Poisoning Attack for Inter-agent Transfer Learning

Zelei Cheng(✉) and Zuotian Li

Purdue University, West Lafayette, IN 47906, USA
cheng473@purdue.edu

Abstract. In reinforcement learning, high sample complexity is a big challenge to deal with. Inter-agent transfer learning is one solution to this challenge that can leverage the experience of other more competent agents. In this paradigm, a student can make a query to the teacher and the teacher will give some action advice given the current state. However, most previous works ignored the instruction reliability problem. In this work, we investigate the instruction reliability issue based on the one-to-one teaching framework and formulate the poisoning attack as an optimization problem. By solving the optimization problem, the attacker can significantly influence the performance of the student in three different query models. Evaluation highlights that we need to consider the instruction reliability when using teacher-student frameworks in reinforcement learning.

Keywords: Poisoning attack · Inter-agent transfer learning · Reinforcement learning

1 Introduction

Reinforcement Learning is learning how to map states to actions such that the expected return can be maximized [27]. The agent learns a policy through the interaction with the environment and the environment will give the feedback in the form of numerical reward. The reinforcement learning framework has gained great popularity in recent years and the related learning methods have been developed to handle increasingly complicated problems [30]. However, as the task is more and more complex, reinforcement learning can suffer from the high sample complexity challenge [18], where the reinforcement learning agents cannot sample all the state space and action space due to various concerns, e.g., limited time, security issue [11]. Therefore, how to address the high sample complexity is an urgent need for solving complex tasks.

Transfer learning leverages the previous knowledge in one or more source tasks to assist the current target task [9]. Recently, it has been combined with reinforcement learning. Clouse et al. [7] proposed a teaching method for reinforcement learning where a human expert teaches the reinforcement learning

J. Garcia-Alfaro et al. (Eds.): SecureComm 2021, LNICST 399, pp. 394–404, 2021.
https://doi.org/10.1007/978-3-030-90022-9_21

agent to accelerate the learning rate. Then multiple teacher-student frameworks are proposed to transfer the knowledge from the teacher to the student. Samples of previous interactions with the environment will then be mapped to the student's policy space. Moreover, the teacher is not limited to human experts, but also well-trained reinforcement learning agents [29] and reinforcement learning agents learning in progress [22]. Typically, it is a one-teacher-one-student pairwise relationship.

Although most previous work assumes that the student should take the teacher's action advice without any condition, the instruction reliability should be considered in practice. Here are some reasons: first, the student's query or the teacher's advice might be corrupted or lost, e.g., due to unreliable communication channel; second, teachers can be compromised or malicious to pollute the student's policy; even if we assume that all teachers are honest, teachers might have worse policies than the student's in some cases. Therefore, it is meaningful to investigate the performance of the teacher-student framework considering the instruction reliability issue. Felipe Leno Da Silva et al. [11] pointed out that how to perform instruction reliability determination effectively is still an open problem.

In this work, we investigate the instruction reliability issue of the teacher-student framework under a one-to-one scenario, where one teacher teaches one student. Especially, we construct poisoning attacks for this scenario and formulate the attacks as an optimization problem. Through experiments, we can see the damage caused by our proposed attacks. Future work will be investigating the more complicated teacher-student frameworks.

2 Related Work

2.1 Transfer Learning for Reinforcement Learning

In reinforcement learning [28], an agent learns an optimal policy to finish a task through trial and error approaches. The policy learning process is enabled by the interaction between the agent and the environment. Typically, reinforcement learning can be modeled as a Markov Decision Process. When the agent observe its current state s_t, it will take some action a_t from its policy π_t. Then the environment will give the feedback, i.e., the agent's new state s_{t+1} and numeric reward r_t. The reinforcement learning agent uses $< s_t, a_t, r_t >$ to update its policy π_t in order to maximize its cumulative reward in the long run. The interaction will occur repeatedly until the end of the learning process. However, sometimes the state-action space is too large to sample, it might be expensive for agents to explore all possible state-action pairs [18,19]. To this end, transfer learning is applied to solve this issue where the agents can leverage the knowledge of multiple source tasks to accelerate the accomplishment of target tasks.

Typically, transfer learning for reinforcement learning can be divided into two categories, i.e., single-agent transfer and inter-agent transfer [11]. Single-agent transfer includes value function transfer [31], policy reuse [15], and multi-task learning [33]. Learning from feedback [16], action advising [10], and learning

from demonstration [5] are examples of inter-agent transfer. Especially, in this paper, we focus on inter-agent transfer learning where a student leverages the experience of another, more competent agent. Torrey and Taylor [32] introduced a teacher-student framework for reinforcement learning where a student will execute the action his teacher advises and proposed a budget model for teaching where the teacher may only give such advice a limited number of times. Based on the budget model, Zimmer et al. [35] modeled when to teach as a reinforcement learning problem and provided an efficient technique to choose the right moment to give advice. However, most work assumes that the teacher is an expert that has domain knowledge. Recently, Omidshafiei et al. [22] suggested that every agent can be another's teacher and proposed a two-level learning framework for cooperative multi-agent reinforcement learning, i.e., task-level learning and advising-level learning.

Nevertheless, previous work mostly does not take instruction reliability into account. They simply assumed that all instructions are reliable, and proceeds immediately to the knowledge merging step [11].

2.2 Poisoning Attack

Poisoning attacks usually occur in the training phase of the machine learning system with the target to compromise the integrity [3], for example, injecting some malicious data into the training dataset. The training phase involves training dataset collection and the learning process. Most poisoning attacks happen during the collection of the training datasets as known as data poisoning attacks [4,14,17]. For example, Cao et al. [6] proposed data poisoning attacks to local differential privacy protocols that can inject fake users into the protocols such that the local perturbed data will be carefully crafted before submitted to the data collector. A few works investigate the poisoning attack occurring in the learning process. Fang et al. [13] proposed local model poisoning attacks to federated learning. The attackers upload the adversarial local update by calculating how to deviate the global model to the inverse direction of the correct weight update. Our work falls in the second category, i.e., poisoning attack in the learning process.

For the objective of the poisoning attacks, generally, there are two types: untargeted poisoning attack [20,25,34] which aims to cause a higher testing error for testing examples, and targeted poisoning attack [24,26], where the machine learning model produces unexpected predictions. The poisoning attack we'll discuss here will fall into the first category.

3 System Model

3.1 Reinforcement Learning Setting

Here we consider a multi-agent reinforcement learning setting, where two agents are involved as Fig. 1 shows. One reinforcement learning agent is the teacher and another is the student.

The teacher and the student interact with the same environment and they have the same task. Therefore, there is no need to measure the similarity between the teacher's task and the student's task as [2]. Typically, the teacher should have more experience than the student. For example, the teacher has already interacted with the environment several times before the student starts interacting with the environment.

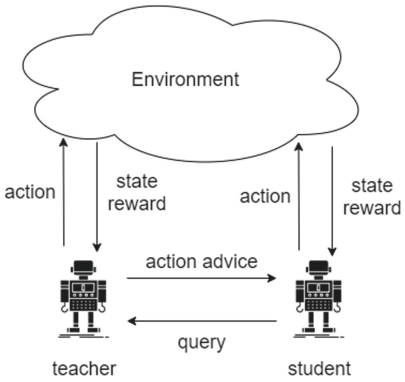

Fig. 1. One-to-one teaching relationship

We assume that the teacher and the student share the same state space $\mathcal{S}^+$ and action space $\mathcal{A}$, i.e., a homogeneous setting. Thus, the teacher's advice can be directly adopted without the need for action space mapping.

3.2 Student Query Model

The query budget model as [32] will be considered in this paper to simulate the actual limit of communication condition. Here we consider three query models:

1. Random query: The student makes a query with probability p;
2. Ask important [1]: If $I(s) = \max_a Q(s,a) - \min_a Q(s,a) > threshold$, the student will make an query;
3. Ask uncertain [8]: If $I(s) = \max_a Q(s,a) - \min_a Q(s,a) < threshold$, the student will make an query.

4 Attacking Inter-agent Transfer Learning

4.1 Threat Model

We characterize our threat model with respect to an attacker's goal and capability.

Attacker's Goal: Given a one-to-one teaching framework, the attacker's goal is to pollute the student's policy and minimize the student's long-term accumulated

reward. Note that such attacks are categorized as untargeted poisoning attacks, which make the student's reinforcement learning system unusable. This attack is well-motivated in real-world reinforcement learning applications. For instance, an attacker may be interested in providing misleading or life-threatening information to reinforcement learning agents such that he can cause damage to the competitor's reinforcement learning systems. Previous work regarding adversarial attacks to reinforcement learning systems such as [12,21,23] achieves this goal by manipulating the reward function or the environment transition dynamics.

Attacker's Capability: In our threat model, we assume the attacker is able to inject a malicious teacher into the pairwise teaching system. The malicious teacher can send arbitrary action advice in the query process to the student. The attacker can have different levels of knowledge of the environment. In particular, here we consider two cases: full knowledge and partial knowledge. Full knowledge means that the attacker is an expert who knows the optimal policy. Partial knowledge means that the attacker estimates the optimal policy through previous interactions with the environment. The attacker can use either a deterministic policy or stochastic policy to teach agents.

4.2 Formulating Poisoning Attacks

We formulate the poisoning attacks for inter-agent transfer learning as an optimization problem, which should minimize the accumulated reward of the student. No matter whether the attacker is an expert, the attacker knows or has an estimate of the optimal policy $\pi^*(\mathcal{S}^+)$. What the attacker needs to do is make the student's policy deviate from the optimal policy as much as possible. More specifically, in each iteration when the student inquires the attacker, the attacker should choose the action advice sampled from the non-optimal policy. However, to make the attack nontrivial to be detected, the short-term observation of the student should not be changed too much.

Given the current state s_t, the attacker uses the (estimated) optimal policy $\pi^*(s_t)$. Suppose the advising policy of the attacker is denoted as $\pi'(s_t)$. The attacker's goal is to find the optimal policy to make $\pi'(s_t)$ deviate from $\pi^*(s_t)$ as much as possible without letting the student know. It requires to ensure that the short-term observation should not be changed too much. Mathematically,

$$\begin{aligned} &\max D_{\mathrm{KL}}(\pi'(s_t)\|\pi^*(s_t)) \\ &\text{s.t. } \|r_t^{'} - r_t^*\| < \epsilon \end{aligned} \tag{1}$$

where $r_t^{'}$ denotes the reward after executing the teacher's action advice and r_t^* denotes the reward after executing the action sampled from the optimal policy. The Kullback-Leibler divergence $D_{\mathrm{KL}}(.,.)$ is defined as

$$D_{\mathrm{KL}}(P\|Q) = \sum_{x\in\mathcal{X}} P(x)\log\left(\frac{P(x)}{Q(x)}\right) \tag{2}$$

where P and Q are two discrete probability distribution defined on the same probability space. Note that KL divergence also applies to continues probability distribution where summation can be replaced with integral.

4.3 Solving the Optimization Problem

The proposed attack is based on solving the aforementioned optimization problem in Eq. (1). However, the optimal solution depends on the concrete applications.

For example, when it comes to Hallway Game which we will use for evaluation later, the reward will always be 0 until the agent reaches the opposite state. Therefore, as long as the agent not reaching the opposite state, the reward will be 0 regardless of what action to be chosen. Thus, the optimization problem has been simplified as solving the optimal advising policy $\pi^{'}(s_t)$ such that the KL divergence is maximized. Since in Hallway Game, if the agent is initiated on the left side, the optimal policy should be always going right, i.e., $\pi^*(s_t) = [0, 1]$. To make the KL divergence maximized, $\pi^{'}(s_t) = [1, 0]$ should be the optimal solution. The analysis is similar for initiating the agent on the right side.

Therefore, in a target-oriented reward mechanism, i.e., that the reward only occurs if the agent reaches the target, we can relax our optimization problem without the constraint. The new optimization problem should be

$$\max D_{\mathrm{KL}}(\pi'(s_t) \| \pi^*(s_t)) \tag{3}$$

For simplicity, let's consider the action space to be a discrete space. Assume that there are n actions can be chosen. Then the two distributions in Eq. (3) can be written as

$$\pi'(s_t) = [p_1^{'}, p_2^{'}, \ldots, p_n^{'}] \tag{4}$$

with the constraint $\sum_{i=1}^{n} p_i^{'} = 1$.

$$\pi^*(s_t) = [p_1^*, p_2^*, \ldots, p_n^*] \tag{5}$$

with the constraint $\sum_{i=1}^{n} p_i^* = 1$.

Then the KL divergence can be interpreted as

$$D_{\mathrm{KL}}(\pi'(s_t) \| \pi^*(s_t)) = \sum_{i=1}^{n} p_i^{'} \log(\frac{p_i^{'}}{p_i^*}) \tag{6}$$

If $\exists j$ such that $p_j^* = 0$, obviously, one optimal solution is that $p_j^{'} = 1$ with other elements in $\pi'(s_t)$ equals to zero.

Otherwise, if $\forall j, p_j^* \neq 0$, w.l.g., we assume that $p_1^* \leq p_2^* \leq \cdots \leq p_n^*$. To maximize the KL divergence, the optimal solution should be $p_1^{'} = 1$ with other elements in $\pi'(s_t)$ equals to zero.

5 Evaluation

5.1 Experiment Setup

Environment: Hallway Game. In Hallway Game [22], an agent receives +1 reward by navigating to opposite states in a 17-grid hallway.

The game will terminate if the agent has reached the terminal state or the number of iterations reaches the threshold.

Compared Cases: We will consider four different assumptions for the teacher: 1) Honest teacher who will teach the optimal policy; 2) No teacher; 3) Malicious teacher (fixed policy) who knows the optimal policy but will teach the worst policy; 4) Malicious teacher (grid search) who estimates the optimal policy through previous interactions with the environment but will teach the worst policy (which is solved by the aforementioned optimization problem).

The performance of the student under three query models will be tested in the experiments, i.e., random query, ask important, ask uncertain.

Reinforcement Learning Setting: There are two agents involved in this reinforcement learning task. One is the teacher and the other is the student.

The student uses a Q-learning algorithm to finish the reinforcement learning task. Once the query condition satisfies, the student will ask the teacher for advice. Then the teacher gives action advice, and the student will directly adopt this advice and reach a new state. The detailed algorithm we use in the experiments is in Algorithm 1.

Algorithm 1: Q-learning with query

parameter : step size α, discount rate γ, small $\epsilon > 0$
Initialize $Q(s, a)$, for all $s \in \mathcal{S}^+, a \in \mathcal{A}(s)$, arbitrarily except that $Q(\text{terminal}, \cdot) = 0$;
foreach *episode* **do**
  Initialize S;
  foreach *step of episode* **do**
    if *the query condition satisfies* **then**
      Query the teacher
      Get the teacher's action advice A
    end if
    else
      Use $\epsilon - greedy$ method to choose action
      A for S according to the policy π derived from Q table
    end if
    Take action A, observe R, S';
    $Q(S, A) \leftarrow Q(S, A) + \alpha[R + \gamma \max_a Q(S', a) - Q(S, A)]$;
    $S \leftarrow S'$;
  end foreach
end foreach

5.2 Results

We perform experiments corresponding to the aforementioned settings. We test the performance of four cases under three query models. Figure 2 shows the performance of the student under the random query model with different teacher assumptions. Figure 3 shows the performance of the student under the ask important model with different teacher assumptions. Figure 4 shows the performance of the student under the ask uncertain model with different teacher assumptions.

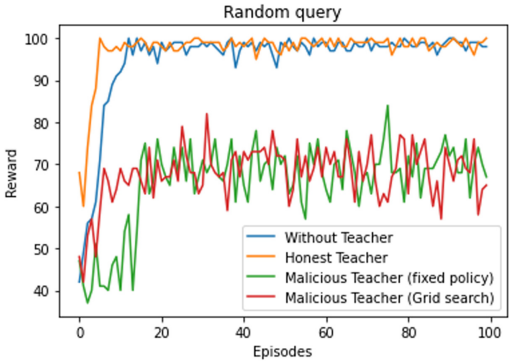

Fig. 2. Random query

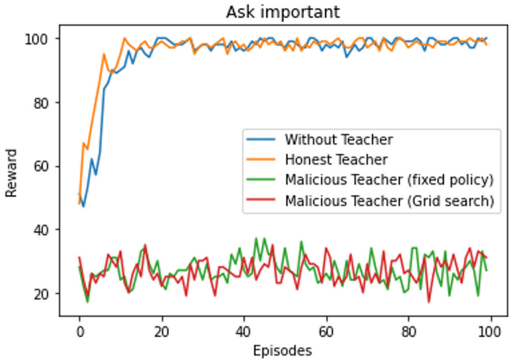

Fig. 3. Ask important

Generally, an honest teacher can accelerate the learning process of the student. Compared with the case that no teacher is involved, the student can gain higher rewards with an honest teacher (an expert).

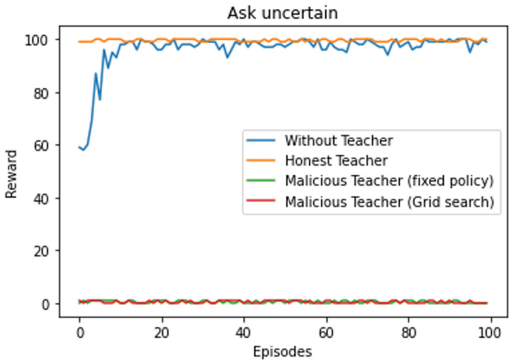

Fig. 4. Ask uncertain

Nevertheless, a malicious teacher can actually damage the student's learning process. Especially, for the ask uncertain model, under the instruction of a malicious teacher, the student will learn almost nothing. It is properly due to the fact that at the beginning of the learning process, the student is unclear and needs the teacher's assistance. A malicious teacher will give him wrong instruction and make him not learn any useful information from the feedback provided by the environment.

6 Future Work

Our work is limited to untargeted poisoning attacks in a one-to-one teaching framework. It would be interesting to study targeted poisoning attacks under this framework. Moreover, it might be trivial for the student to detect the attacks in an easy task. In complicated tasks where a single agent has difficulty in exploring all the state-action space, the attacks should be efficient. It should be more interesting to investigate other more complex teacher-student relationships.

7 Conclusion

In this paper, we investigate the one-to-one teaching framework. We demonstrate that inter-agent transfer learning is vulnerable to our poisoning attacks that give poisonous action advice to the student during the learning process. In particular, to minimize the accumulated reward of the student, an attacker can craft the action advice such that the advising policy deviates the most from the optimal policy. Moreover, finding such crafted action advice can be formulated as an optimization problem. Through evaluation, our results highlight that we need to consider the instruction reliability when using teacher-student frameworks in reinforcement learning.

References

1. Amir, O., Kamar, E., Kolobov, A., Grosz, B.J.: Interactive teaching strategies for agent training. In: Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence, pp. 804–811 (2016)
2. Barekatain, M., Yonetani, R., Hamaya, M.: Multipolar: multi-source policy aggregation for transfer reinforcement learning between diverse environmental dynamics. arXiv preprint arXiv:1909.13111 (2019)
3. Barreno, M., Nelson, B., Sears, R., Joseph, A.D., Tygar, J.D.: Can machine learning be secure? In: Proceedings of the 2006 ACM Symposium on Information, Computer and Communications Security, pp. 16–25 (2006)
4. Biggio, B., Nelson, B., Laskov, P.: Poisoning attacks against support vector machines. In: Proceedings of the 29th International Conference on International Conference on Machine Learning, pp. 1467–1474 (2012)
5. Brys, T., Harutyunyan, A., Suay, H.B., Chernova, S., Taylor, M.E., Nowé, A.: Reinforcement learning from demonstration through shaping. In: Twenty-fourth International Joint Conference on Artificial Intelligence (2015)
6. Cao, X., Jia, J., Gong, N.Z.: Data poisoning attacks to local differential privacy protocols. In: 30th USENIX Security Symposium (USENIX Security 2021) (2021)
7. Clouse, J.A., Utgoff, P.E.: A teaching method for reinforcement learning. In: Machine Learning Proceedings 1992, pp. 92–101. Elsevier (1992)
8. Clouse, J.A.: On integrating apprentice learning and reinforcement learning. University of Massachusetts Amherst (1996)
9. Da Silva, F.L., Costa, A.H.R.: A survey on transfer learning for multiagent reinforcement learning systems. J. Artif. Intell. Res. **64**, 645–703 (2019)
10. Da Silva, F.L., Hernandez-Leal, P., Kartal, B., Taylor, M.E.: Uncertainty-aware action advising for deep reinforcement learning agents. In: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 34, pp. 5792–5799 (2020)
11. Da Silva, F.L., Warnell, G., Costa, A.H.R., Stone, P.: Agents teaching agents: a survey on inter-agent transfer learning. Auton. Agent. Multi-Agent Syst. **34**(1), 1–17 (2020)
12. Everitt, T., Krakovna, V., Orseau, L., Legg, S.: Reinforcement learning with a corrupted reward channel. In: Proceedings of the 26th International Joint Conference on Artificial Intelligence, pp. 4705–4713 (2017)
13. Fang, M., Cao, X., Jia, J., Gong, N.: Local model poisoning attacks to byzantine-robust federated learning. In: 29th USENIX Security Symposium (USENIX Security 2020), pp. 1605–1622 (2020)
14. Fang, M., Yang, G., Gong, N.Z., Liu, J.: Poisoning attacks to graph-based recommender systems. In: Proceedings of the 34th Annual Computer Security Applications Conference, pp. 381–392 (2018)
15. Fernández, F., Veloso, M.: Probabilistic policy reuse in a reinforcement learning agent. In: Proceedings of the fifth International Joint Conference on Autonomous Agents and Multiagent Systems, pp. 720–727 (2006)
16. Griffith, S., Subramanian, K., Scholz, J., Isbell, C.L., Thomaz, A.L.: Policy shaping: Integrating human feedback with reinforcement learning. Georgia Institute of Technology (2013)
17. Jagielski, M., Oprea, A., Biggio, B., Liu, C., Nita-Rotaru, C., Li, B.: Manipulating machine learning: poisoning attacks and countermeasures for regression learning. In: 2018 IEEE Symposium on Security and Privacy (SP), pp. 19–35. IEEE (2018)

18. Kakade, S.M.: On the sample complexity of reinforcement learning. Ph.D. thesis, UCL (University College London) (2003)
19. Lattimore, T., Hutter, M., Sunehag, P.: The sample-complexity of general reinforcement learning. In: International Conference on Machine Learning, pp. 28–36. PMLR (2013)
20. Li, B., Wang, Y., Singh, A., Vorobeychik, Y.: Data poisoning attacks on factorization-based collaborative filtering. In: Proceedings of the 30th International Conference on Neural Information Processing Systems, pp. 1893–1901 (2016)
21. Ma, Y., Zhang, X., Sun, W., Zhu, X.: Policy poisoning in batch reinforcement learning and control. In: Advances in Neural Information Processing Systems (2019)
22. Omidshafiei, S., et al.: Learning to teach in cooperative multiagent reinforcement learning. In: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 33, pp. 6128–6136 (2019)
23. Rakhsha, A., Radanovic, G., Devidze, R., Zhu, X., Singla, A.: Policy teaching via environment poisoning: training-time adversarial attacks against reinforcement learning. In: International Conference on Machine Learning, pp. 7974–7984. PMLR (2020)
24. Shafahi, A., et al.: Poison frogs! Targeted clean-label poisoning attacks on neural networks. In: Proceedings of the 32nd International Conference on Neural Information Processing Systems, pp. 6106–6116 (2018)
25. Sharif, M., Bhagavatula, S., Bauer, L., Reiter, M.K.: Accessorize to a crime: real and stealthy attacks on state-of-the-art face recognition. In: Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security, pp. 1528–1540 (2016)
26. Suciu, O., Marginean, R., Kaya, Y., Daume III, H., Dumitras, T.: When does machine learning *fail*? Generalized transferability for evasion and poisoning attacks. In: 27th USENIX Security Symposium (USENIX Security 2018), pp. 1299–1316 (2018)
27. Sutton, R.S., Barto, A.G.: Reinforcement Learning: An Introduction. MIT Press, Cambridge (2018)
28. Sutton, R.S., Barto, A.G., et al.: Introduction to Reinforcement Learning, vol. 135. MIT Press, Cambridge (1998)
29. Taylor, A., Duparic, I., Galván-López, E., Clarke, S., Cahill, V.: Transfer learning in multi-agent systems through parallel transfer (2013)
30. Taylor, M.E., Stone, P.: Transfer learning for reinforcement learning domains: a survey. J. Mach. Learn. Res. **10**(7), 1633–1685 (2009)
31. Taylor, M.E., Stone, P., Liu, Y.: Transfer learning via inter-task mappings for temporal difference learning. J. Mach. Learn. Res. **8**(9), 2125–2167 (2007)
32. Torrey, L., Taylor, M.: Teaching on a budget: agents advising agents in reinforcement learning. In: Proceedings of the 2013 International Conference on Autonomous Agents and Multi-agent Systems, pp. 1053–1060 (2013)
33. Wilson, A., Fern, A., Ray, S., Tadepalli, P.: Multi-task reinforcement learning: a hierarchical Bayesian approach. In: Proceedings of the 24th International Conference on Machine Learning, pp. 1015–1022 (2007)
34. Yang, G., Gong, N.Z., Cai, Y.: Fake co-visitation injection attacks to recommender systems. In: NDSS (2017)
35. Zimmer, M., Viappiani, P., Weng, P.: Teacher-student framework: a reinforcement learning approach. In: AAMAS Workshop Autonomous Robots and Multirobot Systems (2014)