Preference Transformer: Modeling Human Preference Using Transformer for RL

Changyeon Kim, Jongjin Park, Jinwoo Shin, Honglak, Pieter Abbeel, Kimin Lee
ICLR 2023

PREFERENCE TRANSFORMER: MODELING HUMAN PREFERENCES USING TRANSFORMERS FOR RL

Changyeon Kim[1*] **Jongjin Park**[1*] **Jinwoo Shin**[1] **Honglak Lee**[2,3] **Pieter Abbeel**[4] **Kimin Lee**[5]
[1]KAIST [2]University of Michigan [3]LG AI Research [4]UC Berkeley [5]Google Research

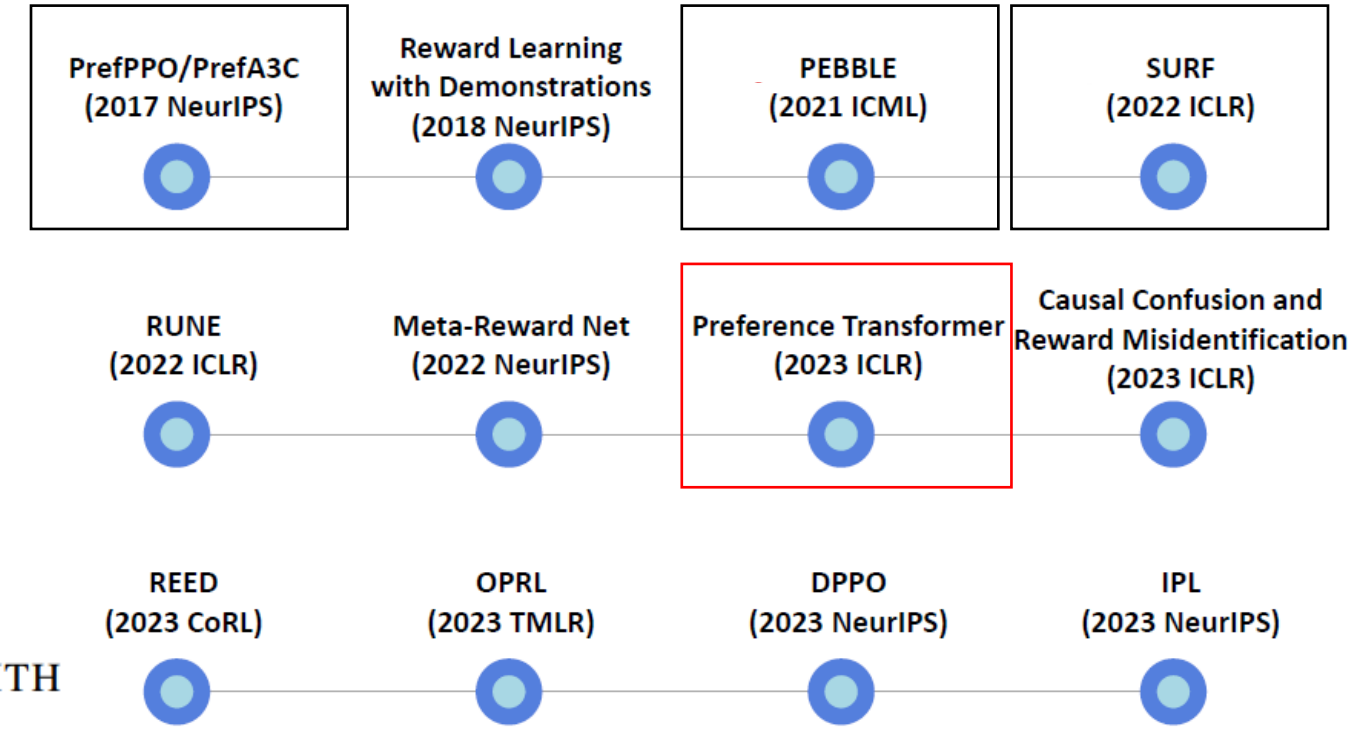
2025. 06. 12

Learning Agents 강화학습 논문 리뷰 스터디

Minkyoung Kim

Agenda

- Introduction
- Contribution
- Method
- Experiments



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

Preference Transformer: Modeling Human Preferences using Transformers for RL

Changyeon Kim[1*] **Jongjin Park**[1*] **Jinwoo Shin**[1] **Honglak Lee**[2,3] **Pieter Abbeel**[4] **Kimin Lee**[5]
[1]KAIST [2]University of Michigan [3]LG AI Research [4]UC Berkeley [5]Google Research

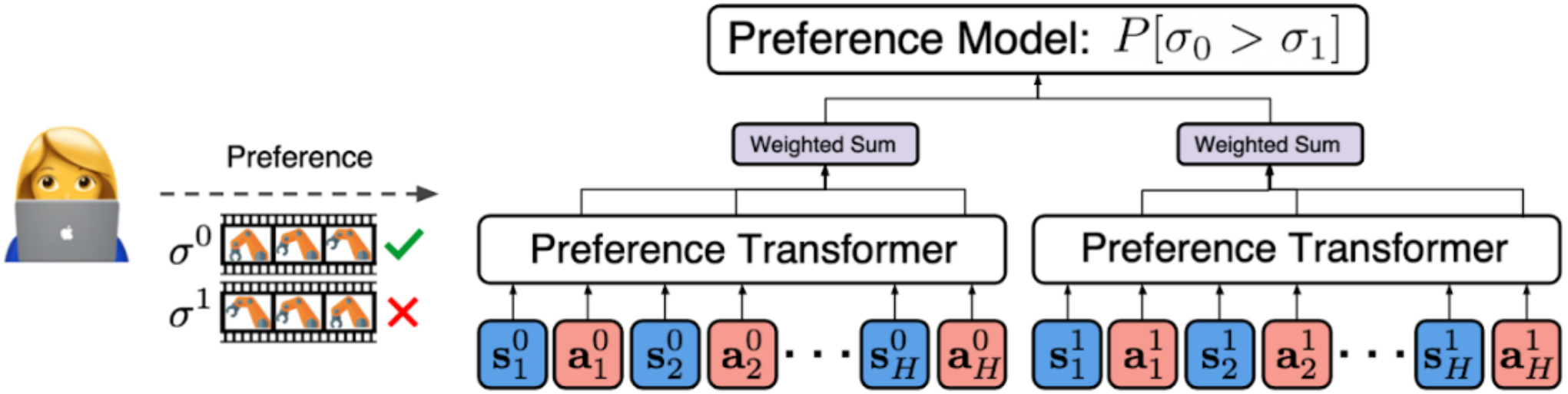
PrefPPO
- introduction of PbRL
- Reward Ensemble and Sampling
- on-policy Algorithm (PPO)

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

SURF
- semi-supervised learning
- proposed data augmentation

TL;DR

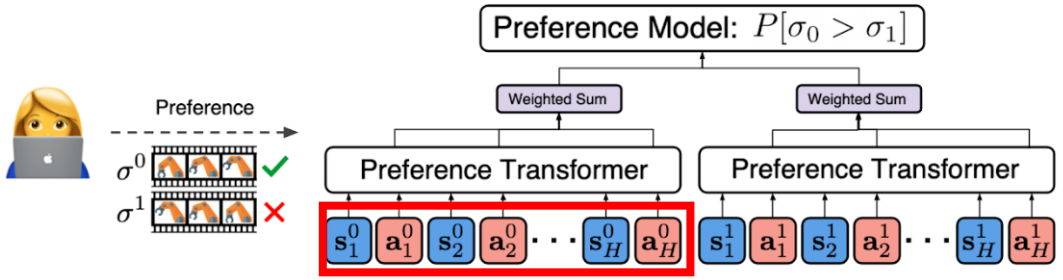
- Unlike prior approaches assuming human judgement is based on the Markovian rewards which decision equally,
- Introduce a new transformer-based architecture for preference-based RL considering non-Markovian rewards.



Introduction

- Prior preference-based RL assumes
- (1) the reward function is Markovian(i.e. depending only on the current state and action)
- (2) human evaluates the quality of a trajectory (agent's behavior) based on the sum of rewards with equal weights.

- These assumptions are flawed:
- (1) There are various tasks where rewards depend on the visited states(i.e. non-Markovian)
  - 특히, PbRL에서는 trajectory segment가 사람에게 순차적으로 제공(e.g. video clip)
- (2) Since humans are highly sensitive to remarkable moments, credit assignment within the trajectory is required.



Contribution

- **Preference Transformer**
  - based **weighted sum of non-Markovian rewards**
  - Capture the temporal dependencies in human decisions and infer critical events in the trajectory.

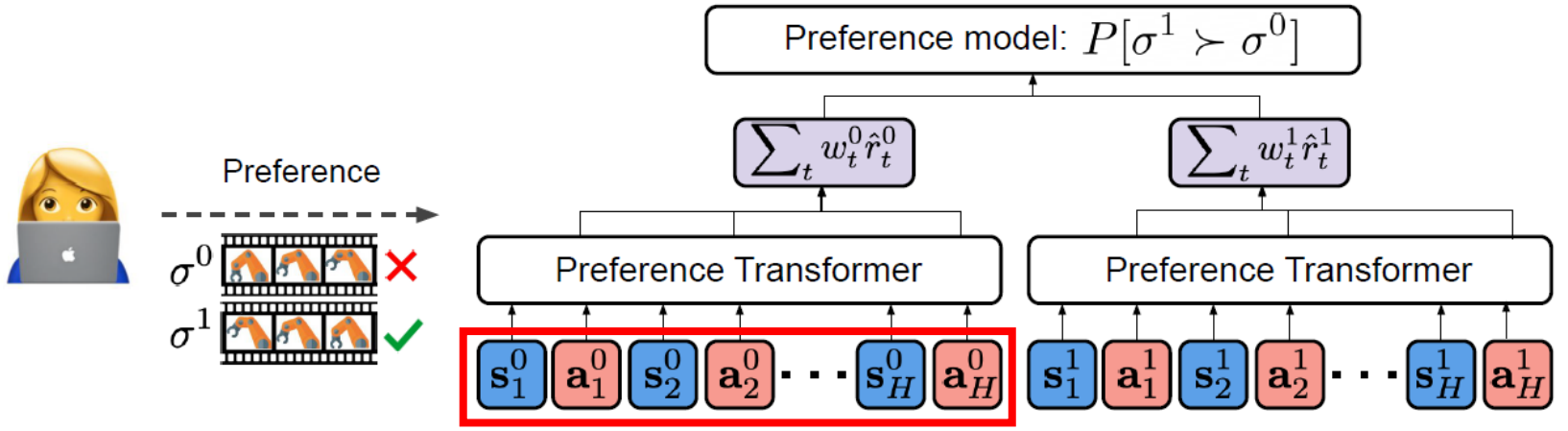


Figure 1: Illustration of our framework. Given a preference between two trajectory segments (σ^0, σ^1), Preference Transformer generates non-Markovian rewards $\hat{r}_t$ and their importance weights w_t over each segment. We then model the preference predictor based on the weighted sum of non-Markovian rewards (*i.e.*, $\sum_t w_t \hat{r}_t$), and align it with human preference.

Preference Transformer(PT)

- stacks **casual transformer** and **bidirectional self-attention layer**

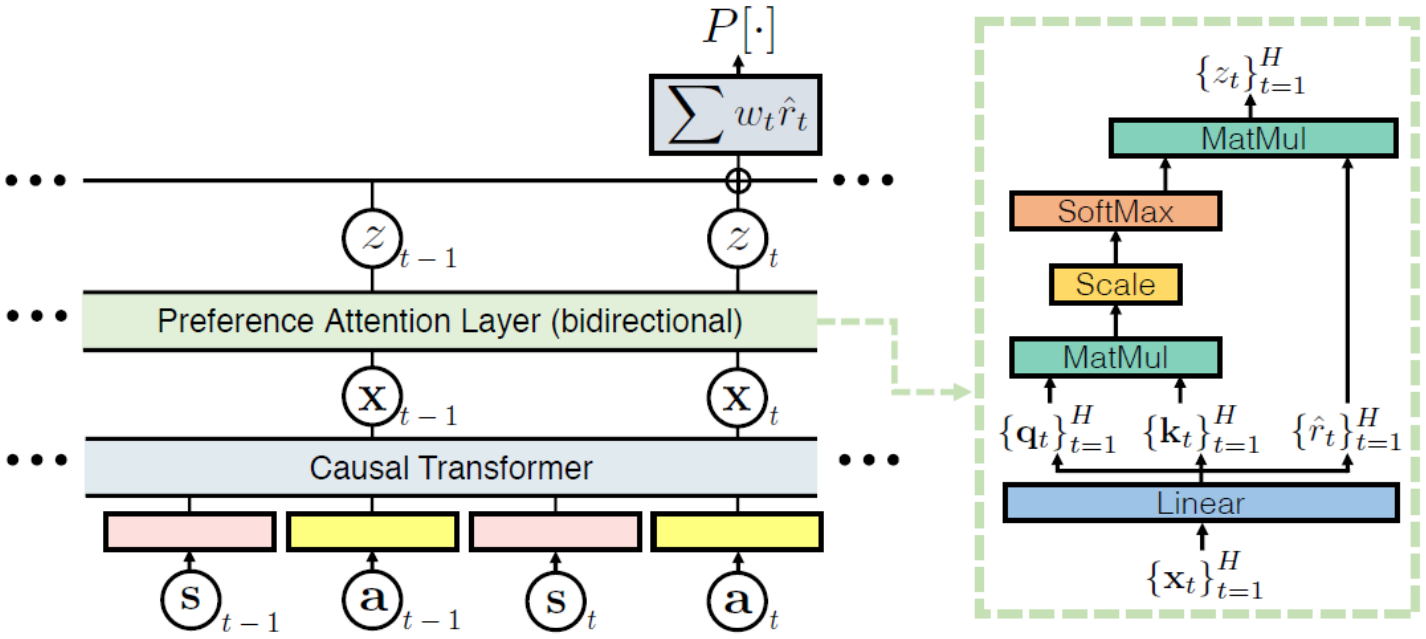
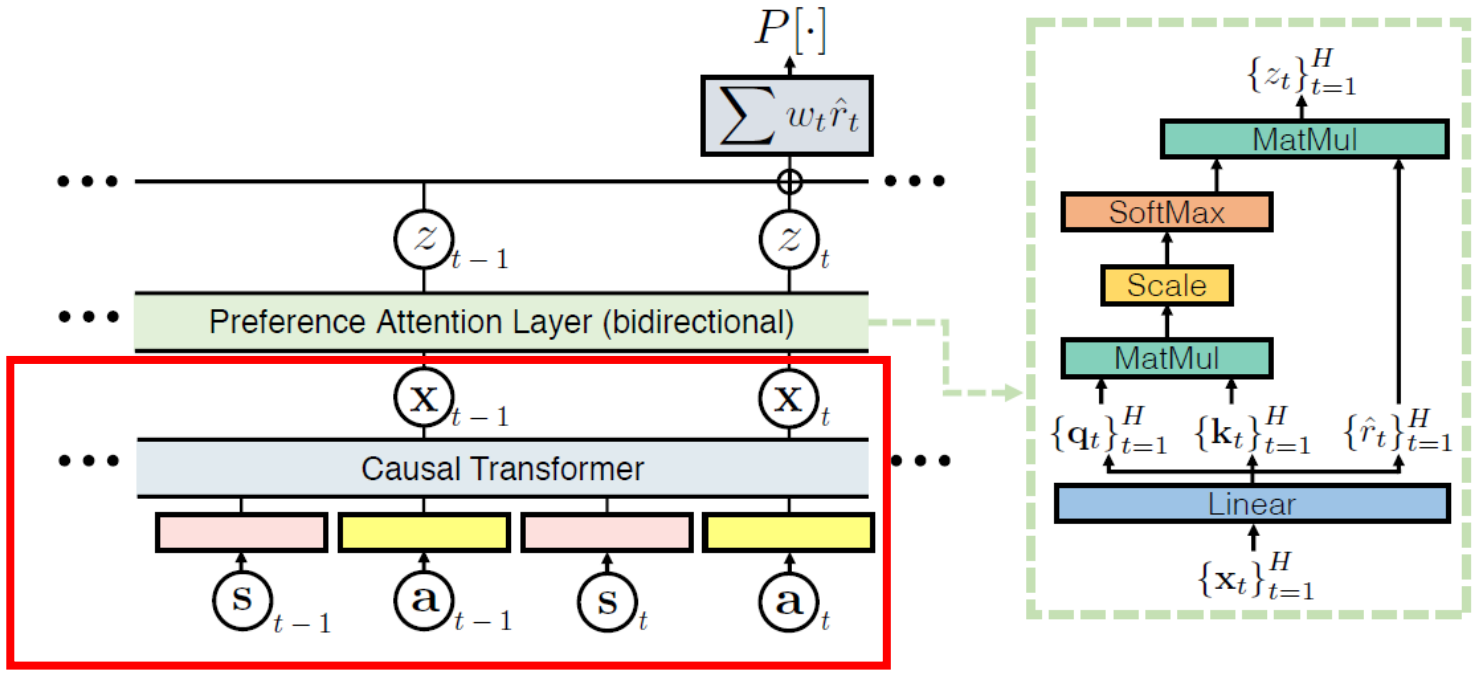


Figure 2: Overview of Preference Transformer. We first construct hidden embeddings $\{\mathbf{x}_t\}$ through the causal transformer, where each represents the context information from the initial timestep to timestep t. The preference attention layer with a bidirectional self-attention computes the non-Markovian rewards $\{\hat{r}_t\}$ and their convex combinations $\{z_t\}$ from those hidden embeddings, then we aggregate $\{z_t\}$ for modeling the weighted sum of non-Markovian rewards $\sum_t w_t \hat{r}_t$.

Preference Transformer(PT)

- **Casual Transformer: use the GPT architecture**
  - transformer architecture with casually masked self-attention
  - output embedding $\{\mathrm{X}_t\}_{t=1}^{H}$: *t*-th output depends on input embeddings up to t.



Preference Transformer(PT)

- Preference Modeling

$$\mathcal{L}^{\text{CE}}(\psi) = -\mathop{\mathbb{E}}_{(\sigma^0,\sigma^1,y)\sim\mathcal{D}}\Big[(1-y)\log P[\sigma^0 \succ \sigma^1;\psi] + y\log P[\sigma^1 \succ \sigma^0;\psi]\Big].$$

trajectory segment $\sigma = \{(\mathbf{s}_1, \mathbf{a}_1), ..., (\mathbf{s}_H, \mathbf{a}_H)\}$

length = H

이전 reward predictor는
Markovian reward를 따름

$$P[\sigma^1 \succ \sigma^0;\psi] = \frac{\exp\left(\sum_t \hat{r}(\mathbf{s}_t^1, \mathbf{a}_t^1;\psi)\right)}{\exp\left(\sum_t \hat{r}(\mathbf{s}_t^1, \mathbf{a}_t^1;\psi)\right) + \exp\left(\sum_t \hat{r}(\mathbf{s}_t^0, \mathbf{a}_t^0;\psi)\right)}.$$

full preceding sub-trajectory at time t

$$P[\sigma^1 \succ \sigma^0;\psi] = \frac{\exp\left(\sum_t w\left(\{(\mathbf{s}_i^1, \mathbf{a}_i^1)\}_{i=1}^H;\psi\right)_t \cdot \hat{r}\left(\{(\mathbf{s}_i^1, \mathbf{a}_i^1)\}_{i=1}^t;\psi\right)\right)}{\sum_{j\in\{0,1\}} \exp\left(\sum_t w\left(\{(\mathbf{s}_i^j, \mathbf{a}_i^j)\}_{i=1}^H;\psi\right)_t \cdot \hat{r}\left(\{(\mathbf{s}_i^j, \mathbf{a}_i^j)\}_{i=1}^t;\psi\right)\right)}.$$

weighted sum of non-Markovian rewards

Preference Transformer(PT)

key $\mathbf{k}_t \in \mathbb{R}^d$, query $\mathbf{q}_t \in \mathbb{R}^d$, and value $\hat{r}_t \in \mathbb{R}$,

key, query vs. value의 dimension이 다름

- **Preference attention layer**
  : to model preference predictor **using the weighted sum of the non-Markovian rewards**

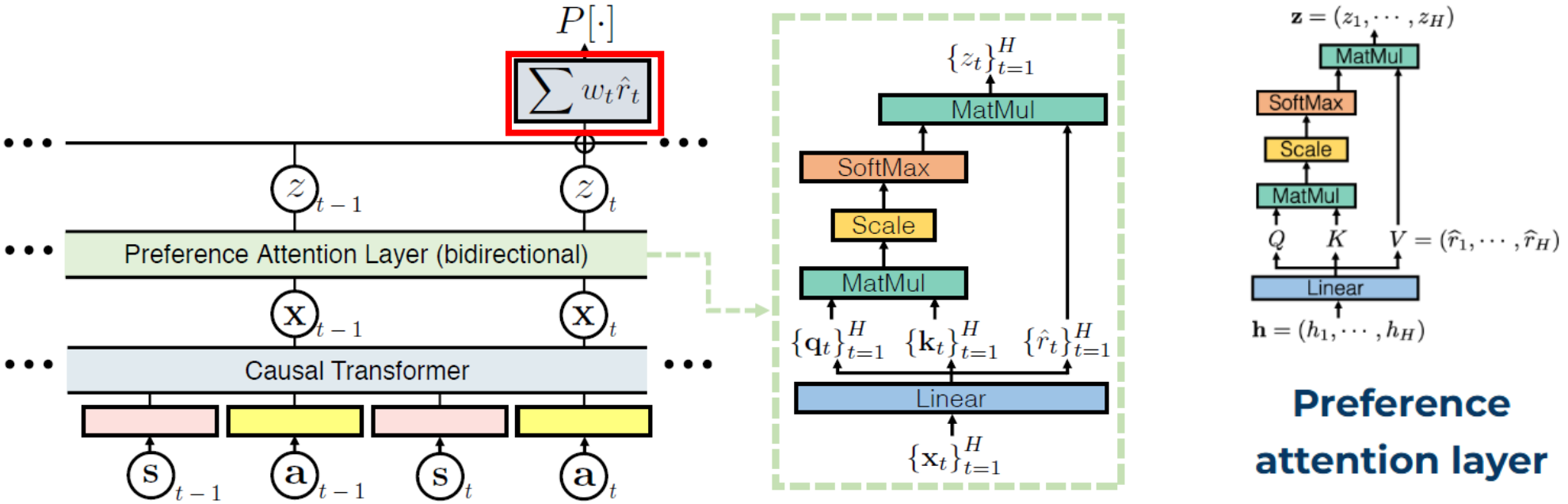
$$z_i = \sum_{t=1}^{H} \texttt{softmax}(\{\langle \mathbf{q}_i, \mathbf{k}_{t'} \rangle\}_{t'=1}^{H})_t \cdot \hat{r}_t.$$

$$\frac{1}{H}\sum_{i=1}^{H} z_i = \frac{1}{H}\sum_{i=1}^{H}\sum_{t=1}^{H} \texttt{softmax}(\{\langle \mathbf{q}_i, \mathbf{k}_{t'} \rangle\}_{t'=1}^{H})_t \cdot \hat{r}_t = \sum_{t=1}^{H} w_t \hat{r}_t$$

$\hat{r}\left(\{(\mathbf{s}_i, \mathbf{a}_i)\}_{i=1}^{t}\right)$

non-Markovian reward

$w_t = \frac{1}{H}\sum_{i=1}^{H} \texttt{softmax}(\{\langle \mathbf{q}_i, \mathbf{k}_{t'} \rangle\}_{t'=1}^{H})_t$



Preference Transformer(PT)

- Preference Modeling

$$\mathcal{L}^{\text{CE}}(\psi) = -\mathop{\mathbb{E}}_{(\sigma^0,\sigma^1,y)\sim\mathcal{D}}\Big[(1-y)\log P[\sigma^0 \succ \sigma^1;\psi] + y\log P[\sigma^1 \succ \sigma^0;\psi]\Big].$$

trajectory segment $\sigma = \{(\mathbf{s}_1, \mathbf{a}_1), ..., (\mathbf{s}_H, \mathbf{a}_H)\}$

length = H

이전 reward predictor는
Markovian reward를 따름

$$P[\sigma^1 \succ \sigma^0;\psi] = \frac{\exp\left(\sum_t \hat{r}(\mathbf{s}_t^1, \mathbf{a}_t^1;\psi)\right)}{\exp\left(\sum_t \hat{r}(\mathbf{s}_t^1, \mathbf{a}_t^1;\psi)\right) + \exp\left(\sum_t \hat{r}(\mathbf{s}_t^0, \mathbf{a}_t^0;\psi)\right)}.$$

entire trajectory segment

full preceding sub-trajectory at time t

$$P[\sigma^1 \succ \sigma^0;\psi] = \frac{\exp\left(\sum_t w\left(\{(\mathbf{s}_i^1, \mathbf{a}_i^1)\}_{i=1}^H;\psi\right)_t \cdot \hat{r}\left(\{(\mathbf{s}_i^1, \mathbf{a}_i^1)\}_{i=1}^t;\psi\right)\right)}{\sum_{j\in\{0,1\}} \exp\left(\sum_t w\left(\{(\mathbf{s}_i^j, \mathbf{a}_i^j)\}_{i=1}^H;\psi\right)_t \cdot \hat{r}\left(\{(\mathbf{s}_i^j, \mathbf{a}_i^j)\}_{i=1}^t;\psi\right)\right)}.$$

weighted sum of non-Markovian rewards

able to perform the credit assignment within segment

Experiments

- Can Transformer solve complex control task using real human preference?
- Can Preference Transformer induce a well-aligned reward and attend to critical events?
- How well does Preference Transformer perform with synthetic preference?
  (i.e., scripted teacher settings)

Setups

- Evaluate complex control tasks in **offline setting** using D4RL, Robomimic benchmark.
  - to focus on evaluating the performance of reward learning

- For reward modeling, select quires (pairs of trajectory segments) uniformly at random from offline datasets and collect preferences from real human trainers
  - AntMaze(medium: 100, large: 1000)
  - Gym-Mujoco locomotion(medium-replay: 500, medium-expert: 100)
  - human data: maximum 10 minutes
    (except gym-mujoco locomotion-medium-expert and antmaze-large, 1 ~ 2 hours)

- **train RL agents using Implicit Q-learning(IQL)**

Setups

- As baselines, standard preference modeling based on **Markovian reward(MR)** or **non-Markovian reward(NMR)**
  - MR: use MLP model

$$P[\sigma^1 \succ \sigma^0; \psi_{\text{MR}}] = \frac{\exp\left(\sum_t \hat{r}(\mathbf{s}_t^1, \mathbf{a}_t^1; \psi_{\text{MR}})\right)}{\sum_j \exp\left(\sum_t \hat{r}(\mathbf{s}_t^j, \mathbf{a}_t^j; \psi_{\text{MR}})\right)}$$

  - NMR: use LSTM-based model

$$P[\sigma^1 \succ \sigma^0; \psi_{\text{NMR}}] = \frac{\exp\left(\sum_t \hat{r}\left(\{(\mathbf{s}_i^1, \mathbf{a}_i^1)\}_{i=1}^t; \psi_{\text{NMR}}\right)\right)}{\sum_j \exp\left(\sum_t \hat{r}\left(\{(\mathbf{s}_i^j, \mathbf{a}_i^j)\}_{i=1}^t; \psi_{\text{NMR}}\right)\right)}$$

  - based on sum of reward with equal weight

Benchmark tasks with real human teachers

Table 1: Averaged normalized scores of IQL on AntMaze, Gym-Mujoco locomotion tasks, and success rate on Robosuite manipulation tasks with different reward functions. Using the same dataset of preferences from real human teachers, we train Preference Transformer (PT), MLP-based Markovian reward (MR; Christiano et al. 2017; Lee et al. 2021b), and LSTM-based non-Markovian reward (NMR; Early et al. 2022). The result shows the average and standard deviation averaged over 8 runs.

Dataset	IQL with task reward	IQL with preference learning		
		MR	NMR	PT (ours)
antmaze-medium-play-v2	73.88 ± 4.49	31.13 ± 16.96	62.88 ± 5.99	70.13 ± 3.76
antmaze-medium-diverse-v2	68.13 ± 10.15	19.38 ± 9.24	20.13 ± 17.12	65.25 ± 3.59
antmaze-large-play-v2	48.75 ± 4.35	24.25 ± 14.03	14.13 ± 3.60	42.38 ± 9.98
antmaze-large-diverse-v2	44.38 ± 4.47	5.88 ± 6.94	0.00 ± 0.00	19.63 ± 3.70
antmaze-v2 total	58.79	20.16	24.29	49.35
hopper-medium-replay-v2	83.06 ± 15.80	11.56 ± 30.27	57.88 ± 40.63	84.54 ± 4.07
hopper-medium-expert-v2	73.55 ± 41.47	57.75 ± 23.70	38.63 ± 35.58	68.96 ± 33.86
walker2d-medium-replay-v2	73.11 ± 8.07	72.07 ± 1.96	77.00 ± 3.03	71.27 ± 10.30
walker2d-medium-expert-v2	107.75 ± 2.02	108.32 ± 3.87	110.39 ± 0.93	110.13 ± 0.21
locomotion-v2 total	84.37	62.43	70.98	83.72
lift-ph	96.75 ± 1.83	84.75 ± 6.23	91.50 ± 5.42	91.75 ± 5.90
lift-mh	86.75 ± 2.82	91.00 ± 4.00	90.75 ± 5.75	86.75 ± 5.95
can-ph	74.50 ± 6.82	68.00 ± 9.13	62.00 ± 10.90	69.67 ± 5.89
can-mh	56.25 ± 8.78	47.50 ± 3.51	30.50 ± 8.73	50.50 ± 6.48
robosuite total	78.56	72.81	68.69	74.66

complex task

PT가 대부분 높은 성능을 보임

Benchmark tasks with real human teachers

- **To check whether learned rewards are indeed aligned with human preference,** generate query from agents trained with two different reward, and human evaluator decides which trajectory is better.

PT로 학습된 agent의 trajectory가 더 선호도가 높음

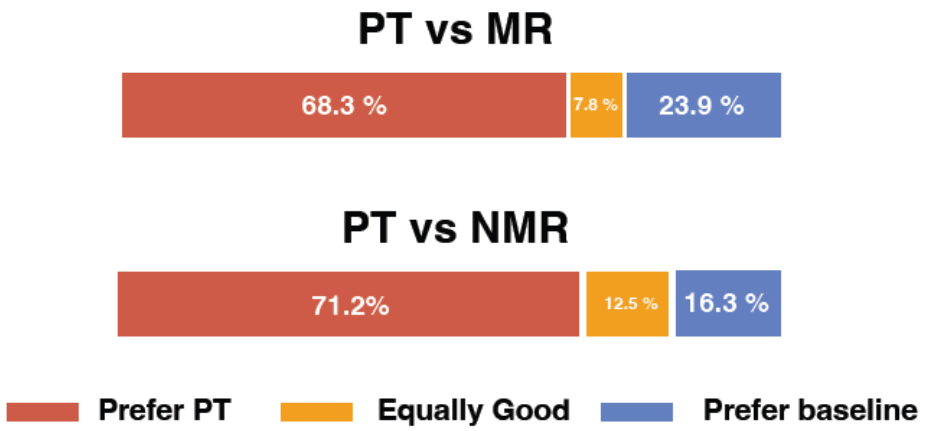


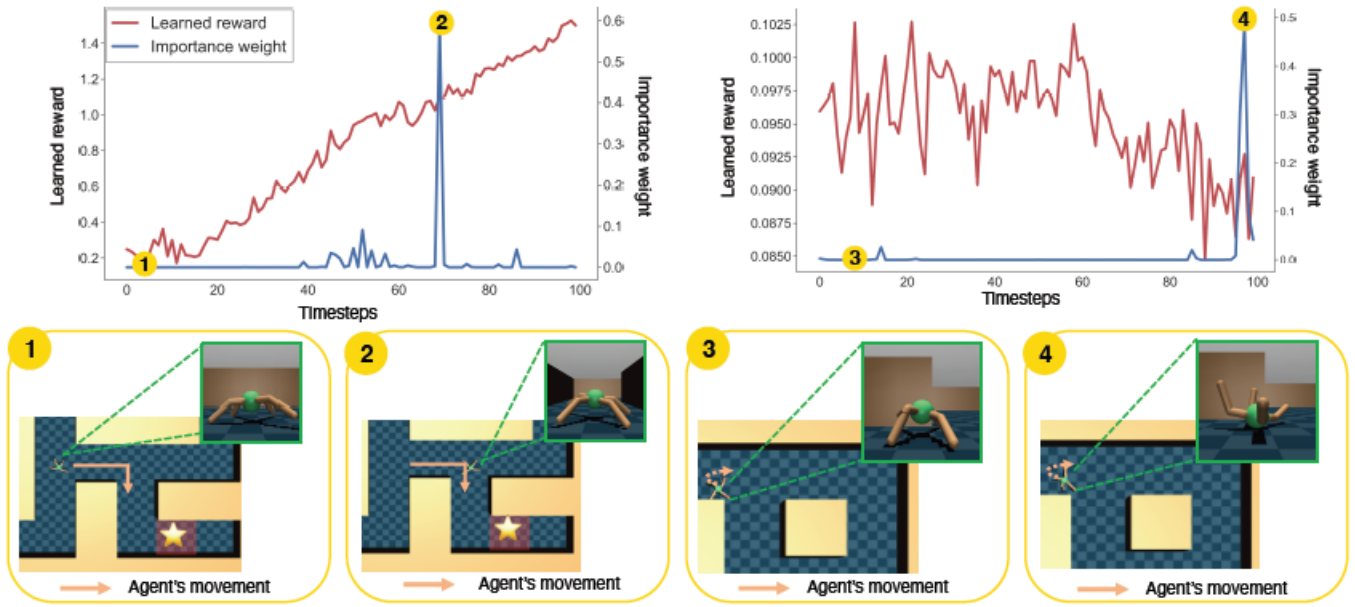
Figure 4: Averaged human evaluation results on 4 AntMaze tasks. Numbers denote the statistics of the evaluators' responses over 40 trials. PT received higher ratings compared to both MR and NMR.

Reward and weight analysis

- Can Preference Transformer induce a well-aligned reward and attend to critical events?

turning right to reach the goal

flipping



(a) Successful trajectory (b) Failure trajectory

Figure 3: Time series of learned reward function (red curve) and importance weight (blue curve) on (a) successful trajectory segment and (b) failure trajectory segment from `antmaze-large-play-v2`. For both cases, spikes in the importance weight correspond to critical events: turning right to reach the goal (point 2), or flipping (point 4). The learned reward is also well-aligned with human intent: reward increases as the agent gets close to the goal, while it decreases when agent is flipped.

https://sites.google.com/view/preference-transformer

Benchmark tasks with scripted teachers

scripted teacher: sum of the segment with **markovian** reward

scripted teacher에서 PT의 성능이 human teacher와 유사함

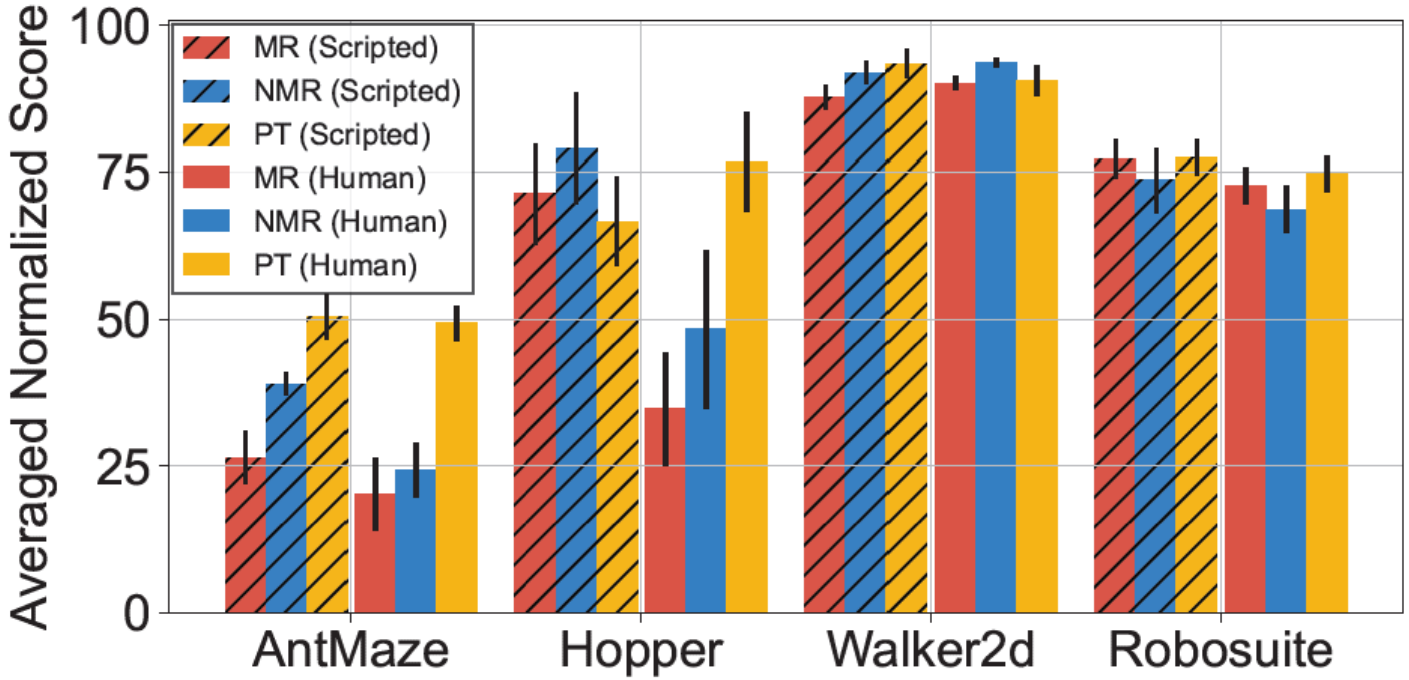
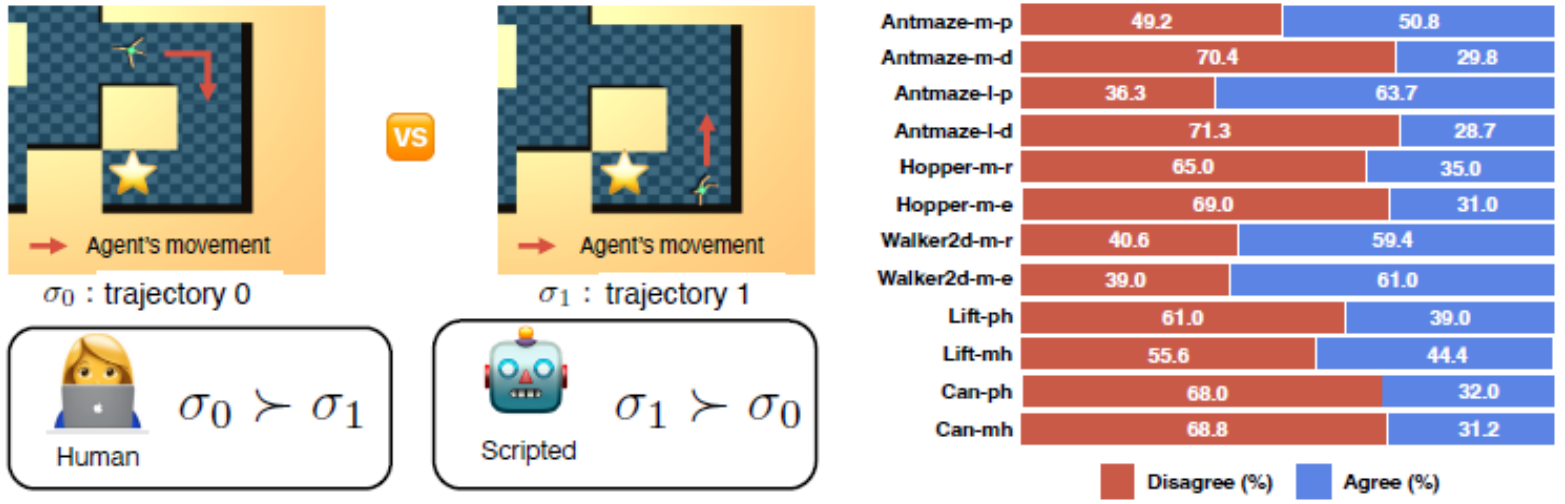


Figure 5: Averaged normalized scores of IQL with various reward functions trained from human and synthetic preferences on AntMaze, Gym-Mujoco locomotion (Hopper and Walker2d), and Robosuite robotic manipulation tasks. The result shows the mean and standard deviation averaged over 8 runs. Our method (PT) achieves strong performances on both scripted and human teachers, while the performances of baselines (MR and NMR) are significantly reduced on human teachers.

Benchmark tasks with scripted teachers

scripted teacher can not catch the context of the agent's behavior correctly



(a) Examples of trajectories

(b) Agreement rates (%)

Figure 6: Difference between the human and scripted teacher. (a) Examples of trajectories shown to the human and scripted teacher on AntMaze task. The human teacher provides the correct label by catching the context of behavior (*i.e.* direction) while the scripted teacher does not. (b) Agreement between human teachers and scripted teachers. We find that disagreement rates are quite high across all tasks, implying that evaluation on scripted teacher can generate misleading information.

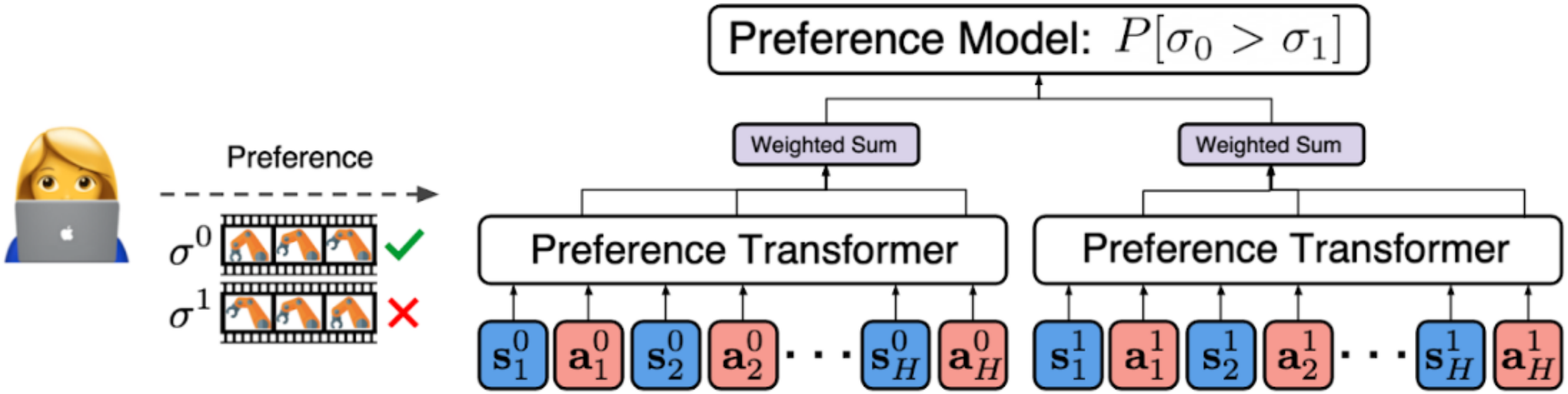
Benchmark tasks with scripted teachers

issues: scripted teacher does not model real human behavior exactly

Dataset	IQL with task reward	IQL with preference learning					
		Scripted Teacher			Human Teacher		
		MR	NMR	PT (ours)	MR	NMR	PT (ours)
antmaze-medium-play-v2	73.88 ± 4.49	64.75 ± 5.23	66.13 ± 3.36	66.13 ± 3.36	31.13 ± 16.96	62.88 ± 5.99	70.13 ± 3.76
antmaze-medium-diverse-v2	68.13 ± 10.15	4.25 ± 6.27	67.00 ± 5.90	68.13 ± 4.88	19.38 ± 9.24	20.13 ± 17.12	65.25 ± 3.59
antmaze-large-play-v2	48.75 ± 4.35	21.00 ± 14.23	10.75 ± 1.98	23.13 ± 13.10	24.25 ± 14.03	14.13 ± 3.60	42.38 ± 9.98
antmaze-large-diverse-v2	44.38 ± 4.47	15.75 ± 6.32	12.00 ± 3.59	44.50 ± 5.90	5.88 ± 6.94	0.00 ± 0.00	19.63 ± 3.70
antmaze-v2 average	58.79	26.31	38.97	50.00	20.16	24.29	49.35
hopper-medium-replay-v2	83.06 ± 15.80	62.77 ± 9.36	72.33 ± 0.01	94.19 ± 6.08	11.56 ± 30.27	57.88 ± 40.63	84.54 ± 4.07
hopper-medium-expert-v2	73.55 ± 41.47	80.00 ± 33.06	85.97 ± 37.91	39.14 ± 29.33	57.75 ± 23.70	38.63 ± 35.58	68.96 ± 33.86
walker2d-medium-replay-v2	73.11 ± 8.07	65.69 ± 8.17	73.63 ± 7.35	77.08 ± 9.84	72.07 ± 1.96	77.00 ± 3.03	71.27 ± 10.30
walker2d-medium-expert-v2	107.75 ± 2.02	109.95 ± 0.54	110.41 ± 0.82	109.99 ± 0.63	108.32 ± 3.87	110.39 ± 0.93	110.13 ± 0.21
locomotion-v2 average	84.37	79.60	85.56	80.10	62.43	70.98	83.72
lift-ph	96.75 ± 1.83	95.75 ± 2.71	80.00 ± 19.18	92.50 ± 4.24	84.75 ± 6.23	91.50 ± 5.42	91.75 ± 5.90
lift-mh	86.75 ± 2.82	92.25 ± 4.83	91.50 ± 5.42	93.00 ± 3.55	91.00 ± 4.00	90.75 ± 5.75	86.75 ± 5.95
can-ph	74.50 ± 6.82	62.25 ± 12.02	67.75 ± 6.80	69.75 ± 9.04	68.00 ± 9.13	62.00 ± 10.90	69.75 ± 5.89
can-mh	56.25 ± 8.78	59.00 ± 1.10	54.25 ± 5.57	55.25 ± 6.65	47.50 ± 3.51	30.50 ± 8.73	50.50 ± 6.48
robosuite average	78.56	77.31	73.37	77.63	72.81	68.69	74.68

Conclusion

- present a new framework for modeling human preferences based on **the weighted sum of non-Markovian rewards using transformer-based architecture.**
- experiments on offline RL benchmarks
  - observe that learned preference attention layer can indeed capture the events critical to the human decision



Reference

- https://sites.google.com/view/preference-transformer