Extended Abstract

Motivation In multi-turn dialogues, users often revise or contradict their earlier preferences as their intent evolves, new context emerges, or misunderstandings occur. These contradictions pose a unique challenge for language models: without mechanisms to detect and adapt to conflicting signals, the model may rely on outdated or inconsistent preferences, producing incoherent responses that degrade dialogue quality. The problem we address in this work is enabling LLMs to recognize when user-side preference contradictions occur and to generate responses that reflect the user's most current intent.

Method We propose a methodology for improving LLM alignment with contradicting user preferences through targeted synthetic data generation and preference-based fine-tuning. Our novel approach begins by generating multi-turn conversations where user preferences contradict earlier statements, based on a six-category typology of preference changes. We incorporate Chain-of-Thought (CoT) prompting to encourage the model to reason explicitly about contradicting preferences. For each identified contradiction, we create a pair of agent responses: a preferred response that adapts to the updated preference and a dispreferred response that ignores it. We then fine-tune a base Qwen model using Direct Preference Optimization (DPO), specifically employing a sigmoid-based DPO loss that encourages the model to assign higher probability to preferred responses. This training teaches the model to prioritize adaptive, preference-consistent responses, improving its ability to handle nuanced user interactions.

Implementation We implemented a two-pass LLM-based data generation pipeline. In the first pass, we sampled personas from PersonaHub (Ge et al., 2025) and prompted ChatGPT-4o to generate multi-turn dialogues exhibiting preference shifts, using CoT prompting to enhance reasoning quality. In the second pass, Gemini-2.5-Flash annotated contradictions within these dialogues, classified them into six predefined categories, and generated paired agent responses (preferred and dispreferred). We produced both a CoT-augmented and a non-CoT dataset. We fine-tuned Qwen1.5-1.8B with Low-Rank Adaptation (LoRA) on both datasets using DPO with a sigmoid-based binary cross-entropy loss on the log-odds of chosen versus rejected responses (Eq. 1). Training prompts included conversation history, contradiction context, and CoT examples for the reasoning variant. Finally, we evaluated model performance by comparing Qwen models fine-tuned with and without CoT augmentation, and by comparing the Qwen+CoT model against GPT-4o. We tracked training loss, validation loss, preference accuracy (how often the model prefers the chosen response), the log-probability gap between chosen and rejected responses, and the reward margin (the confidence difference between preferred and dispreferred responses). We also used a held-out test set of conversations and computed cosine similarity between the fine-tuned Qwen agent's responses and the preferred responses, and between GPT-4o's responses and the preferred responses.

Results The Qwen model fine-tuned with CoT reasoning outperforms the Qwen model trained without CoT across multiple metrics. It demonstrates faster convergence, reaching near-zero evaluation loss several steps earlier than the model trained without CoT. Its preference accuracy reaches 100% by evaluation step $\sim$25, while the non-CoT model only achieves perfect accuracy around step $\sim$60. In reward-based evaluation, the CoT-trained model maintains a larger and more stable gap between chosen and rejected rewards throughout training, with a final reward margin converging at approximately 12, compared to a margin of around 10 for the baseline model. These results indicate that CoT prompting improves the model's ability to separate preferred from dispreferred responses with greater confidence. We also compared the Qwen+CoT model to GPT-4o on a held-out test set. The Qwen+CoT model achieved a higher match rate (0.432 vs. 0.338) and higher average similarity (0.362 vs. 0.330) to human-preferred responses, demonstrating improved semantic alignment and stronger preference consistency relative to GPT-4o under the same evaluation framework.

Discussion Our study is currently limited by focusing on synthetic dialogues containing a single annotated contradiction per conversation, which may not fully capture the complexity of natural user interactions. Future work should address more diverse conversation scenarios. First, we aim to extend data generation to include more complex contradiction patterns, such as multiple overlapping preference shifts and longer context dependencies. Second, we plan to incorporate human-in-the-loop evaluation using pairwise preference trials and ELO scoring to better assess alignment with user

intent. Finally, we will explore adaptive CoT prompting to enable the model to dynamically reason about preference contradictions during inference without relying on static, manually constructed examples.

Conclusion Our study demonstrates that incorporating CoT prompting into the data generation and fine-tuning process improves an LLM's ability to handle contradicting user preferences. CoT-augmented DPO fine-tuning enables faster convergence, stronger preference alignment, and better semantic consistency between agent responses and preferred responses. These findings highlight the value of integrating reasoning signals when training models to manage complex, evolving user preferences in dialogues. Our results are important for supporting more coherent, context-aware, and trustworthy LLMs in multi-turn interactions.

Less Details, But Be Thorough: Addressing Contradicting User Preferences in Multi-turn LLM-based Conversation

Eugenie Shi
Department of Computer Science
Stanford University
yqshi@stanford.edu

Haorui Guo
Department of Computer Science
Stanford University
haorui@stanford.edu

Shuojia Fu
Department of Civil and Environmental Engineering
Stanford University
shuojia@stanford.edu

Abstract

We address the challenge of enabling Large Language Models (LLMs) to handle contradicting user preferences in multi-turn dialogues. We propose a methodology combining synthetic data generation, a six-category typology of preference contradictions, Chain-of-Thought (CoT) prompting, and Direct Preference Optimization (DPO) fine-tuning. Using a two-pass data pipeline, we generate annotated dialogues with conflicting preferences and paired agent responses, then fine-tune Qwen1.5-1.8B with and without CoT. Our results show that CoT-augmented DPO improves preference accuracy, reward margin, and semantic alignment, outperforming both the Qwen model fine-tuned without CoT and GPT-4o on a held-out test set. Future work will explore more complex contradiction patterns, human-in-the-loop evaluation, and adaptive CoT prompting.

1 Introduction

Detecting contradictions, inconsistencies, and preference reversals in user input is essential for developing reliable and trustworthy large language models (LLMs). In multi-turn dialogues, users often revise their preferences, shift goals, or unintentionally contradict themselves due to evolving intent, ambiguous language, or misunderstandings of the dialogue context. Such user-side contradictions pose a significant challenge to maintaining coherent and contextually appropriate interactions. If not properly detected, these inconsistencies can cause LLMs to misinterpret the user's current intent and generate inappropriate or misaligned responses, reducing the overall quality and reliability of the dialogue.

While recent work has advanced the alignment of LLMs with human preferences, most approaches assume that user preferences are stable and coherent throughout a conversation. In practice, users frequently revise, shift, or contradict their earlier preferences during multi-turn dialogues, either intentionally or unintentionally. However, existing preference optimization research, including InstructGPT (Ouyang et al., 2022), MTPO (Shani et al., 2024), and P-RLHF (Li et al., 2024), tends to treat user feedback as consistent, which can lead to misaligned responses when contradictions arise. Addressing this gap is essential for enabling LLMs to maintain coherent, context-aware interactions and correctly interpret the user's most recent intent in the presence of conflicting signals.

In this study, we propose a method for training LLMs to recognize user-side contradictions and adjust their responses accordingly. We construct a multi-turn dialogue dataset that explicitly includes diverse forms of user preference contradictions, organized into a six-category typology. For each scenario, we generate paired agent responses: a preferred response that adapts to the user's updated intent, and a dispreferred response that fails to do so. We then fine-tune a Qwen LLM using Direct Preference Optimization (DPO), guided by Chain-of-Thought (CoT) prompting to improve the model's reasoning about conflicting preferences. Our goal is to equip LLMs with contradiction awareness, enabling them to produce more consistent, contextually aligned responses in evolving dialogues. This capability is especially useful in interactive domains such as customer support, education, and everyday dialogue systems, where adapting to evolving user preferences helps maintain a more coherent and satisfying user experience.

2 Related Work

The alignment of Large Language Models (LLMs) with human judgments has predominantly been advanced by Reinforcement Learning from Human Feedback (RLHF). The foundational principle of RLHF involves leveraging human assessments, often in the form of pairwise comparisons or rankings of model-generated responses, to refine the LLM's learning process (Ouyang et al., 2022). This feedback is typically used to train a reward model that predicts human preferences, which in turn guides the optimization of the LLM's policy.

Traditional RLHF works by emulating preferences at the single-turn level, limiting their capabilities in multi-turn conversations, which is natural and frequent in human dialogues. PrefEval (Zhao et al., 2025) introduces a benchmark showing that state-of-the-art LLM chatbots often struggle to follow user preferences through long multi-turn dialogues. Extensions like Multi-turn Preference Optimization (MTPO)(Shani et al., 2024) learn from feedback on entire conversations, enabling agents to develop strategies considering long-term interaction quality. However, its learning signal relies on consistent preference judgment comparing two complete conversational trajectories, and does not include mechanisms to explicitly identify or resolve contradictory preferences within a multi-turn conversation.

Another notable development in this field is personalization. Recent methods like Personalized-RLHF (P-RLHF) (Li et al., 2024) aim to tailor LLM responses by learning user-specific embeddings, improving from traditional RLHF, which assumes all human preferences share the same distribution. While this approach provides a general framework for personalization, handling contradictions within a single user's conversation is not desirable. If a user provides contradictory statements, the embedding might settle into a state reflecting the dominant trend or average across conflicting signals. Other recent works including (Poddar et al., 2024) investigate divergent preferences across populations. While they focus on inter-user preference divergence, they challenge the underlying assumption in many RLHF frameworks of a single, coherent preference function.

Therefore, the identified gap prompts a compelling research problem to develop a robust RL mechanism designed specifically for managing user-introduced contradictions, leading to a more robust, intelligent, and trustworthy LLM.

3 Method

3.1 Synthetic Data Generation

We generated synthetic data using a large language model (LLM) because existing public datasets do not support our focus: identifying and modeling various types of contradictions in user-stated preferences during dialogue with an agent. Since these contradictions naturally arise in conversations between users and agents, generating such interactions with an LLM is a fitting and purposeful choice—it allows us to simulate the exact conditions our system is designed to handle.

To ensure diversity and realism, we followed a persona-driven approach, sampling a wide range of base personas from PersonaHub (Ge et al., 2025), a repository of roughly one billion web-curated personas. For each selected base persona, an LLM enriched it with further details to create a more developed character. Following persona enrichment, a relevant conversational topic was chosen, covering diverse domains such as financial planning and dating advice.

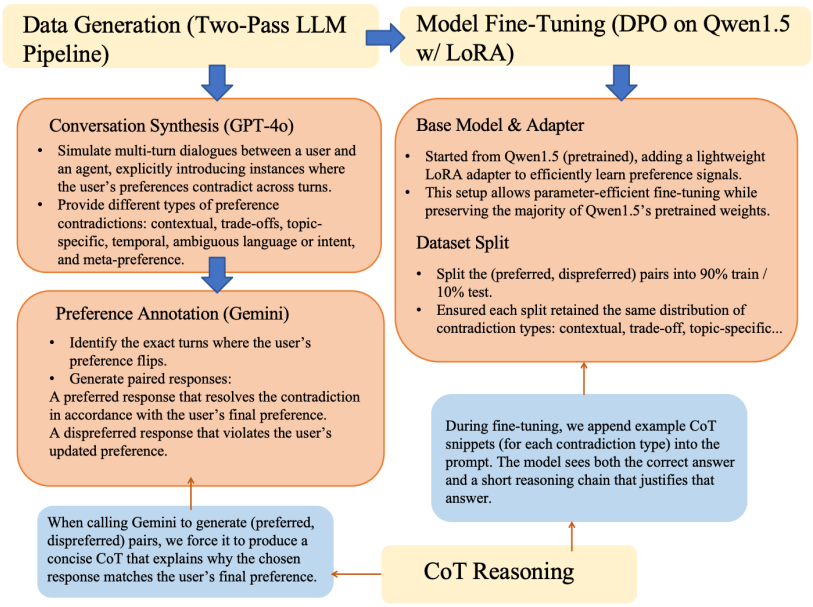

Figure 1: Method Overview

The core of our data generation relied on a two-pass LLM pipeline:

1. First Pass (Conversation Generation): In the initial pass, an LLM GPT-4o (OpenAI, 2024) was tasked with creating a multi-turn conversation between the enriched user persona and an AI agent. This step focused on generating naturalistic dialogue flows where preference changes could organically emerge.
2. Second Pass (Annotation and Response Pair Generation): In the subsequent pass, a different LLM Gemini-2.5-flash (Google DeepMind, 2025) performed two tasks:
   (a) Contradiction Annotation: This LLM analyzed the generated conversation to identify and annotate instances of user preference contradictions. The annotations included details such as the previous preference, the newly expressed (current) preference, and the specific category of the preference change (based on our defined categories, as detailed in Section 4.1.1).
   (b) Response Pair Generation for DPO: The same LLM was also tasked with generating a pair of agent responses to the final user turn in the conversation: a "preferred" response that appropriately addresses the identified preference change, and a "dispreferred" response that fails to do so (e.g., by adhering to an outdated preference or ignoring the change). These response pairs serve as the preference signal for training our model via Direct Preference Optimization (DPO).

3.2 Chain-of-thought prompting

We experimented with chain-of-thought prompting in our second pass. The rationale behind employing CoT was to guide the annotator LLM (Gemini-2.5-flash) through a structured reasoning process, encouraging it to explicitly identify and articulate the steps leading to its annotation decisions. This approach aims to improve the reliability of identifying conflicting preferences and correctly categorizing the type of contradiction.

The CoT prompt instructed the LLM to first engage in a detailed reasoning phase before producing the final output. For contradiction annotation, this involved a sequence of analytical steps:

1. **Identify All Preference-Indicating Turns:** The LLM was asked to scan the entire conversation and list every user turn that contains a statement of preference or a direct request, quoting the relevant phrases.

2. **Detect Conflicting Pairs:** It then needed to compare these candidate turns chronologically to find pairs expressing opposing or incompatible preferences, noting the turn IDs and quoted text for any identified conflicts.

3. **Establish Chronology of Conflicting Preferences:** For each conflicting pair, the LLM determined which utterance represented the earlier (original) preference and which represented the later (changed) preference.

4. **Classify Contradiction Type:** The LLM then classified the identified contradiction according to our six predefined categories (Contextual, Trade-off, Topic-Specific, Temporal, Ambiguous Intent, Meta-Preference), providing a brief justification for its choice by linking the conversational evidence to the category definition.

5. **Summarize Reasoning:** Throughout this process, the LLM was prompted to articulate its reasoning, for example, by stating, "I identified a preference for 'short answers' in Turn 2, and a conflicting preference for 'more detail' in Turn 5, which aligns with a trade-off contradiction."

Only after completing this internal reasoning process was the LLM instructed to generate the structured annotation output for the identified preference contradiction. This structured CoT approach was designed to make the annotation process more methodical and transparent. We also apply CoT prompting in the Response Pair Generation task as well.

3.3 Direct Preference Optimization (DPO)

We conducted two Direct Preference Optimization (DPO) training experiments on our generated dataset: one using a dataset with reasoning and one without reasoning. In both experiments, the chosen and rejected responses were extracted from the corresponding generated dataset. However, we used two different prompt formats for the two settings. For the data without reasoning, the prompt included the full conversation history, the specific contradictory statements, and instruction to the agent to generate a relevant and coherent response to the last user message. For the data with reasoning, the prompt contained all of the above, but also included chain-of-thought (CoT) examples. Specifically, for each contradiction type, we provided illustrative contradictory examples along with its chosen and rejected responses, and the rationale explaining why the chosen response was preferred. This additional reasoning was designed to help the model learn the underlying human logic and thus better align its responses with implicit user preferences.

We selected Qwen/Qwen1.5-1.8B as the base model and applied a Low-Rank Adaptation (LoRA) module to reduce the number of trainable parameters during fine-tuning. We use a sigmoid-based loss function for pairwise preference comparison (Eq 1):

$$\mathcal{L}_{\text{DPO}} = -\log \sigma \left(\beta \cdot [\log \pi_\theta(y_{\text{chosen}} \mid x) - \log \pi_\theta(y_{\text{rejected}} \mid x)] \right) \tag{1}$$

For model evaluation, we employed three metrics to assess preference performance. First, We used accuracy, which is a binary metric indicating the frequency the model assigns a higher log-probability to the chosen response than to the rejected one out of all the samples. Second, we used the reward margin, which reflects the model's confidence in its preference. A larger positive margin indicates stronger preference alignment. The rewards are calculated as the log-probabilities under the current model π_θ, and reward margin is calculated as the difference between the rewards of the chosen and rejected responses, as Eq 2:

$$\begin{aligned} r_{\text{chosen}} &= \log \pi_\theta(y_{\text{chosen}} \mid x) \\ r_{\text{rejected}} &= \log \pi_\theta(y_{\text{rejected}} \mid x) \\ \Delta r &= r_{\text{chosen}} - r_{\text{rejected}} \end{aligned} \tag{2}$$

Finally, we examined the DPO loss on the evaluation set (Eq 1), which quantifies how well the model differentiates preferred responses from rejected ones. Lower loss values indicate better preference learning.

4 Experimental Setup

4.1 Dataset Description

Our dataset consists of synthetic conversations generated to capture contradictions in user preferences during dialogues with an AI agent. The data is structured into `JSON` files that include the following components:

1. **Persona Information:** Each file begins with a description of the user's persona (e.g., profession, background, and interests), providing context for the conversation. This persona is enriched from a base profile to create diverse and realistic characters.
2. **Conversation:** The core of the dataset is the conversation between the user and the AI agent. This section includes a series of dialogue turns, each represented by:
   - `turn_id`: Unique identifier for each dialogue turn.
   - `speaker`: Indicates whether the turn is by the user or agent.
   - `text`: The content of the dialogue turn.
3. **Contradiction Annotations:** After the conversation, contradiction annotations are provided, identifying points where the user's preferences change between dialogue turns. Each contradiction includes:
   - `curr_pref`: The user's current preference at the time of contradiction.
   - `prev_pref`: The user's previous preference.
   - `curr_turn_id` and `prev_turn_id`: The specific turns where the contradictions are found.
   - `type`: The type of contradiction.
4. **Response Pairs:** Each conversation is also annotated with two types of agent responses:
   - `preferred_response`: A response that appropriately addresses the user's true preference.
   - `dispreferred_response`: A response that fails to accommodate the change in preference.

4.2 Categories of preference changes

To systematically address the challenge of shifting user preferences, we came up with a typology of six distinct categories of preference changes that can lead to contradictions within a multi-turn user-LLM conversation. Contextual Contradictions occur when a preference changes due to specific situational factors, such as a user who generally likes sweets but declines them when feeling unwell. Trade-off Contradictions emerge when two incompatible preferences are expressed, for instance, initially requesting brief answers but later indicating a need for more detail. Topic-specific contradictions describe variations in preference across different tasks or domains, like wanting short answers for factual queries but more elaborate responses for emotional support. Temporal Contradictions reflect longer-term shifts in opinion, for example a user who previously enjoyed a genre but now finds it stressful. Ambiguous intent captures instances where the user provides conflicting signals within their language, such as expressing a desire for fast replies while also stating the agent can take its time. Finally, Meta-preferences define rules about how preferences themselves should be handled by the agent, for example, instructing the LLM to "keep it brief unless I ask for details."

We incorporated these contradiction types throughout our pipeline: first, in conversation generation prompts, where we explicitly used the typology to guide the creation of diverse examples of preference changes; and second, in the DPO fine-tuning prompts, where we provided type-specific examples of preferred and dispreferred responses for each contradiction category to augment Chain-of-Thought (CoT) reasoning. These categorizations thus not only provide a structured approach to understanding the dynamic nature of user preferences in dialogue, but also enable targeted training to improve the model's ability to handle preference contradictions in a controllable and interpretable manner.

Category	Description & Example	Implication & Solution
Contextual Contradictions	Preferences change with context. *"I like sweets." → "I don't like sweets today, I'm sick."*	System must distinguish temporary context overrides from true preference changes. **Solution:** Timestamp + context tagging (e.g., mood, health).
Trade-off Contradictions	User holds preferences that conflict along a trade-off axis. *"Keep answers short." → "You left out details."*	Recognize trade-offs (brevity vs. informativeness) and adapt response balance. **Solution:** Model preferences as multi-dimensional goals.
Topic-Specific Preferences	Preferences vary by task/topic. *"Short answers for facts" vs. "Long answers for advice."*	Track preference scope by domain or topic. **Solution:** Topic-anchored preference keys.
Temporal Contradictions	Preferences evolve over time, not due to context. *"I like horror movies." → "They stress me out now."*	Prioritize recent preferences while preserving history. **Solution:** Recency decay + user confirmation.
Ambiguous Language or Intent	Preferences are expressed with hedging or uncertainty. *"I kind of like fast responses."* vs. *"Please take your time."*	Model uncertainty and detect hedged language. **Solution:** Certainty weights + hedging detection.
Meta-Preferences	User gives rules for how preferences should be applied. *"Keep things brief unless I ask for more."*	Enable higher-order logic for preference application. **Solution:** Meta-preference rule layer.

Table 1: Taxonomy of Contradictory Preference Types for DPO Data Generation

4.3 Direct Preference Optimization (DPO) Training

For the LoRA configuration, we set the dimensionality of the low-rank decomposition to 16. The lora_alpha is set to 32 to scale the output of the LoRA module and control the influence of the adaptation during training. A dropout rate of 0.05 is applied to the LoRA layers to reduce overfitting. The update of bias parameter is set to "none", meaning that biases in the affected layers are not modified. We define the task_type as "CAUSAL_LM" to specify that the model is fine-tuned for a causal language modeling task. Finally, we inject LoRA into selected transformer modules including "q_proj", "k_proj", "v_proj", and "o_proj" within the attention mechanism, and "gate_proj", "up_proj", and "down_proj" in the feedforward network. This selective adaptation enables efficient fine-tuning by modifying only a subset of the model's parameters while preserving the majority of the original weights.

For the DPO configuration, we set the temperature-like parameter $\beta = 0.1$. The maximum prompt length is set to 1024 tokens, with a total input (prompt+ chosen/rejected response) length up to 1512 tokens. Training is conducted for 2 epochs with a per-device batch size of 1, and gradients are accumulated over 4 steps to form one optimizer step. The learning rate is set to 5×10^{-5} with a cosine scheduler and a warm up ratio of 3%. The model is evaluated and saved every 10 optimizer steps, and only the best-performing model based on evaluation loss is retained. For precision, we use bfloat16 (bf16) and disable FP16.

5 Results

5.1 Evaluation Setup

We first evaluate the relative performance between the two Qwen1.5-1.8B models: one fine-tuned with Direct Preference Optimization (DPO) augmented with Chain-of-Thought (CoT) and one baseline model fine-tuned without CoT, based on intrinsic training and validation metrics. Specifically, we compare `train_loss`, `eval_loss`, `eval_rewards/accuracies`,

`eval_rewards/chosen_vs_rejected`, and `eval_rewards/margin` to compare learning curves and preference reward metrics between the two models.

Beyond intrinsic metrics, we evaluate the Qwen1.5-1.8B model fine-tuned with CoT-augmented DPO, and compare its outputs to those of a GPT-4o model, using both models' generated responses against the preferred responses annotated by Gemini-2.5-Flash on a separate evaluation dataset. For each dialogue context in our held-out test set, we prompt both the fine-tuned Qwen model and GPT-4o to generate a single agent reply.

To measure how well each generated reply aligns with the preferred response, we compute cosine similarity scores using a pretrained `all-mpnet-base-v2` inference model from the Sentence-Transformers library. We use this same inference model to embed both the Qwen model outputs and the GPT-4o outputs, ensuring consistency in the embedding space and reducing potential evaluation bias. We select `all-mpnet-base-v2` because it is a lightweight, general-purpose sentence embedding model with strong performance on semantic similarity tasks, providing consistent and robust embeddings across diverse dialogue domains.

For each dialogue, we compute the cosine similarity between the generated agent reply and the human-preferred reference response. Given embedding vectors $\mathbf{v}_{\text{gen}}$ for the generated reply (either Qwen or GPT-4o) and $\mathbf{v}_{\text{pref}}$ for the Gemini-annotated preferred response, we compute:

$$s_p = \cos(\mathbf{v}_{\text{gen}}, \mathbf{v}_{\text{pref}}) = \frac{\mathbf{v}_{\text{gen}} \cdot \mathbf{v}_{\text{pref}}}{\|\mathbf{v}_{\text{gen}}\| \, \|\mathbf{v}_{\text{pref}}\|}$$

We define a *match* whenever $s_p \geq 0.4$. The match rate is computed as:

$$\text{Match Rate} = \frac{\text{Number of Matches}}{\text{Total Number of Examples}}$$

We also report the average similarity across all examples:

$$\text{Average Similarity} = \frac{1}{N} \sum_{i=1}^{N} s_p^{(i)}$$

where N is the total number of examples and $s_p^{(i)}$ is the cosine similarity for example i.

This setup allows us to directly compare the alignment of GPT-4o and Qwen+CoT outputs to human-preferred responses under a unified evaluation framework, with consistent similarity computations across models to minimize evaluation bias.

5.2 Quantitative Analysis

Models trained with Chain-of-Thought (CoT) reasoning consistently outperform the baseline across every key metric. The CoT-augmented model reaches near-zero evaluation loss several steps earlier than the no-CoT variant, demonstrating markedly faster convergence. Likewise, its preference accuracy climbs to 100 percent by evaluation steps at around 25, whereas the baseline only achieves perfect accuracy around step 60. Throughout training, the CoT model maintains a larger and more stable gap between "chosen" and "rejected" rewards, reflecting higher confidence in its rankings. The baseline model's reward for the "chosen" answers starts small, rises to around $+1.2$ by step 30, then gradually falls to about -0.8 by step 200. Over the same steps, its reward for the "rejected" answers drops steadily from -0.2 down to -10.8. By contrast, the CoT model begins with a higher chosen-answer reward, climbs above $+2.3$ around step 50, and then levels off near $+1.6$. Its rejected-answer reward also falls from $+0.08$ to about -10.1, but the gap between chosen and rejected stays much larger at every checkpoint. This wider gap shows the CoT model is more confident at picking the preferred response. As a result, its reward margins converges at approximately 12, compared to around 10 for the baseline, indicating a more decisive separation between preferred and dispreferred responses.

As shown in Table 2, our Qwen1.5-1.8B model fine-tuned with DPO and CoT reasoning outperforms the GPT-4o model on both match rate and average similarity. The Qwen DPO+CoT model achieves a

higher match rate of 0.432 than GPT's 0.338, indicating that it more consistently generates responses that align with the user's true preference. In addition, it has a higher average similarity of 0.362 than GPT's 0.338, suggesting improved semantic alignment with reference responses. These results demonstrate that incorporating CoT reasoning during fine-tuning enhances the model's ability to match human preferences and generate responses with greater semantic consistency.

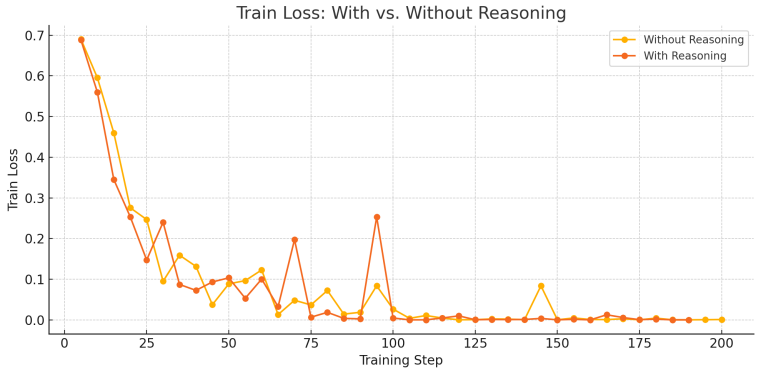

Figure 2: Training loss over time.

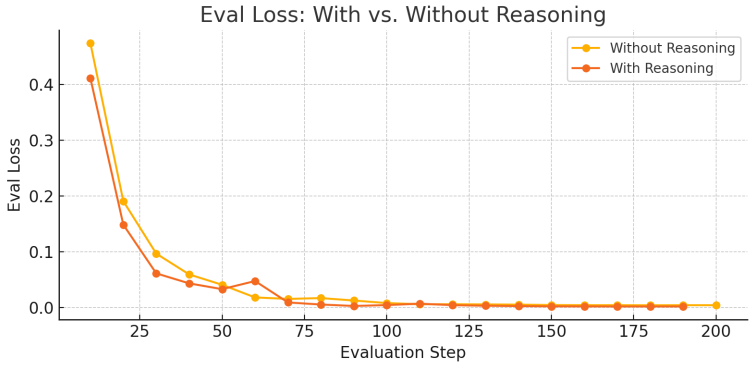

Figure 3: Evaluation loss over time.

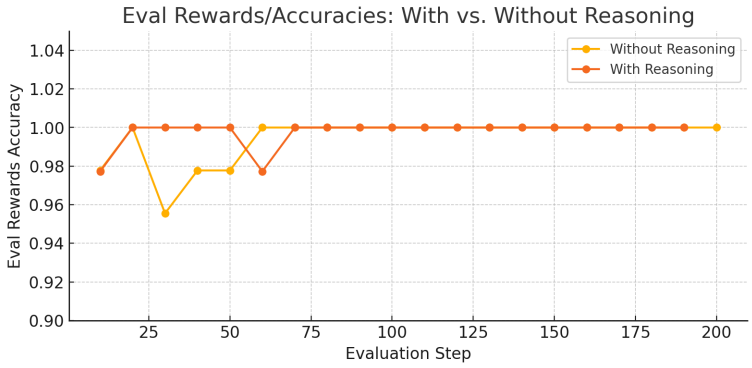

Figure 4: Evaluation reward accuracies.

5.3 Qualitative Analysis

In addition to quantitative metrics, we qualitatively examined examples of generated responses from both GPT-4o and our Qwen1.5-1.8B DPO-finetuned with CoT model. We observed that Qwen+CoT

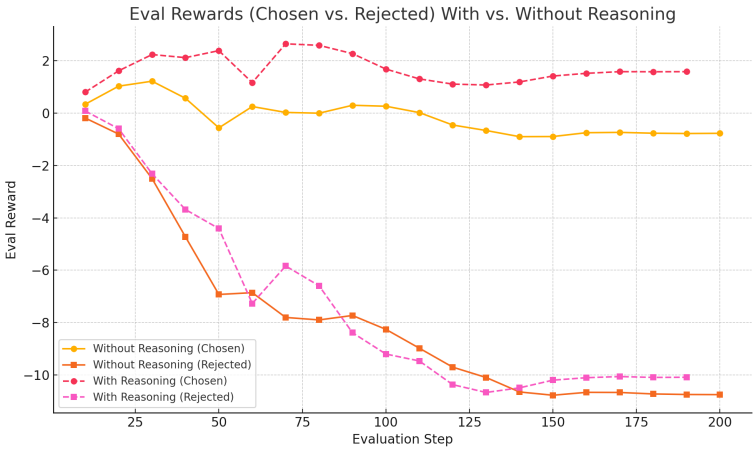

Figure 5: Evaluation rewards for chosen vs. rejected answers.

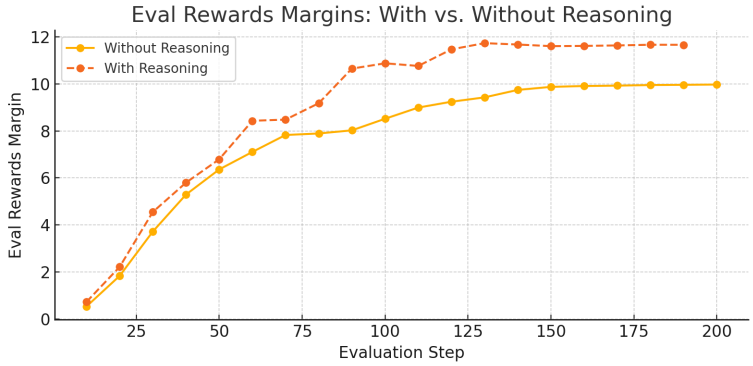

Figure 6: Evaluation reward margin.

Model	Match Rate	Avg Similarity
GPT-4o	0.338	0.330
Qwen1.5-1.8B DPO-finetuned + CoT	0.432	0.362

Table 2: Comparison of Match Rate and Avg Similarity for GPT-4o and Qwen DPO+CoT.

responses often demonstrated stronger alignment with the annotated preferences, particularly in cases involving ambiguous or subtly expressed user intent.

For instance, in one dialogue, the user initially expressed a clear preference for studying alone. Later in the conversation, the agent suggested trying a group study session; the user acknowledged this suggestion as acceptable because the agent framed it positively, but implicitly maintained a preference for studying alone. GPT-4o's reply incorrectly interpreted the user's acceptance of the suggestion as a shift in their true preference and reinforced the group study direction, whereas the Qwen+CoT model correctly identified that the user's underlying preference remained solo study and generated a response that respected this.

More broadly, we found that CoT reasoning contributed to responses that more accurately captured subtle preference signals and provided better interpretability. In scenarios involving ambiguous expressions of intent or meta-preferences, such as when users expressed preferences about how the agent should respond, Qwen+CoT was more likely to incorporate such preferences explicitly and appropriately into its replies. These qualitative trends complement the observed improvements in Match Rate and Average Similarity, suggesting that CoT reasoning enhances not only quantitative alignment metrics but also the fidelity and interpretability of model outputs with respect to user preferences.

6 Discussion

Our experiments demonstrate that incorporating Chain-of-Thought (CoT) prompting into both the data generation and fine-tuning stages improves Qwen's ability to model user preference changes. Our CoT-augmented pipeline leads to faster convergence, higher preference accuracy, and stronger reward separation between chosen and rejected responses compared to the Qwen model fine-tuning without CoT. In addition, the CoT-augmented Qwen model achieves better semantic alignment with preferred responses than GPT-4o under the same evaluation setup.

A key limitation of our current study is that each dialogue contains only a single annotated contradiction, which constrains the model's ability to learn from and reason about more complex, naturally occurring preference shifts. Looking ahead, several extensions to this work are promising. First, we plan to broaden the scope of contradiction scenarios. While our current data includes multi-turn dialogues, future data generation could introduce more complex patterns of conflicting preferences, such as multiple simultaneous or overlapping preference contradictions within a dialogue, nested contradictions, and longer context windows that require tracking evolving user preferences across many turns. These more challenging settings would better test the robustness of our CoT-augmented pipeline in handling dynamic and realistic dialogue interactions.

Second, we aim to incorporate human-in-the-loop evaluation to better assess model alignment with real user intent. One approach is to run live pairwise preference trials in which human raters compare responses from different models. These comparisons can be aggregated into ELO ratings to provide a scalable, interpretable benchmark for tracking fine-tuning progress across configurations.

Finally, we are interested in exploring adaptive CoT prompting. In our current setup, CoT examples are manually constructed and statically inserted into DPO prompts. In future work, we will investigate methods for enabling the model to detect potential preference contradictions at inference time and dynamically generate its own reasoning chains. Such adaptive prompting could reduce reliance on human-annotated examples and further enhance the model's ability to handle dynamic user preferences in open-ended settings.

7 Conclusion

Our project explores the challenge of enabling Large Language Models (LLMs) to detect and manage contradicting user preferences, with a focus on contradictions that arise within multi-turn user-LLM dialogues. Enhancing an LLM's capacity to identify and respond appropriately to such contradictions is important for building conversational agents that remain coherent, context-aware, and aligned with the user's true preferences. This capability is a key step toward more trustworthy and adaptive dialogue systems that can better support complex, evolving user preferences in real-world applications.

Our methodology combines several key contributions. We define a typology of six distinct preference contradiction types: contextual, trade-off, topic-specific, temporal, ambiguous intent, and meta-preference, providing a structured framework for understanding dynamic user preferences. To address the lack of suitable public datasets, we develop a persona-driven, two-pass LLM pipeline for synthetic data generation. Chain-of-Thought (CoT) prompting guides both contradiction annotation and response generation. We fine-tune the Qwen1.5-1.8B model using Direct Preference Optimization (DPO), with carefully designed training and evaluation prompts. Our evaluation setup includes both intrinsic metrics: training loss, preference accuracy, reward margin, and external comparison to GPT-4o using cosine similarity and match rate between agents' responses and preferred responses.

Our experimental results show that CoT-augmented DPO fine-tuning improves the model's ability to handle preference changes. The CoT-trained Qwen model converges faster, achieves higher preference accuracy, and maintains a larger reward margin than the baseline model trained without CoT. In external evaluation, it also outperforms GPT-4o on both match rate and semantic similarity to preferred responses on the held-out test set. These results suggest that explicit reasoning signals can help the model better generalize about why certain responses align with updated user preferences.

This study has several limitations. Our current data generation focuses on single instances of preference contradiction per conversation. Future work should explore more complex cases involving multiple overlapping or nested contradictions and longer context dependencies. In addition, our

evaluation relies on automated similarity-based metrics. Incorporating human-in-the-loop evaluation will be helpful to fully assess real-world model alignment with user preferences.

Personalizing LLMs to rapidly evolving user preferences remains a critical challenge. Future directions include expanding the scope of contradiction scenarios, developing adaptive CoT prompting techniques to enable dynamic reasoning at inference time, and conducting user studies to evaluate the impact of contradiction-aware LLMs on conversational experience.

8 Team Contributions

- **Eugenie Shi:** Designed and implemented the reasoning-augmented pipeline, crafted the dataset generation prompt augmented with preference-contradiction types, and the CoT prompts for dataset annotation and fine-tuning Qwen1.5, built the evaluation framework, analyzed the DPO fine-tuned models' performances, and communicated next steps with the project mentor.
- **Haorui Guo:** Designed and implemented the persona-driven, synthetic data generation pipeline, including research, methodology, processes, and final output. Generated all training and evaluation data. Helped implement the DPO training pipeline by debugging training errors and ensuring successful E2E running of the entire training pipeline.
- **Shuojia Fu:** Designed and implemented DPO training pipeline, including chosen-rejected data curation, hyperparameter tuning, model fine-tuning, and model training. Conducted analysis of DPO model performance. Assisted in debugging response data formatting and pre-processing issues.

Changes from Proposal We discarded the original rule-based preference tracker for direct/indirect types of contradiction, and used a richer, content-based taxonomy of contradiction types during dataset generation. We also moved to a two-LLM pipeline, using one model to generate dialogues and another to annotate them.

References

Tao Ge, Xin Chan, Xiaoyang Wang, Dian Yu, Haitao Mi, and Dong Yu. 2025. Scaling Synthetic Data Creation with 1,000,000,000 Personas. arXiv:2406.20094 [cs.CL] https://arxiv.org/abs/2406.20094

Google DeepMind. 2025. Gemini 2.5 Flash. https://cloud.google.com/vertex-ai/generative-ai/docs/models/gemini/2-5-flash. Accessed: 2025-06-07.

Xinyu Li, Ruiyang Zhou, Zachary C. Lipton, and Liu Leqi. 2024. Personalized Language Modeling from Personalized Human Feedback. arXiv:2402.05133 [cs.CL] https://arxiv.org/abs/2402.05133

OpenAI. 2024. GPT-4o System Card. arXiv:2410.21276 [cs.CL] https://arxiv.org/abs/2410.21276

Long Ouyang, Jeff Wu, Xu Jiang, Diogo Almeida, Carroll L. Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul Christiano, Jan Leike, and Ryan Lowe. 2022. Training language models to follow instructions with human feedback. arXiv:2203.02155 [cs.CL] https://arxiv.org/abs/2203.02155

Sriyash Poddar, Yanming Wan, Hamish Ivison, Abhishek Gupta, and Natasha Jaques. 2024. Personalizing Reinforcement Learning from Human Feedback with Variational Preference Learning. arXiv:2408.10075 [cs.LG] https://arxiv.org/abs/2408.10075

Lior Shani, Aviv Rosenberg, Asaf Cassel, Oran Lang, Daniele Calandriello, Avital Zipori, Hila Noga, Orgad Keller, Bilal Piot, Idan Szpektor, Avinatan Hassidim, Yossi Matias, and Rémi Munos. 2024. Multi-turn Reinforcement Learning from Preference Human Feedback. arXiv:2405.14655 [cs.LG] https://arxiv.org/abs/2405.14655

Siyan Zhao, Mingyi Hong, Yang Liu, Devamanyu Hazarika, and Kaixiang Lin. 2025. Do LLMs Recognize Your Preferences? Evaluating Personalized Preference Following in LLMs. arXiv:2502.09597 [cs.LG] https://arxiv.org/abs/2502.09597

A Dataset Example


```
"Contradiction": {
    "curr_pref": "Wants balanced, flexible plan",
    "curr_turn_id": 16,
    "prev_pref": "Usually prefers detailed plans",
    "prev_turn_id": 12,
    "type": "temporal"
},
"Response": {
    "preferred_response": "That's a fantastic question that perfectly aligns
        with finding that balance! For some unique local artists or specific
        workshops, a bit of advance research and a polite message might open
        doors to truly special experiences. However, definitely leave ample
        space to just 'see what happens' when you get there \u2013 many of the
        most authentic connections arise spontaneously from exploring the local
        scene. Would you like me to help you find some online resources to
        start that initial research?",
    "dispreferred_response": "Given your previous preference for ensuring you
        'see everything' and your desire for 'rich detail' in your itinerary,
        I'd strongly recommend reaching out to local communities and artists
        well in advance. We can start drafting a detailed schedule for those
        meetings right away to ensure nothing is missed."
}
```


Figure 7: Annotation of Preferred vs. Dispreferred Responses


```
{
    "turn_id": 12,
    "speaker": "user",
    "text": "You\u2019re right. I usually prefer a detailed plan, but this
        time around, maybe I should just outline a couple of key events and
        go from there\u2014like finding some cool local artists to meet up
        with or workshops. But I don\u2019t want to lose that rich detail
        in my itinerary..."
```


Figure 8: Previous Preference


```
{
    "turn_id": 16,
    "speaker": "user",
    "text": "That sounds like the perfect balance! I really want to tap
        into that creativity, but without stressing over every detail. Can
        you also find some info on unique local artists in those areas?"
```


Figure 9: Current Preference