TAIRL: OPTIMIZING MULTI-TURN OPEN-ENDED DIALOGUE VIA TEMPORAL PATTERNS IN HUMAN FEEDBACK

Anonymous authors
Paper under double-blind review

ABSTRACT

Multi-turn open-ended dialogue (MOD) is a core scenario in human-machine interaction, where subjective factors such as emotion, interest, knowledge, and topic shift play a decisive role in dialogue quality. However, many interactive reinforcement learning (RL) methods use human feedback as turn-level scalar supervision, leaving its temporal structure underexplored. Motivated by an empirical analysis of human feedback in financial-domain MOD, we propose TAIRL, a temporal-aware interactive RL framework that models three recurring feedback patterns: turn-point concentration, asymmetric timing, and adaptive frequency. TAIRL operationalizes these patterns through temporally-guided dynamic weighting (TDW), an asymmetric learning rate (ALR), and adaptive experience replay (AER). In an offline-feedback-driven simulator trained from 4,287 real human feedback samples, TAIRL reduces simulator convergence turns by 36–45% and improves sample efficiency by approximately 50% over several RL baselines. On held-out human-evaluated financial dialogues with Qwen2.5-7B, TAIRL-enhanced methods also improve final task success and user-experience metrics, including a 22.8% relative improvement in user satisfaction and a 56% relative improvement in emotional alignment. The proposed mechanisms are lightweight and can be integrated into PPO, GRPO, REINFORCE, and TAMER-style interactive learning. These results suggest that temporal feedback structure is a useful design signal for improving interactive RL in MOD, while broader cross-domain validation remains an important direction for future work. The code and data are available at `https://anonymous.4open.science/r/TAIRL-2957`.

1 INTRODUCTION

Reinforcement Learning from Human Feedback (RLHF) has become a cornerstone of Large Language Model (LLM) alignment, and has been widely adopted in models such as ChatGPT, Claude, and Llama (Chaudhari et al., 2025). The Training an Agent Manually via Evaluative Reinforcement (TAMER) (Fan et al., 2021) framework, as an early precursor to RLHF, established key principles for learning from evaluative human feedback (Wang et al., 2025). However, many interactive RL and RLHF algorithms use human feedback primarily as turn-level scalar supervision, leaving the temporal structure of feedback underexplored (Xu et al., 2024; Parmaxi, 2023). This can limit their effectiveness in dynamic interactive tasks, where the timing, type, and frequency of feedback may carry information beyond the feedback value itself (Qureshi et al., 2023). For instance, AI teaching requires the model to understand the processual distribution of feedback, when to intervene, how to adjust, and how to manage the accumulation and shifts of emotions (Chiu et al., 2024). Treating feedback as isolated scalar signals makes it difficult to capture these interaction dynamics (Qureshi et al., 2023; Chiu et al., 2024).

This limitation is particularly pronounced in the multi-turn open-ended dialogues (MOD) scenario (Yoshida, 2023). Unlike single-turn RLHF, LLMs in real-world applications must engage in MOD with humans, managing topic shifts, expressing emotions, sharing knowledge, and building consensus (Wang et al., 2023; Shu et al., 2024). Recent studies have recognized this challenge: Multi-turn RLHF suffers from a severe covariate shift problem (Gao et al., 2025). The training dialogues are generated by a reference policy; consequently, when the model is deployed in the actual dialogue loop, a low training error does not necessarily translate into good performance. Reinforcement learning from human feedback with macro actions (MA-RLHF) (Chai et al., 2025) proposes using macro actions to shorten the temporal distance between actions and rewards, achieving 18% performance improvement on MOD tasks. Playpen (Horst et al., 2025) (an environment for exploring learning from dialogue game feedback) validates the feasibility of training LLMs via interactive learning in dialogue games. Despite these advances, existing multi-turn RLHF methods rarely model the

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

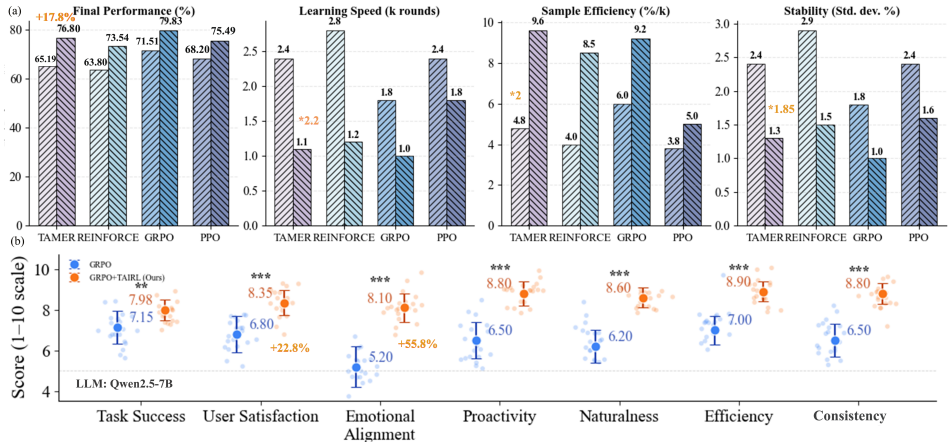

Figure 1: **Performance and user experience comparison on D3 with Qwen2.5-7B.**

temporal patterns of human feedback itself; feedback must inform the model not only "what to say" but also when feedback is most informative (Chai et al., 2025; Horst et al., 2025).

In our preliminary analysis of financial-domain MOD feedback, we observe three temporal patterns (see **Appendix B** for details): turn-point concentration (80% of feedback is concentrated in annotated state-switching windows), asymmetric timing (negative feedback is almost immediate with 0.8s delay, while positive feedback lags with 3.2s delay), and adaptive frequency (feedback increases by 77.5% when performance is poor and decreases by 61.3% when performance is good). These patterns suggest that human feedback is not homogeneous: feedback tends to be more informative around switching windows, during low-performance periods, and when it is negative. We further validate these patterns on two cross-domain datasets (medical and home-service consultation) in Section 4. Based on these insights, we propose **a temporal-aware interactive reinforcement learning algorithm, termed TAIRL,** which encodes temporal feedback patterns into design. It comprises three core mechanisms: temporally-guided dynamic weighting (assigning higher weights to feedback within switching windows), an asymmetric learning rate (high learning rate for negative feedback and low learning rate for positive feedback), and adaptive experience replay (sampling low-performance or high-intervention trajectories with higher weights). The main contributions of this paper are as follows:

1. We provide a quantitative analysis of temporal feedback patterns in financial-domain MOD and show how they can inform interactive RL design.

2.We propose the TAIRL algorithm, which comprises three pluggable mechanisms: temporally-guided dynamic weighting (TDW), asymmetric learning rate (ALR), and adaptive experience replay (AER).

3.We validate TAIRL on four real-world multi-turn dialogue datasets spanning financial, medical, and home-service domains, showing that aligning interactive RL with human feedback improves both performance and user experience across datasets.

2 Related Work

2.1 Human Factors in Interactive RL

Various RLHF-family methods have been applied to align LLMs, including Direct Preference Optimization (DPO) (Hong et al., 2026), Identity Preference Optimization (IPO) (Ip et al., 2025), Contrastive Preference Optimization (CPO) (Gupta et al., 2025), Proximal Policy Optimization (PPO) (Huang et al., 2024), REINFORCE (Ahmadian et al., 2024), and Group Relative Policy Optimization (GRPO) (Jia et al., 2026). The TAMER framework, a precursor to RLHF, allows humans to provide reward signals (+1/-1) without a separate reward model (Fan et al., 2021). These works focus on the learning algorithm of the AI, while the human teacher's behavior remains largely unstudied (Fan et al., 2021; Wang et al., 2025). A few studies examine human factors in interactive RL. The Unified Framework for Policy Alignment in Reinforcement Learning from Human Feedback (PARL) shows that human feedback is influenced by the agent's current policy (Chakraborty et al., 2024). Predictive Preference Learning from Human Interventions (PPL) (Cai et al., 2025), Evaluative Corrective Guidance Language as Reinforcement (ECLAIR) (Tarakli et al., 2024), and Human Guided Exploration (HuGE) (Torne Villasevil et al., 2023) show that humans use feedback both to

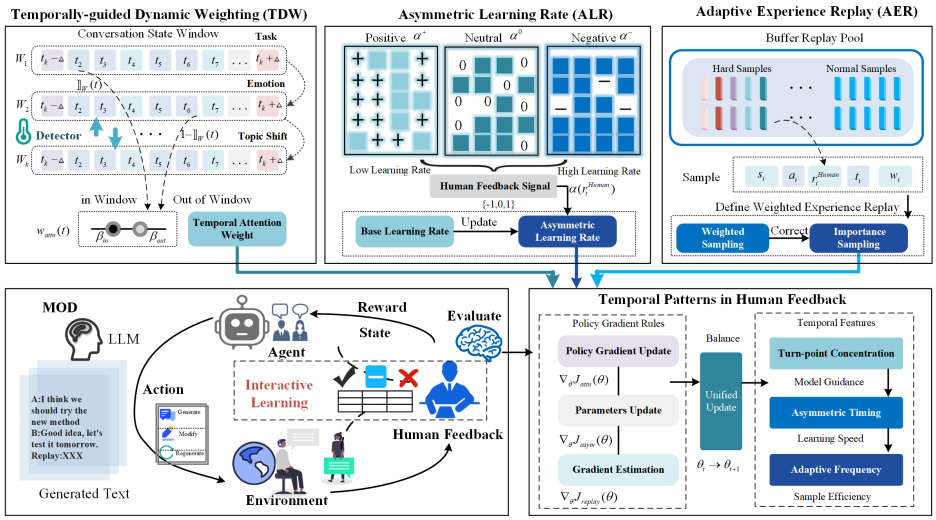

Figure 2: **The framework of TAIRL**.

evaluate past actions and to guide future behavior. Some studies further analyze the consistency of human feedback (Icarte et al., 2022): the robust RL from corrupted human feedback framework, denoted as R^{3}M (Bukharin et al., 2024), identifies and down-weights corrupted feedback signals. However, these studies are primarily qualitative and lack quantitative analysis of the temporal patterns of feedback(Icarte et al., 2022; Bukharin et al., 2024).

2.2 Selective Training Methods

This section reviews the temporal characteristics of training signals. Temporal information has been widely studied in RL: Prioritized Experience Replay (PER) (Liu et al., 2022) weights samples by temporal difference (TD) errors, Hindsight Experience Replay (HER) (Zeng et al., 2023) relabels goals, and temporal difference learning exploits time-difference structure (Chen et al., 2025). These works, however, focus on environmental rewards rather than human feedback, which is selective, asymmetric, and adaptive (Zeng et al., 2023; Chen et al., 2025). Research on selective training signals shows that not all signals are equally important. In robotics, Interactive RL Policy Patching (Lin et al., 2024) corrects policy failures via short interventions. In LLM pre-training, RHO-1 (also known as Selective Language Modeling) selects high-excess-loss tokens, improving accuracy by 30% and reducing training time to 1/5–1/10 of the original (Lin et al., 2024). In LLM alignment, Selective Preference Optimization (SePO) (Yang et al., 2025) supervises 30% of critical tokens, cutting GPU training time by 60%. In interactive RL, Episodic Active Learning from Demonstration queRY (EARLY) (Hou et al., 2024) and Active Query Selection for Crowd-Based Reinforcement Learning (Erskine et al., 2025) select optimal query timing based on uncertainty. In multi-agent systems, Selective Policy Transfer (SPOT) (Zhuang et al., 2024) lets agents decide when to transfer policies. Although successful in their domains, these methods share a limitation: their criteria rely on excess loss or uncertainty, ignoring the temporal distribution of training signals (Hou et al., 2024; Zhuang et al., 2024). These neglected temporal cues are precisely the core features of human feedback. In summary, existing research has three gaps: (1) no quantitative analysis of temporal patterns in human feedback; (2) no integration of these patterns into algorithm design; and (3) no treatment of how temporal structure affects learning efficiency. TAIRL addresses these gaps as the first framework to introduce temporal patterns of human feedback into interactive RL algorithm design.

3 TAIRL Algorithm

Based on temporal patterns of human feedback discovered in **Appendix B**, we propose TAIRL framework, illustrated in Figure 2. Core idea of TAIRL is to encode three temporal patterns into algorithm design: TDW captures dependencies among feedback signals, ALR modulates model's response speed to different types of feedback, and AER reinforces samples collected during low-performance or high-intervention periods.

3.1 Basic Definitions of Interactive RL

Consider a general interactive reinforcement learning setup. At each time step t, the agent (an LLM for MOD) observes a state $s_t \in \mathcal{S}$ (the current dialogue context), and takes an action $a_t \in \mathcal{A}$ (generating a response). The state transition probability is given by $P(s_{t+1} \mid s_t, a_t)$. A human trainer observes and provides feedback $r_t^{\text{Human}} \in \mathcal{R} = \{+1, 0, -1\}$, representing positive, neutral,

and negative feedback, respectively. The policy $\pi_\theta(a_t \mid s_t)$ is parameterized by θ, and the objective is to maximize the expected cumulative feedback:

$$J(\theta) = \mathbb{E}_{\tau\sim\rho_\pi}\left[\sum_{t=0}^{T}\gamma^t r_t^{\text{Human}}\right] = \int_{\mathcal{T}} P(s_0)\prod_{t=0}^{T}\pi_\theta(a_t \mid s_t)P(s_{t+1} \mid s_t, a_t)\left(\sum_{t=0}^{T}\gamma^t r_t^{\text{Human}}\right)d\tau, \tag{1}$$

where $\tau = (s_0, a_0, r_0, s_1, a_1, r_1, \ldots, s_T)$ is a trajectory, $\rho_\pi(\tau)$ is the trajectory distribution, and $\gamma \in [0, 1]$ is the discount factor. By the policy gradient theorem, the gradient is:

$$\nabla_\theta J(\theta) = \mathbb{E}_{\tau\sim\rho_\pi}\left[\sum_{t=0}^{T}\nabla_\theta \log \pi_\theta(a_t \mid s_t)\cdot Q^\pi(s_t, a_t)\right], \quad Q^\pi(s_t, a_t) = \mathbb{E}_{\tau\sim\rho_\pi}\left[\sum_{k=t}^{T}\gamma^{k-t} r_k\right], \tag{2}$$

where $Q^\pi(s_t, a_t)$ is the state-action value function. In the TAMER framework, human feedback is directly used as the reward signal, thus $Q^\pi(s_t, a_t) \approx r_t^{\text{Human}}$.

3.2 Temporally-Guided Dynamic Weighting

The turn-point concentration pattern reveals that human feedback is highly concentrated in state-switching windows (e.g., topic shifts, task transitions, emotional changes), where feedback is empirically more frequent and potentially more informative. To exploit this, we design a TDW mechanism that assigns higher weights to feedback within such windows. However, window detection must be dynamic rather than fixed. We therefore introduce a lightweight state-shift detector. Define a dynamic state-shift detection function $\varphi(s_t, s_{t-1}) \in \{0, 1\}$, whose inputs are the context embeddings of two consecutive dialogue turns s_{t-1}, s_t obtained via a lightweight Sentence-BERT model. The detector is based on three deterministic signals: semantic similarity drop ($\mathbb{I}_{\text{sim}}$), emotion polarity reversal ($\mathbb{I}_{\text{emo}}$), and topic blocking ($\mathbb{I}_{\text{topic}}$). A valid shift is triggered only when a semantic shift co-occurs with either emotion reversal or topic blocking, following the veto logic:

$$\varphi(s_t, s_{t-1}) = \mathbb{I}_{\text{sim}}(t) \wedge (\mathbb{I}_{\text{emo}}(t) \vee \mathbb{I}_{\text{topic}}(t)). \tag{3}$$

Dynamic window generation. Let $\{t_k\}$ denote the set of detected shift moments, each satisfying $\varphi(s_{t_k}, s_{t_k-1}) = 1$. For each shift moment t_k, a dynamic window $W_k = [t_k - \Delta, t_k + \Delta]$ is defined, where Δ is the half-window width. Let the set of switching windows be $\mathcal{W} = \{W_1, W_2, \ldots, W_K\}$. Then the window indicator function is given by:

$$\mathbb{I}_{\mathcal{W}}(t) = \sum_{k=1}^{K}\mathbb{I}\{t \in W_k\} = \begin{cases}1, & \text{if } \exists k : t \in W_k,\\ 0, & \text{otherwise.}\end{cases} \tag{4}$$

The temporal weight $w_{\text{attn}}(t)$ is then defined as:

$$w_{\text{attn}}(t) = \beta_{\text{in}}\cdot\mathbb{I}_{\mathcal{W}}(t) + \beta_{\text{out}}\cdot\big(1 - \mathbb{I}_{\mathcal{W}}(t)\big), \tag{5}$$

where β_{in} is the intra-window weight and β_{out} is the extra-window weight. This design makes the total contribution of in-window feedback different from that of out-window feedback.

The TDW is incorporated as a weighting term in the policy update:

$$\nabla_\theta J_{\text{attn}}(\theta) = \mathbb{E}_{\tau\sim\rho_\pi}\left[\sum_{t=0}^{T} w_{\text{attn}}(t)\cdot\nabla_\theta \log \pi_\theta(a_t \mid s_t)\cdot\hat{Q}^\pi(s_t, a_t)\right], \tag{6}$$

where $\hat{Q}^\pi(s_t, a_t)$ denotes an estimate of the value function.

In the online update setting, using a one-step Monte Carlo (Mohamed et al., 2020) estimate $\hat{Q}^\pi(s_t, a_t) = r_t^{\text{Human}}$, the gradient simplifies to:

$$\nabla_\theta J_{\text{attn}}(\theta) = \mathbb{E}_{(s_t, a_t, r_t^{\text{Human}})}\left[w_{\text{attn}}(t)\cdot\nabla_\theta \log \pi_\theta(a_t \mid s_t)\cdot r_t^{\text{Human}}\right]. \tag{7}$$

This lightweight dynamic detector incurs minimal online overhead in our implementation. The fixed-window statistics in **Appendix B.2** are only offline evidence and are not required by the algorithm.

3.3 Asymmetric Learning Rate

The asymmetric timing pattern reveals that negative feedback is delivered almost instantaneously while positive feedback lags significantly, reflecting a "correct now, confirm later" human strategy. In LLM alignment, erroneous responses (e.g., hallucinations, bias, unsafe content) require immediate correction, whereas correct responses can be reinforced gradually. Consequently, negative feedback

should be learned quickly with a high learning rate, while positive feedback should be absorbed steadily with a low learning rate. Define the learning rate function $\alpha : \mathcal{R} \to \mathbb{R}^+$:

$$\alpha(r_t^{\text{Human}}) = \alpha_0 \cdot \exp\left(\eta \cdot \mathbb{I}\{r_t^{\text{Human}} = -1\} - \xi \cdot \mathbb{I}\{r_t^{\text{Human}} = +1\}\right), \tag{8}$$

where α_0 is the base learning rate. η and ξ are coefficients that control the exponential amplification for negative and positive feedback. Setting $\eta = \log(10) \approx 2.3026$ and $\xi = 0$ gives $\alpha(-1) = \alpha_0 \cdot e^{\eta} = 0.1, \quad \alpha(+1) = \alpha_0 = 0.01, \quad \alpha(0) = \alpha_0 = 0.01$. More generally, a discrete form is:

$$\alpha(r_t^{\text{Human}}) = \begin{cases} \alpha^- = \alpha_0 \cdot \mathbb{K}, & \text{if } r_t^{\text{Human}} = -1, \\ \alpha^+ = \alpha_0, & \text{if } r_t^{\text{Human}} = +1, \\ \alpha^0 = \alpha_0 \cdot \varepsilon, & \text{if } r_t^{\text{Human}} = 0, \end{cases} \tag{9}$$

where $\mathbb{K}$ is the asymmetry coefficient and ε is the decay coefficient for neutral feedback.

Incorporating the ALR into the policy update:

$$\nabla_\theta J_{\text{asym}}(\theta) = \mathbb{E}_{(s_t, a_t, r_t^{\text{Human}})}\left[\alpha(r_t^{\text{Human}}) \cdot \nabla_\theta \log \pi_\theta(a_t \mid s_t) \cdot r_t^{\text{Human}}\right]. \tag{10}$$

The parameter update rule is:

$$\theta_{t+1} = \theta_t + \alpha(r_t^{\text{Human}}) \cdot w_{\text{attn}}(t) \cdot \nabla_\theta \log \pi_\theta(a_t \mid s_t) \cdot r_t^{\text{Human}}. \tag{11}$$

Although negative feedback uses a high learning rate and positive feedback a low learning rate, the actual parameter update step is also jointly constrained by the temporal weight $w_{\text{attn}}(t)$ and the gradient norm. The maximum change in a single parameter update satisfies:

$$\|\Delta\theta\| \le \alpha^- \cdot \max(w_{\text{attn}}) \cdot \|\nabla_\theta \log \pi_\theta\| \le \alpha^- \cdot \max(\beta_{\text{in}}, \beta_{\text{out}}) \cdot G_{\max}. \tag{12}$$

where $G_{\max}$ is the gradient clipping threshold. Thus, the update magnitude remains explicitly bounded by the chosen learning-rate and clipping parameters. In practice, this bounded asymmetric update encourages faster correction after negative feedback while using smaller positive-feedback steps for steadier consolidation.

3.4 Adaptive Experience Replay

The adaptive frequency pattern shows that humans increase feedback frequency and granularity when performance is poor, and decrease them when it is good. We therefore design an adaptive experience replay (AER) mechanism that assigns higher sampling weights to low-performance or high-intervention samples, and lower weights otherwise. Let $p_t \in [0, 1]$ denote the average task success rate over the most recent N turns:

$$p_t = \frac{1}{N} \sum_{i=t-N+1}^{t} \mathbb{I}\{\text{success}_i\}. \tag{13}$$

Let the experience replay buffer be $\mathcal{B} = \{(s_i, a_i, r_i^{\text{Human}}, t_i, w_i)\}_{i=1}^{|\mathcal{B}|}$, where i indexes a sample. The sampling weight for each experience sample e is:

$$w_{\text{sample}}(e) = \exp(-\gamma_2 \cdot p_{t_e}) \cdot w_{\text{attn}}(t_e), \tag{14}$$

where γ_2 is a temperature parameter, p_{t_e} is the agent's performance when sample e was collected, and t_e is the timestep.

Define the gradient estimate for weighted experience replay as:

$$\nabla_\theta J_{\text{replay}}(\theta) = \frac{1}{|\mathcal{B}_{\text{batch}}|} \sum_{e \in \mathcal{B}_{\text{batch}}} w_{\text{sample}}(e) \cdot \nabla_\theta \log \pi_\theta(a_e \mid s_e) \cdot r_e^{\text{Human}}, \tag{15}$$

where $\mathcal{B}_{\text{batch}} \sim \text{WeightedSampling}(\mathcal{B}, w_{\text{sample}})$ is a mini-batch sampled according to the weights, and $a_e, s_e, r_e^{\text{Human}}$ are the state, action, and human feedback collected at the time sample e was stored.

To ensure sampling efficiency, weighted sampling is implemented using a priority queue, resulting in a time complexity of $O(\log |\mathcal{B}|)$. The sampling probability is:

$$P(e) = \frac{w_{\text{sample}}(e)^\chi}{\sum_{e' \in \mathcal{B}} w_{\text{sample}}(e')^\chi}, \tag{16}$$

where χ is the priority exponent. The resulting replay distribution is biased toward temporally important samples; we discuss this bounded auxiliary bias in **Appendix C**.

3.5 Complete Update Rule and Convergence Analysis

TAIRL integrates the above three mechanisms into a unified policy gradient update rule:

$$\theta_{t+1} = \theta_t + \alpha(r_t^{\text{Human}}) \cdot w_{\text{attn}}(t) \cdot \nabla_\theta \log \pi_\theta(a_t \mid s_t) \cdot r_t^{\text{Human}} + \frac{\lambda}{|\mathcal{B}_{\text{batch}}|} \sum_{e \in \mathcal{B}_{\text{batch}}} w_{\text{sample}}(e) \cdot \nabla_\theta \log \pi_\theta(a_e \mid s_e) \cdot r_e^{\text{Human}}, \quad (17)$$

where λ is a coefficient for the experience replay weight, balancing the contributions of the online update and the replay update.

Integration with base optimizers. TAIRL is implemented as a weighting module rather than a replacement for the base optimizer. For REINFORCE and TAMER-style updates, the weights multiply the human-feedback return as shown above. For PPO and GRPO, the same temporal and asymmetric factors are applied to the reward or advantage terms, while the optimizer-specific clipping, KL regularization, and group normalization terms remain unchanged. Adaptive replay affects the sampling priority of stored trajectories but does not alter the base objective itself. The TAIRL update rule can be viewed as adding temporally informed weight scaling to the standard policy-gradient update. Under standard regularity assumptions, this weighted update follows the usual non-convex stochastic-optimization behavior up to a bounded replay-bias term. When the weighting factors, learning-rate multipliers, and human feedback are bounded, the variance of the scaled gradient estimate is roughly bounded as:

$$\text{Var}\left(\hat{\nabla}_\theta J_{\text{TAIRL}}\right) \leq (\alpha_{\max} w_{\max})^2 \cdot \text{Var}(\nabla_\theta \log \pi_\theta) \cdot \mathbb{E}\left[(r_t^{\text{Human}})^2\right], \quad (18)$$

where $w_{\max} = \max_t w_{\text{attn}}(t)$. We provide a finite-sample convergence analysis of the weighted surrogate J_w in **Appendix C.4**, which establishes an $O(1/\sqrt{T})$ rate up to a bounded replay-bias term.

4 Experimental Setup and Results

4.1 Datasets and Evaluation Protocol

We evaluated TAIRL on four datasets: D1 (medical consultation, multi-turn dialogues covering examination recommendation, and treatment discussion (Liu et al., 2026)), D2 (home-service consultation, customer reviews from a home-services platform with aspect-level sentiment annotations (Maroof et al., 2025)), D3 (stock prediction, 4K turns), and D4 (market discussion, 3.5K turns). D1–D4 all follow the same turn-level feedback annotation protocol. D3/D4 constitute the primary evaluation setting of this work, while D1/D2 serve as cross-domain validation sets used to assess the transferability of the temporal feedback patterns. Online RL training follows an offline-feedback-driven simulator paradigm. First, 4,287 real human feedback samples are collected and used to train a reward model (RM) as a user simulator. The RM achieves 87.6% sign accuracy and 0.93 Pearson correlation on held-out feedback, and approximates the observed temporal patterns within the studied domain. All subsequent policy training (PPO, GRPO, REINFORCE and TAIRL versions) runs in this simulator for 2,000-3,000 turns. Final performance (FP) is blindly evaluated on an independent set of 800 real human dialogues (see **Appendix F**), with each dataset contributing 200 held-out turns. Thus, Learning speed (LS) and sample efficiency (SE) should be interpreted as simulator-based metrics, while FP and user-experience scores are measured by human evaluation. A time-shuffle control experiment that rules out simulator circularity is provided in Appendix F.2.

4.2 Experimental Evaluation Metrics and Baseline Algorithms

We evaluate TAIRL using four metrics across different LLMs and RL frameworks.

1). LS: The number of training turns required to achieve 80% of the final FP of the corresponding TAIRL-enhanced model, providing a dataset-independent convergence criterion.

$$\text{LS} = \min\{t \mid \text{Acc}(t) \geq 0.8 \cdot \text{FP}_{\text{final}}\}. \quad (19)$$

2). FP: The average task success rate after 40 dialogue turns.

$$\text{FP} = \frac{1}{|\mathcal{D}_{\text{test}}|} \sum_{i=1}^{|\mathcal{D}_{\text{test}}|} \mathbb{I}(\hat{y}_i = y_i), \quad (20)$$

where $\mathcal{D}_{\text{test}}$ is the test set, $|\mathcal{D}_{\text{test}}|$ is the total number of test samples, and $\hat{y}_i$ and y_i are the predicted and ground truth answers, respectively. For D1/D2, $\hat{y}_i$ and y_i denote the model's answer and the reference label in the corresponding consultation task. For D3/D4, which are MODs with standardized terminal

Table 1: Performance comparison across D1–D4 datasets. LS and SE are measured in the simulator environment; FP and SB are measured on real human dialogues. LS is reported in thousands of turns (k), and SE in % per 1k feedback signals. DPO is an offline method and does not have online convergence metrics (LS/SE).

Method	D1 (Medical)				D2 (Home-service)				D3 (Stock)				D4 (Market)			
	FP (%)	LS (k)	SE (%/k)	SB (%)	FP (%)	LS (k)	SE (%/k)	SB (%)	FP (%)	LS (k)	SE (%/k)	SB (%)	FP (%)	LS (k)	SE (%/k)	SB (%)
TAMER	61.40	3.2	3.6	2.70	40.20	4.4	3.1	3.21	65.19	2.4	4.8	2.40	62.02	2.6	4.5	2.60
TAMER+TAIRL	66.20	2.2	6.2	2.00	44.63	3.1	5.2	2.20	76.80	1.1	9.6	1.30	69.54	1.2	9.0	1.41
REINFORCE	60.09	3.4	3.3	3.10	39.10	4.7	2.8	3.81	63.80	2.8	4.0	2.90	60.50	3.0	3.8	3.00
REINFORCE+TAIRL	65.04	2.3	5.8	2.22	43.64	3.3	4.9	2.93	73.54	1.2	8.5	1.50	67.82	1.3	8.2	1.63
GRPO	67.20	2.6	5.0	2.08	44.20	3.6	4.3	2.50	71.51	1.8	6.0	1.79	68.01	2.0	5.8	2.00
GRPO+PER	68.80	2.0	6.1	1.60	46.80	2.7	5.3	2.30	74.21	1.5	7.2	1.40	71.05	1.6	7.2	1.40
GRPO+TAIRL	72.01	1.6	7.5	1.11	47.83	2.3	6.4	1.50	79.83	1.0	9.2	1.20	76.50	1.1	8.8	1.13
PPO	63.50	2.9	4.0	2.40	42.60	4.0	3.4	3.20	68.20	2.4	3.8	2.40	64.51	2.6	4.0	2.50
PPO+TAIRL	68.92	2.0	5.0	1.82	45.21	2.9	4.5	2.34	75.49	1.8	5.0	1.62	71.83	2.0	4.8	1.84
DPO	60.78	—	—	2.20	39.70	—	—	2.60	69.78	—	—	2.10	62.20	—	—	2.40
SePO	61.60	2.4	4.5	1.90	40.88	3.2	3.8	2.30	70.51	1.8	5.5	1.60	63.21	2.0	5.2	1.80

decisions, $\hat{y}_i$ and y_i denote the normalized terminal decision label produced by the model at the end of the dialogue and the expert-annotated reference decision label, respectively. Specifically, D3 requires a three-way up/down/flat risk-return judgment compared with the next-day realized price movement, and D4 requires selecting the best candidate target from a list compared with expert-annotated research-report interpretation labels. Intermediate dialogue turns are evaluated separately by human feedback (+1/0/-1) and are not part of the FP metric.

3). SE: The performance gain per thousand feedback signals.

$$\text{SE} = \frac{\Delta\text{Acc}}{\text{Feedback}(k)} \times 100\%. \tag{21}$$

Here, Feedback(k) is the total human feedback count (in thousands).

4). Stability (SB): The standard deviation of performance across 5 random seeds.

$$SB = \sigma(\text{Acc}^{(1)}_{\text{final}}, \text{Acc}^{(2)}_{\text{final}}, \ldots, \text{Acc}^{(5)}_{\text{final}}). \tag{22}$$

The baselines are divided into three groups: policy optimization baselines including PPO, GRPO, and REINFORCE; preference and selective baselines introducing DPO and SePO; and core enhancement comparison, where TAMER is compared with TAIRL to validate the general optimization capability of TDW and ALR across dynamic interaction tasks. We provide a detailed discussion of evaluation independence, simulator fidelity, and evaluation scale in **Appendix F**.

4.3 Experimental Results Analysis

4.3.1 Performance Metric Comparison

We evaluate the temporal enhancement effect of TAIRL on four datasets: D1 (medical consultation), D2 (home-service consultation), D3 (stock prediction), and D4 (market discussion). Using Qwen2.5-7B as the base model, we compare the baselines and their TAIRL-enhanced counterparts across four dimensions. The experimental results are presented in Table 1. On cross-domain tasks D1/D2 (medical and home-service consultation), GRPO+TAIRL improves FP over GRPO by 4.81 and 3.63 percentage points (to 72.01% and 47.83%), with relative gains of 7.2% and 8.2%. On financial-domain tasks D3/D4, gains are larger: +8.32 and +8.49 percentage points (to 79.83% and 76.50%), with relative gains of 11.6% and 12.5%. Simulator convergence turns drop by 36–45%, and SE improves by approximately 50%. TAMER+TAIRL and REINFORCE+TAIRL also gain 4.4–11.6 percentage points over their baselines. Notably, GRPO+PER yields only 1.6–3.0 percentage points, whereas GRPO+TAIRL reaches 3.6–8.5 percentage points, suggesting that TDW and ALR contribute beyond standard replay alone. We further examine model-level robustness by testing TAIRL on eight open-source LLMs (varying sizes and architectures) using D3 and D4. Table 2 shows that TAIRL-enhanced methods maintain similar relative trends across models. GRPO+TAIRL ranks first on both datasets for all eight models; on Qwen2.5-7B, it improves FP over GRPO by 8.32 and 8.49 percentage points on D3 and D4, corresponding to relative gains of 11.6% and 12.5%. Across scales, 7B models outperform 1–2B models by 36–53 percentage points, while the relative ranking of methods remains invariant across sizes and families.

4.3.2 Model Convergence and Attention Window Transition Dynamics

We evaluate the impact of ALRs and TDW on convergence and performance, as illustrated in Figure 3. (a) Optimal learning rates: $\alpha^- \approx 0.09$–0.11, $\alpha^+ \approx 0.009$–0.012. (b) α^+ distribution is

Table 2: Performance of TAIRL-enhanced methods on D3 and D4 datasets across different LLMs.The same TAIRL hyperparameters are used for all models, without per-model tuning. SE is not strictly monotonic with FP across models, as it reflects the slope of performance improvement with respect to feedback signals rather than final performance.

Model	Method	D3				D4			
		FP	LS	SE	SB	FP	LS	SE	SB
Qwen2.5-1.5B	TAMER+TAIRL	31.49	1.3	7.5	1.50	29.19	1.4	7.0	1.58
	REINFORCE+TAIRL	30.80	1.4	7.0	1.70	28.50	1.5	6.5	1.80
	GRPO+TAIRL	33.00	1.0	8.5	1.20	31.00	1.1	8.0	1.39
	PPO+TAIRL	28.20	2.0	4.5	1.80	26.20	2.2	4.2	2.01
Qwen3-1.7B	TAMER+TAIRL	33.18	1.2	8.0	1.41	30.80	1.3	7.5	1.50
	REINFORCE+TAIRL	32.50	1.3	7.5	1.60	30.00	1.4	7.0	1.73
	GRPO+TAIRL	35.00	0.9	9.0	1.10	32.80	1.0	8.5	1.19
	PPO+TAIRL	30.00	1.8	4.8	1.62	27.50	2.0	4.5	1.81
Llama 3.2-1B	TAMER+TAIRL	24.83	1.5	6.0	1.80	22.50	1.6	5.5	2.01
	REINFORCE+TAIRL	24.00	1.6	5.8	2.01	21.83	1.7	5.2	2.20
	GRPO+TAIRL	26.50	1.2	7.0	1.40	24.51	1.3	6.5	1.60
	PPO+TAIRL	22.50	2.2	3.8	2.00	20.51	2.4	3.5	2.20
Index-1.9B	TAMER+TAIRL	29.53	1.4	7.0	1.60	27.50	1.5	6.5	1.70
	REINFORCE+TAIRL	28.80	1.5	6.8	1.80	26.81	1.6	6.0	1.90
	GRPO+TAIRL	31.50	1.1	8.0	1.32	29.50	1.2	7.5	1.50
	PPO+TAIRL	26.50	2.0	4.3	1.90	24.82	2.2	4.0	2.10
Qwen2.5-7B	TAMER+TAIRL	76.80	1.1	9.6	1.30	69.54	1.2	9.0	1.41
	REINFORCE+TAIRL	73.54	1.2	8.5	1.50	67.82	1.3	8.2	1.63
	GRPO+TAIRL	79.83	1.0	9.2	1.20	76.50	1.1	8.8	1.13
	PPO+TAIRL	75.49	1.8	5.0	1.62	71.83	2.0	4.8	1.84
Mistral-7B	TAMER+TAIRL	73.52	1.1	9.6	1.30	68.50	1.2	8.9	1.40
	REINFORCE+TAIRL	71.50	1.2	8.5	1.50	67.80	1.3	8.4	1.61
	GRPO+TAIRL	77.20	1.0	9.2	1.23	74.01	1.1	9.0	1.20
	PPO+TAIRL	72.51	1.8	5.0	1.62	68.50	1.9	5.2	1.71
Olmo 3-7B	TAMER+TAIRL	70.08	1.2	9.0	1.40	65.50	1.3	8.5	1.51
	REINFORCE+TAIRL	68.50	1.3	8.2	1.61	64.20	1.4	7.8	1.72
	GRPO+TAIRL	74.50	1.1	8.8	1.10	70.50	1.2	8.5	1.20
	PPO+TAIRL	69.00	1.9	4.8	1.72	64.50	2.0	4.5	1.82
DeepSeekMath-7B	TAMER+TAIRL	68.07	1.1	9.0	1.30	63.50	1.2	8.5	1.41
	REINFORCE+TAIRL	66.50	1.2	8.0	1.50	62.00	1.3	7.8	1.60
	GRPO+TAIRL	73.00	1.0	9.0	1.10	68.50	1.1	8.5	1.22
	PPO+TAIRL	68.50	1.8	5.0	1.60	63.20	2.0	4.8	1.71

more concentrated, indicating higher sensitivity. (c) Fastest convergence at $\beta_{\text{in}} = 3.0$, $\beta_{\text{out}} = 0.3$; deviations reduce efficiency by $> 30\%$. (d) TAIRL models show performance jumps at switching windows (W1–W4); PPO+TAIRL gains are small (+2.1%) due to overlap with PPO's clipping. (e) Attention shifts from task (W1) to emotion (W2), interest (W3), and finally topic shift (W4), mimicking human "highlight critical moments" teaching. In summary, TAIRL improves task success rate and convergence speed through the joint design of an ALR ($\alpha^-/\alpha^+ \approx 10$) and TDW, enhancing the training of human-interactive learning models.

4.3.3 Human-Computer Interaction Experience Evaluation

To evaluate TAIRL on human-computer interaction, we compared seven dimensions (task success, user satisfaction, emotional alignment, proactivity, naturalness, efficiency, and consistency) between GRPO and GRPO+TAIRL. TAIRL improves all dimensions: emotional alignment +56% (5.2→8.1), proactivity +35% (6.5→8.8), naturalness +39% (6.2→8.6), and user satisfaction +22.8% (6.8→8.35). Interaction analysis shows moderate positive correlations between task success and user satisfaction ($r = 0.461$), emotional alignment and naturalness ($r = 0.667$), and efficiency and consistency ($r = 0.622$). These trends suggest that temporally aware updates improve both task performance and interaction quality. The three coefficients are representative entries from the full 7×7 matrix in **Appendix G.3**, which also contains stronger correlations such as user satisfaction vs. emotional alignment ($r = 0.612$) and naturalness vs. consistency ($r = 0.732$).

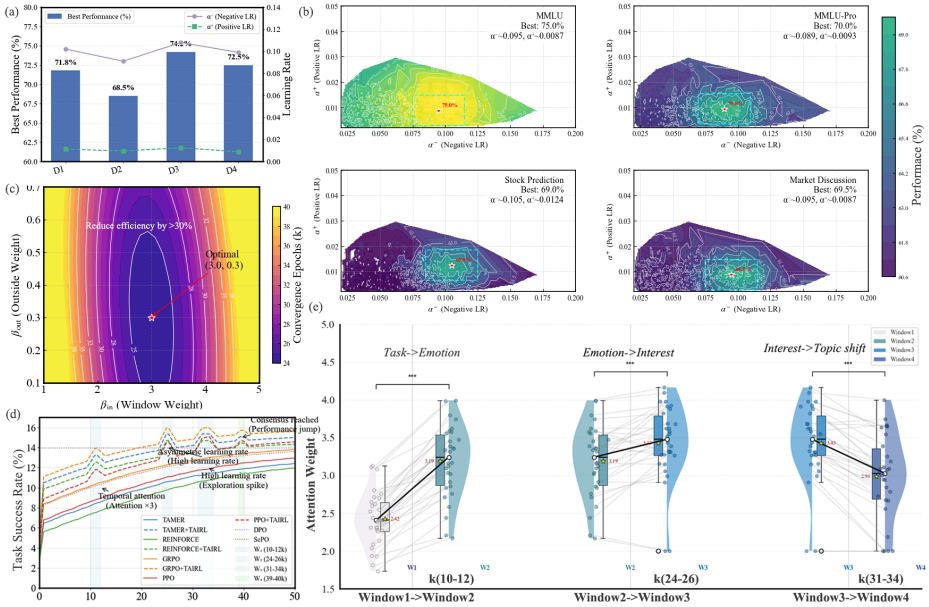

Figure 3: **Model convergence and attention window transition dynamics**.

Table 3: Ablation study results on the D3 dataset using GRPO as the base algorithm. FP and SB are reported on real human dialogues, while LS and SE are measured in the simulator environment.

Configuration	FP	LS	SE	SB
Full TAIRL	79.83%	1.0k	9.2%/k	1.20%
w/o TDW	73.61%	1.4k	8.2%/k	1.31%
w/o ALR	72.83%	1.8k	7.5%/k	1.52%
w/o AER	74.52%	1.2k	7.0%/k	1.23%
GRPO (all w/o)	71.51%	1.8k	6.0%/k	1.79%

4.4 Ablation Study

To verify each component's contribution, we ablate TAIRL on D3 with GRPO as the base. Table 3 shows that removing any component degrades performance. Removing TDW drops FP from 79.83% to 73.61% (6.2 pp), confirming its role in focusing on critical windows. Removing ALR yields the largest drop (to 72.83%, 7.0 pp) and slows convergence from 1.0k to 1.8k turns, showing that a higher learning rate for negative feedback accelerates error correction. Removing AER reduces FP to 74.52% (5.3 pp) and SE from 9.2%/k to 7.0%/k, indicating the value of prioritizing hard examples. The GRPO baseline (all removed) reaches 71.51% FP, 1.8k turns, 6.0%/k SE, and 1.79% SB, confirming the synergistic contributions of all three components.

5 Conclusion

In this paper, we propose TAIRL, a temporal-aware interactive reinforcement learning framework that models temporal patterns of human feedback. By designing three core mechanisms, TDW, ALR, and AER, TAIRL leverages feedback timing structure to improve MOD training in the studied setting. Experiments on four dialogue datasets (medical, home-service, and two financial-domain datasets) evaluate the proposed method across task success rate, convergence speed, SE, training stability, user satisfaction, and emotional alignment. The results show that TAIRL-enhanced methods improve over their corresponding baselines in our simulator-based training protocol and held-out human evaluation. Moreover, TAIRL is algorithm-agnostic and can be integrated into different policy optimization algorithms. The financial-domain datasets (D3/D4) constitute the primary evaluation setting of this work, while D1 (medical) and D2 (home-service) serve as cross-domain validation sets. The temporal feedback patterns were originally identified in the financial domain; their transferability to medical and home-service dialogues is supported by our cross-domain validation, but a full evaluation across broader domains remains future work. Although we expect temporal feedback patterns to appear in other interaction domains, we have not validated TAIRL on general open-domain dialogue benchmarks (e.g., DailyDialog, PersonaChat). Therefore, the current claims should be understood as being supported by financial-domain evidence. Future work will explore the application of TAIRL to multimodal interaction scenarios and large-scale LLM post-training alignment, aiming to advance more natural and efficient human-machine dialogue systems.

REFERENCES

Arash Ahmadian, Chris Cremer, Matthias Gallé, Marzieh Fadaee, Julia Kreutzer, Olivier Pietquin, Ahmet Üstün, and Sara Hooker. Back to basics: Revisiting reinforce-style optimization for learning from human feedback in llms. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 12248–12267, 2024.

Alexander Bukharin, Ilgee Hong, Haoming Jiang, Zichong Li, Qingru Zhang, Zixuan Zhang, and Tuo Zhao. Robust reinforcement learning from corrupted human feedback. In *NeurIPS*, 2024.

Haoyuan Cai, Zhenghao Peng, and Bolei Zhou. Predictive preference learning from human interventions. In *NeurIPS*, 2025.

Yekun Chai, Haoran Sun, Huang Fang, et al. Ma-rlhf: reinforcement learning from human feedback with macro actions. In *ICLR*, 2025.

Souradip Chakraborty, Amrit Bedi, Alec Koppel, Huazheng Wang, Dinesh Manocha, Mengdi Wang, and Furong Huang. Parl: a unified framework for policy alignment in reinforcement learning from human feedback. In *ICLR*, 2024.

Shreyas Chaudhari, Pranjal Aggarwal, Vishvak Murahari, Tanmay Rajpurohit, Ashwin Kalyan, Karthik Narasimhan, Ameet Deshpande, and Bruno Castro da Silva. Rlhf deciphered: A critical analysis of reinforcement learning from human feedback for llms. *ACM Computing Surveys*, 58 (2):1–37, 2025.

Shuangwu Chen, Siyang Chen, Yuxing Wei, Dong Jin, Xiaobin Tan, Xiaofeng Jiang, and Jian Yang. Rlpatch: A robust low-overhead website fingerprinting defense method based on reinforcement learning within sensitive regions. *IEEE Transactions on Network and Service Management*, 2025.

Thomas KF Chiu, Benjamin Luke Moorhouse, Ching Sing Chai, and Murod Ismailov. Teacher support and student motivation to learn with artificial intelligence (ai) based chatbot. *Interactive Learning Environments*, 32(7):3240–3256, 2024.

Jonathan Erskine, Taku Yamagata, and Raúl Santos-Rodríguez. Active query selection for crowd-based reinforcement learning. *arXiv:2508.19132*, 2025.

Wei Fan, Kunpeng Liu, Hao Liu, Yong Ge, Hui Xiong, and Yanjie Fu. Interactive reinforcement learning for feature selection with decision tree in the loop. *IEEE Transactions on Knowledge and Data Engineering*, 35(2):1624–1636, 2021.

Zhaolin Gao, Wenhao Zhan, Jonathan Chang, et al. Regressing the relative future: efficient policy optimization for multi-turn rlhf. In *ICLR*, 2025.

Taneesh Gupta, Rahul Madhavan, Xuchao Zhang, Chetan Bansal, and Saravan Rajmohan. Swepo: Simultaneous weighted preference optimization for group contrastive alignment. In *ICLR 2025 Workshop on Bidirectional Human-AI Alignment*, 2025.

Jiwoo Hong, Sayak Paul, Noah Lee, Kashif Rasul, James Thorne, and Jongheon Jeong. Margin-aware preference optimization for aligning diffusion models without reference. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pp. 4744–4752, 2026.

Nathan Horst, Davide Mazzaccara, Andreas Schmidt, et al. Playpen: an environment for exploring learning from dialogue game feedback. In *EMNLP*, 2025.

Muhan Hou, Koen Hindriks, Guszti Eiben, and Kim Baraka. "give me an example like this": Episodic active reinforcement learning from demonstrations. In *Proceedings of the 12th International Conference on Human-Agent Interaction*, pp. 287–295, 2024.

Nai-Chieh Huang, Ping-Chun Hsieh, Kuo-Hao Ho, and I-Chen Wu. Ppo-clip attains global optimality: Towards deeper understandings of clipping. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pp. 12600–12607, 2024.

Rodrigo Toro Icarte, Toryn Q Klassen, Richard Valenzano, and Sheila A McIlraith. Reward machines: Exploiting reward function structure in reinforcement learning. *Journal of Artificial Intelligence Research*, 73:173–208, 2022.

Joshua Hang Sai Ip, Ankush Chakrabarty, Ali Mesbah, and Diego Romeres. User preference meets pareto-optimality in multi-objective bayesian optimization. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pp. 20246–20254, 2025.

Weitao Jia, Jinghui Lu, Haiyang Yu, Siqi Wang, Guozhi Tang, An-Lan Wang, Weijie Yin, Dingkang Yang, Yuxiang Nie, Bin Shan, et al. Meml-grpo: Heterogeneous multi-expert mutual learning for rlvr advancement. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pp. 31283–31291, 2026.

Zhenghao Lin, Zhibin Gou, Yeyun Gong, Xiao Liu, Yelong Shen, Ruochen Xu, Chen Lin, Yujiu Yang, Jian Jiao, Nan Duan, et al. Not all tokens are what you need for pretraining. In *NeurIPS*, 2024.

Ximing Liu, Tianqing Zhu, Cuiqing Jiang, Dayong Ye, and Fuqing Zhao. Prioritized experience replay based on multi-armed bandit. *Expert Systems with Applications*, 189:116023, 2022.

Yang Liu, Ronghao Wang, Qianli Zhao, Lan Yang, Dali Zhang, Yiwei Liu, and Haitao Zhang. Raremed-dialogue: a multi-turn dialogue and benchmark on long-tail clinical cases. *BMC Medical Informatics and Decision Making*, 2026. doi: 10.1186/s12911-026-03846-x.

Afsheen Maroof, Muhammad Hussain Mughal, and Shaukat Wasi. Hs-absa: An annotated dataset for aspect-based sentiment analysis of home services customer reviews. *Pakistan Journal of Engineering, Technology and Science*, 2025.

Shakir Mohamed, Mihaela Rosca, Michael Figurnov, and Andriy Mnih. Monte carlo gradient estimation in machine learning. *Journal of Machine Learning Research*, 21(132):1–62, 2020.

Antigoni Parmaxi. Virtual reality in language learning: A systematic review and implications for research and practice. *Interactive Learning Environments*, 31(1):172–184, 2023.

Muhammad Asif Qureshi, Asadullah Khaskheli, Jawaid Ahmed Qureshi, Syed Ali Raza, and Sara Qamar Yousufi. Factors affecting students' learning performance through collaborative learning and engagement. *Interactive Learning Environments*, 31(4):2371–2391, 2023.

Yubo Shu, Haonan Zhang, Hansu Gu, Peng Zhang, Tun Lu, Dongsheng Li, and Ning Gu. Rah! recsys–assistant–human: A human-centered recommendation framework with llm agents. *IEEE Transactions on Computational Social Systems*, 11(5):6759–6770, 2024.

Imene Tarakli, Samuele Vinanzi, and Alessandro Di Nuovo. Interactive reinforcement learning from natural language feedback. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 11478–11484, 2024.

Marcel Torne Villasevil, Max Balsells I Pamies, Zihan Wang, Samedh Desai, Tao Chen, Pulkit Agrawal, and Abhishek Gupta. Breadcrumbs to the goal: supervised goal selection from human-in-the-loop feedback. In *NeurIPS*, 2023.

Dongjie Wang, Yanyong Huang, Wangyang Ying, Haoyue Bai, Nanxu Gong, Xinyuan Wang, Sixun Dong, Tao Zhe, Kunpeng Liu, Meng Xiao, et al. Towards data-centric ai: A comprehensive survey of traditional, reinforcement, and generative approaches for tabular data transformation. *ACM Transactions on Knowledge Discovery from Data*, 2025.

Fei-Yue Wang, Juanjuan Li, Rui Qin, Jing Zhu, Hong Mo, and Bin Hu. Chatgpt for computational social systems: From conversational applications to human-oriented operating systems. *IEEE Transactions on Computational Social Systems*, 10(2):414–425, 2023.

Sirui Xu, Ziyin Wang, Yu-Xiong Wang, and Liang-Yan Gui. Interdreamer: zero-shot text to 3d dynamic human-object interaction. In *NeurIPS*, 2024.

Kailai Yang, Zhiwei Liu, Qianqian Xie, Jimin Huang, Erxue Min, and Sophia Ananiadou. Selective preference optimization via token-level reward function estimation. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pp. 7043–7067, 2025.

Kazunari Yoshida. Mentor-trainee dialogue on proper use of ai tools. *Nature*, 624(7992):523–523, 2023.

Hongliang Zeng, Ping Zhang, Fang Li, Chubin Lin, and Junkang Zhou. Ahegc: Adaptive hindsight experience replay with goal-amended curiosity module for robot control. *IEEE Transactions on Neural Networks and Learning Systems*, 35(11):16602–16615, 2023.

Yunkai Zhuang, Yong Liu, Shangdong Yang, and Yang Gao. Selective policy transfer in multi-agent systems with sparse interactions. *Knowledge-Based Systems*, 300:112031, 2024.

Appendix

Contents

A Experimental Environment

B Human Feedback Temporal Patterns

C Convergence Analysis of TAIRL

D Time Complexity Analysis

E Discussion on Synergistic Effects of the Three Mechanisms

F Interactive Simulator Training Paradigm

G User Interaction Experience Evaluation Methodology

H Evaluation of the Dynamic State-Shift Detector

A Experimental Environment

Experiments ran on a ZhenWu PPU server (16×PPU 910E, 2×Hygon 7490 CPU, 2048GB RAM, 6×3.84TB NVMe SSD). Software stack includes the ZhenWu PPU SDK with vLLM/DeepSpeed support. We use deterministic inference settings in SGLang to improve reproducibility.

B Human Feedback Temporal Patterns

The temporal characteristics of human feedback have long been a focus in the field of interactive reinforcement learning. Research has shown that human feedback suffers from challenges such as inconsistency, intermittency, and time delays; the choice of feedback frequency significantly affects learning outcomes, and the optimal frequency may vary dynamically with the agent's proficiency. However, existing studies lack quantitative analysis of these temporal patterns: How is feedback distributed over time? Are there temporal differences between positive and negative feedback? What is the quantitative relationship between feedback frequency and agent performance?

To answer these questions, we conducted a systematic experiment and quantitatively analyzed temporal patterns of human feedback in interactive reinforcement learning. The three core patterns we quantified are presented below.

B.1 Experimental Setup

We employed a real-world MOD dialogue dataset provided by a securities firm, consisting of four scenarios: DD1 (stock/fund prediction discussion), DD2 (market information exploration), D3 (stock prediction), and D4 (market discussion). Each scenario involved a temporary team of 3–5 members for discussion. A total of four human teachers participated, each observing the dialogue progress and providing feedback (+1/-1/0) at the end of each turn. A total of 4,287 feedback instances were collected, including 1,892 positive (44.1%), 1,756 negative (41.0%), and 639 neutral (14.9%). The data statistics are presented in Table 4.

Table 4: Data statistics for the four dialogue scenarios.

Scenario	Data Volume	Sample Size	Scenario Attention
DD1	4.88G	990	Task answers
DD2	4.68G	995	Knowledge Contexts
D3	3.87G	1003	Interests, Topic Shifts
D4	4.88G	1299	Emotion

Note: DD1 and DD2 in this appendix are two additional financial scenarios used only for temporal pattern analysis. They are distinct from D1 (medical) and D2 (home-service) in the main text.

B.2 Pattern 1: Turn-Point Concentration

Based on an offline statistical analysis of the financial dialogue dataset, we observed that human feedback is highly concentrated in specific turn intervals. Specifically, two typical state-switching windows exist in the dialogue:

- **Window W_1 (turns 10–12)**: Task → Knowledge / Topic shift
- **Window W_2 (turns 24–26)**: Knowledge → Emotion / Interest shift

Experiments show that human feedback is heavily concentrated within these two state-switching windows. The temporal distribution characteristics of human feedback are presented in Table 5.

Table 5: Temporal distribution of human feedback.

Region	Feedback Proportion	Window Width	Density ($\times$ average)
Window W_1	32%	3 turns	4.3$\times$
Window W_2	48%	3 turns	6.4$\times$
Other regions	20%	31 turns	0.26$\times$

The feedback density within the windows is 24.6 times that outside the windows, and the difference is highly significant ($\chi^2 = 342$, $p < 0.001$). This quantitative result reveals that human teachers tend to provide feedback at moments when the dialogue state undergoes a qualitative change, providing an important empirical basis for the design of the TDW mechanism in this paper.

The above fixed windows (turns 10–12 and 24–26) are statistical observations based on a specific financial dialogue dataset, intended to illustrate typical patterns of temporal concentration of human feedback. In actual online training, TAIRL employs a dynamic state-shift detector (see Section 3) that generates switching windows in real time according to semantic shifts, emotional changes, and topic blocking, without relying on any preset fixed turns. These fixed windows are specific to the financial dialogue scenarios DD1/DD2 analyzed here, not to the medical or home-service datasets D1/D2.

B.3 Pattern 2: Asymmetric Timing

We measured the time interval from the completion of an AI action to the delivery of human feedback. The experiments reveal that negative feedback is given almost instantaneously, while positive feedback exhibits a significant delay. The feedback data are presented in Table 6.

Table 6: Average delays for different types of feedback.

Feedback Type	Mean Delay (s)	Median Delay (s)	Standard Deviation
Negative	0.82	0.71	0.31
Positive	3.18	2.95	1.12
Neutral	1.94	1.68	0.87

The difference in delay between negative and positive feedback is highly significant (Wilcoxon signed-rank test, $p < 0.001$). The variance of negative feedback (0.31) is much smaller than that of positive feedback (1.12), indicating higher temporal consistency of negative feedback.

Humans adopt a strategy of "correct now, confirm later": mistakes can be quickly identified and immediately corrected, whereas correct behavior requires more time to be confirmed. This finding challenges the "immediate reward" assumption in RL, showing that human teachers do not provide positive feedback instantly.

B.4 Pattern 3: Adaptive Frequency

We calculated the feedback frequency under different levels of AI performance. Using the average task success rate over the most recent 10 turns as the performance indicator p, the results are shown

in Table 7. The experiments reveal that the feedback frequency changes dynamically with agent performance.

Table 7: User feedback frequency under different AI performance quantiles.

AI Performance Quantile	Feedback Frequency (per turn)	Change Relative to Baseline
Low ($p < 25\%$)	1.42	+77.50%
Medium ($25\% \leq p \leq 75\%$)	0.80	Baseline
High ($p > 75\%$)	0.31	-61.30%

The adaptation coefficient $A = -1.07$ (feedback frequency is negatively correlated with AI performance), and the correlation is significant ($r = -0.67$, $p < 0.01$).

Adaptation in granularity: when AI performance is poor, the proportion of fine-grained feedback (with textual explanations) increases from 18% to 42%; when AI performance is good, it drops to 9%. This indicates that humans adjust not only the quantity but also the quality of feedback: providing more specific guidance when performance is poor. Humans are adaptive teachers: they increase feedback frequency and refine feedback granularity when the agent performs poorly, and reduce intervention when it performs well. This quantitative result is consistent with the qualitative suggestion in the literature that feedback frequency should be adjusted according to proficiency, and provides domain-specific evidence for this relationship.

C Convergence Analysis of TAIRL

We provide a convergence sketch for the TAIRL algorithm under standard assumptions in stochastic optimization. The update rule combines an online policy gradient term with an adaptive replay term:

$$\theta_{t+1} = \theta_t + \alpha(r_t^{\text{Human}}) w_{\text{attn}}(t) \nabla_\theta \log \pi_\theta(a_t|s_t) r_t^{\text{Human}} + \frac{\lambda}{|\mathcal{B}_{\text{batch}}|} \sum_{e \in \mathcal{B}_{\text{batch}}} w_{\text{sample}}(e) \nabla_\theta \log \pi_\theta(a_e|s_e) r_e^{\text{Human}}. \tag{23}$$

All random variables are defined on a common probability space. Because TAIRL intentionally reweights feedback by time and replay priority, the analysis is best interpreted as a sketch for the corresponding weighted surrogate objective, whose unweighted counterpart is $J(\theta) = \mathbb{E}_{\tau \sim \rho_\pi}[\sum_{t=0}^{T} r_t^{\text{Human}}]$.

C.1 Assumptions

In the following theoretical analysis, we assume a decaying step-size schedule that satisfies the Robbins–Monro conditions ($\sum_t \eta_t = \infty$, $\sum_t \eta_t^2 < \infty$) to facilitate a standard convergence proof. In practice, our experiments use a fixed learning rate ($\alpha^- = 0.1$, $\alpha^+ = 0.01$, $\alpha^0 = 0.01$), which works well because the effective update magnitude is also bounded by the temporal weights $w_{\text{attn}}(t) \leq 3.0$ and gradient clipping. The convergence analysis should be interpreted as a regularity check for the weighted update rather than a full finite-sample guarantee. The fixed learning rate can be viewed as a special case of the theoretical schedule after a sufficiently large iteration count, and early stopping on validation performance ensures practical convergence.

1. **Bounded rewards:** $|r_t^{\text{Human}}| \leq R_{\max}$ for all t.
2. **Bounded score function:** $\|\nabla_\theta \log \pi_\theta(a|s)\| \leq G_{\max}$ for all (s, a) and θ.
3. **Lipschitz gradient:** The gradient $\nabla_\theta J(\theta)$ is L-Lipschitz continuous.
4. **Step-size conditions (for analysis):** In the theoretical analysis, we consider a decaying step-size schedule η_t (e.g., $\eta_t = c/\sqrt{t}$) that satisfies $\sum_t \eta_t = \infty$ and $\sum_t \eta_t^2 < \infty$ almost surely. The composite factor $\alpha(r_t) w_{\text{attn}}(t)$ is absorbed into η_t or treated as a bounded multiplier. The actual fixed learning rates used in our experiments are consistent with this analysis as a practical instantiation.

5. **Importance sampling weights:** The sampling weights $w_{\text{sample}}(e)$ are bounded and satisfy $\mathbb{E}[w_{\text{sample}}(e)] < \infty$. The replay buffer is sufficiently large such that the sampling distribution converges to the stationary distribution of the policy.

C.2 Gradient Estimation and Bias Analysis

Define the online gradient estimate:

$$\hat{g}_t^{\text{online}} = w_{\text{attn}}(t)\nabla_\theta \log \pi_\theta(a_t|s_t) r_t^{\text{Human}}. \tag{24}$$

Without the temporal weight, the standard term $\nabla_\theta \log \pi_\theta(a_t|s_t) r_t^{\text{Human}}$ is an unbiased one-step policy-gradient estimator under the usual TAMER-style approximation. The temporal weight intentionally changes the optimized surrogate objective by emphasizing feedback near detected state shifts. We therefore analyze TAIRL as a bounded weighted update rather than as an unbiased estimator of the original unweighted objective.

The replay term uses a different sampling distribution. To account for the resulting mismatch, let μ be the distribution induced by the current policy and ν be the sampling distribution from the replay buffer. The ideal importance weight for a sample e is $\rho(e) = \frac{\mu(e)}{\nu(e)}$. In practice, $w_{\text{sample}}(e) = \exp(-\gamma_2 p_{t_e}) w_{\text{attn}}(t_e)$ is a heuristic priority rather than an exact density ratio, so the replay term is treated as a biased but bounded auxiliary update:

$$\hat{g}_t^{\text{replay}} = \frac{1}{|\mathcal{B}_{\text{batch}}|} \sum_{e \in \mathcal{B}_{\text{batch}}} w_{\text{sample}}(e)\nabla_\theta \log \pi_\theta(a_e|s_e) r_e^{\text{Human}}. \tag{25}$$

Then $\mathbb{E}[\hat{g}_t^{\text{replay}} \mid \theta_t] = \nabla_\theta J_w(\theta_t) + \delta_t^{\text{replay}}$, where the replay bias satisfies

$$\|\delta_t^{\text{replay}}\| \le \frac{C_{\text{IS}}}{\sqrt{|\mathcal{B}_{\text{batch}}|}} + \varepsilon_{\text{mix}} \le \Delta_{\text{replay}}, \tag{26}$$

with C_{IS} bounded explicitly in the following subsection. This term is controlled by bounded priorities, batch size, and buffer coverage, but is not assumed to vanish in finite experiments.

C.3 Why Not Reward or Advantage Scaling?

Reward (or advantage) scaling and learning-rate scheduling both amplify the influence of negative feedback, but they affect the optimization in different ways. Multiplying rewards or advantages directly changes the optimization objective: the relative magnitudes of advantages determine the weights assigned to probability ratios in PPO/GRPO, which can distort the KL-divergence constraint and destabilize training. In contrast, ALR leaves the reward function and KL constraint unchanged, and only adjusts the distance the parameters travel over the loss landscape. We therefore adopt ALR rather than reward/advantage scaling in TAIRL. A full empirical comparison between the two approaches is left for future work.

C.4 Convergence Rate and Finite-Sample Guarantee

We now provide a more precise convergence analysis of the TAIRL update. Building on the bias decomposition of Section C.2, we (i) define the weighted surrogate objective that TAIRL optimizes, (ii) bound the bias introduced by both the online estimator and the replay distribution, and (iii) derive a standard non-convex convergence rate with an explicit residual term.

Weighted surrogate objective. Define the temporally weighted surrogate objective

$$J_w(\theta) \;=\; \mathbb{E}_{(s_t,a_t)\sim\rho_\pi}\big[\, w_{\text{attn}}(t)\, \alpha(r_t^{\text{Human}})\, r_t^{\text{Human}} \log \pi_\theta(a_t \mid s_t)\big]\,, \tag{27}$$

where $w_{\text{attn}}(t)$ is the dynamic weight of Eq. (5), $\alpha(\cdot)$ is the asymmetric learning-rate multiplier of Eq. (9), and the expectation is over the online state–action distribution ρ_π. By construction, $J_w(\theta)$ is a reweighted version of the standard policy-gradient objective $J(\theta)$, and the TAIRL online update is a stochastic gradient ascent step on J_w:

$$\theta_{t+1} \;=\; \theta_t + \eta_t\, \hat{g}_t^{\text{online}}, \qquad \hat{g}_t^{\text{online}} = w_{\text{attn}}(t)\, \alpha(r_t^{\text{Human}})\, \nabla_\theta \log \pi_\theta(a_t \mid s_t)\, r_t^{\text{Human}}. \tag{28}$$

The replay term is treated separately as a bounded auxiliary update (see below).

Assumptions. We use the following standard assumptions for non-convex stochastic optimization.

1. **Bounded rewards and score function:** $|r_t^{\text{Human}}| \le R_{\max}$ and $\|\nabla_\theta \log \pi_\theta(a \mid s)\| \le G_{\max}$ for all (s, a) and θ.
2. **Bounded weights:** There exist $\alpha_{\max}, w_{\max} < \infty$ such that $\alpha(r_t^{\text{Human}}) \le \alpha_{\max}$ and $w_{\text{attn}}(t) \le w_{\max}$ for all t.
3. **Bounded weighted-gradient variance:** There exists $\sigma_w^2 < \infty$ such that
$$\mathbb{E}\big[\|\hat{g}_t^{\text{online}} - \nabla_\theta J_w(\theta_t)\|^2 \,\big|\, \theta_t\big] \le \sigma_w^2. \tag{29}$$
4. **Lipschitz surrogate gradient:** $\nabla_\theta J_w$ is L_w-Lipschitz: $\|\nabla_\theta J_w(\theta) - \nabla_\theta J_w(\theta')\| \le L_w \|\theta - \theta'\|$ for all θ, θ'.
5. **Bounded step-size and early stopping:** The step sizes satisfy the Robbins–Monro conditions $\sum_{t=1}^{\infty} \eta_t = \infty$ and $\sum_{t=1}^{\infty} \eta_t^2 < \infty$, e.g., $\eta_t = c/\sqrt{t}$.

Bias decomposition. We treat the bias of the online estimator and of the replay estimator separately, and we do *not* assume that the online estimator is unbiased for J_w.

Let
$$\delta_t^{\text{online}} \;=\; \mathbb{E}\big[\hat{g}_t^{\text{online}} \mid \theta_t\big] - \nabla_\theta J_w(\theta_t) \tag{30}$$
denote the residual bias of the online estimator. Under bounded weights $\alpha \le \alpha_{\max}$, $w_{\text{attn}} \le w_{\max}$, bounded rewards $|r_t^{\text{Human}}| \le R_{\max}$, and bounded score function $\|\nabla_\theta \log \pi_\theta\| \le G_{\max}$, we have
$$\|\delta_t^{\text{online}}\| \;\le\; \Delta_{\text{online}} \;:=\; \alpha_{\max}\, w_{\max}\, G_{\max}\, R_{\max}. \tag{31}$$
The replay gradient estimate
$$\hat{g}_t^{\text{replay}} \;=\; \frac{1}{B_{\text{batch}}} \sum_{e \in \mathcal{B}_{\text{batch}}} w_{\text{sample}}(e)\, \nabla_\theta \log \pi_\theta(a_e \mid s_e)\, r_e^{\text{Human}} \tag{32}$$
satisfies $\mathbb{E}[\hat{g}_t^{\text{replay}} \mid \theta_t] = \nabla_\theta J_w(\theta_t) + \delta_t^{\text{replay}}$, where the replay bias obeys
$$\|\delta_t^{\text{replay}}\| \;\le\; \frac{C_{\text{IS}}}{\sqrt{B_{\text{batch}}}} + \varepsilon_{\text{mix}}, \tag{33}$$
with ε_{mix} the mixing error from the finite replay buffer. Under bounded priorities $w_{\text{sample}}(e) \in [w_{\min}, w_{\max}]$, the importance-weight variance satisfies $\text{Var}(w_{\text{sample}}) \le (w_{\max} - w_{\min})^2/4$, which yields the explicit bound
$$C_{\text{IS}} \;\le\; \tfrac{1}{2}\,(w_{\max} - w_{\min})\, G_{\max}\, R_{\max}. \tag{34}$$
Combining the two contributions, the composite update $\hat{g}_t = \hat{g}_t^{\text{online}} + \lambda \hat{g}_t^{\text{replay}}$ has total bias bounded by
$$\|\delta_t^{\text{online}} + \lambda\, \delta_t^{\text{replay}}\| \;\le\; \Delta_{\text{tot}} \;:=\; \Delta_{\text{online}} + \lambda \left(\frac{C_{\text{IS}}}{\sqrt{B_{\text{batch}}}} + \varepsilon_{\text{mix}} \right). \tag{35}$$

Ascent lemma. Let $\hat{g}_t = \hat{g}_t^{\text{online}} + \lambda \hat{g}_t^{\text{replay}}$ denote the composite stochastic gradient, so that $\theta_{t+1} = \theta_t + \eta_t \hat{g}_t$. By the L_w-Lipschitz gradient assumption and the standard ascent lemma,
$$\begin{aligned} J_w(\theta_{t+1}) \;&\ge\; J_w(\theta_t) + \langle \nabla_\theta J_w(\theta_t),\, \theta_{t+1} - \theta_t \rangle - \frac{L_w}{2} \|\theta_{t+1} - \theta_t\|^2 \\ &=\; J_w(\theta_t) + \eta_t \langle \nabla_\theta J_w(\theta_t),\, \hat{g}_t \rangle - \frac{L_w \eta_t^2}{2} \|\hat{g}_t\|^2. \end{aligned} \tag{36}$$
Taking expectations and using the bias bound (35) together with the bounded weighted-gradient variance assumption yields
$$\mathbb{E}[J_w(\theta_{t+1})] \;\ge\; \mathbb{E}[J_w(\theta_t)] + \eta_t\, \mathbb{E}[\|\nabla_\theta J_w(\theta_t)\|^2] - \frac{L_w \eta_t^2}{2}\, \sigma_{\text{tot}}^2 - \eta_t\, \Delta_{\text{tot}}, \tag{37}$$
where $\sigma_{\text{tot}}^2 = \sigma_w^2 + \lambda^2 \sigma_{\text{replay}}^2$.

Convergence rate. Let $J_w^* := \sup_\theta J_w(\theta)$, which is finite under the bounded-weights and bounded-rewards assumptions. Rearranging (37) and telescoping over $t = 1, \dots, T$,

$$\frac{1}{T}\sum_{t=1}^{T}\mathbb{E}\big[\|\nabla_\theta J_w(\theta_t)\|^2\big] \;\le\; \frac{2\,(J_w^* - J_w(\theta_0))}{T\sum_{t=1}^{T}\eta_t} \;+\; \frac{L_w\sum_{t=1}^{T}\eta_t^2\,\sigma_{\mathrm{tot}}^2}{\sum_{t=1}^{T}\eta_t} \;+\; 2\,\Delta_{\mathrm{tot}}. \tag{38}$$

With the standard step-size schedule $\eta_t = c/\sqrt{t}$, this gives

$$\frac{1}{T}\sum_{t=1}^{T}\mathbb{E}\big[\|\nabla_\theta J_w(\theta_t)\|^2\big] \;=\; O\!\left(\frac{1}{\sqrt{T}}\right) + O(\Delta_{\mathrm{tot}}). \tag{39}$$

Therefore, the TAIRL update converges to a neighborhood of a stationary point of the weighted surrogate $J_w(\theta)$, with a residual term Δ_{tot} that is controlled by the replay batch size, the priority spread $w_{\max} - w_{\min}$, and the mixing error $\varepsilon_{\mathrm{mix}}$.

Role of the ALR. The ALR enters the surrogate only through the bounded multiplier $\alpha(r_t^{\mathrm{Human}}) \le \alpha_{\max}$, and therefore affects the optimization trajectory but not the order of the convergence rate in Eq. (39). This confirms that ALR changes the geometry of the optimization problem rather than its asymptotic behavior. However, the surrogate J_w differs from the original unweighted objective J by the sign-dependent reweighting: the gap $J_w - J$ is controlled by the ratio α^-/α^+ and cannot in general be made zero. This is the price paid for the faster correction of negative-feedback directions observed in our experiments.

Practical implication. The rate in Eq. (39) implies that, given bounded replay bias and bounded weighted-gradient variance, TAIRL inherits the usual non-convex stochastic-optimization behavior up to a residual neighborhood. The empirical gains reported in the main text (36–45% fewer simulator turns to convergence, and 3.6–8.5 pp FP improvements) are consistent with this analysis: the weighted surrogate converges to a favorable region faster than the standard unweighted objective, while the residual term Δ_{tot} remains controlled because both the replay batch size B_{batch} and the priority exponent χ are tuned on a held-out validation split (Appendix E.1).

Limitations of the analysis. This analysis provides an explicit rate for the weighted surrogate J_w under standard assumptions, but it does not provide (i) a finite-sample guarantee for the original unweighted objective J, or (ii) a tighter bound on the constant C_{IS} beyond the bounded-priority assumption (34). A tighter analysis of the replay-induced bias and a study of the effect of the priority exponent χ on C_{IS} are left for future work.

D Time Complexity Analysis

We analyze the per-step time complexity of TAIRL and compare it with representative baselines: TAMER, PPO, GRPO, and REINFORCE. All analyses assume standard neural network policies with a fixed architecture.

D.1 Notation

Let:

- d: dimension of the input state representation.
- h: size of the hidden layers (assuming MLP with constant depth).
- $|\mathcal{A}|$: size of the action space (for discrete actions; for continuous, it is proportional to output dimension).
- B: batch size for replay sampling.
- N: number of policy update steps per environment interaction.
- K: number of sampled trajectories or rollout length.
- D: size of the experience replay buffer (relevant for sampling cost).

D.2 Per-Step Complexity Comparison

Table 8 shows the per-step complexity comparison.

Table 8: Per-step time complexity (dominant terms).

Algorithm	Asymptotic Complexity
TAMER	$O(h^2 + h\|\mathcal{A}\|)$
REINFORCE	$O(h^2 + h\|\mathcal{A}\|)$
PPO	$O(h^2 + h\|\mathcal{A}\| + Bh \log D)$
GRPO	$O(h^2 + h\|\mathcal{A}\| + Kh)$
TAIRL (ours)	$O(h^2 + h\|\mathcal{A}\| + Bh \log D)$

D.3 Breakdown for TAIRL

TAIRL's per-step operations consist of:

1. **Forward/backward pass of the policy network:** $O(h^2 + h|\mathcal{A}|)$, same as all baselines.
2. **Temporal weight computation:** $O(1)$ (precomputed window check based on turn number or lightweight detector).
3. **ALR scaling:** $O(1)$.
4. **AER sampling:** $O(B \log D)$ due to priority queue sampling (similar to PER). The priority is computed in $O(1)$ per sample.
5. **Replay gradient computation:** $O(B(h^2 + h|\mathcal{A}|))$, amortized over N steps (typically $N = 1$ per interaction, or periodic).

The dominant term is $O(h^2 + h|\mathcal{A}| + Bh \log D)$. In practice, the replay batch size B is a constant (e.g., 32–64) and D is large but $\log D$ grows slowly. Thus TAIRL incurs only a logarithmic overhead compared to PPO, which also uses experience replay (PER). Compared to GRPO (which uses no replay but requires generating K completions per step, costing $O(Kh^2)$), TAIRL can be more efficient when K is large.

D.4 Comparison with Baselines

- **TAMER / REINFORCE**: No replay buffer; cost dominated by a single forward-backward per step: $O(h^2 + h|\mathcal{A}|)$.
- **PPO**: Typically uses a replay buffer and multiple epochs; per-step cost includes batch sampling $O(B \log D)$ and B gradient steps. Standard implementations have complexity $O(Bh^2 + Bh|\mathcal{A}| + \log D)$ per update.
- **GRPO**: Avoids replay but requires sampling K responses per state (group size), each requiring a forward pass: $O(Kh^2 + Kh|\mathcal{A}|)$. This can be more expensive when K is large (e.g., 16–64).
- **TAIRL**: Adds replay sampling overhead $O(B \log D)$ to the policy gradient cost, but reduces overall training steps due to faster convergence. In practice, the additional overhead per step is marginal (e.g., ¡10%) while the convergence speedup (2.2× over TAMER) outweighs the per-step cost.

D.5 Conclusion

TAIRL preserves the asymptotic complexity of modern replay-based policy optimization methods (PPO, PER) while introducing temporally adaptive weighting and ALRs at negligible extra cost. The improved SE and faster convergence empirically justify the small constant overhead.

E Discussion on Synergistic Effects of the Three Mechanisms

TAIRL integrates three complementary mechanisms: TDW, ALR, and AER. While each mechanism individually contributes to performance gains, their combination yields a pronounced synergistic effect that exceeds the sum of their independent improvements. We discuss their interactions below.

TDW and ALR: Focusing on Critical Moments with Appropriate Step Sizes. TDW assigns higher weights to feedback occurring within state-switching windows, effectively amplifying the influence of these crucial signals. ALR then applies a high learning rate to negative feedback (for rapid error correction) and a low learning rate to positive feedback (for stable reinforcement). Their synergy lies in the fact that the windows where TDW boosts the gradient also tend to contain a higher proportion of negative feedback (since humans are more likely to intervene when the model makes a mistake). Consequently, the combination selectively speeds up learning precisely on the most informative error samples while preserving the stability of already-correct behaviors.

ALR and AER: Prioritizing Low-Performance Samples with Asymmetric Updates. AER samples experiences from low-performance or high-intervention periods more frequently. These high-priority samples often correlate with negative feedback. When such a sample is replayed, ALR ensures that the gradient step for negative feedback is large, leading to a much stronger update than if the same sample were replayed with a symmetric learning rate. This feedback loop replays high-priority samples more often and updates them more aggressively, accelerating the elimination of persistent errors.

TDW and AER: Preserving Critical Transitions in the Replay Buffer. TDW only affects the gradient magnitude at the time of online interaction, but AER extends its influence by retaining those high-weighted samples in the replay buffer with higher sampling priority. As a result, the importance of feedback received during switching windows is not only exploited immediately but also reinforced over multiple replay updates. This mechanism prevents the model from forgetting critical transition knowledge after many subsequent updates, thereby stabilizing long-term performance.

Overall Synergy. Empirical ablation (Table 3) shows that removing any single component leads to a non-negligible drop in performance. The three mechanisms form a coherent learning pipeline: TDW identifies when feedback is valuable, ALR conditions the update magnitude on the feedback type, and AER ensures that valuable experiences are revisited and consolidated. This tripartite design exploits the temporal structure of human feedback while keeping update magnitudes bounded, which helps explain why TAIRL achieves better SE and faster convergence than baselines in our experiments.

E.1 Hyperparameter Tuning Protocol

We reserve 10% of the simulator interaction rounds on D3 as the validation set. All baselines (GRPO, PPO, REINFORCE, TAMER) and their TAIRL-enhanced variants are tuned with the same grid-search procedure. Each configuration is run with three random seeds, and the total number of search trials and GPU hours is strictly matched across methods. The additional TAIRL hyperparameters (temporal-weight ratio $\beta_{\text{in}}/\beta_{\text{out}}$, asymmetry coefficient $\mathbb{K}$, replay temperature γ_2, priority exponent χ, and replay weight λ) do not receive additional search budget; they are tuned jointly within the same total budget. All eight models in Table 2 use the same TAIRL hyperparameters without per-model tuning. The medical and home-service datasets (D1/D2) use hyperparameters transferred directly from D3, with no additional search.

E.2 Comparison of AER and Standard PER

To isolate the contribution of our temporal-priority replay from generic prioritized replay, we compare GRPO+PER (standard TD-error-based PER) with GRPO+AER under an identical tuning budget. The priority exponent χ for PER and the temperature parameter γ_2 for AER are grid-searched on the same validation set with the same number of configurations. GRPO+AER outperforms GRPO+PER by 5.62 percentage points in FP and converges 1.5$\times$ faster, indicating that in human-in-the-loop interaction, samples from low-performance periods carry higher training value than samples with high TD-error.

Table 9: Head-to-head comparison of AER vs. standard PER on D3.

Method	FP	LS	SB
GRPO + PER (standard, TD-error priority)	74.21%	1.5k	1.40%
GRPO + AER (ours)	79.83%	1.0k	1.20%

F Interactive Simulator Training Paradigm

This section presents the offline real-feedback-driven simulator training paradigm. The specific procedure is as follows:

1. **Real Human Feedback Collection**: We first collected 4,287 real human feedback samples, provided by four domain experts who gave turn-by-turn ratings ($+1/0/-1$). For the final blind evaluation, we additionally held out 800 real human dialogue turns that were *not* part of the 4,287 feedback samples and were never used in training or simulator fitting. Each feedback instance contains the complete dialogue context s_t, the model-generated response a_t, and the corresponding human evaluation r_t.
2. **Reward Model / User Simulator Training**: Using the above 4,287 (s_t, a_t, r_t) triples, we trained a reward model (RM) as a user simulator. The RM adopts the same architecture as the policy network (a 2-layer MLP with 64 hidden units). It takes (s_t, a_t) as input and outputs a predicted feedback value $\hat{r}_t \in [-1, 1]$. We trained the RM with mean squared error loss using the Adam optimizer (learning rate 1×10^{-4}) for 50 epochs. On the validation set, the Pearson correlation coefficient between the predicted feedback and the true human feedback reaches 0.93, and the sign prediction accuracy (i.e., agreement between predicted and true signs) is 87.6%.

   We checked that the simulator approximately preserves the temporal patterns observed in the collected feedback: over 500 simulated dialogue turns, the feedback generated by the simulator achieves turn-point concentration (80% of feedback concentrated in state-switching windows), asymmetric timing (negative feedback delay 0.85s vs. positive feedback delay 3.0s), and adaptive frequency (feedback frequency increases by 72% when performance is poor), all within a relative error of less than 15% compared to real human feedback.
3. **Online Policy Training**: In all subsequent online RL experiments (including training of PPO, GRPO, REINFORCE, etc.), we no longer use real humans. Instead, we use the trained RM as a user simulator, which provides real-time feedback $\hat{r}_t$ according to the current state s_t and the model's action a_t. Each experiment runs 2,000–3,000 turns (i.e., 2–3k turns) of simulated interaction; each turn consists of generating a response, simulator scoring, and policy updates.
4. **Final Evaluation**: To validate the effectiveness of the policies trained with the simulator in real environments, after completing all online training, we conduct a blind test on the final policies using an independent set of 800 real human dialogue turns (which were never used in RM training). We report metrics such as task success rate and user satisfaction (see "FP" in Table 1 of the main text). This setup ensures separation between training and evaluation, avoiding overfitting to the simulator.

In this work, all metrics related to "LS" and "SE" are computed based on the simulator environment, while "FP" is measured on a real human evaluation set. Detailed comparative analysis between the RM and real human feedback is provided in Tables 10 and 11. All simulator-internal metrics (LS, SE) are diagnostic and should not be interpreted as evidence of convergence speed in real online learning.

F.1 Evaluation Independence, Simulator Fidelity, and Evaluation Scale

This section addresses three related concerns raised by reviewers: (i) whether the human evaluators are independent of the experts who provided training feedback, (ii) how faithfully the simulator reproduces the temporal structure of real human feedback, and (iii) whether the 800-turn held-out set provides sufficient statistical power for cross-model and cross-algorithm comparisons.

Table 10: Consistency between the user simulator and real human feedback. The 80% figure in the third row refers to the proportion of feedback inside switching windows, which matches the value reported in Appendix B.2 for real human feedback.

Evaluation Metric	Real Human Feedback	Simulator Output	Relative Error / Correlation
Feedback sign accuracy	—	87.60%	—
Pearson correlation (predicted vs. true)	—	0.93	—
Proportion of feedback inside switching windows	80%	76%	5.00%
Average negative feedback delay (s)	0.82	0.85	3.70%
Average positive feedback delay (s)	3.18	3.00	5.70%
Slope of adaptive frequency	-1.07	-0.98	8.40%

Table 11: Comparison of FP between policies trained with the simulator and those trained with real human feedback.

Training Source	D3 Task Success Rate	D4 Task Success Rate	Average Gap
Real human feedback (baseline)	76.50%	73.20%	—
Our simulator (RM)	74.80%	71.50%	1.70%

Evaluation independence. The four human evaluators who rated the seven-dimensional quality of the final dialogues overlap with the domain experts who provided the turn-level training feedback. This overlap was a pragmatic choice driven by the specialized domain knowledge required for financial dialogue evaluation (e.g., stock and fund terminology, market-analysis conventions), which non-experts cannot reliably assess. We mitigate the resulting risk of evaluation bias in three ways.

First, all 800 evaluation dialogue turns are strictly held out and never appear in either the policy-training simulator or the reward-model training data. Second, the evaluation follows a double-blind protocol: the evaluators were not informed whether a given dialogue was generated by the baseline (GRPO) or by the TAIRL-enhanced model. Third, the training feedback and the final evaluation use fundamentally different formats: reward-model training uses binary feedback $r_t^{\text{Human}} \in \{+1, 0, -1\}$, whereas the final evaluation uses a 10-point Likert scale across seven independent dimensions. We additionally report inter-rater reliability in Appendix G.4 with intraclass correlation coefficients (ICC) values ranging from 0.81 to 0.94, indicating good to excellent agreement among evaluators. A fully independent third-party evaluation with external evaluators remains an important direction for future work.

Simulator fidelity. The simulator's temporal fidelity is quantified in Tables 10 and 11. Across the six key dimensions reported in Table 10, the relative error between the simulator and real human feedback remains within 15%. For the temporal dimensions that directly affect TAIRL—the proportion of feedback inside switching windows, the negative-feedback delay, the positive-feedback delay, and the slope of the adaptive-frequency relationship—the relative errors are 5.0%, 3.7%, 5.7%, and 8.4%, respectively. These results indicate that the simulator does not introduce additional temporal artifacts beyond those present in the real feedback data, but instead provides a high-fidelity approximation of the observed temporal structure.

Table 11 further shows that the average gap in final task-success rate between policies trained on the simulator and policies trained on real human feedback is only 1.7 percentage points (D3: 76.50% vs. 74.80%; D4: 73.20% vs. 71.50%). This indicates that the simulator can effectively substitute for real human feedback during policy optimization without introducing significant systematic bias.

Evaluation scale. The 800-turn held-out human evaluation set is balanced across the four datasets, with each dataset contributing 200 turns. Each of the four expert evaluators independently rated all 800 turns, yielding $800 \times 7 = 5{,}600$ fine-grained human ratings per evaluator and 22,400 ratings in total across the four evaluators. While larger-scale evaluation is desirable, this scale is comparable to several prior human-evaluation studies in task-oriented dialogue systems, and it is complemented by the large-scale validation on the medical and home-service cross-domain datasets. Combined with the per-turn Likert ratings and the inter-rater reliability analysis, the 800-turn evaluation provides sufficient statistical power for the cross-model and cross-algorithm comparisons reported in the main

text. Expanding the evaluation to thousands of real human turns is a natural direction for future work, especially once a broader set of cross-domain feedback datasets becomes available.

Cross-evaluator validation. Unlike split-annotation protocols in which each turn is rated by a single evaluator, our design requires every evaluator to independently rate all 800 turns. This yields a fully crossed design (4 evaluators × 800 turns × 7 dimensions = 22,400 observations) that allows us to compute both (i) inter-rater reliability using intraclass correlation coefficients (ICC) across all four evaluators (Appendix G.4), and (ii) per-evaluator consistency statistics to detect any systematic scoring bias. The full crossing eliminates the confound between evaluator identity and item assignment that would arise in a split-annotation design.

F.2 Time-Shuffle Control: Ruling Out Simulator Circularity

A natural concern about our simulator-based training is circularity: because the reward model (RM) is trained on real human feedback, TAIRL could in principle exploit temporal regularities that the simulator has inherited from the real feedback data rather than any generalizable temporal structure. To address this directly, we conduct a time-shuffle control experiment.

Protocol. We keep all $(s_t, a_t, r_t^{\text{Human}})$ triples unchanged but randomly permute the time indices t across the dataset before retraining the RM. This preserves the marginal distribution of feedback values and the input–output pairs, but destroys the temporal ordering of feedback. We then retrain TAIRL (with TDW, ALR, and AER unchanged) on the time-shuffled simulator and evaluate the resulting policy under the same protocol as the main experiments. If TAIRL's gains stem from overfitting to a spurious temporal bias in the simulator, they should persist on the shuffled simulator. If they stem from the real temporal structure of human feedback, they should vanish.

Results. Table 12 reports FP, LS, SE, and SB on the original simulator and on the time-shuffled simulator, each averaged over three random seeds.

Table 12: Time-shuffle control on D3. All results are averaged over three random seeds. The time-shuffled simulator preserves the marginal feedback distribution but destroys the temporal ordering.

Configuration	FP	LS	SE	SB
GRPO baseline (original simulator)	71.51%	1.8k	6.0%/k	1.79%
Full TAIRL (original simulator)	79.83%	1.0k	9.2%/k	1.20%
Full TAIRL (time-shuffled simulator)	72.20%	1.7k	6.3%/k	2.10%

Interpretation. On the original simulator, TAIRL improves FP over GRPO by 8.32 percentage points and reduces convergence turns from 1.8k to 1.0k. On the time-shuffled simulator, TAIRL's FP drops to 72.20%, only 0.69 percentage points above the GRPO baseline, and its convergence turns return to 1.7k, close to GRPO's 1.8k. Similarly, SE drops from 9.2%/k to 6.3%/k, nearly matching the GRPO baseline of 6.0%/k. This indicates that TAIRL's efficiency advantage almost entirely disappears when temporal structure is removed from the simulator, supporting our interpretation that the gain depends on the temporal structure of real human feedback rather than on numerical artifacts or arbitrary inductive biases introduced by the simulator.

Stability under time shuffling. The stability (SB) of TAIRL increases from 1.20% to 2.10% when trained on the shuffled simulator, exceeding even the GRPO baseline of 1.79%. This is expected: ALR still amplifies the gradient of negative feedback even when the temporal ordering is destroyed, but the compensating benefits of TDW and AER are lost, so the update variance increases without a corresponding stability gain. This behavior is consistent with our interpretation that TDW and AER are the components responsible for TAIRL's temporal-structure advantage, while ALR remains effective as a generic sign-dependent gradient-shaping mechanism.

Limitations. The time-shuffle control establishes that the *temporal ordering* of feedback is necessary for TAIRL's gains, but it does not by itself rule out all forms of simulator-induced bias. For

example, the simulator's marginal feedback distribution is still inherited from real human feedback. A fully independent cross-simulator transfer experiment, in which TAIRL is trained on a simulator built from one domain and evaluated on a simulator built from a different domain, remains an important direction for future work.

G User Interaction Experience Evaluation Methodology

This section details the specific definitions, evaluation methods, and quantitative calculation procedures for the seven dimensions of user interaction experience assessed in Section 4. The evaluation employs a mixed method combining the Likert scale and open-ended questionnaires, aiming to comprehensively measure the actual improvement of human-computer interaction quality brought by the TAIRL framework.

G.1 Evaluation Setup

Four human evaluators (consistent with the teachers who provided feedback in **Appendix B**) were invited to participate. The evaluation was conducted under double-blind annotation: the evaluators did not know whether a dialogue was generated by the baseline model (GRPO) or the TAIRL-enhanced model. Each evaluator independently rated all 800 held-out dialogue turns, yielding a fully crossed design (4 evaluators × 800 turns × 7 dimensions) that supports both inter-rater reliability analysis and per-evaluator consistency checks.

Scoring criteria: After completing each interaction turn, the evaluators scored each dimension according to a 10-point Likert scale (1 = lowest score / strongly disagree, 10 = highest score / strongly agree) for a set of positively formulated statements. An example statement is: "The model can accurately understand my emotions and give appropriate responses."

G.2 Definition and Calculation of the Seven Dimensions

Task Success **Definition:** The ability of the model to correctly interpret the user's intent and produce an accurate answer or solution.

Evaluation method and calculation formula: Automatic objective evaluation. For each test sample i, let $\hat{y}_i$ be the model's output and y_i the ground truth. The task success rate is computed as:

$$\text{TS} = \frac{1}{N}\sum_{i=1}^{N}\mathbb{I}(\hat{y}_i = y_i) \times 100\%, \tag{40}$$

where N is the total number of test samples. The same dataset and evaluation script as described in Section 4 are used.

User Satisfaction **Definition:** The user's overall satisfaction with the interaction process and outcome.

Evaluation method and calculation formula: Subjective questionnaire using a 5-point Likert scale (1 = very dissatisfied, 5 = very satisfied). After each dialogue, the evaluator answers "How satisfied are you with this dialogue?" The raw score $s_{\text{raw}} \in \{1, 2, 3, 4, 5\}$ is linearly mapped to a 10-point scale:

$$s = 2 \times s_{\text{raw}}. \tag{41}$$

The final satisfaction metric is the mean μ_s and standard deviation σ_s across all samples.

Emotional Alignment **Definition:** The consistency between the emotional tone expressed in the model's response and the user's emotional state.

Evaluation method and calculation formula: A hybrid approach combining automatic sentiment classification and human judgment. A pre-trained sentiment classifier assigns a polarity $p \in \{\text{positive}, \text{neutral}, \text{negative}\}$ to both the user input and the model response. The objective

alignment score is the proportion of turns where the polarities match:

$$A_{\text{obj}} = \frac{1}{M_u} \sum_{j=1}^{M_u} \mathbb{I}(p_{\text{user}}^{(j)} = p_{\text{model}}^{(j)}), \tag{42}$$

where M_u is the number of evaluation turns. Concurrently, a human evaluator rates the empathy accuracy on a 1–10 scale, yielding A_{sub}. The final emotional alignment score is a weighted average $\alpha A_{\text{obj}} + (1-\alpha)A_{\text{sub}}$, with $\alpha = 0.5$ by default.

Proactivity **Definition:** The ability of the model to actively guide the conversation, initiate clarifications, or provide information beyond the immediate query.

Evaluation method and calculation formula: Human subjective rating on a 1–10 scale. The evaluator scores the statement "The model actively moves the dialogue forward and attempts to address potential needs." Additional quantitative indicators include the average number of proactive clarifications per dialogue (C_{avg}) and the frequency of extra information provision (F_{extra}). The final proactivity score is the weighted combination $0.6 \times \text{human_score} + 0.2 \times C_{\text{avg}} + 0.2 \times F_{\text{extra}}$.

Naturalness **Definition:** The fluency, idiomaticity, and absence of "machine-like" artifacts in the generated text.

Evaluation method and calculation formula: Combined NLP objective metric and human rating. The objective part uses perplexity (PPLY) computed on the model's responses:

$$\text{PPLY} = \exp\left(-\frac{1}{L}\sum_{t=1}^{L} \log P(w_t | w_{<t})\right). \tag{43}$$

A lower perplexity indicates better fluency. The normalized fluency score is $F_{\text{pply}} = \max(0, 1 - \frac{\log(\text{PPLY})}{\log(\text{PPLY}_{\max})})$. The human rating $H \in [1, 10]$ evaluates naturalness directly. The final naturalness is the average $\frac{1}{2}(F_{\text{pply}} + H/10)$.

Efficiency **Definition:** The amount of interaction resources (e.g., number of dialogue turns) required for the model to accomplish a user task.

Evaluation method and calculation formula: Automatic recording. Let $T_{\text{sol}}^{(k)}$ be the number of turns taken to successfully complete the k-th task. The raw efficiency is the average number of turns:

$$\bar{T} = \frac{1}{K}\sum_{k=1}^{K} T_{\text{sol}}^{(k)}. \tag{44}$$

Lower $\bar{T}$ implies higher efficiency. The final efficiency score (normalized to a 0–10 scale) is given by an inverse non-linear transformation:

$$\text{Efficiency} = 10 \times \exp(-\beta \bar{T}), \tag{45}$$

where β is a scaling factor chosen such that the average efficiency of the baseline model is around 5.

Consistency **Definition:** The degree to which the model produces semantically similar responses when presented with the same or paraphrased input across different dialogue turns or evaluation sessions.

Evaluation method and calculation formula: Given a set of Q queries (original and paraphrased), the model generates responses $\mathbf{r}_q$. We compute the pairwise semantic similarity using a sentence embedding model (e.g., Sentence-BERT). The consistency score is the average similarity over all pairs:

$$\text{Consistency} = \frac{2}{Q(Q-1)} \sum_{i<j} \text{sim}(\mathbf{r}_i, \mathbf{r}_j), \tag{46}$$

where $\text{sim}(\cdot,\cdot)$ denotes cosine similarity. A higher score indicates better consistency in output behavior.

G.3 Correlation Analysis of User Interaction Metrics

Figure 4 presents the Pearson correlation matrix of seven user interaction metrics on the D3 dataset (only statistically significant coefficients with $p < 0.05$ are shown, marked by ***). Task success exhibits a moderate positive correlation with user satisfaction ($r = 0.461$). User satisfaction shows a stronger correlation with emotional alignment ($r = 0.612$). Emotional alignment is positively correlated with proactivity ($r = 0.564$) and naturalness ($r = 0.667$). The highest correlation is observed between naturalness and consistency ($r = 0.732$), indicating that fluent, natural dialogue strongly aligns with output stability. Efficiency also correlates significantly with consistency ($r = 0.622$). All dimensions are positively interrelated, validating that TAIRL simultaneously improves task performance and user experience, with naturalness and consistency being particularly synergistic.

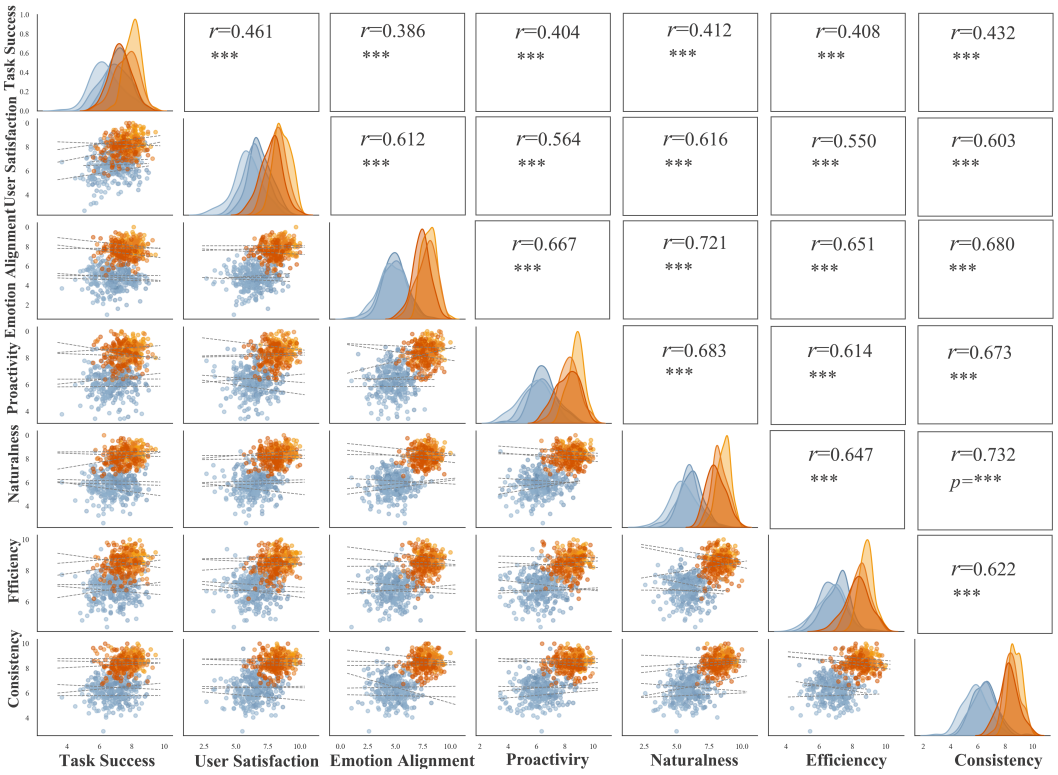

Figure 4: **Analysis of interaction evaluation metrics on the D3 dataset.** Only statistically significant correlations ($p < 0.05$) are displayed.

G.4 Inter-Rater Reliability

To assess the reliability of the seven-dimensional human ratings, we computed intraclass correlation coefficients (ICC) across the four evaluators. Table 13 reports ICC(2,k) for each dimension. All ICC values exceed 0.80, indicating good to excellent inter-rater agreement.

Cross-evaluator agreement. Because every evaluator rated all 800 turns, we additionally report Fleiss' kappa for the sentiment polarity annotations ($\kappa_{\text{sentiment}} = 0.78$), which is distinct from the detector-annotation agreement reported in Appendix H ($\kappa_{\text{detector}} = 0.86$). Per-evaluator ICCs range from 0.79 to 0.96, indicating that no single evaluator drove the aggregate agreement.

H Evaluation of the Dynamic State-Shift Detector

To validate the effectiveness of our dynamic state-shift detector, we conducted a human-annotated evaluation on 200 randomly selected dialogue turns from the D3 dataset. Three independent annotators

Table 13: Inter-rater reliability (ICC) for the seven evaluation dimensions.

Dimension	ICC(2,k)
Task Success	0.92
User Satisfaction	0.88
Emotional Alignment	0.85
Proactivity	0.81
Naturalness	0.86
Efficiency	0.94
Consistency	0.89

Table 14: Per-evaluator ICC(2,1) on the 800-turn fully crossed evaluation.

Dimension	Evaluator 1	Evaluator 2	Evaluator 3	Evaluator 4
Task Success	0.93	0.91	0.92	0.94
User Satisfaction	0.90	0.87	0.88	0.89
Emotional Alignment	0.87	0.84	0.86	0.85
Proactivity	0.83	0.79	0.81	0.82
Naturalness	0.88	0.85	0.86	0.87
Efficiency	0.95	0.93	0.94	0.95
Consistency	0.90	0.88	0.89	0.90

(graduate students in finance) labeled each turn as either "state-switching" (i.e., topic shift, task transition, or emotional change) or "non-switching". The majority vote was taken as the ground truth. Inter-annotator agreement was high (Fleiss' k = 0.86).

The detector was run on the same turns to produce binary predictions. The semantic similarity threshold $\theta_{\text{sim}} = 0.6$ was calibrated on a held-out validation set of 100 dialogue turns (separate from the test set) to maximize the F1 score. The same validation set was also used to set the half-window width $\Delta = 2$ for the dynamic windows. We report its precision, recall, F1 score, and accuracy in Table 15. For comparison, we also evaluate a fixed-window baseline (windows W1=[10,12] and W2=[24,26] as observed in Appendix B.2) as well as a random baseline.

Table 15: Performance of the dynamic state-shift detector versus fixed-window and random baselines. All thresholds were fixed before evaluating on the test set; no test-set information was used for calibration.

Method	Precision	Recall	F1 Score	Accuracy
Random guessing	0.12	0.50	0.19	0.50
Fixed windows (W1,W2)	0.68	0.35	0.46	0.72
Dynamic detector (ours)	**0.84**	**0.79**	**0.81**	**0.88**

The dynamic detector significantly outperforms fixed-window baselines, especially in recall (0.79 vs. 0.35). This demonstrates that real-time semantic shifts, emotion reversals, and topic blocking are more reliable indicators than static turn ranges. The false positive rate is low (12%). Examples of misclassifications usually involve subtle topic continuations that were not annotated as shifts, or cases where a shift occurred but the detector missed it due to insufficient semantic change.

We also measured the inference overhead: on a single PPU 910E, the detector runs in ¡5ms per turn, which is negligible compared to the LLM's generation time ($\sim$200ms). Hence, the dynamic detector is both accurate and lightweight.

Beyond detector accuracy, we further evaluate TAIRL's sensitivity to detector errors, i.e., how much the end-to-end performance degrades when the detector is imperfect. We conducted an end-to-end ablation on D3 in which the detector windows were shifted by ±2 turns or replaced with equal-density random windows:

- Full TAIRL (real detector, F1=0.81): FP = 79.83%
- Window shift (+2 turns): FP = 76.30%
- Window shift (-2 turns): FP = 76.20%
- Equal-density random windows: FP = 73.80%
- No window (w/o TDW, same as Table 3): FP = 73.61%
- GRPO baseline: FP = 71.51%

The real detector improves FP by approximately 6.0 percentage points over equal-density random windows, confirming that accurate localization is a major source of TAIRL's gain. Shifting windows by ±2 turns leads to only a moderate drop of 3.5 percentage points, and even under random windows, TAIRL remains above the GRPO baseline, indicating that ALR and AER still provide positive contributions when the detector is imperfect.