Episodic Multi-agent Reinforcement Learning with Curiosity-driven Exploration

Anonymous Author(s)
Affiliation
Address
email

Abstract

Efficient exploration in deep cooperative *multi-agent reinforcement learning* (MARL) still remains challenging in complex coordination problems. In this paper, we introduce a novel Episodic Multi-agent reinforcement learning with Curiosity-driven exploration algorithm, EMC. We leverage an insight of popular factorized MARL algorithms that the "induced" individual Q-values, i.e., the individual utility functions used for local execution, are the embeddings of local action-observation histories, and can capture the interaction between agents due to reward backpropagation during centralized training. Therefore, we use prediction errors of individual Q-values as intrinsic rewards for coordinated exploration and utilize episodic memory to exploit explored informative experience to boost policy training. As the dynamics of an agent's individual Q-value function captures the novelty of states and the influence from other agents, our intrinsic reward can induce coordinated exploration to new or promising states. We illustrate the advantages of our method by didactic examples, and demonstrate its significant outperformance over state-of-the-art MARL baselines on challenging tasks in the StarCraft II micromanagement benchmark.

1 Introduction

Cooperative *multi-agent reinforcement learning* (MARL) has great promise to solve many real-world multi-agent problems, such as autonomous cars [7] and robots [11]. These complex applications post two major challenges for cooperative MARL: *scalability*, i.e., the joint-action space exponentially grows as the number of agents increases, and *partial observability*, which requires agents to make decentralized decisions based on their local action-observation histories due to communication constraints. Recently, a popular MARL paradigm, called *centralized training with decentralized execution* (CTDE), is adopted to deal with these challenges. With this paradigm, agents' policies are trained with access to global information in a centralized way and executed only based on local histories in a decentralized way. Based on the paradigm of CTDE, many deep MARL methods have been proposed ,including VDN [31], QMIX [27], QTRAN [29], and QPLEX [35].

A core idea of these approaches is to use value factorization, which uses neural networks to represent the joint state-action value as a function of individual utility functions, which can be referred to *individial Q-values* for terminological simplicity. For example, VDN learns a centralized but factorizable joint value function Q_{tot} represented as the summation of individual value functions Q_i. During execution, the decentralized policies can be easily derived for each agent i by greedily selecting actions with respect to its local value function Q_i. By utilizing this factorization structure, an implicit multi-agent credit assignment is realized because Q_i is represented as a latent embedding and is learned by neural network backpropagation from the total temporal-difference error on the single global reward signal, rather than on a local reward signal specific to agent i. This value

Submitted to 35th Conference on Neural Information Processing Systems (NeurIPS 2021). Do not distribute.

factorization technique enables value-based MARL approaches, such as QMIX and QPLEX, to achieve state-of-the-art performance in challenging tasks such as the StarCraft unit micromanagement [28].

Despite current success, since only use simple ϵ-greedy exploration strategy, these deep MARL approaches are found ineffective to solve complex coordination tasks that require coordinated and efficient exploration [35]. Exploration has been extensively studied in single-agent reinforcement learning and many advanced methods have been proposed, including pseudo-counts [4, 21], curiosity [24, 6], and information gain [10]. However, these methods cannot be adopted into MARL directly, due to the exponentially growing state space and partial observability, leaving multi-agent exploration challenging. Recently, only a few works have tried to address the problem. For instance, EDTI [37] uses influence-based methods to quantify the value of agents' interactions and coordinate exploration towards high-value interactions. This approach empirically shows promising results but, because of the need to explicitly estimate the influence among agents, it is not scalable when the number of agents increases. Another method, called MAVEN [16], introduces a hierarchical control method with a shared latent variable encouraging committed, temporally extended exploration. However, since the latent variable still needs to explore in the space of joint behaviours [16], it is not efficient in complex tasks with large state spaces.

In this paper, we propose a novel multi-agent curiosity-driven exploration method. Curiosity is a type of intrinsic motivation for exploration, which uses prediction errors of future observations or states as a reward signal. Recently, curiosity-driven methods have achieved significant success in single-agent reinforcement learning [6, 2, 1]. However, curiosity-driven methods face a fundamental challenge in MARL: how to define curiosity? The straightforward solution is to measure curiosity by the novelty of global observation [6] or joint histories in a centralized way. However, it is inefficient to find locality interaction between agents, which seems too sparse compared with the exponentially growing state space when the number of agents increases. In contrast, if curiosity is defined as the novelty of local observation histories during the decentralized execution, although scalable, it still fails to guide agents to coordinate due to partial observability. Therefore, we find a middle point of centralized curiosity and decentralized curiosity, i.e., utilizing the value factorization of the state-of-the-art multi-agent Q-learning approaches and defining the prediction errors of individual Q-value functions as intrinsic rewards. The significance of this intrinsic reward is two-fold: 1) it provides a novelty measure of joint observation histories with scalability because individual Q-values are latent embeddings of observation histories in factorized multi-agent Q-learning (e.g., VDN or QPLEX) ; and 2) as shown in Figure 1, it captures the influence from other agents due to the implicit credit assignment from global reward signal during centralized training[34], and biases exploration into promising states where strong interdependence may lie between agents. Therefore, with this novel intrinsic reward, our curiosity-driven method enables efficient, diverse, and coordinated exploration for deep multi-agent Q-learning with value factorization.

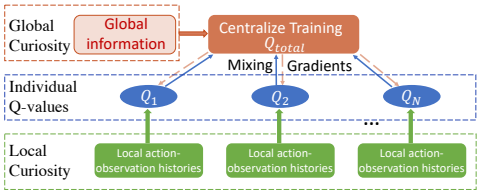


Figure 1: CTDE Framework

Besides efficient exploration, another challenge for deep MARL approaches is how to make the best use of experiences collected by the exploration strategy. Prioritized experience replay based on TD errors shows effectiveness in single-agent deep reinforcement learning. However, it does not carry this promise in factorized multi-agent Q-learning, since the projection error resulted by value factorization is also fused into the TD error and severally degrades the effectiveness of the TD error as a measure of the usefulness of experiences. To efficiently use promising exploratory experience trajectories, we augment factorized multi-agent reinforcement learning with episodic memory [15, 38]. This memory stores and regularly updates the best returns for explored states. We use the results in the episodic memory to regularize the TD loss, which allows fast latching onto past successful experience trajectories collected by curiosity-driven exploration and greatly improves learning efficiency. Therefore, we call our method Episodic Multi-agent reinforcement learning with Curiosity-driven exploration (EMC).

We evaluate EMC in didactic examples, and a broad set of StarCraft II micromanagement benchmark tasks [28]. The didactic examples along with detailed visualization illustrate that our proposed

intrinsic reward can guide agents' policies to novel or promising states, thus enables effectively coordinated exploration. Empirical results on more complicated StarCraft II tasks show that EMC significantly outperforms other multi-agent state-of-the-art baselines.

2 Background

2.1 Dec-POMDP

A cooperative multi-agent task can be modelled as a Dec-POMDP [19], which is defined by a tuple $G =< \mathcal{I}, \mathcal{S}, \mathcal{A}, P, R, \Omega, O, n, \gamma >$, where $\mathcal{I}$ is the sets of n agents, $\mathcal{S}$ is the global state space, $\mathcal{A}$ is the finite action set, $\gamma \in [0, 1)$ is the discount factor. We consider a partially observable setting in a Dec-POMDP, i.e., at each timestep, agent $i \in \mathcal{I}$ only has access to the observation $o_i \in \Omega$ drawn from the observation function $O(s, i)$. Besides, each agent has an action-observation history $\tau_i \in \mathcal{T} \equiv (\Omega \times \mathcal{A})^*$ and constructs its individual policy to jointly maximize team performance. With each agent i selecting an action $a_i \in \mathcal{A}$, the joint action $\boldsymbol{a} \equiv [a_i]_{i=1}^n \in \boldsymbol{\mathcal{A}} \equiv \mathcal{A}^N$ leads to a shared reward $r = R(s, \boldsymbol{a})$ and the next state s' according to the transition function $P(s'|s, \boldsymbol{a})$. The formal objective function is to find a joint policy $\boldsymbol{\pi}$ that maximizes a joint value function $V^{\boldsymbol{\pi}}(s) = \mathbb{E}[\sum_{t=0}^{\infty} \gamma^t r_t | s = s_0, \boldsymbol{\pi}]$, or a joint action-value function $Q^{\pi}(s, \boldsymbol{a}) = r(s, \boldsymbol{a}) + \gamma \mathbb{E}_{s'}[V^{\boldsymbol{\pi}}(s')]$.

2.2 Centralized Training With Decentralized Execution (CTDE)

CTDE is a promising paradigm in deep cooperative multi-agent reinforcement learning [18, 20], where the local agents execute actions only based on local observation histories, while the policies can be trained in centralized manager which has access to global information. During the training process, the whole team cooperate to find the optimal joint action-value function $Q^*_{tot}(s, \boldsymbol{a}) = r(s, \boldsymbol{a}) + \gamma \mathbb{E}_{s'}[max_{\boldsymbol{a'}} Q^*_{tot}(s', \boldsymbol{a'})]$. Due to partial observability, we use $Q_{tot}(\boldsymbol{\tau}, \boldsymbol{a}; \boldsymbol{\theta})$ instead of $Q_{tot}(s, \boldsymbol{a}; \boldsymbol{\theta})$, where $\boldsymbol{\tau} \in \boldsymbol{\mathcal{T}} \equiv \mathcal{T}^N$. Then the Q-value neural network will be trained to minimize the following expected TD-error:

$$\mathcal{L}(\boldsymbol{\theta}) = \mathbb{E}_{\boldsymbol{\tau}, \boldsymbol{a}, \boldsymbol{r}, \boldsymbol{\tau'} \in D} \left[r + \gamma V(\boldsymbol{\tau'}; \boldsymbol{\theta}^-) - Q_{tot}(\boldsymbol{\tau}, \boldsymbol{a}; \boldsymbol{\theta}) \right]^2, \tag{1}$$

where D is the replay buffer and $\boldsymbol{\theta}^-$ denotes the parameters of the target network, which is periodically updated by $\boldsymbol{\theta}$. And $V(\boldsymbol{\tau'}; \boldsymbol{\theta}^-)$ is the one-step expected future return of the TD target. Local agents can only obtain local action-observation history and need inference based on individual Q-value functions $Q_i(\tau_i, a_i)$. Therefore, many works have made efforts in finding the factorization structures between joint Q-value functions Q_{tot} and individual Q-functions $Q_i(\tau_i, a_i)$ [27, 35, 31] and attracted great attention.

3 Related Work

Curiosity-driven Exploration Curiosity-driven exploration has been well studied in single-agent reinforcement learning. Preivious literature [22, 23] has provided a good summary in this topic. Recently, curiosity-driven methods have made great progress in deep reinforcement learning. For example, some works use pseudo-state counts to get intrinsic rewards [4, 21, 33] instead of count-based methods to get better scalability. [30] uses prediction errors in the feature space of an auto-encoder to measure the novelty of states and encourage exploration. On the other hand, [17] proposes to use empowerment, measured by the information gain based on the entropy of actions, as intrinsic rewards for exploring novel states efficiently. Another information-based method [10], tries to maximize information gain about the agent's belief of the environment's dynamics as an exploration strategy. ICM [24] learns an inverse model which predicts the agent's action given its current and next states and tries to predict the next state in the learned hidden space by current state and action. RND [6] uses curiosity as intrinsic rewards in a simpler but effective way, which uses a fixed randomly initialized neural network as a representation network and directly predicts the embedding of the next state.

Multi-agent Exploration Although single-agent exploration is extensively studied and has achieved considerable success, few exploration methods were designed for cooperative MARL. [3] proposes an exploration method which can only be used in repeated single-stage problems. [13] defines intrinsic reward by "social influence" to encourage agents to choose actions that can influence other

agents' actions. [12] uses various simple exploration methods to learn simultaneously and then put the samples of every method in a shared buffer to achieve the coordinated exploration. [37] uses mutual information (MI) to capture the interdependence of the rewards and transitions between agents. MAVEN [16] is the state-of-the-art exploration method in MARL which uses a hierarchical policy to produce a shared latent variable and learns several state-action value functions for each agent. These works, although important, still face the challenge of effective and scalable multi-agent exploration.

Episodic Control Our work is also closely related to episodic control reinforcement learning, which is usually adopted in single-agent settings for better sample efficiency. Previous works propose to use episodic memory in near-deterministic environment[14, 5, 25, 9]. Model-free episodic control [5] uses a completely non-parametric table to keep the best Q-values of state-action pair in a tabular-based memory and uses a k-nearest-neighbors fashion to find the sequence of actions that so far yielded the highest return from a given start state in the memory. Recently, several extensions have been proposed to integrate episodic control with parametric DQN. [8] uses episodic memory to retrieve samples and then average future returns to approximate the action values. EMDQN [15] uses a fixed random matrix as a representation function and uses the projection of states as keys to store the information of episodic memory into a non-parametric model. Using the episodic-memory based target as a regularization term to guide the training process, the performance of EMDQN is significantly improved compared with original DQN. Despite the fruitful progress made in single-agent episodic reinforcement learning, few works study episodic control in a multi-agent setting. To the best of our knowledge, we are the first to utilize the mechanism of episodic control in deep multi-agent reinforcement learning.

4 Episodic Multi-agent Reinforcement Learning with Curiosity-Driven Exploration

In this section, we introduce EMC, a novel episodic multi-agent exploration framework. EMC takes prediction errors of individual Q-value functions as intrinsic rewards for guiding the diverse and coordinated exploration. After collecting informative experience, we leverage an episodic memory to memorize the highly rewarding sequences and use it as the reference of one-step TD target to boost multi-agent Q-learning. First, we analyze the motivations for predicting individual Q-values, then we introduce the curiosity module for exploration. Finally, we describe how to utilize episodic memory to boost training.

4.1 Curiosity-Driven Exploration by Predicting Individual Q-values

As shown in Figure 2, in the paradigm of CDTE, local agents make decisions based on individual Q-value functions, which take local observation histories as inputs, and are updated by the centralized module which has access to global information for training. The key insight is that, different from single-agent cases, individual Q-value functions in MARL are used for both decision-making and embedding historical observations. Furthermore, due to implicit credit assignment by global reward signal during centralized training, individual Q-value functions $Q_i(\tau_i, \cdot)$ are influenced by environment as well as other agents' behaviors. More concretely, it has been proved that [34], when the joint Q-function Q_{tot} is factorized into linear combination of individual Q-functions Q_i, i.e.,$Q_{\text{tot}}^{(t+1)}(\boldsymbol{\tau}, \boldsymbol{a}) = \sum_{i=1}^{N} Q_i^{(t+1)}(\tau_i, a_i)$, then $Q_i^{(t+1)}(\tau_i, a_i)$ has the following closed-form solution:

$$
\begin{aligned}
Q_i^{(t+1)}(\tau_i, a_i) = & \underbrace{\mathop{\mathbb{E}}_{(\tau'_{-i}, a'_{-i}) \sim p_D(\cdot|\tau_i)} \left[y^{(t)} \left(\tau_i \oplus \tau'_{-i}, a_i \oplus a'_{-i} \right) \right]}_{\text{evaluation of the individual action } a_i} \\
& - \frac{n-1}{n} \underbrace{\mathop{\mathbb{E}}_{\boldsymbol{\tau}', \mathbf{a}' \sim p_D(\cdot|\Lambda^{-1}(\tau_i))} \left[y^{(t)} \left(\boldsymbol{\tau}', \mathbf{a}' \right) \right]}_{\text{counterfactual baseline}} + w_i(\tau_i),
\end{aligned}
\tag{2}
$$

where $y^{(t)}(\boldsymbol{\tau}, \mathbf{a}) = r + \gamma \mathbb{E}_{\boldsymbol{\tau}'} \left[\max_{\mathbf{a}'} Q_{\text{tot}}^{(t)} \left(\boldsymbol{\tau}', \mathbf{a}' \right) \right]$ denotes the expected one-step TD target, and $p_D(\cdot|\tau_i)$ denotes the conditional empirical probability of τ_i in the given dataset D. The notation $x_i \oplus x'_{-i}$ denotes $\langle x'_1, \ldots, x'_{i-1}, x_i, x'_{i+1}, \ldots, x'_n \rangle$, and x'_{-i} denotes the elements of all agents except for agent i. $\Lambda^{-1}(\tau_i)$ denotes the set of trajectory histories that may share the same latent-

state trajectory as τ_i. The residue term $\mathbf{w} \equiv [w_i]_{i=1}^n$ is an arbitrary function satisfying $\forall \boldsymbol{\tau} \in \mathbf{\Gamma}$, $\sum_{i=1}^n w_i(\tau_i) = 0$.

Eq. 2 shows that by *linear value factorization*, the individual Q-value $Q_i(\tau_i, a_i)$ is not only decided by local observation histories but also influenced by other agents' action-observation histories. Thus predicting Q_i can capture both the novelty of states and the interaction between agents and lead agents to explore promising states. Motivated by this, in this paper, we use a linear value factorization module separate from the inference module to learn the individual Q-values Q_i, and use the orediction errors of Q_i as intrinsic rewards to guide exploration. In this paper, we define the prediction errors of individual Q-values as *curiosity* and propose our curiosity-driven exploration module.

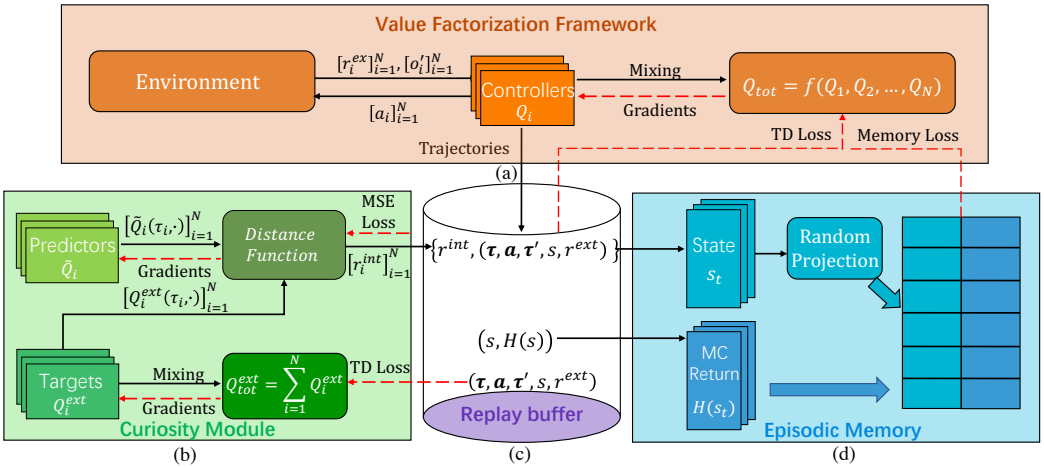


Figure 2: An overview of EMC's framework

Figure 2 (b) demonstrates the *Curiosity Module*, separated from the inference module (Figure 2 (a)). The curiosity module consists of four components: (i) The centralized training part with linear value factorization, which shares the same implementation as VDN [31], but only trained with extrinsic rewards r^{ext} from the environment; (ii) the *Target* for prediction, i.e., the corresponding local Q-values Q_i^{ext}, represented by a recurrent Q-network; (iii) *Predictor* $\widetilde{Q}_i(\tau_i)$, which shares the same network architecture as *Target* Q_i^{ext}; and (iv) *Distance Function*, which measures the distance between Q_i^{ext} and $\widetilde{Q}_i$, e.g., L_2 distance. The predictors are trained by minimizing the Mean Squared Error (MSE) of the distance in an end-to-end manner. The curiosity module predicts the individual Q-values $[Q_i^{ext}]_{i=1}^n$ which are linear factorization of the joint Q-value Q_{tot}^{ext}, i.e., $Q_{tot}^{ext} = \sum_{i=1}^N Q_i^{ext}$, thus matches Eq. 2. Then the curiosity-driven intrinsic reward is generated by the following equation:

$$r^{int} = \frac{1}{N} \sum_{i=1}^{N} \left\| \widetilde{Q}_i(\tau_i, \cdot) - Q_i^{ext}(\tau_i, \cdot) \right\|_2^2, \tag{3}$$

This intrinsic reward is used for the centralized training of the inference module, as shown in Figure 2 (a):

$$\mathcal{L}_{\text{inference}}(\boldsymbol{\theta}) = \mathbb{E}_{\boldsymbol{\tau}, \boldsymbol{a}, \boldsymbol{r}, \boldsymbol{\tau}' \in D} \left[S - Q_{tot}^{all}(s, \boldsymbol{a}; \boldsymbol{\theta}) \right]^2, \tag{4}$$

where $S = r^{ext} + \beta r^{int} + \gamma \max_{\mathbf{a}'} Q_{\text{tot}}(\boldsymbol{\tau}', \mathbf{a}'; \boldsymbol{\theta}^-))$, denoting one step TD target of the inference module, and β is the weight term of the intrinsic reward. For stable training, we use the trick of soft-update to smooth the outputs of the targets and set a decay rate for the weight term β. We use a separate training model for inference (Figure 2 (a)) to avoid the accumulation of projection errors of Q_i during training. The independence of inference module leads to another advantage, that EMC's architecture can be adopted into many value-factorization-based multi-agent algorithms which utilize the CDTE paradigm, such as VDN, QMIX, or QPLEX. In this paper, we utilize the state-of-the-art algorithm QPLEX [35] for inference unless otherwise mentioned. With this curiosity-driven bias plugged into ordinary MARL algorithms, coordinated exploration can be achieved efficiently.

4.2 Episodic Memory

Equipped with efficient exploration ability, another challenge is how to use the best use of good trajectories collected by exploration effectively. Recently, episodic control has become very popular in single-agent reinforcement learning [15, 38], which can replay the highly rewarding sequences, thus boost training. Inspires by this, we generalize single-agent episodic control to propose a multi-agent episodic memory, which records the best memorized Monte-Carlo return, and provide a memory target H as a reference to regularize the ordinary one-step inference TD target estimation in the inference module (Figure 2 (a)):

$$\mathcal{L}_{\text{memory}}(\theta) = \mathbb{E}_{\boldsymbol{\tau},\boldsymbol{a},\boldsymbol{r},\boldsymbol{\tau'}\in D}\left[\left(H - Q_{tot}(s,\boldsymbol{a};\boldsymbol{\theta})\right)^2\right]. \tag{5}$$

However, different from the single-agent episodic control, the action space of MARL exponentially grows as the number of agents increases, and partial observability also limits the information of individual value functions. Thus, we maintain our episodic memory by storing the state-value function on the global state space and utilize the global information during the centralized training process under the CTDE paradigm. Figure 2 (d) shows the architecture of the *Episodic Memory*. We keep a memory table M to record the maximum remembered return of the current state, and use a fixed random matrix drawn from Gaussian distribution as representation function to project states into low-dimensional vectors $\phi(s) : S \to \mathbb{R}^k$, which are used as keys to look up corresponding global state value function $H(\phi(s_t))$. When our exploration method collects a new trajectory, we update our memory table M as follows:

$$H(\phi(s_t)) = \begin{cases} \max\{H(\phi(s_t)), R_t(s_t,\boldsymbol{a}_t)\} & \textit{if } H(\phi(s_t)) \in M \\ R_t(s_t,\boldsymbol{a}_t) & \textit{otherwise.} \end{cases} \tag{6}$$

where $R(s_t,\boldsymbol{a}_t)$ represents the future return when agents taking joint action $\boldsymbol{a}_t$ under global state s_t at the t-th timestep in a new episode. Thanks to this episodic memory, we can directly obtain the maximum remembered return of the current state, and use the one-step TD memory target H as a reference to regularize learning:

$$H(\phi(s_t),\boldsymbol{a}_t) = r_t(s_t,\boldsymbol{a}_t) + \gamma H(\phi(s_{t+1})). \tag{7}$$

Thus, the new objective function for the inference module is:

$$\begin{aligned}\mathcal{L}_{\text{total}}(\theta) &= \mathcal{L}_{\text{inference}}(\theta) + \lambda\mathcal{L}_{\text{memory}}(\theta) \\ &= \mathbb{E}_{\boldsymbol{\tau},\boldsymbol{a},\boldsymbol{r},\boldsymbol{\tau'}\in D}\left[\left(S(s_t,\boldsymbol{a}_t) - Q_{tot}(s_t,\boldsymbol{a}_t;\boldsymbol{\theta})\right)^2 + \lambda\left(H\left(\phi(s_t),\boldsymbol{a}_t\right) - Q_{tot}(s,\boldsymbol{a};\boldsymbol{\theta})\right)^2\right],\end{aligned} \tag{8}$$

where λ is the weighting term to balance the effect of episodic memory's reference. Using the maximum return from the episodic memory to propagate rewards, we can compensate the disadvantage of slow-learning resulted by original one-step reward update and improve sample efficiency.

5 Experiments

In this section, we will analyse experiments results designed for answering the following questions: (1) Is exploration by predicting individual Q-value functions better than exploration by decentralized curiosity or global curiosity? (Section 5.1) (2) Can our method perform efficient coordinated exploration in challenging multi-agent tasks? (Section 5.2-5.3) (3) if so, what role does each key component play in the outperformance? (Section 5.4) We propose a didactic example and demonstrate the advantage of our method in coordinated exploration, and evaluate our method on the StarCraft II micromanagement (SMAC) benchmark [28] compared with existing state-of-the-art multi-agent reinforcement learning (MARL) algorithms: QPLEX [35], Weighted-QMIX[26],QTRAN[29], QMIX[27], VDN[31], RODE[36], and multi-agent exploration method MAVEN[16].

5.1 Didactic Example

Figure 3 shows an 11x12 grid world game that requires coordinated exploration. The blue agent and the red agent can choose one of the five actions: *[up, down, left, right, stay]* at each time step. The wall shown in the picture isolates the two agents, and one agent cannot be observed by the other until it gets into the shaded area. The two agents will receive a positive global reward $r = 10$ if and only if they arrive at the corresponding goal grid at the same time. If only one arrives, the incoordination will be punished by a negative reward $-p$.

To evaluate the effectiveness of our curiosity-driven exploration, we implement our method into QPLEX QMIX, and VDN (denoted as EMC-QPLEX, EMC-QMIX, EMC-VDN) and test them in this toy game compared with the state-of-the-art MARL algorithms: VDN[31], IQL [32], QMIX[27], and QPLEX[35]. Moreover, to demonstrate the motivation of predicting individual Q-functions, we add two more baselines: QPLEX with the prediction error of global state as intrinsic rewards (denoted as QPLEX-Global), and QPLEX with the prediction error of local joint histories as intrinsic rewards (denoted as QPLEX-Local). Both of them use a fixed network to project the inputs into latent embedding, then predict the latent embedding to generate intrinsic reward, just like the Random Network Distillation[6] (RND). We test different punishment degrees, i.e., different p (Please see Appendix C.), and the results show QPLEX-Global and QPLEX-Local are effective enough for exploration when p is relatively small. However, as p increases, the task becomes more challenging since it requires sufficient and coordinated exploration. In figure 4, we show the median test win rate of all methods over 6 random seeds when $p = 2$, and only our methods can learn the optimal policy and win the game, while other methods failed.

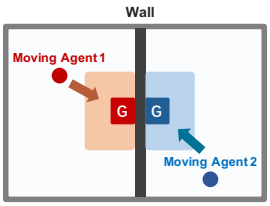


Figure 3: Coordinated Toygame

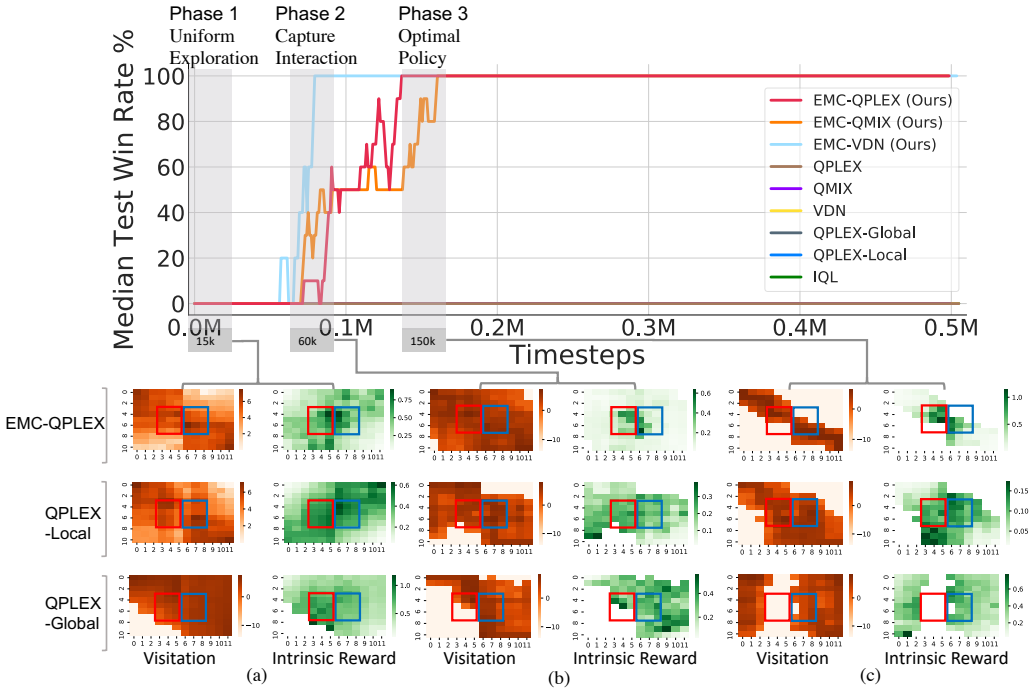


Figure 4: The heat map of gridworld game

To understand this result better, we have made several visualisations to demonstrate our advantage in coordinated exploration. Figure 4 shows the heatmaps of visitation and intrinsic reward by EMC-QPLEX, QPLEX-Global, and QPLEX-Local. During the early stage of training, all methods uniformly explore the whole area (Figure 4 (a)). As the exploration progresses, the global curiosity (QPLEX-Global) encourages agents to visiting all configurations without bias, which is inefficient and fail to leverage the potential locality influence between agents (Figure 4 (b)), resulting in extrinsic rewards beginning to dominate the behaviors (Figure4 (c)). On the other hand, the visitation heatmap of QPLEX-Local shows the decentralized curiosity encourages agents to explore around the goal grid, but it cannot promise encouraging agents to coordinate and gain the reward due to the partial observability in decentralized execution. In contrast, the heatmap of intrinsic reward for EMC-QPLEX shows that predicting individual Q-values will bias exploration into areas where individual Q-values are more dynamic due to the potential correlation between agents. Therefore, QPLEX-Local and QPLEX-Global both fail in this task (Figure 4 (c)), while our methods are able to find the optimal policy. This didactic example shows the global curiosity or local curiosity may fail to handle complex tasks where coordinated exploration need to be addressed. While since individual Q-values Q_i are the

embeddings of historical observations, and are dynamically updated by the backpropagation of the global reward signal gained through cooperation during centralized training. Thus Q_i can implicitly reflect the influence from environment and other agents, and predicting Q_i can capture valuable and spare interactions among agents and bias exploration into new or promising states.

5.2 Predator Prey

Predator Prey is a partially-observable multi-agent coordinated game with miscoordination penalties used by WQMIX [26]. As shown in Figure 5, since extensive exploration is needed to jump out of the local optima, WQMIX is the only baseline algorithm to find optimal policy, due to its shaped data distribution which can be seen as a type of exploration. Other state-of-the-art multi-agent Q-learning algorithms, such as QPLEX and QMIX, fail to solve this task. For QPLEX-Local and QPLEC-Global, although equipped with improved exploration ability, they still failed to address coordination due to uniform exploration nature or partial observability. However, plugged with EMC, EMC-VDN, EMC-QMIX, and EMC-QPLEX can guarantee coordinated exploration effectively and achieve good performance.

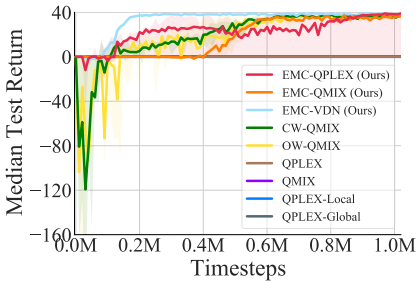


Figure 5: Predator Prey

5.3 StarCraftII Micromanagement (SMAC) Benchmark

StarCraftII Micromanagement (SMAC) is a popular benchmark in MARL[31, 27, 36, 26, 35]. We conduct experiments in 17 benchmark tasks of StarCraft II, which contains 14 popular tasks proposed by SMAC [28] and 3 more super hard cooperative tasks proposed by QPLEX [35]. In the micromanagement scenarios, each unit is controlled by an independent agent that must act based on its own local observation, and the enemy units are controlled by a built-in AI.

For evaluation, we compare EMC with the state-of-the-art algorithms: RODE [36], QPLEX [35], MAVEN [16], and the two variants of QMIX [27]: CW-QMIX and OW-QMIX [26]. All experimental results are illustrated with the median performance and 25-75% percentiles. Figure 6 shows the overall performance of the tested algorithms in all these 17 maps. Due to the effective exploration with episodic memory which can efficiently use promising exploratory experience trajectories, EMC is the best performer on up to 6 tasks, underperforms on just 3 tasks, and ties for the best performer on the rest tasks.

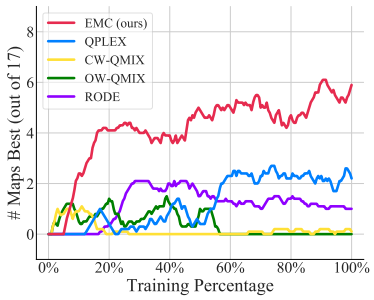
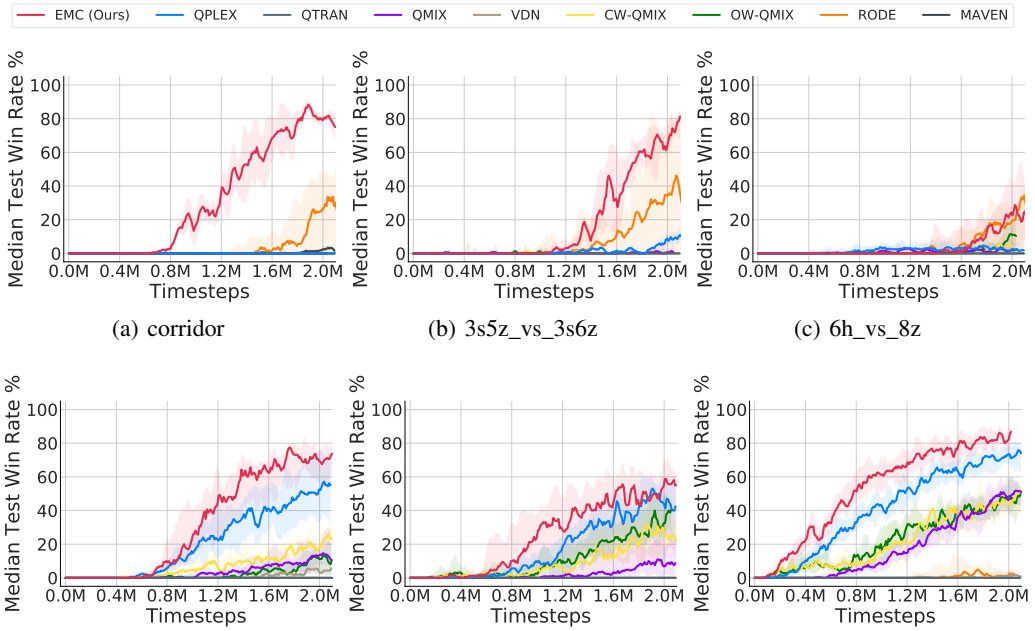


Figure 6: The number of scenarios (out of 17 scenarios) in which the algorithm's median test win rate is the highest by as least 1/32.

The advantages of our algorithm can be mainly illustrated by the results of the 6 hard maps which need sufficient exploration shown in Figure 7. The three maps in the first row are super hard, and solving them particularly needs efficient, coordinated exploration. Thus, we can find that the EMC algorithm significantly outperforms other algorithms in *corridor* and *3s5z_vs_3s6z*, and also achieves the best performance (equal to RODE) in *6h_vs_8z*. To the best of our knowledge, this may be the state-of-the-art results in *corridor* and *3s5z_vs_3s6z*. For the remaining three maps in the second row (*1c3s8z_vs_1c3s9z*, *5s10z*, and *7s7z*), where other baselines can also find winning strategies, due to the boost learning process via episodic memory along with efficient exploration, our algorithm EMC still performs the best in the three maps, with fastest learning speed and the highest rates achieved.

5.4 Ablation Study

To understand the superior performance of EMC, we carry out ablation studies to test the contribution of its two main components: curiosity module and episodic memory. Following methods are included in the evaluation: (i) EMC without curiosity module (EMC-wo-C); (ii) EMC without episodic memory component (EMC-wo-M); (iii) QPLEX, which can be considered as EMC without the episodic memory component nor the curiosity module, provides a natural ablation baseline of EMC.

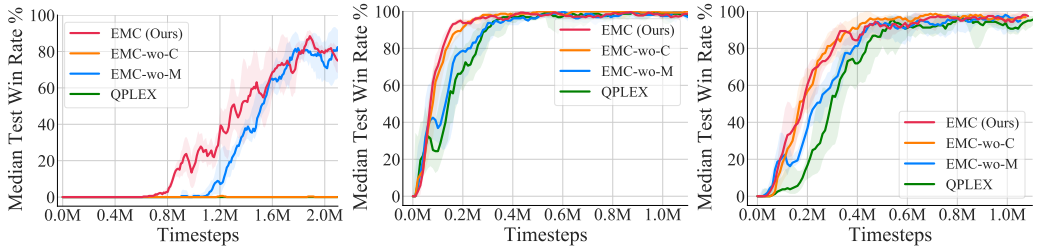


(a) corridor (b) 3s5z_vs_3s6z (c) 6h_vs_8z

(d) 1c3s8z_vs_1c3s9z (e) 5s10z (f) 7s7z

Figure 7: Results of super hard maps in SMAC

Figure 8 (b-c) shows that in easy exploration maps, both EMC and EMC-wo-C achieve the state-of-the-art performance, which implies that in the easy tasks, sufficient exploration can be achieved simply by the popular ϵ-greedy method. However, in super hard exploration maps (Figure 8 (a)), EMC-wo-C cannot solve this task but EMC has excellent performance. These empirical experiments show that the curiosity module plays a vital role in improving performance when sufficient and coordinated exploration is necessary. On the other hand, making the best use of good trajectories collected by exploration is also essential. As shown Figure 8, EMC with episodic memory enjoys better sample efficiency than EMC-wo-M in challenging (Figure 8 (a)) and easy exploration tasks (Figure 8 (b-c)). In general, the curiosity module and the episodic memory complement each other, and efficiently using promising exploratory experience trajectories leads to the outperformance of EMC.



(a) corridor (b) 2s3z (c) 3s5z

Figure 8: Ablation Study

6 Conclusions and Future Work

This paper introduces EMC, a novel episodic multi-agent reinforcement learning algorithm with a curiosity-driven exploration framework that allows for efficient coordinated exploration and boosted policy training by exploiting explored informative experiences. Based on the effective exploration ability, our method shows significant outperformance over state-of-the-art MARL baselines on challenging tasks in the StarCraft II micromanagement benchmark. The limitation of our work lies in the lack of adaptive exploration methods to ensure robustness. Besides, the episodic memory may result in local optimal policies, which contributes to EMC's underperformance in several maps (see Appendix B.). For future work, we may conduct further research in these directions.

References

[1] Adrià Puigdomènech Badia, Bilal Piot, Steven Kapturowski, Pablo Sprechmann, Alex Vitvitskyi, Zhaohan Daniel Guo, and Charles Blundell. Agent57: Outperforming the atari human benchmark. In *International Conference on Machine Learning*, pages 507–517. PMLR, 2020.

[2] Adrià Puigdomènech Badia, Pablo Sprechmann, Alex Vitvitskyi, Daniel Guo, Bilal Piot, Steven Kapturowski, Olivier Tieleman, Martin Arjovsky, Alexander Pritzel, Andrew Bolt, et al. Never give up: Learning directed exploration strategies. In *International Conference on Learning Representations*, 2019.

[3] Eugenio Bargiacchi, Timothy Verstraeten, Diederik Roijers, Ann Nowé, and Hado Hasselt. Learning to coordinate with coordination graphs in repeated single-stage multi-agent decision problems. In *International conference on machine learning*, pages 482–490, 2018.

[4] Marc Bellemare, Sriram Srinivasan, Georg Ostrovski, Tom Schaul, David Saxton, and Remi Munos. Unifying count-based exploration and intrinsic motivation. In *Advances in neural information processing systems*, pages 1471–1479, 2016.

[5] Charles Blundell, Benigno Uria, Alexander Pritzel, Yazhe Li, Avraham Ruderman, Joel Z Leibo, Jack Rae, Daan Wierstra, and Demis Hassabis. Model-free episodic control. *arXiv preprint arXiv:1606.04460*, 2016.

[6] Yuri Burda, Harrison Edwards, Amos Storkey, and Oleg Klimov. Exploration by random network distillation. *arXiv preprint arXiv:1810.12894*, 2018.

[7] Yongcan Cao, Wenwu Yu, Wei Ren, and Guanrong Chen. An overview of recent progress in the study of distributed multi-agent coordination. *IEEE Transactions on Industrial informatics*, 9(1):427–438, 2012.

[8] Samuel J Gershman and Nathaniel D Daw. Reinforcement learning and episodic memory in humans and animals: an integrative framework. *Annual review of psychology*, 68:101–128, 2017.

[9] Steven S Hansen, Pablo Sprechmann, Alexander Pritzel, André Barreto, and Charles Blundell. Fast deep reinforcement learning using online adjustments from the past. In *Proceedings of the 32nd International Conference on Neural Information Processing Systems*, pages 10590–10600, 2018.

[10] Rein Houthooft, Xi Chen, Yan Duan, John Schulman, Filip De Turck, and Pieter Abbeel. Vime: Variational information maximizing exploration. In *Advances in Neural Information Processing Systems*, pages 1109–1117, 2016.

[11] Maximilian Hüttenrauch, Adrian Šošić, and Gerhard Neumann. Guided deep reinforcement learning for swarm systems. *arXiv preprint arXiv:1709.06011*, 2017.

[12] Shariq Iqbal and Fei Sha. Coordinated exploration via intrinsic rewards for multi-agent reinforcement learning. *arXiv preprint arXiv:1905.12127*, 2019.

[13] Natasha Jaques, Angeliki Lazaridou, Edward Hughes, Caglar Gulcehre, Pedro Ortega, DJ Strouse, Joel Z Leibo, and Nando De Freitas. Social influence as intrinsic motivation for multi-agent deep reinforcement learning. In *International Conference on Machine Learning*, pages 3040–3049. PMLR, 2019.

[14] M Lengyel and P Dayan. Hippocampal contributions to control: The third way. In *Twenty-First Annual Conference on Neural Information Processing Systems (NIPS 2007)*, pages 889–896. Curran, 2008.

[15] Zichuan Lin, Tianqi Zhao, Guangwen Yang, and Lintao Zhang. Episodic memory deep q-networks. In *IJCAI*, 2018.

[16] Anuj Mahajan, Tabish Rashid, Mikayel Samvelyan, and Shimon Whiteson. Maven: Multi-agent variational exploration. In *Advances in Neural Information Processing Systems*, pages 7613–7624, 2019.

[17] Shakir Mohamed and Danilo J Rezende. Variational information maximisation for intrinsically motivated reinforcement learning. In *Proceedings of the 28th International Conference on Neural Information Processing Systems-Volume 2*, pages 2125–2133, 2015.

[18] Frans A Oliehoek and Christopher Amato. *A concise introduction to decentralized POMDPs*. Springer, 2016.

[19] Frans A Oliehoek, Christopher Amato, et al. *A concise introduction to decentralized POMDPs*, volume 1. Springer, 2016.

[20] Frans A Oliehoek, Matthijs TJ Spaan, and Nikos Vlassis. Optimal and approximate q-value functions for decentralized pomdps. *Journal of Artificial Intelligence Research*, 32:289–353, 2008.

[21] Georg Ostrovski, Marc G Bellemare, Aäron Oord, and Rémi Munos. Count-based exploration with neural density models. In *International Conference on Machine Learning*, pages 2721–2730, 2017.

[22] Pierre-Yves Oudeyer, Frdric Kaplan, and Verena V Hafner. Intrinsic motivation systems for autonomous mental development. *IEEE transactions on evolutionary computation*, 11(2):265–286, 2007.

[23] Pierre-Yves Oudeyer and Frederic Kaplan. What is intrinsic motivation? a typology of computational approaches. *Frontiers in neurorobotics*, 1:6, 2009.

[24] Deepak Pathak, Pulkit Agrawal, Alexei A Efros, and Trevor Darrell. Curiosity-driven exploration by self-supervised prediction. In *International Conference on Machine Learning*, 2017.

[25] Alexander Pritzel, Benigno Uria, Sriram Srinivasan, Adria Puigdomenech Badia, Oriol Vinyals, Demis Hassabis, Daan Wierstra, and Charles Blundell. Neural episodic control. In *International Conference on Machine Learning*, pages 2827–2836. PMLR, 2017.

[26] Tabish Rashid, Gregory Farquhar, Bei Peng, and Shimon Whiteson. Weighted QMIX: expanding monotonic value function factorisation for deep multi-agent reinforcement learning. In *Proceedings of the Annual Conference on Neural Information Processing Systems 2020 (NeurIPS 2020)*, 2020.

[27] Tabish Rashid, Mikayel Samvelyan, Christian Schroeder, Gregory Farquhar, Jakob Foerster, and Shimon Whiteson. Qmix: Monotonic value function factorisation for deep multi-agent reinforcement learning. In *International Conference on Machine Learning*, pages 4295–4304, 2018.

[28] Mikayel Samvelyan, Tabish Rashid, Christian Schroeder de Witt, Gregory Farquhar, Nantas Nardelli, Tim GJ Rudner, Chia-Man Hung, Philip HS Torr, Jakob Foerster, and Shimon Whiteson. The starcraft multi-agent challenge. In *Proceedings of the 18th International Conference on Autonomous Agents and MultiAgent Systems*, pages 2186–2188, 2019.

[29] Kyunghwan Son, Daewoo Kim, Wan Ju Kang, David Earl Hostallero, and Yung Yi. Qtran: Learning to factorize with transformation for cooperative multi-agent reinforcement learning. *arXiv preprint arXiv:1905.05408*, 2019.

[30] Bradly C Stadie, Sergey Levine, and Pieter Abbeel. Incentivizing exploration in reinforcement learning with deep predictive models. *arXiv preprint arXiv:1507.00814*, 2015.

[31] Peter Sunehag, Guy Lever, Audrunas Gruslys, Wojciech Marian Czarnecki, Vinícius Flores Zambaldi, Max Jaderberg, Marc Lanctot, Nicolas Sonnerat, Joel Z Leibo, Karl Tuyls, et al. Value-decomposition networks for cooperative multi-agent learning based on team reward. In *AAMAS*, pages 2085–2087, 2018.

[32] Ming Tan. Multi-agent reinforcement learning: Independent vs. cooperative agents. In *Proceedings of the tenth international conference on machine learning*, pages 330–337, 1993.

[33] Haoran Tang, Rein Houthooft, Davis Foote, Adam Stooke, OpenAI Xi Chen, Yan Duan, John Schulman, Filip DeTurck, and Pieter Abbeel. # exploration: A study of count-based exploration for deep reinforcement learning. In *Advances in neural information processing systems*, pages 2753–2762, 2017.

[34] Jianhao Wang, Zhizhou Ren, Beining Han, Jianing Ye, and Chongjie Zhang. Towards understanding linear value decomposition in cooperative multi-agent q-learning. *arXiv preprint arXiv:2006.00587*, 2020.

[35] Jianhao Wang, Zhizhou Ren, Terry Liu, Yang Yu, and Chongjie Zhang. Qplex: Duplex dueling multi-agent q-learning. *arXiv preprint arXiv:2008.01062*, 2020.

[36] Tonghan Wang, Tarun Gupta, Anuj Mahajan, Bei Peng, Shimon Whiteson, and Chongjie Zhang. Rode: Learning roles to decompose multi-agent tasks. *arXiv preprint arXiv:2010.01523*, 2020.

[37] Tonghan Wang, Jianhao Wang, Yi Wu, and Chongjie Zhang. Influence-based multi-agent exploration. In *International Conference on Learning Representations*, 2019.

[38] Guangxiang Zhu, Zichuan Lin, Guangwen Yang, and Chongjie Zhang. Episodic reinforcement learning with associative memory. In *International Conference on Learning Representations*, 2019.

Checklist

The checklist follows the references. Please read the checklist guidelines carefully for information on how to answer these questions. For each question, change the default **[TODO]** to [Yes] , [No] , or [N/A] . You are strongly encouraged to include a **justification to your answer**, either by referencing the appropriate section of your paper or providing a brief inline description. For example:

- Did you include the license to the code and datasets? [Yes] See Section 1
- Did you include the license to the code and datasets? [No] The code and the data are proprietary.
- Did you include the license to the code and datasets? [N/A]

Please do not modify the questions and only use the provided macros for your answers. Note that the Checklist section does not count towards the page limit. In your paper, please delete this instructions block and only keep the Checklist section heading above along with the questions/answers below.

1. For all authors...
   (a) Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope? [Yes]
   (b) Did you describe the limitations of your work? [Yes] see Section 6, Conclusions and Future Work.
   (c) Did you discuss any potential negative societal impacts of your work? [N/A]
   (d) Have you read the ethics review guidelines and ensured that your paper conforms to them? [Yes]
2. If you are including theoretical results...
   (a) Did you state the full set of assumptions of all theoretical results? [N/A]
   (b) Did you include complete proofs of all theoretical results? [N/A]
3. If you ran experiments...
   (a) Did you include the code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL)? [Yes] see supplemental material
   (b) Did you specify all the training details (e.g., data splits, hyperparameters, how they were chosen)? [Yes] see Appendix
   (c) Did you report error bars (e.g., with respect to the random seed after running experiments multiple times)? [Yes]

(d) Did you include the total amount of compute and the type of resources used (e.g., type of GPUs, internal cluster, or cloud provider)? [Yes] see Appendix.

4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets...
   (a) If your work uses existing assets, did you cite the creators? [Yes]
   (b) Did you mention the license of the assets? [Yes]
   (c) Did you include any new assets either in the supplemental material or as a URL? [N/A]
   (d) Did you discuss whether and how consent was obtained from people whose data you're using/curating? [N/A]
   (e) Did you discuss whether the data you are using/curating contains personally identifiable information or offensive content? [N/A]
5. If you used crowdsourcing or conducted research with human subjects...
   (a) Did you include the full text of instructions given to participants and screenshots, if applicable? [N/A]
   (b) Did you describe any potential participant risks, with links to Institutional Review Board (IRB) approvals, if applicable? [N/A]
   (c) Did you include the estimated hourly wage paid to participants and the total amount spent on participant compensation? [N/A]