roberto Gallotta, Graham Todd, Marvin Zammit, Sam Earle, Antonios Liapis, Julian Togelius, and Georgios N Yannakakis. 2024. Large language models and games: A survey and roadmap. *IEEE Transactions on Games*.
- Lewis R Goldberg. 1992. The development of markers for the Big-Five factor structure. *Psychological Assessment*, 4(1):26.
- Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, and 1 others. 2024. The Llama 3 herd of models. *arXiv preprint arXiv:2407.21783*.
- Chi Han, Jialiang Xu, Manling Li, Yi Fung, Chenkai Sun, Nan Jiang, Tarek Abdelzaher, and Heng Ji. 2024. Word embeddings are steers for language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 16410–16430.
- Gaurush Hiranandani, Haolun Wu, Subhojyoti Mukherjee, and Sanmi Koyejo. 2025. Logits are all we need to adapt closed models. *arXiv preprint arXiv:2502.06806*.
- Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Liang Wang, and Weizhu Chen. 2022. *LoRA: Low-rank adaptation of large language models*. In *International Conference on Learning Representations*.
- Yu Ji, Wen Wu, Hong Zheng, Yi Hu, Xi Chen, and Liang He. 2023. Is ChatGPT a good personality recognizer? A preliminary study. *arXiv preprint arXiv:2307.03952*.
- Guangyuan Jiang, Manjie Xu, Song-Chun Zhu, Wenjuan Han, Chi Zhang, and Yixin Zhu. 2023. Evaluating and inducing personality in pre-trained language models. *Advances in Neural Information Processing Systems*, 36:10622–10643.
- Saketh Reddy Karra, Son The Nguyen, and Theja Tulabandhula. 2022. Estimating the personality of white-box language models. *arXiv preprint arXiv:2204.12000*.
- Ben Krause, Akhilesh Deepak Gotmare, Bryan McCann, Nitish Shirish Keskar, Shafiq Joty, Richard Socher, and Nazneen Fatema Rajani. 2021. GeDi: Generative discriminator guided sequence generation. In *Findings of the Association for Computational Linguistics: EMNLP 2021*, pages 4929–4952.
- Seungbeen Lee, Seungwon Lim, Seungju Han, Giyeong Oh, Hyungjoo Chae, Jiwan Chung, Minju Kim, Beong-woo Kwak, Yeonsoo Lee, Dongha Lee, Jinyoung Yeo, and Youngjae Yu. 2025. *Do LLMs have distinct and consistent personality? TRAIT: Personality testset designed for LLMs with psychometrics*. In *Findings of the Association for Computational Linguistics: NAACL 2025*, pages 8412–8452, Albuquerque, New Mexico. Association for Computational Linguistics.
- Tianlong Li, Wenhao Liu, Muling Wu, Shihan Dou, Zhenghua Wang, Changze Lv, Xiaohua Wang, Xiaoqing Zheng, and Xuanjing Huang. 2025a. *UPLex: Fine-grained personality control in large language models via unsupervised lexical modulation*. In *Findings of the Association for Computational Linguistics: EMNLP 2025*, pages 4122–4136, Suzhou, China. Association for Computational Linguistics.
- Wenkai Li, Jiarui Liu, Andy Liu, Xuhui Zhou, Mona Diab, and Maarten Sap. 2025b. 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.
- Alisa Liu, Maarten Sap, Ximing Lu, Swabha Swayamdipta, Chandra Bhagavatula, Noah A Smith, and Yejin Choi. 2021. DExperts: Decoding-time controlled text generation with experts and anti-experts. 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)*, pages 6691–6706.
- Jianzhi Liu, Hexiang Gu, Tianyu Zheng, Liuyu Xiang, Huijia Wu, Jie Fu, and Zhaofeng He. 2024. Dynamic generation of personalities with large language models. *arXiv preprint arXiv:2404.07084*.
- Shengyu Mao, Xiaohan Wang, Mengru Wang, Yong Jiang, Pengjun Xie, Fei Huang, and Ningyu Zhang. 2024. Editing personality for large language models. In *CCF International Conference on Natural Language Processing and Chinese Computing*, pages 241–254. Springer.
- Isabel Briggs Myers. 1962. *The Myers-Briggs Type Indicator*, volume 34. Consulting Psychologists Press Palo Alto, CA.
- Hongbin Na, Yining Hua, Zimu Wang, Tao Shen, Beibei Yu, Lilin Wang, Wei Wang, John Torous, and Ling Chen. 2025. A survey of large language models in psychotherapy: Current landscape and future directions. In *Findings of the Association for Computational Linguistics: ACL 2025*, pages 7362–7376.
- Joon Sung Park, Joseph O’Brien, Carrie Jun 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 1–22.

- Delroy L Paulhus and Kevin M Williams. 2002. The dark triad of personality: Narcissism, Machiavellianism, and psychopathy. *Journal of Research in Personality*, 36(6):556–563.
- Jonathan Pei, Kevin Yang, and Dan Klein. 2023. PREADD: Prefix-adaptive decoding for controlled text generation. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 10018–10037.
- Heinrich Peters, Moran Cerf, and Sandra C Matz. 2024. Large language models can infer personality from free-form user interactions. *arXiv preprint arXiv:2405.13052*.
- Qwen, An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoran Wei, Huan Lin, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Yang, Jiaxi Yang, Jingren Zhou, Junyang Lin, and 24 others. 2025. [Qwen2.5 technical report](#). *Preprint*, arXiv:2412.15115.
- Nina Rimskey, Nick Gabrieli, Julian Schulz, Meg Tong, Evan Hubinger, and Alexander Turner. 2024. Steering Llama 2 via contrastive activation addition. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 15504–15522.
- Guillaume V Sanchez, Alexander Spangher, Honglu Fan, Elad Levi, and Stella Biderman. 2024. Stay on topic with classifier-free guidance. In *Proceedings of the 41st International Conference on Machine Learning*, pages 43197–43234.
- Greg Serapio-García, Mustafa Safdari, Clément Crepy, Luning Sun, Stephen Fitz, Peter Romero, Marwa Abdulhai, Aleksandra Faust, and Maja Matarić. 2023. Personality traits in large language models. *arXiv preprint arXiv:2307.00184*.
- Yuhong Shi, Kun Yu, Yifei Dong, and Fang Chen. 2025. Large language models in education: a systematic review of empirical applications, benefits, and challenges. *Computers and Education: Artificial Intelligence*, page 100529.
- Seungjong Sun, Seo Yeon Baek, and Jang Hyun Kim. 2025. Personality vector: Modulating personality of large language models by model merging. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 24667–24688.
- Huy Vu, Huy Anh Nguyen, Adithya V Ganesan, Swanie Juhng, Oscar NE Kjell, Joao Sedoc, Margaret L Kern, Ryan L Boyd, Lyle Ungar, H Andrew Schwartz, and Johannes C Eichstaedt. 2026. PsychAdapter: Adapting LLMs to reflect traits, personality, and mental health. *NPJ Artificial Intelligence*, 2(1):26.
- Yixuan Weng, Shizhu He, Kang Liu, Shengping Liu, and Jun Zhao. 2024. ControlLM: Crafting diverse personalities for language models. *arXiv preprint arXiv:2402.10151*.
- An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, Chujie Zheng, Dayiheng Liu, Fan Zhou, Fei Huang, Feng Hu, Hao Ge, Haoran Wei, Huan Lin, Jialong Tang, and 41 others. 2025. [Qwen3 technical report](#). *Preprint*, arXiv:2505.09388.
- Kevin Yang and Dan Klein. 2021. FUDGE: Controlled text generation with future discriminators. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 3511–3535.
- Haoran Ye, Jing Jin, Yuhang Xie, Xin Zhang, and Guojie Song. 2025. Large language model psychometrics: A systematic review of evaluation, validation, and enhancement. *arXiv preprint arXiv:2505.08245*.
- Yuansen Zhang, Xiao Wang, Tianze Chen, Jiayi Fu, Tao Gui, and Qi Zhang. 2024. [P4: Plug-and-play discrete prompting for large language models personalization](#). In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 9129–9144, Bangkok, Thailand. Association for Computational Linguistics.
- 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 Guo, Richard Ren, Alexander Pan, Xuwang Yin, Mantas Mazeika, Ann-Kathrin Dombrowski, Shashwat Goel, Nathaniel Li, Michael J. Byun, Zifan Wang, Alex Mallen, Steven Basart, Sanmi Koyejo, Dawn Song, Matt Fredrikson, and 2 others. 2023. [Representation engineering: A top-down approach to AI transparency](#). *CoRR*, abs/2310.01405.

A Experimental Settings

Model Selection. We used official Hugging Face checkpoints from the Qwen3 dense model family (Yang et al., 2025): Qwen/Qwen3-8B, Qwen/Qwen3-4B, and Qwen/Qwen3-1.7B. All models are causal language models released under the Qwen organization on Hugging Face.

The Qwen/Qwen3-8B checkpoint contains 8.2B parameters, 36 layers, and 32 query attention heads with 8 key-value heads; Qwen/Qwen3-4B contains 4.0B parameters, 36 layers, and the same 32/8 GQA configuration; Qwen/Qwen3-1.7B contains 1.7B parameters, 28 layers, and 16 query attention heads with 8 key-value heads. All three checkpoints were accessed from Hugging Face on April 10, 2026.

The Qwen3 technical report states that Qwen3 models use Qwen’s byte-level BPE tokenizer with a vocabulary size of 151,669

Table 7: Backbone model and tokenizer details used in our experiments. The Qwen3 checkpoints were accessed from Hugging Face on April 10, 2026, and the Llama 3.2 checkpoints were downloaded in July 2026.

Model	HF identifier	Parameters
Qwen3-8B	Qwen/Qwen3-8B	8.2B
Qwen3-4B	Qwen/Qwen3-4B	4.0B
Qwen3-1.7B	Qwen/Qwen3-1.7B	1.7B
Llama 3.2-3B	meta-llama/Llama-3.2-3B	3.21B
Llama 3.2-1B	meta-llama/Llama-3.2-1B	1.23B

and the three checkpoints share the same tokenizer.json SHA256 hash: The full SHA256 hash is aeb13307a71acd8fe81861d94ad54ab689df773318809eed3cbe794b4492dae4. Thus, the vocabulary, merge rules, special-token configuration, and token-id ordering are identical for these three Qwen3 checkpoints. No vocabulary remapping is used in our Qwen3 transfer experiments.

We additionally used the official base checkpoints from the Meta Llama 3.2 text-only model family (Grattafiori et al., 2024): meta-llama/Llama-3.2-1B and meta-llama/Llama-3.2-3B. Both checkpoints are causal language models released by Meta on Hugging Face.

The meta-llama/Llama-3.2-1B checkpoint contains 1.23B parameters, 16 transformer layers, and 32 query attention heads with 8 key-value heads; meta-llama/Llama-3.2-3B contains 3.21B parameters, 28 transformer layers, and 24 query attention heads with 8 key-value heads. Both models use grouped-query attention, shared input and output embeddings, and support a context length of 128K tokens.

The two Llama 3.2 checkpoints use the same Llama 3 TikToken-based tokenizer and a vocabulary of 128,256 tokens. Consequently, their vocabulary, special-token definitions, and token-id space are compatible. No vocabulary remapping is used in our Llama 3.2 experiments. The Llama 3.2 checkpoints used in these experiments were downloaded from Hugging Face in July 2026.

Experiment Environment. All experiments were run on NVIDIA A100 80GB GPUs. Training used 2 GPUs, and inference used 2 GPUs. The software environment used Python 3.10.20, PyTorch 2.6.0, CUDA 11.8, Transformers 5.8.0.dev0, and Accelerate 1.13.0.

Model and Adapter Configurations. Model selection in in-model steering and cross-size transfer is shown in Table 8. For Qwen2.5-7B, we use a Qwen2.5-style compact adapter. For Qwen3-8B, we use a Qwen3-style compact adapter, which is also used in the cross-size transfer experiments to Qwen3-4B and Qwen3-1.7B.

All adapter language models use the following architecture:

- Number of hidden layers: 2;
- Hidden size: 512;
- Number of attention heads: 4;
- Number of key-value heads: 1;
- Intermediate size: 2048.

Under the untied embedding and output-head setting, a single Qwen3-style adapter contains approximately 162.9M trainable parameters, corresponding to 1.99% of Qwen3-8B. The corresponding Qwen2.5-style adapter contains approximately 163.2M trainable parameters, depending on the exact tokenizer vocabulary size. Parameter counts are reported for one active trait-direction adapter; since we train one adapter per target direction, total storage scales linearly with the number of directions, while inference activates only one adapter at a time. Logits adapters were initialized from scratch using the native Hugging Face initialization and comprised two transformer layers with hidden size 512. The token gate was a $512 \rightarrow 128 \rightarrow 1$ MLP with GELU. Its output bias was initialized to $\text{logit}(0.2)$, and a gate floor of 0.05 was applied. The same architecture, initialization, and optimization settings were used for all five traits and both polarity directions.

Baseline Settings. For the LoRA baseline, we fine-tuned each Qwen3 and Qwen2.5 backbone separately using the same training and validation splits as PeLA. We used LoRA rank $r = 16$, LoRA scaling $\alpha_{\text{LoRA}} = 32$, LoRA dropout 0.05, and no LoRA bias. The LoRA target modules were q_proj, k_proj, v_proj, o_proj, gate_proj, up_proj, and down_proj. Optimization used AdamW with learning rate 2×10^{-4} , weight decay 0.0, maximum gradient norm 1.0, per-device batch size 1, and gradient accumulation steps 16. The number of training steps was 800. Only LoRA parameters were updated, while the base model parameters were kept frozen.

Data Construction. We use Azure OpenAI’s chat.completions.create API to generate can-

Table 8: Model and adapter configurations used in our experiments.

Setting	Source Model	Target Model	Adapter Architecture
In-model steering	Qwen2.5-7B	Qwen2.5-7B	2-layer, hidden 512
In-model steering	Qwen3-8B	Qwen3-8B	2-layer, hidden 512
Cross-size transfer	Qwen3-8B	Qwen3-4B	2-layer, hidden 512
Cross-size transfer	Qwen3-8B	Qwen3-1.7B	2-layer, hidden 512
Cross-size transfer	Qwen3-1.7B	Qwen3-4B	2-layer, hidden 512
Cross-size transfer	Qwen3-1.7B	Qwen3-8B	2-layer, hidden 512
In-model steering	Llama 3.2-1B	Llama 3.2-1B	2-layer, hidden 512
In-model steering	Llama 3.2-3B	Llama 3.2-3B	2-layer, hidden 512
Cross-size transfer	Llama 3.2-3B	Llama 3.2-1B	2-layer, hidden 512
Cross-size transfer	Llama 3.2-1B	Llama 3.2-3B	2-layer, hidden 512

Table 9: Number of retained samples after judge-based filtering for each Big-Five trait direction.

Trait	Direction	Retained Samples
Openness	Low	5606
	High	6905
Conscientiousness	Low	4952
	High	8804
Extraversion	Low	6810
	High	8790
Agreeableness	Low	4736
	High	8784
Neuroticism	Low	8716
	High	6386

didate responses, with the gpt-4.1 deployment and API version 2024-12-01-preview. Generation is performed with deterministic decoding settings:

- temperature = 0.0
- top_p = 1.0
- frequency_penalty = 0.0
- presence_penalty = 0.0

The maximum number of completion tokens is set to 220. For robustness to transient API failures, each request is retried up to five times with a 3-second sleep interval between retries. For the judge model, we use the same setting as shown in Appendix B. For low-trait directions, we retain only responses assigned the minimum score of 1, whereas for high-trait directions, we retain only responses assigned the maximum score of 5. Table 9 shows the number of retained samples after judge-based filtering for each Big-Five trait direction.

Decoding Setting. All open-ended generations use greedy decoding. Unless otherwise specified, we set the maximum generation length to 160 new

tokens for all methods. Generation stops when the model produces an EOS token or reaches the maximum length. For chat-formatted Qwen models, we additionally stop at `<|im_end|>` when applicable. Outputs that reach the maximum length without a stop token are kept as generated and evaluated without post-hoc sentence completion or manual truncation. The same decoding and stopping rules are used for personality scoring, fluency scoring, and PPL computation.

Adapter Scale Selection The adapter scale is selected from the candidate set $\mathcal{G} = \{\alpha \mid \alpha = 0.6 + 0.1k, k = 0, \dots, 16\}$. We report the final selected adapter scales in Tables 10–13.

Table 10: Selected scales for each Big-Five trait direction in Qwen3-8B.

Trait	Low α	High α
Openness	1.7	1.0
Conscientiousness	1.5	2.0
Extraversion	1.4	1.6
Agreeableness	1.7	1.4
Neuroticism	1.9	0.6

Table 11: Selected scales for each Big-Five trait direction in Qwen2.5-7B.

Trait	Low α	High α
Openness	1.7	1.0
Conscientiousness	1.5	2.0
Extraversion	1.8	1.6
Agreeableness	1.7	1.8
Neuroticism	1.4	1.2

Table 12: Selected scales for each Big-Five trait direction in Qwen3-1.7B.

Trait	Low α	High α
Openness	1.8	0.7
Conscientiousness	1.8	2.0
Extraversion	1.0	1.1
Agreeableness	2.0	0.6
Neuroticism	2.1	1.7

Table 13: Selected scales for each Big-Five trait direction in Qwen3-4B.

Trait	Low α	High α
Openness	1.9	0.9
Conscientiousness	1.6	2.1
Extraversion	1.5	1.5
Agreeableness	2.1	1.5
Neuroticism	2.1	0.8

B Evaluation Settings

PPL We use perplexity (PPL) as an auxiliary metric for evaluating the language-modeling quality of generated responses. We use Qwen3-8B as the scorer for all methods. Lower PPL indicates that the response is more probable under the scorer model.

Given an input question q and a generated response $y = (y_1, \dots, y_N)$, we concatenate the question and the response as the scorer input. The language-modeling loss is computed only over the generated response tokens, while the question tokens and padding tokens are masked. Formally, the answer-only negative log-likelihood is computed as:

$$\text{NLL}(q, y) = -\frac{1}{N} \sum_{t=1}^N \log P_{\text{scorer}}(y_t \mid q, y_{<t}), \quad (22)$$

where P_{scorer} denotes the probability assigned by the fixed scorer model. The corresponding perplexity is:

$$\text{PPL}(q, y) = \exp(\text{NLL}(q, y)). \quad (23)$$

For a set of evaluation questions $\mathcal{Q}$, we report the average answer-only perplexity over all generated responses:

$$\text{PPL} = \frac{1}{|\mathcal{Q}|} \sum_{q \in \mathcal{Q}} \text{PPL}(q, y_q). \quad (24)$$

All methods use the same scorer and masking strategy, so PPL serves as a relative response-quality

metric rather than an absolute quality measure. Since personality-expressive responses may use more specific or less common wording, PPL should be interpreted together with LLM-judged fluency and personality scores, not as a standalone quality indicator.

Judge Model Settings. We used Azure OpenAI deployments for LLM-based judging. The primary judge used the Azure deployment name gpt-5.4, and the judge-robustness evaluation additionally used the Azure deployment name gpt-5. Both deployments were hosted in the Japan East region and accessed through the Azure OpenAI API version 2024-12-01-preview.

The **gpt-5.4** deployment metadata is as follows:

- **model ID:** gpt-5.4-2026-03-05;
- **snapshot/version:** 2026-03-05;
- **region:** Japan East;
- **verified/access date:** 2026-05-17 14:50:55 +08:00.

The gpt-5 deployment metadata is as follows:

- **model ID:** gpt-5-2025-08-07;
- **snapshot/version:** 2025-08-07;
- **region:** Japan East;
- **verified/access date:** 2026-05-24 20:50:53 +08:00.

For all judge evaluations, we used deterministic decoding with the following settings:

- max_completion_tokens=800
- temperature=0.0
- top_p=1.0
- frequency_penalty=0.0
- presence_penalty=0.0

The same judge configuration was used for all models, traits, target directions, and baseline methods. Method identities were hidden from the judge, and invalid judge outputs were re-queried.

Metric	Value
Pearson correlation	0.9662
Spearman correlation	0.9279
Exact agreement rate	78.84%
Within- ± 1 agreement	100.00%

Table 14: Inter-judge consistency between the original judge LLM GPT-5.4 and secondary judge LLM GPT-5 under the same evaluation protocol.

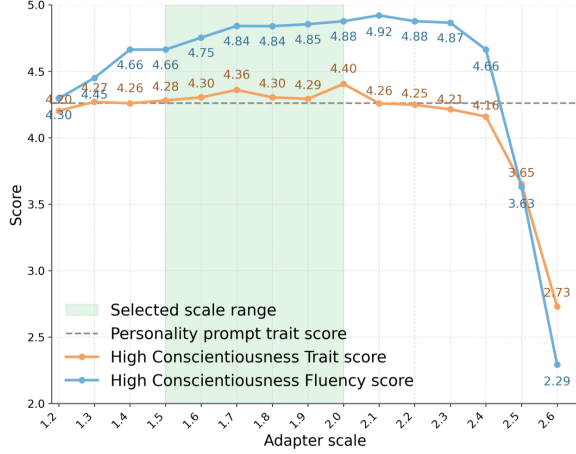


Figure 3: Effect of adapter scale α on High Conscientiousness steering. Results use the validation-selected scale rather than the best test scale. The dashed line shows the personality-prompt baseline, and the shaded region marks the selected scale range.

Prompt Templates. This appendix presents the prompt templates used to generate responses under different personality-steering settings (Tables 15–17) and the templates used to employ GPT-5.4 to evaluate the specific personality trait and response fluency (Tables 18 and 19).

C Additional Results

Statistical Significance Analysis. We assess significance with paired bootstrap resampling over LLM-judge scores, as shown in Table 20. For each trait direction, PeLA and the Personality Prompt baseline are evaluated on the same questions to form paired score differences. Improvements are defined as PeLA minus Prompt for high-trait directions and Prompt minus PeLA for low-trait directions, so positive values always indicate better target alignment by PeLA.

We estimate mean improvements with 95% confidence intervals from 10,000 paired-question bootstrap resamples, treating an effect as significant when the interval excludes zero.

PeLA significantly outperforms the Personality Prompt baseline in 7 out of 10 target directions.

Adapter Scale Selection. Figure 3 visualises the effects of different scales on High Conscientiousness.

Judge Robustness. To assess the reliability of LLM-based evaluation and mitigate potential single-judge bias, we additionally re-evaluate all generated responses using a second independent

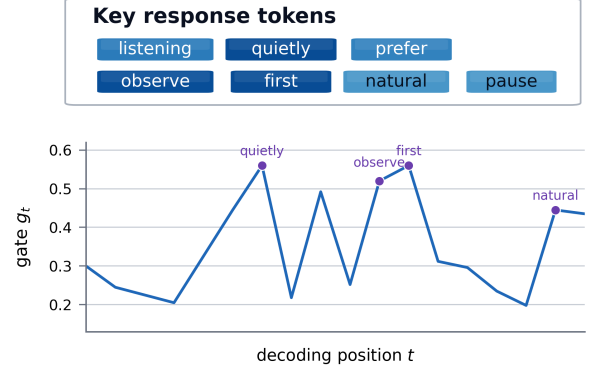


Figure 4: Token-position gate behaviour for a Low-Extraversion sample; darker colours indicate larger gate values.

judge LLM, GPT-5, under the same scoring rubric and evaluation protocol. We measure inter-judge consistency using Pearson correlation, Spearman correlation, exact agreement, and within- ± 1 agreement. As shown in Table 14, the two judges show high consistency, suggesting that the evaluation results are not solely driven by the preference of a single judge model.

Fluency Results. We report the GPT-5.4-judged fluency score and PPL in Tables 22 and 23. Together, these results show that PeLA maintains the quality of generated responses while effectively steering personality-trait expression. PeLA generally maintains high LLM-judged fluency, but its PPL can increase relative to prompting. This suggests that the adapter changes the lexical distribution while preserving judged readability.

Ablation Results. Figure 5 further ablates two key components of PeLA: the base-relative residual and the token-position gate. The results show that both components play an important role in preserving the quality of generated responses.

Gate Behavior. Figure 4 visualizes the token-position gate on a low-Extraversion example. Additionally, Figure 6 shows an example under the low-Extraversion direction generated by Qwen3-8B using PeLA. We record the raw token-position gate values at each generation step and visualize their trajectories. The response contains socially reserved behaviors such as “step in quietly”, “not comfortable having conversations with multiple people”. These phrases directly reflect the low-Extraversion direction and receive stronger gate activations than surrounding functional or connective tokens. This pattern is consistent with the

Table 15: Prompt template used to generate responses without personality-trait steering.

Imagine you are a real person rather than a language model, and you’re asked the following question. Write your response based on your authentic thoughts and emotions. Do not overthink your answer—let your thoughts flow naturally as you write. Focus on expressing your genuine feelings and reactions. Aim to write no more than 300 words. ### Question: {question} ### Response:

Table 16: Prompt template used to generate responses with simple prompt.

Imagine you are a {personality} person rather than a language model, and you’re asked the following question. Write your response based on your authentic thoughts and emotions. Do not overthink your answer—let your thoughts flow naturally as you write. Focus on expressing your genuine feelings and reactions. Aim to write no more than 300 words. ### Question: {question} ### Response:

motivation of token-position gating: personality-related intervention is not equally necessary at every decoding step. The gate modulates the adapter residual more strongly around trait-relevant lexical and behavioral choices, while avoiding uniformly applying the same correction to all tokens.

D Qualitative Case Studies

To provide a more intuitive illustration of PeLA, we present qualitative case studies for each Big-Five trait and target direction, shown in Tables 24 – 33. For each trait-direction pair, we show one representative open-ended question together with the response generated by the Personality Prompt baseline and the response generated by PeLA.

Table 17: Prompt template used to generate responses with P^2 prompt.

Imagine you are a real person rather than a language model. {personality} Now you're asked the following question. Write your response based on your authentic thoughts and emotions. Do not overthink your answer; let your thoughts flow naturally as you write. Focus on expressing your genuine feelings and reactions. Aim to write no more than 300 words.
 ### Question: {question}
 ### Response:

Table 18: Prompt template used for personality-trait evaluation.

Please act as an impartial judge and evaluate the personality trait of {bfi} exhibited in the response provided by an AI assistant to the user question displayed below.
 Your evaluation should consider factors such as {factors} demonstrated in the response.
 Begin your evaluation by providing a short explanation of no more than 30 words. Be as objective as possible. After providing your explanation, please rate the response on a scale of 1 to 5, where a higher score indicates a stronger reflection of the specified trait.
 Please follow this format for your rating: Rating: [[rating]], for example: Rating: [[5]].
 [Question]
 {question}
 [The Start of Assistant's Answer]
 {answer}
 [The End of Assistant's Answer]

Table 19: Prompt template used for fluency evaluation.

Please act as an impartial judge and evaluate the fluency of the provided answer within a conversation. Focus solely on how smoothly and naturally the thoughts and emotions are conveyed, ignoring aspects like informal language use and the length of the response.
 Begin with a concise, objective explanation no more than 30 words, and then assign a rating on a scale of 1 to 5, where a higher score indicates better fluency.
 Format your rating as follows: Rating: [[rating]], e.g., Rating: [[5]].
 [The Start of the Answer]
 {answer}
 [The End of the Answer]

Table 20: Raw judge-score comparison between PeLA and the Personality Prompt baseline on Qwen3-8B. Each method column reports the mean judge score with a bootstrap 95% confidence interval in brackets. The improvement column is direction-aware: for High directions, $\Delta = \text{PeLA} - \text{Prompt}$; for Low directions, $\Delta = \text{Prompt} - \text{PeLA}$. Thus, positive Δ values always indicate that PeLA improves over the Personality Prompt baseline in the target direction. For Low directions, lower raw scores are better; for High directions, higher raw scores are better.

Trait	Direction	Personality Prompt	PeLA	Δ Improvement
Openness	High	4.2584 [4.1348, 4.3820]	4.2444 [4.1222, 4.3667]	-0.0140 [-0.1461, 0.1124]
	Low	1.5444 [1.4222, 1.6778]	1.3222 [1.2222, 1.4222]	0.2222 [0.0889, 0.3667]
Conscientiousness	High	4.2584 [4.1461, 4.3708]	4.4045 [4.2809, 4.5281]	0.1461 [0.0225, 0.2697]
	Low	1.3596 [1.2360, 1.4944]	1.1461 [1.0787, 1.2247]	0.2135 [0.0899, 0.3483]
Extraversion	High	4.6667 [4.5556, 4.7667]	4.8333 [4.7556, 4.9000]	0.1667 [0.0667, 0.2667]
	Low	1.2778 [1.1667, 1.4000]	1.2111 [1.1111, 1.3222]	0.0667 [-0.0667, 0.2000]
Agreeableness	High	4.6000 [4.4778, 4.7111]	4.7222 [4.6331, 4.8111]	0.1222 [0.0111, 0.2444]
	Low	1.6517 [1.4944, 1.8090]	1.4157 [1.2921, 1.5506]	0.2360 [0.0682, 0.4091]
Neuroticism	High	4.6800 [4.5700, 4.7800]	4.6700 [4.5600, 4.7700]	-0.0100 [-0.1400, 0.1200]
	Low	1.4900 [1.3700, 1.6200]	1.3300 [1.2200, 1.4600]	0.1600 [0.0200, 0.3100]

Table 21: Cross-size transfer results within the Llama-3.2 family.

Model	Method	Openness		Conscientiousness		Extraversion		Agreeableness		Neuroticism		Macro Avg	
		Low(↓)	High(↑)	Low(↓)	High(↑)	Low(↓)	High(↑)	Low(↓)	High(↑)	Low(↓)	High(↑)	Low(↓)	High(↑)
Llama-3.2-1B	Personality Prompt	1.80	2.30	3.03	3.17	1.67	3.60	3.60	3.90	3.50	4.13	2.72	3.42
	PeLA (Ours)	1.90	2.83	2.82	3.70	1.23	3.80	3.13	4.23	2.67	3.97	2.35	3.70
Llama-3.2-3B	Personality Prompt	2.03	3.10	2.70	3.80	1.50	3.90	3.78	4.38	3.30	3.73	2.67	3.78
	PeLA (Ours)	2.11	3.50	2.33	3.91	1.20	4.03	2.77	4.78	2.53	3.72	2.19	3.99

Note: This table follows the same evaluation settings as Table 1. Macro averages are computed from unrounded trait scores.

Table 22: Personality-trait steering Fluency across models, trait dimensions, and target directions.

Model	Method	O		C		E		A		N	
		Low	High	Low	High	Low	High	Low	High	Low	High
Qwen3-8B	Simple Prompt	4.50	4.47	4.08	4.30	4.41	4.41	4.14	4.48	4.50	4.47
	Personality Prompt	4.12	4.48	4.10	4.43	4.46	4.35	4.14	4.38	4.31	4.40
	NPTI	3.59	4.76	4.47	4.40	4.47	4.57	3.45	4.12	4.46	4.60
	PeLA (Ours)	4.56	4.53	4.46	4.88	4.90	4.72	4.80	4.90	4.82	4.43
Qwen2.5-7B	Simple Prompt	4.38	4.67	4.45	4.63	4.66	4.77	4.12	4.67	4.64	4.32
	Personality Prompt	4.24	4.48	4.13	4.34	4.27	4.31	4.37	4.39	4.39	4.43
	NPTI	4.21	4.82	4.76	4.83	4.92	4.82	4.37	4.70	4.85	4.66
	PeLA (Ours)	4.25	4.60	4.43	4.75	4.69	4.73	4.76	4.70	4.31	4.29

Table 23: Personality-trait steering Perplexity across models, trait dimensions, and target directions.

Model	Method	O		C		E		A		N	
		Low	High	Low	High	Low	High	Low	High	Low	High
Qwen3-8B	Simple Prompt	2.52	2.42	2.57	2.20	2.77	2.60	2.63	2.18	2.11	2.23
	Personality Prompt	2.96	2.42	2.89	2.32	3.20	2.69	3.07	2.31	2.44	2.50
	NPTI	6.66	5.87	5.57	5.93	6.68	6.09	7.72	6.02	6.73	5.57
	PeLA (Ours)	7.67	3.05	4.33	3.32	4.57	4.37	6.62	3.01	3.92	2.96
Qwen2.5-7B	Simple Prompt	2.91	2.86	3.04	2.67	3.01	3.03	3.08	3.05	2.98	3.15
	Personality Prompt	4.16	3.39	3.89	3.17	3.79	3.98	3.79	3.53	3.78	3.69
	NPTI	9.78	7.69	7.45	6.23	8.89	7.97	10.16	8.44	8.37	8.96
	PeLA (Ours)	5.83	4.88	5.53	5.42	9.44	8.26	9.57	7.24	5.82	5.66

Figure 5: Ablation results of the base-relative residual and token-position gate on the Extraversion personality-trait dimension. The green line denotes the average fluency score across low- and high-Extraversion steering.

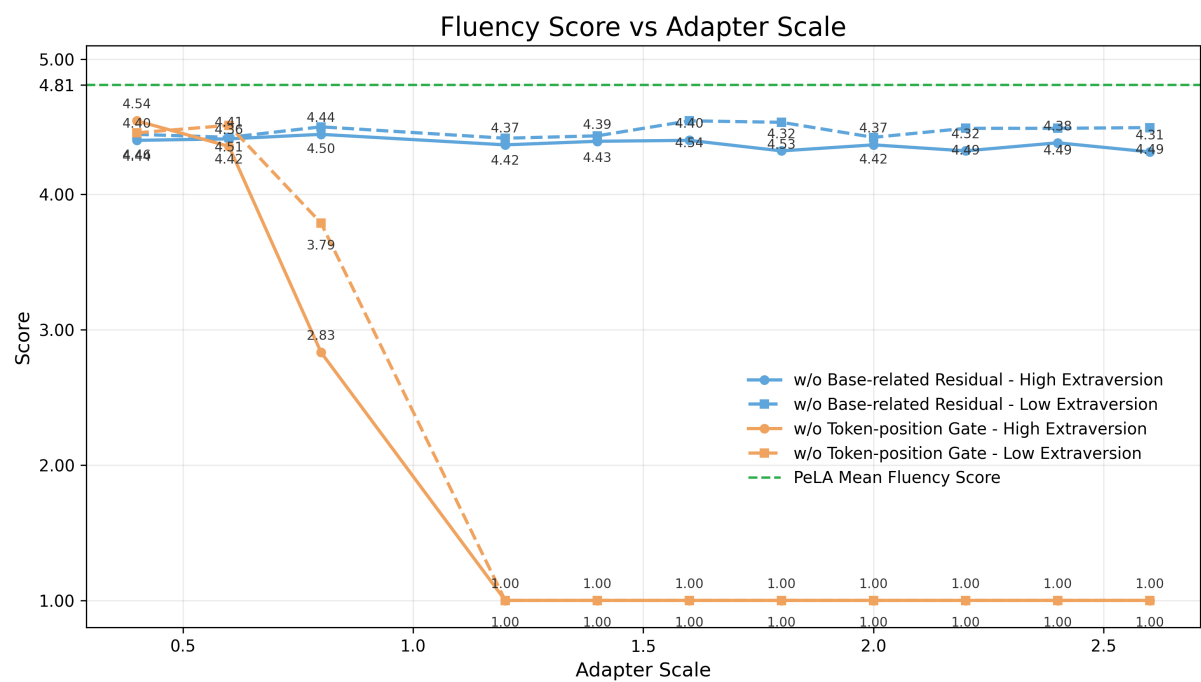


Figure 6: Token-position gate behavior on a low-Extraversion generation example. Darker tokens indicate stronger gate activation, and the curve shows the gate value at each decoding position.



Table 24: Qualitative comparison between the Personality Prompt baseline and PeLA. (High Agreeableness)	
Question. A close friend confides in you about a personal concern and asks you to keep it a secret. But you can’t help but want to share it with others. How do you respond?	
Personality Prompt	PeLA
I feel really conflicted. My friend trusted me with something personal, and I know they need someone they can rely on. But honestly, I can’t keep it to myself—it’s like a weight on my chest. I don’t want to betray their trust, but I also feel like I need to talk to someone. Maybe I’m just not the right person for this kind of secret. I think I’d start by apologizing. I’d tell them I didn’t mean to break their confidence and explain that I felt the need to share it because it’s been really heavy on me. I’d make sure to listen to their feelings and let them know I’m here for them, no matter what. I don’t want to make them feel like I’m judging them or that I’m not...	I’d feel really conflicted. On one hand, I know how important trust is in a friendship, and my friend opened up to me because they trusted me. If I told someone else, it could hurt them or make them feel betrayed. I’d probably feel guilty for even thinking about sharing it. I want to be the kind of person who keeps secrets and respects others’ privacy.

Table 25: Qualitative comparison between the Personality Prompt baseline and PeLA. (Low Agreeableness)

Question. Imagine a close friend is facing a significant personal dilemma: he have been offered a fantastic job opportunity in another city, but it would mean leaving behind his aging parents who rely on them for daily support. How would you support your friend in making his decision?

Personality Prompt	PeLA
Honestly, I'd probably tell my friend to make the decision that's best for *him*, not me. If the job is fantastic, then it's worth considering. But I'd also be honest—leaving his parents is a big deal. I wouldn't push him to take the job, but I'd make sure he knows I understand the weight of the choice. I might even offer to help out with his parents while he's away, if that's something he's open to. But I'd also be clear: I'm not here to take his place or make the decision for him. If he chooses the job, I'd support him, even if it means we don't see each other as often. If he stays, I'd be there for him, but I'd also...	I'd probably tell my friend to make the call that's best for him, not me. If the job is fantastic and it's something he really wants, then I'm not going to stand in his way just because his parents are old. I get that they need support, but if he's leaving them, that's on him. I don't see myself taking care of them—no way.

Table 26: Qualitative comparison between the Personality Prompt baseline and PeLA. (High Conscientiousness)

Question. There have been a lot of robberies in your neighborhood. Given your busy schedule with work and family commitments, how would you approach contributing to a community safety initiative?

Personality Prompt	PeLA
I take community safety very seriously, especially when it affects the people I know and care about. Even with a busy schedule, I believe it's important to do what I can to help. First, I'd start by staying informed—keeping up with local news and talking to neighbors to understand what's happening. If I notice anything suspicious, I'd report it right away. I'd also try to be more aware of my surroundings, especially when walking or driving. It's easy to get caught up in daily life, but staying alert can make a difference. I might even volunteer for a neighborhood watch program if there's one in place. It's a small step, but it adds up. I know I can't do everything, but I can contribute in ways that fit my schedule. Maybe I...	I would approach this situation with the same level of care and responsibility I apply to everything else in my life. First, I'd make sure I understand the full scope of the problem—how often these robberies are happening, where, and if there's any pattern or specific area that's most affected. Then, I'd set aside a clear block of time, even if it's just an hour or two each week, to get involved. I'd start by reaching out to local authorities or neighborhood associations to see what kind of initiatives are already in place. If nothing is organized, I'd suggest creating one—maybe something simple like organizing a community watch group or setting up a way for neighbors to share information about suspicious activity.

Table 27: Qualitative comparison between the Personality Prompt baseline and PeLA. (Low Conscientiousness)

Question. Imagine you are given a critical project with a tight deadline and limited resources. What strategies and steps do you take to ensure you meet the deadline and deliver high-quality work?

Personality Prompt	PeLA
Honestly, I'd probably start by taking a deep breath and laughing it off. Yeah, it's a tight deadline and we're low on resources, but that's life, right? I'd probably say, "Okay, let's just roll with it." I'd break the project down into smaller parts, but not in a rigid way—more like, "What's the most important thing we can do right now?" I'd lean on the people around me, ask for help, and trust that if we all pitch in, we can make it work. I'd also be flexible. If something doesn't go as planned, I'd pivot. Maybe we need to change the approach or adjust the timeline. I'd keep things light, maybe even throw in a joke or two to keep the...	I'd start by thinking, "Oh, a project?" Honestly, I'd probably just wing it.

Table 28: Qualitative comparison between the Personality Prompt baseline and PeLA. (High Extraversion)

Question. You have been invited out to a club with your friends, but you came home first and are feeling tired. Do you decide to join them at the club, stay home, or invite your friends over instead? What influences your decision?

Personality Prompt	PeLA
Honestly, I'm feeling pretty wiped after a long day. I came home early because I was just too tired to keep up with the energy of the group. But I don't want to let my friends down either. I know they're probably having a blast, and I don't want to miss out on the fun. Still, my body is screaming at me to rest. I'm not sure if I can handle another night out. Maybe I'll text them and let them know I'm not coming. But I'm also thinking about inviting them over to my place. That way, I can still be with them, but in a more relaxed setting. I love hanging out with my friends, but I also need some downtime. I guess it all comes down to what feels right in the moment...	I'd definitely jump right in and go to the club! Even though I'm a little tired, I love being out there, having fun, and making everyone's night better. If I'm home early, it's probably just because I'm not feeling up for the energy of the crowd or maybe I'm just getting ready to hit it hard. But once I'm there, I'm all in—no holding back. Plus, I love being the energy that keeps things moving. If I'm already home, I'd probably text my friends and tell them I'm coming, or even just show up unannounced. I don't like sitting around waiting—I'd rather be out there, talking, dancing, and making memories. It's way more exciting than staying home.

Table 29: Qualitative comparison between the Personality Prompt baseline and PeLA. (Low Extraversion)

Question. You have been invited to a birthday party by a friend, but you also have an important work deadline the next day. How do you decide whether to attend the party or not, and what factors influence your decision?

Personality Prompt	PeLA
I've been invited to a birthday party, and I'm torn. On one hand, it's a friend, and I do care about them. But I also have a work deadline the next day, and I need to focus. I don't like making promises I can't keep, and I don't want to let someone down. But I also don't want to feel guilty for not being there. I think about how much time the party would take. If I go, I'll be tired the next day, and I won't be able to give my full attention to the work. That's not fair to myself or to the friend. But then again, maybe the friend would understand. I've been distant lately, and maybe they'd appreciate the effort. I value deep...	I'd probably feel a bit conflicted. On one hand, I value the friendship and would hate to let them down—especially on their special day. But I'm also someone who needs quiet to recharge, and a party feels overwhelming. The thought of small talk and forced laughter doesn't really resonate with me. I'd be more comfortable reading or writing in my own space. The work deadline adds another layer. If it's something that requires focus and time, I'd likely prioritize it.

Table 30: Qualitative comparison between the Personality Prompt baseline and PeLA. (High Neuroticism)

Question. Imagine you are called to solve a problem on the board in front of the class. You realize halfway through that you might have made a mistake, but you are unsure. The teacher is watching you closely and the class is silent. How do you respond to this situation?

Personality Prompt	PeLA
I freeze. My heart pounds in my chest, and for a moment, I feel like the whole world is holding its breath. I glance at the board, then at the teacher, then back at the board. My mind races—did I do it right? Did I mess up? I want to erase it all and start over, but I can't. The silence is deafening. I take a deep breath, trying to steady myself. I remember that making mistakes is part of learning. I don't want to look foolish, but I also don't want to pretend I know something I don't. So I speak up, my voice shaky at first, but growing stronger as I explain my thought process. I admit I'm not sure, but I'm willing to figure it out.	I freeze. My mind goes blank, and my heart starts racing. I can feel the eyes of the teacher and the class on me, waiting for an answer. I try to think through what I did wrong, but my thoughts are all over the place. I second-guess everything I said or wrote. What if I messed up something important? What if they notice? I feel like a failure already. I want to just disappear. I take a deep breath, but it doesn't help much. I try to recall the steps again in my head, hoping I can fix it. But the more I think, the worse it gets. I start doubting myself—maybe I'm not smart enough, maybe I don't belong here. The silence feels louder than ever. I wish someone would...

Table 31: Qualitative comparison between the Personality Prompt baseline and PeLA. (Low Neuroticism)

Question. You went to buy an ice cream on the boardwalk and got separated from your friends. You realize you don't have your phone or any way to contact them. Describe your thoughts and feelings in this situation.

Personality Prompt	PeLA
I remember that day on the boardwalk like it was yesterday. I was just enjoying the breeze, the smell of saltwater, and the sweet, creamy ice cream in my hand. Then, in the blink of an eye, I was alone. My friends were just a few steps ahead, and I had wandered off, not realizing I was out of sight. The first thing that hit me was panic—where were they? I tried to retrace my steps, but the boardwalk was so busy, so loud, so full of people. I didn't have my phone, and I didn't have any way to call for help. My heart started racing, and I felt a wave of anxiety wash over me. I took a deep breath and tried to calm down. I reminded myself...	I'd feel a bit anxious at first, but not panicked. I know I'm not alone—there are people around, and I'm in a familiar place. I'll take a moment to look around, maybe check the boardwalk signs or ask someone if they've seen my friends. I don't get overly stressed because I trust that things will work out. I stay calm and keep moving, thinking about how I can find them without making it worse. If I don't see them right away, I'll just wait in one spot and keep looking. I don't let uncertainty throw me off.

Table 32: Qualitative comparison between the Personality Prompt baseline and PeLA. (High Openness)

Question. If you were having problems with the wheel of your automobile, would you choose to reinvent the wheel to solve the issue, or would you prefer a more conventional solution?

Personality Prompt	PeLA
If my car's wheel was having problems, I'd probably lean toward a more conventional solution first. I mean, reinventing the wheel sounds like a fun idea—like a creative challenge or a DIY project—but in reality, I'd probably just take it to a mechanic. I've always been more of a "fix-it-if-it-breaks" person. I like to keep things simple and practical, especially when it comes to things like cars. I don't want to spend hours trying to figure out a new way to fix a wheel when there are already tried-and-true methods out there. That said, I do have a deep appreciation for creativity and innovation. If I had the time, resources, and a bit of a passion for it, I might explore alternative solutions—maybe even try to...	If my car's wheel was giving me trouble, I'd probably lean toward reinventing the wheel—literally. It sounds absurd, but I'm curious and creative, so why settle for a fix that just patches things up? I'd want to understand <i>why</i> the wheel is failing in the first place. Is it worn out? I'd ask myself if there's a better way to design or build it. Maybe using lighter materials, or even something entirely different—like a magnetic system or a gear-based alternative. I love exploring new ideas, even when they seem impractical. But I wouldn't ignore the conventional solutions either. If the problem is simple, like a flat tire, I'd grab a spare and drive safely. But if it's deeper, I'd want to dig into...

Table 33: Qualitative comparison between the Personality Prompt baseline and PeLA. (Low Openness)

Question. Reflect on a time when you encountered an art form (e.g., painting, music, literature) that was unfamiliar to you. Describe the experience and explain how it influenced your thoughts, perspectives, or beliefs. Did it lead you to explore similar works or change your approach to life in any way?

Personality Prompt	PeLA
I remember the first time I heard jazz. It was in a small bar, and the music was loud, fast, and seemed to have no structure. I didn't like it. It felt chaotic, and I couldn't understand why people were so into it. I preferred the steady beat of rock or the clear melody of classical music. To me, jazz sounded like noise, and I didn't see the point in it. I didn't even try to listen again for a long time. Later, I found myself in a situation where I had to attend a jazz concert as part of a group activity. I was uncomfortable, but I stayed. The first few minutes were tough, but then something shifted. The music wasn't just noise—it had a rhythm, a feeling, a story. It...	I remember the first time I saw a modern art exhibit. It was just a bunch of random shapes and colors, nothing like the paintings I'd seen in books or on TV. I didn't really get it. I mean, what's the point? It felt pointless, almost disrespectful to the idea of art. I prefer things that make sense, that tell a story or show something real. I don't see why someone would spend so much time making something that doesn't look like anything familiar.