SURF: Semi-supervised Reward Learning with Data augmentation for Feedback-Efficient Preference-based Reinforcement Learning

Jongjin Park, Younggyo Seo, Jinwoo Shin, Honglak, Pieter Abbeel, Kimin Lee
ICLR 2022


SURF: SEMI-SUPERVISED REWARD LEARNING WITH DATA AUGMENTATION FOR FEEDBACK-EFFICIENT PREFERENCE-BASED REINFORCEMENT LEARNING

Jongjin Park[1] Younggyo Seo[1] Jinwoo Shin[1] Honglak Lee[2,4] Pieter Abbeel[3] Kimin Lee[3]
[1]KAIST [2]University of Michigan [3]UC Berkeley [4]LG AI Research

2025. 03. 20

Learning Agents 강화학습 논문 리뷰 스터디

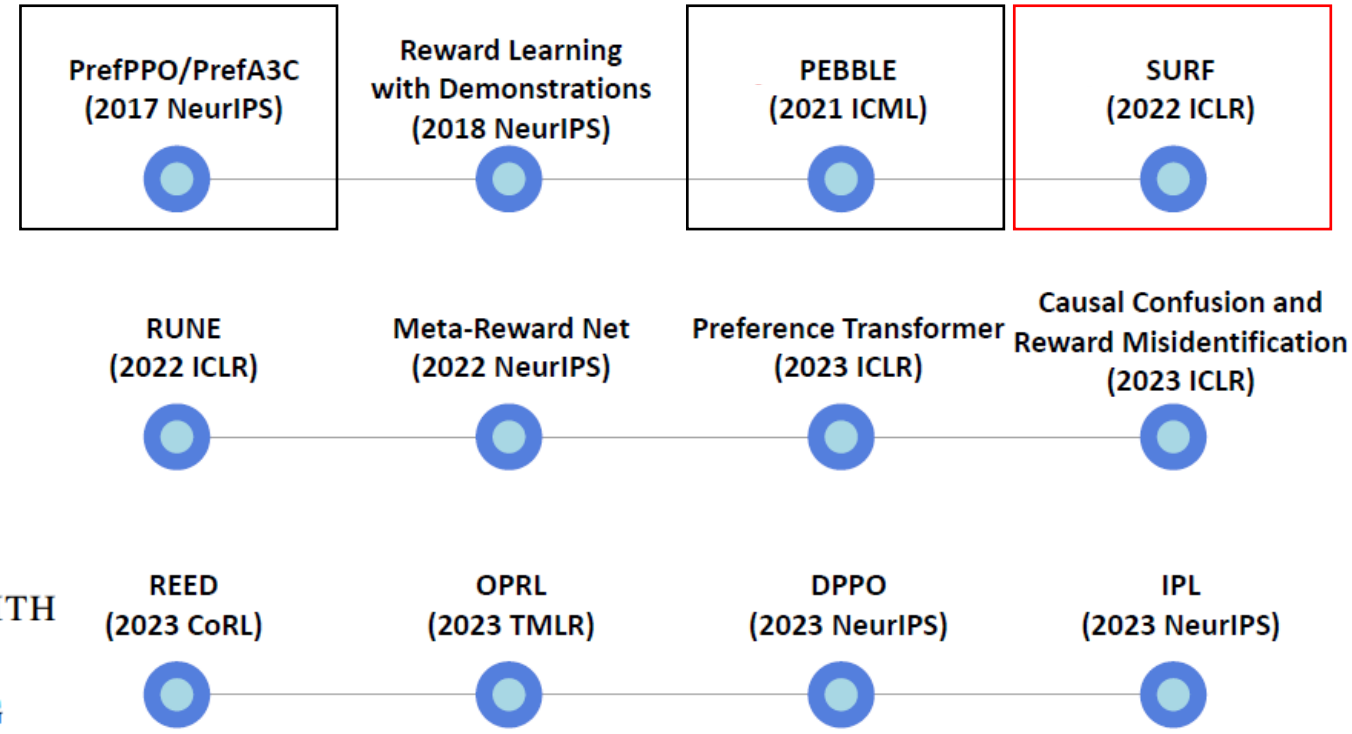
Minkyoung Kim

Agenda

- Introduction
- Related work
- Method
- Experiment

- introduction of PbRL
- Reward Ensemble and Sampling
- on-policy Algorithm (PPO)

- unsupervised Pre-training for Exploration
- off-policy Algorithm (SAC)
- Relabeling Replay Buffer for Stable Learning



SURF: SEMI-SUPERVISED REWARD LEARNING WITH DATA AUGMENTATION FOR FEEDBACK-EFFICIENT PREFERENCE-BASED REINFORCEMENT LEARNING

Jongjin Park[1] Younggyo Seo[1] Jinwoo Shin[1] Honglak Lee[2,4] Pieter Abbeel[3] Kimin Lee[3]
[1]KAIST [2]University of Michigan [3]UC Berkeley [4]LG AI Research

PEBBLE: Feedback-Efficient Interactive Reinforcement Learning via Relabeling Experience and Unsupervised Pre-training

Kimin Lee [*1] Laura Smith [*1] Pieter Abbeel [1]

Introduction

- RL have several issues in reward engineering.
  - designing a suitable reward function requires more human effort as the tasks become more complex.
    - if hand-engineered reward does not fully specify the desired task
    - there are various domain, where a single ground-truth function does not exist
    - personalization is required by modeling different reward functions based on the user's preference

→ Preference-based RL provides an attractive to alternative to avoid reward engineering

- human teacher can guide the agent to perform novel behaviors and mitigate the effects of reward exploitation.

Introduction

- Existing preference-based approaches often suffer from expensive labeling cost.
  - This makes it hard to apply preference-based RL to various applications.

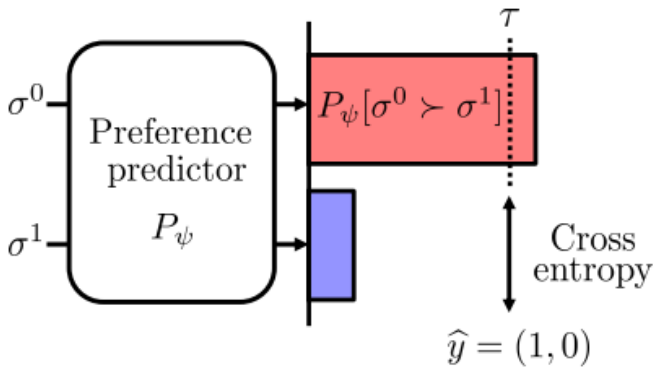
- In computer vision
  - the label-efficiency problem has been successfully addressed through semi-supervised learning approaches using leveraging unlabeled dataset.
  - Data augmentations improve the performance of supervised learning method. (augmentation-invariant representations)

- **SURF**
  - novel combination of **semi-supervised learning and the proposed data augmentation**,
  - has not been considered or evaluated in the context in the preference-based RL.

Introduction

- **SURF**: **S**emi-s**U**pervised **R**eward learning with Data augmentation for **F**eedback-efficient Preference-based Reinforcement Learning

- **Pseudo-labeling**
  - leverage unlabeled data by utilizing the artificial labels generated by learned preference predictor, which makes the reward function produce a confident prediction

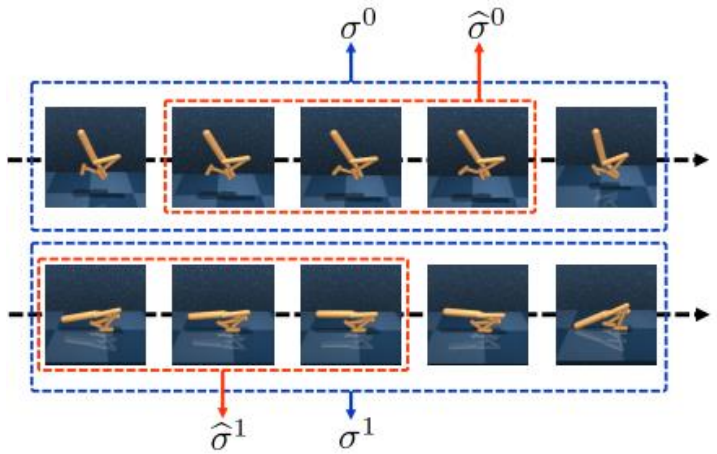


(a) Pseudo-labeling

Introduction

- **SURF**: **S**emi-s**U**pervised **R**eward learning with Data augmentation for **F**eedback-efficient Preference-based Reinforcement Learning

- **Temporal Cropping augmentation**
  - slightly shifted or resized behaviors, which are expected to have the same preferences from a teacher.
  - this data augmentation technique enhances the feedback-efficiency by enforcing consistencies



(b) Temporal cropping

Related Work

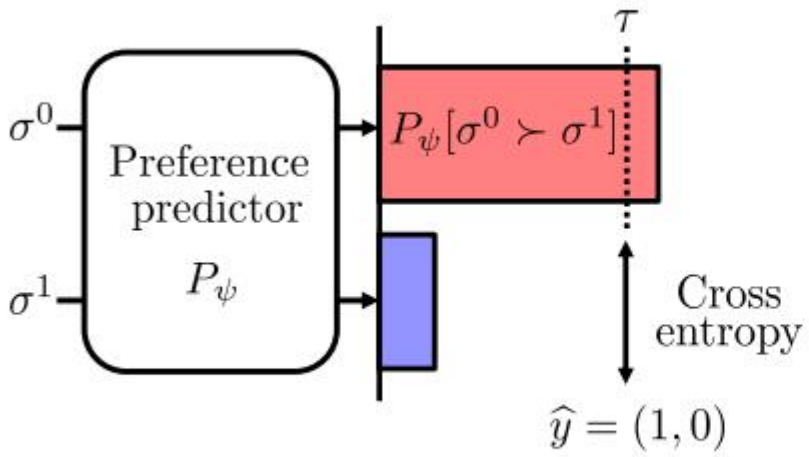
- **Preference-based RL**
  - ↔ utilize unlabeled samples for reward learning.
  - ↔ provide a novel data augmentation technique for the agent behaviors.

- **Data Augmentation for RL**
  - In RL, data augmentation has been widely investigated for improving data-efficiency to be beneficial to learn policy. (↔ for learning reward function)

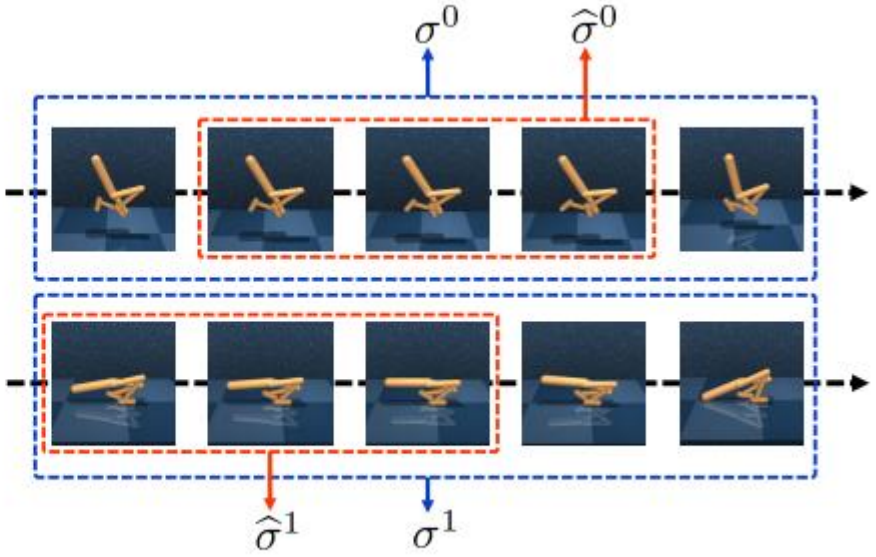
- **Semi-supervised learning**
  - SSL is leveraging unlabeled samples to improve a model's performance when the amount of labeled samples are limited.
  - **FixMatch**

SURF

- SURF can be used in conjunction with any existing preference-based RL methods.
- leverage a large number of unlabeled samples collected from environments for reward learning, by inferring pseudo-labels.
- to increase the effective number of training samples, propose a new data augmentation that temporally crops the subsequence of the agent behaviors.



(a) Pseudo-labeling



(b) Temporal cropping

SURF

- in experiment, use PEBBLE (ICML 2021).

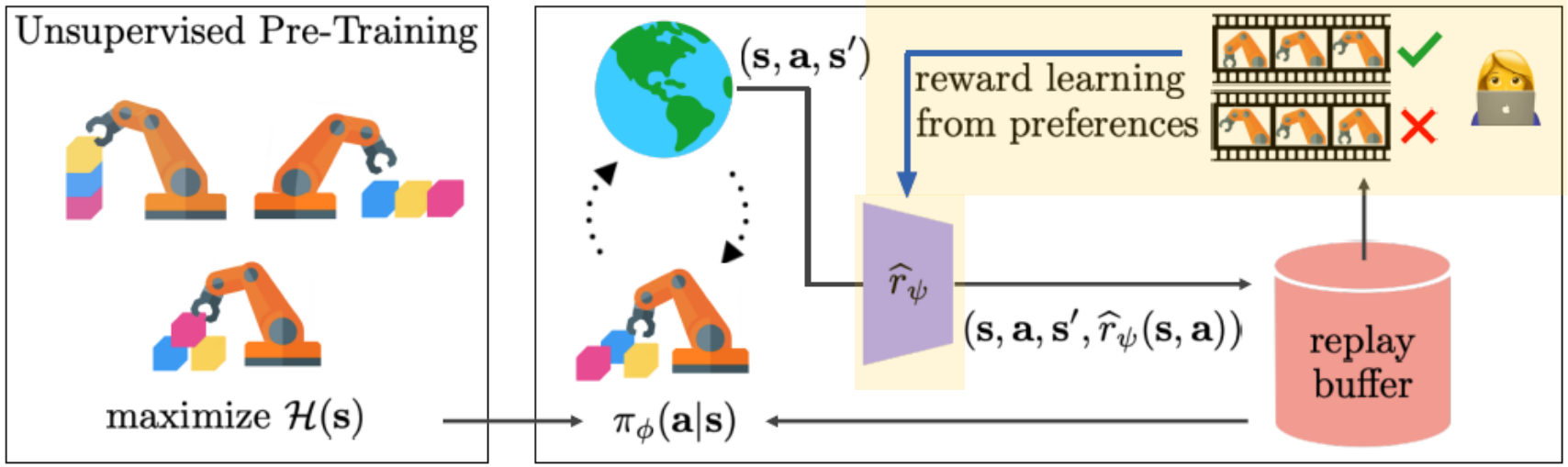


Figure 1. Illustration of our method. First, the agent engages in unsupervised pre-training during which it is encouraged to visit a diverse set of states so its queries can provide more meaningful signal than on randomly collected experience (left). Then, a teacher provides preferences between two clips of behavior, and we learn a reward model based on them. The agent is updated to maximize the expected return under the model. We also relabel all its past experiences with this model to maximize their utilization to update the policy (right).

Method

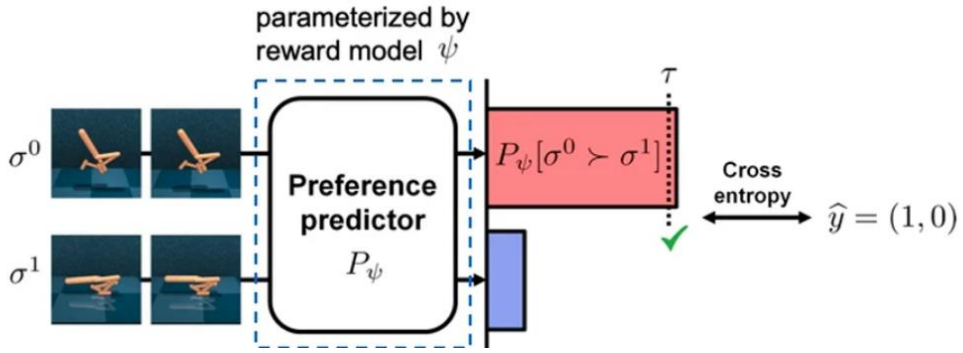
Algorithm 1 SURF

Require: Hyperparameters: unlabeled batch ratio μ, threshold parameter τ, and loss weight λ
Require: Set of collected labeled data $\mathcal{D}_l$, and unlabeled data $\mathcal{D}_u$

for each gradient step **do**
  Sample labeled batch $\{(\sigma_l^0, \sigma_l^1, y)^{(i)}\}_{i=1}^{B} \sim \mathcal{D}_l$
  Sample unlabeled batch $\{(\sigma_u^0, \sigma_u^1)^{(j)}\}_{j=1}^{\mu B} \sim \mathcal{D}_u$
  // DATA AUGMENTATION FOR LABELED DATA
  for i in $1 \ldots B$ **do**
    $(\widehat{\sigma}_l^0, \widehat{\sigma}_l^1)^{(i)} \leftarrow \texttt{TDA}((\sigma_l^0, \sigma_l^1)^{(i)})$ in Algorithm 2
  end for
  // PSEUDO-LABELING AND DATA AUGMENTATION FOR UNLABELED DATA
  for j in $1 \ldots \mu B$ **do**
    Predict pseudo-labels $\widehat{y}((\sigma_u^0, \sigma_u^1)^{(j)})$
    $(\widehat{\sigma}_u^0, \widehat{\sigma}_u^1)^{(j)} \leftarrow \texttt{TDA}((\sigma_u^0, \sigma_u^1)^{(j)})$ in Algorithm 2
  end for
  Optimize $\mathcal{L}^{\text{SSL}}$ (3) with respect to ψ
end for

Method

- **Semi-supervised reward learning(SSL)**
  - to leverage unlabeled experiences in the buffer for reward learning
  - labeled dataset $\mathcal{D}_l = \{(\sigma_l^0, \sigma_l^1, y)^{(i)}\}_{i=1}^{N_l}$
  - unlabeled dataset $\mathcal{D}_u = \{(\sigma_u^0, \sigma_u^1)^{(i)}\}_{i=1}^{N_u}$



  - use unlabeled dataset to optimize the reward model r_ψ
  - generate artificial label $\hat{y}$ by pseudo-labeling for unlabeled dataset D_u

$$\hat{y}(\sigma_u^0, \sigma_u^1) = \begin{cases} 0, & \text{if } P_\psi[\sigma_u^0 \succ \sigma_u^1] > 0.5 \\ 1, & \text{otherwise.} \end{cases}$$

    - to filter out inaccurate pseudo-labels, use unlabeled samples for training when the confidence of the predictor is higher than a pre-defined threshold τ

Method

- **Semi-supervised reward learning(SSL)**
  - reward model r_ψ loss

$$\mathcal{L}^{\mathtt{SSL}} = \mathop{\mathbb{E}}_{\substack{(\sigma_l^0,\sigma_l^1,y)\sim\mathcal{D}_l,\\(\sigma_u^0,\sigma_u^1)\sim\mathcal{D}_u}} \Big[\mathcal{L}^{\mathtt{Reward}}(\sigma_l^0,\sigma_l^1,y) + \lambda\cdot\mathcal{L}^{\mathtt{Reward}}(\sigma_u^0,\sigma_u^1,\widehat{y})\cdot\mathbb{1}\left(P_\psi[\sigma_u^{k^*}\succ\sigma_u^{1-k^*}]>\tau\right)\Big], \quad (3)$$

  - $k^* = \arg\max_{j\in\{0,1\}}\widehat{y}(j)$
    - is an index of the preferred segment from the pseudo-label
  - λ = hyperparameter that balances the losses (use 1)
  - τ = confidence threshold

Method

Algorithm 1 SURF

Require: Hyperparameters: unlabeled batch ratio μ, threshold parameter τ, and loss weight λ
Require: Set of collected labeled data $\mathcal{D}_l$, and unlabeled data $\mathcal{D}_u$

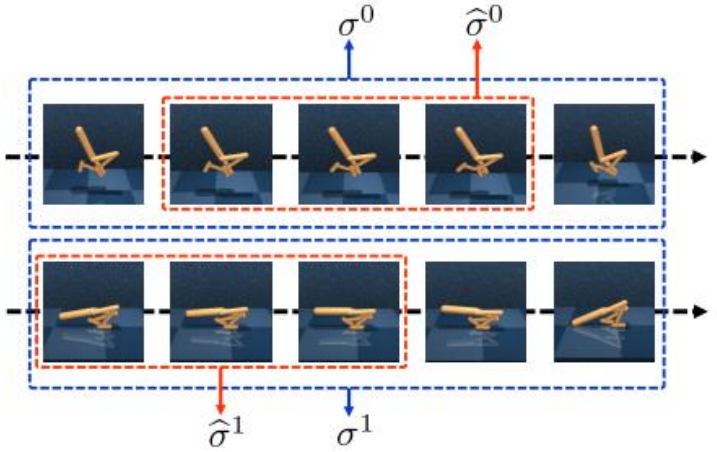
for each gradient step **do**
  Sample labeled batch $\{(\sigma_l^0, \sigma_l^1, y)^{(i)}\}_{i=1}^{B} \sim \mathcal{D}_l$
  Sample unlabeled batch $\{(\sigma_u^0, \sigma_u^1)^{(j)}\}_{j=1}^{\mu B} \sim \mathcal{D}_u$
  // DATA AUGMENTATION FOR LABELED DATA
  for i in $1 \ldots B$ **do**
    $(\widehat{\sigma}_l^0, \widehat{\sigma}_l^1)^{(i)} \leftarrow \texttt{TDA}((\sigma_l^0, \sigma_l^1)^{(i)})$ in Algorithm 2
  end for
  // PSEUDO-LABELING AND DATA AUGMENTATION FOR UNLABELED DATA
  for j in $1 \ldots \mu B$ **do**
    Predict pseudo-labels $\widehat{y}((\sigma_u^0, \sigma_u^1)^{(j)})$
    $(\widehat{\sigma}_u^0, \widehat{\sigma}_u^1)^{(j)} \leftarrow \texttt{TDA}((\sigma_u^0, \sigma_u^1)^{(j)})$ in Algorithm 2
  end for
  Optimize $\mathcal{L}^{\text{SSL}}$ (3) with respect to ψ
end for

Method

- **Temporal data augmentation; Temporal cropping**
  - given two segments and preference (σ^0, σ^1, y), generate augmented segment $(\hat{\sigma}^0, \hat{\sigma}^1, y)$ by cropping the subsequence from each segment.
  - for a given air of behavior clips, the human teacher may keep their relative preferences for slightly shifted or resized version of them. (consistency regularization, in SSL)

Algorithm 2 TDA: Temporal data augmentation for reward learning

Require: Minimum and maximum length $H_{\min}$ and $H_{\max}$, respectively, for cropping
Require: Pair of segments (σ^0, σ^1) with length H
$\sigma^0 = \{(\mathbf{s}_0^0, \mathbf{a}_0^0), ..., (\mathbf{s}_{H-1}^0, \mathbf{a}_{H-1}^0)\}$
$\sigma^1 = \{(\mathbf{s}_0^1, \mathbf{a}_0^1), ..., (\mathbf{s}_{H-1}^1, \mathbf{a}_{H-1}^1)\}$
Sample H' from a range of $[H_{\min}, H_{\max}]$
Sample k_0, k_1 from a range of $[0, H - H']$
// RANDOMLY CROP A SEQUENCE WITH LENGTH H'
$\hat{\sigma}^0 \leftarrow \{(\mathbf{s}_{k_0}^0, \mathbf{a}_{k_0}^0), ..., (\mathbf{s}_{k_0+H'-1}^0, \mathbf{a}_{k_0+H'-1}^0)\}$
$\hat{\sigma}^1 \leftarrow \{(\mathbf{s}_{k_1}^1, \mathbf{a}_{k_1}^1), ..., (\mathbf{s}_{k_1+H'-1}^1, \mathbf{a}_{k_1+H'-1}^1)\}$
Return $(\hat{\sigma}^0, \hat{\sigma}^1)$



(b) Temporal cropping

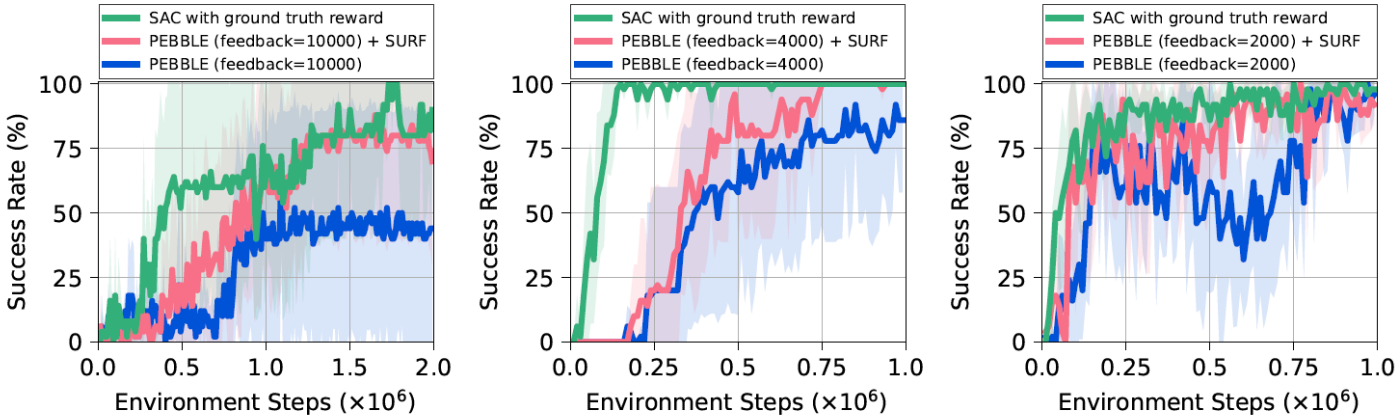
Experiments

- Setup
  - consider a scripted teacher that provides preferences between two trajectory segments to the agent according to the underlying reward function (ground truth reward of the environment).
  - PEBBLE(2021) used for backbone algorithm.
    - use SAC algorithm to learn policy.
    - for query selection strategy, use disagreement-based sampling scheme, which select queries with high uncertainty (ensemble disagreement)
  - sample unlabeled samples as 10 times of labeled ones by uniform sample scheme.
  - unlabeled batch ratio μ = 4.
  - threshold parameter τ = 0.999 for window open, sweep into, cheetah, and use 0.99 for others.

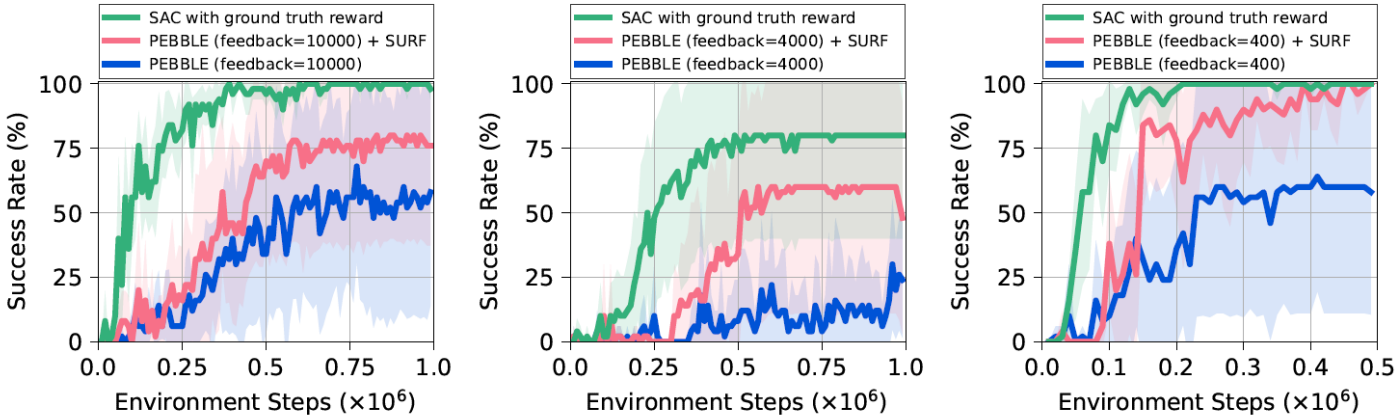
  - Goal is not to outperform SAC, but rather to perform closely using as few preference queries as possible.

Experiments

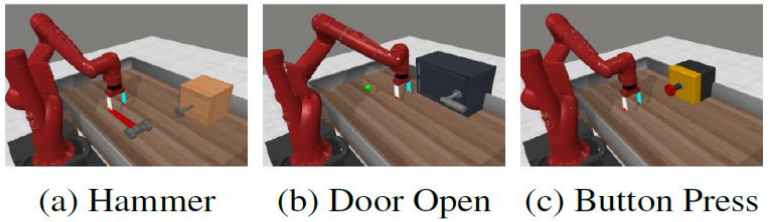
1. How does SURF improve the existing preference-based RL method in terms of feedback efficiency?



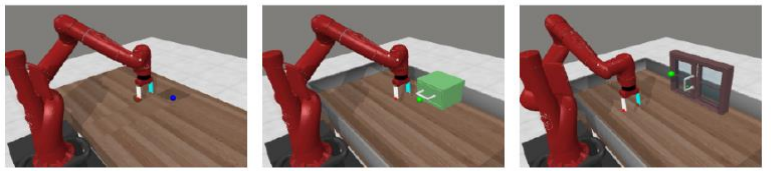
(a) Hammer (b) Door Open (c) Button Press



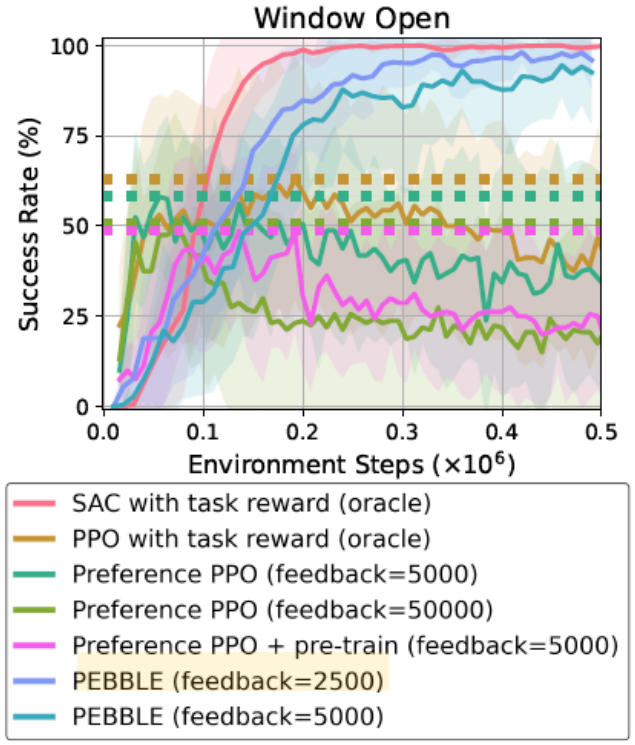
(d) Sweep Into (e) Drawer Open (f) Window Open



(a) Hammer (b) Door Open (c) Button Press



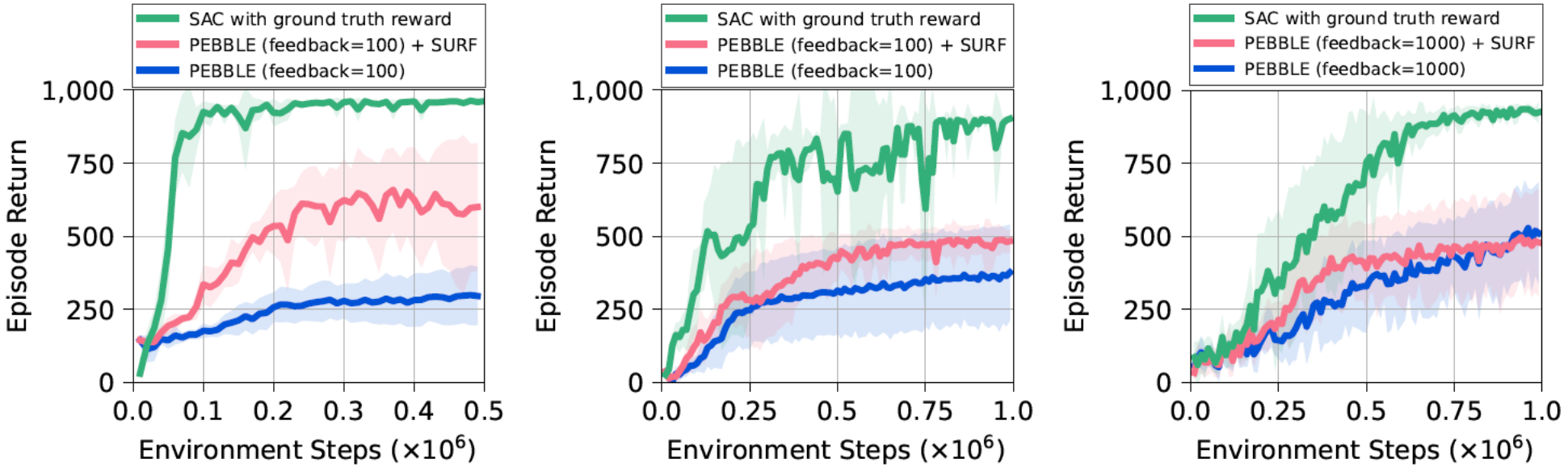
(d) Sweep Into (e) Drawer Open (f) Window Open



PEBBLE need 2500 queries,
6 times more queries than SURF

Experiments

1. How does SURF improve the existing preference-based RL method in terms of feedback efficiency?



(a) Walker (b) Cheetah (c) Quadruped

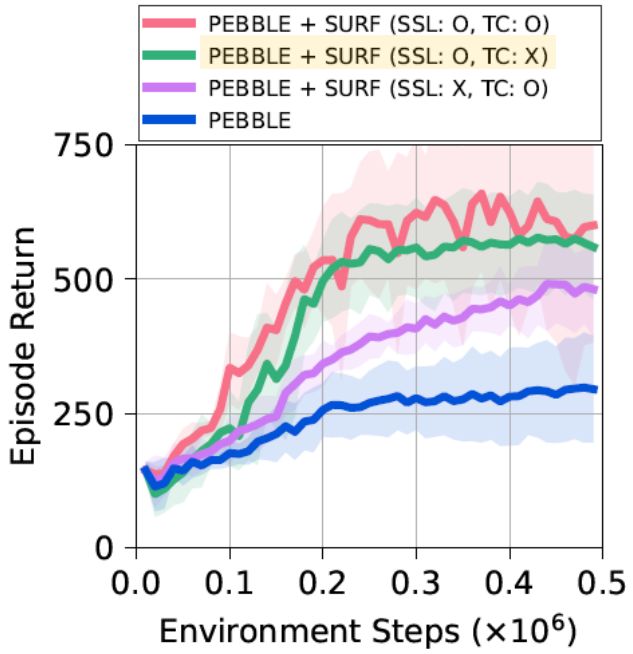
Walker

Cheetah

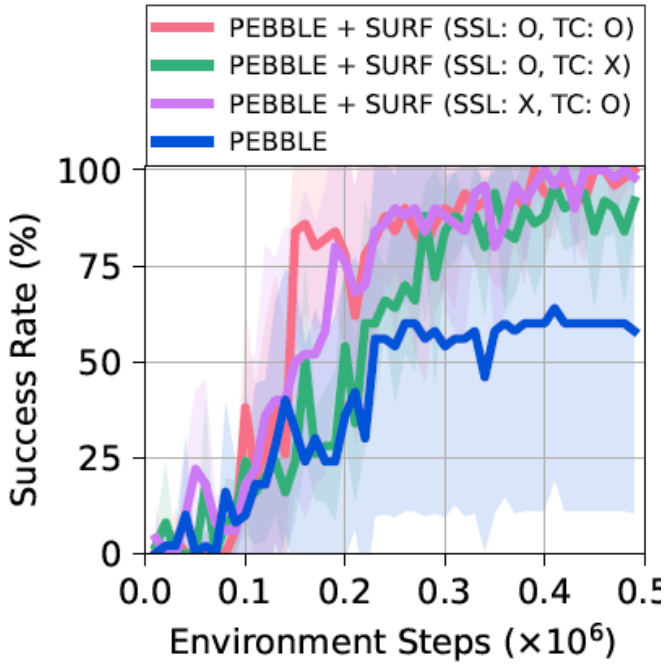
Quadruped

Experiments

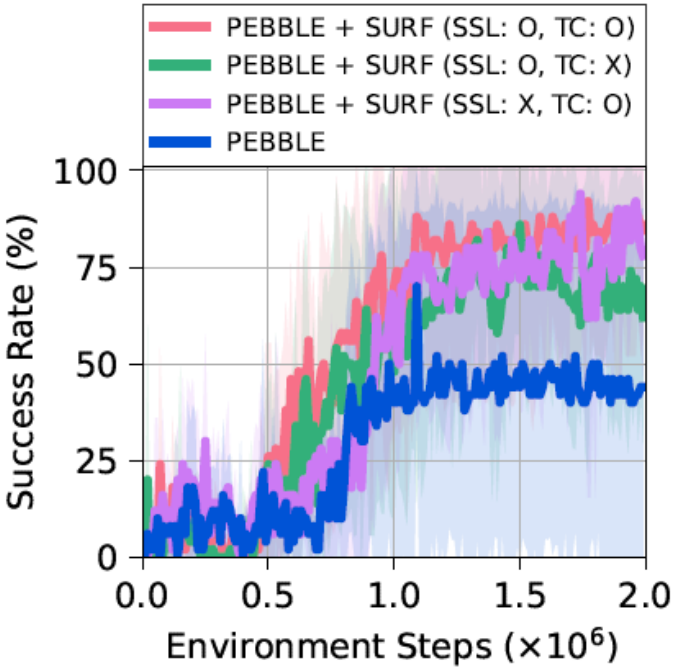
2. What is the contribution of each of the proposed components in SURF?



(a) Contributions of each component on walker-walk



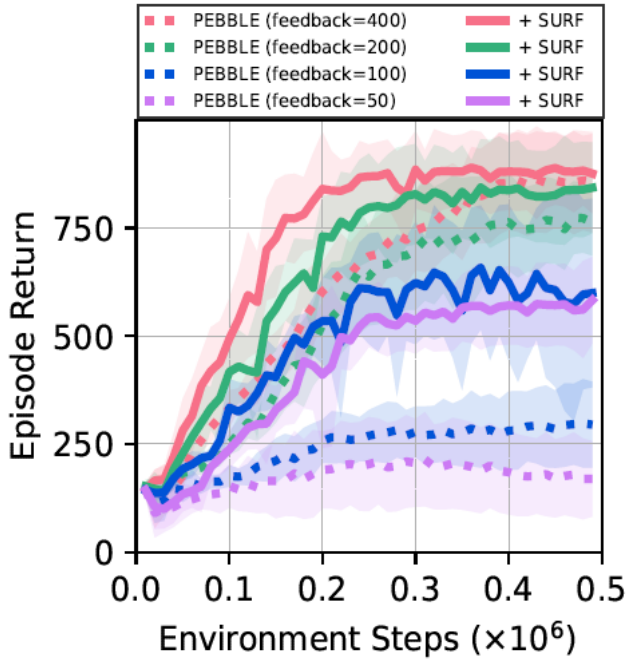
(a) Ablation study on Window Open



(b) Ablation study on Hammer

Experiments

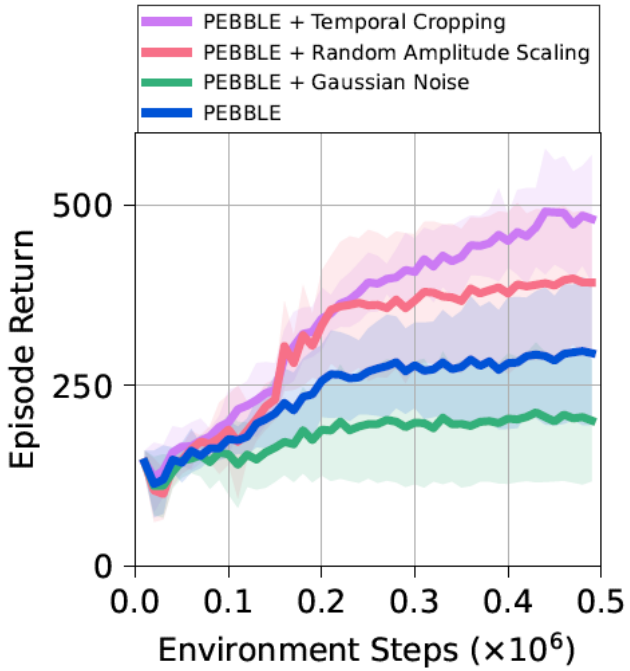
3. How does the number of queries affect the performance of SURF?
   - varying number of queries N ∈ {50, 100, 200, 400}
   - more significant in the extreme label-scarce scenario N ∈ {50, 100}



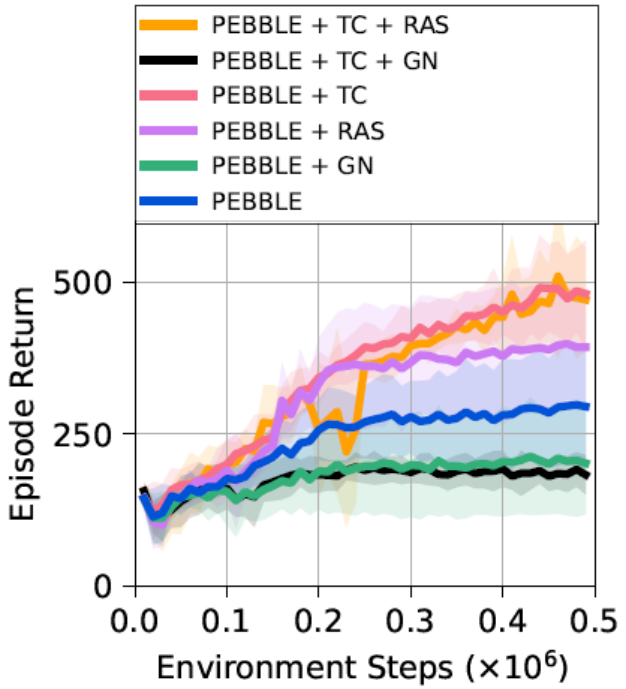
(b) Query size

Experiments

4. Is temporal cropping better than existing state-based data augmentation methods in terms of feedback efficiency? (on Walker-walk)



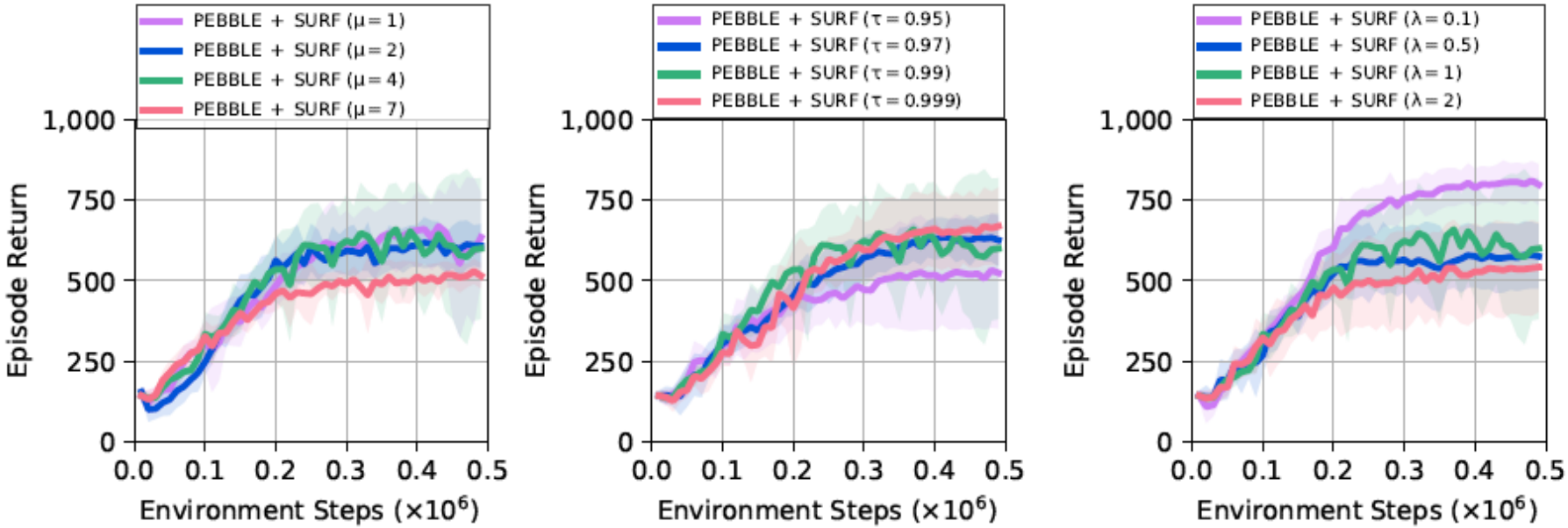
(c) Effects of data augmentation



(a) Joint usage of the augmentations

Experiments

5. Effects of hyperparameters of SURF?

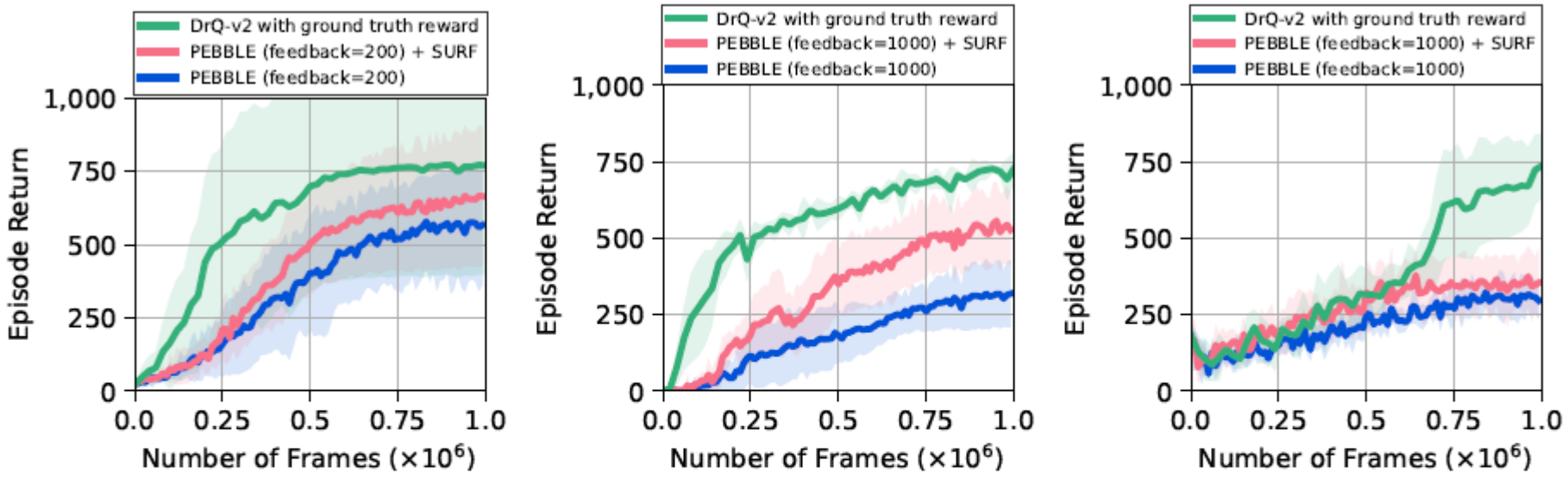


(a) Unlabeled batch ratio μ (b) Threshold parameter τ (c) Loss weight λ

Figure 5: Hyperparameter analysis on Walker-walk using 100 preference queries. The results show the mean and standard deviation averaged over five runs.

Experiments

6. Can SURF improve the performance of preference-based RL methods when we operate on high-dimensional and partially observable inputs?
   - backbone = DrQ-v2



(a) Walker (b) Cheetah (c) Quadruped

Figure 6: Learning curves on locomotion tasks with pixel-based inputs as measured on the ground truth reward. The solid line and shaded regions represent the mean and standard deviation, respectively, across five runs.

References

- [Open DMQA Seminar] RLHF-Preference-based Reinforcement Learning, https://www.youtube.com/watch?v=Vzno0oBbm6w
- https://slideslive.com/38971047/surf-semisupervised-reward-learning-with-data-augmentation-for-feedbackefficient-preferencebased-reinforcement-learning