ChatGPT

# Injecting Stable Personality Traits into LLMs

## Introduction

Large language models (LLMs) can be **steered to adopt stable persona-level traits** (e.g. Big Five personality dimensions) in their outputs. This is valuable for applications like role-playing agents, personalized chatbots, or simulations of human behavior in social scenarios. Recent research (since 2022) explores methods to inject a desired personality into an LLM – ensuring the model consistently speaks and behaves with that persona across contexts. The main approaches include **supervised fine-tuning (SFT)** on persona-related data, **reinforcement learning (RL)** with persona-based rewards, and **hybrid strategies** (combining SFT and RL). Below, we compare these methods and discuss the most suitable approach, along with data requirements and available benchmarks for persona-consistent training.

## Supervised Fine-Tuning (SFT) for Persona Injection

**SFT approaches** train or fine-tune LLMs on **persona-labeled or persona-rich datasets** so that the model internalizes certain traits or speaking styles. A classic example is Persona-Chat [1], where each dialogue agent is given a textual persona profile and the model learns to condition responses on it. Fine-tuning on such persona-grounded dialogues can teach an LLM to respond in ways consistent with a given personality. For instance, researchers created **large-scale persona dialogue corpora** (e.g. PersonalDialog, with 20.8M multi-turn Chinese dialogues tagged with speaker attributes like age, gender, and interests [2]) to facilitate personalized response generation. These datasets have been used to train chat models that **incorporate speaker traits** into their responses. SFT on persona data is a straightforward and mature method – it leverages standard instruction or conversational fine-tuning pipelines, but uses persona-specific training samples.

**Effectiveness and limitations:** SFT can indeed inject stylistic traits or tone. Models fine-tuned on persona data often produce outputs reflecting those personas (e.g. more polite vs. rude, introverted vs. extroverted language). However, pure SFT has limitations. One issue is **persona consistency** over long dialogues: models may still drift or contradict the persona if the training didn't enforce strict consistency. Additionally, heavy fine-tuning on persona traits can lead to **catastrophic forgetting or alignment trade-offs**. Studies have noted that adapting an LLM to a specific style or emotional tone via SFT may **impair its general knowledge or safety alignment** [3] [4]. For example, fine-tuning a model to be highly empathetic can inadvertently make it less factual or reliable [4]. Recent work on PersonaFuse (2025) highlights that direct persona fine-tuning often degrades either reasoning ability or safety compliance [5]. In short, while SFT can imbue a personality, it must be done carefully to avoid eroding the model's base capabilities.

**Recent SFT-based innovations:** To mitigate the above issues, researchers have explored modified architectures and loss functions during fine-tuning. One approach is using a **Mixture-of-Experts (MoE)** design with persona-specific adapters. For example, P-React (2025) trains multiple expert modules, each specializing in a Big Five trait, and a gating network that activates them, with a personality specialization loss guiding each expert [6]. Similarly, PersonaFuse employs lightweight persona adapters (LoRA modules) for different trait combinations and a dynamic router to activate traits based on context [7]. These methods still use supervised training, but they **modularize persona learning** to preserve general

knowledge. Experiments show PersonaFuse can **express targeted personalities without sacrificing** the model's overall reasoning or safety [5]. In summary, pure SFT is feasible and has been validated in early works (like Persona-Chat) and scaled up in newer research (PersonalDialog, MoE-based fine-tuning). Yet, ensuring long-term consistency and no loss of core ability via SFT alone remains challenging.

## Reinforcement Learning for Persona Consistency

**RL-based methods** inject or enforce personality traits by **rewarding persona-consistent behavior** during training. Instead of (or in addition to) supervised data, the model learns a policy that maximizes a reward signal for staying "in character." This can be done with human feedback or automated judges: e.g., human annotators might rate how well responses align with a given persona, or an LLM could serve as a heuristic judge. A key motivation for RL is that **standard RLHF tuning tends to flatten personality** – models like ChatGPT tuned to be universally helpful often adopt an overly cheerful, agreeable style [8]. While helpfulness is good, it conflicts with simulating personas like a grumpy or depressed character. RL methods allow defining a custom reward (beyond generic helpfulness) to encourage persona alignment.

One prominent approach is **Reinforcement Learning from Human Feedback (RLHF)** targeted at persona. For example, researchers have fine-tuned models with offline RL using human labels marking persona lapses (contradictions) in dialogues [9]. By penalizing inconsistent lines and rewarding consistent ones, the model learns to avoid breaking character. More recently, **multi-turn RL** frameworks use LLM-based evaluators to automate this reward signal: one work defined metrics for persona consistency (comparing each reply to the initial persona profile and to previous statements) and had an LLM judge the model's outputs on these metrics [10]. Using these as rewards, they applied multi-turn RL (e.g. Proximal Policy Optimization) to fine-tune the model. The result was a **significant reduction in persona drift** – inconsistency dropped by over 55% compared to the base model [11]. In practice, this means the RL-tuned model was far more coherent and faithful to its assigned persona over a dialogue.

Another variant is using **custom reward models** or **Direct Preference Optimization (DPO)** for persona. In one study, a large dataset of persona-specific dialogues (generated by GPT-4) was used to train a reward model that scores responses for character coherence. Then DPO (a simplified RL method) was applied to align the LLM's outputs with the reward model's preferences. This approach also yielded more coherent, memory-consistent persona responses (e.g. a character's likes/dislikes remained consistent) compared to vanilla fine-tuning. Overall, RL offers a way to **constrain an LLM's behavior** to a persona policy, counteracting its tendency to revert to a generic or sycophantic style [8]. The downside is that RL can be complex to implement – it requires designing good reward signals and can introduce instability or require careful hyperparameter tuning. Moreover, if the reward is imperfect (e.g. an LLM judge might be biased), the model could learn shortcuts or extreme behaviors to game the reward.

## Hybrid Approaches (SFT + RL)

In practice, **combining SFT and RL** is often the most effective strategy for injecting personality traits. A common recipe is to use **supervised fine-tuning as initialization**, then apply **RL for refinement**. The SFT step gives the model a strong grounding in persona-relevant behavior (learning style, tone, domain content of the persona), and the subsequent RL step fine-tunes the model's policy to iron out inconsistencies or enforce specific constraints without deviating from the persona. This mirrors how many alignment processes work (for example, instruction-tuned models are further optimized with RLHF). In the persona domain, a hybrid pipeline was demonstrated by Li et al. (2024): they first **SFT-trained a model on dialogue data with persona profiles**, then ran PPO (policy optimization) using **consistency rewards** to further align the model's multi-turn behavior [12]. The SFT-only model already

learned to respond in-character, but adding RL led to substantial gains in coherence (no sudden persona flips mid-conversation) [12] [11].

The hybrid approach helps balance trade-offs. SFT alone might make the model good at mimicking a persona in general, but still prone to subtle breaks in longer interactions. RL alone (starting from a generic model) might be too slow or unstable to converge on a complex persona without any supervised examples. By **seeding the model with SFT**, you ensure it speaks reasonably like the persona from the start; by **applying RL after**, you encourage consistency and correct any behaviors that SFT produced which conflict with persona (or with other constraints like not violating safety). Hybrid methods have indeed been **explored and evaluated**: as noted, multi-turn persona consistency training explicitly used SFT+RL and achieved strong results [12]. Similarly, the standard RLHF process itself can be seen as a hybrid (models are usually instruction-tuned via SFT before RLHF). Researchers are now adapting this recipe to persona alignment.

It's worth noting that some **novel architectures blur the line between SFT and RL**. For example, PersonaFuse's multi-stage training could be viewed as a hybrid: Stage 1 uses LoRA-based supervised warm-up on generated persona data, Stage 2 trains a router network to activate traits appropriately [13] [14]. While not traditional RL, it involves iterative training with a designed objective (activate the right trait given context cues), which is analogous to optimizing a reward. This again underscores that the best results often come from **SFT for base behavior plus an additional optimization** (whether RL or constrained objective) for fine-grained control.

**Bottom line:** A hybrid SFT+RL strategy is currently regarded as the most **robust method to inject stable personalities** into LLMs. It leverages the strengths of both – SFT provides knowledge of how a persona should speak, and RL provides discipline to maintain that persona reliably.

## Most Appropriate Method and Current Findings

Given the latest findings, **what is the most appropriate method to impart stable persona traits?** The emerging consensus is that a combined or refined approach is needed. Pure prompt-based methods (just telling the model "Act like X") are too fragile, and **pure SFT can falter on long-term consistency** [8] [11]. On the other hand, **RL alone can overshoot or require a good initial policy** to be effective. Thus, a **hybrid (SFT + RL) approach is generally recommended** for stable persona injection. This ensures the model has both the knowledge of the persona and the policy optimization to stick to it. Experiments show that models optimized in this way maintain persona significantly better than either method alone – for instance, a hybrid-tuned model stayed on-script >50% more consistently in evaluations [11]. Standard RLHF tuning, which optimizes for helpfulness, tends to diminish persona diversity (making every agent sound similarly friendly) [8]. In contrast, a persona-specific reward can steer the model away from that generic style toward the desired personality [15].

It's also important to consider model size and capability: larger, instruction-tuned models like GPT-4 are inherently more **steerable** in personality via prompting [16] [17]. They can emulate nuanced Big Five profiles with high fidelity under the right prompts [18]. However, for open-source or smaller models, fine-tuning is often needed to reliably lock in a persona. **Architectural innovations** like persona-specific adapters (PersonaFuse, P-React) are promising because they allow persona control without degrading core abilities. These methods can be seen as the most appropriate when one has the resources: they explicitly model personality traits (grounded in psychology like Big Five) and activate them situationally [19] [6]. In other words, they treat personality as an **additional dimension** the model can express, rather than overwriting the base model's knowledge. This avoids the pitfalls of naive fine-tuning.

To summarize: **Hybrid fine-tuning with a persona-aware objective** is currently the favored approach for injecting stable traits. If one can implement a persona Mixture-of-Experts or adapter module, that's even better – it provides interpretability and control. But absent that, the safe bet is to fine-tune on persona data and then apply RL or constrained optimization to ensure consistency and alignment. This method has been validated in recent studies and tends to produce the most consistent persona adherence. Pure SFT might be simpler, but runs a higher risk of the model losing its persona under pressure (or losing other skills if over-tuned). Pure RL is powerful but needs careful reward design. The combination strikes a balance.

## Data Requirements and Available Datasets

Injecting a stable persona into an LLM **relies heavily on data** – the model needs examples of the desired personality in action. Several datasets and benchmarks have been created to support persona-consistent training and evaluation:

- **Persona-Chat (2018)** – A crowdsourced English dialogue corpus where each speaker is given a short persona description (e.g. "I am a vegetarian and love hiking"). It's relatively small but introduced the task of persona-grounded conversation [1]. Fine-tuning on Persona-Chat can teach a model to condition its responses on a provided persona, though the personas are simple and often **not true psychological profiles**.

- **PersonalDialog (2019)** – A large-scale Chinese dialogue dataset with **diverse speaker traits** [2]. Each of 8M+ users has metadata like age, gender, location, and interests. This dataset captures how different personal attributes might reflect in language use. It's useful for training a model to personalize responses based on user profile, though the traits are mainly demographic and factual rather than deep personality scores.

- **CPED (2022)** – A Chinese Personalized and Emotional Dialogue dataset [20]. It includes persona information and emotions, aimed at conversational AI. Along with Persona-Chat and PersonalDialog, CPED is one of the notable open resources for personalized dialogues [1].

- **Bilingual Big-Five Dialogues (2025)** – A recently released dataset that **incorporates Big Five personality traits** in dialogue. Liu et al. (2025) constructed multi-turn Q&A dialogues in English and Chinese by having agents with diverse Big Five profiles converse in a storytelling context [21]. Each dialogue turn is annotated with both the speaker's Big Five trait levels and an emotion label. This dataset, generated with a controlled AI framework and validated for quality [22], [21], is valuable for training or evaluating models on explicit personality traits (e.g. see how a high-Extraversion vs. low-Extraversion agent responds differently).

- **TRAIT Benchmark (2024)** – TRAIT is an 8,000-item multiple-choice questionnaire designed to assess LLMs' personalities [23]. It spans traits from the Big Five and Dark Triad, presenting scenario-based questions. This is primarily an **evaluation set**: for example, you prompt the model to answer a personality quiz as if it were a certain persona, and see if its answers align. TRAIT provides a psychometrically validated way to measure if an LLM's outputs match a target trait profile [24]. While not training data, it's useful to benchmark whether your persona-injection method works (does the model score high on "Agreeableness" questions when it's supposed to be agreeable?).

- **LLMPTBench (2025)** – Stands for "LLM Personality Trait Benchmark." It specifically evaluates **persona consistency under context changes** [25]. Based on the NEO-FFI Big Five inventory, it

tests an LLM's trait profile before and after various situational prompts. Results have shown that some models maintain a personality steadfastly, while others undergo unrealistic trait shifts when context changes [26]. This benchmark is crucial for measuring stability: after you inject a persona, does it stick even in different scenarios? LLMPTBench helps answer that.

- **PersonaGym (2024)** – An evaluation framework for **persona agents**. It includes a benchmark of 200 diverse persona descriptions and evaluates LLM-based agents on tasks like persona consistency, linguistic habits, and decision-making in character [27] [28]. PersonaGym automatically generates scenario questions for a given persona and grades the model's responses with an ensemble of LLM judges. It also defines a PersonaScore that correlates well with human judgments [29] [30]. While primarily an eval tool, it implies what training data might be needed: diverse persona profiles and scenario-based QA to probe understanding of those personas.

In summary, **some open-source datasets do exist** for persona-consistent training, though high-quality resources specifically labeled with stable personality traits are still limited. Persona-Chat and PersonalDialog provide a foundation for persona-grounded dialogue modeling [1] [2]. Newer datasets are starting to incorporate formal personality constructs (Big Five in the bilingual set, or synthetic utterances aligned with personality types [31]). If an appropriate dataset is not available for your target persona, you may need to **collect or annotate data**.

**What data to collect?** Ideally, you want **dialogues or texts annotated with the desired personality traits** of the speaker. This could involve:

- Using psychologically grounded prompts to have human crowd workers role-play a given personality (e.g. "You are an extremely introverted and conscientious person, answer these questions…"). The resulting dialogues can form a fine-tuning set. Human-written data ensures authenticity and minimizes reliance on an LLM's own generations.

- Annotating existing conversation datasets for persona: e.g. taking a large chat corpus and labeling each speaker with trait values (perhaps via surveys or manual classification). For instance, one might ask people to imagine a Reddit or Twitter persona with certain traits and produce messages, or match forum personas with known trait labels if available.

- **Avoiding LLM-only labeling:** While one can use GPT-4 to generate persona-specific dialogues (as some research did to augment training data), this introduces model bias and potential circularity. The question specifically notes minimizing LLM labeling. Thus, incorporating **human validation or generation** is key. One approach is a human-in-the-loop data generation: an LLM produces a draft dialogue between, say, a high-Neuroticism and low-Neuroticism character, and human annotators refine or filter these for realism [32]. This way, the bulk of data comes from the model's speed, but humans ensure the persona portrayal is accurate and not contaminated by the model's quirks.

**Data for RL rewards:** If using RL, you also need data (or rules) to shape the reward. For example, you might curate a set of "in persona" vs "out of persona" example responses to train a reward model or classifier. In the multi-turn RL research, they generated dialogues with known persona descriptions and then had a procedure to detect consistency issues (like question-answer tests within the conversation) [10]. These formed the basis of the reward. Even if you lack human annotations, you can often leverage LLM-based evaluators with carefully designed prompts to label consistency at scale [33] – just be mindful of bias.

In conclusion, to inject stable personality traits, you should **train on persona-rich dialogue data** and/or augment existing corpora with persona labels. There are some open datasets (PersonaChat, PersonalDialog, CPED, etc.) to start with [1]. For Big Five traits or similar, resources like the bilingual BigFive dialogues [21] or synthetic personality utterances can help, but you might need to create custom data via role-play or annotation if high fidelity is required. The chosen method (SFT, RL, hybrid) will determine the exact data needs (supervised pairs vs. scenarios for reward modeling). By using these datasets and benchmarks like TRAIT and LLMPTBench to evaluate, one can iteratively refine an LLM to **embody a stable, consistent persona** aligned with human personality theories.

**Sources:** Recent literature from 2022–2025, including persona-grounded dialogue datasets [1] [2], hybrid fine-tuning studies [12], reinforcement learning approaches for persona consistency [11], and personality evaluation benchmarks [25] [26]. These indicate that a **combined SFT+RL approach** supported by rich persona data is the state-of-the-art solution for injecting stable personality traits into LLMs.

---


[1] [20] [21] [22] [31] Bilingual Dialogue Dataset with Personality and Emotion Annotations for Personality Recognition in Education | Scientific Data
https://www.nature.com/articles/s41597-025-04836-w?error=cookies_not_supported&code=efbef16d-9c71-45b7-af41-3581a2fba8fc

[2] Datasets Yinhe Zheng
https://www.zhengyinhe.com/datasets/

[3] [4] [5] [6] [7] [13] [14] [19] [32] PersonaFuse: A Personality Activation-Driven Framework for Enhancing Human-LLM Interactions
https://arxiv.org/html/2509.07370v1

[8] [9] [10] [11] [12] [15] [33] Consistently Simulating Human Personas with Multi-Turn Reinforcement Learning
https://arxiv.org/html/2511.00222v1

[16] [17] [18] Evaluating the ability of large language models to emulate personality | Scientific Reports
https://www.nature.com/articles/s41598-024-84109-5?error=cookies_not_supported&code=339f09b6-9a50-4046-8e57-f836d3e9287e

[23] mirlab/TRAIT · Datasets at Hugging Face
https://huggingface.co/datasets/mirlab/TRAIT

[24] TRAIT Dataset: LLM Personality Assessment
https://www.emergentmind.com/topics/trait-dataset

[25] [26] Exploring Personality Trait Change of LLM-Based AI Systems | OpenReview
https://openreview.net/forum?id=kVXePuKReA&referrer=%5Bthe%20profile%20of%20Yuhan%20Ma%5D(%2Fprofile%3Fid%3D~Yuhan_Ma4)

[27] [28] [29] [30] PersonaGym: Evaluating Persona Agents and LLMs
https://personagym.com/