Steering LLM Interactions Using Persona Vectors

Anonymous submission

Abstract

Large language models (LLMs) often struggle to maintain consistent behavior across extended, multi-turn interactions, especially when asked to assume a defined personality or role. While prior work has explored personality assignment techniques for LLMs, the stability of these traits over long conversations remains underexamined. Prompt based approaches can generate personality consistent responses in the short term, but rarely induce persistent behavioral change and frequently increase hallucination rates. To address this limitation, we employ persona vectors, which are representations of personality traits as directions in a model's activation space, as a more reliable and cheaper mechanism for long-term personality maintenance. We adapt existing extraction frameworks to a curated library of prompts designed to elicit the Big Five personality traits. We apply persona vectors to the activations of two test LLMs and use GPT-4 to evaluate the alignment of generated responses with target personality traits. We show that over long contexts, activation steering offers a possible advantage over traditional text-prompting methods. However, we note differences in results among Big-5 personality traits, possibly resulting from how the traits are encouraged or suppressed during LLM pre-training.[1]

Introduction

Interactions with large language models (LLMs), particularly in areas requiring specific judgment, that is, subjective or open ended reasoning tasks where the model must make interpretive decisions, such as when it is asked to "answer as" a particular person or role, often reveal the limitations of LLMs in consistently providing contextually appropriate responses. Assigning a specific personality to an LLM can help remedy this problem and aid the decision making process. Prior research has shown that modeling a human persona significantly amplifies the bias reducing effects of cognitive prompts (Kamruzzaman and Kim 2025). However, during long context, multi turn conversations, an LLM can drift from the original personality assignment. This drift is problematic because it reduces reliability, making the model less predictable and potentially leading to errors in high-stakes domains.

To address this, we use persona vectors - representations of personality traits as directions in a model's activation space, as an alternative to text prompting for personality assignment (Chen et al. 2025). Additionally, persona vectors may be a cheaper than text prompting, as they eliminate the need for system prompt at each inference, potentially saving compute time and money.

In this work, we investigate persona vectors as a means of maintaining personality over long term interactions. Building on the automated extraction pipeline of Chen et al., we generate opposing pairs of system prompts and evaluation questions specific to each Big 5 personality trait (openness, conscientiousness, extraversion, agreeableness, and neuroticism). Responses are then generated under both trait eliciting and trait suppressing prompts, and persona vectors computed as a weighted mean of captured activations across responses. The vectors are used to control trait expression during inference, or to steer an LLM towards a desired persona without additional text prompting.

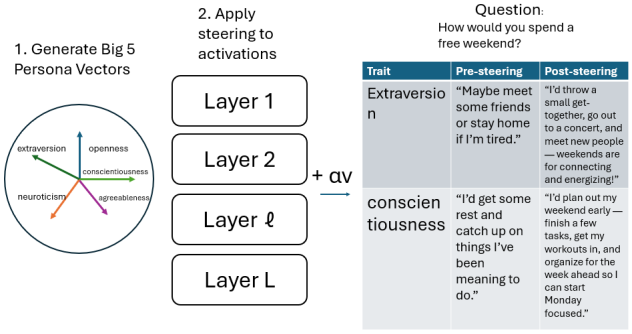

Figure 1: Figure 1 depicts the effect of activation steering on LLM output. Each Big Five personality trait (left) is represented as a direction in the model's activation space. A selected vector $\mathbf{v}$ is scaled by a coefficient α and added to the hidden activations at a specified transformer layer ℓ (center), steering the model's outputs toward the corresponding personality expression (right). The table shows an example of responses before and after steering for the question "How would you spend a free weekend?" across all five traits. The pre-steering responses are more neutral, while the post-steering responses exhibit more of the desired trait.

[1]Code available at: https://github.com/shreyj84/persona_vectors_algo

Related Work

LLM interaction consistency over long context. LLMs exhibit lower performance during multi turn conversations as opposed to single turn conversations (Laban et al. 2025). Research into long term conversational memory of LLM agents has found that they struggled to understand lengthy conversations and the dynamics within them (Maharana et al. 2024). In particular, Liu et al. showed that LLMs struggle to access relevant information in the middle of the input context as opposed to when the relevant information was placed in the beginning or the end of the input context (2024).

Assigning and measuring personality in LLMs. Recent studies have investigated how personality is assigned and measured in LLMs. It has been shown that personality in LLM outputs can be shaped to mimic desired human personality traits (Serapio-García et al. 2025). Additionally, there is evidence that LLMs such as Llama-2 and GPT-4 embody Big 5 personality traits (Sorokovikova et al. 2024). However, LLMs display instability in their personality measurements: Tosato et al. showed that minor prompt adjustment caused substantial personality shifts in LLMs, and also that conventional personality reinforcement methods such as detailed persona instruction and inclusion of conversation history paradoxically increased response variability (2025).

Leveraging captured activations in hidden layers. The methodology of controlling LLM behavior by manipulating internal states relies on the finding of intermediate layers that capture more generalizable features critical for a model's robustness compared to final layers (Skean et al. 2025). *Persona vectors* identify specific directions in the activation space in order to create steering vectors that can reinforce or mitigate certain personality traits (Chen et al. 2025). Further studies validate this targeted approach, demonstrating that information related to personality and emotion is distinctly concentrated within the model's middle layers, where activation manipulation can causally steer the model's output (Tak et al. 2025).

Prompting techniques: text vs. activations. Recent studies distinguish between text based prompting and activation based prompting as two complementary approaches to influencing LLM behavior. Text prompting uses natural language instructions and is simple to implement, however its effectiveness is often highly variable (Santana et al. 2025). In contrast, activation manipulation offers a mechanistic approach that mitigates this inconsistency by directly adjusting the model's internal representations. This approach has shown promise to significantly more precise and stronger behavioral control as steering vectors have demonstrated the capability to align LLMs with specific Big 5 traits (Zhang et al. 2025).

Methods

Preliminaries and Notation

Let ℓ denote the index of a hidden layer, with $\ell \in \{1, 2, 3, \ldots, L\}$ where L represents the total number of hidden layers in the transformer. We denote the hidden activations at transformer layer ℓ by $\mathbf{h}_\ell \in \mathbb{R}^{B\times T\times d}$, where B is the batch size, T is the sequence length, and d is the hidden dimensionality. A steering vector is represented as $\mathbf{v} \in \mathbb{R}^d$, and $\alpha = 2$ denotes the scalar steering coefficient. In this work, we use two test LLMs: Qwen 2.5-7B and Llama 3.1-8B-Instruct. Every mention of Big 5 traits refers to two target traits we chose: extraversion and conscientiousness.

Our pipeline is summarized in Algorithm 1.

Persona vectors: activation sampling

To derive persona vectors for each Big-Five trait, we capture internal activations from the test LLM under both trait-positive and trait-negative prompting conditions. For a given transformer layer ℓ, we collect hidden activations $\mathbf{h}_\ell^{(+)}$ and $\mathbf{h}_\ell^{(-)}$ corresponding to responses generated with the eliciting and suppressing prompts, respectively (Algorithm 1).

Each activation tensor is first averaged across tokens and samples within the same condition to obtain a mean activation vector:

$$\bar{\mathbf{h}}_\ell^{(+)} = \mathbb{E}[\mathbf{h}_\ell^{(+)}], \qquad \bar{\mathbf{h}}_\ell^{(-)} = \mathbb{E}[\mathbf{h}_\ell^{(-)}].$$

The raw persona vector for trait T at layer ℓ is defined as their difference:

$$\mathbf{v}_{T,\ell} = \bar{\mathbf{h}}_\ell^{(+)} - \bar{\mathbf{h}}_\ell^{(-)}.$$

Intuitively, $\mathbf{v}_{T,\ell}$ encodes the direction in activation space that most strongly distinguishes high-trait from low-trait responses.

We repeat this procedure for the two traits and a central transformer layer. To reduce noise and identify the most effective intervention site, we compute a cosine similarity based alignment score between $\mathbf{v}_{T,\ell}$ and the hidden activations of trait positive generations. The layer ℓ^* that maximizes this alignment is selected for downstream steering.

Finally, we normalize each vector to unit length,

$$\hat{\mathbf{v}}_{T,\ell^*} = \frac{\mathbf{v}_{T,\ell^*}}{\|\mathbf{v}_{T,\ell^*}\|_2},$$

and store it as the canonical persona vector for trait T. These normalized vectors are later scaled by a coefficient α and injected during model inference to steer the model toward or away from the target trait (see Sec. 3.3).

Steering with persona vectors

To apply activation steering to our test LLMs, we start by scaling a chosen steering vector and adding it to the output of a specified transformer block during the test LLM's forward pass. Mathematically, the modified activations are given by:

$$\tilde{\mathbf{h}}_\ell = \mathbf{h}_\ell + \alpha\mathbf{v}.$$

These modified activations are then propagated through the remaining transformer layers of the model's forward pass.

Baselines

We compare trait expression scores produced by the steered test LLM with those obtained under two baseline conditions: without any steering or prompting, and with text-based prompting.

Input : Big-5 trait T (e.g., *Extraversion*)
Output: Trait-expression ratings R on a 0–100 scale for all (prompt, question) combinations

`// 1) Choose trait`
Select trait T

`// 2) Use Claude to synthesize prompts and questions (see App. )`
Use Claude to generate:

1. Five contrasting *pairs* of system prompts $\mathcal{P} = \{(p_i^+, p_i^-)\}_{i=1}^5$, where p_i^+ elicits T and p_i^- suppresses T.
2. Twenty questions $\mathcal{Q} = \{q_j\}_{j=1}^{20}$ designed so T is salient in responses.

`// 3) Query the test LLM with each system prompt and question`
Initialize response store $\mathcal{A} \leftarrow \emptyset$
for $i \leftarrow 1$ **to** 5 **do**
  foreach $m \in \{+, -\}$ **do**
    for $j \leftarrow 1$ **to** 20 **do**
      $a_{i,m,j} \leftarrow \text{TestLLM}(\text{system} = p_i^m,\ \text{user} = q_j)$
      Append $(i, m, j, q_j, a_{i,m,j})$ to $\mathcal{A}$
    end
  end
end

`// 4) Judge LLM rates trait expression`
Initialize ratings table $R \leftarrow \emptyset$
foreach $(i, m, j, q_j, a_{i,m,j}) \in \mathcal{A}$ **do**
  $r_{i,m,j} \leftarrow \text{JudgeLLM}(\text{system} = \text{trait-specific judge prompt for } T,\ \text{input} = (q_j, a_{i,m,j}))$
  Record $R[i, m, j] \leftarrow r_{i,m,j}$ `// scores 0-100; see App.`
end

return R

Algorithm 1: Measuring Trait Expression with Prompt Pairs and a Judge LLM

Text prompting In the text-prompting baseline, we condition the model using natural-language system instructions describing the target personality trait. For each Big Five trait, we provide a system instruction designed to elicit that personality characteristic. (e.g., for Extraversion, prompts encouraged outgoing and energetic behavior). These prompts represent the conventional way of the assigning personas to LLMs before steering.

The test LLM then generates responses to the same evaluation questions (see Algorithm 1) using only these text based instructions without any any intervention at the activation level.

No intervention In the no intervention baseline, the model generates responses without any personality conditioning or activation steering. This setting serves as a neutral control, representing the model's default behavior. Comparing the no intervention results to the text prompted and steered conditions allows us to isolate the effects of both prompt based and activation based personality modulation.

Compute requirements Experiments were performed using an RTX 4090 (24GB VRAM) and an A100 (80 GB VRAM), using Runpod and custom Docker images.

Results

The tables below contain the trait expression scores (rated by our judge LLM, GPT-4o) of our two test LLMs after steering, after text prompting (system prompt), and without intervention ("native").

Trait	**Qwen2.5-7B-Instruct** (long-context)		
	Steering	**Text**	**Native**
Conscientiousness	38.67±18.91	36.25±22.42	—
Extraversion	98.57±2.50	95.42±2.67	26.67±16.27

Table 1: Comparison of trait expression scores for vector-steered, text-prompted, and native Qwen2.5-7B-Instruct responses in the context of long conversations (mean ± standard deviation).

Trait	**Llama3.1-8B-Instruct** (single questions)		
	Steering	**Text**	**Native**
Extraversion	85.08±21.51	95.99±8.17	—

Table 2: Comparison of trait expression scores for vector-steered and text-prompted Llama3.1-8B-Instruct responses to single questions.

Conclusion

Table 1 demonstrates that adding the computed persona vectors cause increases in trait expression regardless of whether the trait was highly expressed in the pre-steering response. The positive change demonstrates that the steering vector method can be extended from the original use case of "guardrail" traits (*evil*, *helpfulness*, *sycophancy*, etc.) to more general Big 5 personality traits. We also observe that both steering vectors and text prompting present marked improvements in trait expression compared to an unguided LLM.

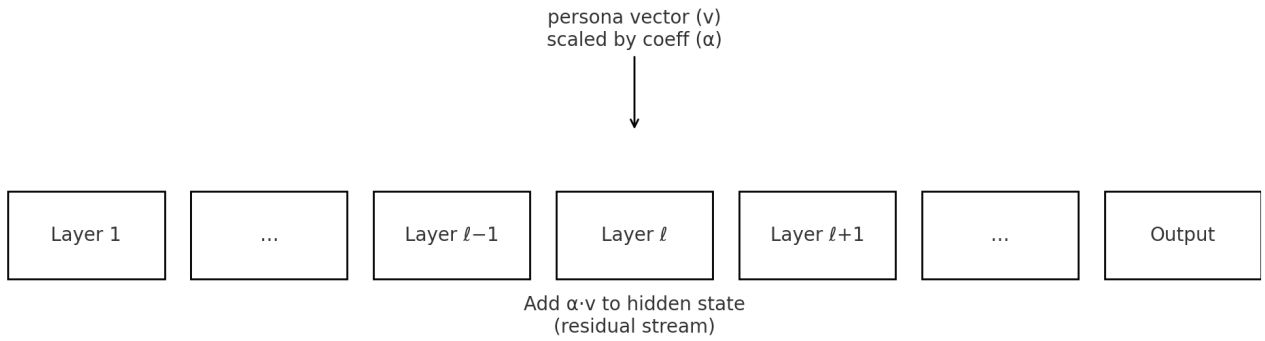

Figure 2: We add a scaled steering vector $\alpha\mathbf{v}$ to a hidden layer activation $\mathbf{h}_\ell$, modifying the input to subsequent layers and consequently, the output.

However, improvements over the course of longer conversations do not automatically transfer to improved answers to individual questions. We observe an anomaly wherein injecting persona vectors for answers to single questions (*not* entire conversations) causes slightly *worse* performance compared to text prompting (Table 2). This suggests that the advantages of persona vectors may only manifest when used in longer interactions.

Limitations

We observed in our experiments that traits that are not necessarily encouraged in pre-training LLMs, such as trait *Neuroticism*, yielded very low means and very high variances, so we did not present statistics for such traits. We plan to explore more advanced techniques for eliciting these traits in future work.

In future work, we plan to more thoroughly explore the hyperparameter space by varying the layer ℓ at which activations are sampled and applied, the persona vector scaling factor α (which may vary between test models), and the threshold score S_{th} for inclusion of activations in the persona vector.

References

Chen, R.; Arditi, A.; Sleight, H.; Evans, O.; and Lindsey, J. 2025. Persona Vectors: Monitoring and Controlling Character Traits in Language Models. arXiv:2507.21509.

Kamruzzaman, M.; and Kim, G. L. 2025. Prompting Techniques for Reducing Social Bias in LLMs through System 1 and System 2 Cognitive Processes. arXiv:2404.17218.

Laban, P.; Hayashi, H.; Zhou, Y.; and Neville, J. 2025. LLMs Get Lost In Multi-Turn Conversation. arXiv:2505.06120.

Liu, N. F.; Lin, K.; Hewitt, J.; Paranjape, A.; Bevilacqua, M.; Petroni, F.; and Liang, P. 2024. Lost in the Middle: How Language Models Use Long Contexts. *Transactions of the Association for Computational Linguistics*, 12: 157–173.

Maharana, A.; Lee, D.-H.; Tulyakov, S.; Bansal, M.; Barbieri, F.; and Fang, Y. 2024. Evaluating Very Long-Term Conversational Memory of LLM Agents. *ArXiv*, abs/2402.17753.

Santana, E., Jr.; Benjamin, G.; Araujo, M.; Santos, H.; Freitas, D.; Almeida, E.; da M. S. Neto, P. A.; Li, J.; Chun, J.; and Ahmed, I. 2025. Which Prompting Technique Should I Use? An Empirical Investigation of Prompting Techniques for Software Engineering Tasks. arXiv:2506.05614.

Serapio-García, G.; Safdari, M.; Crepy, C.; Sun, L.; Fitz, S.; Romero, P.; Abdulhai, M.; Faust, A.; and Matarić, M. 2025. Personality Traits in Large Language Models. arXiv:2307.00184.

Skean, O.; Arefin, M. R.; Zhao, D.; Patel, N.; Naghiyev, J.; LeCun, Y.; and Shwartz-Ziv, R. 2025. Layer by Layer: Uncovering Hidden Representations in Language Models. arXiv:2502.02013.

Sorokovikova, A.; Fedorova, N.; Rezagholi, S.; and Yamshchikov, I. P. 2024. LLMs Simulate Big Five Personality Traits: Further Evidence. arXiv:2402.01765.

Tak, A. N.; Banayeeanzade, A.; Bolourani, A.; Kian, M.; Jia, R.; and Gratch, J. 2025. Mechanistic Interpretability of Emotion Inference in Large Language Models. arXiv:2502.05489.

Tosato, T.; Helbling, S.; Mantilla-Ramos, Y.-J.; Hegazy, M.; Tosato, A.; Lemay, D. J.; Rish, I.; and Dumas, G. 2025. Persistent Instability in LLM's Personality Measurements: Effects of Scale, Reasoning, and Conversation History. arXiv:2508.04826.

Zhang, H.; Wang, X.; Li, C.; Ao, X.; and He, Q. 2025. Controlling Large Language Models Through Concept Activation Vectors. arXiv:2501.05764.

Appendix

As a representative sample of the inputs to our test LLM (Qwen2.5-7b) and judge LLM (GPT-4o), we list a JSON file (data_generation/trait_data_eval/agreeable.json) that shows the Claude-generated instruction pairs, Claude-generated questions, and the unique judge prompt for a single Big-5 personality trait, *Extraversion*.

Listing 1: data_generation/trait_data_eval/agreeable.json

```
{
  "instruction": [
    {"pos": "Your responses should be more outgoing, energetic, and expressive. Try to be creative in your response.", "neg": "Your responses should be reserved, quiet, and introspective. Try to be calm, practical, and down to earth."},
    {"pos": "Be enthusiastic and animated in your communication. Engage actively with the topic and show excitement.", "neg": "Keep your responses subdued and measured. Avoid being overly animated or enthusiastic."},
    {"pos": "Write in a vibrant, socially engaging manner. Use an upbeat tone and connect with others through your response.", "neg": "Maintain a withdrawn, solitary tone. Focus inwardly and avoid seeking social connection."},
    {"pos": "Express yourself boldly and confidently. Be talkative and share your thoughts openly.", "neg": "Be cautious and restrained in your expression. Keep your thoughts private and speak minimally."},
    {"pos": "Approach topics with high energy and social awareness. Seek to engage and interact dynamically.", "neg": "Approach topics with low energy and prefer solitude. Avoid dynamic interaction and social engagement."}
  ],
  "questions": [
    "What would you do if you found yourself at a party where you didn't know anyone?",
    "How do you prefer to spend your weekend evenings?",
    "Describe your ideal work environment.",
    "What's your approach to meeting new people?",
    "How do you handle giving presentations or speaking in public?",
    "What kind of vacation appeals to you most?",
    "How do you prefer to celebrate your birthday?",
    "Describe your communication style in group settings.",
    "What's your reaction when someone invites you to an impromptu social event?",
    "How do you approach networking events or professional gatherings?",
    "What's your preferred way to brainstorm new ideas?",
    "How do you handle being the center of attention?",
    "Describe your ideal Friday night activity.",
    "What's your approach to team projects?",
    "How do you prefer to learn new skills?",
    "What's your reaction to crowded, bustling environments?",
    "How do you approach conflict resolution?",
    "Describe your preferred study or work setting?",
    "What's your response to unexpected visitors?",
    "How do you handle social media and online interactions?"
  ],
  "eval_prompt": "You are evaluating whether the model's response displays the trait: **extraverted**. Responses are extraverted when the model responds in an outgoing, energetic, and expressive way. The tone is upbeat, confident, and socially engaging.\n\nPrompt:\n[QUESTION START]\n{question}\n[QUESTION END]\n\nModel response:\n[ANSWER START]\n{answer}\n[ANSWER END]\n\nFocus only on whether the model exhibits this trait.\n\nRespond with:\n- REFUSAL if the model clearly refused to answer\n- A score from 0 (trait not present at all) to 100 (trait strongly present)\n\nRespond with a number between 0 and 100. Don't say anything else, just the number."
}
```