Decoding Global Preferences: Temporal and Cooperative Dependency Modeling in Multi-Agent Preference-Based Reinforcement Learning

Tianchen Zhu[1], **Yue Qiu**[1], **Haoyi Zhou**[2, 3], **Jianxin Li**[1, 2]
[1]School of Computer Science and Engineering, Beihang University
[2]Zhongguancun Laboratory, Beijing, China
[3]School of Software, Beihang University
{zhutc,qiuyue,zhouhy,lijx}@act.buaa.edu.cn

Abstract

Designing accurate reward functions for reinforcement learning (RL) has long been challenging. Preference-based RL (PbRL) offers a promising approach by using human preferences to train agents, eliminating the need for manual reward design. While successful in single-agent tasks, extending PbRL to complex multi-agent scenarios is nontrivial. Existing PbRL methods lack the capacity to comprehensively capture both temporal and cooperative aspects, leading to inadequate reward functions. This work introduces an advanced multi-agent preference learning framework that effectively addresses these limitations. Based on a cascaded Transformer architecture, our approach captures both temporal and cooperative dependencies, alleviating issues related to reward uniformity and intricate interactions among agents. Experimental results demonstrate substantial performance improvements in multi-agent cooperative tasks, and the reconstructed reward function closely resembles expert-defined reward functions. The source code is available at https://github.com/catezi/MAPT.

Introduction

In recent years, deep reinforcement learning (RL) achieved remarkable success in solving complex sequential decision-making problems across diverse domains like games (Mnih et al. 2013; Vinyals et al. 2019; Silver et al. 2016), autonomous driving (Zhou et al. 2020b), robot control (Chen et al. 2022b), and other settings with well-defined reward structures. In these contexts, agents can develop effective policies by maximizing cumulative rewards when an appropriate reward structure is available. However, formulating practical reward structures often poses difficulties, demanding strong expertise in reward engineering (Schenck and Fox 2017; Yahya et al. 2017; Peng et al. 2020). The effectiveness of the created reward structure dramatically hinges on the designer's grasp of task objectives, operational principles, and pertinent background knowledge (Leike et al. 2018). Even domain experts must devote significant time to experiment with various methods to navigate the complexities of reward engineering (Christiano et al. 2017; Lee, Smith, and Abbeel 2021). This challenge is particularly pronounced in multi-agent collaborative scenarios characterized by intricate interactions, where explicitly defining reward structures can be problematic (Song et al. 2018). Furthermore, even if explicit reward structures are crafted, agents might fall into reward manipulation during the policy learning and optimization phase (Skalse et al. 2022). It involves exploiting vulnerabilities in the reward structure to maximize cumulative rewards without genuinely addressing the intended tasks (Leike et al. 2018; Ouyang et al. 2022).

Several methods exist to address the challenges of reward engineering, including imitation learning (Ho and Ermon 2016). This method involves mimicking expert trajectories to learn implicit reward functions. While imitation learning performs well in some tasks, surpassing human-level performance remains difficult. To overcome this limitation, human feedback-based RL emerges as a more versatile and practical alternative. This approach aims to learn implicit reward functions by leveraging expert human feedback and providing reward signals to the agent. Different types of human feedback are employed, including value feedback (Daniel et al. 2014), expert demonstrations (Ng and Russell 2000), preference feedback (Akrour, Schoenauer, and Sebag 2011; Wilson, Fern, and Tadepalli 2012; Kim et al. 2023), and language instructions (Fu et al. 2019). Preference-based RL (PbRL) has gained recent research attention due to its cost-effectiveness and rich information (Wirth et al. 2017; Chen et al. 2022a). In PbRL, the agent's reward function is derived from human preferences for pairs of trajectories, guiding the agent toward specific goals or desired behaviors (Christiano et al. 2017; Stiennon et al. 2020). Recent studies highlight the performance improvement achieved by learning implicit reward functions from human trajectory preferences (Lee, Smith, and Abbeel 2021; Park et al. 2022; Liang et al. 2022; Liu et al. 2022). In summary, PbRL offers an effective way to acquire reward functions based on human intent rather than predefined designs. Its effectiveness spans diverse domains like robot control (Lee, Smith, and Abbeel 2021) and dialogue systems (Ouyang et al. 2022).

PbRL shows promise in enhancing reward and policy learning for single-agent tasks. However, applying these methods to multi-agent systems faces a crucial challenge: agent interactions. Despite numerous proposed PbRL approaches, achieving superior performance improvement through human feedback alone remains challenging for

Multi-Agent RL (MARL). The central question is **how to effectively model reward functions that consider agent interactions and untangle the complex global dependencies between human preferences in multi-agent tasks**. In fact, human preferences for multi-agent trajectories stem primarily from evaluating joint actions and cooperative effects, rather than isolated assessments of individual actions. Creating separate reward models for each agent would overlook the ability to capture agent cooperation, resulting in suboptimal outcomes (Wang et al. 2021). On the other hand, establishing a centralized, shared reward model among all agents introduces challenges in assigning credit (Song et al. 2018; Zhou et al. 2020a), often impeding enhancements in the collective performance of multi-agent systems.

To address this, we extend traditional PbRL methods to handle multi-agent scenarios and accurately represent the connections between human preferences and agents' combined rewards. Our main idea is to create a decoupled reward model by capturing global dependencies (including temporal and cooperation two aspects) for each agent, reflecting their unique contributions. Consequently, we propose a novel global dependency-enhanced multi-agent preference model that effectively captures the interdependence among human preferences, temporal context, and individual cooperation. It allows us to infer each agent's contributions and critical actions within and across trajectories. Drawing inspiration from successful network structures like the Transformer model (Vaswani et al. 2017), known for effective sequence modeling and RL training (Chen et al. 2021; Wen et al. 2022), we introduce Multi-Agent Preference Transformer (MAPT), a cascaded Transformer-based architecture. MAPT integrates a self-attention layer and a cooperative-aware preference attention layer to capture cooperative global dependencies and calculate significance weights for each agent. It involves extracting pertinent cooperative context from each agent's trajectory segment, generating independent cooperative-dependent rewards for agents. Furthermore, it employs a temporal-aware preference attention layer to capture temporal global dependencies within combined rewards. These weights recalibrate the combined temporal-dependent rewards, shaping a centralized and decoupled collective reward distribution. Finally, we apply the cascaded Transformer architecture to define our multi-agent preference model. The contributions are summarized as follows:

- We pioneer a comprehensive framework for modeling human preferences in multi-agent collaboration tasks, focusing on combined rewards considering temporal and cooperative factors.
- We present MAPT, an innovative reward model enriched with global dependencies. Built on a cascaded Transformer architecture, it incorporates novel preference attention from both history and agent perspectives.
- Extensive experiments across various multi-agent tasks and four benchmarks showcase MAPT's ability to learn precise joint reward distributions from human preferences. The resulting policy model, trained on this distribution, consistently surpasses various baselines.

Background

Preference-Based Reinforcement Learning. We study a cooperative MARL framework, extending the Markov decision processes (MDPs) concept (Littman 1994). A Markov game with N agents is defined by $(N, \mathcal{S}, \{\mathcal{A}_i\}_{i=1}^N, \mathcal{T}, \mathcal{R}, \gamma)$, where $\mathcal{S}$ is the state set, and $\{\mathcal{A}_i\}_{i=1}^N$ is the action set for each agent. The transition function $\mathcal{T} : \mathcal{S} \times \mathcal{A}_1 \times \ldots \times \mathcal{A}_N \to P(\mathcal{S})$ models state transitions. At timestep t, given state $\mathbf{o}_t = (o_t^1, \ldots, o_t^N)$, agents select actions $\mathbf{a}_t = (a_t^1, \ldots, a_t^N)$, and the state transitions to $\mathbf{o}_{t+1}$ with probability $T(\mathbf{o}_{t+1}|\mathbf{o}_t, a_t^1, \ldots, a_t^N)$. The shared reward is $r_t = R(\mathbf{o}, \mathbf{a})$. The joint policy $\boldsymbol{\pi}_\theta = [\pi_{\theta_1}, \ldots, \pi_{\theta_N}]$ represents individual agent policies. For convenience, policy parameters θ might be omitted. Each agent has full access to state information. To denote agents from i^k to i^j $(1 \le k \le j \le N)$, we use the notation $k:j$, where $\boldsymbol{\pi}^{k:j}$ is $\pi^k, \pi^{k+1}, \ldots, \pi^j$. MARL aims to learn a policy maximizing expected return.

Designing precise reward functions that capture human intent is challenging in many applications. PbRL addresses this by learning reward functions from human preferences. In line with previous works (Christiano et al. 2017; Lee, Smith, and Abbeel 2021; Liu et al. 2022; Kim et al. 2023), we consider preferences expressed between pairs of trajectory segments. These segments involve N agents and have a length denoted as T, where $\boldsymbol{\sigma} = \{\sigma^1, \ldots, \sigma^N\}$, with $\sigma^k = \{(o_1, a_1^k), \ldots, (o_T, a_T^k)\}$ for $1 \le k \le N$. Human preferences for a segment pair $(\boldsymbol{\sigma}^0, \boldsymbol{\sigma}^1)$ are indicated by $y \in \{0, 1, 0.5\}$. Here, $y = 1$ denotes a preference for $\boldsymbol{\sigma}^\mathbf{0}$, $y = 0$ for $\boldsymbol{\sigma}^\mathbf{1}$, and $y = 0.5$ for equal preference. We represent preference as $\boldsymbol{\sigma}^k \succ \boldsymbol{\sigma}^j$ if segment k is preferred over segment j. The preference dataset, denoted as $\mathcal{D}$, stores each feedback as $(\boldsymbol{\sigma}^0; \boldsymbol{\sigma}^1; y)$.

To obtain a scalar reward function $\hat{r}$ parameterized by ψ, most prior works define a preference predictor following the Bradley-Terry model (Bradley and Terry 1952) under single-agent settings, and we extend it to the multi-agent form:

$$P[\boldsymbol{\sigma}^0 \succ \boldsymbol{\sigma}^1; \psi] = \frac{\exp\Big(\sum\limits_{\substack{\mathbf{o}_t, \mathbf{a}_t \in \boldsymbol{\tau}^0 \\ 1 \le t \le |\boldsymbol{\tau}^0|}} \hat{r}(\mathbf{o}_t^0, \mathbf{a}_t^0; \psi)\Big)}{\sum_{m=0}^{1} \exp\Big(\sum\limits_{\substack{\mathbf{o}_t, \mathbf{a}_t \in \boldsymbol{\tau}^m \\ 1 \le t \le |\boldsymbol{\tau}^m|}} \hat{r}(\mathbf{o}_t^m, \mathbf{a}_t^m; \psi)\Big)}. \tag{1}$$

Then, given a dataset of preferences $\mathcal{D}$, the reward function $\hat{r}$ is updated by minimizing the cross-entropy loss between this preference predictor and the actual human labels:

$$\mathcal{L}^{\mathrm{CE}}(\psi) = - \mathop{\mathbb{E}}_{(\boldsymbol{\sigma}^0, \boldsymbol{\sigma}^1, y) \sim \mathcal{D}} [y \log P(\boldsymbol{\sigma}^0 \succ \boldsymbol{\sigma}^1; \psi) + (1 - y) \log P(\boldsymbol{\sigma}^1 \succ \boldsymbol{\sigma}^0; \psi)]. \tag{2}$$

The joint policy $\boldsymbol{\pi}_\theta$ can be updated using any MARL algorithm, ensuring that it maximizes the expected returns concerning the learned reward.

Multi-Agent Policy Gradient. MAPPO (Yu et al. 2022), HAPPO (Kuba et al. 2022) and MAT (Wen et al. 2022) establish a sequential modeling framework for MARL and are

state-of-the-art algorithms based on the popular *Proximal Policy Optimization* (PPO) method (Schulman et al. 2017), which is known for its simplicity and stable performance.

Methodology

This section introduces the Multi-agent Preference Transformer (MAPT), a novel cascaded Transformer architecture for modeling human preferences in multi-agent environments. Our approach captures temporal and cooperative dependencies among agents' behaviors, facilitating two-dimensional credit assignments. A new preference predictor is employed, utilizing a global dependency-enhanced reward model to improve agent behavior representation. We detail the architecture of MAPT, as shown in Fig. (1), designed to accommodate the proposed preference predictor.

Problem Statement

Existing PbRL algorithms follow a two-step approach involving reward modeling and policy optimization. The challenge lies in accurately reconstructing reward functions from preference data, directly affecting policy optimization efficiency. To improve reward reconstruction quality, PbRL methods for single-agent scenarios explore whether the reward function stationarity assumption holds (Early et al. 2022; Kim et al. 2023). They adopt a time-independent paradigm (local perception, rewards based on current state and action) when stationarity holds (Christiano et al. 2017; Lee, Smith, and Abbeel 2021), and a time-dependent paradigm (global perception, rewards based on historical states and actions) when it does not (Early et al. 2022).

Multi-agent PbRL is intricate due to interactions among agents. Directly applying single-agent PbRL methods to multi-agent scenarios can lead to substantial reward modeling biases, impeding effective policy learning. In cooperative settings, reward function non-stationarity is influenced by both temporal and cooperative factors. Consider a team-based soccer match: offensive and defensive strategies evolve, altering the reward function (optimization goal) over different phases, indicating temporal non-stationarity. Moreover, an individual agent's reward distribution can change based on teammates' states and actions. For instance, the success of agent A passing to teammate B, yielding a high reward, depends on A's skill and B's positioning. It highlights the need to perceive global interdependencies across time and cooperation levels when modeling and learning from human preferences in multi-agent collaborative tasks.

The two left images in Fig. (2) depict the reward distribution learned through reward model training using an extension of classic single-agent PbRL methods (Christiano et al. 2017; Kim et al. 2023) to collaborative multi-agent settings. The right image in Fig. (2) shows empirically grounded reward distributions for these state-action pairs. These explicit rewards are based on heuristic reward functions, reflecting human insights, and provide diverse experiences that enable agents to explore and produce coherent, high-quality behaviors. In contrast, inferred rewards from reward models appear more congested, needing clear differentiation among most state-action pair feedback. These homogeneous rewards result in almost identical feedback for any action in any state, leading to low variance in the advantage function. As a result, agents need help to obtain valuable positive incentives, limiting diverse action and policy exploration. In MARL, exploration is vital for effective interaction and coordination among agents. Reward differentiation becomes crucial to ensure diverse strategy exploration, enhancing exploration efficiency and convergence.

Preference Modeling

As discussed earlier, local-dependency preference predictors have limitations in learning from human preferences, especially in multi-agent scenarios. How information is aggregated significantly affects the diversity of individual reward distribution and agent exploration, particularly in multi-agent contexts. When creating independent reward models and local-dependency preference predictors, each agent constructs its reward function based on local observations, treating other agents as part of the environment rather than explicitly modeling interactions. It tends to have similar reward functions for individual agents, failing to capture intricate interactions and action importance among agents. It necessitates credit assignment both within and across trajectories. To address these challenges, we introduce a novel multi-agent preference predictor. This predictor utilizes a global dependency-enhanced reward model, assuming that preferring a trajectory segment depends exponentially on the weighted sum of rewards assigned to individual agents and timesteps. In this framework, an agent's reward depends on the weighted aggregation of globally enhanced rewards specific to that agent:

$$P[\boldsymbol{\sigma}^0 \succ \boldsymbol{\sigma}^1; \psi] = \frac{\exp\left(\sum_{t=1}^{|\boldsymbol{\tau}^0|} w_t^0 \cdot \mathrm{AGG}(\left\{\hat{r}_t^{n,0}\right\}_{n=1}^{N})\right)}{\sum_{m=0}^{1} \exp\left(\sum_{t=1}^{|\boldsymbol{\tau}^m|} w_t^m \cdot \mathrm{AGG}(\{\hat{r}_t^{n,m}\}_{n=1}^{N})\right)}, \tag{3}$$

where AGG denotes an aggregation operator used to aggregate individual rewards to the collective reward.

To capture global dependencies in both temporal and cooperative aspects, we introduce an enhanced reward function $\hat{r}_t^n$ for each agent i^n at timestep t. This function takes the complete previous sub-trajectory of length T for agent i^n as input. Additionally, we use the AGG operator to capture cooperative-wise dependencies. It enables credit assignment across segments. We also introduce temporal importance weights w_t^n for the T-length sub-trajectory segment associated with agent i^n, allowing credit assignment within segments. This formulation encompasses traditional design and converges to the standard model in specific scenarios. Overall, our multi-agent preference predictor considers joint actions from all agents within the global state, effectively capturing global dependencies between action advantages and environmental states among agents. It fosters a diversified reward structure, recognizing the mutual influence of agents' behaviors and facilitating cooperative actions.

Architecture of Multi-Agent Preference Learning

To implement the multi-agent preference predictor described in Eq. (3), we introduce a new cascaded

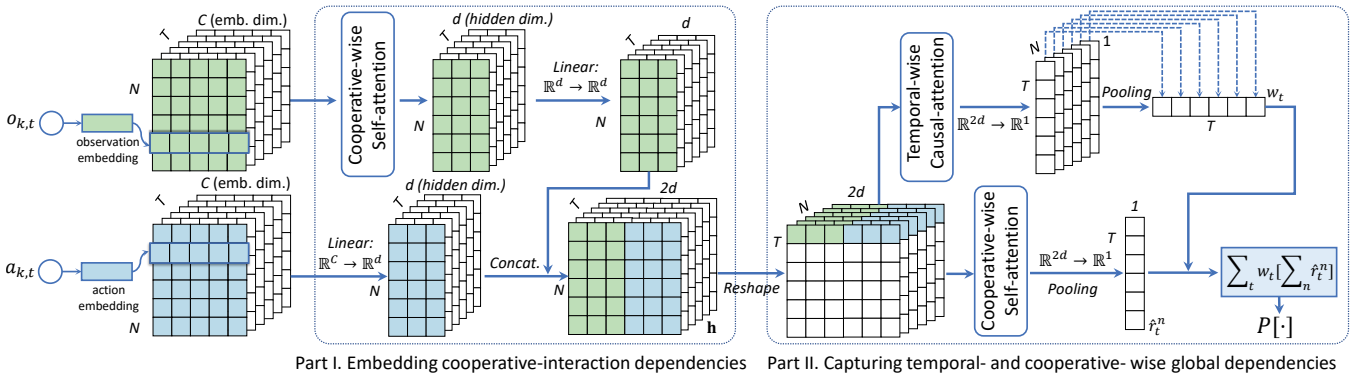

Figure 1: Overview of MAPT. We first construct cooperative hidden embeddings through the self-attention layer, where each represents the context information from the agent i^1 to i^N. Then we construct two preference attention layers with bidirectional and causal self-attention mechanisms to compute the rewards $\hat{r}_t^n$ and aggregate them for modeling the weighted sum of global dependency-enhanced rewards $\sum_t [w_t \sum_n (\hat{r}_t^n)]$.

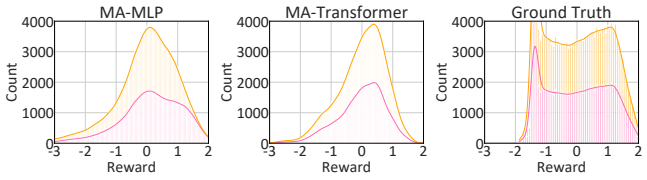

Figure 2: Comparison of reward histograms: Traditional PbRL vs. Human-designed environmental rewards in multi-agent context. Data from 100,000 state-action pairs in Door-CloseOutward training. Each color denotes an agent.

Transformer-based architecture named MAPT. This architecture includes the following components:

Cooperative-Interaction Dependency Embedding. We employ the Transformer architecture (Vaswani et al. 2017) as the foundation, renowned for its effectiveness in sequential decision problems (Chen et al. 2021; Janner, Li, and Levine 2021). Specifically, following (Wen et al. 2022), we utilize the self-attention architecture. Observations are embedded as $o^n \in \mathbb{R}^{T\times C}$ and actions as $a^n \in \mathbb{R}^{T\times C}$ for agent i^n, with T denoting the segment length and C indicating the embedding dimension. Subsequently, a self-attention layer is applied to encode the entire observation sequence $\mathbf{o} = (o^1, \ldots, o^N)$. Then the encoded observation embedding sequence and action embedding sequence $\mathbf{a} = (a^1, \ldots, a^N)$ are concatenated into a hidden embedding as $\mathbf{h} = (h^1, \ldots, h^n) \in \mathbb{R}^{T\times 2d}$, with d denoting the hidden dimension.

Cooperative-Aware Preference Attention Layer. We introduce a cooperative-wise preference attention layer to incorporate cooperative-wise global dependencies for reward modeling as defined in Eq. (3). As depicted in Fig. (3c), this layer takes hidden embeddings $\mathbf{h}_t$ from the embedding layer as input and generates individual rewards $\hat{r}_t^n$ at timestep t for agent i^n. $\mathbf{h}_t$ is embedded by a self-attention block. In detail, each input h_t^n is linearly transformed into a key $\mathbf{K}^n \in \mathbb{R}^d$, query $\mathbf{Q}^n \in \mathbb{R}^d$, and value $\mathbf{V}^n \in \mathbb{R}^d$. Notably, $\hat{r}_t^n$ denotes the cooperative global-dependency enhanced individual reward. Specifically, for agent i^n, we utilize N state-action hidden embeddings $(h_t^1, \ldots, h_t^N)$ from all agents to approximate the individual reward $\hat{r}_t^n$ during training. Leveraging the principles of self-attention (Vaswani et al. 2017), the individual reward $\hat{r}_t^n$ for agent i^n is defined as a convex combination of values with attention weights derived from the n-th query and N keys:

$$\hat{r}_t^n = \text{softmax}(\{\langle \mathbf{Q}^n, \mathbf{K}^k \rangle\}_{k=1}^N)_t^n \cdot \mathbf{V}_t^n, \tag{4}$$

Then, we apply the average pooling operator to aggregate the individual rewards $\hat{r}_t^n$, resulting in the collective reward $\hat{r}_t$ given by the equation $\hat{r}_t = \frac{1}{N}\sum_{n=1}^N \hat{r}_t^n$. Overall, The cooperation-aware preference attention layer and the average pooling layer make up the operator AGG in Eq. (3).

Temporal-Aware Preference Attention Layer. We also introduce a temporal-wise preference attention layer to enhance the multi-agent preference predictor in the temporal global dependencies manner according to Eq. (3). In Fig. (3d), this layer takes $\hat{r}_t$ and $\mathbf{h}$ as input and generates temporal importance weights w_t. Specifically, the input $h_t = \frac{1}{N}\sum_{n=1}^N \mathbf{h}_t^n$ is transformed into a query $\mathbf{Q}_t \in \mathbb{R}^d$, while the input $\hat{r}_t^{1:N}$ is transformed into key $\mathbf{K}_t \in \mathbb{R}^d$ and value $V_t = \hat{r}_t \in \mathbb{R}$. According to the self-attention mechanism (Vaswani et al. 2017), the output x_k at k-th timestep is determined by a weighted combination of values, using temporal-wise attention weights from the corresponding query and keys:

$$x_k = \sum_{t=1}^T \text{softmax}(\{\langle \mathbf{Q}_k, \mathbf{K}_{t'} \rangle\}_{t'=1}^T)_t \cdot \hat{r}_t. \tag{5}$$

Then the weighted sum of rewards can be computed by the average of outputs $\{x_1, \ldots, x_T\}$ as follows:

$$\frac{1}{T}\sum_{k=1}^T x_k = \frac{1}{T}\sum_{k=1}^T\sum_{t=1}^T \text{softmax}(\{\langle \mathbf{Q}_k, \mathbf{K}_{t'} \rangle\}_{t'=1}^T)_t \cdot \hat{r}_t$$
$$= \sum_{t=1}^T \frac{1}{T}\sum_{k=1}^T \text{softmax}(\{\langle \mathbf{Q}_k, \mathbf{K}_{t'} \rangle\}_{t'=1}^T)_t \cdot \hat{r}_t = \sum_{t=1}^T w_t \hat{r}_t, \tag{6}$$

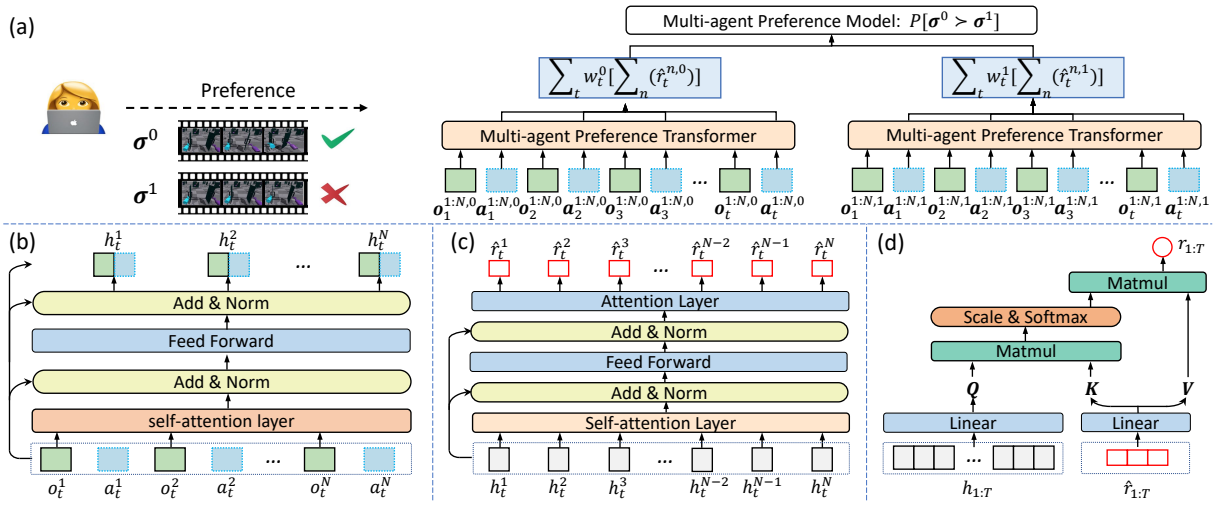

Figure 3: The overall model architecture of MAPT and its critical components: (a) Overall architecture of MAPT. (b) Cooperative-interaction dependency modeling layer. (c) Cooperative-wise preference attention layer for cooperative global dependency modeling. (d) Temporal-wise preference attention layer for temporal global dependency modeling.

where $w_t = \frac{1}{T}\sum_{k=1}^{T} \text{softmax}(\{\langle \mathbf{Q}_k, \mathbf{K}_{t'} \rangle\}_{t'=1}^{T})_t$.

In summary, we model cooperative- and temporal-wise global dependency-enhanced rewards by averaging outputs from the cooperative and temporal-wise attention layers. We assume that the probability of preferring a segment is proportional to this aggregation. The overall architecture of the proposed MAPT is depicted in Fig. (3a).

Training and Inference

Training. We train MAPT using cross-entropy loss on the preference dataset $\mathcal{D}$ to align its preference predictor with human labels. This process guides deriving a suitable reward function and identifying important agent behaviors within trajectory segments. In Part I, we transform each trajectory segment (o, a) of length T into two 3D matrices of size $T \times N \times C$. A self-attention layer is applied to extract cooperative interaction features, and the action embeddings are concatenated to N encoded observation embeddings as $\mathbf{h} \in \mathbb{R}^{T \times N \times 2d}$. In Part II, two branches are pursued. First, the feature $\mathbf{h}$ from Part I is reshaped to $\mathbb{R}^{N \times T \times 2d}$, and cooperative global dependencies are extracted using a cooperation-wise preference attention layer and mapped to $\mathbb{R}^{N \times T \times 1}$. And then the results are aggregated along the agent dimension to produce $\mathbb{R}^{1 \times T}$. Simultaneously, the feature $\mathbf{h}$ obtained in Part I is also pooled along the agent dimension to produce $\mathbb{R}^{T \times 2d}$. Then, temporal global dependencies are captured using a temporal-wise preference attention layer, generating temporal weights $w \in \mathbb{R}^{T \times 1}$. Using the temporal weights, a weighted sum computation yields the preference reward for the trajectory segment.

Inference. We label all state-action pairs using the learned global dependency-enhanced reward function during MARL training. To achieve this, we supply MAPT with T previous transitions $(\mathbf{o}_{t-T+1}^{1:N}, \mathbf{a}_{t-T+1}^{1:N}, \ldots, \mathbf{o}_t^{1:N}, \mathbf{a}_t^{1:N})$ and utilize the t-th value r_t from the temporal-wise preference attention layer as the reward for the timestep t.

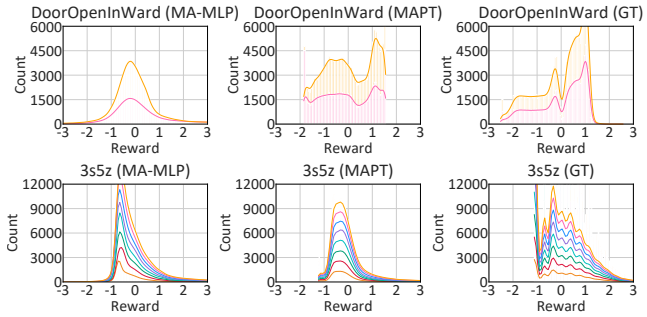

Figure 4: Comparative reward histograms: Traditional PbRL (MA-MLP), MAPT, and human-designed rewards (GT) in multi-agent settings. Based on 100,000 state-action pairs, uniformly sampled from replay buffers during training in two environments. Colors indicate different agents.

Experiments

In this section, we assess MAPT's performance across four benchmarks and compare it against various baseline methods. Our experiments address three key questions: (1) Can human preferences effectively guide multi-agent learning in complex control and coordination tasks, surpassing expert trajectories? (2) Does the integration of global dependencies in preference predictors lead to diverse rewards, alleviating exploration challenges arising from reward homogenization? (3) Is the reward function learned from the model consistent with the ground truth reward function?

Experimental Setup

Benchmark Datasets. We evaluated MAPT using four benchmarks: StarCraftII Multi-Agent Challenge (SMAC) (Samvelyan et al. 2019), Google Research Football (Football) (Kurach et al. 2020), Bimanual Dexterous Hands Manipulation (Bi-DexHands) (Chen et al. 2022b),

Tasks	Training with task rewards	Training with demonstations	Training with preferences			
	MAT	**OMAR**	**MA-MLP**	**MA-LSTM**	**MA-Transformer**	**MAPT (Ours.)**
3m	20.0±0.00	19.82±0.02	0.66±0.01	19.87±0.11	0.87±0.03	**19.98±0.05**
3s5z	20.00±0.02	19.87±0.11	1.77±0.17	14.52±0.49	5.86±0.28	**19.59±0.42**
6h vs 8z	19.78±0.07	18.33±0.15	9.57±0.14	9.76±0.15	6.36±0.17	**17.05±0.53**
MMM2	20.52±0.09	15.99±0.22	0.80±0.05	1.39±0.08	1.89±0.16	**3.68±0.10**
3 vs 1	4.89±0.02	4.55±0.34	0.72±0.06	1.31±0.03	1.06±0.02	**3.78±0.07**
counter attack	4.77±0.16	1.14±0.18	2.98±0.08	0.83±0.06	0.24±0.06	**4.21±0.10**
pass and shoot	4.83±0.11	2.72±0.58	2.20±0.06	1.90±0.03	0.66±0.06	**4.76±0.04**
CatchOver	25.32±0.88	16.85±1.21	8.62±0.34	**25.34±1.59**	4.75±0.13	25.12±0.64
DoorOpenInward	402.13±0.44	114.47±34.31	224.96±31.32	242.42±33.92	171.23±14.53	**372.60±11.38**
DoorOpenOutward	440.17±2.46	113.62±12.85	64.95±4.46	123.76±12.88	28.88±7.85	**228.08±10.98**
DoorCloseOutward	981.82±0.43	818.76±2.43	515.81±36.61	737.67±25.29	492.45±28.05	**786.70±26.03**
HalfCheetah 6×1	4483.95±74.75	4088.93±165.67	-88.75±11.62	1132.20±116.15	1317.90±147.77	**2423.50±128.33**

Table 1: Average accumulated trajectory rewards of learning from different feedbacks across baselines on 4 benchmarks (SMAC, Football, Bi-dexhands, and Ma-Mujoco). Using the same multi-agent preference dataset from scripted teachers, we train 3 baselines and MAPT. The result shows the average and standard deviation averaged over 8 runs.

and Multi-agent MuJoCo (Ma-Mujoco) (de Witt et al. 2020). Synthetic preferences generated by scripted teachers were used, akin to previous studies (Christiano et al. 2017; Lee, Smith, and Abbeel 2021; Kim et al. 2023). The deterministic teacher generates preferences based on task rewards, with preference label y determined by $y = \arg\min_j \sum_{t=1}^{T} r(\mathbf{o}_t^j, \mathbf{a}_t^j)$. The multi-agent preference datasets comprised 50,000 segment pair preferences for SMAC and Football, and 30,000 for Bi-DexHands and Ma-Mujoco. Preferences were extracted from replay buffers during scripted teacher training using HAPPO and MAT, structured as $(\boldsymbol{\tau}^0; \boldsymbol{\tau}^1; y)$, encapsulating sequential segments $\boldsymbol{\tau}^0$ and $\boldsymbol{\tau}^1$ with associated preference label y.

Baselines. We compare our method against various multi-agent imitation learning and classical PbRL methods. Similar to (Kim et al. 2023), we consider different aggregation modes of preference reward function as baselines, including the standard Markov reward model and the reward model based on temporal global dependency aggregation. For the standard Markov reward model, the preference predictor employs an MLP-modeled reward function similar to (Christiano et al. 2017; Lee, Smith, and Abbeel 2021). For global temporal dependency aggregation-based reward models, LSTM-based (Early et al. 2022) and Transformer-based (Kim et al. 2023) architectures are used as reward functions for shaping the preference predictor. To adapt existing PbRL algorithms to the multi-agent context, we formulate them within the multi-agent framework using Eq. (1). Our evaluation involves baseline algorithms, including MAGAIL (Song et al. 2018), OMAR (Pan et al. 2022), multi-agent MLP-based preference model (MA-MLP), multi-agent LSTM-based preference model (MA-LSTM), and multi-agent Transformer-based preference model (MA-Transformer).

Implementation Details. We randomly select pairs of trajectory segments from offline datasets for reward learning and gather preferences from scripted teachers. These preferences are used to create multi-agent preference datasets, which are then utilized to develop a reward function for training RL agents. We train RL agents using MAT (Wen et al. 2022), a recent state-of-the-art algorithm known for its strong performance on various multi-agent cooperative benchmarks. Baselines are implemented according to their official repositories, with hyperparameters maintained at their original best-performing settings. For reward models, we set the learning rate to 1e-4 and the hidden dimension to 256. As for policy models, following MAT, the learning rates for actors and critics are 5e-4 for SMAC and Football, and 5e-5 for Bi-DexHands and Ma-Mujoco. Our models are trained on a single NVIDIA Tesla V100 GPU.

Main Results

Tab. (1) summarizes the performance evaluation of the policy model across various reward functions learned by baseline algorithms. Notably, MAPT consistently outperforms all baseline algorithms across most tasks. Our approach successfully aligns the policy model's performance with task rewards, especially in complex multi-agent cooperative tasks where most baselines struggle to contribute meaningfully. In benchmarks with continuous action spaces like Bi-Dexhands and Ma-Mujoco, MAPT achieves an average performance improvement of 46%. In tasks with discrete action spaces like SMAC and Football, MAPT consistently enhances accumulative trajectory rewards by 35% to 189% under diverse difficulty levels. In challenging SMAC tasks demanding high-difficulty intricate cooperation, such as 3s5z and 6s vs 8Z, baseline algorithms fail to guide agent learning, yielding a 0% win rate effectively. In contrast, deploying MAPT substantially boosts the win rate to 94% and 62%, highlighting the importance of capturing temporal- and cooperative-wise global dependencies. Remarkably, MAPT outperforms specific SOTA multi-agent imitation learning methods guided by human demonstrations in some environ-

Tasks	MAPT	MAPT-TPA	MAPT-CPA
OpenIn.	**292.16±11.38**	23.43±0.95	182.13±13.68
CloseOut.	**786.70±26.03**	718.16±22.38	782.37±24.05
3s5z	**19.59±0.42**	8.32±0.38	9.09±0.10
6h vs 8z	**17.05±0.53**	13.26±0.31	5.56±0.04

Table 2: Ablation study to assess global dependency effects. MAPT-TPA omits the temporal-aware preference layer. MAPT-CPA omits the cooperation-aware preference layer.

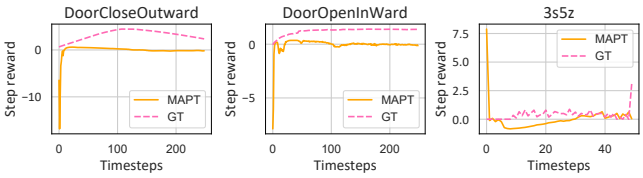

Figure 5: Time series of the learned reward function and the ground truth (GT) reward using rollouts from a policy optimized by MAPT.

ments. Human preference may be a more efficient and less biased feedback form than expert demonstrations, holding significant promise for offline MARL. These results underscore the effectiveness of our global dependency-enhanced preference model using a cascaded Transformer architecture. MAPT's success in reward learning demonstrates its ability to derive suitable reward functions from human preferences, guiding agents towards meaningful behaviors.

Reward Analysis

To evaluate the efficacy of introducing preference modeling with global dependencies in addressing the reward function homogenization issue and enhancing reward diversity, we visualized reward distributions of MAPT and the local dependency modeling method (MA-MLP) across numerous state-action pairs. These were compared with the reward distribution generated by human-designed Ground Truth (GT), as depicted in Fig. (4). The outcomes reveal that reward functions based on local dependency tend to yield concentrated marginal distributions, offering similar rewards for most cases, making it challenging to differentiate advantageous actions. In contrast, MAPT, incorporating global dependency modeling showcases a more dispersed marginal reward distribution akin to the Ground Truth. This diverse feedback effectively distinguishes advantageous actions and guides agent training. Notably, as agent numbers increase, the credit assignment challenge intensifies, aggravating homogenization effects. Even in this scenario, MAPT demonstrates advantages.

To assess the quality of learned reward functions, we compare them with ground truth rewards. Fig. (5) illustrates MAPT's optimized learned reward functions from scripted teachers across various environments. While the scale differs from the ground truth reward due to unconstrained output, the learned reward function remains reasonably aligned.

Tasks	MAPT	MAPT*	MAPT*-CPA
OpenIn.	**292.16±11.38**	213.42±12.74	276.96±5.27
CloseOut.	**786.70±26.03**	487.12±19.64	729.31±30.72
3s5z	**19.59±0.42**	18.43±0.32	19.48±0.39
6h vs 8z	**17.05±0.53**	14.54±0.33	17.89±0.49

Table 3: Ablation study to assess self-attention architecture. MAPT* uses an LSTM network instead of self-attention. MAPT*-CPA removes the cooperation-aware preference layer from MAPT*.

Ablation Studies

Contribution of Global Dependencies. To evaluate the enhanced effects of global dependencies on reward function reconstruction and preference modeling, we conducted experiments by removing the Temporal-aware Preference Attention (TPA) layer and the Cooperation-aware Preference Attention (CPA) layer from the base MAPT. It helps us assess the impact of introducing independent temporal-wise and cooperative-wise global dependencies. Tab. (2) presents the results of the two variations of MAPT across four environments. Removing the TPA layer leads to a significant performance decline, indicating its importance in modeling reward functions tied to crucial actions and temporal events. Likewise, eliminating the CPA layer also results in substantial performance reduction. The introduction of cooperative-wise global dependency enables non-independent reward function modeling, aiding the separation of individual agent contributions and effective confidence allocation. These findings underscore the critical role of the components within MAPT, demonstrating their significance in the success of our approach.

Contribution of Self-Attention Mechanism. To demonstrate the superiority of the proposed cascaded Transformer, particularly its self-attention mechanism, in capturing global dependencies, we replaced the self-attention network in the model with an LSTM network. The results in Tab. (3) indicate that MAPT with the self-attention network performs significantly better than MAPT with the LSTM network. This improvement can be attributed to the self-attention mechanism's robust ability to model long-distance dependencies effectively.

Conclusion

In this study, we present MAPT, a new framework focusing on global dependency modeling using a cascaded Transformer architecture. It incorporates specific preference attention layers to capture complex global relationships among agents. By utilizing self-attention, we assign weight to rewards for each agent, resulting in MAPT outperforming baseline methods in challenging multi-agent tasks. Our analysis underscores the importance of self-attention in capturing distant correlations, essential for extracting global dependencies and interpreting reward signals accurately. Additionally, our reconstructed reward function closely resembles expert-defined distributions.

Acknowledgements

We thank all reviewers for their thoughtful and insightful suggestions. This work was supported by the National Key R&D Program of China (2021ZD0113903), grants from the Natural Science Foundation of China (62225202, 62202029), and Young Elite Scientists Sponsorship Program by CAST (No. 2023QNRC001). Thanks for the computing infrastructure provided by Beijing Advanced Innovation Center for Big Data and Brain Computing. This work was also sponsored by CAAI-Huawei MindSpore Open Fund. Haoyi Zhou is the corresponding author.

References

Akrour, R.; Schoenauer, M.; and Sebag, M. 2011. Preference-Based Policy Learning. In *ECML/PKDD*, volume 6911, 12–27.

Biyik, E.; and Sadigh, D. 2018. Batch Active Preference-Based Learning of Reward Functions. In *CoRL*, volume 87, 519–528.

Bradley, R. A.; and Terry, M. E. 1952. Rank Analysis of Incomplete Block Designs: I. The Method of Paired Comparisons. *Biometrika*, 39: 324.

Chen, L.; Lu, K.; Rajeswaran, A.; Lee, K.; Grover, A.; Laskin, M.; Abbeel, P.; Srinivas, A.; and Mordatch, I. 2021. Decision Transformer: Reinforcement Learning via Sequence Modeling. In *NeurIPS*, 15084–15097.

Chen, X.; Zhong, H.; Yang, Z.; Wang, Z.; and Wang, L. 2022a. Human-in-the-loop: Provably Efficient Preference-based Reinforcement Learning with General Function Approximation. In *ICML*, volume 162, 3773–3793.

Chen, Y.; Wu, T.; Wang, S.; Feng, X.; Jiang, J.; Lu, Z.; McAleer, S.; Dong, H.; Zhu, S.; and Yang, Y. 2022b. Towards Human-Level Bimanual Dexterous Manipulation with Reinforcement Learning. In *NeurIPS*.

Christiano, P. F.; Leike, J.; Brown, T. B.; Martic, M.; Legg, S.; and Amodei, D. 2017. Deep Reinforcement Learning from Human Preferences. In *NeurIPS*, 4299–4307.

Daniel, C.; Viering, M.; Metz, J.; Kroemer, O.; and Peters, J. 2014. Active Reward Learning. In *Robotics: Science and Systems*.

Dasari, S.; and Gupta, A. 2020. Transformers for One-Shot Visual Imitation. In *CoRL*, volume 155, 2071–2084.

de Witt, C. S.; Peng, B.; Kamienny, P.; et al. 2020. Deep Multi-Agent Reinforcement Learning for Decentralized Continuous Cooperative Control. *arXiv preprint arXiv:2003.06709*.

Early, J.; Bewley, T.; Evers, C.; and Ramchurn, S. D. 2022. Non-Markovian Reward Modelling from Trajectory Labels via Interpretable Multiple Instance Learning. In *NeurIPS*.

Fu, J.; Korattikara, A.; Levine, S.; and Guadarrama, S. 2019. From Language to Goals: Inverse Reinforcement Learning for Vision-Based Instruction Following. In *ICLR*.

Ho, J.; and Ermon, S. 2016. Generative Adversarial Imitation Learning. In *NeurIPS*, 4565–4573.

Ibarz, B.; Leike, J.; Pohlen, T.; Irving, G.; Legg, S.; and Amodei, D. 2018. Reward learning from human preferences and demonstrations in Atari. In *NeurIPS*, 8022–8034.

III, D. J. H.; and Sadigh, D. 2022. Few-Shot Preference Learning for Human-in-the-Loop RL. In *CoRL*, volume 205, 2014–2025.

Janner, M.; Li, Q.; and Levine, S. 2021. Offline Reinforcement Learning as One Big Sequence Modeling Problem. In *NeurIPS*, 1273–1286.

Kim, C.; Park, J.; Shin, J.; Lee, H.; Abbeel, P.; and Lee, K. 2023. Preference Transformer: Modeling Human Preferences using Transformers for RL. In *ICLR*.

Kuba, J. G.; Chen, R.; Wen, M.; Wen, Y.; Sun, F.; Wang, J.; and Yang, Y. 2022. Trust Region Policy Optimisation in Multi-Agent Reinforcement Learning. In *ICLR*.

Kurach, K.; Raichuk, A.; Stanczyk, P.; et al. 2020. Google Research Football: A Novel Reinforcement Learning Environment. In *AAAI*, 4501–4510.

Lee, K.; Smith, L. M.; and Abbeel, P. 2021. PEBBLE: Feedback-Efficient Interactive Reinforcement Learning via Relabeling Experience and Unsupervised Pre-training. In *ICML*, volume 139, 6152–6163.

Leike, J.; Krueger, D.; Everitt, T.; Martic, M.; Maini, V.; and Legg, S. 2018. Scalable agent alignment via reward modeling: a research direction. *arXiv preprint arXiv:1811.07871*.

Liang, X.; Shu, K.; Lee, K.; and Abbeel, P. 2022. Reward Uncertainty for Exploration in Preference-based Reinforcement Learning. In *ICLR*.

Littman, M. L. 1994. Markov Games as a Framework for Multi-Agent Reinforcement Learning. In *ICML*, 157–163.

Liu, R.; Bai, F.; Du, Y.; and Yang, Y. 2022. Meta-Reward-Net: Implicitly Differentiable Reward Learning for Preference-based Reinforcement Learning. In *NeurIPS*.

Meng, L.; Wen, M.; Le, C.; et al. 2023. Offline Pre-trained Multi-agent Decision Transformer. *Mach. Intell. Res.*, 20(2): 233–248.

Mnih, V.; Kavukcuoglu, K.; Silver, D.; Graves, A.; Antonoglou, I.; Wierstra, D.; and Riedmiller, M. A. 2013. Playing Atari with Deep Reinforcement Learning. *arXiv preprint arXiv:1312.5602*.

Ng, A. Y.; and Russell, S. 2000. Algorithms for Inverse Reinforcement Learning. In *ICML*, 663–670.

Ouyang, L.; Wu, J.; Jiang, X.; et al. 2022. Training language models to follow instructions with human feedback. In *NeurIPS*.

Pan, L.; Huang, L.; Ma, T.; and Xu, H. 2022. Plan Better Amid Conservatism: Offline Multi-Agent Reinforcement Learning with Actor Rectification. In *ICML*, volume 162, 17221–17237.

Parisotto, E.; Song, H. F.; Rae, J. W.; et al. 2020. Stabilizing Transformers for Reinforcement Learning. In *ICML*, volume 119, 7487–7498.

Park, J.; Seo, Y.; Shin, J.; Lee, H.; Abbeel, P.; and Lee, K. 2022. SURF: Semi-supervised Reward Learning with Data Augmentation for Feedback-efficient Preference-based Reinforcement Learning. In *ICLR*.

Peng, X. B.; Coumans, E.; Zhang, T.; Lee, T. E.; Tan, J.; and Levine, S. 2020. Learning Agile Robotic Locomotion Skills by Imitating Animals. In *Robotics: Science and Systems*.

Reed, S. E.; Zolna, K.; Parisotto, E.; et al. 2022. A Generalist Agent. *Trans. Mach. Learn. Res.*, 2022.

Sadigh, D.; Dragan, A. D.; Sastry, S.; and Seshia, S. A. 2017. Active Preference-Based Learning of Reward Functions. In *Robotics: Science and Systems*.

Samvelyan, M.; Rashid, T.; de Witt, C. S.; et al. 2019. The StarCraft Multi-Agent Challenge. In *AAMAS*, 2186–2188.

Schenck, C.; and Fox, D. 2017. Visual closed-loop control for pouring liquids. In *ICRA*, 2629–2636.

Schulman, J.; Wolski, F.; Dhariwal, P.; Radford, A.; and Klimov, O. 2017. Proximal Policy Optimization Algorithms. *arXiv preprint arXiv:1707.06347*.

Silver, D.; Huang, A.; Maddison, C. J.; et al. 2016. Mastering the game of Go with deep neural networks and tree search. *Nature*, 529(7587): 484–489.

Skalse, J.; Howe, N. H. R.; Krasheninnikov, D.; and Krueger, D. 2022. Defining and Characterizing Reward Hacking. *arXiv preprint arXiv:2209.13085*.

Song, J.; Ren, H.; Sadigh, D.; and Ermon, S. 2018. Multi-Agent Generative Adversarial Imitation Learning. In *NeurIPS*, 7472–7483.

Stiennon, N.; Ouyang, L.; Wu, J.; Ziegler, D. M.; Lowe, R.; Voss, C.; Radford, A.; Amodei, D.; and Christiano, P. F. 2020. Learning to summarize with human feedback. In *NeurIPS*.

Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A. N.; Kaiser, L.; and Polosukhin, I. 2017. Attention is All you Need. In *NeurIPS*, 5998–6008.

Vinyals, O.; Babuschkin, I.; Czarnecki, W. M.; et al. 2019. Grandmaster level in StarCraft II using multi-agent reinforcement learning. *Nature*, 575(7782): 350–354.

Wang, H.; Yu, L.; Cao, Z.; and Ermon, S. 2021. Multi-agent Imitation Learning with Copulas. In *ECML/PKDD*, volume 12975, 139–156.

Wen, M.; Kuba, J. G.; Lin, R.; Zhang, W.; Wen, Y.; Wang, J.; and Yang, Y. 2022. Multi-Agent Reinforcement Learning is a Sequence Modeling Problem. In *NeurIPS*.

Wilson, A.; Fern, A.; and Tadepalli, P. 2012. A Bayesian Approach for Policy Learning from Trajectory Preference Queries. In *NeurIPS*, 1142–1150.

Wirth, C.; Akrour, R.; Neumann, G.; and Fürnkranz, J. 2017. A Survey of Preference-Based Reinforcement Learning Methods. *J. Mach. Learn. Res.*, 18: 136:1–136:46.

Yahya, A.; Li, A.; Kalakrishnan, M.; Chebotar, Y.; and Levine, S. 2017. Collective robot reinforcement learning with distributed asynchronous guided policy search. In *IROS*, 79–86.

Yu, C.; Velu, A.; Vinitsky, E.; Gao, J.; Wang, Y.; Bayen, A.; and Wu, Y. 2022. The Surprising Effectiveness of PPO in Cooperative Multi-Agent Games. In *NeurIPS*.

Zambaldi, V. F.; Raposo, D.; Santoro, A.; et al. 2019. Deep reinforcement learning with relational inductive biases. In *ICLR*.

Zhao, M.; Liu, F.; Lee, K.; and Abbeel, P. 2022. Towards More Generalizable One-shot Visual Imitation Learning. In *ICRA*, 2434–2444.

Zhou, M.; Liu, Z.; Sui, P.; Li, Y.; and Chung, Y. Y. 2020a. Learning Implicit Credit Assignment for Cooperative Multi-Agent Reinforcement Learning. In *NeurIPS*.

Zhou, M.; Luo, J.; Villela, J.; et al. 2020b. SMARTS: Scalable Multi-Agent Reinforcement Learning Training School for Autonomous Driving. *arXiv preprint arXiv:2010.09776*.