ICASSP 2024 - 2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP) | 979-8-3503-4485-1/24/$31.00 ©2024 IEEE | DOI: 10.1109/ICASSP48485.2024.10448500

A NEW PRE-TRAINING PARADIGM FOR OFFLINE MULTI-AGENT REINFORCEMENT LEARNING WITH SUBOPTIMAL DATA

Linghui Meng[1,2], *Xi Zhang*[1,2], *Dengpeng Xing*[1,2], *Bo Xu*[1,2]

[1]Institute of Automation, Chinese Academy of Sciences, China
[2]School of Artificial Intelligence, University of Chinese Academy of Sciences, China
{menglinghui2019, xi.zhang, dengpeng.xing, xubo}@ia.ac.cn

ABSTRACT

Offline multi-agent reinforcement learning (MARL) with pre-training paradigm, which uses a large quantity of trajectories for offline pre-training and online deployment, has become fashionable lately. While performing well on various tasks, conventional pre-trained decision-making models based on imitation learning typically require many expert trajectories or demonstrations, which limits the development of pre-trained policies in multi-agent case. To address this problem, we propose a new setting, where a multi-agent policy is pre-trained offline using suboptimal (non-expert) data and then tested online with the expectation of high rewards. In this practical setting inspired by contrastive learning , we propose YANHUI, a simple yet effective framework utilizing a well-designed reward contrast function for multi-agent policy representation learning from a dataset including various reward-level data instead of just expert trajectories. Furthermore, we enrich the multi-agent policy pre-training with mixture-of-experts to dynamically represent it. With the same quantity of offline StarCraft Multi-Agent Challenge datasets, YANHUI achieves significant improvements over offline MARL baselines. In particular, our method surprisingly competes in performance with earlier state-of-the-art approaches, even with 10% of the expert data used by other baselines and the rest replaced by poor data.

Index Terms— Multi-agent system, reinforcement learning, pre-training

1. INTRODUCTION

Multi-Agent Reinforcement Learning (MARL) has attracted more attention in recent research aiming at efficient cooperation or competition for real-time strategy games [1], ball-based games [2, 3], and autonomous driving cases [4]. Deep MARL empowers multiple agents to solve challenging tasks with policies parametrized by neural networks, learning from historical trajectories of interactions with the simulated environment. However, these successes are built upon considerable interactions after sufficient exploration on a policy space, which grows exponentially with the number of agents. In addition, the tremendous interaction costs are high, which induces offline MARL algorithms that utilize previously collected offline data to train a policy and test it online [5]. There are mainly two research lines in offline MARL, including conservatively constraining the state-action value function and imitating from the offline dataset with transformer-based autoregressive models [6, 7, 8, 9, 10, 11]. In this paper, we mainly focus on the latter approaches based on large-scale models due to the scalability of attention-based models [12].

Despite the empirical success of these methods by filtering out most low-reward trajectories (called poor data), it remains unclear how to use suboptimal data. Existing methods like MADT are limited by the requirement for expert trajectories with high rewards [11, 13]. However, to perform well online for multi-agent policy, the extensive poor trajectories in realistic scenarios are not negligible. There is a practical reason for many poor data in offline datasets: In the real world, collected demonstrations performed by humans, such as in robotics, struggle with expert rewards. Leveraging this imperfect offline data with expert data together encourages recent research in imitation learning and offline RL. [14, 15, 16] attempt to use the adversarial-based method to imitate poor data. In addition, [17] uses data with unlabeled reward and reweight relabeled data for offline single-agent RL. To address this issue for multi-agent case, we propose a new setting called Weakly Supervised Pre-trained Multi-agent Reinforcement Learning, WSPM, to address the problem of multi-agent policy pre-training on datasets with different levels. The goal is to obtain a multi-agent policy that can perform well online from an offline dataset characterized by mixed rewards.

An intuitive idea to transplant existing methods in this new setting is to use rejection sampling by filtering poor trajectories out and pre-training the policy offline. However, as described above, the proportion of poor data is considerable and should not be overlooked although the filtering process is feasible with rewards. Intuitively, poor data can steer the policy away from bad policies. This contributes to the im-

This work is supported by the National Key R&D Program of China (No.2022ZD0116405) and the Strategic Priority Research Program of the Chinese Academy of Sciences (No.XDA27030300).

provement of overall performance. We resolve the problems in WSPM by contrastively pre-training multi-agent policies from trajectories with different rewards. In detail, the policies learn from both expert and poor data by leveraging the contrastive learning methods to make our training policy close to the policies of good history and far away from the policies of undesirable history. While contrastive learning using positive and negative samples (sometimes called triplet loss) can well discriminate the quality of different trajectories [18], hard assignment results in challenges in the inability to classify different samples within the same interval.

To address hard assignment issues, we define a reward contrast function computed by offline data to inform the distance between a pre-training policy and the policy with the maximum reward. In addition, we create an architecture that automatically assigns multiple agents based on a combination of experts. Our contributions are summarized as follows: 1) We explicitly formulate a novel setting for offline pre-trained MARL, called WSPM, that makes full use of trajectories with different levels of quality, including expert and poor data. 2) To enrich the multi-agent policy pre-training, we use mixture-of-experts to represent it dynamically. To tackle the mixed data issues, we introduce a reward contrast function to inform the policy of how far it is from policies underlying offline trajectories using soft scores measured by reward. 3) We evaluate YANHUI on the challenging multi-agent task SMAC. Using the same hyperparameters for all tasks, YANHUI outperforms prior offline RL and pre-training methods on the datasets with various rewards.

2. METHODOLOGY

This section introduces the proposed setting, weakly-supervised pre-trained MARL (WSPM), and the corresponding method, YANHUI, to utilize mixed data in WSPM for better policies. The mixed data contains two kinds of datasets $\mathcal{D}_E$ and $\mathcal{D}_P$ generated by π_e and π_p, which are expert and poor policies with high and low rewards, respectively. This setup is harder for pre-training but is much more practical as described earlier. However, many previous pre-training policy methods aim to exclude data generated from π_p because of poor performance and only learn from high-quality data, which is impractical. Therefore, we formally define this setting as WSPM and propose a pre-training method, YANHUI.

The key idea of YANHUI is to appropriately push and pull our pre-training multi-agent policy to expert and poor policies, respectively. We propose a reward contrast function to contrastively learn policy representations, leveraging mixture-of-experts to automatically modularize each agent.

2.1. Weakly Supervised Pre-trained MARL

WSPM is a weakly supervised pre-training setup for pre-training a multi-agent policy from amounts of trajectories with different reward levels as $\tau = \langle s^t, r^t, (o_i^t, a_i^t)_{i=1}^n)_{t=1}^T \rangle$. The pre-trained multi-agent policy is expected to act on the environment online to achieve high rewards without further fine-tuning. Given offline trajectories, $\mathcal{D} := \{\mathcal{D}_E \cup \mathcal{D}_P\}$, in which $\mathcal{D}_E$ denotes trajectories with high rewards, called expert data, and $\mathcal{D}_P$ denotes trajectories with low rewards, called poor data. Learning a multi-agent policy $\pi(a|\mathcal{O}(s))$ offline from $\mathcal{D}$ and then deploying it online for promising performance is the objective of the WSPM setting.

2.2. Algorithm of YANHUI

We are curious about how to design a framework offline to represent the multi-agent policy better. Typically, MARL methods, including MAPPO and QMIX [19, 20], learn the sharing policy or value network across agents by regarding each agent's policy modeling as a single task learning process. Inspired by this idea, in the offline stage, we regard multi-agent policy modeling as a multi-task and an input-conditional problem. An intuitive solution is to ensemble several individual agent policy networks, which costs a lot. In summary, we highlight two points for solving the WSPM problem with our presented method: 1) The reward contrast function is softly computed based on the input encoder instantiated by the strong architecture MADT with sigmoid-style approximations of the positive and negative score functions in the MARL context. 2) The mixture-of-experts (MOE) is used to model the multi-agent policy dynamically. The ensemble method costs much for training multiple times neural network parameters. [21] shows the modular method can help multi-task learning for reinforcement learning and control tasks. Therefore, we choose to dynamically learn a mixture-of-experts network to represent each agent's policy $\pi_i(a_i|\tau_i) = \sum_{k=1}^K p_k(a_i|\tau_i)/\mathcal{Z}$, where $\mathcal{Z}$ is a normalization constant instead of assigning combined weights to each ensembled sub-policy.

Derivative of Objective In contrast with prior works, we leverage the offline dataset to modularize the multi-agent policy with mixture-of-experts (MOE). For clear representation, we denote π_i as $P(a_i^t|\mathcal{O}_i(s^t);\theta)$ with mixture-of-experts. Then the objective can be derived as:

$$\begin{aligned} &\max_\theta \sum_t \prod_i P(a_i^t|\mathcal{O}_i(s^t);\theta) Q(s^t, \mathbf{a}^t) \\ &\Longrightarrow \max_\theta \sum_t \prod_i P(z|o_i^t) P(\hat{a}_i^t|o_i^t, z;\theta) P(r^t|s^t) \end{aligned} \tag{1}$$

where $P(z|o_i^t)$ denotes the gating function that organizes the instance discrimination task into K simpler subtasks representing with z by weighting the experts based on the input observation. The gating function can be computed as: $P(z|o_i) = \exp(\omega^T g_n/\tau)/(\sum_{k=1}^K (\omega^T g_k/\tau))$, where g_n denotes the observation embedding and $\{\omega_j\}_{j=1}^K$ denotes the gating prototypes. Besides, $P(\hat{a}_i^t|o_i^t, z;\theta)$ represents an

expert embedding observation of each agent into the latent space. Then we can formulate the probability with MOE.

Reward contrast function We especially propose a simple-form reward contrast function for contrastively optimizing the policy softly. We aim to optimize the encoder in each expert $P(\hat{a}_i^t|o_i^t, z; \theta)$, using the reward value as metrics to decide how closely the policy needs to approximate the policies in good and poor data. Furthermore, we extend the idea of $\min_\theta D(\pi_\theta, \pi_e)\ s.t.\ \max_\theta D(\pi_\theta, \pi_p)$. Instead of identifying trajectories, we define positivity and negativity scores, sc^+ and sc^-, using both reward and state-action pair geometric proximity measures, and function $G(\theta) = \sum_{a\in\mathcal{A}} \left(\log\left(1 + 1^\top \exp\left(\eta \mathbf{sc}^+ - \mu\right)\right)/\eta + \log\left(1 + 1^\top \exp\left(\mu - \nu \mathbf{sc}^-\right)\right)/\nu\right)$. The objective incorporating the reward contrast function and mixture-of-experts:

$$\mathcal{L}_{\theta,\phi} = \sum_n \sum_k q(z = k|a_n, \tau_n)(G(\theta) + \log q(z = k|a_n, \tau_n; \phi) - \log p(a_n|\tau_n, z = k)) \tag{2}$$

3. EXPERIMENTS

We experimentally evaluate YANHUI on the challenging multi-agent benchmark, SMAC [1], which includes various difficulty maps conforming to POMDP. We compare YANHUI with several strong baselines in the offline pre-trained MARL setup. Then we analyze how much expert data they need to obtain an expected policy and their robustness in facing different proportions of poor data. We further show ablation studies comparing the orthogonal choice of modules or objectives for the pre-training multi-agent policy.

3.1. Offline MARL datasets

Similar to datasets in single-agent offline RL [22], we leverage datasets collected from multi-agent challenging tasks, including several SMAC maps provided by MADT [11]. They train a multi-agent policy with the state-of-the-art multi-agent policy gradient algorithm, MAPPO, on diverse maps and collect trajectories in buffers synchronously [19]. The dataset includes lots of episodes in a total of 1e7 episodes. The quality is determined according to the discounted return r, in which episodes with $r \in [0, 10], [10, 18], [18, 20]$ are prescribed as poor, medium, and expert data, respectively. In this work, we investigate the expert and poor data extracted to describe the research problem more clearly.

3.2. Baseline methods and training details

We evaluate YANHUI compared with the existing well-known offline MARL algorithm. The highest performance reported on the SMAC dataset is MADT, an extension of the Decision Transformer in a multi-agent setting [11]. We compare our method with two groups of baselines. For imitation-based learning, we consider MADT and behavior cloning (BC) to validate the effectiveness. For offline RL, conventional conservative off-policy methods, including ICQ, CQL, and BCQ [9, 7, 6], are employed as baselines. We train each multi-agent policy from offline data on an NVIDIA Tesla A100 GPU with 40GB of memory.

Table 1: $JP_{0\%}$/$JP_{50\%}$, Jump point (the initial performance (returns) of agents online) results of offline MARL algorithms when the proportion of poor data is 0% and 50% keeping the episode number fixed on three maps of different difficulty

Maps	BCQ [6]	CQL [7]	ICQ [9]	BC	MADT [11]	**YANHUI (ours)**
3m (easy)	19.36/8.53	18.59/8.87	18.77/9.58	19.96/8.52	20.00/12.38	**20.00/18.57**
3s vs. 5z (hard)	17.7/7.32	17.23/8.55	17.02/8.89	17.95/9.25	18.56/11.67	**19.76/17.25**
corridor (super hard)	16.22/7.74	17.5/7.67	17.03/8.23	17.26/6.86	**19.59**/9.33	19.36/**17.55**

Table 2: *JP*, Jump point (the initial performance (returns) of agents online after training 100 epochs offline) results of YANHUI variants by dropping different modules on two easy and super hard tasks from SMAC. The dataset is set aligning with experiments in Section 3.3. All quantities are provided on a scale of 0.01. Standard errors over five random seeds are provided in brackets.

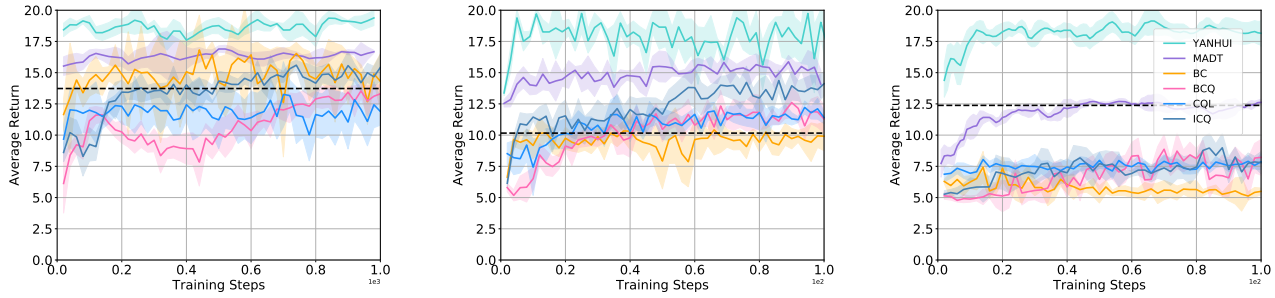
Method	3m (easy)	MMM2 (super hard)
YANHUI (our method)	**19.85** (± **0.13**)	**18.74** (± **0.56**)
without mixture-of-experts	18.23 (± 0.22)	16.53 (± 1.12)
without reward contrast function	14.55 (± 0.32)	10.25 (± 2.35)
without contrastive learning	10.24 (± 0.97)	7.37 (± 1.54)
without cross-entropy loss	10.54 (± 0.77)	6.52 (± 1.76)

3.3. How does YANHUI perform comparing with offline MARL baselines in WSPM regime?

To validate the effectiveness of YANHUI, we use trajectories collected from three SMAC maps ranging from easy to super hard for pre-training. Furthermore, according to the weakly supervised setting with mixed-quality data, we consider setting the proportion of poor trajectories in the whole set as p. In this section, we compare the performance of our method with other baselines with $p = 50\%$. Our results are given in Figure 1. We see that YANHUI outperforms other baselines on all difficulty tasks, showing the effectiveness of our approach. For further verification, we empirically compare the effect of directly increasing p from 0% to 50%, and the results are shown in Table 1. On easy and hard tasks, all the methods in experiments do not suffer significantly from the increased proportion of poor data. For extensively exploring the sensitivity of our method and baselines to the proportion of poor data, we validate performance by adjusting p more smoothly to show the robustness of YANHUI in Section 3.4.

3.4. The robustness of YANHUI to poor data

To further investigate the sensitivity of each method to data quality, we also design experiments by limiting the total



(a) 3m (easy) (b) 3s vs. 5z (hard) (c) corridor (super hard)

Fig. 1: Returns of learning models on easy and (super-)hard maps. Results suggest that YANHUI can consistently outperform other baselines when given 200 episodes in total with half poor data.

episode number to 40. Initially, this mini-dataset has 40 episodes of expert data without poor data. With the proportion of poor data increasing from 0% to 100% We select an easy SMAC map 3m for precise description and plot the curve of the converged performance in Figure 2. We see that without enough good data, baselines, especially imitation-based ones, learn worse when the proportion decreases to 50%, perhaps because they are harmed by the mixed poor data on which the expert policy is trained. In addition, YANHUI achieves a promising performance even with $p = 90\%$. In other words, even with only 10% expert data, our learned policy can perform comparably compared with conventional approaches pre-trained by 100% expert data.

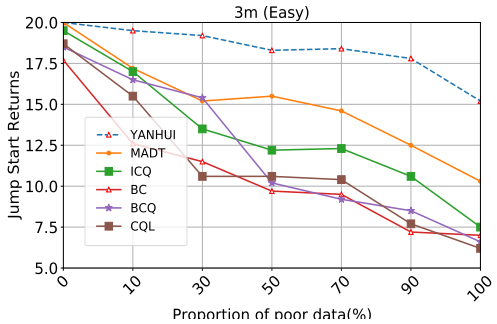


Fig. 2: Robustness with the proportion p of the poor data.

3.5. Ablation Studies

We validate the necessity of each module by dropping them out orthogonally. We consider answering the research questions: **RQ1:** Can mixture-of-experts help multi-agent offline pre-training? **RQ2:** What about using hard assignment-based contrastive learning even without CL rather than soft contrastive learning? **RQ3:** Is the cross-entropy loss necessary? We implement four types of ablations: (1) Remove the MOE architecture and maintain soft contrastive learning and offline cross-entropy loss. The removed pre-training framework can be viewed as MADT with a soft version of CL using rewards as input. (2) Remove the soft assignment mechanism and use hard assignment to compute the contrastive loss, where positive and negative samples are divided according to expert and poor data with explicit thresholds. Remove CL by comparing YANHUI with the removed part, i.e., MADT+MOE. (3) Ablate the supervised objective from the action ground truth. Table 2 shows the comparison performance among YANHUI variants with the above modification. In terms of offline data efficiency, YANHUI without MOE converges slower, which suggests MOE enhances multi-agent modeling. The reward contrast function we proposed is important for leveraging the mixed-quality data, removing it decreases the performance on the two maps sharply. The results also indicate the hard assignment cannot deal with the suboptimal data in our setting well. Furthermore, the results on the third row indicate the idea we presented that pulling the learning policy to the expert policies and pushing it away from the poor ones indeed helps solve the problems in this mixed-quality setting.

4. CONCLUSION

This paper presented a formal definition of a novel and practical setting for offline MARL pre-trained with suboptimal data. Furthermore, we proposed to leverage a well-designed reward contrast function to pre-train the multi-agent policy, composed of a mixture of experts, to solve problems in this new setting. We validated the presented method using the expert and poor trajectories. Empirical results on the challenging task SMAC show the robustness and superiority of YANHUI compared with other strong baselines. Even with only 10% expert data, our learned policy can perform as well as conventional approaches using exclusively expert data. For future work, we are interested in studying offline pre-training with sparse rewards and further exploring how to leverage less expert trajectories to develop our method.

5. REFERENCES

[1] M. Samvelyan, T. Rashid, C. S. De Witt, G. Farquhar, N. Nardelli, T. G. Rudner, C.-M. Hung, P. H. Torr, J. Foerster, and S. Whiteson, "The starcraft multi-agent challenge," *arXiv preprint arXiv:1902.04043*, 2019.

[2] H. Jia, Y. Hu, Y. Chen, C. Ren, T. Lv, C. Fan, and C. Zhang, "Fever basketball: A complex, flexible, and asynchronized sports game environment for multi-agent reinforcement learning," *arXiv preprint arXiv:2012.03204*, 2020.

[3] W. Shang, L. Espeholt, A. Raichuk, and T. Salimans, "Agent-centric representations for multi-agent reinforcement learning," *arXiv preprint arXiv:2104.09402*, 2021.

[4] M. Zhou, J. Luo, J. Villella, Y. Yang, D. Rusu, J. Miao, W. Zhang, M. Alban, I. Fadakar, Z. Chen *et al.*, "Smarts: Scalable multi-agent reinforcement learning training school for autonomous driving," *arXiv preprint arXiv:2010.09776*, 2020.

[5] S. Levine, A. Kumar, G. Tucker, and J. Fu, "Offline reinforcement learning: Tutorial, review, and perspectives on open problems," *arXiv preprint arXiv:2005.01643*, 2020.

[6] S. Fujimoto, D. Meger, and D. Precup, "Off-policy deep reinforcement learning without exploration," in *International Conference on Machine Learning*. PMLR, 2019, pp. 2052–2062.

[7] A. Kumar, A. Zhou, G. Tucker, and S. Levine, "Conservative q-learning for offline reinforcement learning," *arXiv preprint arXiv:2006.04779*, 2020.

[8] A. Kumar, J. Fu, M. Soh, G. Tucker, and S. Levine, "Stabilizing off-policy q-learning via bootstrapping error reduction," *Advances in Neural Information Processing Systems*, vol. 32, pp. 11 784–11 794, 2019.

[9] Y. Yang, X. Ma, L. Chenghao, Z. Zheng, Q. Zhang, G. Huang, J. Yang, and Q. Zhao, "Believe what you see: Implicit constraint approach for offline multi-agent reinforcement learning," *Advances in Neural Information Processing Systems*, vol. 34, 2021.

[10] L. Chen, K. Lu, A. Rajeswaran, K. Lee, A. Grover, M. Laskin, P. Abbeel, A. Srinivas, and I. Mordatch, "Decision transformer: Reinforcement learning via sequence modeling," *arXiv preprint arXiv:2106.01345*, 2021.

[11] L. Meng, M. Wen, Y. Yang, C. Le, X. Li, W. Zhang, Y. Wen, H. Zhang, J. Wang, and B. Xu, "Offline pre-trained multi-agent decision transformer: One big sequence model conquers all starcraftii tasks," *arXiv preprint arXiv:2112.02845*, 2021.

[12] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, "Attention is all you need," *arXiv preprint arXiv:1706.03762*, 2017.

[13] W.-C. Tseng, T.-H. Wang, Y.-C. Lin, and P. Isola, "Offline multi-agent reinforcement learning with knowledge distillation," in *Thirty-Sixth Conference on Neural Information Processing Systems*, 2022. [Online]. Available: https://openreview.net/forum?id=yipUuqxveCy

[14] Y.-H. Wu, N. Charoenphakdee, H. Bao, V. Tangkaratt, and M. Sugiyama, "Imitation learning from imperfect demonstration," in *International Conference on Machine Learning*. PMLR, 2019, pp. 6818–6827.

[15] J. Fu, A. Singh, D. Ghosh, L. Yang, and S. Levine, "Variational inverse control with events: A general framework for data-driven reward definition," *Advances in neural information processing systems*, vol. 31, 2018.

[16] L. Chen, R. Paleja, and M. Gombolay, "Learning from suboptimal demonstration via self-supervised reward regression," *arXiv preprint arXiv:2010.11723*, 2020.

[17] T. Yu, A. Kumar, Y. Chebotar, K. Hausman, C. Finn, and S. Levine, "How to leverage unlabeled data in offline reinforcement learning," *arXiv preprint arXiv:2202.01741*, 2022.

[18] A. Van den Oord, Y. Li, and O. Vinyals, "Representation learning with contrastive predictive coding," *arXiv e-prints*, pp. arXiv–1807, 2018.

[19] C. Yu, A. Velu, E. Vinitsky, Y. Wang, A. Bayen, and Y. Wu, "The surprising effectiveness of mappo in cooperative, multi-agent games," *arXiv preprint arXiv:2103.01955*, 2021.

[20] S. Whiteson, "Qmix: Monotonic value function factorisation for deep multi- agent reinforcement learning," *ICML*, 2018.

[21] R. Yang, H. Xu, Y. Wu, and X. Wang, "Multi-task reinforcement learning with soft modularization," *Advances in Neural Information Processing Systems*, vol. 33, pp. 4767–4777, 2020.

[22] J. Fu, A. Kumar, O. Nachum, G. Tucker, and S. Levine, "D4rl: Datasets for deep data-driven reinforcement learning," *arXiv preprint arXiv:2004.07219*, 2020.