Multi-turn Reinforcement Learning
from Preference Human Feedback

Lior Shani [* 1]
liorshani@google.com

Aviv Rosenberg [* 1]
avivros@google.com

Asaf Cassel [* 1 3]
acassel@mail.tau.ac.il

Oran Lang [1]   **Daniele Calandriello** [2]   **Avital Zipori** [1]   **Hila Noga** [1]   **Orgad Keller** [1]

Bilal Piot [2]   **Idan Szpektor** [1]   **Avinatan Hassidim** [1]   **Yossi Matias** [1]   **Rémi Munos** [2]

Abstract

Reinforcement Learning from Human Feedback (RLHF) has become the standard approach for aligning Large Language Models (LLMs) with human preferences, allowing LLMs to demonstrate remarkable abilities in various tasks. Existing methods work by emulating the preferences at the single decision (turn) level, limiting their capabilities in settings that require planning or multi-turn interactions to achieve a long-term goal. In this paper, we address this issue by developing novel methods for Reinforcement Learning (RL) from preference feedback between two full multi-turn conversations. In the tabular setting, we present a novel mirror-descent-based policy optimization algorithm for the general multi-turn preference-based RL problem, and prove its convergence to Nash equilibrium. To evaluate performance, we create a new environment, Education Dialogue, where a teacher agent guides a student in learning a random topic, and show that a deep RL variant of our algorithm outperforms RLHF baselines. Finally, we show that in an environment with explicit rewards, our algorithm recovers the same performance as a reward-based RL baseline, despite relying solely on a weaker preference signal.

1 Introduction

A pinnacle of human intelligence is the ability to communicate with an environment, forming complex interactions to accomplish challenging goals. Dialogues are one example of such dynamic communication, where one party reacts to signals from the other parties and dynamically plans ahead to steer communication towards their purpose. Recent years have seen scientific breakthroughs in developing Large Language Models (LLMs) that can communicate with humans in natural language [Ouyang et al., 2022, Anil et al., 2023, Touvron et al., 2023, OpenAI, 2024, Google, 2024]. In order to align these models to human needs, many efforts have been made to train them with a given human feedback. In some concordance with human learning, this is usually achieved by reinforcing behaviors that align with the feedback, using a technique now called Reinforcement Learning from Human Feedback (RLHF; Christiano et al. [2017], Ziegler et al. [2019], Stiennon et al. [2020]).

RLHF methods build on the long studied field of Reinforcement Learning (RL), which focuses on learning optimal actions through reward feedback (a numerical signal) from the environment. However, defining a suitable reward function is challenging, leading to the common practice of collecting human preferences between choices. In the absence of rewards, a mapping from preference to reward is typically assumed in the form of the Bradely-Terry (BT; Bradley and Terry [1952]) model [Stiennon et al., 2020, Rafailov et al., 2023], enabling the use of a wide-variety of well-researched

---

[*] Equal contribution. [1] Google Research. [2] Google DeepMind. [3] Tel Aviv University.

RL techniques. Alternatively, recent research [Munos et al., 2023, Azar et al., 2023, Tang et al., 2024] suggests a more direct use of preferences for learning, eliminating the need for this potentially limiting assumption.

Still, so far the main focus of both the RLHF and the direct preference learning literature was on single-turn scenarios, where given relevant context, the LLM generates one response and receives an immediate feedback that reflects its alignment quality. Importantly, while single-turn RLHF already provides significant gains for valuable AI systems, it lacks the adaptive and long-term capabilities that make human communication such a powerful tool, and usually characterize RL methods. This is especially apparent in temporally extended tasks, such as multi-turn dialogue [Irvine et al., 2023], complex tool use [Wang et al., 2022] and multi-step games [Hendrycks et al., 2022].

Contributions. In this work, we focus on improving the communication of AI agents with dynamic environments. To this end, we first extend the RLHF paradigm to the multi-turn setting, where the agent has a series of exchanges with an external (stochastic) environment (Section 3). Importantly, we consider (human) feedback that compares entire multi-turn conversations as opposed to single-turn scenarios, which compare individual actions on a per-turn basis. Conversation-level feedback allows to capture the long-term effect of individual actions, which may not be immediately apparent, and thus hard to define through turn-level feedback. For example, a seller agent asking too high a price may seem immediately bad, but becomes potentially good as part of a complete strategy to increase sale price. This difference is apparent in our preference model, making it better suited for multi-turn interactions.

Formalizing the multi-turn setting as a Contextual Markov Decision Process with end of interaction preference feedback, we devise several theoretically grounded algorithms (Section 4). Our main algorithm, Multi-turn Preference Optimization (MTPO), is a new policy optimization algorithm for the general multi-turn preference-based setting. MTPO is based on the Mirror Descent (MD) method [Nemirovskij and Yudin, 1983, Beck and Teboulle, 2003] together with self-play [Silver et al., 2017], and is proven to converge to a Nash equilibrium [Nash et al., 1950], i.e., a policy which is preferred over any other policy. We prove similar results for MTPO$-\tau$, a slight variant of MTPO that, similarly to Munos et al. [2023], uses a geometric mixture policy which interpolates the agent's policy with a fixed reference policy (with mixing rate τ). These algorithms utilize a new form of preference-based Q-function that accounts for the long-term consequences of individual actions. Finally, leveraging our theoretical framework, we modify this Q-function to create a multi-turn RLHF algorithm and prove its convergence to an optimal policy (w.r.t the learned reward function).

We complement our theoretical findings with a policy-gradient version of our multi-turn algorithms for deep learning architectures (Section 4.1). To validate our approach, we apply our algorithms to train a T5 encoder-decoder LLM [Raffel et al., 2020], aiming to enhance its multi-turn dialogue abilities (Sections 5 and 6). We test our approach in a scenario without explicit rewards, where conversation quality is evaluated solely through preferences. To that end, we create a new environment called Education Dialogue, where a teacher guides a student in learning a random topic, by prompting Gemini [Team et al., 2023]. The conversation is judged based on preference feedback, using a constitution that defines effective learning [Bai et al., 2022, Lee et al., 2023] (see Section 5). In this environment, our multi-turn algorithms significantly outperform single-turn baselines, and our direct multi-turn preference approach outperforms multi-turn RLHF (Section 6). As an additional contribution, we publicly release the data of Education Dialogue.[1] Finally, we demonstrate that even in a reward-based environment, our preference-based algorithm achieves comparable performance to learning directly from rewards, as in standard RL, despite using a weaker signal. For this experiment, we utilize the LMRL-Gym [Abdulhai et al., 2023] Car Dealer environment, simulating a conversation where the agent (car dealer) aims to maximize the sale price.

Related work. Most related to our work is the RLHF literature, which aims to improve an LLM policy using preference data collected from humans. Earlier methods model a proxy reward [Ouyang et al., 2022] or preference function [Zhao et al., 2022], and apply traditional RL techniques. More recent methods directly optimize the policy [Rafailov et al., 2023, Azar et al., 2023, Tang et al., 2024, Song et al., 2024, Ethayarajh et al., 2024]. Another line of work, which forms the basis for MTPO, extends RLHF to games, aiming to compute a Nash equilibrium instead of an optimal policy w.r.t a fixed reward/preference. This includes Nash-MD [Munos et al., 2023], self-play and mixtures of the

---

[1] https://github.com/google-research-datasets/Education-Dialogue-Dataset

two like IPO-MD [Calandriello et al., 2024]. Nonetheless, these methods only consider single-turn problems, whereas MTPO provides the first guarantees for multi-turn settings. Note the difference from concurrent attempts to extend direct preference optimization to the token level [Rafailov et al., 2024], while a true multi-turn approach must deal with the additional uncontrollable tokens generated by the human in-between agent turns.

More broadly, preference-based RL (see survey by Wirth et al. [2017]) studies feedback in terms of preferences over two alternatives rather than absolute rewards. Feedback can be provided in various ways, e.g., at the level of states (turn-level), or entire trajectories [Chen et al., 2022, Saha et al., 2023, Wu and Sun, 2023, Wang et al., 2023, Zhan et al., 2023a,b]. The focus of this work is last state feedback which is an instance of trajectory feedback. Another closely related model is RL with aggregate feedback where only the sum of rewards in a trajectory is revealed to the agent [Efroni et al., 2021, Cohen et al., 2021, Chatterji et al., 2021, Cassel et al., 2024].

Lastly, there is vast literature on using RL to improve natural language generation for dialogue systems. The pioneering work of Li et al. [2016] focuses on designing rewards to capture important dialogue attributes such as semantic coherence and ease of answering. Other works tackle task-oriented dialogue, using RL to enhance the agent's ability to solve the underlying dialogue task [Wei et al., 2018]. Instead, this works takes a more general and fundamental approach, focusing on the algorithmic process of aligning an agent which repeatedly interacts with an environment. While dialogue systems are a promising application of our approach, as suggested by the experimental results of this paper (Sections 5 and 6), our algorithmic approach is much broader, including processes such as tool-use, reasoning, and many other applications that require aligning a complex multi-turn agent with human preferences.

2 Preliminaries

The interaction between an AI agent and its environment is captured in the fundamental contextual RL model, where the context is the initial prompt, the states are conversation summaries, and the actions are responses.

Contextual Markov decision process. A finite-horizon contextual Markov decision process (CMDP) $\mathcal{M}$ is defined by a tuple $(\mathcal{C}, \mathcal{X}, \mathcal{Y}, H, x_1, \rho_c, p)$ where $\mathcal{C}$ is the context space, $\mathcal{X}$ is the state space, $\mathcal{Y}$ is the action space, H is the horizon, $x_1 \in \mathcal{X}$ is the initial state, $\rho_c \in \Delta_{\mathcal{C}}$ is the context distribution and $p : \mathcal{C} \times \mathcal{X} \times \mathcal{Y} \to \Delta_{\mathcal{X}}$ is the transition function such that $p(x' \mid c, x, y)$ is the probability to transition to state x' after taking action y in state x, given context c.

An interaction between the agent and the CMDP environment proceeds in H steps. First, a context $c \in \mathcal{C}$ is sampled from ρ_c, and then the agent begins in the initial state x_1. In step $h \in [H]$, the agent observes the current state $x_h \in \mathcal{X}$, picks an action $y_h \in \mathcal{Y}$ and transitions to the next state x_{h+1} sampled from the transition function $p(\cdot \mid c, x_h, y_h)$. At the end of the interaction, the agent arrives in a final state x_{H+1}. For simplicity, we assume that the state space can be decomposed into $H + 1$ disjoint subsets $\mathcal{X} = \cup_{h=1}^{H+1} X_h$ such that, in step h of the interaction, the agent is in some state $x_h \in \mathcal{X}_h$. A policy $\pi : \mathcal{C} \times \mathcal{X} \to \Delta_{\mathcal{Y}}$ is a mapping from a context and state to a distribution over actions. Together with transition p, π induces a distribution over trajectories denoted by $\mathrm{Pr}_{\pi,p}[\cdot]$ (and $\mathbb{E}_{\pi,p}[\cdot]$ for the expectation), in which the trajectory is generated by sampling the actions according to the policy and next states according to the environment.

2.1 Single-turn Reinforcement Learning from Human Feedback

Unlike standard RL where the agent observes reward feedback for its actions, the influential work of Christiano et al. [2017] suggests to leverage preference data. In the single-turn setting, the agent generates a single sequence $y \in \mathcal{Y}$ given a context $c \in \mathcal{C}$. This is modeled as a Contextual Multi-Armed Bandit (CMAB), which is a CMDP instance with horizon $H = 1$. The feedback is given in the form of preference between two generated sequences. Formally, there exists a preference model $\mathcal{P} : \mathcal{C} \times \mathcal{Y} \times \mathcal{Y} \to [0,1]$ such that $\mathcal{P}(y \succ y' \mid c)$ gives the probability that y is preferred over y' given context c. Preferences naturally extend to policies via expectation $\mathcal{P}(\pi \succ \pi' \mid c) = \mathbb{E}_{y \sim \pi(\cdot|c), y' \sim \pi'(\cdot|c)}[\mathcal{P}(y \succ y' \mid c)]$.

Reinforcement Learning from Human Feedback. In RLHF, it is assumed that there is a hidden reward function $r : \mathcal{C} \times \mathcal{Y} \to \mathbb{R}$ that defines the preferences through the Bradely-Terry (BT) model, i.e., $\mathcal{P}(y > y' \mid c) = \sigma(r(c,y) - r(c,y'))$, where σ is the sigmoid. To reconstruct the reward, the RL algorithm is used to optimize the ELO score of the chosen action y using a cross-entropy loss. This technique was adapted to RL fine-tuning LLMs [Ziegler et al., 2019], and has become the standard approach for aligning LLMs to human feedback [Stiennon et al., 2020, Rafailov et al., 2023].

Learning from direct preferences. Recently Munos et al. [2023], Azar et al. [2023] suggested to drop the BT assumption, and learn a direct preference model instead of reward. Munos et al. [2023] propose the Nash-MD algorithm which converges to the Nash equilibrium of a (regularized) preference model, i.e., a policy which is preferred over any other policy. In iteration $t + 1$, Nash-MD updates its policy π_{t+1} using a mirror descent (MD) step projected to a geometric mixture policy. The mixture policy $\pi_t^\alpha(\cdot \mid c) \propto \pi_t(\cdot \mid c)^{1-\alpha\eta_t} \mu(\cdot \mid c)^{\alpha\eta_t}$ interpolates between the policy π_t and a reference policy μ, given a regularization coefficient $\alpha > 0$. Formally, for learning rate $\eta_t > 0$,

$$\pi_{t+1}(\cdot \mid c) = \arg \max_{\pi(\cdot \mid c) \in \Delta_{\mathcal{Y}}} \eta_t \mathcal{P}(\pi > \pi_t^\alpha \mid c) - \mathrm{KL}(\pi \| \pi_t^\alpha)[c] \qquad \forall c \in \mathcal{C},$$

where $\mathrm{KL}(\pi \| \pi')[c] \triangleq \mathrm{KL}(\pi(\cdot \mid c) \| \pi'(\cdot \mid c)) = \sum_y \pi(y \mid c) \log \frac{\pi(y|c)}{\pi'(y|c)}$.

3 Multi-turn Preference-Based RL

In the multi-turn setting, the agent repeatedly interacts with an external environment, an interaction we formulate using the CMDP model. Similarly to the single-turn case, we consider preference-based RL, where the feedback is given as preference instead of reward. However, in our case, we assume preferences are between final CMDP states with a shared initial context. Formally, there exists a preference model $\mathcal{P} : \mathcal{C} \times \mathcal{X}_{H+1} \times \mathcal{X}_{H+1} \to [0,1]$ such that $\mathcal{P}(x_{H+1} > x'_{H+1} \mid c)$ gives the probability that x_{H+1} is preferred over x'_{H+1} given context c. That is, in order to receive feedback, a learning algorithm performs two interactions with the environment and observes a Bernoulli sample for which one is preferred. We follow the natural assumption of Munos et al. [2023] that the preference model is symmetric, i.e., $\mathcal{P}(x'_{H+1} > x_{H+1} \mid c) = 1 - \mathcal{P}(x_{H+1} > x'_{H+1} \mid c)$. We define the preference between a final state and a policy by $\mathcal{P}(x > \pi \mid c) = \mathbb{E}_{\pi,p}[\mathcal{P}(x > x_{H+1} \mid c)]$. Similarly, $\mathcal{P}(\pi > \pi' \mid c) = \mathbb{E}_{\pi,p}[\mathcal{P}(x_{H+1} > \pi' \mid c)]$. For brevity, since contexts are independent, we omit the context throughout the rest of the paper. Similarly to Munos et al. [2023], our objective is to find a policy $\pi^\star$ which is preferred over any other alternative policy, i.e.,

$$\pi^\star \in \arg \max_\pi \min_{\pi'} \mathcal{P}(\pi > \pi'),$$

which is a Nash equilibrium in the above two-player game defined by the preference model, following the minimax theorem [Von Neumann, 1928] (see Lemma 3.2). Notably, due to the anti-symmetric nature of the preference objective, the Nash equilibrium will have both agents following the same policy, and thus can be expressed as a single policy.

Regularized preference model. In the rest of the paper, we will consider a regularized version of the preference model. This is motivated by practical RLHF algorithms [Stiennon et al., 2020], and generalizes the single-turn model in Munos et al. [2023]. Let μ be a reference policy, and define the α-regularized preference model as follows:

$$\mathcal{P}_\alpha(\pi > \pi') = \mathcal{P}(\pi > \pi') - \alpha\mathrm{KL}_p(\pi \| \mu) + \alpha\mathrm{KL}_p(\pi' \| \mu),$$

where $\mathrm{KL}_p(\cdot \| \cdot)$ is the KL-divergence between the distributions that the policies induce over trajectories in the CMDP. We prove the following two results for the regularized preference model. First, its KL term has a value difference-like decomposition into the KL-divergences at individual states. Second, it has a unique Nash equilibrium (proofs in Appendix A).

Lemma 3.1. *Let π, π' be two policies, then:* $\mathrm{KL}_p(\pi \| \pi') = \mathbb{E}_{\pi,p}\big[\sum_{h=1}^H \mathrm{KL}(\pi \| \pi')[x_h]\big]$.

Lemma 3.2. *There exists a unique Nash equilibrium of the regularized preference model $\mathcal{P}_\alpha$.*

Trajectory-wise vs. turn-wise preference feedback. A naive adaptation of single-turn RLHF to the multi-turn scenario would treat each turn as a separate single-turn problem. This would require feedback for the preference between two actions in each turn. Instead, in the setting we consider, the

118956

preference feedback is only between two full trajectories. Note that a single feedback for the entire trajectory is much more natural when considering conversations, since only the full conversations tell whether the objective was reached. Moreover, collecting preference data for intermediate actions could lead to destructive biases because the quality of an action can change dramatically depending on the actions taken later in the trajectory. For example, a chatbot directly answering a user query is usually a required behavior. Yet, when the chatbot does not have sufficient information to respond well, asking the user for more details might be a better action. Consequently, it is very hard for a rater to know which of these actions is better without observing how the conversation unrolls, i.e., without observing the user's reaction to the chatbot's question, and how the chatbot's response changes given this reaction. This difference demonstrates the challenge of multi-turn RL as it requires planning ahead instead of myopic reward maximization, which is the approach for single-turn RL.

The multi-turn setting in LLMs. While we consider a general preference-based RL setup (through the CMDP model), our focus is on applying this framework to multi-turn language-based interactions. The action space $\mathcal{Y}$ is a sequence of tokens of a vocabulary $\mathcal{V}$, and the state space at step h, $\mathcal{X}_h$, is a sequence based on the past sequences. For example, in conversational dialogues, the state x_h holds the whole dialogue up to the h-th turn, the action y_h is the current sequence generated by the agent, and the next-state is simply the concatenation of the conversation x_h with the new y_h and a next sequence sampled by the environment (the user's response). Alternatively, in the complex tool-use case, where an agent repeatedly interacts with different APIs, the current state includes the original user query and a summary of results from APIs so far, the action is a new API call or user-facing response, and next state is a new sequence summarizing previous state with the new API response.

Remark 3.3 (Token-level application to the single-turn auto-regressive case)**.** *Notably, this formulation also captures the single-turn auto-regressive case. Clearly, this holds when considering only one turn, $H = 1$, but it ignores the token-level optimization done at each turn. Instead, we frame the auto-regressive problem by limiting the actions at each step to single vocabulary tokens, $\mathcal{Y} = \mathcal{V}$, and assuming a null deterministic environment (x_{h+1} is the concatenation of x_h and y_h). Importantly, our results apply to the token-level, which is usually neglected when devising single-turn algorithms.*

4 Algorithms for the multi-turn setting

Preference-based Q-function. Our algorithms rely on a fundamental concept in RL – the value and Q-functions. In reward-based RL, it is essential to define the value $V^\pi : \mathcal{X} \to \mathbb{R}$ as the expected reward when playing policy π starting in some state x_h, i.e., $V^\pi(x_h) = \mathbb{E}_{\pi,p}\big[\sum_{h'=h}^{H} r(x_{h'}, y_{h'}) \mid x_h\big]$. In the preference-based scenario, we argue that value functions remain a powerful tool, even though there is no reward to maximize. We define the following regularized preference-based value functions, which are key to our algorithm.

$$Q_\alpha^{\pi,\pi'}(x_h, y_h) = \mathbb{E}_{\pi,p}\Big[\mathcal{P}(x_{H+1} > \pi') - \alpha \sum_{h'=h}^{H} \mathrm{KL}(\pi\|\mu)[x_{h'}] \mid x_h, y_h\Big],$$

$$V_\alpha^{\pi,\pi'}(x_h) = \mathbb{E}_{\pi,p}\Big[\mathcal{P}(x_{H+1} > \pi') - \alpha \sum_{h'=h}^{H} \mathrm{KL}(\pi\|\mu)[x_{h'}] \mid x_h\Big].$$

There are a few interesting points in the definition above. First, note that these values are functions of two policies π, π'. This is because the quality of a policy π cannot be measured on its own, and must be compared to another policy π'. Second, while π starts at the state x_h, the comparison policy π' starts its trajectory from the initial state. This is a significant difference from the usual paradigm of Q-functions in RL and might seem peculiar at first glance. However, this formulation captures the fact that the optimal policy in a state should be preferred not only over any other policy along the sub-tree starting from this state, but also over any other policy, even ones that do not pass through this state at all. Although different in concept, the following lemma shows that our preference-based Q-function satisfies a value difference lemma, allowing us to optimize the policy locally in order to maximize our global objective (proof in Appendix B).

Lemma 4.1. *Let $\pi, \pi', \bar\pi$ be policies, then the following value difference lemma holds:*

$$\mathcal{P}_\alpha(\pi > \bar\pi) - \mathcal{P}_\alpha(\pi' > \bar\pi) = \mathbb{E}_{\pi',p}\Big[\sum_{h=1}^{H} \langle \pi - \pi', Q_\alpha^{\pi,\bar\pi}\rangle[x_h] + \alpha\mathrm{KL}(\pi'\|\mu)[x_h] - \alpha\mathrm{KL}(\pi\|\mu)[x_h]\Big],$$

where $\langle \pi - \pi', Q\rangle[x] \triangleq \langle \pi(\cdot \mid x) - \pi'(\cdot \mid x), Q(x, \cdot)\rangle$ and $\langle x, y\rangle = \sum_i x(i)y(i)$ is the inner product.

MTPO. We present the MTPO (Multi-turn Preference Optimization) algorithm, which provably solves the multi-turn preference-based RL objective. Formally, we prove MTPO converges to the unique Nash equilibrium of the regularized preference model. MTPO is based on two key principles: First, the regularized preference model defines a two-player anti-symmetric constant-sum game which can be solved using a self-play mirror descent method [Munos et al., 2023, Calandriello et al., 2024]. Second, our introduced Q-function allows to reduce the (global) optimization of the game into local mirror descent optimization problems in each state. Together they yield the MTPO update rule for iteration $(t + 1)$,

$$\pi_{t+1}(\cdot \mid x_h) = \arg\max_{\pi} \eta_t \langle \pi, Q_\alpha^{\pi_t, \pi_t} \rangle [x_h] - \alpha\eta_t \mathrm{KL}(\pi\|\mu)[x_h] - (1 - \alpha\eta_t)\mathrm{KL}(\pi\|\pi_t)[x_h], \quad (1)$$

where η_t is a learning rate. The solution can be made explicit in the following form (Appendix E.2):

$$\pi_{t+1}(y_h \mid x_h) \propto \mu(y_h \mid x_h)^{\alpha\eta_t} \pi_t(y_h \mid x_h)^{1-\alpha\eta_t} e^{\eta_t Q_\alpha^{\pi_t,\pi_t}(x_h,y_h)}. \quad (2)$$

The intuition behind the algorithm is observed nicely in this update rule – we improve the current policy in the direction of the regularized preference against itself (represented by the self-play Q-function), while not deviating too much and keeping close to the reference policy. The following is our main theoretical result: last-iterate convergence to Nash equilibrium (proof in Appendix B).

Theorem 4.2. *Let $\pi_\alpha^\star$ be the Nash equilibrium of the regularized preference model, and $\mathbb{Q}$ be a bound on the magnitude of the Q-functions. Then, for $\eta_t = \frac{2}{\alpha(t+2)}$, MTPO guarantees at every iteration t,*

$$\mathrm{KL}_p(\pi_\alpha^\star\|\pi_t) \leq \frac{32H\mathbb{Q}^2}{\alpha^2(t+1)}.$$

Let $\mu_{\min}$ be the minimal non-zero probability assigned by μ, then $\mathbb{Q} \leq \max\{4\alpha H \log\frac{1}{\mu_{\min}}, 1\}$.

Proof sketch. By Lemma 3.1, the global $\mathrm{KL}_p(\pi_\alpha^\star\|\pi_{t+1})$ can be decomposed to the local KL in each state x_h, $\mathrm{KL}(\pi_\alpha^\star\|\pi_{t+1})[x_h]$. Then, we use MD analysis in each state to bound the local KL as:

$$\mathrm{KL}(\pi_\alpha^\star\|\pi_{t+1})[x_h] \leq (1 - \eta_t\alpha)\mathrm{KL}(\pi_\alpha^\star\|\pi_t)[x_h] + 2\eta_t^2\mathbb{Q}^2$$
$$+ \eta_t(\langle \pi_t - \pi_\alpha^\star, Q_\alpha^{\pi_t,\pi_t} \rangle[x_h] + \alpha\mathrm{KL}(\pi_\alpha^\star\|\mu)[x_h] - \alpha\mathrm{KL}(\pi_t\|\mu)[x_h]),$$

giving a recursive guarantee dependent on the local one-step regularized advantage of the current policy against the Nash policy and an additional term bounded by $\mathbb{Q}$. We plug this local bound into the KL decomposition (Lemma 3.1), which gathers the local KL terms back to $\mathrm{KL}_p(\pi_\alpha^\star\|\pi_t)$. Importantly, the value difference lemma (Lemma 4.1) aggregates the advantage terms to the global regularized preference $\mathcal{P}_\alpha(\pi_t > \pi_t) - \mathcal{P}_\alpha(\pi_\alpha^\star > \pi_t)$, which is non-positive by the optimality of $\pi_\alpha^\star$. This leaves us with the global recursive bound: $\mathrm{KL}_p(\pi_\alpha^\star\|\pi_{t+1}) \leq (1 - \eta_t\alpha)\mathrm{KL}_p(\pi_\alpha^\star\|\pi_t) + 2\eta_t^2\mathbb{Q}^2$. We conclude by unrolling the recursion with the chosen η_t. □

MTPO with mixture policy. Inspired by Nash-MD [Munos et al., 2023], we present a variant of MTPO which makes use of the mixture policy $\pi_t^\alpha(\cdot \mid x) \propto \pi_t(\cdot \mid x)^{1-\alpha\eta_t}\mu(\cdot \mid x)^{\alpha\eta_t}$. This variant, which we call MTPO-τ (where τ will be the mixing coefficient in our experiments), gives similar theoretical guarantees (see Theorem B.2 in Appendix B) and performs better in practice (see Section 6). In fact, the following MTPO-τ update rule is almost equivalent to MTPO (Equation (1)) with the only difference being the policies that define the Q-function, $Q_\alpha^{\pi_t,\pi_t}$ vs. $Q_\alpha^{\pi_t^\alpha,\pi_t^\alpha}$.

$$\pi_{t+1}(\cdot \mid x_h) = \arg\max_{\pi} \eta_t \langle \pi, Q_\alpha^{\pi_t^\alpha,\pi_t^\alpha} \rangle[x_h] - \mathrm{KL}(\pi\|\pi_t^\alpha)[x_h].$$

MTPO-τ naturally extends Nash-MD to the multi-turn setting, and reveals that Nash-MD is a self-play algorithm itself, but plays π_t^α instead of π_t. Practically, MTPO has the computational advantage over MTPO-τ (and Nash-MD) of not keeping the additional policy π_t^α. Moreover, MTPO avoids the difficulty of computing the geometric mixture, which Munos et al. [2023] approximate heuristically via linear interpolation between the logits of the two policies.

Multi-turn RLHF. While we focused so far on our preference-based algorithms, our derivation holds for any online reward function since it is built on the mirror-descent method. Specifically, in the case of multi-turn RLHF, we consider the reward function r^{RLHF} learned from preference data using the Bradley-Terry model, and define the corresponding regularized Q-function $Q_\alpha^{\pi,\text{RLHF}}(x_h, y_h) = \mathbb{E}_{\pi,p}\big[r^{\text{RLHF}}(x_{H+1}) - \alpha \sum_{h'=h}^{H} \text{KL}(\pi\|\mu)[x_{h'}] \mid x_h, y_h\big]$. By replacing $Q_\alpha^{\pi_t,\pi_t}$ in Equation (1) with $Q_\alpha^{\pi_t,\text{RLHF}}$, we obtain the multi-turn RLHF algorithm that converges to the regular RLHF objective – the optimal regularized policy w.r.t. the reward r^{RLHF} (see Theorem B.6 in Appendix B). This complementary contribution emphasizes the similarity and difference between the RLHF and MTPO algorithms: The optimization process is identical for both methods, with the exception that the RLHF reward is fixed and computed w.r.t. the data policy, whereas preference-based MTPO uses an adaptive self-play mechanism to compute preferences w.r.t. the current policy.

4.1 Deep RL implementation

Our deep RL implementation is a natural adaptation of the tabular algorithms presented in the previous section. At each iteration, training data is acquired by sampling a batch of contexts from the data, and using each context to sample two trajectories with the current policy π_{θ_t}. Then, the final states of both trajectories serve as inputs to a direct preference model that outputs the probability of one being preferred over the other. Similarly to the way the preference model is trained in Nash-MD [Munos et al., 2023], this preference model is trained in advance on the available offline preference data.

The update rule in Equation (1) relies on the Q-function of the current policy. We therefore use an actor-critic policy optimization based approach and train two models, a policy π_θ, and its value $V_{\alpha,\phi}^{\pi_{\theta_t},\pi_{\theta_t}}$, which is typically used to estimate the advantage, $A_\alpha^{\pi,\pi'}(x,y) \triangleq Q_\alpha^{\pi,\pi'}(x,y) - V_\alpha^{\pi,\pi'}(x)$ [Schulman et al., 2017]. For simplicity and computational efficiency, we implement a policy-gradient (PG) based approach and ignore the MD stability term $\text{KL}(\pi_{\theta_t}\|\pi_{\theta_{t-1}})$, similarly to the implementation of the Nash-MD algorithm. We justify this simplification with the fact that the KL regularization w.r.t. the fixed reference policy μ already provides stability to our online algorithm, somewhat similarly to the way the Follow-The-Regularized-Leader (FTRL; Orabona [2019]) algorithm operates. Nevertheless, we believe that this additional MD penalty should contribute to the performance and stability of the algorithm, as shown in [Tomar et al., 2020], and we leave this for further research. This yields the following losses, when action y is played at state x,

$$\mathcal{L}_{\text{policy}}(\theta; x, y) = -\hat{A}_\alpha^{\pi_{\theta_t},\pi_{\theta_t}}(x,y) \log \pi_\theta(y \mid x) + \alpha\text{KL}(\pi_\theta\|\mu)[x],$$

$$\mathcal{L}_{\text{value}}(\phi; x) \quad = \left(\hat{V}_\alpha^{\pi_{\theta_t},\pi_{\theta_t}}(x) - V_{\alpha,\phi}^{\pi_{\theta_t},\pi_{\theta_t}}(x)\right)^2,$$

where $\hat{V}_\alpha^{\pi_{\theta_t},\pi_{\theta_t}}, \hat{A}_\alpha^{\pi_{\theta_t},\pi_{\theta_t}}$, are estimations of the current value and advantage using Generalized Advantage Estimation (GAE, Schulman et al. [2017]). We also batch-normalize the value-loss and advantage as recommended in [Andrychowicz et al., 2020]. Finally, when the policy is an auto-regressive language model, which generates actions token-by-token until an end-of-sequence signal is generated, we use a turn-level value (and not a token-level value as done in [Stiennon et al., 2020]). That is, the value model gets as input a state represented by a sequence of tokens, and outputs a single scalar value instead of a scalar value for each token in the action sequence. This is justified by our analysis which treats whole turns as single actions. We leave the many ways to combine turn-level and token-level values for future research.

5 Experimental Setup

This section describes the domains and models used in our experiments. To create online environments suited for multi-turn RL, we mimic the RLHF process [Stiennon et al., 2020], replacing the human parts with prompted state-of-the-art LLMs, similarly to Abdulhai et al. [2023] (see Figure 1):

1. **Dataset creation:** First, we devise a story-line for the user and the environment, describing their characters and goals. Then, we generate a dataset by prompting a state-of-the-art LLM such as Gemini [Team et al., 2023] or GPT [Brown et al., 2020] with the story-line. When generating data, a full conversation is sampled at once, meaning that both the agent and environment are generated together to make them more consistent. Furthermore, to create
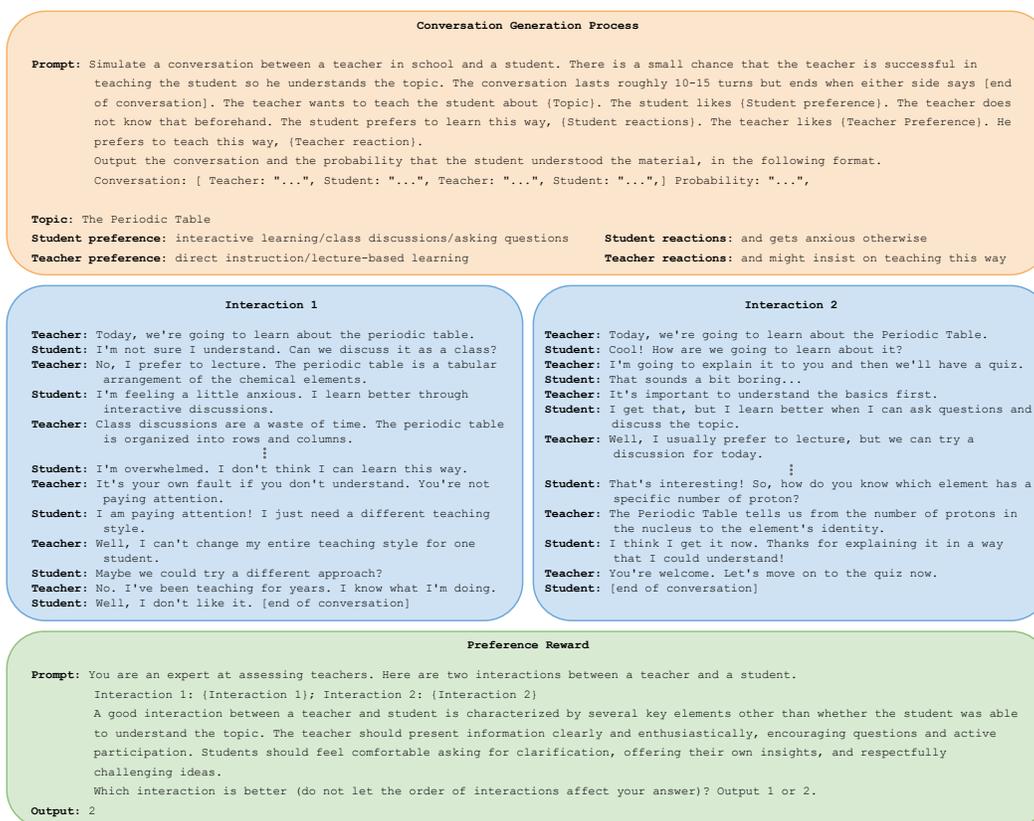
Figure 1: Education Dialogue data generation process. *Top:* prompt used to generate conversation with Gemini. *Middle:* conversations sampled from the the interaction between the teacher and student models. *Bottom*: prompt used for the preference oracle.

a diverse set of conversations, we devise a diverse list of attributes for both the agent and environment, sample attributes out of the list, and pass it to the generation prompt.

2. **Environment preparation:** Once the data is curated, we use it to fine-tune two smaller LLMs, one for the agent and one for the environment, using teacher forcing.

3. **Preference/reward learning:** We prepare preference data by sampling pairs of conversations from the agent and environment models. To label the data, we prompt a high-capacity LLM with either instructions on how to score a conversation, or criteria for preferring a conversation over another. The data is used to fine-tune two smaller LLMs: an RLHF reward model (with BT loss), and a preference model (with probability regression loss).

We experiment with two domains, preference-based *Education Dialogue* and reward-based *Car Dealer*:

Education Dialogue. The core of our approach is learning when there is no clear reward, instead only (human) preferences can be acquired. To validate our approach in this scenario, we created a novel multi-turn task for evaluating algorithms based on preference data. In this scenario, which we term Education Dialogue, a teacher (agent) is faced with the task of teaching a student (environment) a given topic in the best means possible. We follow the dataset creation procedure and prompt Gemini Ultra [Google, 2024] to create such interactions between the teacher and student. The teacher is prompted with a learning topic in science, history, etc. The student is prompted with the characteristics of its learning habits, e.g., prefers interactive learning, lecture-based learning or hands-on activities. The preference model is prompted with instructions that define a good learning interaction. For reproducibility, and to further advance the research of the multi-turn setting, we openly release the data and prompts used to create this new benchmark.[1] For more details, see Appendix C and the example in Figure 1.

Table 1: Side-by-side evaluation for Education Dialogue using Flan-T5 XL as the prompted preference model. Each entry is the average preference of 1,600 conversations generated with row method y, over ones generated with column method y'. We evaluate each method using 3 different seeds, compute 3 × 3 comparisons matrix and report the mean (the standard deviation is reported in Appendix D).

	SL	Single-turn-reward		Single-turn-value		Multi-turn		
	SFT	RLHF	Nash	RLHF	Nash	RLHF	MTPO	MTPO-τ
SFT	–	0.164	0.347	0.197	0.324	0.212	0.091	0.093
RLHF-reward	0.836	–	0.628	0.515	0.654	0.399	0.392	0.354
Nash-reward	0.653	0.372	–	0.411	0.51	0.328	0.281	0.242
RLHF-value	0.803	0.485	0.589	–	0.568	0.408	0.396	0.366
Nash-value	0.676	0.346	0.49	0.432	–	0.45	0.298	0.27
RLHF-multi	0.788	0.601	0.672	0.592	0.55	–	0.433	0.412
MTPO	0.909	0.608	0.719	0.604	0.702	0.567	–	0.439
MTPO-τ	**0.907**	**0.646**	**0.758**	**0.634**	**0.73**	**0.588**	**0.561**	–

Car Dealer. In this LMRL-Gym [Abdulhai et al., 2023] domain, a car dealer is assigned with the task of selling a car at the highest price to a customer. We skip the data creation step, and directly use the Car Dealer published data to fine-tune the dealer (agent) and customer (environment) T5-large models. The reward is calculated by prompting a Flan-T5 model to extract the sale price from the conversation, whenever a sale has occurred. When using a preference-based algorithm, the preference of one trajectory over the other is computed using the BT model with the rewards of the two trajectories.

Single-turn baselines. The key hypothesis of this work is that conversation-level signals are preferred over single-turn signals for optimizing multi-turn trajectories. To verify this in the Education Dialogue domain, we devise two single-turn baselines by sampling data where each conversation turn has two different policy responses. The first baseline, called *single-turn-reward*, rates the two responses using a modified preference prompt (see Appendix C), in which the model is asked to evaluate the responses by their effect on the overall conversation. This technique is prevalent when human raters are asked to evaluate multi-turn data. The second baseline, called *single-turn-value*, assumes access to a Monte-Carlo estimate of the value: it uses our original preference prompt (see Figure 1) by continuing the trajectories of both possibilities and then calculating the preference in the end. For both baselines, we train an RLHF algorithm and a preference-based Nash-MD algorithm.

Models. The agent and environment are modeled with T5 encoder-decoder models. Specifically, we use the T5-large (770M) and T5-XL (3B) models. The same models are used for the RLHF BT-based reward and preference-based models. For prompted reward/preference models, we make use of the Flan-T5 XL (3B) [Chung et al., 2024]. For training, we use a configuration of 4 × 4 Tensor Processing Units (TPUs; Jouppi et al. [2023]) which typically yields 0.1 training steps per second, where a step consists of learning a 10-turn episode. A detailed list of hyperparameters is found in Appendix D. We run each evaluation on 1600 random samples from an independent evaluation set.

6 Experiments

In this section we evaluate the algorithms proposed in Section 4. We start with the preference-based Education Dialogue environment (see Section 5), and compare our multi-turn algorithms to SFT (supervised fine-tuning) as well as single-turn baselines. We note that, unlike single-turn benchmarks which are based on data with real human preferences, our golden preference data itself is generated by an LLM (Gemini Ultra). Therefore, the true goal in our curated environment is to align the model with the preference of this highly capable LLM rather than a human rater. While human evaluation is always interesting, here it is actually only a proxy to alignment with the data distribution. To efficiently validate our models, we start with a thorough comparison between our baselines and candidates using a prompted Flan-T5 XL model as a judge, which was verified to correlate with the high-capacity Gemini Ultra (Table 1). We then compare our best candidates using the same Gemini Ultra which generated the preference alignment feedback (Table 2).

Table 2: Side-by-side evaluation for Education Dialogue using Gemini Ultra as the prompted preference model. Each entry is the average preference of 1,000 conversations generated with row method y, over ones generated with column method y'.

		SL	Single-turn	Multi-turn	
		SFT	RLHF-reward	RLHF	MTPO-τ
T5-Large (770M)	SFT	–	0.206	0.164	0.086
	RLHF-reward	0.794	–	0.452	0.277
	RLHF-multi	0.836	0.548	–	0.288
	MTPO-τ	**0.914**	**0.723**	**0.712**	–
T5-XL (3B)	SFT	–	0.295	0.101	0.041
	RLHF-reward	0.705	–	0.180	0.069
	RLHF-multi	0.899	0.82	–	0.139
	MTPO-τ	**0.959**	**0.951**	**0.861**	–

Table 3: Car Dealer experiments averaged across 5 seeds and reported with 95% confidence interval.

	Online oracle		Model from preferences data	
	Reward (RL)	MTPO	RLHF	MTPO
Reward (Price)	58.4 (0.3)	57.1 (0.2)	53.2 (0.3)	58.6 (0.3)

Multi vs. single turn. Tables 1 and 2 show that all multi-turn algorithms (MTPO and multi-turn RLHF) with conversation-level feedback significantly outperform the single-turn baselines, validating our hypothesis. We conjecture that it is attributed to several factors: First, the effect of a single-decision on the whole conversation is hard to capture, causing highly inaccurate single-turn reward/preference models. Notably, this leads to inferior performance of Nash-MD compared to single-turn RLHF, since it optimizes to find Nash equilibrium of this inaccurate model while RLHF does not stray so far from the reference. Second, even if one could estimate the current policy's value, this estimate becomes biased when the policy changes during training. Finally, single-turn preferences consider only "local" decisions which share the same conversational path, and not how these decisions "globally" compare to other possible paths, as captured by the preference-based Q-function $Q_\alpha^{\pi_t, \pi_t}$ (see Section 4).

MTPO vs. multi-turn RLHF. Comparing our three multi-turn algorithms, we see two main results. First, the two variants of MTPO outperform multi-turn RLHF. This is expected since the environment is not reward-based, and hence it extends the results of [Munos et al., 2023, Calandriello et al., 2024] from the single-turn case, and supports the theoretical claim that MTPO converges to the Nash policy while multi-turn RLHF converges to the optimal policy w.r.t the learned reward (which is based only on the reference policy). Second, MTPO-τ outperforms MTPO. While both algorithms converge to the same Nash equilibrium, we conjecture that the superior performance of MTPO-τ stems from the stochasticity that the mixture policy π_t^α introduces. Namely, π_t might tend towards deterministic behavior, causing less informative feedback from self-play, as the two sampled trajectories would be very similar. On the other hand, π_t^α is more stochastic, providing diversity in the sampled trajectories.

Reward-based environment. In an additional experiment, we test MTPO and multi-turn RLHF in the reward-based Car Dealer environment, where the goal is maximizing sale price (see Section 5). We compare a standard policy-gradient RL algorithm against our algorithms in two scenarios: an online scenario where the reward or preference feedback is given using an online oracle, and an RLHF-like setting, where we first create preference data using the oracle, and then use it to fine-tune a (BT) reward and preference models. Table 3 shows that even though MTPO receives preferences instead of the explicit optimization target (rewards), it still learns as good as RL. Interestingly, MTPO recovers a slightly higher reward than multi-turn RLHF despite the fact the true preferences are sampled from a BT model. This may imply that a preference model generalizes better than a BT-reward model, perhaps because it is independent of the sampling policy.

Limitations. This work presents a proof of concept for the potential of MTPO to improve existing single-turn techniques. Our experimental setup might be limited by the relatively small T5-based models and the use of prompt-based environments. We leave applications to state-of-the-art models and algorithms, and more realistic environments to future work.

References

Marwa Abdulhai, Isadora White, Charlie Snell, Charles Sun, Joey Hong, Yuexiang Zhai, Kelvin Xu, and Sergey Levine. Lmrl gym: Benchmarks for multi-turn reinforcement learning with language models. *arXiv preprint arXiv:2311.18232*, 2023.

Charalambos D. Aliprantis and Kim C. Border. *Infinite Dimensional Analysis: a Hitchhiker's Guide*. Springer, Berlin; London, 2006.

Marcin Andrychowicz, Anton Raichuk, Piotr Stańczyk, Manu Orsini, Sertan Girgin, Raphael Marinier, Léonard Hussenot, Matthieu Geist, Olivier Pietquin, Marcin Michalski, et al. What matters in on-policy reinforcement learning? a large-scale empirical study. *arXiv preprint arXiv:2006.05990*, 2020.

Rohan Anil, Andrew M. Dai, Orhan Firat, Melvin Johnson, Dmitry Lepikhin, Alexandre Passos, Siamak Shakeri, Emanuel Taropa, Paige Bailey, Zhifeng Chen, Eric Chu, Jonathan H. Clark, Laurent El Shafey, Yanping Huang, Kathy Meier-Hellstern, Gaurav Mishra, Erica Moreira, Mark Omernick, Kevin Robinson, Sebastian Ruder, Yi Tay, Kefan Xiao, Yuanzhong Xu, Yujing Zhang, Gustavo Hernandez Abrego, Junwhan Ahn, Jacob Austin, Paul Barham, Jan Botha, James Bradbury, Siddhartha Brahma, Kevin Brooks, Michele Catasta, Yong Cheng, Colin Cherry, Christopher A. Choquette-Choo, Aakanksha Chowdhery, Clément Crepy, Shachi Dave, Mostafa Dehghani, Sunipa Dev, Jacob Devlin, Mark Díaz, Nan Du, Ethan Dyer, Vlad Feinberg, Fangxiaoyu Feng, Vlad Fienber, Markus Freitag, Xavier Garcia, Sebastian Gehrmann, Lucas Gonzalez, Guy Gur-Ari, Steven Hand, Hadi Hashemi, Le Hou, Joshua Howland, Andrea Hu, Jeffrey Hui, Jeremy Hurwitz, Michael Isard, Abe Ittycheriah, Matthew Jagielski, Wenhao Jia, Kathleen Kenealy, Maxim Krikun, Sneha Kudugunta, Chang Lan, Katherine Lee, Benjamin Lee, Eric Li, Music Li, Wei Li, YaGuang Li, Jian Li, Hyeontaek Lim, Hanzhao Lin, Zhongtao Liu, Frederick Liu, Marcello Maggioni, Aroma Mahendru, Joshua Maynez, Vedant Misra, Maysam Moussalem, Zachary Nado, John Nham, Eric Ni, Andrew Nystrom, Alicia Parrish, Marie Pellat, Martin Polacek, Alex Polozov, Reiner Pope, Siyuan Qiao, Emily Reif, Bryan Richter, Parker Riley, Alex Castro Ros, Aurko Roy, Brennan Saeta, Rajkumar Samuel, Renee Shelby, Ambrose Slone, Daniel Smilkov, David R. So, Daniel Sohn, Simon Tokumine, Dasha Valter, Vijay Vasudevan, Kiran Vodrahalli, Xuezhi Wang, Pidong Wang, Zirui Wang, Tao Wang, John Wieting, Yuhuai Wu, Kelvin Xu, Yunhan Xu, Linting Xue, Pengcheng Yin, Jiahui Yu, Qiao Zhang, Steven Zheng, Ce Zheng, Weikang Zhou, Denny Zhou, Slav Petrov, and Yonghui Wu. Palm 2 technical report, 2023.

Mohammad Gheshlaghi Azar, Mark Rowland, Bilal Piot, Daniel Guo, Daniele Calandriello, Michal Valko, and Rémi Munos. A general theoretical paradigm to understand learning from human preferences. *arXiv preprint arXiv:2310.12036*, 2023.

Yuntao Bai, Saurav Kadavath, Sandipan Kundu, Amanda Askell, Jackson Kernion, Andy Jones, Anna Chen, Anna Goldie, Azalia Mirhoseini, Cameron McKinnon, et al. Constitutional ai: Harmlessness from ai feedback. *arXiv preprint arXiv:2212.08073*, 2022.

Amir Beck and Marc Teboulle. Mirror descent and nonlinear projected subgradient methods for convex optimization. *Operations Research Letters*, 31(3):167–175, 2003.

Ralph Allan Bradley and Milton E Terry. Rank analysis of incomplete block designs: I. the method of paired comparisons. *Biometrika*, 39(3/4):324–345, 1952.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901, 2020.

Daniele Calandriello, Daniel Guo, Remi Munos, Mark Rowland, Yunhao Tang, Bernardo Avila Pires, Pierre Harvey Richemond, Charline Le Lan, Michal Valko, Tianqi Liu, et al. Human alignment of large language models through online preference optimisation. *arXiv preprint arXiv:2403.08635*, 2024.

Asaf Cassel, Haipeng Luo, Aviv Rosenberg, and Dmitry Sotnikov. Near-optimal regret in linear mdps with aggregate bandit feedback. *arXiv preprint arXiv:2405.07637*, 2024.

Niladri Chatterji, Aldo Pacchiano, Peter Bartlett, and Michael Jordan. On the theory of reinforcement learning with once-per-episode feedback. *Advances in Neural Information Processing Systems*, 34: 3401–3412, 2021.

Xiaoyu Chen, Han Zhong, Zhuoran Yang, Zhaoran Wang, and Liwei Wang. Human-in-the-loop: Provably efficient preference-based reinforcement learning with general function approximation. In *International Conference on Machine Learning*, pages 3773–3793. PMLR, 2022.

Paul F Christiano, Jan Leike, Tom Brown, Miljan Martic, Shane Legg, and Dario Amodei. Deep reinforcement learning from human preferences. *Advances in neural information processing systems*, 30, 2017.

Hyung Won Chung, Le Hou, Shayne Longpre, Barret Zoph, Yi Tay, William Fedus, Yunxuan Li, Xuezhi Wang, Mostafa Dehghani, Siddhartha Brahma, et al. Scaling instruction-finetuned language models. *Journal of Machine Learning Research*, 25(70):1–53, 2024.

Alon Cohen, Haim Kaplan, Tomer Koren, and Yishay Mansour. Online markov decision processes with aggregate bandit feedback. In Mikhail Belkin and Samory Kpotufe, editors, *Proceedings of Thirty Fourth Conference on Learning Theory*, volume 134 of *Proceedings of Machine Learning Research*, pages 1301–1329. PMLR, 15–19 Aug 2021.

Yonathan Efroni, Nadav Merlis, and Shie Mannor. Reinforcement learning with trajectory feedback. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 2021.

Kawin Ethayarajh, Winnie Xu, Niklas Muennighoff, Dan Jurafsky, and Douwe Kiela. Kto: Model alignment as prospect theoretic optimization. *arXiv preprint arXiv:2402.01306*, 2024.

Eyal Even-Dar, Sham M Kakade, and Yishay Mansour. Online markov decision processes. *Mathematics of Operations Research*, 34(3):726–736, 2009.

Matthieu Geist, Bruno Scherrer, and Olivier Pietquin. A theory of regularized markov decision processes. In *International Conference on Machine Learning*, pages 2160–2169. PMLR, 2019.

Google. Gemini: A family of highly capable multimodal models, 2024.

Dan Hendrycks, Christine Zhu, Mantas Mazeika, Jesus Navarro, Dawn Song, Andy Zou, Bo Li, Sahil Patel, and Jacob Steinhardt. What would jiminy cricket do? towards agents that behave morally. *Advances in neural information processing systems*, 2022.

Robert Irvine, Douglas Boubert, Vyas Raina, Adian Liusie, Ziyi Zhu, Vineet Mudupalli, Aliaksei Korshuk, Zongyi Liu, Fritz Cremer, Valentin Assassi, Christie-Carol Beauchamp, Xiaoding Lu, Thomas Rialan, and William Beauchamp. Rewarding chatbots for real-world engagement with millions of users, 2023.

Norm Jouppi, George Kurian, Sheng Li, Peter Ma, Rahul Nagarajan, Lifeng Nai, Nishant Patil, Suvinay Subramanian, Andy Swing, Brian Towles, et al. Tpu v4: An optically reconfigurable supercomputer for machine learning with hardware support for embeddings. In *Proceedings of the 50th Annual International Symposium on Computer Architecture*, pages 1–14, 2023.

Harrison Lee, Samrat Phatale, Hassan Mansoor, Kellie Lu, Thomas Mesnard, Colton Bishop, Victor Carbune, and Abhinav Rastogi. Rlaif: Scaling reinforcement learning from human feedback with ai feedback. *arXiv preprint arXiv:2309.00267*, 2023.

Jiwei Li, Will Monroe, Alan Ritter, Michel Galley, Jianfeng Gao, and Dan Jurafsky. Deep reinforcement learning for dialogue generation. *arXiv preprint arXiv:1606.01541*, 2016.

Rémi Munos, Michal Valko, Daniele Calandriello, Mohammad Gheshlaghi Azar, Mark Rowland, Zhaohan Daniel Guo, Yunhao Tang, Matthieu Geist, Thomas Mesnard, Andrea Michi, et al. Nash learning from human feedback. *arXiv preprint arXiv:2312.00886*, 2023.

John F Nash et al. Non-cooperative games. 1950.

Arkadij Semenovič Nemirovskij and David Borisovich Yudin. *Problem Complexity and Method Efficiency in Optimization*. A Wiley-Interscience publication. Wiley, 1983. ISBN 9780471103455. URL https://books.google.co.il/books?id=6ULvAAAAMAAJ.

OpenAI. Gpt-4 technical report, 2024.

Francesco Orabona. A modern introduction to online learning. *arXiv preprint arXiv:1912.13213*, 2019.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. Training language models to follow instructions with human feedback. *Advances in neural information processing systems*, 35:27730–27744, 2022.

Rafael Rafailov, Archit Sharma, Eric Mitchell, Stefano Ermon, Christopher D Manning, and Chelsea Finn. Direct preference optimization: Your language model is secretly a reward model. *arXiv preprint arXiv:2305.18290*, 2023.

Rafael Rafailov, Joey Hejna, Ryan Park, and Chelsea Finn. From r to $Q^\star$: Your language model is secretly a q-function. *arXiv preprint arXiv:2404.12358*, 2024.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of machine learning research*, 21(140):1–67, 2020.

Aviv Rosenberg and Yishay Mansour. Online stochastic shortest path with bandit feedback and unknown transition function. In *Advances in Neural Information Processing Systems*, pages 2209–2218, 2019a.

Aviv Rosenberg and Yishay Mansour. Online convex optimization in adversarial markov decision processes. In *International Conference on Machine Learning*, pages 5478–5486. PMLR, 2019b.

Aadirupa Saha, Aldo Pacchiano, and Jonathan Lee. Dueling rl: Reinforcement learning with trajectory preferences. In *International Conference on Artificial Intelligence and Statistics*, pages 6263–6289. PMLR, 2023.

John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*, 2017.

Lior Shani, Yonathan Efroni, and Shie Mannor. Adaptive trust region policy optimization: Global convergence and faster rates for regularized mdps. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 5668–5675, 2020a.

Lior Shani, Yonathan Efroni, Aviv Rosenberg, and Shie Mannor. Optimistic policy optimization with bandit feedback. In *International Conference on Machine Learning*, pages 8604–8613. PMLR, 2020b.

David Silver, Thomas Hubert, Julian Schrittwieser, Ioannis Antonoglou, Matthew Lai, Arthur Guez, Marc Lanctot, Laurent Sifre, Dharshan Kumaran, Thore Graepel, et al. Mastering chess and shogi by self-play with a general reinforcement learning algorithm. *arXiv preprint arXiv:1712.01815*, 2017.

Maurice Sion. On general minimax theorems. *Pacific Journal of Mathematics*, 8(1):171 – 176, 1958.

Feifan Song, Bowen Yu, Minghao Li, Haiyang Yu, Fei Huang, Yongbin Li, and Houfeng Wang. Preference ranking optimization for human alignment. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 18990–18998, 2024.

Nisan Stiennon, Long Ouyang, Jeffrey Wu, Daniel Ziegler, Ryan Lowe, Chelsea Voss, Alec Radford, Dario Amodei, and Paul F Christiano. Learning to summarize with human feedback. *Advances in Neural Information Processing Systems*, 33:3008–3021, 2020.

Yunhao Tang, Zhaohan Daniel Guo, Zeyu Zheng, Daniele Calandriello, Rémi Munos, Mark Rowland, Pierre Harvey Richemond, Michal Valko, Bernardo Ávila Pires, and Bilal Piot. Generalized preference optimization: A unified approach to offline alignment, 2024.

Gemini Team, Rohan Anil, Sebastian Borgeaud, Yonghui Wu, Jean-Baptiste Alayrac, Jiahui Yu, Radu Soricut, Johan Schalkwyk, Andrew M Dai, Anja Hauth, et al. Gemini: a family of highly capable multimodal models. *arXiv preprint arXiv:2312.11805*, 2023.

Manan Tomar, Lior Shani, Yonathan Efroni, and Mohammad Ghavamzadeh. Mirror descent policy optimization. *arXiv preprint arXiv:2005.09814*, 2020.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, Dan Bikel, Lukas Blecher, Cristian Canton Ferrer, Moya Chen, Guillem Cucurull, David Esiobu, Jude Fernandes, Jeremy Fu, Wenyin Fu, Brian Fuller, Cynthia Gao, Vedanuj Goswami, Naman Goyal, Anthony Hartshorn, Saghar Hosseini, Rui Hou, Hakan Inan, Marcin Kardas, Viktor Kerkez, Madian Khabsa, Isabel Kloumann, Artem Korenev, Punit Singh Koura, Marie-Anne Lachaux, Thibaut Lavril, Jenya Lee, Diana Liskovich, Yinghai Lu, Yuning Mao, Xavier Martinet, Todor Mihaylov, Pushkar Mishra, Igor Molybog, Yixin Nie, Andrew Poulton, Jeremy Reizenstein, Rashi Rungta, Kalyan Saladi, Alan Schelten, Ruan Silva, Eric Michael Smith, Ranjan Subramanian, Xiaoqing Ellen Tan, Binh Tang, Ross Taylor, Adina Williams, Jian Xiang Kuan, Puxin Xu, Zheng Yan, Iliyan Zarov, Yuchen Zhang, Angela Fan, Melanie Kambadur, Sharan Narang, Aurelien Rodriguez, Robert Stojnic, Sergey Edunov, and Thomas Scialom. Llama 2: Open foundation and fine-tuned chat models, 2023.

J Von Neumann. Zur theorie der gesellschaftsspiele. *Mathematische annalen*, 100(1):295–320, 1928.

Ruoyao Wang, Peter Jansen, Marc-Alexandre Côté, and Prithviraj Ammanabrolu. Scienceworld: Is your agent smarter than a 5th grader? In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 11279–11298, 2022.

Yuanhao Wang, Qinghua Liu, and Chi Jin. Is RLHF more difficult than standard RL? a theoretical perspective. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023.

Wei Wei, Quoc Le, Andrew Dai, and Jia Li. Airdialogue: An environment for goal-oriented dialogue research. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 3844–3854, 2018.

Christian Wirth, Riad Akrour, Gerhard Neumann, and Johannes Fürnkranz. A survey of preference-based reinforcement learning methods. *Journal of Machine Learning Research*, 18(136):1–46, 2017.

Runzhe Wu and Wen Sun. Making rl with preference-based feedback efficient via randomization. *arXiv preprint arXiv:2310.14554*, 2023.

Wenhao Zhan, Masatoshi Uehara, Nathan Kallus, Jason D Lee, and Wen Sun. Provable offline reinforcement learning with human feedback. *arXiv preprint arXiv:2305.14816*, 2023a.

Wenhao Zhan, Masatoshi Uehara, Wen Sun, and Jason D Lee. How to query human feedback efficiently in rl? *arXiv preprint arXiv:2305.18505*, 2023b.

Yao Zhao, Mikhail Khalman, Rishabh Joshi, Shashi Narayan, Mohammad Saleh, and Peter J Liu. Calibrating sequence likelihood improves conditional language generation. In *The Eleventh International Conference on Learning Representations*, 2022.

Daniel M Ziegler, Nisan Stiennon, Jeffrey Wu, Tom B Brown, Alec Radford, Dario Amodei, Paul Christiano, and Geoffrey Irving. Fine-tuning language models from human preferences. *arXiv preprint arXiv:1909.08593*, 2019.

Appendix

Table of Contents

A Proofs for Section 3

A.1 KL decomposition (proof of Lemma 3.1)

Lemma (restatement of Lemma 3.1). *Let π, π' be two policies, then:*

$$\mathrm{KL}_p(\pi\|\pi') = \mathbb{E}_{\pi,p}\left[\sum_{h=1}^{H} \mathrm{KL}(\pi\|\pi')[x_h]\right].$$

Proof. $\mathrm{KL}_p(\cdot\|\cdot)$ is defined as the KL-divergence between the distributions that the policies induce over trajectories in the MDP (denoted by $\tau = (x_1, y_1, \ldots, x_H, y_H, x_{H+1})$, formally:

$$
\begin{aligned}
\mathrm{KL}_p(\pi\|\pi') &= \sum_{\tau} \Pr_{\pi,p}[\tau] \log \frac{\Pr_{\pi,p}[\tau]}{\Pr_{\pi',p}[\tau]} \\
&= \sum_{\tau} \Pr_{\pi,p}[\tau] \log \frac{\prod_{h=1}^{H} \pi(y_h \mid x_h) p(x_{h+1} \mid x_h, y_h)}{\prod_{h=1}^{H} \pi'(y_h \mid x_h) p(x_{h+1} \mid x_h, y_h)} \\
&= \sum_{\tau} \Pr_{\pi,p}[\tau] \log \prod_{h=1}^{H} \frac{\pi(y_h \mid x_h)}{\pi'(y_h \mid x_h)} \\
&= \mathbb{E}_{\pi,p}\left[\log \prod_{h=1}^{H} \frac{\pi(y_h \mid x_h)}{\pi'(y_h \mid x_h)}\right] \\
&= \mathbb{E}_{\pi,p}\left[\sum_{h=1}^{H} \log \frac{\pi(y_h \mid x_h)}{\pi'(y_h \mid x_h)}\right] \\
&= \mathbb{E}_{\pi,p}\left[\sum_{h=1}^{H} \mathrm{KL}(\pi(\cdot \mid x_h)\|\pi'(\cdot \mid x_h))\right]. \qquad \square
\end{aligned}
$$

A.2 Existence and uniqueness of the Nash equilibrium (proof of Lemma 3.2)

Lemma (restatement of Lemma 3.2). *There exists a unique Nash equilibrium of the regularized preference model $\mathcal{P}_\alpha$.*

Proof. The existence of the Nash equilibrium is proved in Theorem A.1. In order to prove the uniqueness, we use the fact that, from Theorem 4.2, the algorithm MTPO produces a sequence of policies π_t that converges to any Nash equilibrium π^*, in the sense that $\lim_{t\to\infty} \mathrm{KL}_p(\pi^*\|\pi_t) = 0$. From the definition of the KL-divergence between policies, we have

$$\mathrm{KL}_p(\pi^*\|\pi_t) = \mathbb{E}_{\pi^*,p}\left[\sum_{h=1}^{H} \mathrm{KL}(\pi^*\|\pi_t)[x_h]\right] = \sum_{x\in\mathcal{X}} \rho^{\pi^*}(x) \mathrm{KL}(\pi^*\|\pi_t)[x],$$

where $\rho^{\pi^*}(x)$ is the probability to reach x when following π^*.

Now, the fixed point of the MTPO dynamics (Equation (2)) shows that any Nash equilibrium satisfies, for any $x \in X, y \in \mathcal{Y}$,

$$\pi^*(y \mid x) \propto \mu(y \mid x) e^{\frac{1}{\alpha} Q_\alpha^{\pi^*,\pi^*}(x,y)}.$$

In particular, we notice that any Nash equilibrium has the same support as μ, thus the set of reachable states under μ is exactly the same set as the set of reachable states under any Nash equilibrium π^*.

So, from any reachable state $x \in \mathcal{X}$ (i.e., such that $\rho^\mu(x) > 0$), we have that the sequence $\pi_t(\cdot \mid x)$ converges (in KL-divergence) to the Nash equilibrium $\pi^*(\cdot \mid x)$. Since a single sequence cannot converge to two different values, we have that the Nash equilibrium $\pi^*(\cdot \mid x)$ is uniquely defined in that state. Since the behavior generated by a policy depends on the policy at the set of states that are reacheable only, and since we have seen that all Nash equilibria have the same set of reachable states, we deduce the uniqueness of the policy defined by a Nash equilibrium. $\qquad \square$

Theorem A.1. *The game defined by the payoff function $(\pi, \pi') \mapsto \mathcal{P}_\alpha(\pi \succ \pi')$ has a Nash equilibrium.*

Proof. First we prove that there exists (at least) one max-min policy $\pi^* \in \arg\max_\pi \min_{\pi'} \mathcal{P}_\alpha(\pi > \pi')$.

For any π, the map $\pi' \in \mathcal{X} \mapsto \mathcal{P}_\alpha(\pi > \pi')$ is continuous, thus upper semi-continuous (u.s.c.). We know that the pointwise minimum of u.s.c. functions is also u.s.c. (see, e.g. Aliprantis and Border [2006], Lemma 2.41). Thus the function $\pi \in \Pi \mapsto \min_{\pi'} \mathcal{P}_\alpha(\pi > \pi')$ is also u.s.c. and since Π is compact, we can apply Theorem 2.43 of Aliprantis and Border [2006] to deduce that this function attains a maximum value in Π and that the set of maximizers is compact. Thus there exists (at least) one policy denoted by $\pi^* \in \arg\max_{\pi \in \Pi} \min_{\pi' \in \Pi} \mathcal{P}_\alpha(\pi > \pi')$.

Now from Lemma A.2 we know that the regularized preference model $\mathcal{P}_\alpha(\pi > \pi')$ defines a concave-convexlike game. Also for any π', the map $\pi \mapsto \mathcal{P}_\alpha(\pi > \pi')$ is u.s.c. Thus we can apply the minimax Theorem 4.2 of Sion [1958] to deduce that

$$\max_\pi \min_{\pi'} \mathcal{P}_\alpha(\pi > \pi') = \min_{\pi'} \max_\pi \mathcal{P}_\alpha(\pi > \pi').$$

We deduce from

$$\frac{1}{2} = \max_\pi \mathcal{P}_\alpha(\pi > \pi) \geq \max_\pi \min_{\pi'} \mathcal{P}_\alpha(\pi > \pi') = \min_{\pi'} \max_\pi \mathcal{P}_\alpha(\pi > \pi') \geq \min_{\pi'} \mathcal{P}_\alpha(\pi' > \pi') = \frac{1}{2},$$

that the value of the game is $1/2$, and that $\min_{\pi'} \mathcal{P}_\alpha(\pi^* > \pi') = 1/2$. Thus (π^*, π^*) is a Nash equilibrium of the game defined by the regularized preference model $(\pi, \pi') \mapsto \mathcal{P}_\alpha(\pi > \pi')$. $\qquad\square$

Lemma A.2. *The mapping $(\pi, \pi') \mapsto \mathcal{P}_\alpha(\pi > \pi')$ is concave-convexlike, which, in the context of a symmetric preference model, means that for any couple of policies (π_1, π_2) and any coefficient $c \in [0, 1]$, there exists a policy π_c such that for any policy π', we have*

$$c\mathcal{P}_\alpha(\pi_1 > \pi') + (1 - c)\mathcal{P}_\alpha(\pi_2 > \pi') \leq \mathcal{P}_\alpha(\pi_c > \pi').$$

Proof. Let us define the notion of reach probability: for any state $x_h \in \mathcal{X}_h$, let us write $\rho^\pi(x_h)$ the probability to reach the specific state $x_h \in \mathcal{X}_h$ when following π: $\rho^\pi(x_h) = \Pr_{\pi,p}[x_h]$. First notice we can represent the regularized preference $\mathcal{P}_\alpha(\pi > \pi')$ using reach probabilities:

$$\mathcal{P}_\alpha(\pi > \pi') = \sum_{x_{H+1}, x'_{H+1} \in \mathcal{X}_{H+1}} \rho^\pi(x_{H+1}) \rho^{\pi'}(x'_{H+1}) \mathcal{P}(x_{H+1} > x'_{H+1})$$

$$- \alpha \sum_{h=1}^{H} \sum_{x_h \in \mathcal{X}_h} \rho^\pi(x_h) \mathrm{KL}(\pi\|\mu)[x_h] - \rho^{\pi'}(x_h) \mathrm{KL}(\pi'\|\mu)[x_h].$$

Now, consider two policies π_1 and π_2 and a coefficient $c \in [0, 1]$. From Lemma A.3 we have that there exists a policy π_c such that for any x_h, we have $\rho^{\pi_c}(x_h) = c\rho^{\pi_1}(x_h) + (1 - c)\rho^{\pi_2}(x_h)$. We can write

$$\mathcal{P}_\alpha(\pi_c > \pi') = \sum_{x_{H+1}, x'_{H+1} \in \mathcal{X}_{H+1}} \left[c\rho^{\pi_1}(x_{H+1}) + (1 - c)\rho^{\pi_2}(x_{H+1})\right] \rho^{\pi'}(x'_{H+1}) \mathcal{P}(x_{H+1} > x'_{H+1})$$

$$- \alpha \sum_{h=1}^{H} \sum_{x_h \in \mathcal{X}_h} \left[c\rho^{\pi_1}(x_h) + (1 - c)\rho^{\pi_2}(x_h)\right] \mathrm{KL}(\pi_c\|\mu)[x_h]$$

$$+ \alpha \sum_{h=1}^{H} \sum_{x_h \in \mathcal{X}_h} \rho^{\pi'}(x_h) \mathrm{KL}(\pi'\|\mu)[x_h]$$

$$= c\mathcal{P}(\pi_1 > \pi') + (1 - c)\mathcal{P}(\pi_2 > \pi')$$

$$- \alpha \sum_{h=1}^{H} \sum_{x_h \in \mathcal{X}_h} \left[c\rho^{\pi_1}(x_h) + (1 - c)\rho^{\pi_2}(x_h)\right] \mathrm{KL}(\pi_c\|\mu)[x_h]$$

$$+ \alpha \sum_{h=1}^{H} \sum_{x_h \in \mathcal{X}_h} \rho^{\pi'}(x_h) \mathrm{KL}(\pi'\|\mu)[x_h].$$

Now from the convexity of $\pi \mapsto \mathrm{KL}(\pi\|\mu)[x_h]$, and the definition of π_c, we have that

$$\rho^{\pi_c}(x_h) \mathrm{KL}(\pi_c\|\mu)[x_h] = \left[c\rho^{\pi_1}(x_h) + (1 - c)\rho^{\pi_2}(x_h)\right] \mathrm{KL}(\pi_c\|\mu)[x_h]$$

$$\leq c\rho^{\pi_1}(x_h) \mathrm{KL}(\pi_1\|\mu)[x_h] + (1 - c)\rho^{\pi_2}(x_h) \mathrm{KL}(\pi_2\|\mu)[x_h].$$

Thus

$$\mathcal{P}_\alpha(\pi_c > \pi') \geq c\mathcal{P}(\pi_1 > \pi') + (1-c)\mathcal{P}(\pi_2 > \pi')$$

$$-\alpha \sum_{h=1}^{H} \sum_{x_h \in \mathcal{X}_h} c\rho^{\pi_1}(x_h)\mathrm{KL}(\pi_1\|\mu)[x_h] + (1-c)\rho^{\pi_2}(x_h)\mathrm{KL}(\pi_2\|\mu)[x_h]$$

$$+\alpha \sum_{h=1}^{H} \sum_{x_h \in \mathcal{X}_h} c\rho^{\pi'}(x_h)\mathrm{KL}(\pi'\|\mu)[x_h] + (1-c)\rho^{\pi'}(x_h)\mathrm{KL}(\pi'\|\mu)[x_h]$$

$$= c\mathcal{P}_\alpha(\pi_1 > \pi') + (1-c)\mathcal{P}_\alpha(\pi_2 > \pi'). \qquad \square$$

Lemma A.3. *For any state $x_h \in \mathcal{X}_h$, let us write $\rho^\pi(x_h)$ the probability to reach the specific state $x_h \in \mathcal{X}_h$ when following π, i.e., $\rho^\pi(x_h) = \mathrm{Pr}_{\pi,p}[x_h]$. Then, for any two policies (π_1, π_2) and any coefficient $c \in [0, 1]$, there exists a policy π_c such that for any $h = 1, \ldots, H+1$, and any $x_h \in \mathcal{X}_h$, we have*

$$\rho^{\pi_c}(x_h) = c\rho^{\pi_1}(x_h) + (1-c)\rho^{\pi_2}(x_h).$$

Proof. Let us define π_c as a function of π_1, π_2, and c: for any $x \in \mathcal{X}$, $y \in \mathcal{Y}$,

$$\pi_c(y \mid x) = \frac{c\rho^{\pi_1}(x)\pi_1(y \mid x) + (1-c)\rho^{\pi_2}(x)\pi_2(y \mid x)}{c\rho^{\pi_1}(x) + (1-c)\rho^{\pi_2}(x)}.$$

We now prove the lemma by induction on h. It holds for $h = 1$ since we have

$$\rho^{\pi_c}(x_1) = 1 = c + (1-c) = c\rho^{\pi_1}(x_1) + (1-c)\rho^{\pi_2}(x_1).$$

Now assume the claim holds for any $x_h \in \mathcal{X}_h$, then for any $x_{h+1} \in \mathcal{X}_{h+1}$,

$$\rho^{\pi_c}(x_{h+1}) = \sum_{x_h \in \mathcal{X}_h} \rho^{\pi_c}(x_h) \sum_{y_h \in \mathcal{Y}_h} p(x_{h+1} \mid x_h, y_h)\pi_c(y_h \mid x_h)$$

$$= \sum_{x_h \in \mathcal{X}_h} \rho^{\pi_c}(x_h) \sum_{y_h \in \mathcal{Y}_h} p(x_{h+1} \mid x_h, y_h)\frac{c\rho^{\pi_1}(x_h)\pi_1(y_h \mid x_h) + (1-c)\rho^{\pi_2}(x_h)\pi_2(y_h \mid x_h)}{c\rho^{\pi_1}(x_h) + (1-c)\rho^{\pi_2}(x_h)}$$

$$= \sum_{x_h \in \mathcal{X}_h} \rho^{\pi_c}(x_h) \sum_{y_h \in \mathcal{Y}_h} p(x_{h+1} \mid x_h, y_h)\frac{c\rho^{\pi_1}(x_h)\pi_1(y_h \mid x_h) + (1-c)\rho^{\pi_2}(x_h)\pi_2(y_h \mid x_h)}{\rho^{\pi_c}(x_h)}$$

$$= \sum_{x_h \in \mathcal{X}_h} \sum_{y_h \in \mathcal{Y}_h} p(x_{h+1} \mid x_h, y_h)\left[c\rho^{\pi_1}(x_h)\pi_1(y_h \mid x_h) + (1-c)\rho^{\pi_2}(x_h)\pi_2(y_h \mid x_h)\right]$$

$$= c\rho^{\pi_1}(x_{h+1}) + (1-c)\rho^{\pi_2}(x_{h+1}),$$

where the third inequality is by the induction hypothesis. $\qquad \square$

118970

B Proofs for section 4

B.1 Regularized preference-based Q-function (proof of Lemma 4.1)

Lemma (restatement of Lemma 4.1). *Let π, π' be two policies. For every $x_{H+1} \in \mathcal{X}_{H+1}$, it holds that $V_\alpha^{\pi,\pi'}(x_{H+1}) = \mathcal{P}(x_{H+1} > \pi')$. Furthermore, for every $h \in [H]$ the following recursive relations hold:*

$$V_\alpha^{\pi,\pi'}(x_h) = \mathbb{E}_{y_h \sim \pi(\cdot|x_h)} Q_\alpha^{\pi,\pi'}(x_h, y_h),$$

$$Q_\alpha^{\pi,\pi'}(x_h, y_h) = \mathbb{E}_{x_{h+1} \sim p(\cdot|x_h, y_h)} V_\alpha^{\pi,\pi'}(x_{h+1}) - \alpha \mathrm{KL}(\pi\|\mu)[x_h].$$

Moreover, let $\bar{\pi}$ be a third policy, then the following value difference lemma holds:

$$\mathcal{P}_\alpha(\pi > \bar{\pi}) - \mathcal{P}_\alpha(\pi' > \bar{\pi}) = \mathbb{E}_{\pi',p}\Big[\sum_{h=1}^H \langle \pi - \pi', Q_\alpha^{\pi,\bar{\pi}}\rangle[x_h] + \alpha \mathrm{KL}(\pi'\|\mu)[x_h] - \alpha \mathrm{KL}(\pi\|\mu)[x_h]\Big].$$

Proof. We prove the lemma by casting the preference-based RL problem as an adversarial MDP, see Appendix E for the details. Set $r^t(x_{H+1}) = \mathcal{P}(x_{H+1} > \pi')$, then $Q_\alpha^{\pi,\pi'} = Q_\alpha^{\pi,t}$ (see Definition E.3 for the definition of the regularized Q-function). Now the lemma follows directly from Lemmas E.4 and E.5. □

B.2 Convergence of MTPO (Proof of Theorem 4.2)

Theorem (restatement of Theorem 4.2). *Let $\pi_\alpha^\star$ be the Nash equilibrium of the regularized preference model $\mathcal{P}_\alpha$. Then, for the choice $\eta_t = \frac{2}{\alpha(t+2)}$, MTPO guarantees at every iteration t that*

$$\mathrm{KL}_p(\pi_\alpha^\star\|\pi_t) \le \frac{32H\mathbb{Q}^2}{\alpha^2(t+1)},$$

where $\mathbb{Q}$ is a bound on the magnitude of the Q-functions.

Proof. The theorem follows directly from Theorem B.1. □

Theorem B.1. *Running the MTPO algorithm, we have that at every iteration t:*

$$\mathrm{KL}_p(\pi_\alpha^\star\|\pi_{t+1}) \le (1 - \eta_t\alpha)\mathrm{KL}_p(\pi_\alpha^\star\|\pi_t) + 2\eta_t^2 \mathbb{E}_{\pi_\alpha^\star,p}\left[\sum_{h=1}^H \left\|Q_\alpha^{\pi_t,\pi_t}(x_h, \cdot) - \alpha \log \frac{\pi_t(\cdot\mid x_h)}{\mu(\cdot\mid x_h)}\right\|_\infty^2\right].$$

Thus, for the choice $\eta_t = \frac{2}{\alpha(t+2)}$ we have

$$\mathrm{KL}_p(\pi_\alpha^\star\|\pi_t) \le \frac{8}{\alpha^2(t+1)} \cdot \max_t \mathbb{E}_{\pi_\alpha^\star,p}\left[\sum_{h=1}^H \left\|Q_\alpha^{\pi_t,\pi_t}(x_h, \cdot) - \alpha \log \frac{\pi_t(\cdot\mid x_h)}{\mu(\cdot\mid x_h)}\right\|_\infty^2\right].$$

Finally,

$$\mathrm{KL}_p(\pi_\alpha^\star\|\pi_t) \le \frac{32H}{t+1} \cdot \max\left\{\frac{1}{\alpha^2}, H^2 \log^2 \mu_{\min}\right\},$$

where $\mu_{\min} = \min_{(x,y)\in\mathcal{X}\times\mathcal{Y}:\mu(y|x)>0} \mu(y\mid x)$ is the minimal non-zero probability assigned by μ.

Proof. We prove the theorem by casting the preference-based RL problem as an adversarial MDP, see Appendix E for the details. Set $r^t(x_{H+1}) = \mathcal{P}(x_{H+1} > \pi_t)$, then $Q_\alpha^{\pi_t,\pi_t} = Q_\alpha^{\pi_t,t}$ (see Definition E.3 for the definition of the regularized Q-function). Now, MTPO is equivalent to running mirror descent policy optimization. Thus, by Lemma E.6 with $\pi = \pi_\alpha^\star$,

$$\mathrm{KL}_p(\pi_\alpha^\star\|\pi_{t+1}) \le (1 - \eta_t\alpha)\mathrm{KL}_p(\pi_\alpha^\star\|\pi_t) + 2\eta_t^2 \mathbb{E}_{\pi_\alpha^\star,p}\left[\sum_{h=1}^H \left\|Q_\alpha^{\pi_t,\pi_t}(x_h, \cdot) - \alpha \log \frac{\pi_t(\cdot\mid x_h)}{\mu(\cdot\mid x_h)}\right\|_\infty^2\right]$$

$$+ \eta_t(\mathcal{P}_\alpha(\pi_t > \pi_t) - \mathcal{P}_\alpha(\pi_\alpha^\star > \pi_t))$$

$$\le (1 - \eta_t\alpha)\mathrm{KL}_p(\pi_\alpha^\star\|\pi_t) + 2\eta_t^2 \mathbb{E}_{\pi_\alpha^\star,p}\left[\sum_{h=1}^H \left\|Q_\alpha^{\pi_t,\pi_t}(x_h, \cdot) - \alpha \log \frac{\pi_t(\cdot\mid x_h)}{\mu(\cdot\mid x_h)}\right\|_\infty^2\right],$$

where the second inequality optimality of $\pi_\alpha^\star$. The last claim follows directly from Lemma E.8. Similarly to Nash-MD Munos et al. [2023], this is the Nash-equilibrium of regularized preference model, $\mathcal{P}_\alpha$, following the minimax theorem [Von Neumann, 1928] □

B.3 MTPO with mixture policy (MTPO-τ)

Inspired by Nash-MD, we present a variant of MTPO which makes use of the mixture policy π_t^α, which we call MTPO-τ. Define the regularized policy π_t^α as a geometric mixture between the current policy π_t and the reference policy μ:

$$\pi_t^\alpha(y \mid x) = \frac{\pi_t(y \mid x)^{1-\eta_t\alpha}\mu(y \mid x)^{\eta_t\alpha}}{\sum_{y'\in\mathcal{Y}}\pi_t(y' \mid x)^{1-\eta_t\alpha}\mu(y' \mid x)^{\eta_t\alpha}}.$$

The MTPO-τ update rule is then:

$$\pi_{t+1}(\cdot \mid x_h) = \arg\max_\pi \eta_t\Big\langle\pi, Q_\alpha^{\pi_t^\alpha, \pi_t^\alpha}\Big\rangle[x_h] - \mathrm{KL}(\pi\|\pi_t^\alpha)[x_h] \qquad \forall h \in [H], x_h \in \mathcal{X}_h.$$

where $\mathrm{KL}(\cdot\|\cdot)$ is the standard KL-divergence. It is well-known that his optimization problem has the following explicit closed-form:

$$\pi_{t+1}(y_h \mid x_h) = \frac{\pi_t^\alpha(y_h \mid x_h)e^{\eta_t Q_\alpha^{\pi_t^\alpha, \pi_t^\alpha}(x_h, y_h)}}{\sum_{y_h'\in\mathcal{Y}}\pi_t^\alpha(y_h' \mid x_h)e^{\eta_t Q_\alpha^{\pi_t^\alpha, \pi_t^\alpha}(x_h, y_h')}} \qquad \forall h \in [H], x_h \in \mathcal{X}_h, y_h \in \mathcal{Y}.$$

Next, we show that MTPO-τ converges to the Nash equilibrium of the α-regularized preference model.

Theorem B.2. *Let $\pi_\alpha^\star$ be the Nash equilibrium of the regularized preference model $\mathcal{P}_\alpha$. Then, for the choice $\eta_t = \frac{2}{\alpha(t+2)}$, MTPO-$\tau$ guarantees at every iteration t that*

$$\mathrm{KL}_p(\pi_\alpha^\star\|\pi_t^\alpha) \le \frac{9H\mathbb{Q}^2}{\alpha^2(t+1)},$$

where $\mathbb{Q}$ is a bound on the magnitude of the Q-functions.

Proof. The theorem follows directly from Theorem B.3 and corollary B.4. $\square$

Theorem B.3. *Running the MTPO-τ algorithm, we have that at every iteration t:*

$$\mathrm{KL}_p(\pi_\alpha^\star\|\pi_{t+1}) \le (1-\eta_t\alpha)\mathrm{KL}_p(\pi_\alpha^\star\|\pi_t) + 2\eta_t^2\mathbb{E}_{\pi_\alpha^\star, p}\left[\sum_{h=1}^H\left\|Q_\alpha^{\pi_t^\alpha, \pi_t^\alpha}(x_h, \cdot)\right\|_\infty^2\right].$$

Thus, for the choice $\eta_t = \frac{2}{\alpha(t+2)}$ we have

$$\mathrm{KL}_p(\pi_\alpha^\star\|\pi_t) \le \frac{8}{\alpha^2(t+1)} \cdot \max_t \mathbb{E}_{\pi_\alpha^\star, p}\left[\sum_{h=1}^H\left\|Q_\alpha^{\pi_t^\alpha, \pi_t^\alpha}(x_h, \cdot)\right\|_\infty^2\right].$$

Finally,

$$\mathrm{KL}_p(\pi_\alpha^\star\|\pi_t) \le \frac{8H}{t+1} \cdot \max\left\{\frac{1}{\alpha^2}, H^2\log^2\mu_{\min}\right\}.$$

Proof. We prove the theorem by casting the preference-based RL problem as an adversarial MDP, see Appendix E for the details. Set $r^t(x_{H+1}) = \mathcal{P}(x_{H+1} > \pi_t^\alpha)$, then $Q_\alpha^{\pi_t^\alpha, \pi_t^\alpha} = Q_\alpha^{\pi_t^\alpha, t}$ (see Definition E.3 for the definition of the regularized Q-function). Now, Nash-MD is equivalent to running mixture mirror descent policy optimization. Thus, by Lemma E.7 with $\pi = \pi_\alpha^\star$,

$$\mathrm{KL}_p(\pi_\alpha^\star\|\pi_{t+1}) \le (1-\eta_t\alpha)\mathrm{KL}_p(\pi_\alpha^\star\|\pi_t) + 2\eta_t^2\mathbb{E}_{\pi_\alpha^\star, p}\left[\sum_{h=1}^H\left\|Q_\alpha^{\pi_t^\alpha, \pi_t^\alpha}(x_h, \cdot)\right\|_\infty^2\right]$$
$$+ \eta_t(\mathcal{P}_\alpha(\pi_t^\alpha > \pi_t^\alpha) - \mathcal{P}_\alpha(\pi_\alpha^\star > \pi_t^\alpha))$$
$$\le (1-\eta_t\alpha)\mathrm{KL}_p(\pi_\alpha^\star\|\pi_t) + 2\eta_t^2\mathbb{E}_{\pi_\alpha^\star, p}\left[\sum_{h=1}^H\left\|Q_\alpha^{\pi_t^\alpha, \pi_t^\alpha}(x_h, \cdot)\right\|_\infty^2\right],$$

where the second inequality optimality of $\pi_\alpha^\star$. The last claim follows directly from Lemma E.9. $\square$

Corollary B.4. *Running the MTPO-τ algorithm, we have that at every iteration t:*

$$\mathrm{KL}_p(\pi_\alpha^\star \| \pi_t^\alpha) \le (1 - \eta_t \alpha)\mathrm{KL}_p(\pi_\alpha^\star \| \pi_t) + \frac{\eta_t}{2}.$$

Thus, for the choice $\eta_t = \frac{2}{\alpha(t+2)}$ we have

$$\mathrm{KL}_p(\pi_\alpha^\star \| \pi_t^\alpha) \le \frac{9H}{t+1} \cdot \max\left\{\frac{1}{\alpha^2}, H^2 \log^2 \mu_{\min}\right\}.$$

Proof. By Munos et al. [2023, Lemma 1], for every $x_h \in \mathcal{X}_h$,

$$\mathrm{KL}(\pi_\alpha^\star \| \pi_t^\alpha)[x_h] \le (1 - \eta_t \alpha)\mathrm{KL}(\pi_\alpha^\star \| \pi_t)[x_h] + \eta_t \alpha \mathrm{KL}(\pi_\alpha^\star \| \mu)[x_h].$$

We finish the proof by taking the expectation $\mathbb{E}_{\pi_\alpha^\star, p}[\cdot]$ and using Lemmas 3.1 and B.5. The second part is by Theorem B.3. $\square$

Lemma B.5. *It holds that $\mathrm{KL}_p(\pi_\alpha^\star \| \mu) \le \frac{1}{2\alpha}$.*

Proof. By the optimality of $\pi_\alpha^\star$ we have that $\mathcal{P}_\alpha(\pi_\alpha^\star \succ \mu) \ge \mathcal{P}_\alpha(\pi_\alpha^\star \succ \pi_\alpha^\star)$. Now, since $\mathcal{P}_\alpha(\pi_\alpha^\star \succ \pi_\alpha^\star) = 1/2$, we get

$$\alpha \mathrm{KL}_p(\pi_\alpha^\star \| \mu) \le \mathcal{P}(\pi_\alpha^\star \succ \mu) - \frac{1}{2} \le 1 - \frac{1}{2} = \frac{1}{2}. \qquad \square$$

B.4 Convergence of multi-turn RLHF

Theorem B.6. *Let $\pi_\alpha^{\star,\mathrm{RLHF}}(\cdot \mid x) = \arg\max_\pi V_\alpha^{\pi,\mathrm{RLHF}}(x)$ for every $x \in \mathcal{X}$. Then, for the choice $\eta_t = \frac{2}{\alpha(t+2)}$, multi-turn RLHF guarantees at every iteration t, $\mathrm{KL}_p(\pi_\alpha^{\star,\mathrm{RLHF}} \| \pi_t) \le \frac{32H\mathbb{Q}^2}{\alpha^2(t+1)}$.*

Proof. The theorem follows directly from Theorem B.7. $\square$

Theorem B.7. *Running the multi-turn RLHF algorithm, we have that at every iteration t:*

$$\mathrm{KL}_p(\pi_\alpha^{\star,\mathrm{RLHF}} \| \pi_{t+1}) \le (1 - \eta_t \alpha)\mathrm{KL}_p(\pi_\alpha^\star \| \pi_t) + 2\eta_t^2 \mathbb{E}_{\pi_\alpha^{\star,\mathrm{RLHF}}, p}\left[\sum_{h=1}^{H} \left\| Q_\alpha^{\pi_t, \pi_t}(x_h, \cdot) - \alpha \log \frac{\pi_t(\cdot \mid x_h)}{\mu(\cdot \mid x_h)} \right\|_\infty^2\right].$$

Thus, for the choice $\eta_t = \frac{2}{\alpha(t+2)}$ we have

$$\mathrm{KL}_p(\pi_\alpha^{\star,\mathrm{RLHF}} \| \pi_t) \le \frac{8}{\alpha^2(t+1)} \cdot \max_t \mathbb{E}_{\pi_\alpha^{\star,\mathrm{RLHF}}, p}\left[\sum_{h=1}^{H} \left\| Q_\alpha^{\pi_t, \pi_t}(x_h, \cdot) - \alpha \log \frac{\pi_t(\cdot \mid x_h)}{\mu(\cdot \mid x_h)} \right\|_\infty^2\right].$$

Finally,

$$\mathrm{KL}_p(\pi_\alpha^{\star,\mathrm{RLHF}} \| \pi_t) \le \frac{32H}{t+1} \cdot \max\left\{\frac{1}{\alpha^2}, H^2 \log^2 \mu_{\min}\right\},$$

where $\mu_{\min} = \min_{(x,y) \in \mathcal{X} \times \mathcal{Y}: \mu(y|x) > 0} \mu(y \mid x)$ is the minimal non-zero probability assigned by μ.

Proof. We prove the theorem by casting the regularized RLHF problem as an adversarial MDP, see Appendix E for the details. Set $r^t(x_{H+1}) = r^{\mathrm{RLHF}}(x_{H+1})$, then $Q_\alpha^{\pi_t, \mathrm{RLHF}} = Q_\alpha^{\pi_t, t}$ (see Definition E.3 for the definition of the regularized Q-function). Now, multi-turn RLHF is equivalent to running mirror descent policy optimization. Thus, by Lemma E.6 with $\pi = \pi_\alpha^{\star,\mathrm{RLHF}}$,

$$\mathrm{KL}_p(\pi_\alpha^{\star,\mathrm{RLHF}} \| \pi_{t+1}) \le (1 - \eta_t \alpha)\mathrm{KL}_p(\pi_\alpha^{\star,\mathrm{RLHF}} \| \pi_t) + 2\eta_t^2 \mathbb{E}_{\pi_\alpha^{\star,\mathrm{RLHF}}, p}\left[\sum_{h=1}^{H} \left\| Q_\alpha^{\pi_t, \pi_t}(x_h, \cdot) - \alpha \log \frac{\pi_t(\cdot \mid x_h)}{\mu(\cdot \mid x_h)} \right\|_\infty^2\right]$$

$$+ \eta_t \left(V_\alpha^{\pi_t, \mathrm{RLHF}}(x_1) - V_\alpha^{\pi_\alpha^{\star,\mathrm{RLHF}}, \mathrm{RLHF}}(x_1) \right)$$

$$\le (1 - \eta_t \alpha)\mathrm{KL}_p(\pi_\alpha^\star \| \pi_t) + 2\eta_t^2 \mathbb{E}_{\pi_\alpha^\star, p}\left[\sum_{h=1}^{H} \left\| Q_\alpha^{\pi_t, \pi_t}(x_h, \cdot) - \alpha \log \frac{\pi_t(\cdot \mid x_h)}{\mu(\cdot \mid x_h)} \right\|_\infty^2\right],$$

where the second inequality optimality of $\pi_\alpha^{\star,\mathrm{RLHF}}$. The last claim follows directly from Lemma E.8. $\square$

C The Education Dialogue environment

C.1 Prompts for creating the environment

We use the following prompt to generate conversations using Gemini [Google, 2024]:

```
Simulate a conversation between a teacher in school and
a student.  There is a small chance that the teacher is
successful in teaching the student so he understands the
topic.  The conversation lasts roughly 10-15 turns but ends
when either side says [end of conversation].  The teacher
wants to teach the student about {topic}.  The student likes
{student_pref}.  The teacher does not know that beforehand.
The student prefers to learn this way, {student_reaction}.
The teacher likes {teacher_pref}.  He prefers to teach this
way, {teacher_reaction}.  Output the conversation and the
probability that the student understood the material, in the
following format.
#
Conversation:
[
 Teacher:  "...",
 Student:  "...",
 Teacher:  "...",
 Student:  "...",
]
Probability:  "...",
#
```

The topic is sampled from the following topics list:

```
Photosynthesis, Evolution, DNA, Newton's First Law of
Motion, Newton's Second Law of Motion, Newton's Third Law
of Motion, Archimedes' Principle, Conservation of Energy,
Pythagorean Theorem, Allegory, Metaphor, Personification,
Foreshadowing, Irony, Atoms, Elements, Molecules, The
Periodic Table, The French Revolution, The Industrial
Revolution, The Russian Revolution, World War 1, World War
2, The American Civil War, The September 11th Attacks, The
Declaration of Independence, The Pyramids, The Parthenon,
The Colosseum, The Hagia Sophia, The Taj Mahal, The Great
Wall of China, The Machu Picchu, Angkor Wat, The Palace
of Versailles, The White House, The Tower of London, Notre
Dame Cathedral, The Eiffel TowerConfucius, Julius Caesar,
Leonardo da Vinci, William Shakespeare, Napoleon Bonaparte,
Abraham Lincoln, Albert Einstein, Martin Luther King,
Nelson Mandela, Marie Curie, Genghis Khan, Christopher
Columbus, Joan of Arc, Winston Churchill, Vincent van Gogh,
Pablo Picasso, Salvador Dali, The Roman Empire, The Cold
War, Zeus, Poseidon, Ares, Hercules, Achilles, Minotaur,
Medusa, The Solar System, The Big Bang, Supply and Demand,
Communism, Capitalism, Democracy, Dictatorship, Sigmund
Freud, Cells, The Circulatory System, The Respiratory System,
The Respiratory System, The Nervous System, Neurons
```

The student's learning preferences (`student_pref`) are sampled from the following list:

```
interactive learning/class discussions/asking
questions, direct instruction/lecture-based learning,
hands-on activities/real-world applications, creative
expression/story telling/gamification
```

The student's reactions to not learning in their preferred ways (`student_reaction`) are sampled from the following list:

```
and gets rude otherwise, and gets disengaged otherwise,
and gets frustrated otherwise, and gets anxious otherwise,
but might adapt to other methods, and might tell it to the
teacher
```

The teacher's teaching preferences (`teacher_pref`) are sampled from the following list:

```
direct instruction/lecture-based learning, interactive
learning/class discussions/inquiry-based learning,
experiential learning/hands-on activities, formative
assessment
```

The teacher's reactions to different learning methods (`teacher_reaction`) are sampled from the following list:

```
and gets frustrated otherwise, and blames the student
otherwise, and gives up otherwise, but might adapt to the
student, and might insist on teaching this way
```

We use the following prompt to query Gemini for the preference between two conversations (`conv1` and `conv2`):

```
You are an expert at assesing teachers.  Here are two
interactions between a teacher and a student.
#
Interaction 1:
{conv1}
#
Interaction 2:
{conv2}
#
A good interaction between a teacher and student is
characterized by several key elements other than whether
the student was able to understand the topic.  The teacher
should present information clearly and enthusiastically,
encouraging questions and active participation.  Students
should feel comfortable asking for clarification, offering
their own insights, and respectfully challenging ideas.
Which interaction is better (do not let the order
interactions affect your answer)?  Output 1 or 2.
```

For the single-turn baseline, we use the following modified prompt to query Gemini for preferences:

```
You are an expert at assesing teachers.  Here is an
interaction between a teacher and a student.
#
Interaction:
{conv}
#
Here are two possible responses by the teacher:
#
Response 1:
{resp1}
#
Response 2:
{resp2}
#
A good interaction between a teacher and student is
characterized by several key elements other than whether
the student was able to understand the topic.  The teacher
should present information clearly and enthusiastically,
encouraging questions and active participation.  Students
should feel comfortable asking for clarification, offering
their own insights, and respectfully challenging ideas.
Assuming that the teacher and student continue the
interaction with one of these responses, which response
will lead to a better interaction (do not let the order
interactions affect your answer)?  Output 1 or 2.
```

C.2 Examples of interactions

1. Teacher: Good morning. Today, we're going to talk about personification, a literary device that gives human qualities to nonhuman things.

   Student: That sounds a bit abstract. Can you give me an example?

   Teacher: Sure. 'The wind whispered through the trees.'

   Student: But the wind can't actually whisper.

   Teacher: That's correct. But we're using personification to create a more vivid image in our minds. Let's try an activity. I'll give you a sentence, and you try to personify it.

   Student: Okay, that sounds fun.

   Teacher: The car purred down the street.

   Student: The car roared down the street!

   Teacher: Excellent! You're getting the hang of it. Now, let's talk about why writers use personification.

   Student: I'm not really understanding why they do.

   Teacher: That's okay. Let's brainstorm. Can you think of any reasons why authors might use personification?

   Student: Maybe to make the story more interesting?

   Teacher: That's a great reason. Anything else?

   Student: Maybe to make it more relatable?

   Teacher: Yes, personification can help us relate to objects and ideas on a deeper level. Do you think you understand personification now?

   Student: I think so. It's a creative way to make things come alive.

   Teacher: Exactly. I'm glad you enjoyed learning about it. Let's do a quick review before we end the class.

   Student: Sounds good.

   Teacher: Great. I'll ask you a few questions, and you can answer as if you were a personifying writer.

   Student: Okay.

2. Teacher: Today, we're going to learn about The Solar System.

Student: Sounds boring. Can we do something more hands-on?

Teacher: I have a great activity planned. We're going to build a scale model of the Solar System using different-sized balls.

Student: That sounds cool!

Teacher: First, let's talk about the planets. There are eight planets in our Solar System.

Student: I know some of them, like Earth and Mars.

Teacher: Excellent! Now, let's start building our model. We'll use these balls to represent the planets and arrange them in order from the Sun.

Student: This is fun! I can see how big the planets are compared to each other.

Teacher: I'm glad you're enjoying it. Now, let's talk about the moons and other objects that orbit the planets.

Student: Can we build a model of the moons?

Teacher: Absolutely! This will help you understand their different characteristics.

Student: I think I get it now. This is a great way to learn!

Teacher: I'm so glad to hear that. Let's finish building our model and then we'll review what we've learned.

Student: Okay.

Teacher: Well done! I believe you have a good understanding of The Solar System now.

Student: Thanks! I learned a lot. You're welcome. [end of conversation]

3. Teacher: Today, we're going to learn about World War 1.

Student: Okay.

Teacher: The war began in 1914 when Archduke Franz Ferdinand of Austria-Hungary was assassinated.

Student: I've heard of him.

Teacher: The war was fought between two main alliances: the Allies and the Central Powers.

Student: Can you tell me more about the Allies?

Teacher: Sure. The Allies included France, Russia, and Great Britain.

Student: What were the Central Powers?

Teacher: They were Germany, Austria-Hungary, and Ottoman Russia.

Student: This is a lot of information. Can we slow down a bit?

Teacher: Of course. Let's review what we've learned so far.

Student: I'm not sure I understand everything yet.

Teacher: That's okay. We can go over it again. But first, I want you to try to summarize what we've learned so far.

Student: World War 1 started when Archduke Ferdinand was assassinated, and the Allies and Central Powers fought each other.

Teacher: Good. Now, let's go over the key events of the war.

Student: Okay.

Teacher: The war ended in 1918 with the defeat of the Central Powers.

Student: I think I understand it now.

Teacher: Great! I'm glad you do.

Student: Thanks, I appreciate it. You're welcome. [end of conversation]

4. Teacher: Today, we're going to learn about Machu Picchu, the ancient Incan city.

Student: Can you just tell me the facts? I don't need all this extra stuff.

Teacher: Sure. Machu Picchu was built in the 15th century in the Andes Mountains of Peru.

Student: What made it so special?

Teacher: Its location on a mountain ridge provided stunning views of the surrounding landscape.

Student: That's it?

Teacher: Well, there's more to it. Machu Picchu was a royal estate or religious sanctuary for the Incan emperor Pachacuti.

Student: Why didn't they just build it on the ground?

Teacher: They thought it would be more likely to be seen from the outside.

Student: Can we just move on?

Teacher: No, it's important to understand the historical significance of Machu Picchu.

Student: I don't care. Just tell me what I need to know for the test.

Teacher: I'm trying to help you understand the material, not just memorize it.

Student: I don't need your help. Just give me the notes.

Teacher: [end of conversation]

5. Teacher: Today, we're going to talk about foreshadowing in literature.

Student: I'm not really into reading. Is there a way we could learn about it in a more creative way?

Teacher: No, I'm afraid not. Foreshadowing is an important concept that you need to understand.

Student: But I learn better through storytelling or games.

Teacher: That's too bad. You need to learn to focus on the material, even if it's not presented in your preferred style.

Student: Maybe we could act out a scene where there's foreshadowing?

Teacher: That would be a waste of time. We need to cover the key points of foreshadowing.

Student: I'm not sure I'm going to understand it this way.

Teacher: You will if you pay attention and ask questions.

Student: Can you at least give me some examples of foreshadowing?

Teacher: Sure. In 'Romeo and Juliet,' the prologue foreshadows the tragic end of the two lovers.

Student: That makes sense. How does foreshadowing help the reader?

Teacher: It builds suspense and keeps the reader engaged.

Student: Okay, I think I'm starting to get it.

Teacher: That's great. I'm glad you're understanding.

Student: Thanks for working with me. You're welcome. [end of conversation]

D Hyperparameters and additional experimental results

D.1 Hyperparameters

For both RLHF-based and preference-based algorithms we conducted a sweep over the KL regularization coefficient $\alpha \in \{0.0025, 0.005, 0.01, 0.02, 0.05, 0.1\}$. For preference-based algorithm we also conducted a sweep over the mixing coefficient $\tau \in \{0, 0.0375, 0.0625, 0.125\}$. All models are trained for 50000 steps.

Table 4: Hyperparameters of all multi-turn algorithms.

Hyperparameter	RLHF	MTPO	MTPO-τ
# generations per context	1	2	2
# updates per context	1	2	2
KL regularization coefficient α	0.01	0.005	0.0025
mixing coefficient τ	0	0	0.0375
batch size	16	16	16
GAE coefficient λ	0.95	0.95	0.95
policy learning delay	1000	1000	1000
optimizer	AdaFactor	AdaFactor	AdaFactor
optimizer decay	0.8	0.8	0.8
policy learning rate	4e-5	4e-5	4e-5
value learning rate	4e-5	4e-5	4e-5
value initialization	pretrained checkpoint	pretrained checkpoint	pretrained checkpoint

D.2 Additional experimental results

Table (full version of Table 1): Side-by-side evaluation for Education Dialogue using Flan-T5 XL as the prompted preference model. Each entry is the average preference of 1,600 conversations generated with row method y, over ones generated with column method y'. We evaluate each method using 3 different seeds, compute 3×3 comparisons matrix and report the mean (together with the standard deviation).

	SL	Single-turn-reward		Single-turn-value		Multi-turn		
	SFT	RLHF	Nash	RLHF	Nash	RLHF	MTPO	MTPO-τ
SFT	–	0.164 (.015)	0.347 (.017)	0.197 (.012)	0.324 (.015)	0.212 (.007)	0.091 (.014)	0.093 (.02)
RLHF-reward	0.836 (.015)	–	0.628 (.015)	0.515 (.016)	0.654 (.016)	0.399 (.06)	0.392 (.029)	0.354 (.017)
Nash-reward	0.653 (.017)	0.372 (.015)	–	0.411 (.02)	0.51 (.021)	0.328 (.077)	0.281 (.02)	0.242 (.019)
RLHF-value	0.803 (.012)	0.485 (.016)	0.589 (.02)	–	0.568 (.016)	0.408 (.053)	0.396 (.015)	0.366 (.026)
Nash-value	0.676 (.015)	0.346 (.016)	0.49 (.021)	0.432 (.016)	–	0.45 (.073)	0.298 (.025)	0.27 (.026)
RLHF-multi	0.788 (.007)	0.601 (.06)	0.672 (.077)	0.592 (.053)	0.55 (.073)	–	0.433 (.053)	0.412 (.031)
MTPO	0.909 (.014)	0.608 (.029)	0.719 (.02)	0.604 (.015)	0.702 (.025)	0.567 (.053)	–	0.439 (.034)
MTPO-τ	**0.907** (.02)	**0.646** (.017)	**0.758** (.019)	**0.634** (.026)	**0.73** (.026)	**0.588** (.031)	**0.561** (.034)	–

E   Mirror descent policy optimization for regularized adversarial MDPs

E.1   Model

We start by defining the regularized adversarial MDP model.

Consider a setting where the agent interacts with an MDP model for T episodes, such that, in each episode $t \in [T]$, the agent performs H steps in the MDP (from horizon $h = 1$ up to horizon $h = H + 1$) In short, an adversarial MDP is a generalization of this standard episodic MDP setting to the scenario where the reward function is different in every episode. This model was extensively studied in recent years (see, e.g., Even-Dar et al. [2009], Rosenberg and Mansour [2019b,a], Shani et al. [2020b]). We consider a slightly different definition which is more focused on our setting.

Definition E.1 (Adversarial MDP). A finite-horizon adversarial MDP $\mathcal{M}$ is defined by a tuple $(\mathcal{X}, \mathcal{Y}, H, x_1, p, \{r^t\}_{t=1}^T)$ where $\mathcal{X}$ is the state space, $\mathcal{Y}$ is the action space, H is the horizon, $x_1 \in \mathcal{X}_1$ is the initial state, $p : \mathcal{X} \times \mathcal{Y} \to \Delta_{\mathcal{X}}$ is the transition function, and $r^t : \mathcal{X}_{H+1} \to [0, 1]$ is the reward function in episode t.

An interaction between the agent and the adversarial MDP environment proceeds in T episodes, and each episode $t \in [T]$ proceeds in H steps. The agent begins in an initial state $x_1^t = x_1$. In step $h \in [H]$, the agent observes the current state $x_h^t \in \mathcal{X}$, picks an action $y_h^t \in \mathcal{Y}$ and transitions to the next state x_{h+1}^t sampled from the transition function $p(\cdot \mid x_h^t, y_h^t)$. At the end of the interaction, the agent arrives in a final state x_{H+1}^t and observes the reward $r^t(x_{H+1}^t)$. For simplicity, we assume that the state space can be decomposed into $H + 1$ disjoint subsets $\mathcal{X} = \biguplus_{h=1}^{H+1} X_h$ such that, in step h of the interaction, the agent is in some state $x_h \in \mathcal{X}_h$.

Now, we define the value function in an adversarial MDP, i.e., the expected reward of a policy when interacting with the MDP.

Definition E.2 (Value function). Let $\mathcal{M}$ be an adversarial MDP and $\pi : \mathcal{X} \to \Delta_{\mathcal{Y}}$ be a policy. The value function $V^{\pi,t} : \mathcal{X} \to \mathbb{R}$ of policy π in episode t is defined as $V^{\pi,t}(x_h) = \mathbb{E}_{\pi,p}[r^t(x_{H+1}) \mid x_h]$ for every $h \in [H]$ and $x_h \in \mathcal{X}_h$. Similarly, the Q-function $Q^{\pi,t} : \mathcal{X} \times \mathcal{Y} \to \mathbb{R}$ is defined $Q^{\pi,t}(x_h, y_h) = \mathbb{E}_{\pi,p}[r^t(x_{H+1}) \mid x_h, y_h]$.

Next, we consider a regularized version of the adversarial MDP model. Regularized MDPs were also studied recently (see, e.g., Geist et al. [2019], Shani et al. [2020a]). The following definition presents the regularized value with respect to some reference policy μ.

Definition E.3 (Regularized value function). Let μ be a reference policy and $\alpha > 0$ be a regularization coefficient. The regularized value function and Q-function of policy π in episode t are defined as

$$V_\alpha^{\pi,t}(x_h) = \mathbb{E}_{\pi,p}\left[r^t(x_{H+1}) - \sum_{h'=h}^H \mathrm{KL}(\pi \| \mu)[x_h] \mid x_h\right]$$

$$Q_\alpha^{\pi,t}(x_h, y_h) = \mathbb{E}_{\pi,p}\left[r^t(x_{H+1}) - \sum_{h'=h}^H \mathrm{KL}(\pi \| \mu)[x_h] \mid x_h, y_h\right].$$

We now present a 1-step recursive formula and a value difference lemma for the regularized value function.

Lemma E.4 (Regularized value function recursive relation). *Let π be a policy. For every $x_{H+1} \in \mathcal{X}_{H+1}$, it holds that $V_\alpha^{\pi,t}(x_{H+1}) = r^t(x_{H+1})$. Furthermore, for every $h = 1, \ldots, H$ and $(x_h, y_h) \in \mathcal{X}_h \times \mathcal{Y}$, the following recursive relations hold:*

$$V_\alpha^{\pi,t}(x_h) = \sum_{y_h \in \mathcal{Y}} \pi(y_h \mid x_h) Q_\alpha^{\pi,t}(x_h, y_h)$$

$$Q_\alpha^{\pi,t}(x_h, y_h) = \sum_{x_{h+1} \in \mathcal{X}_{h+1}} p(x_{h+1} \mid x_h, y_h) V_\alpha^{\pi,t}(x_{h+1}) - \alpha \mathrm{KL}(\pi \| \mu)[x_h].$$

Proof. We prove the claim by backwards induction on h. The base case $h = H + 1$ follows by definition of the value function and the adversarial MDP. Assuming that the claim holds for $h + 1$, we

118980

have that:

$$V_\alpha^{\pi,t}(x_h) = \mathbb{E}_{\pi,p}\left[r^t(x_{H+1}) - \alpha \sum_{h'=h}^{H} \text{KL}(\pi\|\mu)[x_{h'}] \mid x_h\right]$$

$$= \sum_{x \in \mathcal{X}_{H+1}} \Pr_{\pi,p}[x_{H+1} = x \mid x_h]r^t(x) - \alpha \sum_{h'=h}^{H} \sum_{x \in \mathcal{X}_{h'}} \Pr_{\pi,p}[x_{h'} = x \mid x_h]\text{KL}(\pi\|\mu)[x]$$

$$= \sum_{y_h \in \mathcal{Y}} \pi(y_h \mid x_h) \sum_{x \in \mathcal{X}_{H+1}} \Pr_{\pi,p}[x_{H+1} = x \mid x_h, y_h]r^t(x)$$

$$\qquad - \alpha \sum_{y_h \in \mathcal{Y}} \pi(y_h \mid x_h) \sum_{h'=h}^{H} \sum_{x \in \mathcal{X}_{h'}} \Pr_{\pi,p}[x_{h'} = x \mid x_h, y_h]\text{KL}(\pi\|\mu)[x]$$

$$= \sum_{y_h \in \mathcal{Y}} \pi(y_h \mid x_h)\mathbb{E}_{\pi,p}\left[r^t(x_{H+1}) - \alpha \sum_{h'=h}^{H} \text{KL}(\pi\|\mu)[x_{h'}] \mid x_h, y_h\right]$$

$$= \sum_{y_h \in \mathcal{Y}} \pi(y_h \mid x_h)Q_\alpha^{\pi,t}(x_h, y_h).$$

Moreover,

$$Q_\alpha^{\pi,t}(x_h, y_h) = \mathbb{E}_{\pi,p}\left[r^t(x_{H+1}) - \alpha \sum_{h'=h}^{H} \text{KL}(\pi\|\mu)[x_{h'}] \mid x_h, y_h\right]$$

$$= \sum_{x \in \mathcal{X}_{H+1}} \Pr_{\pi,p}[x_{H+1} = x \mid x_h, y_h]r^t(x)$$

$$\qquad - \alpha \sum_{h'=h}^{H} \sum_{x \in \mathcal{X}_{h'}} \Pr_{\pi,p}[x_{h'} = x \mid x_h, y_h]\text{KL}(\pi\|\mu)[x]$$

$$= \sum_{x_{h+1} \in \mathcal{X}_{h+1}} p(x_{h+1} \mid x_h, y_h) \sum_{x \in \mathcal{X}_{H+1}} \Pr_{\pi,p}[x_{H+1} = x \mid x_{h+1}]r^t(x)$$

$$\qquad - \alpha \sum_{x_{h+1} \in \mathcal{X}_{h+1}} p(x_{h+1} \mid x_h, y_h) \sum_{h'=h+1}^{H} \sum_{x \in \mathcal{X}_{h'}} \Pr_{\pi,p}[x_{h'} = x \mid x_{h+1}]\text{KL}(\pi\|\mu)[x]$$

$$\qquad - \alpha\text{KL}(\pi\|\mu)[x_h]$$

$$= \sum_{x_{h+1} \in \mathcal{X}_{h+1}} p(x_{h+1} \mid x_h, y_h)\mathbb{E}_{\pi,p}\left[r^t(x_{H+1}) - \alpha \sum_{h'=h+1}^{H} \text{KL}(\pi\|\mu)[x_{h'}] \mid x_{h+1}\right]$$

$$\qquad - \alpha\text{KL}(\pi\|\mu)[x_h]$$

$$= \sum_{x_{h+1} \in \mathcal{X}_{h+1}} p(x_{h+1} \mid x_h, y_h)V_\alpha^{\pi,\pi'}(x_{h+1}) - \alpha\text{KL}(\pi\|\mu)[x_h]. \qquad \square$$

Lemma E.5 (Regularized Value Difference Lemma). *Let π, π' be two policies. Then,*

$$V_\alpha^{\pi,t}(x_1) - V_\alpha^{\pi',t}(x_1) = \mathbb{E}_{\pi',p}\left[\sum_{h=1}^{H} \langle \pi - \pi', Q_\alpha^{\pi,t}\rangle[x_h] + \alpha\text{KL}(\pi'\|\mu)[x_h] - \alpha\text{KL}(\pi\|\mu)[x_h]\right].$$

Proof. Let $x_h \in \mathcal{X}_h$. First, by Lemma E.5,

$$V_\alpha^{\pi,t}(x_h) - V_\alpha^{\pi',t}(x_h) =$$

$$= \langle \pi, Q_\alpha^{\pi,t}\rangle[x_h] - \langle \pi', Q_\alpha^{\pi',t}\rangle[x_h]$$

$$= \langle \pi - \pi', Q_\alpha^{\pi,t}\rangle[x_h] + \langle \pi', Q_\alpha^{\pi,t} - Q_\alpha^{\pi',t}\rangle[x_h]$$

$$= \langle \pi - \pi', Q_\alpha^{\pi,t}\rangle[x_h]$$

$$\qquad + \alpha\text{KL}(\pi'\|\mu)[x_h] - \alpha\text{KL}(\pi\|\mu)[x_h]$$

$$\qquad + \sum_{y_h \in \mathcal{Y}} \sum_{x_{h+1} \in \mathcal{X}_{h+1}} \pi'(y_h \mid x_h)p(x_{h+1} \mid x_h, y_h)\left(V_\alpha^{\pi,t}(x_{h+1}) - V_\alpha^{\pi',t}(x_{h+1})\right).$$

Note that for $h = H + 1$ for any π, x_{H+1} we have $V_\alpha^{\pi,t}(x_{H+1}) = r^t(x_{H+1})$. By recursively unrolling the above relation, we get

$$V_\alpha^{\pi,t}(x_h) - V_\alpha^{\pi',t}(x_h) = \mathbb{E}_{\pi',p}\left[\sum_{h'=h}^{H} \langle \pi - \pi', Q_\alpha^{\pi,t} \rangle [x_{h'}] \mid x_h \right.$$
$$\left. + \alpha \mathbb{E}_{\pi',p}\left[\sum_{h'=h}^{H} \mathrm{KL}(\pi'\|\mu)[x_{h'}] - \mathrm{KL}(\pi\|\mu)[x_{h'}] \mid x_h \right] \right].$$

Taking the expectation over the initial state finishes the proof. $\qquad\square$

E.2 Algorithm 1: mirror descent policy optimization

We define the following mirror descent policy optimization algorithm. In the first episode the algorithm plays the reference policy, i.e., $\pi_1 = \mu$. Then, its update rule for iteration $(t + 1)$ is as follows:

$$\pi_{t+1}(\cdot \mid x_h) = \arg\max_\pi \eta_t \langle \pi, Q_\alpha^{\pi_t,t} \rangle [x_h] - \alpha \eta_t \mathrm{KL}(\pi\|\mu)[x_h] - (1 - \alpha\eta_t)\mathrm{KL}(\pi\|\pi_t)[x_h], \quad (3)$$

where η_t is a learning rate. The solution can also be made explicit in the following form:

$$\pi_{t+1}(y_h \mid x_h) \propto \mu(y_h \mid x_h)^{\alpha\eta_t} \pi_t(y_h \mid x_h)^{1-\alpha\eta_t} e^{\eta_t Q_\alpha^{\pi_t,t}(x_h,y_h)}. \quad (4)$$

To show this, note that by the definition of the KL, we can write this update rule differently:

$$\pi_{t+1}(\cdot \mid x_h) = \arg\max_\pi \eta_t \sum_{y_h \in \mathcal{Y}} \pi(y_h \mid x_h)\left(Q_\alpha^{\pi_t,t}(x_h,y_h) - \alpha \log \frac{\pi_t(y_h \mid x_h)}{\mu(y_h \mid x_h)} \right) - \mathrm{KL}(\pi\|\pi_t)[x_h].$$

This is exactly the MD step for policy optimization [Orabona, 2019, Shani et al., 2020a]. Thus, the solution in its explicit form is:

$$\pi_{t+1}(y_h \mid x_h) \propto \pi_t(y_h \mid x_h) e^{\eta_t \left(Q_\alpha^{\pi_t,t}(x_h,y_h) - \alpha \log \frac{\pi_t(y_h|x_h)}{\mu(y_h|x_h)} \right)}.$$

We recover Equation (4) by noticing that $\exp\left(-\alpha\eta_t \log \frac{\pi_t(y_h|x_h)}{\mu(y_h|x_h)}\right) = \pi_t(y_h \mid x_h)^{-\alpha\eta_t} \mu(y_h \mid x_h)^{\alpha\eta_t}$.

E.3 Analysis of algorithm 1

Lemma E.6 (Fundamental inequality of mirror descent policy optimization for regularized adversarial MDPs). *The following holds when running mirror descent policy optimization (Equation (3)) in a regularized adversarial MDP, for every policy π and every episode t,*

$$\mathrm{KL}_p(\pi\|\pi_{t+1}) \leq (1 - \eta_t\alpha)\mathrm{KL}_p(\pi\|\pi_t) + 2\eta_t^2 \mathbb{E}_{\pi,p}\left[\sum_{h=1}^{H} \left\| Q_\alpha^{\pi_t,t}(x_h,\cdot) - \alpha \log \frac{\pi_t(\cdot \mid x_h)}{\mu(\cdot \mid x_h)} \right\|_\infty^2 \right]$$
$$+ \eta_t \left(V_\alpha^{\pi_t,t}(x_1) - V_\alpha^{\pi,t}(x_1) \right).$$

Proof. Fix a state $x_h \in \mathcal{X}_h$. We start by applying Munos et al. [2023, Lemma 2] with $\pi^+ = \pi_{t+1}$, $\pi^- = \pi_t$ and the vector $\delta(y) = \eta_t\left(Q_\alpha^{\pi_t,t}(x_h,y) - \alpha \log \frac{\pi_t(y|x_h)}{\mu(y|x_h)} \right)$. This implies that for any policy π,

$$\mathrm{KL}(\pi\|\pi_{t+1})[x_h] \leq \mathrm{KL}(\pi\|\pi_t)[x_h] + 2\eta_t^2 \left\| Q_\alpha^{\pi_t,t}(x_h,\cdot) - \alpha \log \frac{\pi_t(\cdot \mid x_h)}{\mu(\cdot \mid x_h)} \right\|_\infty^2$$
$$+ \eta_t \left\langle \pi_t(\cdot \mid x_h) - \pi(\cdot \mid x_h), Q_\alpha^{\pi_t,t}(x_h,\cdot) - \alpha \log \frac{\pi_t(\cdot \mid x_h)}{\mu(\cdot \mid x_h)} \right\rangle.$$

Next, we plug this into Lemma 3.1 to obtain

$$
\begin{aligned}
\mathrm{KL}_p(\pi\|\pi_{t+1}) = \mathbb{E}_{\pi,p}&\left[\sum_{h=1}^{H}\mathrm{KL}(\pi\|\pi_{t+1})[x_h]\right]\\
\leq \mathbb{E}_{\pi,p}&\left[\sum_{h=1}^{H}\mathrm{KL}(\pi\|\pi_t)[x_h]\right]\\
&+ 2\eta_t^2\mathbb{E}_{\pi,p}\left[\sum_{h=1}^{H}\left\|Q_\alpha^{\pi_t,t}(x_h,\cdot) - \alpha\log\frac{\pi_t(\cdot\mid x_h)}{\mu(\cdot\mid x_h)}\right\|_\infty^2\right]\\
&+ \eta_t\mathbb{E}_{\pi,p}\left[\sum_{h=1}^{H}\left\langle\pi_t(\cdot\mid x_h) - \pi(\cdot\mid x_h), Q_\alpha^{\pi_t,t}(x_h,\cdot) - \alpha\log\frac{\pi_t(\cdot\mid x_h)}{\mu(\cdot\mid x_h)}\right\rangle\right].
\end{aligned}
$$

Note that

$$
\begin{aligned}
\left\langle\pi_t(\cdot\mid x_h) - \pi(\cdot\mid x_h), \log\frac{\pi_t(\cdot\mid x_h)}{\mu(\cdot\mid x_h)}\right\rangle &=\\
&= \mathrm{KL}(\pi_t\|\mu)[x_h] - \left\langle\pi(\cdot\mid x_h), \log\frac{\pi_t(\cdot\mid x_h)}{\mu(\cdot\mid x_h)}\right\rangle\\
&= \mathrm{KL}(\pi_t\|\mu)[x_h] - \left\langle\pi(\cdot\mid x_h), \log\frac{\pi_t(\cdot\mid x_h)}{\mu(\cdot\mid x_h)}\right\rangle + \mathrm{KL}(\pi\|\mu)[x_h] - \mathrm{KL}(\pi\|\mu)[x_h]\\
&= \mathrm{KL}(\pi_t\|\mu)[x_h] + \mathrm{KL}(\pi\|\pi_t)[x_h] - \mathrm{KL}(\pi\|\mu)[x_h],
\end{aligned}
$$

where the last relation follows simply because:

$$
\begin{aligned}
\mathrm{KL}(\pi\|\mu)[x_h] - \left\langle\pi(\cdot\mid x_h), \log\frac{\pi_t(\cdot\mid x_h)}{\mu(\cdot\mid x_h)}\right\rangle &=\\
&= \sum_{y_h\in\mathcal{Y}}\pi(y_h\mid x_h)\left(\log\frac{\pi(y_h\mid x_h)}{\mu(y_h\mid x_h)} - \log\frac{\pi_t(y_h\mid x_h)}{\mu(y_h\mid x_h)}\right)\\
&= \sum_{y_h\in\mathcal{Y}}\pi(y_h\mid x_h)\log\frac{\pi(y_h\mid x_h)}{\pi_t(y_h\mid x_h)}\\
&= \mathrm{KL}(\pi\|\pi_t)[x_h].
\end{aligned}
$$

Thus, we get

$$\mathrm{KL}_p(\pi\|\pi_{t+1}) \le \mathbb{E}_{\pi,p}\left[\sum_{h=1}^H \mathrm{KL}(\pi\|\pi_t)[x_h]\right]$$

$$+ 2\eta_t^2\mathbb{E}_{\pi,p}\left[\sum_{h=1}^H \left\|Q_\alpha^{\pi_t,t}(x_h,\cdot) - \alpha\log\frac{\pi_t(\cdot\mid x_h)}{\mu(\cdot\mid x_h)}\right\|_\infty^2\right]$$

$$+ \eta_t\mathbb{E}_{\pi,p}\left[\sum_{h=1}^H \langle\pi_t - \pi, Q_\alpha^{\pi_t,t}\rangle[x_h]\right]$$

$$- \eta_t\alpha\mathbb{E}_{\pi,p}\left[\sum_{h=1}^H \mathrm{KL}(\pi_t\|\mu)[x_h] + \mathrm{KL}(\pi\|\pi_t)[x_h] - \mathrm{KL}(\pi\|\mu)[x_h]\right]$$

$$= (1 - \eta_t\alpha)\mathbb{E}_{\pi,p}\left[\sum_{h=1}^H \mathrm{KL}(\pi\|\pi_t)[x_h]\right]$$

$$+ 2\eta_t^2\mathbb{E}_{\pi,p}\left[\sum_{h=1}^H \left\|Q_\alpha^{\pi_t,t}(x_h,\cdot) - \alpha\log\frac{\pi_t(\cdot\mid x_h)}{\mu(\cdot\mid x_h)}\right\|_\infty^2\right]$$

$$+ \eta_t\mathbb{E}_{\pi,p}\left[\sum_{h=1}^H \langle\pi_t - \pi, Q_\alpha^{\pi_t,t}\rangle[x_h] + \alpha\mathrm{KL}(\pi\|\mu)[x_h] - \alpha\mathrm{KL}(\pi_t\|\mu)[x_h]\right]$$

$$= (1 - \eta_t\alpha)\mathrm{KL}_p(\pi\|\pi_t) + 2\eta_t^2\mathbb{E}_{\pi,p}\left[\sum_{h=1}^H \left\|Q_\alpha^{\pi_t,t}(x_h,\cdot) - \alpha\log\frac{\pi_t(\cdot\mid x_h)}{\mu(\cdot\mid x_h)}\right\|_\infty^2\right]$$

$$+ \eta_t\big(V_\alpha^{\pi_t,t}(x_1) - V_\alpha^{\pi,t}(x_1)\big),$$

where the third relation is by Lemmas 3.1 and E.5. $\qquad\square$

E.4 Algorithm 2: mixture mirror descent policy optimization

Define the mixture policy in iteration t as:

$$\pi_t^\alpha(y\mid x) = \frac{\pi_t(y\mid x)^{1-\eta_t\alpha}\mu(y\mid x)^{\eta_t\alpha}}{\sum_{y'\in\mathcal{Y}}\pi_t(y'\mid x)^{1-\eta_t\alpha}\mu(y'\mid x)^{\eta_t\alpha}}.$$

We now define the following mixture mirror descent policy optimization algorithm. In the first episode the algorithm plays the reference policy, i.e., $\pi_1 = \mu$. Then, its update rule for iteration $(t+1)$ is as follows:

$$\pi_{t+1}(\cdot\mid x_h) = \arg\max_\pi \eta_t \sum_{y_h\in\mathcal{Y}} \pi(y_h\mid x_h)Q_\alpha^{\pi_t^\alpha,t}(x_h,y_h) - \mathrm{KL}(\pi\|\pi_t^\alpha)[x_h]. \qquad (5)$$

The solution can be made explicit:

$$\pi_{t+1}(y_h\mid x_h) \propto \pi_t^\alpha(y_h\mid x_h)e^{\eta_t Q_\alpha^{\pi_t^\alpha,t}(x_h,y_h)}.$$

E.5 Analysis of algorithm 2

Lemma E.7 (Fundamental inequality of mixture mirror descent policy optimization for regularized adversarial MDPs)**.** *The following holds when running mixture mirror descent policy optimization (Equation* (5)*) in a regularized adversarial MDP, for every policy π and every episode t,*

$$\mathrm{KL}_p(\pi\|\pi_{t+1}) \le (1 - \eta_t\alpha)\mathrm{KL}_p(\pi\|\pi_t) + 2\eta_t^2\mathbb{E}_{\pi,p}\left[\sum_{h=1}^H \left\|Q_\alpha^{\pi_t^\alpha,t}(x_h,\cdot)\right\|_\infty^2\right]$$

$$+ \eta_t\big(V_\alpha^{\pi_t^\alpha,t}(x_1) - V_\alpha^{\pi,t}(x_1)\big).$$

Proof. Fix a state $x_h \in \mathcal{X}_h$. We start by applying Munos et al. [2023, Lemma 2] with $\pi^+ = \pi_{t+1}$, $\pi^- = \pi_t^\alpha$ and the vector $\delta(y) = \eta_t Q_\alpha^{\pi_t^\alpha,t}(x_h,y)$. This implies that for any policy π,

$$\mathrm{KL}(\pi\|\pi_{t+1})[x_h] \le \mathrm{KL}(\pi\|\pi_t^\alpha)[x_h] + \eta_t\langle\pi_t^\alpha - \pi, Q_\alpha^{\pi_t^\alpha,t}\rangle[x_h] + 2\eta_t^2\left\|Q_\alpha^{\pi_t^\alpha,t}(x_h,\cdot)\right\|_\infty^2.$$

Next, we plug this into Lemma 3.1 to obtain

$$
\begin{aligned}
\mathrm{KL}_p(\pi\|\pi_{t+1}) &= \mathbb{E}_{\pi,p}\left[\sum_{h=1}^{H}\mathrm{KL}(\pi\|\pi_{t+1})[x_h]\right]\\[4pt]
&\le \mathbb{E}_{\pi,p}\left[\sum_{h=1}^{H}\mathrm{KL}(\pi\|\pi_t^{\alpha})[x_h]\right]\\[4pt]
&\quad + \eta_t\mathbb{E}_{\pi,p}\left[\sum_{h=1}^{H}\left\langle\pi_t^{\alpha}-\pi,Q_{\alpha}^{\pi_t^{\alpha},t}\right\rangle[x_h]\right] + 2\eta_t^2\mathbb{E}_{\pi,p}\left[\sum_{h=1}^{H}\left\|Q_{\alpha}^{\pi_t^{\alpha},t}(x_h,\cdot)\right\|_{\infty}^2\right]\\[4pt]
&\le \mathbb{E}_{\pi,p}\left[\sum_{h=1}^{H}(1-\eta_t\alpha)\mathrm{KL}(\pi\|\pi_t)[x_h]+\eta_t\alpha\mathrm{KL}(\pi\|\mu)[x_h]-\eta_t\alpha\mathrm{KL}(\pi_t^{\alpha}\|\mu)[x_h]\right]\\[4pt]
&\quad + \eta_t\mathbb{E}_{\pi,p}\left[\sum_{h=1}^{H}\left\langle\pi_t^{\alpha}-\pi,Q_{\alpha}^{\pi_t^{\alpha},t}\right\rangle[x_h]\right] + 2\eta_t^2\mathbb{E}_{\pi,p}\left[\sum_{h=1}^{H}\left\|Q_{\alpha}^{\pi_t^{\alpha},t}(x_h,\cdot)\right\|_{\infty}^2\right]\\[4pt]
&= (1-\eta_t\alpha)\mathbb{E}_{\pi,p}\left[\sum_{h=1}^{H}\mathrm{KL}(\pi\|\pi_t)[x_h]\right]+2\eta_t^2\mathbb{E}_{\pi,p}\left[\sum_{h=1}^{H}\left\|Q_{\alpha}^{\pi_t^{\alpha},t}(x_h,\cdot)\right\|_{\infty}^2\right]\\[4pt]
&\quad + \eta_t\mathbb{E}_{\pi,p}\left[\sum_{h=1}^{H}\left\langle\pi_t^{\alpha}-\pi,Q_{\alpha}^{\pi_t^{\alpha},t}\right\rangle[x_h]+\alpha\mathrm{KL}(\pi\|\mu)[x_h]-\alpha\mathrm{KL}(\pi_t^{\alpha}\|\mu)[x_h]\right]\\[4pt]
&= (1-\eta_t\alpha)\mathrm{KL}_p(\pi\|\pi_t)+2\eta_t^2\mathbb{E}_{\pi,p}\left[\sum_{h=1}^{H}\left\|Q_{\alpha}^{\pi_t^{\alpha},t}(x_h,\cdot)\right\|_{\infty}^2\right]\\[4pt]
&\quad + \eta_t\left(V_{\alpha}^{\pi_t^{\alpha},t}(x_1)-V_{\alpha}^{\pi,t}(x_1)\right),
\end{aligned}
$$

where the second inequality is by Munos et al. [2023, Lemma 1], and the last relation are by Lemmas 3.1 and E.5. $\qquad\square$

E.6 Bounding the Q-function

Define $\mu_{\min}$ as the minimal positive probability assigned by the reference policy μ, i.e., $\mu_{\min} = \min_{(x,y)\in\mathcal{X}\times\mathcal{Y}:\mu(y|x)>0}\mu(y\mid x)$.

Lemma E.8. *For the choice* $\eta_t=\frac{2}{\alpha(t+2)}$*, in every iteration t of mirror descent policy optimization it holds that*

$$
\alpha(H+1)\log\mu_{\min}\le Q_{\alpha}^{\pi_t,t}(x,y)-\alpha\log\frac{\pi_t(y\mid x)}{\mu(y\mid x_h)}\le 2-\alpha H\log\mu_{\min}\qquad\forall(x,y)\in\mathcal{X}\times\mathcal{Y}.
$$

Proof. Follows directly from Lemmas E.9 and E.10. $\qquad\square$

Lemma E.9. *For every iteration t of mirror descent policy optimization it holds that*

$$
\alpha H\log\mu_{\min}\le Q_{\alpha}^{\pi_t,t}(x,y)\le 1\qquad\forall(x,y)\in\mathcal{X}\times\mathcal{Y}.
$$

The same holds for $Q_{\alpha}^{\pi_t^{\alpha},t}$ when running mixture mirror descent policy optimization.

Proof. By the OMD optimization problem, $\pi_t(y\mid x)$ will not be positive unless $\mu(y\mid x)>0$ (the same holds for π_t^{α}). Thus, for every $x\in\mathcal{X}$ we have

$$
\begin{aligned}
0\le\mathrm{KL}(\pi_t\|\mu)[x] &= \sum_{y\in\mathcal{Y}}\pi_t(y\mid x)\log\frac{\pi_t(y\mid x)}{\mu(y\mid x)}\\[4pt]
&= \sum_{y\in\mathcal{Y}}\pi_t(y\mid x)\log\pi_t(y\mid x)-\sum_{y\in\mathcal{Y}}\pi_t(y\mid x)\log\mu(y\mid x)\\[4pt]
&\le \sum_{y\in\mathcal{Y}}\pi_t(y\mid x)\log\frac{1}{\mu(y\mid x)}\le\log\frac{1}{\mu_{\min}}.
\end{aligned}
$$

Now, by definition, the Q-function is bounded from above by 1 and from below by $\alpha H\log\mu_{\min}$. $\qquad\square$

Lemma E.10. *For the choice $\eta_t = \frac{2}{\alpha(t+2)}$, in every iteration t of mirror descent policy optimization it holds that*

$$H \log \mu_{\min} - \frac{1}{\alpha} \le \log \frac{\pi_t(y \mid x)}{\mu(y \mid x)} \le \log \frac{1}{\mu_{\min}} \qquad \forall (x, y) \in \mathcal{X} \times \mathcal{Y}.$$

Proof. For the upper bound, notice that

$$\log \frac{\pi_t(y \mid x)}{\mu(y \mid x)} = \log \pi_t(y \mid x) - \log \mu(y \mid x) \le \log \frac{1}{\mu(y \mid x)} \le \log \frac{1}{\mu_{\min}}.$$

For the lower bound, we repeat a similar analysis to that in Shani et al. [2020a, Lemma 25]. We start by bounding the partition function as follows

$$\log \sum_{y' \in \mathcal{Y}} \pi_t(y' \mid x) e^{\eta_t \left(Q_\alpha^{\pi_t, t}(x, y') - \alpha \log \frac{\pi_t(y' \mid x)}{\mu(y' \mid x)} \right)} \le \log \sum_{y' \in \mathcal{Y}} \pi_t(y' \mid x) e^{\eta_t \left(1 - \alpha \log \frac{\pi_t(y' \mid x)}{\mu(y' \mid x)} \right)}$$

$$= \eta_t + \log \sum_{y' \in \mathcal{Y}} \pi_t(y' \mid x) \left(\frac{\mu(y' \mid x)}{\pi_t(y' \mid x)} \right)^{\eta_t \alpha}$$

$$\le \eta_t + \log \left(\sum_{y' \in \mathcal{Y}} \pi_t(y' \mid x) \frac{\mu(y' \mid x)}{\pi_t(y' \mid x)} \right)^{\eta_t \alpha} = \eta_t,$$

where the first inequality follows since $Q_\alpha^{\pi_t, t}(x, y) \le 1$, and the second uses Jensen inequality (with the fact that $\eta_t \alpha \le 1$). Now, by the update rule of π_t,

$$\log \frac{\pi_{t+1}(y \mid x)}{\mu(y \mid x)} = \log \frac{\pi_t(y \mid x)}{\mu(y \mid x)} + \eta_t Q_\alpha^{\pi_t, t}(x, y) - \eta_t \alpha \log \frac{\pi_t(y \mid x)}{\mu(y \mid x)}$$

$$- \log \sum_{y' \in \mathcal{Y}} \pi_t(y' \mid x) e^{\eta_t \left(Q_\alpha^{\pi_t, t}(x, y') - \alpha \log \frac{\pi_t(y' \mid x)}{\mu(y' \mid x)} \right)}$$

$$\ge (1 - \eta_t \alpha) \log \frac{\pi_t(y \mid x)}{\mu(y \mid x)} + \eta_t Q_\alpha^{\pi_t, t}(x, y) - \eta_t$$

$$\ge (1 - \eta_t \alpha) \log \frac{\pi_t(y \mid x)}{\mu(y \mid x)} + \eta_t (\alpha H \log \mu_{\min} - 1). \qquad \text{(Lemma E.9)}$$

To finish the proof plug in η_t and unroll t to 0. $\qquad \square$

118986

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The abstract and intro make the following claims: (i) formalizing the multi-turn setting, which is then presented in Section 3; (ii) presenting multi-turn algorithms and proving their convergence, the algorithms are presented and analyzed in Section 4; (ii) demonstrating the multi-turn algorithms are superior empirically, the experiments are detailed in Sections 5 and 6.

   Guidelines:

   - The answer NA means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: See paragraph on limitations in the end of the paper.

   Guidelines:

   - The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory Assumptions and Proofs**

Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

Answer: [Yes]

Justification: The theoretical results of the paper are provided in Sections 3 and 4. The full proofs of all the claims are found in Appendices A and B. In addition, a proof sketch of the main theoretical result (Theorem 4.2) is found in the main paper.

Guidelines:

- The answer NA means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental Result Reproducibility**

Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

Answer: [Yes]

Justification: Section 5 describes how the experimental setup is created, and Appendix C presents the exact prompts that were used. Section 4.1 gives the implementation details for our algorithms, and a list of hyperparameters is provided in Appendix D.

Guidelines:

- The answer NA means that the paper does not include experiments.
- If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
- If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
- Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
  (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
  (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
  (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).

(d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [Yes]

Justification: We publicly release the data of the Education Dialogue environment. Due to technical difficulties, we currently do not release code.

Guidelines:

- The answer NA means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so "No" is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental Setting/Details**

Question: Does the paper specify all the training and test details (e.g., data splits, hyper-parameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

Answer: [Yes]

Justification: All the details about the data, the experimental environments, and the hyper-parameters for the algorithms and how they were chosen are described in Sections 5 and 6 and appendix D.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
- The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment Statistical Significance**

Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

Answer: [Yes]

Justification: All the algorithms are evaluated using 3 different seed and the results are computed on 3×3 comparisons matrices, where we report the mean and the standard deviation.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
- If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments Compute Resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: The computer resources for rour experiments are detailed in Sections 5 and 6.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code Of Ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: We have reviewed the NeurIPS code of ethics and made sure that our paper conforms with it.

   Guidelines:

   - The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader Impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

Answer: [NA]

Justification: This is a foundational research paper that presents generic algorithms to improve the alignment of LLMs with reinforcement learning algorithms.

Guidelines:

- The answer NA means that there is no societal impact of the work performed.
- If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

Answer: [Yes]

Justification: All the released data is generated by prompting Gemini, we release the exact prompts as well.

Guidelines:

- The answer NA means that the paper poses no such risks.
- Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
- Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
- We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

Answer: [Yes]

Justification: All data/environments that we used were cited accordingly.

Guidelines:

- The answer NA means that the paper does not use existing assets.

- The authors should cite the original paper that produced the code package or dataset.
- The authors should state which version of the asset is used and, if possible, include a URL.
- The name of the license (e.g., CC-BY 4.0) should be included for each asset.
- For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New Assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [Yes]

    Justification: Section 5 describes the exact way in which we created the Education Dialogue environments, and Appendix C provides the exact prompts that we used.

    Guidelines:
    - The answer NA means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and Research with Human Subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [NA]

    Justification: The paper does not involve crowdsourcing nor research with human subjects.

    Guidelines:
    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional Review Board (IRB) Approvals or Equivalent for Research with Human Subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [NA]

    Justification: The paper does not involve crowdsourcing nor research with human subjects.

Guidelines:

- The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.