HAIL: HETEROGENEITY-AWARE LEARNING FROM HUMAN INTERVENTIONS WITH SUBOPTIMAL EXPERTS

Anonymous authors
Paper under double-blind review

ABSTRACT

Human-in-the-loop reinforcement learning (HITL-RL) has recently attracted increasing attention as a means to improve the learning efficiency and safety of reinforcement learning in complex tasks by incorporating human expert interventions. However, existing methods often implicitly assume that human experts can consistently provide high-quality guidance during training. In practice, human experts are often suboptimal, and the supervision they provide exhibits both intervention-level heterogeneity and temporal-level heterogeneity. To address these issues, we propose HAIL (Heterogeneity-Aware Learning from Human Interventions with Suboptimal Experts), a framework that explicitly models the heterogeneity of human supervision provided by suboptimal experts. Specifically, Quality-Weighted Behavioral Learning (QWBL) estimates the relative quality of different intervention behaviors and dynamically adjusts their optimization weights, allowing intervention samples with different quality levels to contribute differently to policy optimization. Effective-Horizon Retention (EH) selects preference information with higher learning value from different predicted future states while retaining part of the remaining candidates through sparse tail sampling, reducing the interference from preference information with lower learning value while maintaining trajectory-level supervision coverage. Experimental results in autonomous driving and continuous control environments demonstrate that HAIL improves policy performance with suboptimal human experts.

1 INTRODUCTION

Reinforcement learning (RL) has achieved significant progress in complex decision-making tasks, such as autonomous driving (Kendall et al., 2019) and robotic control (Ha et al., 2021). However, applying reinforcement learning in real-world environments remains challenging due to strict safety constraints (Li et al., 2022), and difficulties in reward design (Leike et al., 2018). Particularly in safety-critical tasks, agents often struggle to obtain reliable policies through large-scale trial-and-error exploration (Saunders et al., 2018). Therefore, how to effectively utilize external supervision to improve policy learning efficiency has become an important research problem in real-world reinforcement learning (Xue et al., 2023). Human-in-the-loop reinforcement learning (HITL-RL) incorporates human expert interventions to provide behavioral corrections or preference feedback (Spencer et al., 2020). These human signals help agents reduce ineffective exploration and improve learning efficiency (Peng et al., 2022). Compared with reinforcement learning methods that rely solely on environment rewards, HITL-RL is more suitable for tasks where rewards are difficult to define or exploration costs are high (Lee et al., 2021; Li et al., 2022). However, since human feedback is costly to obtain, practical systems usually have access to only limited human intervention data (Hoque et al., 2021; Lee et al., 2021). Therefore, how to fully exploit the useful information provided by human interventions is crucial for further improving HITL-RL performance.

In recent years, HITL-RL methods (Kelly et al., 2019; Celemin et al., 2022) have gradually evolved from direct behavioral imitation (Mandlekar et al., 2020) to approaches based on value propagation (Peng et al., 2023) and preference learning (Cai et al., 2025), improving feedback utilization efficiency by expanding information from individual interventions. However, existing methods often implicitly assume that different human interventions provide consistent and reliable supervi-

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

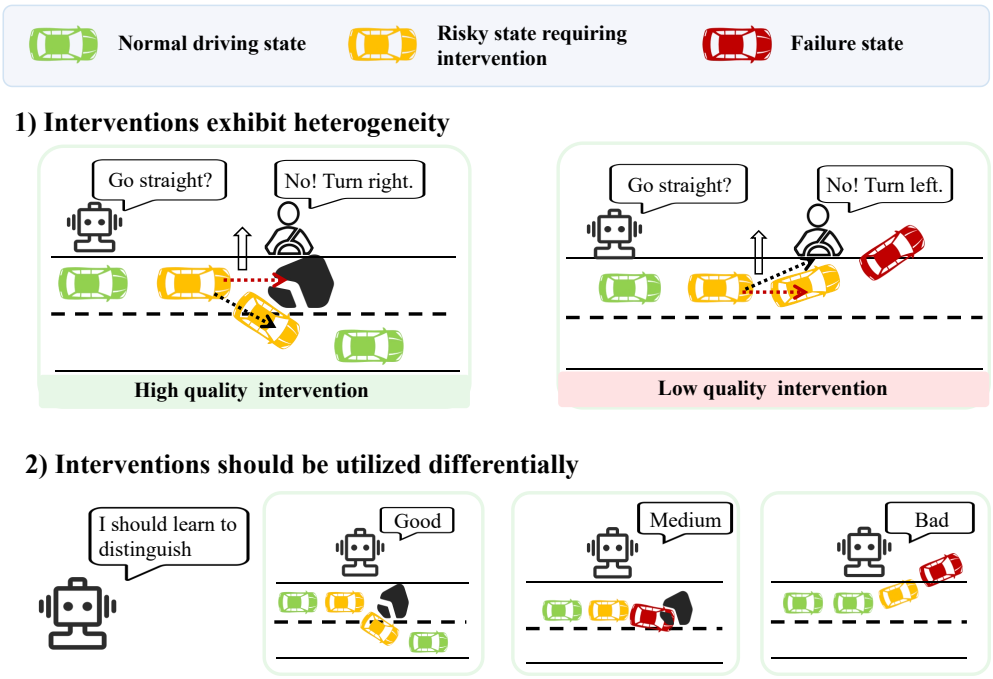


Figure 1: Illustration of different human interventions. Without assuming that human experts are optimal, we consider how to distinguish and utilize heterogeneous human interventions.

sion (Xue et al., 2025; Muslimani & Taylor, 2025). They also treat future preference information generated through preference propagation from a single intervention as equally effective supervision signals. This assumption essentially relies on the existence of an ideal human expert during training (Zhang et al., 2021; Xue et al., 2025). Such an assumption does not hold in practical scenarios, as it overlooks the inherent differences in human supervision during real interactions.

As shown in Fig. 1, we identify the key challenge under suboptimal human experts as how to distinguish and utilize human interventions with heterogeneous learning values. Specifically, this work focuses on two types of supervision heterogeneity in HITL-RL. First, different intervention behaviors may have different learning values due to variations in expert capability, tiredness, carelessness, or other interaction factors (Xue et al., 2023). We refer to this phenomenon as intervention-level heterogeneity. Treating all interventions as homogeneous supervision may allow interventions with lower learning value to negatively affect policy updates (Zhang et al., 2021; Xu et al., 2022). To address this issue, we propose Quality-Weighted Behavioral Learning (QWBL), which learns relative quality relationships among intervention behaviors and dynamically adjusts the contribution of different intervention samples in behavioral learning according to their quality scores. Second, we propagate the preference information from the intervention state to future states through future trajectory prediction. However, future states deviate from the original intervention state to different degrees, resulting in the preference information from the same intervention having different learning values. We refer to this phenomenon as temporal-level heterogeneity. Directly utilizing all preference information across the full prediction horizon may introduce preference information with lower learning value and interfere with preference learning. To address this issue, we propose Effective-Horizon Retention (EH). For each intervention, EH determines which future preference information associated with different predicted future states is selected to construct trajectory preference pairs for preference learning.

Based on these two mechanisms, we propose HAIL (Heterogeneity-Aware Learning from Human Interventions with Suboptimal Experts), a HITL-RL framework designed for the setting of suboptimal human experts. Different from existing methods that mainly focus on increasing the amount of intervention information or expanding supervision coverage, HAIL further investigates how to identify and utilize more reliable supervision information from human interventions. In the HAIL

framework, QWBL and EH address two complementary dimensions of human supervision. QWBL addresses the question of which intervention information is more valuable to learn among different human intervention events. EH addresses the question of which preference information is more valuable to learn among the preference information generated from a single intervention. The main contributions are summarized as follows:

(1) We identify and analyze the problem of heterogeneity in human intervention supervision in HITL-RL, showing that human intervention information exhibits differences from two perspectives: intervention-level heterogeneity and temporal-level heterogeneity.

(2) To address intervention-level heterogeneity, we propose Quality-Weighted Behavioral Learning (QWBL), which improves the utilization efficiency of intervention information with higher learning value through quality assessment and continuous weight adjustment, while reducing the interference of suboptimal interventions on policy optimization.

(3) To address temporal-level heterogeneity, we propose Effective-Horizon Retention (EH), which selects future preference information with higher learning value from the predicted trajectory and provides it to the agent for preference learning.

(4) Experiments on the MetaDrive autonomous driving environment and the Hopper-v5 continuous control task demonstrate the effectiveness of HAIL in improving policy performance under suboptimal human experts.

2 Related Work

Human-in-the-loop reinforcement learning (HITL-RL). HITL-RL improves the learning efficiency of reinforcement learning agents in complex tasks by incorporating human expert feedback (Christiano et al., 2017; Warnell et al., 2018; Guan et al., 2021). Early HITL-RL methods mainly utilized human action information through expert behavioral demonstration (Mandel et al., 2017). For example, Behavior Cloning (Ravichandar et al., 2020) and DAgger (Ross et al., 2011) guide policy learning by collecting expert behaviors. However, directly utilizing human action supervision typically relies on a large amount of high-quality expert behavior, while obtaining sufficient human feedback during real-world interactions can be costly (Hoque et al., 2021). Therefore, extracting richer information from limited human supervision has become an important research direction in HITL-RL (Luo et al., 2024).

Intervention-based Preference Learning. In reinforcement learning with human interventions, interventions provide not only action guidance but also implicitly encode relative preference relationships between agent behaviors and expert behaviors (Spencer et al., 2020; Macuglia et al., 2026). Some studies utilize human interventions to construct preference information for extracting richer supervision signals (Stiennon et al., 2020; Ouyang et al., 2022). Methods such as PVP (Peng et al., 2023) construct preference information from human interventions and expand supervision coverage through value propagation. However, due to the limited number of human interventions, preference information constructed from behavioral preferences is often sparse (Christiano et al., 2017), making it difficult to sufficiently capture long-term behavioral constraints in complex tasks and limiting the generalization ability of policy learning.

Preference Learning with Future Propagation. Preference-based reinforcement learning optimizes policies by learning relative preferences among behaviors or trajectories (Wirth et al., 2017; Liu et al., 2022; Hu et al., 2024). Existing approaches typically train reward models through preference comparisons (Christiano et al., 2017; Palan et al., 2019) or directly update policies based on the relative relationships between preferred and non-preferred behaviors (Rafailov et al., 2023; Hejna et al., 2024). PPL extends preference optimization to reinforcement learning with human interventions by predicting future trajectories and propagating behavior preference information to multiple future states (Cai et al., 2025) along the trajectory to construct trajectory-level preference pairs (Dai et al., 2026). This enables agents to learn richer behavioral constraints from a single intervention.

However, existing methods that propagate intervention preferences to future prediction horizons typically assume that preference information from different prediction horizons of the same intervention has the same learning value. In practice, as predicted states gradually deviate from the original intervention state, the degree of contextual change varies across different future horizons (Luan

et al., 2025; Heo et al., 2026), resulting in differences in the learning value of the corresponding preference information. Therefore, identifying effective prediction horizons and retaining preference information with higher learning value remain important challenges. Moreover, existing human intervention-based methods typically assume that different human interventions have the same learning value (Xue et al., 2023). Therefore, how to identify the quality differences and effectively utilize such differences to improve learning efficiency remains insufficiently explored.

3 Problem Definition

We study a HITL-RL problem under suboptimal human experts. In this setting, the agent learns a policy through interactions with the environment and receives additional supervision from human experts when interventions occur. We formulate the environment as a Markov Decision Process (MDP) $\mathcal{M} = \langle \mathcal{S}, \mathcal{A}, P, r, \gamma, d_0 \rangle$, where $\mathcal{S}$ and $\mathcal{A}$ denote the state and action spaces, respectively, $P(s'|s,a)$ represents the transition probability, $r(s,a)$ denotes the environment reward function, γ is the discount factor, and d_0 represents the initial state distribution. Reinforcement learning aims to learn a parameterized policy $\pi_\theta(a|s)$ that maximizes the expected cumulative return $J(\pi_\theta) = \mathbb{E}_{\tau \sim P_{\pi_\theta}}[\sum_{t=0}^{\infty} \gamma^t r(s_t, a_t)]$.

However, in safety-critical applications such as autonomous driving and robotic control, accurately designing task-specific reward functions is often challenging, while directly relying on trial-and-error exploration may result in inefficient or unsafe behaviors. Therefore, HAIL considers a reinforcement learning setting with human interventions, where policy optimization does not rely on manually designed complex reward functions and instead leverages human interventions as additional supervision signals. During the interaction process, the agent generates an action $a_t^n \sim \pi_\theta(\cdot|s_t)$ based on its current policy. When the agent behavior involves potential risks or deviates from desired objectives, the human expert provides a corrective action a_t^h and decides whether to intervene. We define an intervention indicator $I_t \in \{0, 1\}$, where $I_t = 1$ indicates that human intervention occurs. The final executed action is formulated as $a_t = (1 - I_t)a_t^n + I_t a_t^h$. Through the above online interaction process, we collect human intervention data $\mathcal{D}_h = \{(s_t, a_t^n, a_t^h, I_t, s_{t+1})\}$.

Therefore, the central problem studied in this work is: under suboptimal human experts, how can we identify and utilize human supervision with higher learning value from both intervention-level and temporal-level perspectives to improve policy learning efficiency?

4 Method

4.1 Framework Overview

Based on the intervention-level heterogeneity and temporal-level heterogeneity defined in Section 3, we propose HAIL (Heterogeneity-Aware Learning from Human Interventions with Suboptimal Experts). The overall framework is illustrated in Figure 2. Specifically, at state s_t, the agent first generates an original action a_t^n according to its current policy. To obtain future state information, HAIL adopts the short-term trajectory prediction mechanism introduced in PPL (Cai et al., 2025). Given the current state and the agent action, the future H-step state sequence is predicted as $f(s_t, a_t^n, H) = (s_t, \hat{s}_{t+1}, \ldots, \hat{s}_{t+H})$. The predicted trajectory is used to assist human experts in determining whether an intervention is required and to provide future states for preference learning. The trajectory prediction model itself is not modified in HAIL. When the predicted trajectory indicates that the agent behavior may deviate from the task objective or involve potential risks, the human expert provides a corrective action a_t^h. For each human intervention, HAIL extracts two types of supervision signals and processes them with QWBL and EH. After processing by QWBL and EH, HAIL obtains the quality-weighted behavioral learning objective $\mathcal{L}_{QWBL}$ and the preference learning objective $\mathcal{L}_{pref}^{EH}$ from each human intervention. These two types of supervision signals are jointly applied to the same policy network. Specifically, HAIL optimizes the policy with the following objective:

$$\mathcal{L}_{total}(\theta) = \lambda_{BL}\mathcal{L}_{QWBL}(\theta) + \mathcal{L}_{pref}^{EH}(\theta), \tag{1}$$

where λ_{BL} controls the contribution of $\mathcal{L}_{QWBL}$ to the joint optimization.

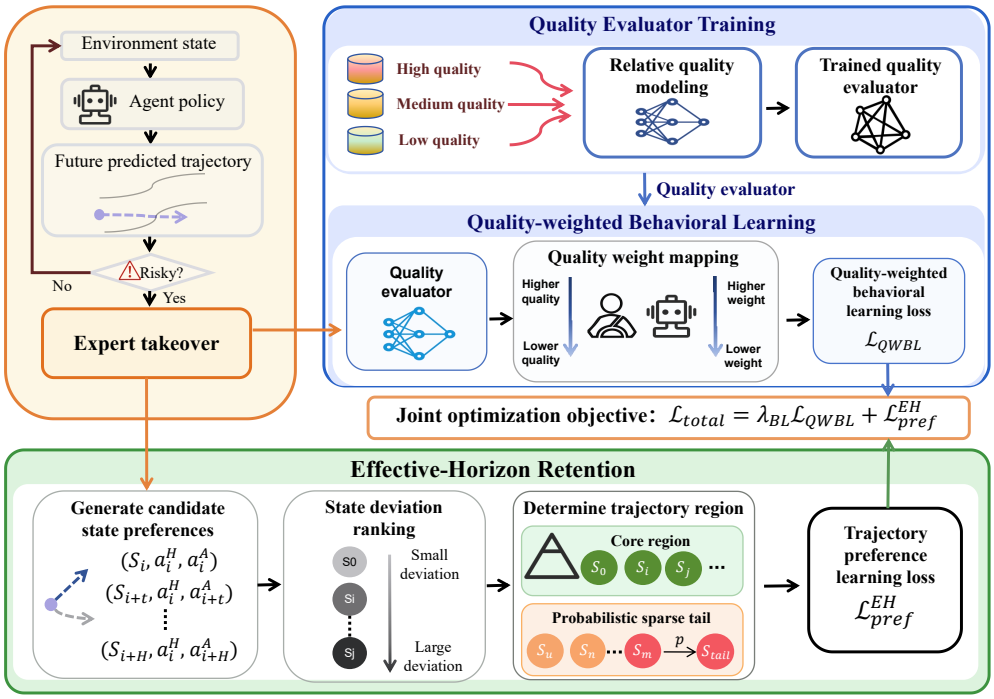


Figure 2: Overview of the HAIL framework. Human expert interventions are provided as shared inputs to QWBL and EH. QWBL learns intervention-level quality differences through a pretrained quality evaluator, while EH identifies effective preference horizons based on temporal-level heterogeneity. Both modules jointly optimize the final objective.

4.2 Quality-Weighted Behavioral Learning

Directly estimating the absolute learning value of individual interventions is challenging. Instead, QWBL exploits relative quality relationships among interventions by training a quality evaluator. These scores serve as a practical proxy for reflecting the learning value differences among interventions.

Quality Evaluator Learning. To learn the relative quality relationships among different intervention behaviors, human intervention segments with different quality levels are collected, where each segment is cropped into a fixed length. These quality labels are constructed based on intervention segment execution outcomes. The detailed construction process is provided in Appendix D.

During training, we sample intervention segments from different quality levels, denoted as $\tau_i = \{(s_t, a_t), ..., (s_{t+L}, a_{t+L})\}$. The quality evaluator Q_ϕ learns to assess intervention segment quality by capturing relative quality relationships between intervention segments τ_i and τ_j. Specifically, for a segment τ_i, the quality evaluator evaluates each state-action pair in τ_i sequentially and aggregates them into an overall quality score $\bar{r}_i$. For any two segments τ_i and τ_j, we define the score difference and label difference as $\Delta r_{ij} = \bar{r}_i - \bar{r}_j$ and $\Delta y_{ij} = y_i - y_j$. Then, we construct the relative ranking weight $c_{ij} = \exp(\max(\Delta y_{ij}, 0))$ and define the global quality ranking loss as:

$$\mathcal{L}_{global} = \frac{1}{B_q^2} \sum_{i=1}^{B_q} \sum_{j=1}^{B_q} \exp(-c_{ij} \Delta r_{ij}) \tag{2}$$

where B_q denotes the number of segments within a batch. $\mathcal{L}_{\text{global}}$ enables the quality evaluator to assign distinguishable scores to intervention segments with different quality levels, without predicting absolute quality values. Furthermore, to constrain the consistency of scores within the same

segment, we introduce a local consistency loss:

$$\mathcal{L}_{local} = \frac{1}{B_q L_q^2} \sum_{i=1}^{B_q} \sum_{l=1}^{L_q} \sum_{l'=1}^{L_q} \left[r_\phi(s_{i,l}, a_{i,l}) - r_\phi(s_{i,l'}, a_{i,l'}) \right]^2 \tag{3}$$

Finally, the quality evaluator is trained by jointly optimizing the global quality ranking loss and the local consistency loss:

$$\mathcal{L}_{eval} = \mathcal{L}_{global} + \mathcal{L}_{local}.$$

After training, the parameters of the quality evaluator Q_ϕ are fixed and used for subsequent online policy learning. The training details and performance evaluation of the quality evaluator are provided in Appendix C and Appendix D, respectively.

Quality-aware Intervention Weighting. During the online training stage, the fixed quality evaluator Q_ϕ evaluates the human intervention (s_i, a_i^h) and outputs a quality score $q_i = Q_\phi(s_i, a_i^h)$. QWBL further maps q_i into continuous weights w_i. During the mapping process, directly normalizing quality scores based on fixed score bounds is difficult to adapt to the changing distribution of intervention quality during online training. Therefore, QWBL introduces an adaptive upper bound M_t. M_t is dynamically adjusted according to the quality score distribution of recent intervention samples. The detailed update procedures of M_t are provided in Appendix C. Finally, the quality weight of (s_i, a_i^h) is defined as

$$w_i = \text{clip}\left(\frac{\max(q_i, q_{\min}) - q_{\min}}{M_t - q_{\min}}, w_{\min}, 1 \right) \tag{4}$$

where $q_{\min}$ denotes the lower bound of quality scores and $w_{\min} > 0$ represents the minimum supervision weight. QWBL allows different intervention behaviors to contribute to policy updates with different weights according to their quality. A larger w_i indicates that the corresponding intervention behavior has a higher relative quality level, resulting in a larger gradient contribution during policy optimization. A smaller w_i indicates relatively lower quality, but the corresponding intervention sample is not completely discarded, thereby preserving sample diversity. The quality-weighted behavioral learning objective is defined as:

$$\mathcal{L}_{QWBL}(\theta) = \frac{1}{B} \sum_{i=1}^{B} w_i \frac{1}{d_a} \left\| \pi_\theta(s_i) - a_i^h \right\|_2^2. \tag{5}$$

where B denotes the batch size of intervention samples and d_a denotes the action dimension.

4.3 Effective-Horizon Retention

To measure the contextual deviation of different future states relative to the intervention state, EH computes the state deviation between the k-th predicted future state and the intervention state as $d_{t,k} = \|\hat{s}_{t+k} - \hat{s}_t\|_2,\ k \geq 1$. Rather than explicitly estimating the learning value of preference information, this work considers $d_{t,k}$ as a proxy metric for selecting preference information with higher learning value. A smaller state deviation $d_{t,k}$ indicates that the preference information from the intervention state is more likely to remain applicable at state s_{t+k}, suggesting that the corresponding preference pair is more valuable for preference learning.

Based on $d_{t,k}$, EH ranks the candidate future states $\hat{s}_{t+1}, ..., \hat{s}_{t+H}$ generated in ascending order of state deviation. Let the indices of the ranked future states be $k_t^{(1)}, k_t^{(2)}, ..., k_t^{(N_t)}$. Subsequently, EH determines the core supervision budget $K_\rho = \min(N_t, \lceil \rho H \rceil)$ according to the core retention ratio ρ and the maximum prediction horizon H. The first K_ρ candidates after sorting are selected as the core region $\mathcal{J}_t^{core}$.

However, retaining only the core region reduces the coverage advantage of trajectory preferences over state preferences. Therefore, EH further applies a sparse tail sampling mechanism to candidates outside the core region. Tail sampling is triggered with a low probability P_{tail}. When tail sampling is activated, candidates are sampled according to an exponentially decaying probability $P_t(j)$ based on their ranking positions. Let the additional candidate set obtained through tail sampling be denoted as $\mathcal{J}_t^{tail}$.

Preference Optimization. The final trajectory preference pairs are represented as $\mathcal{P}_t^{EH} = \{p_{t,k} \mid k \in \mathcal{J}_t^{core} \cup \mathcal{J}_t^{tail}\}$, where $p_{t,k} = (\hat{s}_{t+k}, a_t^h, a_t^n)$. Here, $p_{t,k}$ indicates that at the predicted future state $\hat{s}_{t+k}$, the intervention action a_t^h is still considered more applicable than the original agent action a_t^n.

During training, the trajectory preference pairs generated from all interventions are further aggregated as $\mathcal{D}_{pref}^{EH} = \bigcup_t \mathcal{P}_t^{EH}$. Only the preference information contained in the trajectory preference pair set $\mathcal{D}_{pref}^{EH}$ participates in preference learning. EH does not directly modify the preference learning objective, but determines which preference information participates in preference learning. Specifically, HAIL adopts a preference learning optimization approach based on preference comparison (Rafailov et al., 2023). Based on $\mathcal{D}_{pref}^{EH}$, the preference optimization objective is defined as

$$\mathcal{L}_{\text{pref}}^{\text{EH}}(\theta) = -\mathbb{E}_{(s,a^+,a^-)\sim\mathcal{D}_{\text{pref}}^{\text{EH}}} \left[\log \sigma \left(\beta \log \pi_\theta(a^+|s) - \beta \log \pi_\theta(a^-|s)\right)\right]. \tag{6}$$

where $\log \pi_\theta(a|s)$ reflects the tendency of the policy to generate action a under state s, $\sigma(\cdot)$ denotes the sigmoid function, and β controls the scale of preference comparison.

5 Experiments

5.1 Experimental Setup

Tasks. We evaluate HAIL in two environments: the MetaDrive autonomous driving environment (Li et al., 2023) and the MuJoCo Hopper-v5 continuous control environment (Todorov et al., 2012). MetaDrive serves as the primary experimental environment for evaluating the policy learning performance of HAIL and different baseline methods under human expert interventions, while Hopper-v5 is used as a supplementary environment to evaluate the applicability of HAIL across tasks with different characteristics and dynamics. In both environments, the core mechanisms of QWBL and EH remain unchanged, with only the environment interfaces adapted accordingly.

Simulated Human Experts. Since real human feedback experiments are costly and difficult to reproduce consistently, we adopt a simulated human intervention setting (Cai et al., 2025; Kwon et al., 2026). Specifically, SAC policies (Haarnoja et al., 2018) are used to simulate human experts, and suboptimal human experts are constructed by varying the capability levels of expert policies. The training configurations and details of the SAC expert policies are provided in Appendix B. Experiments with SAC experts of different performance levels are provided in Appendix F.

Experimental Details. During online policy optimization, environmental rewards are not used. The environmental reward is used for performance evaluation during testing. All methods use the same training budget and evaluation protocol. All methods use the same expert policies. Detailed implementation settings, training hyperparameters, and environment configurations are provided in Appendix B.

Evaluation Metrics. We evaluate methods in terms of task performance and human intervention requirements. In MetaDrive, we use Success Rate, Mean Reward, and Route Completion to evaluate policy performance. For online HITL-RL methods, we additionally report Total Takeover Count to measure the number of human interventions required during training. For offline imitation learning methods, this metric is not applicable since they do not involve online human interaction. In the Hopper-v5 environment, we use Mean Episode Reward, Episode Length, and Takeover Cost to evaluate policy performance and expert intervention cost. All methods are evaluated without human interventions during testing using five random seeds, and the mean performance and standard deviation are reported. The safety performance analysis for the MetaDrive task, including Crash Rate and Out-of-road Rate, is further provided in Appendix G.

Baselines. We compare HAIL with several representative methods. In the MetaDrive environment, we compare HAIL with BC (Ravichandar et al., 2020), GAIL (Ho & Ermon, 2016), DAgger (Ross et al., 2011), HACO (Li et al., 2022), PVP (Peng et al., 2023), PVP4Real (Peng et al., 2025), and PPL (Cai et al., 2025). BC and GAIL are included as offline imitation learning baselines, and DAgger is an interactive imitation learning baseline. HACO, PVP, PVP4Real, and PPL are considered representative HITL-RL baselines. In the Hopper-v5 environment, we select PVP, PVP4Real, and PPL as comparison methods to evaluate the effectiveness of HAIL across different continuous control tasks.

Table 1: Performance comparison of different methods on MetaDrive.

Method	Human-in-the-Loop	Training		Testing		
		Total Takeover Count ↓	Total Timesteps	Success Rate ↑	Mean Reward ↑	Route Completion ↑
Expert Policy	-	-	-	0.360 ± 0.012	240.613 ± 15.236	0.756 ± 0.013
BC	×	-	20K	0.000 ± 0.000	28.328 ± 20.217	0.123 ± 0.005
GAIL	×	-	20K	0.082 ± 0.053	87.313 ± 10.196	0.347 ± 0.008
DAgger	✓	5974.748 ± 1160.833	20K	0.097 ± 0.024	125.540 ± 24.473	0.505 ± 0.020
HACO	✓	8630.191 ± 1676.758	20K	0.060 ± 0.017	89.626 ± 7.826	0.304 ± 0.008
PVP	✓	6804.574 ± 1322.059	20K	0.190 ± 0.087	183.853 ± 15.474	0.565 ± 0.075
PVP4Real	✓	6140.713 ± 1193.078	20K	0.213 ± 0.050	209.404 ± 19.932	0.606 ± 0.057
PPL	✓	4978.957 ± 967.360	20K	0.280 ± 0.035	207.809 ± 11.757	0.619 ± 0.028
HAIL	✓	$\mathbf{3817.200 \pm 741.643}$	20K	$\mathbf{0.523 \pm 0.081}$	$\mathbf{286.233 \pm 15.415}$	$\mathbf{0.780 \pm 0.038}$

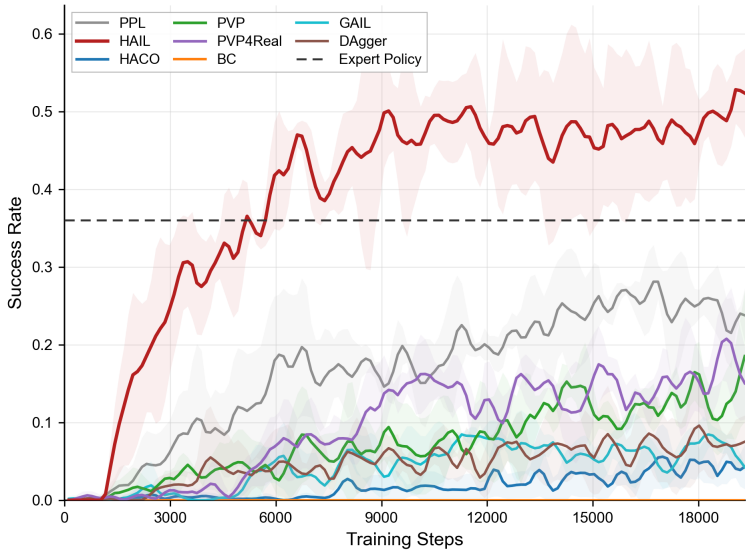


Figure 3: Training curves of different methods in the MetaDrive environment.

5.2 Comparison with Baselines.

Comparison with Baselines. Table 1 summarizes the performance comparison of different methods in the MetaDrive environment. Overall, HAIL outperforms all compared methods on success rate, mean reward, and route completion. Meanwhile, HAIL requires the fewest human interventions measured by Total Takeover Count among online human intervention methods. Furthermore, HAIL achieves higher task performance than the suboptimal expert policy during evaluation. This indicates that HAIL can learn a higher-performing policy under suboptimal expert interventions. Figure 3 shows the training curves. HAIL achieves higher performance with fewer training steps and maintains a stable improvement trend throughout training.

Additional Evaluation on Hopper-v5. Table 2 presents the comparison of HAIL with existing HITL-RL baselines. HAIL still achieves the highest Mean Episode Reward and Mean Episode Length, while obtaining the lowest Takeover Cost.

5.3 Ablation and Component Analysis.

To investigate the effectiveness of QWBL and EH, we conduct ablation studies on the MetaDrive environment. For QWBL, we compare different quality weighting strategies. For EH, we compare different future preference selection strategies. All experiments use the same experimental settings, and the results are summarized in Table 3.

Quality Weighting Strategy. Removing QWBL degrades performance, indicating that different interventions do not provide equivalent learning value. Uniform down-weighting and binary filtering

Table 2: Performance comparison of different methods on Hopper-v5.

Method	Total Timesteps	Mean Episode Reward ↑	Mean Episode Length ↑	Takeover Cost ↓
PVP	150K	1200.237 ± 157.140	362.047 ± 48.852	22680.632 ± 3217.654
PVP4Real	150K	1687.703 ± 485.143	504.813 ± 144.457	24850.260 ± 2092.225
PPL	150K	2219.488 ± 997.290	650.453 ± 278.107	15969.371 ± 797.120
HAIL	**150K**	$\mathbf{3093.576 \pm 46.526}$	$\mathbf{972.263 \pm 21.406}$	$\mathbf{12134.256 \pm 411.755}$

Table 3: Ablation and design choice analysis of QWBL and EH on MetaDrive.

Method	Quality Weighting Strategy	Future Preference Selection	Testing		
			Success Rate ↑	Route Completion ↑	Total Takeover Count ↓
HAIL w/o QWBL	No quality weighting	EH	0.341 ± 0.031	0.593 ± 0.054	8489.200 ± 1764.230
QWBL-Uniform Weighting	Uniform down-weighting	EH	0.427 ± 0.064	0.633 ± 0.119	5147.000 ± 750.917
QWBL-Binary Filtering	Binary quality filtering	EH	0.460 ± 0.053	0.720 ± 0.037	4271.600 ± 1182.939
HAIL w/o EH	QWBL	Full prediction horizon	0.353 ± 0.070	0.611 ± 0.098	7259.800 ± 653.508
EH-Fixed first-K horizon	QWBL	Fixed first-3 temporal candidates	0.380 ± 0.140	0.627 ± 0.044	5296.000 ± 1713.876
EH-Core Only	QWBL	Core candidates only	0.490 ± 0.113	0.666 ± 0.024	4570.200 ± 1053.466
EH-Tail Only	QWBL	Tail candidates only	0.367 ± 0.061	0.625 ± 0.052	6463.000 ± 495.840
HAIL	**QWBL**	**EH**	$\mathbf{0.513 \pm 0.081}$	$\mathbf{0.780 \pm 0.038}$	$\mathbf{3817.200 \pm 741.643}$

achieve better performance than the variant without QWBL but remain inferior to the full QWBL method, indicating that interventions should be differentially utilized according to their relative quality.

Future Preference Selection Strategy. Using the Full prediction horizon without EH leads to performance degradation, showing that preference information from different prediction horizons has different learning values. Fixed-horizon, core-only, and tail-only selection strategies also perform worse than the full EH method, indicating that deviation-based selection is more suitable than fixed temporal selection and that retaining sparse tail candidates helps preserve preference coverage.

Removing either QWBL or EH results in performance degradation, demonstrating that QWBL and EH provide complementary improvements by addressing intervention-level and temporal-level heterogeneity, respectively. Additional analysis of QWBL and EH mechanisms is provided in Appendix E. We further conduct sensitivity analysis to study the impact of key hyperparameters on HAIL performance, with detailed results provided in Appendix H.

6 Conclusion

In this work, we propose HAIL, a heterogeneity-aware HITL-RL framework for learning under suboptimal human experts. HAIL identifies two types of heterogeneity in human supervision, including intervention-level heterogeneity and temporal-level heterogeneity, and proposes QWBL and EH to address them, respectively. Finally, HAIL enables the identification of heterogeneous learning values among human interventions. Based on these identified values, HAIL performs heterogeneity-aware utilization of supervision information. Experimental results demonstrate that HAIL effectively improves policy learning performance under suboptimal human experts while reducing the number of required human interventions during training.

Limitations. (1) QWBL relies on a trained quality evaluator to assess intervention quality. Since the evaluator parameters remain fixed after training, evaluation errors may affect the quality weighting of different intervention samples, and the evaluator cannot further adapt using newly collected human interventions during online training. (2) The quality score serves as a proxy metric rather than a direct estimate of intervention quality. (3) EH uses $d_{t,k}$ as a proxy metric for the learning value of future preference information. Future work will explore online adaptive quality evaluation, more accurate modeling of intervention behavior quality and preference information value.

AI use statement

During the preparation of this manuscript, we used large language models (LLMs) for language refinement, including improving clarity, conciseness, and grammatical correctness of the paper. We also used LLMs for retrieval and discovery of related research works to assist literature exploration.

LLMs were not used to generate experimental results, design the proposed algorithms, develop the core research ideas, analyze experimental findings, or produce scientific claims. All technical contributions, methodological designs, experiments, and interpretations were conceived, conducted, and verified by the authors.

We have carefully reviewed all AI-assisted content and take full responsibility for the final content of this paper, including all text, claims, and experimental results.

Ethics statement

This research adheres to the ICLR Code of Ethics. All experiments were conducted in publicly available simulation environments. To reduce ethical concerns and avoid privacy issues associated with collecting data from human subjects, human experts were simulated using pre-trained reinforcement learning agents. This simulated expert setting enables the evaluation of human-in-the-loop reinforcement learning methods without collecting private human information, thereby aligning with principles of responsible research.

Reproducibility statement

We provide detailed descriptions of the proposed framework, optimization objectives, algorithm procedures, and experimental settings to facilitate reproducibility. The main paper describes the motivation, formulation, and methodology of HAIL, while the appendices provide additional implementation details, hyperparameters, environment configurations, and quality evaluator training procedures. All experiments are conducted in publicly available simulation environments, and the experimental protocols, evaluation metrics, and training configurations are reported in detail. The supplementary materials provide anonymized code.

References

Haoyuan Cai, Zhenghao Peng, and Bolei Zhou. Predictive preference learning from human interventions. In *Advances in Neural Information Processing Systems*, volume 38, 2025.

Carlos Celemin, Rodrigo Pérez-Dattari, Eugenio Chisari, Giovanni Franzese, Leandro de Souza Rosa, Ravi Prakash, Zlatan Ajanovic, Marta Ferraz, Abhinav Valada, and Jens Kober. Interactive imitation learning in robotics: A survey. *Foundations and Trends in Robotics*, 2022.

Paul F. Christiano, Jan Leike, Tom B. Brown, Miljan Martic, Shane Legg, and Dario Amodei. Deep reinforcement learning from human preferences. In *Advances in Neural Information Processing Systems*, volume 30, pp. 4299–4307, 2017.

Lifei Dai, Yiqun Liu, Hao Zhang, Zhuping Wang, and Changzhu Zhang. Trajectory-constrained human-guided reinforcement learning for autonomous driving. *IEEE Transactions on Neural Networks and Learning Systems*, pp. 1–14, 2026.

Lin Guan, Mudit Verma, Sihang Guo, Ruohan Zhang, and Subbarao Kambhampati. Widening the pipeline in human-guided reinforcement learning with explanation and context-aware data augmentation. In *Advances in Neural Information Processing Systems*, volume 34, 2021.

Sehoon Ha, Peng Xu, Zhenyu Tan, Sergey Levine, and Jie Tan. Learning to walk in the real world with minimal human effort. In *Proceedings of the 2020 Conference on Robot Learning*, volume 155 of *Proceedings of Machine Learning Research*, pp. 1110–1120. PMLR, 2021.

Tuomas Haarnoja, Aurick Zhou, Pieter Abbeel, and Sergey Levine. Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor. In *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pp. 1861–1870. PMLR, 2018.

Joey Hejna, Rafael Rafailov, Harshit Sikchi, Chelsea Finn, Scott Niekum, W. Bradley Knox, and Dorsa Sadigh. Contrastive preference learning: Learning from human feedback without RL. In *International Conference on Learning Representations*, 2024.

Jongkook Heo, Jaehoon Kim, Young Jae Lee, Min Gu Kwak, Youngjoon Park, and Seoung Bum Kim. Policy likelihood-based query sampling and critic-exploited reset for efficient preference-based reinforcement learning. In *The Fourteenth International Conference on Learning Representations*, 2026.

Jonathan Ho and Stefano Ermon. Generative adversarial imitation learning. In *Advances in Neural Information Processing Systems*, volume 29, pp. 4565–4573, 2016.

Ryan Hoque, Ashwin Balakrishna, Ellen Novoseller, Albert Wilcox, Daniel S. Brown, and Ken Goldberg. ThriftyDAgger: Budget-aware novelty and risk gating for interactive imitation learning. In *Proceedings of the 5th Conference on Robot Learning*, volume 164 of *Proceedings of Machine Learning Research*, pp. 598–608. PMLR, 2021.

Xiao Hu, Jianxiong Li, Xianyuan Zhan, Qing-Shan Jia, and Ya-Qin Zhang. Query-policy misalignment in preference-based reinforcement learning. In *The Twelfth International Conference on Learning Representations*, 2024.

Michael Kelly, Chelsea Sidrane, Katherine Driggs-Campbell, and Mykel J. Kochenderfer. HG-DAgger: Interactive imitation learning with human experts. In *2019 International Conference on Robotics and Automation (ICRA)*, pp. 8077–8083. IEEE, 2019.

Alex Kendall, Jeffrey Hawke, David Janz, Przemyslaw Mazur, Daniele Reda, John-Mark Allen, Vinh-Dieu Lam, Alex Bewley, and Amar Shah. Learning to drive in a day. In *2019 International Conference on Robotics and Automation (ICRA)*, pp. 8248–8254. IEEE, 2019.

Daehyeok Kwon, Seung-Woo Seo, and Sang-Hyun Lee. Self-improving autonomous vehicles via real-world reinforcement learning. In *Proceedings of the Thirty-Fifth International Joint Conference on Artificial Intelligence*, pp. 4430–4438, 2026.

Kimin Lee, Laura M. Smith, and Pieter Abbeel. PEBBLE: Feedback-efficient interactive reinforcement learning via relabeling experience and unsupervised pre-training. In *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pp. 6152–6163. PMLR, 2021.

Jan Leike, David Krueger, Tom Everitt, Miljan Martic, Vishal Maini, and Shane Legg. Scalable agent alignment via reward modeling: A research direction. *arXiv preprint arXiv:1811.07871*, 2018.

Quanyi Li, Zhenghao Peng, and Bolei Zhou. Efficient learning of safe driving policy via human-AI copilot optimization. In *International Conference on Learning Representations*, 2022.

Quanyi Li, Zhenghao Peng, Lan Feng, Qihang Zhang, Zhenghai Xue, and Bolei Zhou. MetaDrive: Composing diverse driving scenarios for generalizable reinforcement learning. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(3):3461–3475, 2023.

Runze Liu, Fengshuo Bai, Yali Du, and Yaodong Yang. Meta-reward-net: Implicitly differentiable reward learning for preference-based reinforcement learning. In *Advances in Neural Information Processing Systems*, 2022.

Yao Luan, Ni Mu, Yiqin Yang, Bo Xu, and Qing-Shan Jia. STAIR: Addressing stage misalignment through temporal-aligned preference reinforcement learning. In *Advances in Neural Information Processing Systems*, 2025.

Jianlan Luo, Perry Dong, Yuexiang Zhai, Yi Ma, and Sergey Levine. RLIF: Interactive imitation learning as reinforcement learning. In *International Conference on Learning Representations*, 2024.

Maël Macuglia, Paul Friedrich, and Giorgia Ramponi. Fine-tuning behavioral cloning policies with preference-based reinforcement learning. In *International Conference on Learning Representations*, 2026.

Travis Mandel, Yun-En Liu, Emma Brunskill, and Zoran Popovic. Where to add actions in human-in-the-loop reinforcement learning. In *Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence*, pp. 2322–2328. AAAI Press, 2017.

Ajay Mandlekar, Danfei Xu, Roberto Martín-Martín, Yuke Zhu, Li Fei-Fei, and Silvio Savarese. Human-in-the-loop imitation learning using remote teleoperation. *arXiv preprint arXiv:2012.06733*, 2020.

Calarina Muslimani and Matthew E. Taylor. Leveraging sub-optimal data for human-in-the-loop reinforcement learning. In *International Conference on Learning Representations*, 2025.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll L. Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. Training language models to follow instructions with human feedback. *arXiv preprint arXiv:2203.02155*, 2022.

Malayandi Palan, Gleb Shevchuk, Nicholas Charles Landolfi, and Dorsa Sadigh. Learning reward functions by integrating human demonstrations and preferences. In *Proceedings of Robotics: Science and Systems*, 2019.

Zhenghao Peng, Quanyi Li, Chunxiao Liu, and Bolei Zhou. Safe driving via expert guided policy optimization. In *Proceedings of the 5th Conference on Robot Learning*, volume 164 of *Proceedings of Machine Learning Research*, pp. 1554–1563. PMLR, 2022.

Zhenghao Peng, Wenjie Mo, Chenda Duan, Quanyi Li, and Bolei Zhou. Learning from active human involvement through proxy value propagation. In *Advances in Neural Information Processing Systems*, volume 36, pp. 77969–77992, 2023.

Zhenghao Peng, Zhizheng Liu, and Bolei Zhou. Data-efficient learning from human interventions for mobile robots. In *2025 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 7407–7414. IEEE, 2025.

Rafael Rafailov, Archit Sharma, Eric Mitchell, Christopher D. Manning, Stefano Ermon, and Chelsea Finn. Direct preference optimization: Your language model is secretly a reward model. In *Advances in Neural Information Processing Systems*, volume 36, pp. 53728–53741, 2023.

Harish Ravichandar, Athanasios S. Polydoros, Sonia Chernova, and Aude Billard. Recent advances in robot learning from demonstration. *Annual Review of Control, Robotics, and Autonomous Systems*, 3:297–330, 2020.

Stéphane Ross, Geoffrey Gordon, and Drew Bagnell. A reduction of imitation learning and structured prediction to no-regret online learning. In *Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics*, volume 15 of *Proceedings of Machine Learning Research*, pp. 627–635. JMLR Workshop and Conference Proceedings, 2011.

William Saunders, Girish Sastry, Andreas Stuhlmueller, and Owain Evans. Trial without error: Towards safe reinforcement learning via human intervention. In *Proceedings of the 17th International Conference on Autonomous Agents and MultiAgent Systems*, pp. 2067–2069. International Foundation for Autonomous Agents and Multiagent Systems, 2018.

Jonathan Spencer, Sanjiban Choudhury, Matt Barnes, Matthew Schmittle, Mung Chiang, Peter Ramadge, and Siddhartha Srinivasa. Learning from interventions: Human-robot interaction as both explicit and implicit feedback. In *Proceedings of Robotics: Science and Systems*, 2020.

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

Emanuel Todorov, Tom Erez, and Yuval Tassa. MuJoCo: A physics engine for model-based control. In *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 5026–5033. IEEE, 2012.

Garrett Warnell, Nicholas R. Waytowich, Vernon Lawhern, and Peter Stone. Deep TAMER: Interactive agent shaping in high-dimensional state spaces. In *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence*, pp. 1545–1554. AAAI Press, 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.

Haoran Xu, Xianyuan Zhan, Honglei Yin, and Huiling Qin. Discriminator-weighted offline imitation learning from suboptimal demonstrations. In *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pp. 24725–24742. PMLR, 2022.

Zhenghai Xue, Zhenghao Peng, Quanyi Li, Zhihan Liu, and Bolei Zhou. Guarded policy optimization with imperfect online demonstrations. In *International Conference on Learning Representations*, 2023.

Zhenghai Xue, Bo An, and Shuicheng Yan. Policy optimization under imperfect human interactions with agent-gated shared autonomy. In *The Thirteenth International Conference on Learning Representations*, 2025.

Songyuan Zhang, Zhangjie Cao, Dorsa Sadigh, and Yanan Sui. Confidence-aware imitation learning from demonstrations with varying optimality. In *Advances in Neural Information Processing Systems*, volume 34, pp. 12340–12350, 2021.

A Algorithm Pseudocode

We summarize the training procedure of HAIL in Algorithm 1.

Algorithm 1 Overall Training Procedure of HAIL

Input: Prediction horizon H, core retention ratio ρ, tail sampling probability p_{tail}, decay coefficient λ_{tail}, and frozen quality evaluator Q_ϕ.
for each timestep $k = 0, H, 2H, \dots$ **do**
  Agent samples action $a_t^n \sim \pi_\theta(\cdot|s_t)$.
  Generate future predicted states $\hat{\tau}_t = f(s_t, a_t^n, H) = (s_t, \tilde{s}_{t+1}, \cdots, \tilde{s}_{t+H})$
  Human expert observes the predicted trajectory $\hat{\tau}_t$ and determines whether intervention is required based on the predicted future risk.
  for timestep $t = k, k+1, \cdots, k+H-1$ **do**
    if Human expert intervenes **then**
      Human expert takes action $a_t^h \sim \pi_h(s_t)$.
      Add (s_t, a_t^h) to the human intervention dataset $\mathcal{D}_h$.
      Compute intervention quality score $q_t = Q_\phi(s_t, a_t^h)$.
      Compute quality weight w_t from q_t for L_{QWBL}.
      Generate candidate predicted state preferences $P_t = \{(\tilde{s}_{t+i}, a_t^h, a_t^n)\}_{i=1,\cdots,H}$
      Construct effective trajectory preference pairs $\mathcal{P}_{EH}$ using EH.
      Add $\mathcal{P}_{EH}$ to the preference dataset $\mathcal{D}_{pref}$ for L_{pref}^{EH}.
      Observe $s_{t+1} \sim P(\cdot|s_t, a_t^h)$.
    else
      Agent samples action $a_t^n \sim \pi_\theta(\cdot|s_t)$.
      Observe $s_{t+1} \sim P(\cdot|s_t, a_t^n)$.
    end if
  end for
  Update policy π_θ using $\mathcal{L}_{total}(\theta) = \lambda_{BL}\mathcal{L}_{QWBL}(\theta) + \mathcal{L}_{pref}^{EH}(\theta)$.
end for
Output: Policy π_θ.

B Supplementary Experimental Settings

B.1 Common Training Configuration

All methods use the same expert policy and intervention trigger criterion. The common training configurations are summarized in Table 4.

Table 4: Common training configuration

Category	Parameter	Value
Training	Random Seeds	5
	Learning Starts	10
	Expert Policy	Same expert policy
	Intervention Mechanism	Same trigger criterion
	Optimizer	Adam
MetaDrive	Total Training Steps	20,000
Hopper-v5	Total Training Steps	150,000
Evaluation	Report Metric	Mean $\pm$ Std

Table 5: Hyperparameters for SAC expert training

Category	Hyperparameter	Value
Algorithm	Algorithm	SAC
Training	Optimizer	Adam
	Learning Rate	1×10^{-4}
	Batch Size	1024
	Replay Buffer Size	1,000,000
	Learning Starts	10,000 steps
	Discount Factor γ	0.99
	Target Update Rate τ	0.005
	Gradient Steps	1
	Train Frequency	1 step

B.2 Expert Policy Training Configuration

The SAC expert policies used as simulated human experts are trained with the configurations summarized in Table 5.

B.3 Shared Optimization Configuration

The optimization configurations used in different methods are summarized in Table 6. Shared parameters are consistently applied across different methods, while algorithm-specific parameters are only applicable to methods using the corresponding components.

B.4 Method-specific Hyperparameters

Method-specific hyperparameters that differ from the shared configurations are listed separately in Table 7. Parameters not listed in this table follow the default shared configurations in Table 6. For classical imitation learning baselines (BC, GAIL, and DAgger), we follow their standard implementations and default training configurations. The same training budget and evaluation protocol are applied for fair comparison.

C QWBL Implementation Details

C.1 Quality Evaluator Training Details

Before training the main policy, QWBL first trains a quality evaluator offline to learn the relative quality relationships among different intervention behaviors. The evaluator is optimized using the constraint defined in Section 4.2. Therefore, the quality evaluator learns the relative quality relationships among different behaviors rather than an absolute score with a fixed physical meaning.

Table 6: Shared optimization configuration

Parameter	Value
Batch Size	1024
Learning Rate	1×10^{-4}
Learning Starts	10
Replay Buffer Size	50,000
Soft Update Coefficient τ	0.005
Policy Network	[256,256]
CQL Coefficient	1.0
Intervention Threshold	0.95
Q Value Bound	1.0
Evaluation Frequency (MetaDrive)	150 steps
Evaluation Episodes (MetaDrive)	50
Evaluation Frequency (Hopper-v5)	2,000 steps
Evaluation Episodes (Hopper-v5)	10

Table 7: Method-specific hyperparameters

Method	Parameter	Value
HACO	Discount Factor γ	0.99
PVP	BC Loss Weight	0.0
PVP4Real	BC Loss Weight	1.0
PPL	BC Loss Weight	1.0
PPL	Prediction Horizon H	20
PPL	Preference Horizon L	19
HAIL	BC Loss Weight	1.0
HAIL	Quality Levels	$\mathcal{D}^{(0)}, \mathcal{D}^{(1)}, \mathcal{D}^{(2)}$
HAIL	Weight Lower Bound $w_{\min}$	0.1
HAIL	Prediction Horizon H	20
HAIL	Maximum Candidate Horizon $L_{\max}$	19
HAIL	Core Retention Ratio ρ	0.15
HAIL	Tail Keep Probability p_{tail}	0.05
HAIL	Tail Sampling Decay λ_{tail}	1.5

The quality evaluator takes fixed-length intervention segments as inputs and is parameterized by a multilayer perceptron with a single hidden layer. The hidden dimension is set to 256, and the evaluator outputs a scalar quality score for each intervention segment. All segments are standardized using unified statistics before training to ensure a consistent feature scale across different quality levels. Given the segment mean μ and standard deviation σ computed from the offline quality dataset, the segment normalization is formulated as

$$\tilde{s} = \frac{s - \mu}{\sigma + 10^{-3}}. \tag{7}$$

For each parameter update, 32 intervention segments are sampled from each of the low-, medium-, and high-quality levels, resulting in a training batch containing 96 samples. The different quality levels are sampled equally in each update to maintain a balanced quality distribution throughout training. The main training configurations of the quality evaluator are summarized in Table 8.

The trained segment normalization statistics and quality evaluator parameters are saved together and reused for subsequent online quality evaluation of human intervention behaviors, ensuring consistency between the input scales during offline training and online evaluation. In implementation, the normalization parameters are stored as the corresponding shift and scale values.

Table 8: Parameters of the QWBL quality evaluator

Parameter	Value
Quality Levels	3
Quality Labels	0 / 1 / 2
Batch Size per Quality Level	32
Total Batch Size	96
Hidden Dimension	256
Hidden Depth	1
Optimizer	Adam
Learning Rate	5×10^{-3}
Training Iterations	10,000
Observation Normalization	$\frac{s-\mu}{\sigma+10^{-3}}$

Table 9: Weight mapping parameters of QWBL

Hyperparameter	Value
Score Lower Bound $q_{\min}$	1.5
Initial / Minimum Upper Bound M_0	2.2
EMA Coefficient α	0.99
Upper-bound Quantile	0.95
Minimum Weight $w_{\min}$	0.1
Maximum Weight	1.0

C.2 Quality Weight Mapping Details

During online policy optimization, QWBL dynamically maintains a score upper bound M_t based on recent intervention quality scores, which is used to transform the quality evaluator outputs into behavioral learning weights. Before performing quality weight mapping, QWBL first computes the 95-th percentile of quality scores within the current intervention batch, denoted as $Q_{0.95}^{(t)}$.

The score upper bound is then updated using an exponential moving average:

$$M_t = \max[M_0, \alpha M_{t-1} + (1-\alpha) Q_{0.95}^{(t)}]. \tag{8}$$

where M_0 denotes the initial value and lower bound of the score upper bound, and α is the exponential moving average coefficient. Using a high-percentile statistic instead of the maximum score in the current batch reduces the influence of individual abnormal high-score samples on the weight scale. The exponential moving average further reduces short-term fluctuations of the score upper bound across different batches. The main parameters for online quality weight mapping are summarized in Table 9.

The final quality weights satisfy $w_i \in [0.1, 1.0]$. Through the above mapping, the relative quality scores provided by the quality evaluator are converted into continuous weights. These weights are applied to the per-sample behavioral learning loss defined in Section 4.2.

D Construction of Quality Ranking Dataset

D.1 Quality Ranking Dataset Generation

The intervention segments used for quality-level assignment are extracted from human expert intervention data collected before policy training. The quality labels do not require additional human annotation and are constructed based on observable task outcomes and performance metrics during environment interactions, which are referred to as outcome-based labels.

Table 10: Quality-level criteria in the MetaDrive environment

Quality level	Criteria
$\mathcal{D}^{(2)}$	Successful completion without collision or out-of-road events and with stable driving performance.
$\mathcal{D}^{(1)}$	Successful completion with minor instability or suboptimal driving behavior.
$\mathcal{D}^{(0)}$	Collision, out-of-road events, or task failure.

Table 11: Quality-level criteria in the Hopper-v5 environment

Quality level	Criteria
$\mathcal{D}^{(2)}$	Top 30% episode return with stable motion states maintained.
$\mathcal{D}^{(1)}$	Middle 40% episode return with basic locomotion but relatively lower stability.
$\mathcal{D}^{(0)}$	Bottom 30% episode return with shorter survival length and inability to maintain effective motion states or early termination.

The quality ranking process is only applied to intervention segments and does not participate in the construction of the reinforcement learning reward function or modify the evaluation metrics during testing. The quality labels do not represent absolute optimal behaviors in the environment but characterize the relative learning value of intervention segments under the current task objectives.

In the MetaDrive environment, the quality levels are determined based on task completion, safety events, and driving stability. The detailed criteria are summarized in Table 10.

The Hopper-v5 task does not contain explicit safety constraints. Therefore, in Hopper-v5, intervention segments are first ranked according to episode return and further classified based on survival length and motion stability. To ensure sufficient training samples for each quality level, the intervention segments are divided using a 30%-40%-30% ratio among different quality levels.

The quality-level criteria in the Hopper-v5 environment are summarized in Table 11.

Although the quality label construction differs across environments, the underlying principle remains the same: quality labels are constructed based on task-related outcomes that reflect the effectiveness of human interventions.

Based on the quality criteria, expert intervention segments are divided into three quality levels, $\mathcal{D}^{(0)}$, $\mathcal{D}^{(1)}$, and $\mathcal{D}^{(2)}$. To avoid the influence of imbalanced numbers of samples across quality levels on quality evaluator training, we randomly select 100 intervention segments from each quality level to construct a balanced dataset. The quality evaluator is then trained using samples from different quality levels.

D.2 Evaluation of Quality Evaluator on Independent Test Sets

To evaluate whether the quality evaluator in QWBL can effectively distinguish intervention segments with different quality levels, we construct independent test intervention segment sets $\mathcal{T}^{(0)}$, $\mathcal{T}^{(1)}$, and $\mathcal{T}^{(2)}$ using the same quality criteria as those used during training. Each quality level contains 50 non-overlapping intervention segments. The quality evaluator is then evaluated on these test sets.

Since QWBL focuses on the relative quality relationships among different intervention behaviors rather than absolute quality prediction, we adopt Pairwise Ranking Accuracy as the evaluation metric. This metric evaluates whether the quality evaluator can correctly identify the relative ranking relationships between intervention segments with different quality levels, rather than measuring its ability to predict absolute quality values.

Table 12: Pairwise ranking accuracy of the quality evaluator on independent test sets

Quality comparison	Pairwise ranking accuracy
$\mathcal{T}^{(2)}$ vs $\mathcal{T}^{(0)}$	96.08%
$\mathcal{T}^{(2)}$ vs $\mathcal{T}^{(1)}$	90.16%
$\mathcal{T}^{(1)}$ vs $\mathcal{T}^{(0)}$	91.68%
Average	92.64%

For any two test intervention segments τ_i and τ_j from different quality levels, if the predefined quality relationship satisfies $\tau_i \succ \tau_j$ and the quality evaluator predicts $q(\tau_i) > q(\tau_j)$, the ranking result is considered correct. The Pairwise Ranking Accuracy is defined as:

$$\text{Accuracy} = \frac{N_{\text{correct}}}{N_{\text{pair}}}, \tag{9}$$

where N_{correct} denotes the number of segment pairs correctly ranked by the quality evaluator, and N_{pair} represents the total number of evaluated segment pairs.

For any two quality-level datasets, all cross-level segment pairs are evaluated. Therefore, each comparison contains 2,500 ranking pairs, resulting in a total of 7,500 pairwise ranking samples. The results are shown in Table 12.

The results show that the quality evaluator can preserve reliable ranking performance on independent test sets. The comparison between $\mathcal{T}^{(2)}$ and $\mathcal{T}^{(0)}$ achieves higher accuracy because these two quality levels have larger quality differences. In comparison, the ranking between $\mathcal{T}^{(2)}$ and $\mathcal{T}^{(1)}$, as well as between $\mathcal{T}^{(1)}$ and $\mathcal{T}^{(0)}$, is more challenging due to smaller quality differences, while the evaluator can still distinguish their relative quality relationships.

This experiment verifies that the quality evaluator in QWBL can learn relative quality representations of intervention segments based on task-related quality criteria.

D.3 Local Input Perturbation Robustness of Quality Evaluator

To further analyze the stability of the quality evaluator, we conduct local input perturbation experiments. Specifically, we investigate whether the quality evaluator can still distinguish the relative quality relationships among intervention segments when the input intervention segments are perturbed by local state observation errors and action execution errors.

Since the objective of the quality evaluator in QWBL is to learn the relative quality relationships among intervention segments rather than evaluate absolute quality values, we analyze its stability from the perspective of ranking consistency. Specifically, this experiment is conducted on the independent test sets $\mathcal{T}^{(0)}$, $\mathcal{T}^{(1)}$, and $\mathcal{T}^{(2)}$ constructed in Section D.2. Given an original pair of intervention segments τ_i and τ_j from different quality levels satisfying $q(\tau_i) > q(\tau_j)$, the state-action pairs (s, a) within τ_i and τ_j are first normalized using the same statistics as those used during training. Then, perturbations δ_s and δ_a are added to the normalized state and action inputs, respectively. If the quality evaluator can still determine that $q(\tilde{\tau}_i) > q(\tilde{\tau}_j)$. This indicates that the quality evaluator preserves the relative quality relationship between intervention segments. Here, δ_s and δ_a represent local perturbations applied to the state and action inputs, respectively. They are generated as zero-mean Gaussian noise:

$$\delta_s = \sigma\epsilon_s, \quad \delta_a = \sigma\epsilon_a, \tag{10}$$

where $\epsilon_s, \epsilon_a \sim \mathcal{N}(0, I)$ denote random perturbations, and σ controls the perturbation magnitude. Since this experiment aims to simulate local state observation errors and action execution errors that may occur during real interactions, rather than generating new intervention segments, the perturbation magnitude is controlled to ensure that the perturbed inputs still preserve the original quality relationship of the intervention segments.

Four perturbation levels are considered in this experiment: $\sigma \in \{0, 0.05, 0.10, 0.20\}$. The case $\sigma = 0$ corresponds to the original evaluation results reported in Section D.2, while $\sigma = 0.05$,

Table 13: Local input perturbation robustness of the quality evaluator

Noise Level σ	$\mathcal{T}^{(2)}$ vs $\mathcal{T}^{(0)}$	$\mathcal{T}^{(2)}$ vs $\mathcal{T}^{(1)}$	$\mathcal{T}^{(1)}$ vs $\mathcal{T}^{(0)}$	Average
0.00	96.08%	90.16%	91.68%	92.64%
0.05	94.04%	88.24%	85.08%	89.12%
0.10	92.88%	86.36%	83.96%	87.73%
0.20	83.56%	77.32%	74.24%	78.40%

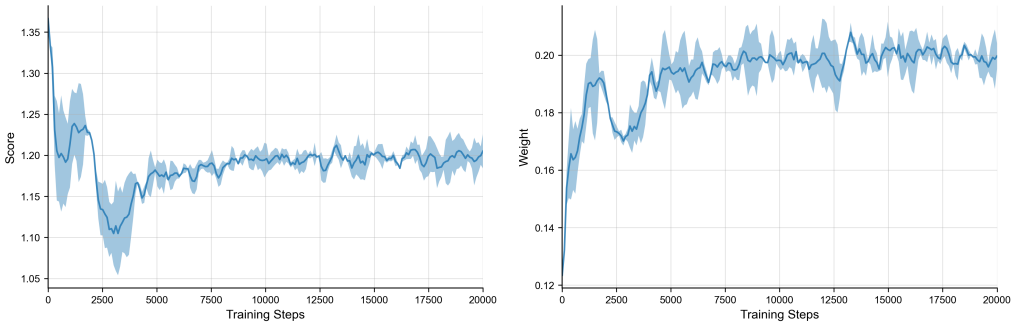


Figure 4: Changes of mean quality score and mean quality weight during QWBL training.

0.10, and 0.20 are used to further evaluate the performance variation under stronger local input perturbations.

Table 13 shows the performance of the quality evaluator under different input perturbation levels. As the perturbation magnitude increases, the Pairwise Ranking Accuracy gradually decreases, indicating that stronger input disturbances make the relative quality ranking more challenging.

When $\sigma = 0.20$, the average ranking accuracy decreases to 78.40%, since the perturbed inputs differ more substantially from the original intervention segments. However, the quality evaluator still maintains average ranking accuracy above 87.73% under moderate perturbation levels ($\sigma \leq 0.10$), indicating that the quality evaluator can maintain stable performance under typical local input perturbations.

E QWBL AND EH MECHANISM ANALYSIS

To further analyze the practical effects of the proposed modules during policy training, we visualize key internal statistics recorded throughout the training process. This section focuses on how QWBL and EH respectively adjust the contribution of intervention segments and the utilization range of future supervision information in human intervention learning.

E.1 QWBL QUALITY-WEIGHTED BEHAVIORAL LEARNING ANALYSIS

The objective of QWBL is to assign differentiated learning weights to human intervention segments, such that intervention segments with higher learning value contribute more to policy optimization. To analyze the behavior of QWBL during training, we record and visualize the changes of the mean quality score and mean quality weight of intervention segments during training, as shown in Figure 4.

As shown in Figure 4, the mean quality score exhibits significant fluctuations at the early training stage. This is because the policy has limited decision-making capability at the beginning of training, causing human interventions to occur in diverse states with different difficulties and quality characteristics. As training progresses, the mean quality score gradually enters a relatively stable changing range and continues to vary within this range. This indicates that the quality evaluator continuously adjusts the scores according to the quality differences among intervention segments, rather than mapping different intervention segments to a fixed quality value.

Table 14: EH candidate selection example

Rank	Candidate Index	Deviation Distance	Core Region	Tail Sampling	Final Selection
1	0	0.0000	✓		✓
2	1	0.5761	✓		✓
3	4	0.5771	✓		✓
4	2	0.5780			
5	3	0.5789			
6	5	0.5798			
7	6	0.5806			
8	7	0.5814			
9	8	0.5821			
10	9	0.5828			
11	10	0.5834			
12	11	0.5840			
13	12	0.5845			
14	13	0.5850			
15	14	0.5855		✓	✓
16	15	0.5895			
17	16	0.5963			
18	17	0.6166			
19	18	0.6269			

Figure 4 presents the variation of the mean quality weight. Similar to the quality score, the weight shows large fluctuations at the early training stage because the quality distribution of collected intervention segments changes significantly across different training batches. With further training, the variation of the mean quality weight decreases to a reasonable range and remains within the predefined weight interval. This demonstrates that the quality evaluator preserves its ability to distinguish intervention segments with different quality levels, and the quality weight mapping mechanism continuously adjusts the contribution of different intervention segments during policy optimization.

Overall, Figure 4 shows that QWBL does not treat all human intervention segments as having identical learning value. Instead, it assigns different learning weights according to the relative quality differences among intervention segments, allowing higher-quality intervention segments to provide larger contributions during policy updates.

E.2 Case Analysis of Effective-Horizon Retention

To further illustrate the operation of EH, we present a case study from the training process. This example demonstrates the complete procedure of EH, including future preference candidate generation, deviation-based ranking, core region retention, tail sampling, and effective preference set construction.

For one human intervention segment, the trajectory prediction model generates 19 future trajectory preference candidates. Among them, K_ρ candidates are selected as the core region. EH first computes the deviation distance between each predicted future state and the intervention state, and then sorts all candidates in ascending order according to the deviation distance. The ranking results are shown in Table 14.

In this example, EH first ranks all future trajectory preference candidates according to the deviation distance. Candidates 0, 1, and 4 exhibit the lowest deviation distances from the intervention state and are therefore selected as the core region. These candidates are directly retained for subsequent preference learning. This process shows that EH does not select future preference information according to a fixed prediction horizon, but determines the effective supervision region based on the relative state deviation of predicted future states from the intervention state.

EH further performs sparse tail sampling outside the core region. In this example, candidate 14 is not included in the core region but is additionally retained through probabilistic tail sampling.

Table 15: Effect of simulated expert quality on MetaDrive

Expert quality	Method	Success Rate ↑	Mean Reward ↑	Route Completion ↑
Suboptimal expert	PVP	0.190 ± 0.087	183.853 ± 15.474	0.565 ± 0.075
	PVP4Real	0.213 ± 0.050	209.404 ± 19.932	0.606 ± 0.057
	PPL	0.280 ± 0.035	207.809 ± 11.757	0.619 ± 0.028
	HAIL	$\mathbf{0.523 \pm 0.081}$	$\mathbf{286.233 \pm 15.415}$	$\mathbf{0.780 \pm 0.038}$
High-quality expert	PVP	0.420 ± 0.020	172.859 ± 91.208	0.711 ± 0.040
	PVP4Real	0.447 ± 0.031	218.664 ± 76.095	0.722 ± 0.041
	PPL	0.580 ± 0.174	246.456 ± 71.592	0.782 ± 0.083
	HAIL	$\mathbf{0.640 \pm 0.152}$	$\mathbf{278.934 \pm 47.099}$	$\mathbf{0.813 \pm 0.018}$

The final effective preference set generated by this intervention consists of the retained core candidates and the sampled tail candidate:

$$P^{EH} = \{p_0, p_1, p_4, p_{14}\}.$$

F Effect of Simulated Expert Quality Variation on Policy Performance

To analyze the effect of simulated expert quality variation on policy performance, we further conduct MetaDrive experiments using a higher-quality simulated expert. Specifically, compared with the suboptimal simulated expert used in the main experiments (a middle-stage SAC checkpoint with an average return of approximately 240), the high-quality simulated expert is selected from a later-stage SAC checkpoint with an average return of approximately 300. Both experts are obtained from different checkpoints of the same SAC training process, and their difference only comes from different convergence stages. Other experimental settings, including environment configuration, training budget, intervention mechanism, and evaluation protocol, remain consistent with those in the main experiments. The results are shown in Table 15.

The results show that different simulated expert quality settings affect the performance of different methods. Under the high-quality simulated expert setting, all methods obtain improved performance on the main evaluation metrics. This is because higher-quality simulated experts generate intervention segments with higher overall quality, which provide more effective supervision information for policy optimization. Meanwhile, HAIL still outperforms all compared methods in terms of Success Rate, Mean Reward, and Route Completion.

When the simulated expert quality decreases, the performance of different methods exhibits different levels of sensitivity to the quality variation. HAIL maintains relatively stable performance under both expert quality settings, indicating that modeling both intervention-level heterogeneity and temporal-level heterogeneity can reduce the sensitivity of policy learning to expert quality changes. This experiment also verifies that HAIL does not rely on a fixed-quality simulated expert.

G MetaDrive Safety Evaluation

Due to the safety-critical characteristics of the MetaDrive autonomous driving task, we further report safety-related metrics to analyze whether different human-in-the-loop reinforcement learning methods maintain safe behaviors while improving policy performance.

This evaluation is conducted under the same experimental settings as the main experiments. Two safety metrics, Crash Rate and Out-of-road Rate, are adopted for evaluation. Crash Rate represents the proportion of episodes involving collision events and measures the collision risk during driving. Out-of-road Rate represents the proportion of episodes in which the vehicle leaves the valid road area and measures the road-following capability of the learned policy. Lower values of both metrics indicate better safety performance. Since the Hopper-v5 environment does not include safety-related evaluation metrics, this evaluation is only conducted in the MetaDrive environment. The results are shown in Table 16.

Table 16: Safety evaluation on MetaDrive.

Method	Crash Rate ↓	Out-of-road Rate ↓
PVP	0.147 ± 0.081	0.660 ± 0.069
PVP4Real	0.100 ± 0.072	0.627 ± 0.081
PPL	0.180 ± 0.035	0.620 ± 0.060
HAIL	$\mathbf{0.060 \pm 0.053}$	$\mathbf{0.347 \pm 0.081}$

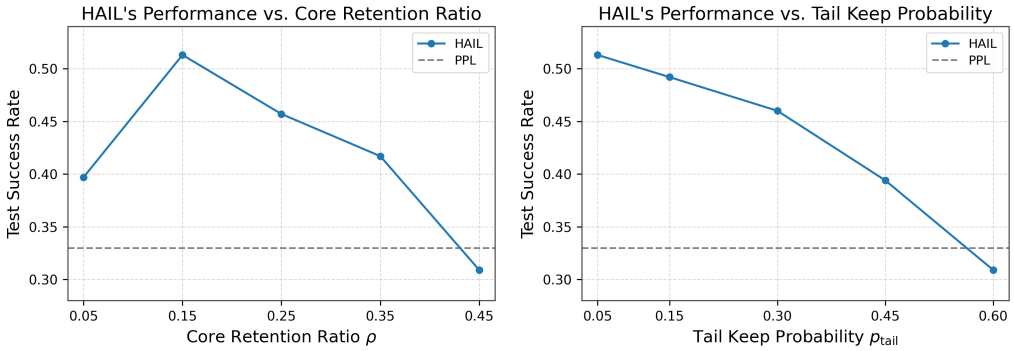


Figure 5: Sensitivity analysis of EH-related parameters in MetaDrive. The left and right figures correspond to the core retention ratio ρ and tail keep probability p_{tail}, respectively. The dashed line denotes the PPL baseline.

The results show that HAIL achieves lower safety-related metric values compared with PVP, PVP4Real, and PPL. Specifically, HAIL obtains a Crash Rate of 0.060 and an Out-of-road Rate of 0.347, indicating fewer collision events and fewer episodes leaving the valid road area under the same evaluation setting.

H Sensitivity Analysis

To analyze the effect of key parameters on the performance of HAIL, we investigate the core retention ratio ρ and tail keep probability p_{tail} in EH while keeping other experimental settings unchanged. The performance of PPL is used as the reference baseline.

As shown in Figure 5, the performance improves when the core retention ratio ρ increases from 0.05 to 0.15, while further increasing ρ leads to performance degradation. This indicates that an excessively small core region may fail to retain sufficient effective preference information, whereas an excessively large core region may introduce more preference information with low learning value. Therefore, EH requires a balance between supervision effectiveness and preference coverage.

Figure 5 illustrates the effect of the tail keep probability p_{tail}. The performance shows an overall decreasing trend as p_{tail} increases. This suggests that although retaining non-core candidates can improve preference coverage, excessive retention may introduce more preference information with low learning value into policy optimization.

Overall, HAIL maintains better performance than the PPL baseline over a relatively wide range of parameter settings.