ATPO: ADAPTIVE TREE POLICY OPTIMIZATION FOR MULTI-TURN MEDICAL DIALOGUE

Anonymous authors
Paper under double-blind review

ABSTRACT

Effective information seeking in multi-turn medical dialogues is critical for accurate diagnosis, especially when dealing with incomplete information. Aligning Large Language Models (LLMs) for these interactive scenarios is challenging due to the uncertainty inherent in user-agent interactions, which we formulate as a Hierarchical Markov Decision Process (H-MDP). While conventional Reinforcement Learning (RL) methods like Group Relative Policy Optimization (GRPO) struggle with long-horizon credit assignment and Proximal Policy Optimization (PPO) suffers from unstable value estimation in this context, we propose a novel uncertainty-aware **A**daptive **T**ree **P**olicy **O**ptimization (**ATPO**) algorithm. Our method adaptively allocates the rollout budget to states with high uncertainty, quantified by a composite metric of Bellman error and action-value variance. This strategy enables more accurate value estimation, while fostering more efficient and diverse exploration. To mitigate the high computational cost of tree-based RL, we introduce two key optimizations: an uncertainty-guided pruning mechanism to minimize the number of rollouts, and an asynchronous search architecture that leverages KV cache reuse to maximize inference throughput. Extensive experiments on three public medical dialogue benchmarks demonstrate that our algorithm significantly outperforms several strong baselines, culminating in Qwen3-8B model surpassing the much larger GPT-4o ($+0.92\%$ accuracy). [1]

1 INTRODUCTION

In recent years, Large Language Models (LLMs) such as GPT-4 Achiam et al. (2023), Gemini 2.5 Comanici et al. (2025), Qwen3 Yang et al. (2025), and DeepSeek-R1 Guo et al. (2025a) have demonstrated exceptional capabilities across a range of natural language processing tasks, including open-domain question answering, dialogue generation, and code generation, continuously pushing the boundaries of AI performance (Chen et al., 2025). These models are increasingly being applied to downstream domains like education (Chu et al., 2025), law (Siino et al., 2025), and healthcare (Awasthi et al., 2025). Within the medical field, medical LLMs are consistently achieving state-of-the-art results on various benchmarks (Sellergren et al., 2025; Li et al., 2025a), such as professional medical examinations (Jiang et al., 2025) and disease diagnosis tasks (Xu et al., 2025; McDuff et al., 2025), and show immense potential for providing preliminary medical advice and assisting in clinical decision-making (Kopka et al., 2025).

However, despite these achievements, a critical aspect has long been overlooked. Current training and evaluation of medical LLMs predominantly focus on single-turn interaction scenarios, where models are expected to provide faithful responses based on the user's initial input. In real-world medical dialogues, however, the information provided by users is often incomplete, making it difficult to generate a satisfactory response based solely on the vague or fragmented initial query (Auerbach et al., 2024; Wu et al., 2025a). This necessitates the model's ability to proactively ask clarifying questions to gather more essential information. Unfortunately, this capability for dynamic, multi-turn information gathering is a significant deficiency in current models (Laban et al., 2025).

Previous work has attempted to fill this gap. Some approaches Li et al. (2024); Liu et al. (2025a); Hu et al. (2024) have used prompt engineering to elicit proactive questioning, but these methods often

[1] Code: https://anonymous.4open.science/r/ATPO-03D0/.

fail to fundamentally enhance the model's multi-turn interactive capabilities and can even lead to poorer performance than simply responding with incomplete information. Other efforts Liao et al. (2023); Liu et al. (2025b) have employed supervised fine-tuning (SFT) to improve dynamic interaction, yet these models tend to merely imitate the training data. Furthermore, some studies Shi et al. (2024); Xiong et al. (2024) have extended single-turn preference optimization to the trajectory level, but they rely on costly preference data and are highly sensitive to distribution shift. While reinforcement learning offers a promising, goal-driven alternative with stronger generalization (Guo et al., 2025a), current methods are also flawed. For instance, Group Relative Policy Optimization Shao et al. (2024) struggles with long-horizon credit assignment, and Proximal Policy Optimization Schulman et al. (2017) often suffers from unstable value estimation, hindering effective policy learning for complex multi-turn dialogues (Feng et al., 2025a).

In this work, we introduce Adaptive Tree Policy Optimization, a novel uncertainty-aware algorithm illustrated in Figure 1. ATPO employs an adaptive tree search where, for each node, it calculates an uncertainty metric to decide whether to expand the search further. This metric is a composite of two key signals: the Bellman error, which prioritizes nodes beneficial for critic training, and the action-value variance, which encourages sampling diversity. Furthermore, ATPO achieves high training efficiency by reusing shared prefixes to fully leverage the Key-Value (KV) cache Kwon et al. (2023) mechanism, combined with an asynchronous execution strategy. We conduct comprehensive evaluations on three Qwen3 models Yang et al. (2025) with different sizes (Qwen3 1.7B, 4B and 8B) using three multi-turn medical dialogue datasets meticulously adapted from public multiple-choice question datasets (MedQA Jin et al. (2020), MedMCQA Pal et al. (2022), and MedicalExam Liao et al. (2024)). The experimental results demonstrate that our proposed method significantly outperforms strong competitive RL baselines across all datasets and model sizes, validating its effectiveness and generalization capabilities.

Our contributions are as follows:

- We propose the Adaptive Tree Policy Optimization algorithm, which adaptively allocates rollout budgets based on turn-level uncertainty in multi-turn medical dialogues. This method enhances sampling diversity while simultaneously improving the critic model's accuracy.
- We design ATPO to be highly efficient by reusing shared prefixes to fully leverage the KV cache, and we implement it with an asynchronous execution strategy to achieve substantial improvements in inference throughput.
- Extensive experiments demonstrate that ATPO not only consistently and significantly outperforms strong RL baselines, but also achieves this with far greater sample efficiency, validating its robust generalization and effectiveness.

2 Related Work

2.1 Reinforcement Learning in Multi-turn Medical Dialogues

Recent efforts have applied reinforcement learning to train medical LLMs for proactive, multi-turn dialogue. A common approach involves simulating doctor-patient interactions within a multi-agent framework. For instance, DoctorAgent-RL Feng et al. (2025b) uses such a setup where a "doctor agent" and "patient agent" (termed the "assistant" and "user", respectively, in our work) interact, with the former learning an optimal questioning strategy guided by a comprehensive evaluator. To better guide this learning process, ProMed Ding et al. (2025) introduces a process reward based on Shapley Information Gain, which uses cooperative game theory to formally quantify the clinical utility of each question, enabling more targeted policy optimization. In contrast, Savage Conversation Forests (SCF) Savage (2025) concentrates on the training architecture, which employs a branched conversation structure to help the model learn how early conversational choices impact downstream outcomes by exploring multiple dialogue paths simultaneously. Despite these excellent works, a robust reinforcement learning algorithm capable of training a model to handle the complexities of real-world medical dialogues remains to be fully explored.

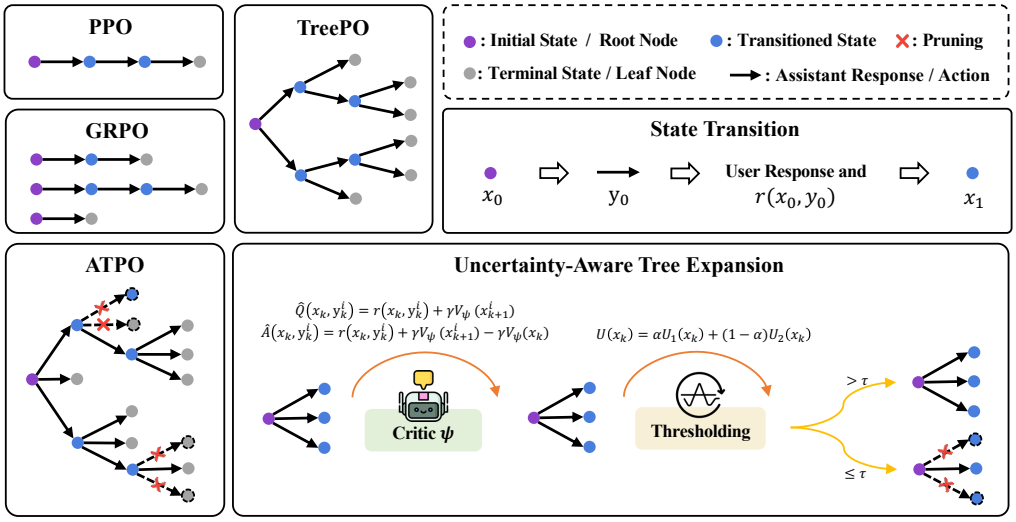

Figure 1: Overview of ATPO algorithm. ATPO generates training data via an adaptive tree search. At each expansion step, it generates candidate child nodes and computes a composite uncertainty score based on their Bellman error U_1 and Q-value variance U_2. Nodes with high uncertainty U are fully expanded, while low-uncertainty nodes are pruned by randomly selecting a single child for the subsequent rollout. The collected trajectories are then used for policy and critic updates.

2.2 Tree-based Reinforcement Learning

To enhance the reasoning capabilities of LLMs, recent reinforcement learning approaches have begun to integrate tree-based search, particularly for single-turn tasks. These methods primarily focus on three distinct goals. First, some aim to improve value estimation; for example, VinePPO Kazemnejad et al. (2024) uses auxiliary "vine" rollouts to compute more accurate Monte Carlo values, mitigating the impact of inaccurate value estimation. Second, another line of work leverages model uncertainty to guide exploration, typically by expanding the search tree at tokens with high entropy (Hou et al., 2025; Dong et al., 2025). Third, others refine the search structure itself for better credit assignment or efficiency, such as SPO Guo et al. (2025b), which defines "segments" based on low-probability tokens, or TreePO Li et al. (2025b), which uses a fixed N-ary tree for computational gains. Despite their innovations, these methods are fundamentally limited by their single-turn, token-level focus. Their operational units (tokens, segments) do not naturally translate to the macro-level decisions required in multi-turn dialogue. Moreover, their uncertainty metrics are either indirect proxies (e.g., token entropy) or absent entirely in methods with fixed structures. Our approach directly addresses these limitations. We introduce an uncertainty measure based on the variance of turn-level Q-values, which quantifies the uncertainty over future rewards for different macro-actions (i.e., conversational turns). This makes our method inherently better suited for the long-horizon planning challenges of multi-turn interactions.

3 Methods

3.1 Multi-Turn Dialogues as a Hierarchical MDP

Similar to ArCHer Zhou et al. (2024), we model multi-turn dialogues as a Hierarchical Markov Decision Process (H-MDP), which comprises a high-level MDP with a nested low-level MDP. For the high-level MDP, a macro-action is defined as the full token sequence of the assistant's response in a single turn. A micro-action in the low-level MDP is a single token, such that generating a sequence of micro-actions in the low-level MDP leads to a single macro-action in the high-level MDP. Formally, each state x_k in the high-level MDP consists of the interaction history between the user and the assistant prior to the k-th turn, together with the user's query q_k at the k-th turn. The macro-action y_k is defined as a variable-length token sequence representing the assistant's response to q_k. The low-level MDP models the process of generating a macro-action, where each low-level

action $y_{k,t}$ corresponds to an individual token (i.e., the t-th token in the k-th macro-action). A low-level state x_k, t is defined as the concatenation of the high-level state x_k and the partial sequence $y_{k,<t}$, namely the tokens that have already been generated in the current turn up to (but excluding) the t-th token.

3.2 Dialogue Exploration via Tree Expansion

In H-MDP optimization, accurate state-value estimation is crucial for effective policy improvement. Existing methods either rely on pure Monte Carlo estimates from full trajectories (e.g., GRPO Shao et al. (2024)), which can be high-variance and lead to unstable training, or solely on a learned critic (e.g., PPO Schulman et al. (2017)), which may suffer from approximation errors. To strike a balance between accuracy and efficiency, we propose an exploration strategy based on incremental tree expansion. This approach efficiently reuses the computation of shared dialogue prefixes and adaptively allocates the sampling budget to the most promising or uncertain parts of the dialogue space, rather than repeatedly re-exploring from the root.

In this framework, the multi-turn dialogue process is viewed as the expansion of a search tree. The initial user query forms the root node, which is at depth 0 and corresponds to state x_0. From there, the tree expands turn-by-turn. At any non-terminal node corresponding to state x_k, the assistant produces a macro-action, which can be either a clarifying question or a definitive answer. In the case of a clarifying question, the user's reply completes the state transition to x_{k+1}, resulting in the creation of a new node at the next depth of the tree. In the case of a definitive answer, the dialogue terminates along that branch and the node becomes a terminal leaf; for consistency, the subsequent user feedback in this scenario is considered empty. Thus, the depth of a node corresponds to the number of turns elapsed in the dialogue, with the root at depth 0, and the nodes at depth k representing all possible dialogue states after k turns of interaction.

This entire procedure is guided by an uncertainty-aware principle. The core idea is to prioritize the expansion of nodes exhibiting high uncertainty, as they provide more diverse and informative samples for driving effective policy updates and improving critic accuracy. In the multi-turn dialogue setting, this uncertainty can be broadly categorized into two types: **Epistemic uncertainty**, arising from the inherent cognitive limitations of the model and manifests as uncertainty in its action. **Aleatoric uncertainty**, stemming from the inherent randomness in the environment, such as the variability in user responses.

3.3 Uncertainty-Aware Tree Expansion Mechanism

To operationalize our tree expansion strategy, we first quantify the uncertainty of each frontier node. Consider a specific node at depth k, representing the dialogue state x_k. We sample a set of N candidate macro-actions $\{y_k^i\}_{i=1}^N$ from the policy $\pi_\theta(\cdot|x_k)$. For each candidate action y_k^i, we define its estimated action-value $\hat{Q}(x_k, y_k^i)$ using a one-step lookahead:

$$\hat{Q}(x_k, y_k^i) = r(x_k, y_k^i) + \gamma V_\psi(x_{k+1}^i), \tag{1}$$

where x_{k+1}^i is the state resulting from action y_k^i, and the next-state value V_ψ is given by the critic model. Based on this, the uncertainty is calculated as:

$$U_1(x_k) = \left| V_\psi(x_k) - \frac{1}{N} \sum_{i=1}^{N} \hat{Q}(x_k, y_k^i) \right|, \tag{2}$$

$$U_2(x_k) = \mathrm{Var}_{i\in[N]}\left[\hat{Q}(x_k, y_k^i)\right] = \frac{1}{N} \sum_{i=1}^{N} \left(\hat{Q}(x_k, y_k^i) - \frac{1}{N} \sum_{j=1}^{N} \hat{Q}(x_k, y_k^j) \right)^2, \tag{3}$$

$$U(x_k) = \alpha\, U_1(x_k) + (1-\alpha)\, U_2(x_k). \tag{4}$$

The first term, $U_1(x_k)$, measures the Bellman error between the critic's current state-value estimate $V_\psi(x_k)$ and the empirical one-step lookahead value averaged over all candidates. It serves as a proxy for **aleatoric uncertainty**; a large error indicates an inaccurate value estimate for the current state. The second term, $U_2(x_k)$, quantifies the variance of the action-value estimates. This term captures a blend of **both epistemic and aleatoric uncertainty**. High variance can arise from two

distinct sources: the policy's own indecision leading it to explore a diverse set of candidate actions (epistemic), or the environment's inherent randomness where different state transitions yield widely varying Q-values (aleatoric). The notation $\mathrm{Var}(\cdot)$ represents the variance across the N candidate actions. This raw value is subsequently normalized using Z-score scaling based on historical samples to create a stable, comparable metric. The hyperparameter $\alpha \in [0, 1]$ balances the contributions of these two uncertainty sources.

With the uncertainty metric $U(x_k)$ defined, the expansion process is guided by a threshold-based rule. When the expansion process reaches a node at depth k, representing one of the possible dialogue states after k turns, we compare its calculated uncertainty against a predefined threshold τ: **If** $U(x_k) > \tau$, the node is considered highly uncertain and all N branches are retained. **Conversely, if** $U(x_k) \leq \tau$, the node is deemed sufficiently understood. To conserve computational resources, we primarily prune the search by randomly selecting only one of the N candidate branches. However, to maintain a baseline level of sampling diversity, we bypass this pruning with a small probability (e.g., 10%) and instead expand all N branches.

The uncertainty-driven expansion continues along the retained branches until all dialogues terminate or the total number of leaf nodes (i.e., the tree width) reaches a predefined budget. Once this budget is met, no further nodes are expanded, and all current leaf nodes proceed to the rollout phase until termination.

3.4 Value Traceback, Tree Decomposition and Model Optimization

Following the completion of the tree expansion process, we compute the state-values and advantages for all nodes within the generated tree via a recursive backward pass starting from the leaf nodes. First, we calculate the target value $\hat{V}(x_k)$ for each state x_k, which serves as a robust empirical estimate of the Monte Carlo return from that state and can be used as the value target for subsequent critic updates:

$$\hat{V}(x_k) = \begin{cases} r(x_k, y_k), & \text{if leaf node,} \\ \dfrac{1}{B(x_k)} \displaystyle\sum_{i=1}^{B(x_k)} \left[r(x_k, y_k^i) + \gamma \hat{V}(x_{k+1}^i) \right], & \text{otherwise.} \end{cases} \tag{5}$$

If a state is terminal, its target value equals the immediate reward; otherwise, it is the average one-step TD target over all child branches. Here, $B(x_k)$ denotes the number of child nodes, which according to the expansion rule is either N for fully expanded nodes or 1 for pruned nodes, and the superscript i indexes the i-th branch among these children.

With the target values established, we compute the estimated advantage for each macro-action y_k^i using the standard one-step temporal-difference formulation:

$$\hat{A}(x_k, y_k^i) = r(x_k, y_k^i) + \gamma\, V_\psi(x_{k+1}^i) - V_\psi(x_k). \tag{6}$$

We use the critic's value estimates instead of the target values $\hat{V}(\cdot)$ because with only one retained branch, $\hat{V}(x_k)$ equals its one-step return, resulting in a zero advantage, whereas the critic's estimates preserve non-zero learning signals.

Once values and advantages have been computed for all nodes, the expanded tree is decomposed into a set of independent trajectories for model optimization. Each unique path from the root to a leaf node constitutes one trajectory, so a tree with M leaf nodes directly yields M such trajectories. The number of turns in the j-th trajectory, denoted by $K^{(j)}$, can vary across trajectories because different dialogues may terminate after different numbers of turns. The policy π_θ is then updated by maximizing the following PPO-style objective, aggregated over all tokens from all trajectories:

$$J(\theta) = \mathbb{E}_{x_0 \sim \mathcal{D}, \{y_k \sim \pi_\theta(x_k), x_{k+1} \sim P(\cdot|x_k, y_k)\}_{k=0}^{K-1}} \left[\frac{1}{M} \sum_{j=1}^{M} \frac{1}{K^{(j)}} \sum_{k=1}^{K^{(j)}} \frac{1}{C(x_k^{(j)})\, L(y_k^{(j)})} \right. \tag{7}$$
$$\left. \sum_{t=1}^{L(y_k^{(j)})} \min\Big(\rho_{k,t}^{(j)} \hat{A}(x_k, y_{k,t}^{(j)}), \mathrm{clip}\big(\rho_{k,t}^{(j)},\, 1-\epsilon,\, 1+\epsilon\big) \Big) - \beta D_{\mathrm{KL}}(\pi_\theta, \pi_{\mathrm{ref}}) \right],$$

$$\text{where } \rho_{k,t}^{(j)} = \frac{\pi_\theta\left(y_{k,t}^{(j)} \,\middle|\, x_k^{(j)}, y_{k,<t}^{(j)}\right)}{\pi_{\text{ref}}\left(y_{k,t}^{(j)} \,\middle|\, x_k^{(j)}, y_{k,<t}^{(j)}\right)}, \text{ and } \hat{A}(x_k, y_{k,t}^{(j)}) = \hat{A}(x_k, y_k^{(j)}) \text{ for } 1 \le t \le L(y_k^{(j)}).$$

In this objective, the expectation $\mathbb{E}(\cdot)$ is taken over initial user queries x_0 sampled from the dataset $\mathcal{D}$, along with subsequent states and actions. The term $\rho_{k,t}^{(j)}$ denotes the probability ratio between the current policy π_θ and the reference policy π_{ref} for generating the t-th token in the k-th turn of the j-th trajectory. Consistent with our hierarchical MDP formulation, the advantage $\hat{A}(x_k, y_{k,t}^{(j)})$ for each token is set equal to the macro-action's advantage $\hat{A}(x_k, y_k^{(j)})$, thereby uniformly distributing the turn-level credit across the tokens generated in that turn. The update is normalized by two factors: $C(x_k^{(j)})$, the visit count of the state across all trajectories, which prevents over-optimizing on frequently visited nodes; and $L(y_k^{(j)})$, the length of the response, which ensures the turn-level advantage is properly scaled for each token. The $\min(\cdot)$ and $\text{clip}(\cdot)$ constrain the policy update to stabilize training.

Finally, the critic model, parameterized by ψ, is trained to predict the target state values. The critic consists of the LLM backbone and a linear value head. Its value estimate $V_\psi(x_k)$ is obtained as the average of the predictions over the final h special tokens. The critic is trained by minimizing the following mean squared error loss:

$$\mathcal{L}(\psi) = \frac{1}{M}\sum_{j=1}^{M} \frac{1}{K^{(j)} \cdot h} \sum_{k=1}^{K^{(j)}} \sum_{t=L(x_k^{(j)})-h}^{L(x_k^{(j)})} \frac{1}{2}\left[V_\psi(x_{k,t}^{(j)}) - \hat{V}(x_k^{(j)})\right]^2, \tag{8}$$

where $V_\psi(x_{k,t}^{(j)})$ is the value predicted at the t-th token position of the input, with $L(x_k^{(j)})$ denoting the total number of tokens of state $x_k^{(j)}$. This loss trains the critic to make the predictions at the final h token positions for state $x_k^{(j)}$ match the ground-truth target value $\hat{V}(x_k)$.

4 Experiments

4.1 Experiment Setup

Environment and Task Description. We establish a multi-turn clinical case reasoning environment with two LLM-based agents: a User Simulator and an Assistant Agent. (1) **User Simulator**: Implemented using Qwen3-8B, it is instructed to answer the Assistant's questions strictly based on a given set of atomic facts, refusing to respond to any out-of-scope queries. To ensure reliability, we continuously monitor its behavior during training and verify it with GPT-4o, achieving 100% accuracy in following instructions and rejecting irrelevant queries, with a hallucination rate of only 1.2%, demonstrating high fidelity. Detailed prompts are provided in Appendix A.3. (2) **Assistant Agent**: The Assistant Agent is tasked with resolving a clinical case question by selecting the correct answer from a list of options. It begins with an initial context and can iteratively query the User Simulator for additional information. The process terminates when the agent either commits to a final answer or the pre-defined turn limit is reached. An interaction example between the Assistant Agent and the User Simulator is provided in Appendix A.1.

Baselines. We compare our ATPO with several baselines to evaluate its effectiveness:

1) **Zero-shot Prompting**: To benchmark performance, we evaluate several base models (Qwen3-1.7B, Qwen3-4B, and Qwen3-8B) under two distinct settings. The first is a Direct single-turn setting, where the agent must respond using only the initial context. The second is an interactive MEDIQ Li et al. (2024) setting, which allows the agent to interact with the User Simulator for up to 8 turns in total, including the final answering turn. Prompts for both settings are provided in Appendix A.4.

2) **SFT**: To establish a stronger baseline, we fine-tune the Qwen3 instruct models to encourage multi-turn information-seeking behavior instead of directly producing an answer in the first turn. Using the MEDIQ validation dataset as the source of atomic facts, questions, and answer options, we employ the expert model Gemini-2.5-Pro in a self-play setup, where it role-plays both user and assistant

to generate $1,269$ multi-turn dialogues. We ensure no information leakage during this process and retain only dialogues with a correct final answer. The resulting dataset is used to train the models with supervised fine-tuning and dynamic fine-tuning (DFT Wu et al. (2025b)).

3) **SFT+RL**: We compare our algorithm with standard RL post-training methods under the same environment, using rewards solely based on the correctness of the final answer. *Critic-based Methods*: PPO (MDP) treats text generation as a standard MDP, assigning a unique value to each token. PPO (H-MDP) adopts a hierarchical formulation, estimating a single value per turn and propagating the corresponding advantage to all tokens in that turn. *Critic-free Methods*: We implement GRPO, which assigns a single advantage to an entire trajectory and shares it across all tokens. We also adopt TreePO (Guo et al., 2025b; Li et al., 2025b), where we construct a binary search tree in which each non-terminal node expands into two child nodes, constrained only by a maximum depth corresponding to the dialogue turn limit, without any pruning. After the search completes, we perform a backward pass to compute the aggregate return for each node as its target value $\hat{V}(x_k)$. The advantage is calculated as $\hat{A}(x_k, y_k) = r(x_k, y_k) + \gamma\hat{V}(x_{k+1}) - \hat{V}(x_k)$, and is assigned to all tokens generated in that sequence for policy optimization.

Implementation Details. For RL experiments, we use the SFT-trained Qwen3-1.7B, Qwen3-4B, and Qwen3-8B models as the initial policy for the Assistant Agent. The reward function is based solely on final-answer correctness: $+3$ for a correct answer, 0 for an incorrect answer, and -1 for an invalid format. The training data contains $14,256$ samples, with 66% $(9,400)$ drawn from the MEDIQ training dataset and 34% $(4,856)$ constructed from the MedMCQA training dataset.

Policy learning rate 1×10^{-6}, critic learning rate 1×10^{-5}, KL penalty weight $\beta = 0.01$, and discount factor $\gamma = 1$. The critic is initialized from the actor's weights and warmed up for 5 steps. Method-specific settings include a group size of 32 for GRPO, and for ATPO, an expansion size $N = 4$ with a total expansion budget of 128. In ATPO (U_1), the uncertainty threshold is $\tau = 0.5$; in ATPO ($U_1 + U_2$), we set $\alpha = 0.3$ and $\tau = 1.5$.

Our TreePO and ATPO implementations build upon the VeRL (Sheng et al., 2025) Agentic RL framework, integrating tree search, reward computation, and advantage estimation into a single concurrent phase. This design eliminates the need for a multi-stage pipeline by producing ready-to-train trajectories directly from the search process. High throughput is achieved via asynchronous rollouts across tree nodes and efficient prefix sharing, with vLLM KV cache enabling speeds of up to $2{,}500$ tokens/sec/GPU on a 1.7B model with TreePO. Both our implementation and the associated datasets are available at `https://anonymous.4open.science/r/ATPO-03D0/`.

4.2 RESULTS

Evaluation Setup. We conduct evaluations on three Qwen3 models Yang et al. (2025) of different sizes (Qwen3-1.7B, Qwen3-4B, and Qwen3-8B), along with GPT-4o as a strong baseline to assess the effectiveness of our method. Experiments are performed on three multi-turn medical dialogue datasets adapted from public multiple-choice question datasets: MedQA, obtained directly from the MEDIQ Li et al. (2024) test set; MedMCQA, constructed from its original training data Pal et al. (2022); and MedicalExam, sourced directly from AIE Liao et al. (2024). Each sample is reformulated into a set of atomic facts, a concise initial context, an atomic question that excludes factual details, and several answer options with exactly one correct choice (details in Appendix A.2). The primary evaluation metric is *final-answer accuracy*, defined as the percentage of test cases where the Assistant Agent's chosen option matches the ground-truth answer. For statistical robustness, we report the mean and standard deviation of five independent runs.

Main Findings. From Table 1, we observe that in the zero-shot setting, the MEDIQ prompting strategy performs worse than the Direct single-turn prompt, consistent with the finding in MEDIQ Li et al. (2024) that prompting LLMs to ask questions can reduce accuracy. Supervised fine-tuning brings only modest gains in final-answer accuracy while being crucial for enabling multi-turn information seeking and providing a solid foundation for subsequent reinforcement learning.

Our proposed ATPO achieves the highest accuracy in most settings, even surpassing GPT-4o at the 8B scale (e.g., exceeding GPT-4o on MedQA by 0.92%). This demonstrates the strong effectiveness of the method. Further, the results show that both uncertainty metrics are valuable and complementary: ATPO ($U_1 + U_2$) generally outperforms ATPO (U_1), which in turn achieves higher accuracy

Table 1: Performance comparison (%) on **MedicalExam**, **MedQA**, and **MedMCQA**. **Bold** indicates the best performance, underlined the second-best.

Model	Method Type	Method Name	MedicalExam	MedQA	MedMCQA
Qwen3-1.7B	Prompt	Direct	35.07 ± 1.12	34.05 ± 0.38	32.54 ± 0.49
		MEDIQ	34.00 ± 2.26	34.20 ± 0.75	32.35 ± 1.73
	SFT	DFT	29.07 ± 1.46	28.38 ± 0.80	21.08 ± 0.90
		SFT	32.27 ± 4.77	33.42 ± 0.95	28.10 ± 2.32
	SFT+RL	PPO (MDP)	39.33 ± 4.01	38.64 ± 1.17	35.37 ± 0.80
		PPO (H-MDP)	39.33 ± 2.79	39.08 ± 1.85	34.89 ± 1.00
		GRPO	42.93 ± 1.80	41.17 ± 0.64	36.57 ± 3.26
		TreePO	43.33 ± 1.56	42.05 ± 1.03	38.47 ± 2.00
		ATPO (U_1)	**45.73 ± 1.53**	42.54 ± 0.39	38.66 ± 0.66
		ATPO ($U_1 + U_2$)	43.20 ± 1.85	**42.87 ± 0.77**	**39.93 ± 1.05**
Qwen3-4B	Prompt	Direct	48.13 ± 0.87	44.94 ± 0.35	41.53 ± 0.39
		MEDIQ	45.87 ± 1.20	40.11 ± 0.60	31.64 ± 1.41
	SFT	DFT	43.07 ± 1.61	41.72 ± 1.27	33.28 ± 1.68
		SFT	48.93 ± 2.14	47.15 ± 1.01	39.18 ± 1.22
	SFT+RL	PPO (MDP)	50.13 ± 2.80	50.60 ± 0.90	42.50 ± 0.84
		PPO (H-MDP)	52.40 ± 2.24	48.58 ± 1.48	43.32 ± 2.22
		GRPO	53.87 ± 2.08	51.17 ± 1.08	43.84 ± 0.78
		TreePO	56.13 ± 0.99	53.74 ± 0.56	45.22 ± 0.65
		ATPO (U_1)	56.80 ± 1.28	53.15 ± 0.55	**46.23 ± 1.25**
		ATPO ($U_1 + U_2$)	**59.73 ± 2.61**	**55.47 ± 0.99**	45.93 ± 1.13
Qwen3-8B	Prompt	Direct	52.40 ± 0.37	45.22 ± 0.34	46.16 ± 1.04
		MEDIQ	51.87 ± 3.69	46.03 ± 0.75	41.60 ± 0.91
	SFT	DFT	51.86 ± 3.63	48.80 ± 1.30	42.20 ± 0.83
		SFT	55.87 ± 0.30	53.75 ± 1.18	46.87 ± 1.74
	SFT+RL	PPO (MDP)	59.20 ± 3.84	57.38 ± 0.84	50.00 ± 0.81
		PPO (H-MDP)	59.07 ± 3.15	57.81 ± 1.29	51.98 ± 0.67
		GRPO	60.93 ± 1.86	57.92 ± 0.68	51.12 ± 1.29
		TreePO	65.33 ± 3.09	61.81 ± 0.90	**54.74 ± 1.99**
		ATPO (U_1)	65.52 ± 3.12	62.57 ± 0.41	53.22 ± 1.30
		ATPO ($U_1 + U_2$)	**65.87 ± 3.72**	**64.07 ± 0.43**	53.66 ± 1.52
GPT-4o	Prompt	MEDIQ	64.00 ± 3.53	63.15 ± 0.82	53.03 ± 0.89

than TreePO. Combining U_1 and U_2 yields the best results, with absolute gains over TreePO on MedQA of 0.82%, 1.73%, and 2.26% for the 1.7B, 4B, and 8B models, respectively.

ATPO also exhibits markedly higher sample efficiency, as shown in Figure 2 (a). For instance, on MedQA with Qwen3-4B, ATPO ($U_1 + U_2$) achieves approximately 52.7% accuracy while using only 55% of the training turns required by TreePO. Additionally, hierarchical modeling proves beneficial in multi-turn dialogue: PPO (H-MDP) slightly but consistently surpasses PPO (MDP), scoring higher in 5 out of 9 evaluation settings. Among critic-free methods, the tree-based approach demonstrates clear superiority, with TreePO substantially outperforming GRPO. This indicates that, for complex multi-turn tasks, structuring credit assignment via a search tree is more effective than relying on a single trajectory-level advantage.

4.3 Ablation and Analysis

We conducted several ablation experiments on ATPO, which provide the following key insights.

The dual uncertainty metrics enhance both sampling diversity and critic optimization. Figure 2 (b) shows that guiding node expansion with $U_1 + U_2$ produces a high variance of sample returns, comparable to GRPO and markedly higher than TreePO, while using only U_1 reduces diversity. At the same time, Figure 2 (c) indicates that the critic's value loss under $U_1 + U_2$ is substantially lower than PPO (both MDP and H-MDP), with U_1 alone ranking second. These results highlight that uncertainty-aware tree search benefits value function learning. This advantage partly stems from intelligent budget allocation (Figures 2 (d), (e)): U_1 alone drives aggressive early exploration, concentrating expansions at shallow depths (3–4) and causing steep local drops in node values, whereas $U_1 + U_2$ achieves deeper coverage and maintains a more uniform value variance, enabling a more globally effective search.

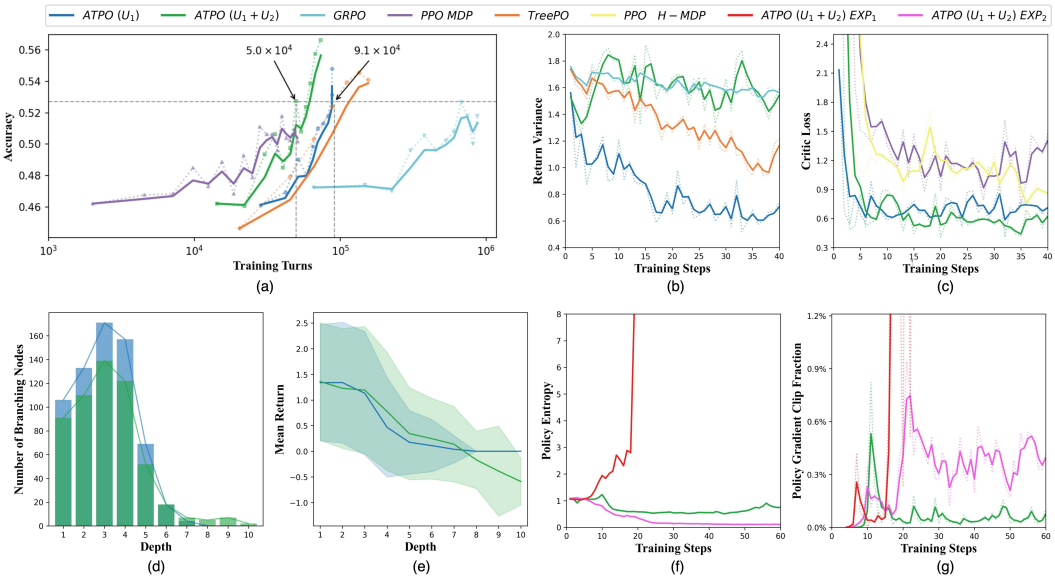

Figure 2: Analysis of the ATPO algorithm on Qwen3-4B. (a) Training efficiency and performance comparison of various algorithms, plotting accuracy against the number of generated turns. (b), (c) Return variance and critic loss for ATPO and baseline methods. (d), (e) Distribution of branching nodes and returns by depth for samples from ATPO at a representative training step. (f), (g) Stability analysis of ATPO with and without visit-count-based down-weighting.

We also find that down-weighting policy updates by node visit count is crucial for training stability. An ablation study compares the default ATPO (policy advantage down-weighted) with two variants: EXP_1 (no policy down-weighting) and EXP_2 (down-weighting both policy advantage and value loss). EXP_1 results in uncontrolled entropy growth and excessive policy clipping (Figures 2 (f), (g)), since ignoring visit counts leads to disproportionate policy updates on frequently visited nodes, causing the policy to diverge rapidly from its reference state. In contrast, EXP_2 induces rapid entropy collapse. High-visit nodes provide the most reliable value estimates due to abundant samples, and underweighting the value loss on these nodes destabilizes the critic, increasing value variance in early layers over time. As a result, the policy learns to distrust its flawed value function, abandons multi-turn exploration, and regresses to suboptimal single-turn strategies.

5 Conclusions and Future Work

In this paper, we present ATPO, a novel adaptive tree search method that intelligently guides exploration in multi-turn dialogues through state-uncertainty evaluation. By selectively expanding nodes that maximize sampling diversity and benefit critic optimization, ATPO achieves superior performance with markedly fewer exploration steps, surpassing strong RL baselines such as TreePO and GRPO across several clinical dialogue benchmarks, and even outperforming GPT-4o on MedQA with the Qwen3-8B model.

Future work could pursue several promising directions. First, replacing the current fixed-threshold expansion mechanism with a learnable, soft control policy may reduce hyperparameter tuning effort and enable the expansion strategy to adapt dynamically as the policy evolves. This idea could be further extended to adaptively determine the number of nodes to expand based on uncertainty metrics, rather than relying on random selection. Second, refining credit assignment within the Hierarchical MDP (H-MDP) framework could yield additional gains. A more sophisticated approach to distributing high-level advantages to low-level token actions, moving beyond uniform cloning, may allow for more precise policy optimization and improved performance.

REFERENCES

Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*, 2023.

Andrew D Auerbach, Tiffany M Lee, Colin C Hubbard, Sumant R Ranji, Katie Raffel, Gilmer Valdes, John Boscardin, Anuj K Dalal, Alyssa Harris, Ellen Flynn, et al. Diagnostic errors in hospitalized adults who died or were transferred to intensive care. *JAMA Internal Medicine*, 184 (2):164–173, 2024.

Raghav Awasthi, Sai Prasad Ramachandran, Shreya Mishra, Dwarikanath Mahapatra, Hajra Arshad, Aarit Atreja, Anirban Bhattacharyya, Atharva Bhattad, Nishant Singh, Jacek B Cywinski, et al. Artificial intelligence in healthcare: 2024 year in review. *medRxiv*, pp. 2025–02, 2025.

Qingyu Chen, Yan Hu, Xueqing Peng, Qianqian Xie, Qiao Jin, Aidan Gilson, Maxwell B Singer, Xuguang Ai, Po-Ting Lai, Zhizheng Wang, et al. Benchmarking large language models for biomedical natural language processing applications and recommendations. *Nature communications*, 16(1):3280, 2025.

Zhendong Chu, Shen Wang, Jian Xie, Tinghui Zhu, Yibo Yan, Jinheng Ye, Aoxiao Zhong, Xuming Hu, Jing Liang, Philip S Yu, et al. Llm agents for education: Advances and applications. *arXiv preprint arXiv:2503.11733*, 2025.

Gheorghe Comanici, Eric Bieber, Mike Schaekermann, Ice Pasupat, Noveen Sachdeva, Inderjit Dhillon, Marcel Blistein, Ori Ram, Dan Zhang, Evan Rosen, et al. Gemini 2.5: Pushing the frontier with advanced reasoning, multimodality, long context, and next generation agentic capabilities. *arXiv preprint arXiv:2507.06261*, 2025.

Hongxin Ding, Baixiang Huang, Yue Fang, Weibin Liao, Xinke Jiang, Zheng Li, Junfeng Zhao, and Yasha Wang. Promed: Shapley information gain guided reinforcement learning for proactive medical llms. *arXiv preprint arXiv:2508.13514*, 2025.

Guanting Dong, Hangyu Mao, Kai Ma, Licheng Bao, Yifei Chen, Zhongyuan Wang, Zhongxia Chen, Jiazhen Du, Huiyang Wang, Fuzheng Zhang, et al. Agentic reinforced policy optimization. *arXiv preprint arXiv:2507.19849*, 2025.

Xiao Feng, Bo Han, Zhanke Zhou, Jiaqi Fan, Jiangchao Yao, Ka Ho Li, Dahai Yu, and Michael Ng. DyPO: Dynamic policy optimization for multi-turn interactive reasoning. In *ICML 2025 Workshop on Programmatic Representations for Agent Learning*, 2025a. URL https://openreview.net/forum?id=OWDBiMKYdo.

Yichun Feng, Jiawei Wang, Lu Zhou, Zhen Lei, and Yixue Li. Doctoragent-rl: A multi-agent collaborative reinforcement learning system for multi-turn clinical dialogue. *arXiv preprint arXiv:2505.19630*, 2025b.

Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Ruoyu Zhang, Runxin Xu, Qihao Zhu, Shirong Ma, Peiyi Wang, Xiao Bi, et al. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *arXiv preprint arXiv:2501.12948*, 2025a.

Yiran Guo, Lijie Xu, Jie Liu, Dan Ye, and Shuang Qiu. Segment policy optimization: Effective segment-level credit assignment in rl for large language models. *arXiv preprint arXiv:2505.23564*, 2025b.

Zhenyu Hou, Ziniu Hu, Yujiang Li, Rui Lu, Jie Tang, and Yuxiao Dong. Treerl: Llm reinforcement learning with on-policy tree search. *arXiv preprint arXiv:2506.11902*, 2025.

Zhiyuan Hu, Chumin Liu, Xidong Feng, Yilun Zhao, See-Kiong Ng, Anh Tuan Luu, Junxian He, Pang Wei W Koh, and Bryan Hooi. Uncertainty of thoughts: Uncertainty-aware planning enhances information seeking in llms. *Advances in Neural Information Processing Systems*, 37: 24181–24215, 2024.

Xinke Jiang, Ruizhe Zhang, Yongxin Xu, Rihong Qiu, Yue Fang, Zhiyuan Wang, Jinyi Tang, Hongxin Ding, Xu Chu, Junfeng Zhao, et al. Hykge: A hypothesis knowledge graph enhanced rag framework for accurate and reliable medical llms responses. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 11836–11856, 2025.

Di Jin, Eileen Pan, Nassim Oufattole, Wei-Hung Weng, Hanyi Fang, and Peter Szolovits. What disease does this patient have? a large-scale open domain question answering dataset from medical exams. *arXiv preprint arXiv:2009.13081*, 2020.

Amirhossein Kazemnejad, Milad Aghajohari, Eva Portelance, Alessandro Sordoni, Siva Reddy, Aaron Courville, and Nicolas Le Roux. Vineppo: Refining credit assignment in rl training of llms. *arXiv preprint arXiv:2410.01679*, 2024.

Marvin Kopka, Niklas von Kalckreuth, and Markus A Feufel. Accuracy of online symptom assessment applications, large language models, and laypeople for self–triage decisions. *npj Digital Medicine*, 8(1):178, 2025.

Woosuk Kwon, Zhuohan Li, Siyuan Zhuang, Ying Sheng, Lianmin Zheng, Cody Hao Yu, Joseph E. Gonzalez, Hao Zhang, and Ion Stoica. Efficient memory management for large language model serving with pagedattention. In *Proceedings of the ACM SIGOPS 29th Symposium on Operating Systems Principles*, 2023.

Philippe Laban, Hiroaki Hayashi, Yingbo Zhou, and Jennifer Neville. Llms get lost in multi-turn conversation. *arXiv preprint arXiv:2505.06120*, 2025.

Ao Li, Bin Yan, Bingfeng Cai, Chenxi Li, Cunzhong Zhao, Fugen Yao, Gaoqiang Liu, Guanjun Jiang, Jian Xu, Liang Dong, et al. Quarkmed medical foundation model technical report. *arXiv preprint arXiv:2508.11894*, 2025a.

Stella Li, Vidhisha Balachandran, Shangbin Feng, Jonathan Ilgen, Emma Pierson, Pang Wei W Koh, and Yulia Tsvetkov. Mediq: Question-asking llms and a benchmark for reliable interactive clinical reasoning. *Advances in Neural Information Processing Systems*, 37:28858–28888, 2024.

Yizhi Li, Qingshui Gu, Zhoufutu Wen, Ziniu Li, Tianshun Xing, Shuyue Guo, Tianyu Zheng, Xin Zhou, Xingwei Qu, Wangchunshu Zhou, et al. Treepo: Bridging the gap of policy optimization and efficacy and inference efficiency with heuristic tree-based modeling. *arXiv preprint arXiv:2508.17445*, 2025b.

Yusheng Liao, Yutong Meng, Hongcheng Liu, Yanfeng Wang, and Yu Wang. An automatic evaluation framework for multi-turn medical consultations capabilities of large language models. *arXiv preprint arXiv:2309.02077*, 2023.

Yusheng Liao, Yutong Meng, Yuhao Wang, Hongcheng Liu, Yanfeng Wang, and Yu Wang. Automatic interactive evaluation for large language models with state aware patient simulator. *arXiv preprint arXiv:2403.08495*, 2024.

Ruoyu Liu, Kui Xue, Xiaofan Zhang, and Shaoting Zhang. Interactive evaluation for medical llms via task-oriented dialogue system. In *Proceedings of the 31st International Conference on Computational Linguistics*, pp. 4871–4896, 2025a.

Zijie Liu, Xinyu Zhao, Jie Peng, Zhuangdi Zhu, Qingyu Chen, Xia Hu, and Tianlong Chen. Dialogue is better than monologue: Instructing medical llms via strategical conversations. *arXiv preprint arXiv:2501.17860*, 2025b.

Daniel McDuff, Mike Schaekermann, Tao Tu, Anil Palepu, Amy Wang, Jake Garrison, Karan Singhal, Yash Sharma, Shekoofeh Azizi, Kavita Kulkarni, et al. Towards accurate differential diagnosis with large language models. *Nature*, pp. 1–7, 2025.

Ankit Pal, Logesh Kumar Umapathi, and Malaikannan Sankarasubbu. Medmcqa: A large-scale multi-subject multi-choice dataset for medical domain question answering. In *Conference on health, inference, and learning*, pp. 248–260. PMLR, 2022.

Thomas Savage. Conversation forests: The key to fine tuning large language models for multi-turn medical conversations is branching. *arXiv preprint arXiv:2507.04099*, 2025.

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

Andrew Sellergren, Sahar Kazemzadeh, Tiam Jaroensri, Atilla Kiraly, Madeleine Traverse, Timo Kohlberger, Shawn Xu, Fayaz Jamil, Cían Hughes, Charles Lau, et al. Medgemma technical report. *arXiv preprint arXiv:2507.05201*, 2025.

Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, YK Li, Yang Wu, et al. Deepseekmath: Pushing the limits of mathematical reasoning in open language models. *arXiv preprint arXiv:2402.03300*, 2024.

Guangming Sheng, Chi Zhang, Zilingfeng Ye, Xibin Wu, Wang Zhang, Ru Zhang, Yanghua Peng, Haibin Lin, and Chuan Wu. Hybridflow: A flexible and efficient rlhf framework. In *Proceedings of the Twentieth European Conference on Computer Systems*, EuroSys '25, pp. 1279–1297. ACM, March 2025. doi: 10.1145/3689031.3696075. URL http://dx.doi.org/10.1145/3689031.3696075.

Wentao Shi, Mengqi Yuan, Junkang Wu, Qifan Wang, and Fuli Feng. Direct multi-turn preference optimization for language agents. *arXiv preprint arXiv:2406.14868*, 2024.

Marco Siino, Mariana Falco, Daniele Croce, and Paolo Rosso. Exploring llms applications in law: A literature review on current legal nlp approaches. *IEEE Access*, 2025.

Chaoyi Wu, Pengcheng Qiu, Jinxin Liu, Hongfei Gu, Na Li, Ya Zhang, Yanfeng Wang, and Weidi Xie. Towards evaluating and building versatile large language models for medicine. *npj Digital Medicine*, 8(1):58, 2025a.

Yongliang Wu, Yizhou Zhou, Zhou Ziheng, Yingzhe Peng, Xinyu Ye, Xinting Hu, Wenbo Zhu, Lu Qi, Ming-Hsuan Yang, and Xu Yang. On the generalization of sft: A reinforcement learning perspective with reward rectification, 2025b. URL https://arxiv.org/abs/2508.05629.

Wei Xiong, Chengshuai Shi, Jiaming Shen, Aviv Rosenberg, Zhen Qin, Daniele Calandriello, Misha Khalman, Rishabh Joshi, Bilal Piot, Mohammad Saleh, et al. Building math agents with multi-turn iterative preference learning. *arXiv preprint arXiv:2409.02392*, 2024.

Yongxin Xu, Xinke Jiang, Xu Chu, Rihong Qiu, Yujie Feng, Hongxin Ding, Junfeng Zhao, Yasha Wang, and Bing Xie. Dearllm: Enhancing personalized healthcare via large language models-deduced feature correlations. *Proceedings of the AAAI Conference on Artificial Intelligence*, 39 (1):941–949, Apr. 2025. doi: 10.1609/aaai.v39i1.32079. URL https://ojs.aaai.org/index.php/AAAI/article/view/32079.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, et al. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*, 2025.

Yifei Zhou, Andrea Zanette, Jiayi Pan, Sergey Levine, and Aviral Kumar. Archer: Training language model agents via hierarchical multi-turn rl. In *International Conference on Machine Learning*, pp. 62178–62209. PMLR, 2024.

A Appendix

A.1 Interaction Process between Assistant Agent and User Simulator

Figure 3 presents an example of the interaction flow in our multi-turn clinical reasoning environment. The process starts from an incomplete initial user query representing a clinical case. The Assistant Agent receives this query along with a set of answer options. The initial information is intentionally limited; attempting to answer directly at this stage often leads to incorrect results. Therefore, the Assistant Agent must actively ask targeted questions to the User Simulator in order to obtain the missing key facts. The User Simulator responds strictly within the scope of predefined atomic facts, refusing to provide any information beyond this scope. This iterative exchange continues until the Assistant Agent commits to a final answer or the predefined turn limit is reached.

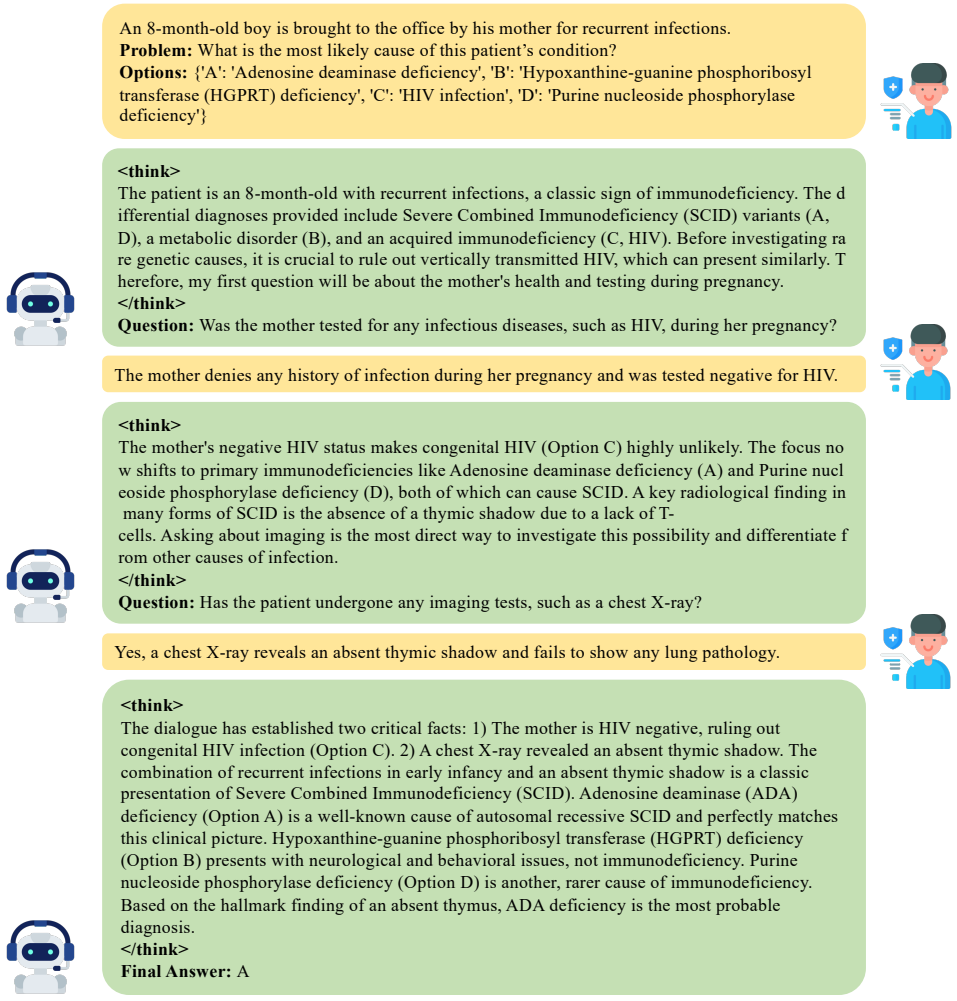

Figure 3: Schematic diagram of the interaction flow between the Assistant Agent and the User Simulator in the multi-turn clinical reasoning environment. The process starts from an incomplete initial user query, after which the Assistant Agent asks targeted questions and the User Simulator responds strictly within the scope of predefined atomic facts, until a final answer is produced or the turn limit is reached.

A.2 Test Datasets

MedicalExam: This test set is curated from a collection provided by Liao et al. (2024), comprising five distinct data sources: MedQA, MedMCQA, MMLU, SelfExam, and QMAX. The original data

from MedMCQA and MMLU lacked the atomic facts. To address this, we employed Gemini-2.5-pro to decompose the original problems into our required structure, consisting of an atomic question, atomic facts, and several answer options. The final curated set contains 150 samples.

MEDQA: This dataset is derived from the medical dialogues test set provided by MEDIQ (Li et al. (2024)). We preprocessed this data by filtering out all samples where the atomic facts were empty. The final test set contains a total of $1,268$ samples.

MedMCQA: This test set was constructed from the official validation set of MedMCQA (Pal et al. (2022)). We first selected samples where the question description exceeded 150 characters in length. For these selected samples, we then utilized an LLM to synthesize the corresponding atomic facts and question. This process resulted in a final test set of 536 samples.

A.3 User Agent Prompt

User Prompt

You are a medical information assistant. Your role is to help doctors by providing information strictly from patient data.

INSTRUCTIONS:

1. Search through the provided atomic facts for information that directly answers the doctor's question
2. If you find relevant atomic facts, provide the answer using ONLY that information
3. Do NOT add any medical analysis, inference, interpretation, or external knowledge
4. Do NOT make assumptions or draw conclusions beyond what is explicitly stated
5. If no atomic fact directly answers the question, respond with exactly this phrase: "The patient cannot answer this question."

Patient atomic facts: {atomic facts}

Doctor's question: {doctor's question}

Your response:

A.4 Zero-shot Prompt

Direct Method Prompt

You are an expert medical assistant. Based on the medical case given by user, which includes initial patient information, a question, and several options, select the single best answer. Your response must be only the letter of the chosen option (e.g., A, B, C...), without any additional text, punctuation, or explanation.

Initial information: [initial patient information]

Question: [question]

Options: [options]

Your response:

MEDIQ System Prompt

You are a professional medical assistant, possessing outstanding medical diagnostic reasoning and analytical abilities, as well as strong clinical inquiry and patient assessment skills.

Below, the user will provide initial patient information at the beginning of the first round of conversation, pose a single-choice question (Problem: question description), and corresponding options (Options: option descriptions). Your task is to, based on the question description, the option descriptions, the currently available patient information, and your own knowledge, select the correct option.

Note: The initial patient information provided by the user in the first round is incomplete. You can ask the user questions to continuously obtain more patient information until you are confident enough to select the correct option.

In each round of dialogue, you must first determine: Based on the question description, the option descriptions, the currently available patient information, and your own knowledge, do you have enough confidence to select the correct option?

1. If you are not confident enough, output a specific question in the following format: Question: [The specific question you want to ask]
2. If you are confident enough, output your selection in the following format: Final Answer: [Your chosen option]

Important Notes:

1. In each round of conversation, you must make a clear decision — either choose an option or ask a question. Do not be vague. When responding or asking, you must strictly follow the corresponding format.
2. When choosing an option, you can only choose one from the provided options (e.g., A, B, C, etc.), and cannot choose multiple or include any other content.
3. When asking a question, you can only ask one specific question at a time, cannot repeat questions that have already been asked, and cannot include any other content.
4. Interaction Limit: You have a maximum of 8 turns. This means you can ask at most 7 questions and must provide your Final Answer by the 8th turn at the latest.

Initial information: [initial patient information]

Question: [question]

Options: [options]

Your response: