

Building A Contextual Reasoning Aware Social-Intel Agent with Reinforcement Learning

Stanford CS224N Custom Project

Ying Lu, Binbin Li, Da Sun

Department of Computer Science

Stanford University

yinglu@stanford.edu, binbinli@stanford.edu, dasun@stanford.edu

Abstract

We propose a novel Contextual Reasoning Aware reinforcement learning framework for training an agent that explicitly reasons about the client’s state and strategy, and learns to respond to the client’s messages based on conversation history and available tools. The agent builds rapport, addresses client concerns, and facilitates booking decisions when appropriate. We build a simulated self-play patient-agent interactive environment and generate multi-turn consultation episodes featuring structured chain-of-thought reasoning and tool use, using Gemini-2.5-Flash and Sotopia’s framework. We first teach the assistant model to follow this expert thought process via Supervised Fine-Tuning (SFT). In order to evaluate the agent during reinforcement learning, we train a custom 1.5B reasoning reward model via SFT and Group Relative Policy Optimization (GRPO), utilizing a paired consistency reward on augmented data to mitigate position bias. During the assistant’s GRPO training, we use this custom judge and gpt-4o judge with golden-anchored Bradley-Terry reward function within GRPO to guide the agent to reason, use tools and respond. We evaluate the agent on a held-out test set of 500 prompts using multiple LLM judges. After correcting for position and length bias, we find that the GRPO-trained agent achieves an unbiased BT score of +0.516 relative to the expert baseline (0.0), outperforming the 3B-SFT model (−0.036), the 3B base model (−0.840) and even the larger 7B-SFT model (−0.195). The improvement is most pronounced on challenging patient scenarios (+0.880), showing that our RL-trained model learns adaptive strategies for complex scenarios.

Key Information to include

- Mentor: Fang Wu
- External Collaborators: (None)
- Late Days (if you are using any):

Team member	Late days remaining
Ying Lu	2
Binbin Li	2
Da Sun	2
Total pooled	6
Team size	3
Late days applied for the report	2

1 Introduction

Deploying AI agents in customer-facing and high-stake dialogue settings, such as aesthetic medical consultations, poses unique challenges. The agent must genuinely understand the client’s emotional and cognitive state, reason through strategies, and guide the conversation toward recommendations that put the client’s interest first. We define this capability as contextual reasoning: the structured decomposition of the consultant’s thought process into client state assessment (Client), strategy reflection (Reflect), booking decision (Book), and action planning (Plan). These multi-dimensional objectives make it difficult to define a single scalar reward, and the multi-turn nature of consultations demands long-horizon reasoning.

Stanford CS224N Natural Language Processing with Deep Learning

Existing approaches typically rely on supervised fine-tuning (SFT) on demonstration data or reinforcement learning from human feedback (RLHF) with point-wise reward models [1]. However, SFT maximizes the likelihood of the average expert strategy and is unable to discover responses that surpass expert quality. Moreover, point-wise reward models struggle to capture nuanced quality differences in multi-turn, high-stake conversations, where pairwise comparison is more natural and relatively easy to judge.

We address these limitations with a three-stage approach:

Stage 1 (Data Generation): We generate 3,851 multi-turn consultation episodes using the Sotopia self-play simulation framework [2] with 90 diverse patient profiles across 40 scenarios. Each agent turn includes a structured reasoning block (Client/Reflect/Book/Plan) followed by the consultant’s response, which together serve as the expert golden reference.

Stage 2 (SFT): We fine-tune Qwen2.5-3B-Instruct on this data via QLoRA [3] to teach the model the structured reasoning format and domain knowledge.

Concurrently, we fine-tune a 1.5B reward model on reasoning distillation data, teaching it to evaluate responses by generating its own structured thought process before rendering a verdict.

Stage 3 (RL via GRPO): We execute reinforcement learning in two sequential phases to ensure high-quality evaluation and policy updates:

- **Phase A (Reward Model RLVR):** We optimize the 1.5B reward model via GRPO using a novel paired consistency reward on position-swapped augmented data. This forces the model to develop position-invariant reasoning, effectively mitigating the position bias which is common in LLM-as-a-judge systems.
- **Phase B (Assistant Model GRPO RL):** We apply GRPO [4] with golden-anchored Bradley-Terry reward function to the 3B assistant model.

For each prompt, we generate multiple rollouts and pair them along with the golden response for pairwise judging by gpt-4o and our trained reasoning reward model (1.5B-RW-R1-GRPO-Paired). We randomize presentation order to mitigate position bias, fit a Bradley-Terry model to the pairwise outcomes, and anchor the expert response score to zero, which produces rewards calibrated relative to expert quality. We then follow with GRPO training for policy update.

We evaluate on 500 held-out test prompts from 353 eval episodes with three LLM judges (gpt-4o, Claude 3.5 Sonnet, Gemini 2.0 Flash) and correct for position and length bias. The GRPO-trained model achieves a debiased BT score of +0.516 over the expert baseline, outperforming all other models including 3B-SFT (−0.036) and 7B-SFT (−0.195). Task-specific evaluation shows the largest gains in strategic reasoning dimensions: booking decision quality, plan quality, and reflect quality, suggesting GRPO primarily sharpens the model’s internal reasoning. The improvement is most pronounced on challenging patients (+0.880 vs. +0.595/+0.504), where GRPO’s per-prompt rollout comparison creates sharper learning signals due to higher rollout variance.

Our key contributions are as follows:

- (a) A self-play data generation pipeline with a two-phase architecture: structured contextual reasoning block followed by response generation.
 - (b) A custom 1.5B reasoning reward model trained via SFT and GRPO utilizing a paired consistency reward on position-swapped augmented data to mitigate position bias.
 - (c) A novel golden-anchored Bradley-Terry reward for GRPO that estimates latent response quality (reasoning and generation) relative to the golden response via pairwise comparison, maintaining a calibrated quality reference throughout training.
 - (d) A comprehensive multi-judge evaluation framework with position and length bias debiasing, demonstrating that GRPO training yields substantial improvements over both SFT and base models.
- The remainder of the paper is organized as follows: Section 2 reviews related work, Section 3 details approach, Section 4 presents experiment details. Section 5 discusses analysis, and Section 6 discusses the conclusion and future work.

2 Related Work

Social Simulation for Training Data. Sotopia [2] provides a framework for generating and evaluating social interactions between AI agents. Self-play simulation offers lower cost, scalability, and fine-grained control over patient diversity and scenario design. We generate domain-specific consultation episodes with structured reasoning beyond Sotopia’s multi-agent simulation.

Chain-of-Thought and Structured Reasoning. Chain-of-thought prompting [5] has shown that eliciting intermediate reasoning steps improves LLM performance on complex tasks. Our structured reasoning block (Client/Reflect/Book/Plan) extends this idea to conversational response: rather than

free-form chain-of-thought, we impose a domain-specific structural reasoning format, which provides interpretability and enables targeted evaluation of each reasoning component.

Reasoning Reward Models. DeepSeek-R1 [6] demonstrates that reinforcement learning can improve reasoning quality in language models. The RM-R1 framework [7] extends this insight to reward modeling: rather than assigning scalar scores, RM-R1 trains reward models via Reinforcement Learning with Verifiable Rewards (RLVR) to produce chain-of-thought reasoning before issuing a pairwise verdict. We run experiments with both our trained 1.5B reasoning reward model and gpt-4o as pairwise judges during GRPO training.

Reinforcement Learning for Dialogue. RLHF has been widely applied to align language models with human preferences [1, 8]. Recent work explores alternatives to pointwise reward models: Direct Preference Optimization (DPO) [9] optimizes preferences directly, while GRPO [4] eliminates the critic network by using group-level relative advantages computed from multiple rollouts. iGRPO [10] furthers this idea by applying iterative rounds of rollout generation and policy improvement. Our work extends to multi-turn dialogue with a Bradley-Terry pairwise reward anchored on golden responses.

Pairwise Evaluation and LLM-as-Judge. The Bradley-Terry model [11] has been widely adopted for pairwise preferences, notably in Chatbot Arena for ranking LLMs from human judgments. LLM-as-judge approach [12] automates pairwise comparison using strong models, but exhibits systematic biases including position bias and length bias. We mitigate these biases through presentation order randomization during training and joint estimation of position and length bias parameters during evaluation. Beyond using BT for evaluation, we anchor the rollout response scores on the golden response’s BT score to produce relative and calibrated scores to be used in GRPO.

3 Approach

Figure 1 illustrates our Contextual Reasoning Aware framework, which consists of a three-stage framework designed to train a strategic assistant and a robust judge. Stage 1 leverages Sotopia [2] to generate expert demonstration data. Stage 2 performs Supervised Fine-Tuning (SFT) to initialize both the 3B assistant model and the 1.5B reward model in parallel. Finally, Stage 3 applies Group Relative Policy Optimization (GRPO) in two phases: first the 1.5B reward model via a paired consistency penalty, and then optimizing the 3B assistant using a golden-anchored Bradley-Terry reward computed by our customized judge and GPT-4o. The details of the reward and assistant models are in Sections 3.1 and 3.2, respectively.

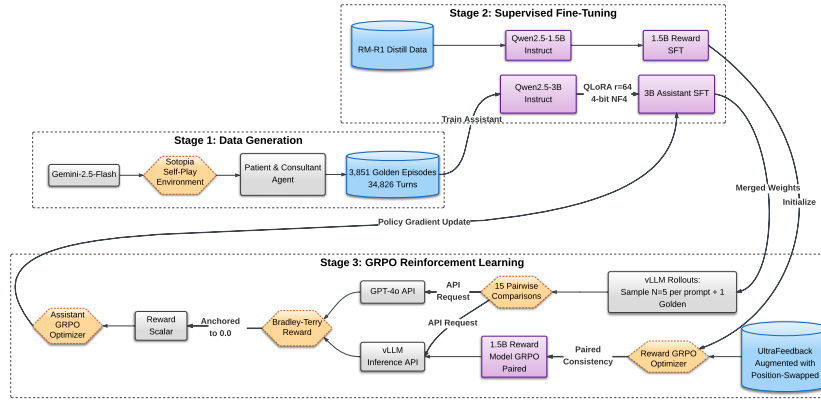


Figure 1: Overview of our Contextual Reasoning Aware framework, detailing the progression from synthetic data generation (Stage 1), through parallel supervised fine-tuning (Stage 2), to sequential GRPO reinforcement learning (Stage 3).

3.1 Reward Model

To provide the pairwise judgments required for the Assistant’s GRPO training, we trained a customized Reasoning Reward Model through domain adaptation and debiasing.

Reward with Reasoning: We adopt the RM-R1 strategy [7], equipping reward models with reasoning capabilities by utilizing preference modeling as a structured reasoning task rather than a scalar prediction. A Reasoning Reward Model is trained to first generate a detailed reasoning chain-of-rubric before deciding which of two responses is superior and outputting a preference. This shift

enhances the interpretability of the model’s choices and improves its accuracy by forcing it to justify its evaluations through a coherent logical framework.

Initial Approach (Reasoning Distillation & RLVR): We initially implemented this pipeline using a two-stage process inspired by RM-R1 [7]. First, a Reasoning Distillation stage (SFT) used high-quality chains synthesized through a two-pass bootstrapping method (leveraging models such as Claude-3.7 and OpenAI o3 to correct flawed logic). Second, the model underwent Reinforcement Learning with Verifiable Rewards (RLVR) to optimize its reasoning performance. In parallel, we hypothesized that a model already pre-trained on reasoning traces could bypass the SFT stage and have a good performance, so we apply RLVR directly to DeepSeek-R1-Distill-Qwen-1.5B.

Domain Mismatch and Position Bias: While these initial models successfully generated reasoning traces, we observed two critical failures during evaluation. First, because the baseline datasets heavily focused on math and coding, the models struggled to evaluate the nuanced "soft skills" of our consultant application, such as empathy, linguistic precision, and professional tone. Second, these models suffer from severe position bias which favors whichever response appears first regardless of content quality. We also found that the GRPO amplifies this bias because it calculates advantages purely within a group, lacking an absolute baseline.

Data Augmentation and Paired Consistency Reward Strategy: To mitigate position bias, we augment our dataset by creating position-swapped pairs. For each training example, we generate a mirror copy with reversed A/B orderings and store them as adjacent pairs. During GRPO training, the model simultaneously evaluates both orderings of the same example within each batch. To address the domain mismatch, we also replace the training data with our UltraFeedback Binarized dataset, a more diverse dataset containing human and AI preferences across multiple dimensions.

To enforce position-invariant judgment, we design a *paired consistency reward* that evaluates the model’s decisions across both orderings jointly, rather than rewarding isolated correct or wrong:

- **Both correct:** Position-invariant reasoning; strong positive reward (e.g., +1).
- **One correct:** Position-biased shortcut; moderate penalty (e.g., -0.5).
- **Both wrong:** Fundamental evaluation failure; heavy penalty (e.g., -1 or -2).

3.2 Assistant Model

Data Generation: We use the Sotopia framework [2] to simulate multi-turn consultations. Our simulation pairs a single consultant agent (Sophia Chen) with diverse patient profiles (profession, demographics, personality, knowledge level) drawn from a pool of 90 agents across 40 consultation scenarios, yielding a combinatorial space of 3,600 possible episodes. All patient profiles and scenarios are generated using Gemini-2.5-Flash.

Patient Agents: We designed three categories of patient profiles: a) medical specific (30 agents): Diverse demographics (ages from 24 to 70) with realistic cosmetic concerns, varying knowledge levels, and decision-making styles. b) Challenging (20 agents): Angry confrontational patients, indecisive patients, patients with language barriers, minors accompanied by parents, patients with body dysmorphic concerns, etc. c) General Sotopia agent profiles (40 agents): Standard social agents from the Sotopia framework for diversity.

Two-Phase Generation: This two-phase architecture decouples reasoning from execution, allowing the model to think before acting (see Appendix A.1 for the exact structured XML output format).

Tool System: For simplicity, the consultant has access to four tools: (1) `get_client_profile` for predefined patient’s demographics, (2) `get_knowledge_base` for medical facts from a curated knowledge base, (3) `get_personal_experience` for real community discussions and anecdotes, and (4) `get_calendar` for appointment availability.

Dataset Statistics: We generated 3,851 episodes (mean 10.8 turns), yielding 3,498 training and 353 test episodes. For RL training, each user turn is extracted as a separate prompt with full conversation history and previous agent thinking and response, producing roughly 18,600 RL training prompts.

Supervised Finetuning: We fine-tune Qwen2.5-3B-Instruct [13] using QLoRA [3] with 4-bit NF4 quantization and Flash Attention 2. For hyperparameters, we use LoRA rank 64, alpha 128, targeting all linear layers, learning rate 1×10^{-4} with cosine annealing and 3% warmup; batch size 8 (micro-batch 1, gradient accumulation $8 \times$); 5 epochs; max sequence length 4,096 tokens in bfloat16. This ensures we fit the entire training on a single GPU. In addition, in order to achieve format compliance, we add four special tokens: `<think>`, `</think>`, `<response>`, `</response>`, which are trainable. We apply label masking to ensure that only assistant turns contribute to token-level cross-entropy loss. We train from 3498 episodes consisting of 34,826 assistant turns.

Golden-Anchored Bradley-Terry Reward Score: For each training prompt, we generate $N=5$ candidate responses using vLLM [14] with our trained SFT model and retrieve the corresponding expert golden response from the SFT data, yielding $N+1=6$ candidates. We compute all $\binom{6}{2}=15$ pairwise comparisons using gpt-4o and our trained reasoning reward model, with 50% presentation order randomization to mitigate position bias. We then fit the Bradley-Terry model [11] via iterative Minorization-Maximization (MM): $\log(\frac{p_{ij}}{1-p_{ij}}) = \beta_i - \beta_j$, where β_i and β_j are the quality score of rollout i and j . We skip the pair results for ties. We get a matrix of $N \times N$, and we add 0.5 for smoothing to every cell. We fit the Bradley-Terry model iteratively. After convergence, we anchor the golden response to zero: $bt_i = \log p_i - \log p_{\text{golden}}$. Positive BT scores indicate the rollout exceeds golden response; negative scores mean the quality is below the golden response. This anchoring ensures a calibrated quality reference throughout training.

Final Reward Score: In addition to the BT score, we add a format reward ($w_{\text{fmt}}=1.0$ vs. BT weight $w_{\text{bt}}=2.0$) with 4 binary checks at 0.25 points each: (1) <think> block exists, (2) all required fields present, (3) <response> block exists, (4) correct ordering. The final reward per rollout is: $r = 2.0 \cdot s_{\text{BT}} + 1.0 \cdot s_{\text{format}}$.

GRPO training: After SFT, we merge the LoRA adapter into the base model and apply GRPO [4] for RL training. We train a new LoRA adapter (rank 64, alpha 64, all-linear targets) at learning rate 5×10^{-6} with low-variance KL loss ($\beta=0.001$), max prompt length is 4096, and max response length is 512. Rollout generation uses vLLM with GPU memory utilization 0.45 on a single NVIDIA RTX 5090 (32 GB). Gradient checkpointing, optimizer offload to CPU, and conservative vLLM allocation enable single-GPU GRPO training on a single 5090 GPU. For reward scoring, we leverage reasoning reward model API and gpt-4o, so no extra memory cost is involved. Finally, we randomly sample 2,000 training data, and apply the GRPO training with batch size of 8 and 250 steps and run for 1 epoch. We save checkpoints every 50 steps.

Implementation: We build and extend the data generation pipeline beyond Sotopia framework [2], perform SFT training, leverage vllm server for rollouts [14], reward model building and integration based on [7], GRPO training loop with verl, and evaluation framework.

4 Experiments

4.1 Reward Model

Data & Model Configurations: Our reward model training utilized multiple datasets to isolate formatting, logic, and conversational alignment, including RM-R1-Distill-SFT, RM-R1-Reasoning-RLVR, and a domain-filtered version of UltraFeedback Binarized with pairwise augmented data. We evaluated four custom 1.5B variants and one custom 7B variant against the official 7B RM-R1 baseline to isolate the optimal judge. Detailed dataset statistics, model architecture ablations, and hyperparameter configurations (Table 5) are provided in Appendix A.2.

Data: We leveraged the RM-R1-Distill-SFT dataset (8,753 examples), which contains chain-of-thought reasoning traces distilled from DeepSeek-R1, to train the base model with pairwise evaluation capabilities via supervised fine-tuning. The model learns to analyze two candidate responses step-by-step in a <think> block before producing a binary verdict. We then used verifiable binary UltraFeedback data (60,924 pairs, position-augmented to 121,848 examples) to continue training the model with paired consistency reinforcement learning (GRPO), which reduces position bias by requiring the model to produce consistent judgments across augmented batches.

Evaluation & Metrics: We evaluate on two datasets to rigorously assess the reward models: UltraFeedback-Paired-Eval (1,992 pairs unseen conversational pairs split of UltraFeedback) and RM-Bench (1,327 pairs) using a two-pass inference (prompting with both original and swapped candidate orderings). We report Position Bias, Consistency, Debaised Accuracy, Total Accuracy, and Format Adherence. The exact math formulations for these metrics are provided in Appendix A.3.

Experimental Details: We train the 1.5B Reward Model by applying our novel paired consistency reward across augmented batches. Comprehensive baseline model setups, ablation architectures, and full hyperparameter configurations are detailed in Appendix A.2.

Results: We evaluate the reward models across in-domain conversational data (UltraFeedback) and generalized tasks (RM-Bench).

The Position Bias Crisis in Small-Parameter Baselines: Table 1 shows that the standard RLVR baseline for the 1.5B model (1.5B Reasoning and 1.5B GeneralJudge) presents severe position bias, with its A-selection rate reaching 74.3% and 62.5% on UltraFeedback. Consequently, its

debiased accuracy stays low at 28.8% and 36.1%, revealing that small-parameter models easily hack positional guessing during GRPO without structured rewards. In contrast, the larger 7B RM-R1 baseline has enough parameter capacity to resist this reward hacking, maintaining a well-balanced 55.2% position bias and 57.6% debiased accuracy.

Ablation on Training Strategy & Reward Shaping: Comparing the intermediate 1.5B variants isolates the impact of our pipeline choices in Table 1. The 1.5B General-Judge (trained via direct RLVR, skipped SFT) improved slightly over the baseline but still suffered from high position bias (62.5%). Applying the paired data augmentation with a strict penalty structure (1.5B Paired-Collator using +1/0/-1/-2) successfully lowered position bias to 54.3% but collapsed in consistency (54.6%) and debiased accuracy (33.7%). This demonstrates two critical insights: (1) it is necessary to have SFT initialization for the model’s stable structural reasoning before RL, and (2) the penalty for inconsistency cannot be too harsh, or the model will stop learning.

Efficacy of the Final Paired Consistency Reward: Our 1.5B Paired model combines SFT initialization with a balanced penalty (+1/-1/-0.5) and achieves the optimal performance. On the UltraFeedback test set in Table 1, position bias is corrected to a near-ideal 51.7%, and consistency is as high as 78.5%. Crucially, the strict debiased accuracy spikes to 57.8%, which doubles the 1.5B Reasoning baseline (28.8%) and effectively matches the performance of the much larger 7B RM-R1 baseline. This confirms our hypothesis: the paired penalty helps a small model to achieve 7B-level robustness with roughly 5x fewer parameters.

Model	Pos Bias	Consist	Debiased Acc	Total Acc	Fmt
1.5B Reasoning	74.3%	42.7%	28.8%	57.30%	86%
1.5B Paired	51.7%	78.5%	57.8%	68.75%	99%
1.5B PairedCollator	54.3%	54.6%	33.7%	56.30%	99%
1.5B GeneralJudge	62.5%	55.2%	36.1%	58.55%	99%
7B Reasoning	50.9%	71.2%	52.3%	66.50%	99%
7B RM-R1	55.2%	73.4%	57.6%	71.00%	100%

Table 1: Model Performance Evaluation (UltraFeedback Paired Eval)

Generalization (RM-Bench): On the multi-domain RM-Bench test set in Table 2, the 1.5B Paired model maintains its debiasing performance, significantly reducing position bias compared to the 1.5B baseline (from 65.9% to 55.2%) and improving consistency. However, total accuracy gains are not optimal in this generalization setting. This suggests that while the paired penalty successfully addressed position-bias, evaluating complex domains (like math and coding) remains fundamentally bounded by the 1.5B parameter capacity.

Model	Pos Bias	Consist	Debiased Acc	Total Acc	Fmt
1.5B Reasoning	65.9%	49.4%	30.8%	55.8%	88%
1.5B Paired	55.2%	56.3%	32.4%	54.4%	95%
1.5B PairedCollator	52.4%	49.2%	31.0%	56.3%	93%
1.5B GeneralJudge	53.4%	54.7%	36.6%	59.2%	93%
7B Reasoning	48.7%	72.7%	56.2%	69.6%	92%
7B RM-R1	66.2%	59.1%	40.6%	61.0%	99%

Table 2: Model Performance Evaluation (RM-Bench)

Training Dynamics: Tracking the GRPO optimization of the 1.5B Paired model (see Appendix A.6, Figure 2) reveals a two-phase learning trajectory with bootstrap CIs for Position Bias and Accuracy.

4.2 Assistant Model

Data: Our dataset consists of 3,851 multi-turn consultation episodes (3,498 train / 353 test). The task is single-turn response generation conditioned on multi-turn conversation context: given the system prompt, conversation history, and client’s message, the model must produce a structured thinking block followed by an appropriate response to the client.

Evaluation method: We evaluate on 500 prompts drawn from 353 held-out episodes, covering all three patient categories. We compare 7 model variants: 3B-base (Qwen2.5-3B-Instruct), 3B-SFT, 3B-GRPO-RM-reward, 3B-GRPO-gpt-4o-reward, 7B-base (Qwen2.5-7B-Instruct), 7B-SFT, and golden response. We use three LLM judges on the evaluation dataset: gpt-4o, Claude 3.5 Sonnet, and Gemini 2.0 Flash. We extend the standard BT model with a position bias parameter δ and length bias parameter γ per judge. Additionally, we measure format compliance rate, and ask gpt-4o to provide

task-specific scores from 1 to 10 on 6 dimensions: trust and comfort, information quality, booking intention, reflection quality, booking decision quality, and plan quality.

Experimental details: For comprehensive SFT and GRPO hyperparameter configurations (including LoRA ranks, learning rates, and KL coefficients), refer to Appendix A.4 (Table 6).

Results on BT score: Table 3 below shows the position and length debiased BT scores across all models and judges. The GRPO trained model with gpt-4o as reward achieves a substantial advantage of +0.516 aggregated from multiple judges over the expert golden baseline, while SFT and GRPO model with our trained reward reasoning model hover near zero. The base model scores worse. This shows that with RL and the right reward, the quality of the model improves.

Model	gpt-4o	Claude	Gemini	Aggregate
3B-GRPO-gpt-4o-reward	+0.793	+0.469	+0.501	+0.516
Golden (expert)	0.000	0.000	0.000	0.000
3B-SFT	+0.039	-0.044	-0.063	-0.036
3B-GRPO-RM-reward	+0.030	-0.054	-0.068	-0.051
7B-SFT	-0.166	-0.185	-0.228	-0.195
3B-base	-0.990	-1.467	-0.655	-0.840
7B-base	-1.340	-2.288	-0.815	-1.249

Table 3: Bradley-Terry scores over 500 prompts for seven models. Each judge evaluates independently on the pair from the rollout of each model plus the golden response. Per-judge columns are individual position-aware BT fits. Aggregate is a joint BT MLE across all judges with per-judge position (δ) and length (γ) bias parameters. Golden response anchored at 0.

Format Score and Tool Usage: On held-out evaluation (detailed in Appendix A.5, Table 7), SFT achieves 100% and GRPO 99.8% perfect format compliance compared with around 60% for 3B-base. The rare failures are due to the output response length constraint. Tool usage increases from 3-5% (base) to roughly 33-35% (SFT/GRPO), indicating successful learned domain-specific behavior.

Patient Type Ablation: Table 4 shows 3B-GRPO-gpt-4o-reward gains are most pronounced on challenging patient scenarios (+0.880), where our trained model navigates and adapts well.

Patient Type	BT Score	N
medical-specific	+0.595	169
challenging	+0.880	81
general Sotopia	+0.504	250
All	+0.516	500

Table 4: BT scores by patient type with position and length debiased. Challenging patients show the largest GRPO gains.

Task-Specific Metrics: GPT-4o task-specific evaluation across 6 dimensions (detailed in Appendix A.5, Table 8) shows 3B-GRPO-gpt-4o-reward leads across all categories. The largest gains occur in booking decision quality (+0.49 over SFT) and plan quality (+0.47 over SFT), indicating RL training effectively sharpens strategic contextual reasoning. Trust and Comfort as well as booking intention perform the best among all dimensions.

3B-GRPO-RM-reward vs 3B-GRPO-gpt-4o-reward: We find that models guided by gpt-4o in general perform better than models guided by our trained reward reasoning model. This is because judges are also gpt-4o or closed source LLM family, which might favor the model trained with the same kind of reward. In addition, our reward reasoning model is trained on math and ultra-feedback data, which may not have enough knowledge on differentiating patient-client conversation nuances.

5 Analysis

Findings: From these experiments, we have several interesting findings:

1. Paired Consistency addresses position bias. Small-parameter reward models struggle with position bias. The paired consistency penalty mitigates this by enforcing symmetric evaluation across augmented data, demonstrating the necessity to have a structured reward function and pairwise data.
2. GRPO can do better than SFT and base model. The SFT model learns to imitate expert demonstrations but does not learn when to deviate from the demonstrated strategy. This is the limitation of SFT model. GRPO, through pairwise comparison against the golden response, learns that some rollout variants perform better than golden response, particularly on challenging patients. This suggests RL enables the model to discover strategies the expert data does not fully cover.
3. Both SFT and GRPO preserve format very well. Compared with base model, SFT and GRPO

nearly score perfect on the eval dataset, indicating that this capability is learned in the model weights.

4. GRPO performs significantly better on challenging patients. The 3B-GRPO-gpt-4o-reward model’s BT score on challenging patients (+0.880) is 48% higher than medical-specific patients (+0.595). This is because for standard patients, the rollout variance is small, while for challenging patients, rollout variance is high, creating sharp contrast and diverse reward signals that are leveraged by the GRPO algorithm, leading to better strategy learning of how to adapt to those conversations.

5. Early turns are GRPO’s weakness. Golden response wins over GRPO model at 22.1% in early turns (0-3) vs only 7.9% in late turns (10+). GRPO improves as the conversation progresses, indicating it leverages multi-turn context better. This is true for all patient types.

Qualitative Examples: To illustrate, we examine a challenging patient called Shannon Brooks, a 40-year-old counselor who worries about developing compulsive cosmetic behavior. After a brow lamination caused asymmetric eyebrows, she asks the consultant to examine her brows. All three judges unanimously prefer the GRPO response over the golden response. Here is the golden response: I completely understand why you’re feeling upset and confused. Let’s take a very close look together. If you could just come a little closer, I’ll gently examine your brow area. Here is the GRPO response: I’m going to gently examine your brows and the surrounding area now. What I’m looking for is any potential asymmetry or how the muscle activity might be affecting the lift. Sometimes, with the muscles around the forehead and brows, there can be subtle differences in strength or how they’ve responded to the treatment, which can lead to one side appearing higher or different than the other. My goal is to understand exactly what’s happening so we can figure out the best way forward. The golden response acknowledges the patient’s feelings and moves to the exam, but provides no explanation. The GRPO model proactively explains the scientific reasons and information depth behind it and educates an anxious patient during the exam, which is a more informative and reassuring strategy that the expert demonstrations did not fully cover.

6 Conclusion

We present an end-to-end training pipeline for building a domain-specific aesthetic consultant agent. We find that after SFT or RL, structured chain-of-thought reasoning persists nearly perfectly. We also trained a custom 1.5B reasoning model via a novel paired consistency penalty to address position bias. In addition, anchoring BT scores to golden responses provides a calibrated reward signal that enables the GRPO model to surpass expert quality. Finally, the largest gains occur on challenging patients requiring adaptive strategy, demonstrating that RL can enable the model to learn capabilities beyond SFT. The model is deployed as a production service via vLLM inference server, Redis session management, and Cloudflare Tunnel on a single RTX 5090 GPU. The domain is <https://chat.robovector.ai/>.

Our primary limitations are as below: Firstly, our 1.5B Reward Model suffers parameter capacity. Secondly, we employ per-turn training and evaluation, which is unable to optimize for episode-level outcomes such as booking rate. Thirdly, we lack human evaluation. Lastly, position and length bias are adjusted only at evaluation time rather than during the training step. Future work includes pairing a 7B+ teacher with our 1.5B reward model, and using multi-turn outcome reward (rewarding full episodes rather than individual turns so we guide the agent toward final goal rather than intermediate outcome), and scale to larger base models. We also believe estimating position and length bias at training time is beneficial to further debias the reward signal. If time permits, we would like to have human evaluation in addition to LLM judges, so we can capture the nature of empathy. Finally, we will also explore alternative RL algorithms such as DAPO and compare with GRPO.

Team contributions (Required for multi-person team)

Ying Lu: Self-play conversation simulation and data generation pipeline, assistant SFT model training, GRPO model training with golden-anchored Bradley Terry reward algorithms, assistant model variants experimentation, comprehensive evaluation, ablation study and comparison of various model performance and analysis. Qualitative example and analysis. Production service deployment.

Binbin Li: Design the framework to train and fine-tune reward model (1.5B Reasoning, 1.5B Pairwise, 7B Reasoning) built on veRL’s GRPO trainer, including data augmentation, reward configuration, swap-and-compare evaluation protocol.

Da Sun: Parallelize the reward model training for the 1.5B General-Judge and 1.5B Paired-Collator variants, and conducted benchmarking eval across the UltraFeedback and RM-Bench datasets. Designed and implemented the high-throughput vLLM inference API to integrate the custom reward model into the assistant’s GRPO training loop.

References

- [1] Long et al. Training language models to follow instructions with human feedback. 2022.
- [2] Zhou et al. Sotopia: Interactive evaluation for social intelligence in language agents. In *International Conference on Learning Representations*, 2024.
- [3] Dettmers et al. Qlora: Efficient finetuning of quantized llms. 2023.
- [4] Zhihong et al. Deepseekmath: Pushing the limits of mathematical reasoning in open language models. 2024.
- [5] Wei et al. Chain-of-thought prompting elicits reasoning in large language models. 2023.
- [6] DeepSeek-AI, Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, et al. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. 2025.
- [7] Chen et al. Rm-r1: Reward modeling as reasoning. 2025.
- [8] Bai et al. Training a helpful and harmless assistant with reinforcement learning from human feedback. 2022.
- [9] Rafailov et al. Direct preference optimization: Your language model is secretly a reward model. 2023.
- [10] Ali et al. igrpo: Self-feedback–driven llm reasoning. 2026.
- [11] Bradley et al. Rank analysis of incomplete block designs: I. the method of paired comparisons. 1952.
- [12] Zheng et al. Judging llm-as-a-judge with mt-bench and chatbot arena. 2023.
- [13] Yang et al. Qwen2.5 technical reports. 2025.
- [14] Kwon et al. Efficient memory management for large language model serving with pagedattention. 2023.

A Appendix

A.1 Assistant Model Output Format

Each assistant turn is generated in two phases. In the reasoning phase, the model produces a structured thinking block containing four fields: Client/Reflect/Book/Plan. In the response phase, the model generates the spoken response incorporating any tool outputs. This two-phase architecture decouples reasoning from execution, allowing the model to think before acting. The full output format is:

```
<think>
Client: anxious, building trust, exploring, concern=cost
Reflect: Validation worked, opened up about concern.
Book: NO, needs cost clarity before booking.
Plan: educate, tool=get_knowledge_base,
      goal: exploring->considering
</think>
<tool_call>get_knowledge_base</tool_call>
<tool_result name="get_knowledge_base">
(tool output appears here)
</tool_result>
<response>
Your spoken response incorporating tool results.
</response>
```

A.2 Reward Model Configurations & Hyperparameters

To isolate the optimal judge architecture, we tracked the performance of five custom models through our training process and compared them against the official 7B baseline released by the RM-R1 authors.

- **1.5B Reasoning:** Qwen2.5-1.5B-Instruct trained via SFT then RLVR on the baseline gaotang/RM-R1-Reasoning-RLVR datasets. Uses a standard +1/-1 reward.
- **7B Reasoning:** A scaled-up variant of Model 1 using Qwen2.5-7B-Instruct to test if increased parameter size resolved task deficiencies on the baseline data.
- **1.5B General-Judge:** DeepSeek-R1-Distill-1.5B trained via direct RLVR (skipping SFT) on the baseline gaotang/RM-R1-after-Distill-RLVR datasets, using a standard +1/-1 reward to test the viability of reasoning-distilled base weights.
- **1.5B Paired-Collator:** DeepSeek-R1-Distill-1.5B trained via direct RLVR on augmented UltraFeedback position-swapped pairs. Evaluated with an asymmetric +1/0/-1/-2 reward function penalizing inconsistency and formatting errors.
- **1.5B Paired (Final Model):** Qwen2.5-1.5B-Instruct trained via SFT, then RLVR on augmented UltraFeedback pairs using a +1/-1/-0.5 paired consistency reward. This model achieved the optimal balance of debiasing and accuracy.
- **7B RM-R1 (Baseline):** The official release from the RM-R1 paper, used as an external benchmark for comparison.

Below is the experiment training parameters used for SFT and GRPO with RLVR.

	1.5B Reasoning	1.5B Paired	1.5B General-Judge	1.5B Paired-Collator	7B Reasoning	7B RM-R1
Base Model	Qwen2.5-1.5B-Instruct	Qwen2.5-1.5B-Instruct	DeepSeek-R1-Distill-1.5B	DeepSeek-R1-Distill-1.5B	Qwen2.5-7B-Instruct	DeepSeek-R1-Distill-7B
Reward Function	+1 / -1	+1 / -1 / -0.5	+1 / -1	+1 / 0 / -1 / -2	+1 / -1	+1 / -1
Reward Meaning	correct / wrong	correct / wrong / biased	correct / wrong	correct / wrong / biased / format error	correct / wrong	correct / wrong
Data Augmentation	None	Pairwise (original + swapped, adjacent)	None	Pairwise (original + swapped, adjacent)	None	None
Training Data	RM-R1-Distill-SFT, RM-R1-Reasoning-RLVR	RM-R1-Distill-SFT, Ultrafeedback-Binarized	RM-R1-after-Distill-RLVR	Ultrafeedback-Binarized	RM-R1-Distill-SFT, RM-R1-Reasoning-RLVR	RM-R1-Reasoning-RLVR

Table 5: Comparison of Reward Shaping and Debiasing Strategies in GRPO-based Training

A.3 Reward Model Evaluation Metrics

For every pairwise comparison in these test sets, we employ a two-pass inference protocol. We prompt the model twice: once with the original A/B candidate ordering (Pass 1) and once with the swapped B/A ordering (Pass 2). By analyzing the model’s verdicts across both passes, we compute the following metrics:

- **Position Bias (A-selection rate):** Average probability of selecting response A across both orderings: $\frac{A_sel_pass1 + A_sel_pass2}{2}$. Ideal = 50%. Values above 50% indicate bias toward the first-presented response.
- **Consistency:** Fraction of examples where the model selects the same underlying response across both orderings. Measures position-invariance.
- **Debiased Accuracy:** Fraction of examples where the model selects the correct (ground-truth better) response in *both* orderings. This is the strictest metric, requiring the model to be both correct and consistent.
- **Total Accuracy:** Average of Pass 1 and Pass 2 accuracy: $\frac{acc_pass1 + acc_pass2}{2}$. Measures raw correctness regardless of consistency.
- **Format Adherence:** Fraction of responses containing a parseable `<answer>[[A]]</answer>` or `<answer>[[B]]</answer>` verdict.

A.4 Assistant Model Training Hyperparameters

Parameter	3B-SFT	3B-GRPO-gpt-4o-reward
Base Model	Qwen2.5-3B-Instruct	SFT-merged model
LoRA Rank / Alpha	64 / 128	64 / 64
Learning Rate	1×10^{-4}	5×10^{-6}
Batch Size	8	8 prompts
Max Seq Length	4,096	4,096(max prompt length) + 512(max response length)
Epochs / Steps	5 epochs	1 epoch, 250 steps
KL Coefficient	NA	0.001
Rollouts per Prompt	NA	5 (+1 golden)
GPU	RTX 5090 (32GB)	RTX 5090 (32GB)

Table 6: Training configurations for SFT and GRPO.

A.5 Extended Assistant Model Evaluation Metrics

Model	Format	Perfect %	Tool Use %
3B-base	0.673	58.8	3.0
3B-SFT	1.000	100.0	33.4
3B-GRPO-RM-reward	1.000	99.8	35.2
3B-GRPO-gpt-4o-reward	1.000	99.8	33.2
7B-base	0.252	16.8	5.4
7B-SFT	0.998	99.0	33.2

Table 7: Metrics on format compliance and tool usage.

Dimension	3B-GRPO-gpt-4o-reward	3B-GRPO-RM-reward	SFT	Base
Trust and Comfort	8.91	8.68	8.68	8.36
Information Quality	7.53	7.34	7.26	6.76
Booking Intention	8.72	8.37	8.42	8.04
Reflect Quality	6.27	5.78	5.76	5.67
Book Decision Quality	7.75	7.23	7.26	7.03
Plan Quality	6.62	6.12	6.15	6.05

Table 8: gpt-4o rated task-specific scores (1 to 10 scale) across consultation quality dimensions.

A.6 Training Dynamics of the 1.5B Paired Model

Between steps 100 and 225, position bias drops sharply from $\sim 70\%$ down to the ideal 50% line. Only after this positional shortcut is unlearned does the total accuracy reliably increase from 58% to its peak near 69%. Steps 0 - 100 are a "warm-up" phase with no visible learning; the inflection point is around step 125. This sequential dynamic proves the penalty structure operates well as it needs to break the model’s reliance on position before teaching it true preference alignment.

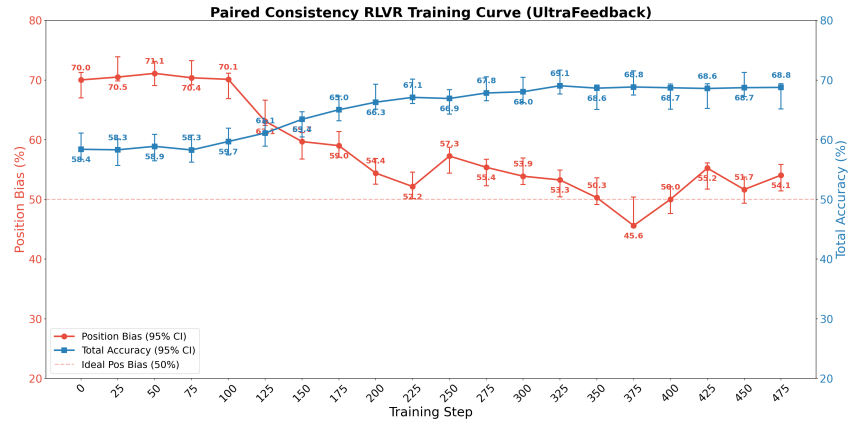


Figure 2: Evaluation of the 1.5B Paired Model on UltraFeedback