ELA: Exploited Level Augmentation for Offline Learning in Zero-Sum Games

Extended Abstract

Shiqi Lei*
Institute of Automation,
Chinese Academy of Sciences
China
hunter2000118@gmail.com

Kanghoon Lee*
KAIST
South Korea
leehoon@kaist.ac.kr

Linjing Li
Institute of Automation,
Chinese Academy of Sciences
China
linjing.li@ia.ac.cn

Jinkyoo Park
KAIST
South Korea
jinkyoo.park@kaist.ac.kr

Jiachen Li
University of California, Riverside
USA
jiachen.li@ucr.edu

ABSTRACT

Offline learning derives effective policies from expert demonstrators' datasets without direct interaction. While recent research consider dataset characteristics like expertise level or multiple demonstrators, a distinct approach is necessary in zero-sum games, where outcomes significantly depend on the opponent's strategy. In this study, we introduce a novel approach using unsupervised learning techniques to estimate the exploited level (EL) of each trajectory from the offline dataset of zero-sum games made by diverse demonstrators. The estimated EL is then integrated into offline learning to maximize the influence of the dominant strategy. Our method enables interpretable EL estimation in multiple zero-sum games, effectively identifying dominant strategies. Also, EL augmented offline learning significantly enhances the imitation and offline reinforcement learning algorithms in zero-sum games.

KEYWORDS

Offline Learning, Imitation Learning, Offline Reinforcement Learning, Representation Learning, Zero-Sum Games

ACM Reference Format:
Shiqi Lei, Kanghoon Lee, Linjing Li, Jinkyoo Park, and Jiachen Li. 2024. ELA: Exploited Level Augmentation for Offline Learning in Zero-Sum Games: Extended Abstract. In *Proc. of the 23rd International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2024), Auckland, New Zealand, May 6 – 10, 2024*, IFAAMAS, 3 pages.

1 INTRODUCTION

Reinforcement learning often involves costly online interactions [1–4, 10]. To address the issue, methods like behavior cloning replicates actions from the offline datasets [9] assuming demonstrators are experts, with ease but sensitivity to suboptimal demonstrations, while offline reinforcement learning aims for optimal policies, showing robustness but facing challenges with small or biased datasets [6, 7]. While extensive research on learning from offline data exists in multi-agent systems [8], addressing the unique characteristics of demonstrators remains underexplored. In competitive environments like zero-sum games, data distribution is influenced by attributes and expertise of the players, necessitating the extraction of suitable representations by considering individual characteristics. Successful learning in zero-sum games from offline data, given the diverse forms of strategies [5], requires a deep understanding and consideration of these factors.

In this work, we (1) introduce the Partially-trainable-conditioned Variational Recurrent Neural Network (P-VRNN) for learning strategy representation in multi-agent games, (2) define the EL and proposes an unsupervised method for estimating it in zero-sum game datasets, (3) introduce EL Augmentation (ELA) enhancing offline learning across various algorithms, and (4) demonstrate its effectiveness in Two-player Pong and Limit Texas Hold'em.

2 METHOD

Trajectories, denoted as $\tau^i = ((o^i_0, a^i_0), ..., (o^i_{T^i}, a^i_{T^i}))$, consist of observations o^i_t and actions a^i_t at time t, with length T^i. The dataset of all trajectories is represented as Γ. We assume that within a single trajectory, the strategy of a player is consistent.

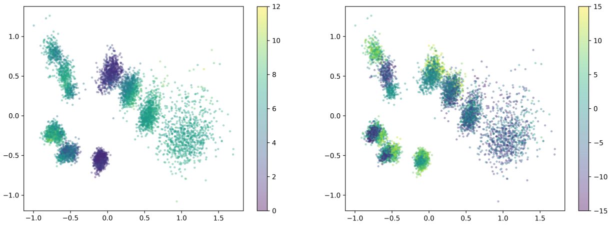
Learning Strategy Representation. We propose a P-VRNN where partially-trainable-conditioned means that part of the condition on the neural network is trainable, and the condition acquisition process is entirely unsupervised. The proposed P-VRNN has four major components as introduced below.

The **trajectory encoder** is defined as $\phi_\mathrm{e}(h_{t-1}, o_t, a_t, l) = [\mu^z_t, \sigma^z_t]$, $z_t \mid h_{t-1}, o_t, a_t, l \sim \mathcal{N}(\mu^z_t, \mathrm{diag}((\sigma^z_t)^2))$. The **prior estimator** is defined as $\phi_\mathrm{p}(h_{t-1}, o_t, l) = [\hat{\mu}_t, \hat{\sigma}_t]$, $z_t \mid h_{t-1}, o_t, l \sim \mathcal{N}(\hat{\mu}_t, \mathrm{diag}(\hat{\sigma}^2_t))$. The **action decoder** is defined as $\phi_\mathrm{d}(h_{t-1}, z_t, o_t, l) = [\mu^x_t, \sigma^x_t]$, $a_t \mid h_{t-1}, z_t, o_t, l \sim \mathcal{N}(\mu^x_t, \mathrm{diag}((\sigma^x_t)^2))$. The **recurrent unit** is defined as $\phi_\mathrm{r}(h_{t-1}, a_t, z_t, o_t, l) = h_t$.

The P-VRNN loss function includes the **reconstruction loss** ($\mathcal{L}_{\mathrm{recon},t}$), ensuring the encoder-decoder closely aligns with the true data through cross-entropy $CE(p, q) = -\int p(x) \log q(x) \mathrm{d}x$. Additionally, the **regularization loss** ($\mathcal{L}_{\mathrm{KL},t}$) seeks alignment between

[*] Both authors contributed equally to the paper

Proc. of the 23rd International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2024), N. Alechina, V. Dignum, M. Dastani, J.S. Sichman (eds.), May 6 – 10, 2024, Auckland, New Zealand. © 2024 International Foundation for Autonomous Agents and Multiagent Systems (www.ifaamas.org).



(a) Exploited level

(b) Trajectory reward

Figure 1: The trajectory representations of the Two-player Pong environment.

the prior estimator prediction and the trajectory encoder result, expressed as $KL(\mathcal{N}(\mu_t^z, \text{diag}((\sigma_t^z)^2)) \parallel \mathcal{N}(\hat{\mu}_t, \text{diag}(\hat{\sigma}_t^2))$. Finally, the total loss is denoted as $\mathcal{L} = \sum_{t=1}^{T} \left(\mathcal{L}_{\text{recon},t} + \mathcal{L}_{\text{KL},t}\right)$.

When learning the strategy representation, the trainable random representation vector l^i is initiated for each trajectory τ^i. The condition part of P-VRNN consists of observation o_t which changes over time and the representation vector l which is consistent during the whole trajectory and trainable. During training, all the l's are optimized together with the parameters of ϕ_p, ϕ_e, ϕ_d, and ϕ_r. The networks are trained to perform better in predicting the next step of trajectories, while the representations are optimized differently for each trajectory to provide customized predictions.

Exploited Level Estimator. We define measure $\mathrm{d}\pi$ on strategy space Π according to the probability of π chosen in the whole dataset. Denote the trajectory as τ, the representation function learned above as $f(\tau)$, and the reward of τ as $r(\tau)$. We remark that a trajectory τ should be mapped to a probability distribution of strategies such that $\int_{\pi \in \Pi} \tau(\pi) \mathrm{d}\pi = 1$, where $\tau(\pi)$ is the probability of using strategy π when having trajectory τ, instead of a single strategy. But we can view the mixture of π with probability $\tau(\pi)$ as a single mixed strategy $\int_{\pi \in \Pi} \pi \tau(\pi) \mathrm{d}\pi$, so we can still use notation $\pi(\tau)$ to represent the strategy of τ. We can approximate $E(\pi(\tau))$ by using $\left| E(\pi(\tau)) - \max_{d(f(\tau'), f(\tau)) < \delta} [-r(\tau')] \right| < \epsilon$. But in order to measure the exploitability of τ, we should calculate $E(\tau) := \int_{\pi \in \Pi} \tau(\pi) E(\pi) \mathrm{d}\pi$ instead of $E\left(\int_{\pi \in \Pi} \pi \tau(\pi) \mathrm{d}\pi\right)$. In fact, there will be an underestimation if we use this approximation, since $\int_{\pi \in \Pi} \tau(\pi) E(\pi) \mathrm{d}\pi \geq E\left(\int_{\pi \in \Pi} \pi \tau(\pi) \mathrm{d}\pi\right)$. Also, using maximum alone abandons almost all the information of nearby trajectories, which makes the approximation unstable.

We define the EL as

$$EL(\tau) = \mathbb{E}_\pi \left[-r(\pi, \pi(\tau)) \mid r(\pi, \pi(\tau)) \leq 0 \right].$$

And we assume that EL is proportional to exploitability:

$$E(\tau) \propto EL(\tau) = \frac{\int_{\pi \in \Pi} (-r(\pi, \pi(\tau))^+ \mathrm{d}\pi}{\int_{\pi \in \Pi} \mathbb{1}_{r(\pi, \pi(\tau)) \leq 0} \mathrm{d}\pi}.$$

To estimate EL with latent representation space, we provide an alternative definition of EL_δ:

$$EL_\delta(\tau) = \frac{\sum_{d(f(\tau), f(\tau')) < \delta} (-r(\hat{\pi}, \pi(\tau')))^+}{\sum_{d(f(\tau), f(\tau')) < \delta} \mathbb{1}_{r(\hat{\pi}, \pi(\tau')) \leq 0}}.$$

We have $\lim_{\delta \to 0^+} EL_\delta(\tau) = EL(\tau)$ and the trajectories that perform similarly to Nash Equilibrium can be detected with an EL_δ near 0. Since EL is the average of values satisfying conditions with distance constraints on the representation space, we can train an operator L to estimate EL from representation. We have representation l and reward r for each trajectory τ, and we intend to minimize $\sum_{r^i \geq 0} ||L(l^i) - r^i||_2$ so that the prediction from $L(l)$ becomes close to the mean of satisfying reward $r \geq 0$ nearby.

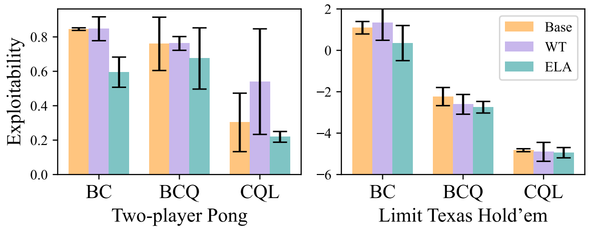


Figure 2: Exploitability supported on the demonstrator set generating the offline dataset. Lower is better.

EL Augmentation for Offline Learning. We formulate ELA for the offline learning objective as follows, emphasizing trajectories with a small EL: $\mathcal{L}^{\text{ELA}}(\pi) = \mathbb{E}_\tau \left[\mathbb{1}(EL(\tau) < EL_{\text{thresh}}) \cdot \mathcal{L}^{\text{OL}}(\pi, \tau) \right]$, where EL_{thresh} is a threshold that specifies the minimum value of an EL suitable for training. It provides data by sampling only for trajectories smaller than this value. $\mathcal{L}^{\text{OL}}$ represents an arbitrary method that allows offline learning by leveraging a trajectory such as imitation learning or offline RL methods.

3 EXPERIMENTS AND CONCLUSION

We validate our approach using two-player zero-sum games: Two-player Pong and Limit Texas Hold'em [11]. RL methods are used to generate diverse demonstrators, and behavior models are selected from multiple checkpoints to produce offline data.

At first, we analysis the strategy representation of trajectories reduced to two dimensions using PCA, followed by coloring based on estimated EL and reward in Figure 2. The strategy representation is separated into eight clusters, revealing both EL and reward distributions. Figure 1a shows that EL better reflects the strength of each player, as the values within each cluster are more consistent.

In our evaluation of the EL-augmented offline learning approach, we considered two main categories of methodologies. For IL, we employed BC, while in the domain of offline RL, we adopted representative algorithms BCQ [6] and CQL [7]. As an additional baseline for ELA, we trained the offline learning algorithm by exclusively selecting the winning trajectory (WT) from the dataset. In Figure 2, a comparison of exploitability is presented, supported on the demonstrator set. Basically, offline RL algorithms show better performance than the imitation learning approach on average because of the offline datasets from mixed demonstrators. Notably, ELA consistently outperforms alternative methods. While WT enhances the performance of the original offline algorithm in some cases, it occasionally hinders performance due to Q-value overestimation stemming from data bias by only selecting the winning trajectory.

In this work, we proposed ELA, enhancing offline learning in zero-sum games, and introduced the EL to measure proximity to Nash equilibrium, boosting offline learning universally. Our P-VRNN network effectively identifies trajectory strategy distribution, and theoretical groundwork, along with positive experimental results, supports the efficacy of ELA.

REFERENCES

[1] OpenAI: Marcin Andrychowicz, Bowen Baker, Maciek Chociej, Rafal Jozefowicz, Bob McGrew, Jakub Pachocki, Arthur Petron, Matthias Plappert, Glenn Powell, Alex Ray, et al. 2020. Learning dexterous in-hand manipulation. *The International Journal of Robotics Research* 39, 1 (2020), 3–20.

[2] Christopher Berner, Greg Brockman, Brooke Chan, Vicki Cheung, Przemysław Dębiak, Christy Dennison, David Farhi, Quirin Fischer, Shariq Hashme, Chris Hesse, et al. 2019. Dota 2 with large scale deep reinforcement learning. *arXiv preprint arXiv:1912.06680* (2019).

[3] Mariusz Bojarski, Davide Del Testa, Daniel Dworakowski, Bernhard Firner, Beat Flepp, Prasoon Goyal, Lawrence D Jackel, Mathew Monfort, Urs Muller, Jiakai Zhang, et al. 2016. End to end learning for self-driving cars. *arXiv preprint arXiv:1604.07316* (2016).

[4] Jianyu Chen, Bodi Yuan, and Masayoshi Tomizuka. 2019. Model-free deep reinforcement learning for urban autonomous driving. In *2019 IEEE intelligent transportation systems conference (ITSC)*. IEEE, 2765–2771.

[5] Wojciech M Czarnecki, Gauthier Gidel, Brendan Tracey, Karl Tuyls, Shayegan Omidshafiei, David Balduzzi, and Max Jaderberg. 2020. Real world games look like spinning tops. *Advances in Neural Information Processing Systems* 33 (2020), 17443–17454.

[6] Scott Fujimoto, David Meger, and Doina Precup. 2019. Off-policy deep reinforcement learning without exploration. In *International conference on machine learning*. PMLR, 2052–2062.

[7] Aviral Kumar, Aurick Zhou, George Tucker, and Sergey Levine. 2020. Conservative q-learning for offline reinforcement learning. *Advances in Neural Information Processing Systems* 33 (2020), 1179–1191.

[8] Ling Pan, Longbo Huang, Tengyu Ma, and Huazhe Xu. 2022. Plan better amid conservatism: Offline multi-agent reinforcement learning with actor rectification. In *International Conference on Machine Learning*. PMLR, 17221–17237.

[9] Dean A Pomerleau. 1988. Alvinn: An autonomous land vehicle in a neural network. *Advances in neural information processing systems* 1 (1988).

[10] Oriol Vinyals, Igor Babuschkin, Wojciech M Czarnecki, Michaël Mathieu, Andrew Dudzik, Junyoung Chung, David H Choi, Richard Powell, Timo Ewalds, Petko Georgiev, et al. 2019. Grandmaster level in StarCraft II using multi-agent reinforcement learning. *Nature* 575, 7782 (2019), 350–354.

[11] Daochen Zha, Kwei-Herng Lai, Songyi Huang, Yuanpu Cao, Keerthana Reddy, Juan Vargas, Alex Nguyen, Ruzhe Wei, Junyu Guo, and Xia Hu. 2020. RLCard: A Platform for Reinforcement Learning in Card Games. In *IJCAI*.