Optimizing Deep Q-Learning Experience Replay with SHAP Explanations: Exploring Minimum Experience Replay Buffer Sizes in Reinforcement Learning

author 1 anonymous[1][0000−1111−2222−3333] author 2 anonymous[1][0000−1111−2222−3333]

[1] Affilation,
[2] email@email.com

Abstract. Lack of explainability in Deep Reinforcement Learning (DRL) makes it difficult and costly to gain acceptance by regulated organizations where cost, safety and ethical concerns exist. Although Explainable Reinforcement Learning (XRL) is emerging, debugging and interpreting DRL decision-making is challenging. A lack of understanding of DRL internal functions such as Experience Replay, used to sample and store data from the environment, risks it burdening resources. This paper contributes an XRL-based DQL system that uses SHAP (SHapley Additive exPlanations) values, which is a popular tool in XRL to explain how the input features contribute to model predictions. Data is sampled from the replay buffer instead of the environment, creating heatmaps to understand how inputs to the neural network Q-value approximator led to actions taken by the agent, given a state. The XRL system is used to determine for simulations of various complexities, how small can the Experience Replay buffer size be and why.

Keywords: Deep Reinforcement Learning · Experience Replay · SHAP.

1 Introduction

DRL can optimise complex control and decision-making processes. However, it lacks explainability, limiting its widespread use in regulated environments like manufacturing, finance and medicine, where rising cost, safety and ethical concerns exist. Experience Replay, is an internal DRL sampling technique, inspired by neurons during sleep, to break data correlation and stabilise deep off policy learning. Although Explainable Reinforcement Learning (XRL) is emerging, DQL is challenging to debug and interpret with inefficiencies burdening resources. SHAP values are a popular tools to explain model predictions. This paper creates an XRL-based system that produces SHAP heatmaps to explain how input samples from the experience replay buffer affect the actions taken by a DQL agent. These SHAP heatmaps are further used to investigate the impact of reducing the minimum buffer size on an agent's performance in simulations of varying complexity.

2 Related work

Previous studies by (Mnih et al., 2015; Sutton & Barto, 2018) introduced Deep Q-Learning (DQL), using the (Bellman, 1957) equation to optimise the control process in complex environments. The agent uses equation 1 through trial and error to learn the quality of taking an action in a given Markov Decision Making Process (MDP) to find an optimal policy that maximises its cumulative reward.

$$Q_t(s,a) = Q_{t-1}(s,a) + \alpha(R(s,a)\gamma Max_a Q(s',a') - Q_{t-1}(S,A)) \quad (1)$$

Agents explore in simulated environments that mimic real word problems, like the (Bellemare, Naddaf, Veness, & Bowling, 2013) Arcade Learning Environment, to provide much-needed training data safely. Evaluation of performance is comparing an agent to a handcrafted, human expert, or random policy. Approximating Q-values with a neural network in large and complex state environments destabilises learning. Mnih resolved this by using Experience Replay (Lin, 1992) to sample data and store from the environment for the approximator to later reuse. (Hayes et al., 2021), stated this emerged from observing biological neurons during sleep, primarily in the hippocampus. However, its drawbacks include samples leading to correlated data, buffer having limited capacity causing an agent to forget information, outdated samples due to non-stationery environments and buffer samples being memorised leading to overfitting. To solve these issues, several variations of Experience Replay were developed such as Prioritized Experience Replay (PER) (Schaul, Quan, Antonoglou, & Silver, 2016) and Attention based Experience Replay (Ramicic & Bonarini, 2017) Better understanding of Experience replay is crucial to improving its efficiency. Deepmind's Agent57 and MEME (Kapturowski et al., 2022) which beat human champions in Atari contained expensive data inefficiencies, Agent57 required 80 billion frames of experience to achieve optimal performance. Consequently many consider Experience Replay to be flawed with most wanting it replaced. Asynchronous Actor-Critic (A3C) by (Mnih et al., 2016) is a popular alternative that trains multiple agents in parallel, to explore the environment then update a shared network. It converges faster on an optimal policy but requires more resources training multiple agents. In contrast, Experience replay, although slower, is more memory efficient only requiring stored transitions not multiple copies of the network. (S. Zhang & Sutton, 2017) highlighted that the Experience Replay memory buffer size is a neglected hyperparameter that required tuning, showing a large replay buffer sizes hurting performance. (Bruin, Kober, Tuyls, & Babuška, 2018) stated the control problem determined the buffer size and to keep it high with a rule of thumb of 90% of total environment steps needed to reach final performance level. (Fedus et al., 2020) later revisited this due to its misunderstanding with most defaulting to Mnih's 1M transitions for the buffer size capacity. Experiments on the Atari Arcade Learning Environment, showed increasing experience replay buffer capacity from 1M to 10M transitions while also decreasing the age of the oldest policy did improve performance. It was not mentioned what the minimum size of the replay capacity should be. Any increase in buffer size further burdens resources. Explainability can help find the right size. Custom explainers can be created (Keramati, Durand, Girardeau, Gutkin, & Ahmed, 2017;

Miralles-Pechuán, Jiménez, Ponce, & Martinez-Villaseñor, 2020) but (Lundberg & Lee, 2017) developed SHAP (SHapley Additive exPlanations), to assign each feature an importance value for a particular prediction, to understand why a model makes certain predictions. Reviewing XRL (Heuillet, Couthouis, & Díaz-Rodríguez, 2021; Ras, Xie, van Gerven, & Doran, 2022; Vouros, 2022), noted SHAP was a popular chose to explain blackbox models. (Kumar, Vishal, & Ravi, 2022; Thirupathi, Alhanai, & Ghassemi, 2022; K. Zhang, Zhang, Xu, Gao, & Gao, 2022) had success applying it. (Liessner, Dohmen, & Wiering, 2021) RL-SHAP diagram is state of art showing different environment features and their effect on action selection. (Sovrano, Raymond, & Prorok, 2021) is state of art for Experience Replay explainability, by partitioning the experience buffer into clusters and labelling them on a per-explanation basis, to replace PER. This paper implements SHAP to improve explainability of experience replay aiming to reduce the replay buffer capacity in several simulations of varying complexity, one of which is a custom built Addiction simulator for gym environment.

3 Design

A DQL Agent was placed in three simulations; CartPole, LunarLander and a custom-built Addiction simulator. Assuming all hyperparameters held constant, the capacity was reduced 1 mill; 500k; 100k; 50k; 10k; 5k; 1k; and 500 transitions respectively. The null hypothesis being tested was no significant difference ($p<0.05$) between reward scores when experience replay buffer capacity was reduced. Agent's neural network included an input layer for observations, a hidden layer with 30 linear neurons, an action output layer and Q-value weights. The replay buffer held 100 batches of the previous state, next state, action selected and reward received, 10% used as shap training and testing data. Adam's Optimiser and SoftMax for exploration-exploitation policy was used with gamma at 0.9, learning rate at 0.001 and temperature at 100%. Hardware was constrained

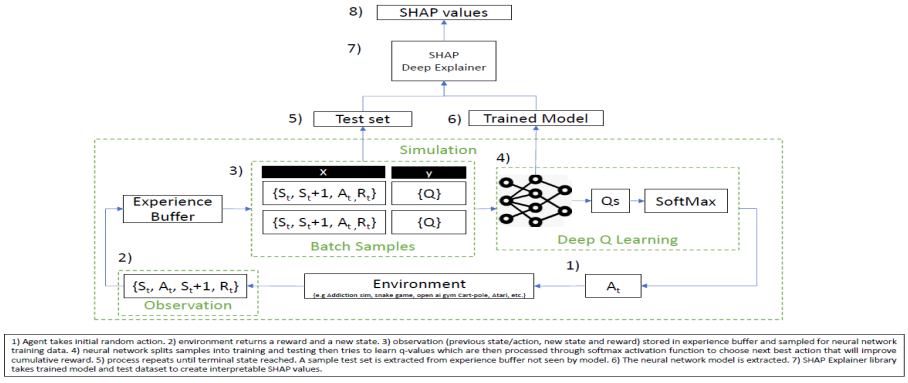

Fig. 1. Shap XAI-based DQL Architecture

to a windows laptop, with i7 4 cores, 8 threads, 2.8GHz CPU, 128GB SSD, 1TB HDD, 16 GB Ram, Nvidia GTX 1060 4GB GPU. CartPole and LunarLander,

ran for 100 episodes. The custom addiction simulator that mimicked a rat resting, lever pulling or taking cocaine ran for 3,600 episodes (4 hours). Reward was recorded and plotted on a line chart on the y axis while the simulation time or steps taken was plotted on the x axis, explaining HOW the agent performed. It and capacity sizes were also saved as a csv file to later perform ANOVA testing. A test sample was taken from the randomised Experience replay batch sampling that the Q-approximator neural network uses. This unseen sample, along with the trained model was passed to a SHAP Deep Explainer, where SHAPly values are generated. Shap values were also visualised to help understand WHY the agent took the action in that given state by seeing which batch inputs from experience replay led to the Q-values predicted.

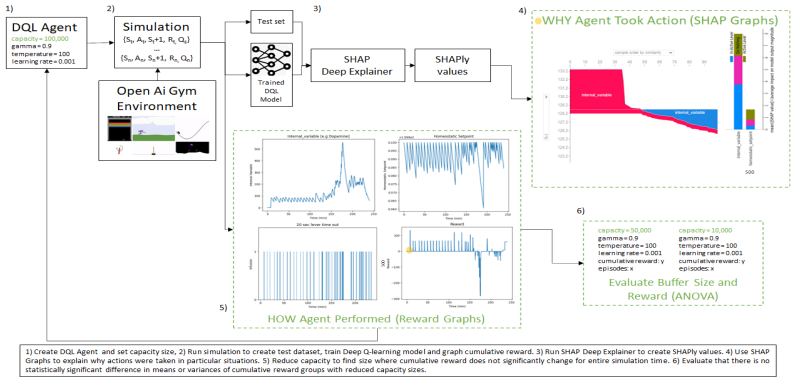

Fig. 2. Experiment Design Process

4 Results and discussion

For Addiction simulator, ANOVA result showed p-value below the 0.05 threshold, suggesting to reject the null hypothesis. A large F-score, supports that "capacity" has a significant effect on the "reward". SHAP value visualisation indicated lower capacity caused the agent to focus on the homeostatic setpoint instead of managing the internal variable (e.g dopamine levels), as addicts would do but at 1000 this was reversed. This indicates anything greater than 1000 and less than 1M transitions will affect the simulation but 500 to 1000 won't (<90%).

Table 1. Anova Result

	sum_sq	df	F	PR(>F)
C(capacity)	8.645837×10^7	7.0	2584.636845	0.0
Residual	1.376265×10^8	28800.0	NaN	NaN

5 Conclusion

In conclusion, the proposed XAI-based system using SHAP values can provide a more transparent, interpretable explanation of actions taken by a DQL agent. By visualizing the contribution of batch states sampled from experience replay, to each action, we gain insights into how the agent is learning and how it might be improved. In future work, Ethical considerations, trade-offs and limitations, comparison to other XRL methods, generalizability and interpretability of SHAP values with user surveys will be considered.

References

Bellemare, M. G., Naddaf, Y., Veness, J., & Bowling, M. (2013, may). The arcade learning environment: An evaluation platform for general agents. *J. Artif. Int. Res.*, *47*(1), 253–279.

Bellman, R. (1957). *Dynamic programming.* Dover Publications.

Bruin, T. D., Kober, J., Tuyls, K., & Babuška, R. (2018, 1). Experience selection in deep reinforcement learning for control. *J. Mach. Learn. Res.*, *19*, 347-402.

Fedus, W., Ramachandran, P., Agarwal, R., Bengio, Y., Larochelle, H., Rowland, M., & Dabney, W. (2020). Revisiting fundamentals of experience replay. JMLR.org.

Hayes, T. L., Krishnan, G. P., Bazhenov, M., Siegelmann, H. T., Sejnowski, T. J., & Kanan, C. (2021, 10). Replay in deep learning: Current approaches and missing biological elements. *Neural Computation*, *33*, 2908-2950. Retrieved from https://doi.org/10.1162/neco$_{a0}$1433 doi: https://doi.org/10.1162/neco$_{a0}$1433

Heuillet, A., Couthouis, F., & Díaz-Rodríguez, N. (2021). Explainability in deep reinforcement learning. *Knowledge-Based Systems*, *214*, 106685. Retrieved from https://www.sciencedirect.com/science/article/pii/S0950705120308145 doi: https://doi.org/https://doi.org/10.1016/j.knosys.2020.106685

Kapturowski, S., Campos, V., Jiang, R., Rakićević, N., van Hasselt, H., Blundell, C., & Badia, A. P. (2022). *Human-level atari 200x faster.*

Keramati, M., Durand, A., Girardeau, P., Gutkin, B., & Ahmed, S. H. (2017, 3). Cocaine addiction as a homeostatic reinforcement learning disorder. *Psychol. Rev.*, *124*, 130-153.

Kumar, S., Vishal, M., & Ravi, V. (2022). *Explainable reinforcement learning on financial stock trading using shap.*

Liessner, R., Dohmen, J., & Wiering, M. (2021). Explainable reinforcement learning for longitudinal control. In A. P. Rocha, L. Steels, & J. van den Herik (Eds.), (p. 874-881). SciTePress. (Publisher Copyright: © 2021 by SCITEPRESS - Science and Technology Publications, Lda.; 13th International Conference on Agents and Artificial Intelligence, ICAART 2021 ; Conference date: 04-02-2021 Through 06-02-2021)

Lin, L.-J. (1992, 5). Self-improving reactive agents based on reinforcement learning, planning and teaching. *Mach. Learn.*, *8*, 293-321. Retrieved from https://doi.org/10.1007/BF00992699 doi: https://doi.org/10.1007/BF00992699

Lundberg, S. M., & Lee, S.-I. (2017). A unified approach to interpreting model predictions. In (p. 4768-4777). Curran Associates Inc.

Miralles-Pechuán, L., Jiménez, F., Ponce, H., & Martinez-Villaseñor, L. (2020). A methodology based on deep q-learning/genetic algorithms for optimizing covid-19 pandemic government actions. In (p. 1135-1144). Association for Computing Machin-

ery. Retrieved from https://doi.org/10.1145/3340531.3412179 doi: https://doi.org/10.1145/3340531.3412179

Mnih, V., Badia, A. P., Mirza, M., Graves, A., Lillicrap, T., Harley, T., ... Kavukcuoglu, K. (2016, 11). Asynchronous methods for deep reinforcement learning. In M. F. Balcan & K. Q. Weinberger (Eds.), (Vol. 48, p. 1928-1937). PMLR. Retrieved from https://proceedings.mlr.press/v48/mniha16.html

Mnih, V., Kavukcuoglu, K., Silver, D., Rusu, A. A., Veness, J., Bellemare, M. G., ... Hassabis, D. (2015). Human-level control through deep reinforcement learning. *Nature*, *518*, 529-533. Retrieved from https://doi.org/10.1038/nature14236 doi: https://doi.org/10.1038/nature14236

Ramicic, M., & Bonarini, A. (2017). Attention-based experience replay in deep q-learning. In (p. 476-481). Association for Computing Machinery. Retrieved from https://doi.org/10.1145/3055635.3056621 doi: https://doi.org/10.1145/3055635.3056621

Ras, G., Xie, N., van Gerven, M., & Doran, D. (2022, 5). Explainable deep learning: A field guide for the uninitiated. *J. Artif. Int. Res.*, *73*. Retrieved from https://doi.org/10.1613/jair.1.13200 doi: https://doi.org/10.1613/jair.1.13200

Schaul, T., Quan, J., Antonoglou, I., & Silver, D. (2016). Prioritized experience replay. In Y. Bengio & Y. LeCun (Eds.), *4th international conference on learning representations, ICLR 2016, san juan, puerto rico, may 2-4, 2016, conference track proceedings.*

Sovrano, F., Raymond, A., & Prorok, A. (2021). Explanation-aware experience replay in rule-dense environments. *CoRR*, *abs/2109.14711*. Retrieved from https://arxiv.org/abs/2109.14711

Sutton, R. S., & Barto, A. G. (2018). *Reinforcement learning: An introduction* (Second ed.). The MIT Press. Retrieved from http://incompleteideas.net/book/the-book-2nd.html

Thirupathi, A. N., Alhanai, T., & Ghassemi, M. M. (2022). A machine learning approach to detect early signs of startup success. Association for Computing Machinery. Retrieved from https://doi.org/10.1145/3490354.3494374 doi: https://doi.org/10.1145/3490354.3494374

Vouros, G. A. (2022, 12). Explainable deep reinforcement learning: State of the art and challenges. *ACM Comput. Surv.*, *55*. Retrieved from https://doi.org/10.1145/3527448 doi: https://doi.org/10.1145/3527448

Zhang, K., Zhang, J., Xu, P.-D., Gao, T., & Gao, D. W. (2022). Explainable ai in deep reinforcement learning models for power system emergency control. *IEEE Transactions on Computational Social Systems*, *9*, 419-427. doi: https://doi.org/10.1109/TCSS.2021.3096824

Zhang, S., & Sutton, R. (2017, 12). A deeper look at experience replay.