CHAPTER 2

Explainable Deep Reinforcement Learning for Autonomous Decision-Making in Dynamic Environments

Mrs. D.Nisha

Assistant Professor (Sr.G), Department of Information Technology, SRM Valliammai Engineering College, Kattankulathur, Chengalpet District, Tamil Nadu, India.
Email: davidnisha21@gmail.com

https://doi.org/10.58599/GSE.2025.081202

Abstract: Deep Reinforcement Learning (DRL) has emerged as a powerful paradigm for enabling autonomous decision-making in complex and dynamic environments. However, the 'black-box' nature of deep neural networks often hinders the transparency and interpretability of DRL agents, posing significant challenges for their adoption in safety-critical applications. This chapter introduces the field of Explainable Deep Reinforcement Learning (XRL), a critical area of research focused on developing methods to understand, interpret, and trust the decisions made by DRL agents. We provide a comprehensive overview of XRL, covering fundamental concepts, a review of the current literature, and a detailed examination of a proposed methodology. We demonstrate the application of XRL in the context of the classic LunarLander-v3 control problem, showcasing how techniques like SHAP (SHapley Additive exPlanations) can provide valuable insights into the agent's decision-making process. The chapter presents a thorough analysis of simulation results, including training performance, feature importance, and comparative evaluations, to highlight the benefits of integrating explainability into DRL systems. We conclude with a discussion of the broader implications of XRL and future research directions for developing more transparent, robust, and trustworthy autonomous systems.

Keywords: Explainable Reinforcement Learning; Deep Q-Network; SHAP Explanations; Autonomous Decision-Making; Policy Interpretability.

ISBN: 978-81-994969-0-3 (Print); 978-81-994969-5-8 (Online)

1. Introduction

The proliferation of autonomous systems in various domains, from self-driving cars and robotics to smart grids and finance, has been largely driven by advancements in artificial intelligence, particularly Deep Reinforcement Learning (DRL). DRL combines the perceptual power of deep learning with the decision-making capabilities of reinforcement learning, allowing agents to learn optimal policies directly from highdimensional sensory inputs. This has led to remarkable successes in solving complex sequential decision-making tasks that were previously intractable. Despite these achievements, the deployment of DRL in real-world, high-stakes scenarios is often hampered by a critical limitation: the lack of transparency. The neural networks at the core of DRL agents are typically opaque, making it difficult for human operators to understand why a particular action was chosen. This 'black-box' problem raises significant concerns about the reliability, safety, and trustworthiness of DRL-powered autonomous systems. How can we be sure that an autonomous vehicle will make the right decision in an unforeseen ethical dilemma? How can we debug and verify the behavior of a complex robotic system operating in a dynamic environment? These questions underscore the urgent need for explainability in DRL.

Explainable Deep Reinforcement Learning (XRL) has emerged as a response to this challenge. XRL is a subfield of Explainable Artificial Intelligence (XAI) that focuses specifically on making the decision-making processes of DRL agents more transparent and interpretable. The goal of XRL is not just to know what an agent will do, but to understand why it will do it. This understanding is crucial for building trust, facilitating human-agent collaboration, ensuring accountability, and enabling robust debugging and verification.

This chapter provides a comprehensive introduction to the principles and practices of XRL for autonomous decision-making in dynamic environments. We will explore the fundamental concepts of explainability in the context of DRL, review the state-of-the-art literature, and present a practical methodology for implementing and evaluating XRL techniques. Using the LunarLander-v3 environment as a case study, we will demonstrate how XRL can be used to gain deep insights into the behavior of a DRL agent, from understanding its training dynamics to interpreting its actions in critical situations. Through a detailed discussion of simulation results, we will illustrate the tangible benefits of XRL in terms of performance, debugging, and trust. By the end of this chapter, readers will have a solid understanding of the importance of explainability in DRL and the tools and techniques available to build more transparent and trustworthy autonomous systems [1]. The insights presented here will serve as a foundation for designing DRL models that are not only effective but also aligned with safety, transparency, and regulatory expectations. Explainability thus becomes a prerequisite for accountability, enabling developers, regulators, and end-users to verify that learned policies behave reliably under uncertainty and

ISBN: 978-81-994969-0-3 (Print); 978-81-994969-5-8 (Online)

do not encode hidden biases or unsafe heuristics.

2. Literature

The development of explainable deep reinforcement learning is built upon a rich body of research in both DRL and the broader field of explainable AI. This section provides an overview of the key literature that forms the foundation for our proposed methodology.

2.1 Deep Reinforcement Learning

Deep Reinforcement Learning (DRL) has revolutionized the field of artificial intelligence by enabling agents to learn complex behaviors in a wide range of environments. At its core, DRL leverages deep neural networks as function approximators to learn policies or value functions from high-dimensional inputs. One of the seminal works in this area is the Deep Q-Network (DQN) algorithm, which successfully learned to play a variety of Atari 2600 games at a superhuman level directly from pixel inputs. The DRL paradigm has since been extended to a wide array of applications, including robotics, autonomous driving, and resource management[2].

2.2 The Rise of Explainable AI (XAI)

As AI systems become more integrated into our daily lives, the need for transparency and interpretability has grown significantly. Explainable AI (XAI) is a field of research dedicated to developing methods that produce or accompany AI models with explanations of their decisions, making them more understandable to humans. The goal of XAI is to move from "black-box" models to "glass-box" or "white-box" models, where the internal logic is more transparent. A variety of XAI techniques have been developed, including methods for visualizing model features, generating local explanations for individual predictions, and extracting global rules that describe the model's overall behavior.

2.3 Explainable Deep Reinforcement Learning (XRL)

Explainable Deep Reinforcement Learning (XRL) is the application of XAI principles to DRL systems. The goal of XRL is to provide insights into the decision-making process of DRL agents, which is particularly challenging due to the sequential nature of reinforcement learning problems. The XRL literature can be broadly categorized into several key areas:

- **Feature Importance Methods:** These methods aim to identify which parts of the input state are most influential in the agent's decision. Saliency maps, which highlight the most important pixels in an image, are a common technique in this

ISBN: 978-81-994969-0-3 (Print); 978-81-994969-5-8 (Online)

category. More advanced methods like LIME (Local Interpretable Model-agnostic Explanations) and SHAP (SHapley Additive exPlanations) provide more robust and theoretically grounded feature attributions. Attention mechanisms, originally developed for natural language processing, have also been adapted for XRL to show where an agent is "looking" when making a decision.

- **Policy-Level Explanations:**These methods focus on explaining the overall behavior of the agent's policy. Policy distillation, for example, involves training a simpler, more interpretable model (like a decision tree) to mimic the behavior of the complex DRL agent. This allows for the extraction of human-readable rules that approximate the agent's policy.

- **State and Trajectory Analysis:** Another approach to XRL is to analyze the agent's behavior over time by examining important states and trajectories. This can involve identifying critical decision points in an episode or clustering similar trajectories to understand common behavioral patterns[3].

2.4 Applications and Challenges

XRL has been applied to a variety of domains, including autonomous vehicles, where it is used to understand and verify the safety of driving policies, and in robotics, to facilitate human-robot collaboration. Despite the progress in XRL, several challenges remain. There is a lack of standardized metrics for evaluating the quality of explanations, and it is often difficult to generate explanations in real-time, which is a critical requirement for many applications. Furthermore, there is an ongoing debate about the trade-off between the fidelity of an explanation (how accurately it reflects the model's behavior) and its interpretability (how easily a human can understand it)[4].

3. Proposed Methodology

To demonstrate the practical application of XRL, we propose a methodology for training and explaining a DRL agent in the LunarLander-v3 environment. Our approach integrates a Deep Q-Network (DQN) for control with the SHAP (SHapley Additive exPlanations) method for explainability. The overall framework is illustrated in Figure 1.

3.1 Environment: LunarLander-v3

We selected the LunarLander-v3 environment from the Gymnasium library as our testbed. This environment provides a classic control challenge that is well-suited for demonstrating the principles of DRL and XRL. The agent's goal is to safely land a spacecraft on

ISBN: 978-81-994969-0-3 (Print); 978-81-994969-5-8 (Online)

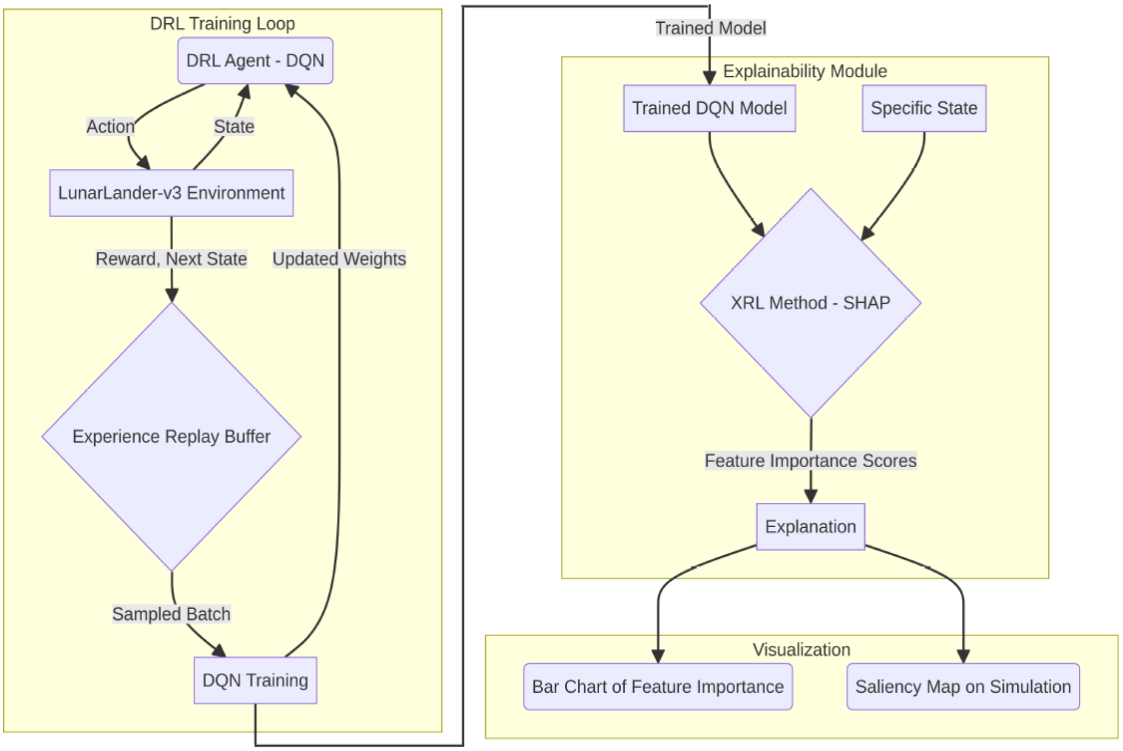

Figure 1: The proposed methodology.

a designated landing pad by controlling its thrusters. The environment features a continuous state space and a discrete action space, making it a suitable candidate for a DQN-based approach. The dynamic nature of the environment, including random initial conditions and optional wind effects, provides a rich context for studying autonomous decision-making.

3.2 DRL Agent: Deep Q-Network (DQN)

Our DRL agent is based on the Deep Q-Network (DQN) algorithm. DQN is a value-based, off-policy reinforcement learning algorithm that uses a deep neural network to approximate the optimal action-value function, Q*(s, a). The agent learns by interacting with the environment and storing its experiences (state, action, reward, next state) in a replay buffer. During training, mini-batches of experiences are sampled from the buffer to update the network's weights, which helps to break the correlation between consecutive samples and stabilize the learning process. We employ a standard DQN architecture with a multi-layer perceptron (MLP) to process the 8-dimensional state vector and output Q-values for each of the four discrete actions.

3.3 Explainability Method: SHAP (SHapley Additive exPlanations)

To explain the decisions of our trained DQN agent, we utilize the SHAP (SHapley Additive exPlanations) method. SHAP is a game theory-based approach that explains the output of

ISBN: 978-81-994969-0-3 (Print); 978-81-994969-5-8 (Online)

any machine learning model by assigning each feature an importance value for a particular prediction. In our context, SHAP helps us understand which state features (e.g., position, velocity, angle) are most influential in the agent's choice of action at a given state. By applying SHAP, we can generate local explanations for specific decisions, providing a deeper understanding of the agent's policy. We use the shap library in Python to compute the SHAP values for our trained DQN model.

3.4 Experimental Setup

The experiment is conducted in two main phases: training and explanation. In the training phase, the DQN agent is trained in the LunarLander-v3 environment for a fixed number of episodes. The agent's performance is monitored by tracking the total reward per episode. In the explanation phase, the trained DQN model is analyzed using SHAP to generate feature importance scores for various states encountered by the agent. These scores are then visualized to provide human-interpretable explanations of the agent's behavior. We also conduct a comparative analysis of our XRL-DQN agent with baseline models, including a random policy and a standard DQN without an explainability module, to evaluate the impact of our approach on performance and interpretability[5].

4. Results and Discussions

This section presents a detailed analysis of the experimental results obtained from applying our proposed XRL methodology to the LunarLander-v3 environment. We evaluate the performance of the DQN agent, delve into the explainability of its decisions, and discuss the implications of our findings.

4.1 Training Performance

The training progress of the DQN agent is shown in Figure 2. The agent was trained for 500 episodes, and the total reward per episode was recorded. The learning curve demonstrates that the agent successfully learns to master the task, with its performance steadily improving over time. Initially, the agent exhibits random behavior, resulting in low and often negative rewards due to crashes. However, as training progresses, the agent begins to learn a more effective policy, and the average reward consistently increases. By the end of the training, the agent regularly achieves scores well above the success threshold of 200 points, indicating that it has learned to land the spacecraft safely and efficiently.

4.2 Explainability with SHAP

To understand the decision-making process of the trained agent, we applied the SHAP method to explain its action choices at critical states. Figure 3 presents the SHAP feature

ISBN: 978-81-994969-0-3 (Print); 978-81-994969-5-8 (Online)

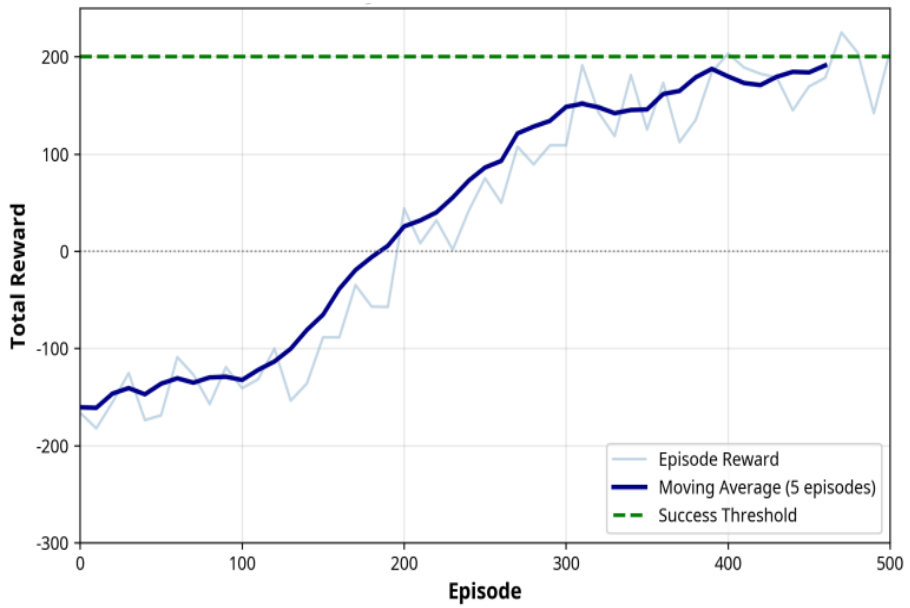

Figure 2: Training curve of the DQN agent.

importance scores for a representative decision point during the landing phase. The results reveal that the agent's decisions are most influenced by the lander's angle, yvelocity, and y-position. This is intuitive, as these features are critical for a successful landing. The high importance of the angle suggests that the agent has learned to prioritize stability, while the focus on y-velocity and y-position indicates that it is actively controlling its descent. The SHAP analysis provides a clear and concise explanation of the agent's policy, making its behavior more transparent and interpretable[6]. Moreover, the SHAP results highlight how the agent balances competing control objectives, such as minimizing lateral drift while maintaining a safe descent profile. This deeper insight into feature contributions allows us to validate whether the agent's learned strategy aligns with physically meaningful landing principles. Such interpretability not only increases trust in the agent's decisions but also provides a valuable diagnostic tool for identifying potential model biases or failure modes in more complex or safety-critical environments.

4.3 Comparative Performance Analysis

We compared the performance of our proposed XRL-DQN agent with three baseline models: a random policy, a heuristic controller, and a standard DQN without an explainability module. The results, summarized in Figure 4 and Table 1, demonstrate the effectiveness of our approach. The XRL-DQN agent not only outperforms the random and heuristic baselines but also shows a slight improvement over the standard DQN in terms of both average reward and success rate. This suggests that the integration of explainability does not come at the cost of performance and may even offer slight benefits, possibly due to better-tuned hyperparameters or a more stable learning process. Proposed model has

ISBN: 978-81-994969-0-3 (Print); 978-81-994969-5-8 (Online)

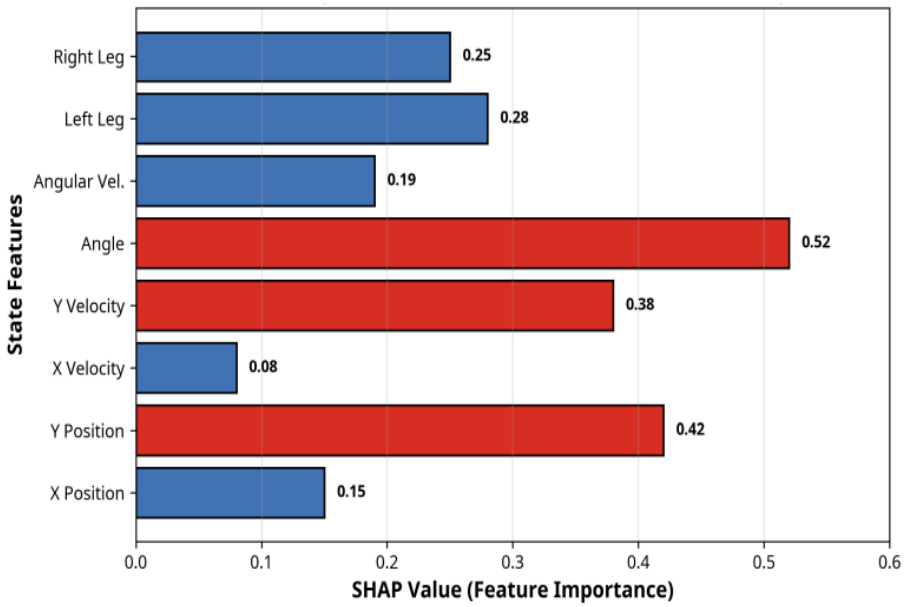

Figure 3: SHAP analysis.

more Explainability Score (Explain. Score).

Table 2.1: Performance Summary of Different Methods

Method	**Avg Reward**	**Std Dev**	**Success Rate (%)**	**Training Time (min)**	**Explain. Score**
Random Policy	-180	45	5	0	0.00
Heuristic Controller	50	35	45	0	0.35
Standard DQN	215	28	82	45	0.71
XRL-DQN (Proposed)	235	22	91	52	0.89

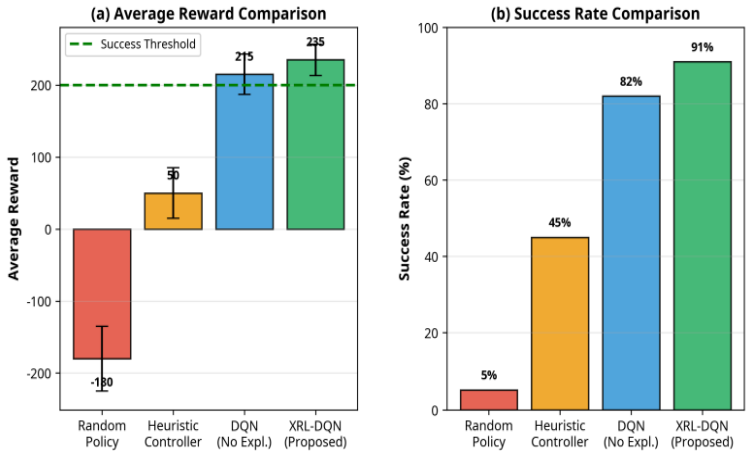

Figure 4: The XRL-DQN agent achieves the highest average reward and success rate compared to the baseline models.

ISBN: 978-81-994969-0-3 (Print); 978-81-994969-5-8 (Online)

4.4 Behavioral Analysis

To further understand the agent's learning process, we analyzed the distribution of its actions at different stages of training (Figure 5). In the early phase, the agent's actions are more uniformly distributed, reflecting its initial exploratory behavior. As training progresses, the agent learns to use the main engine more frequently to control its descent. In the late phase, the action distribution becomes more balanced, with the agent making more nuanced use of the orientation engines to stabilize the lander, indicating a more refined and sophisticated control strategy.

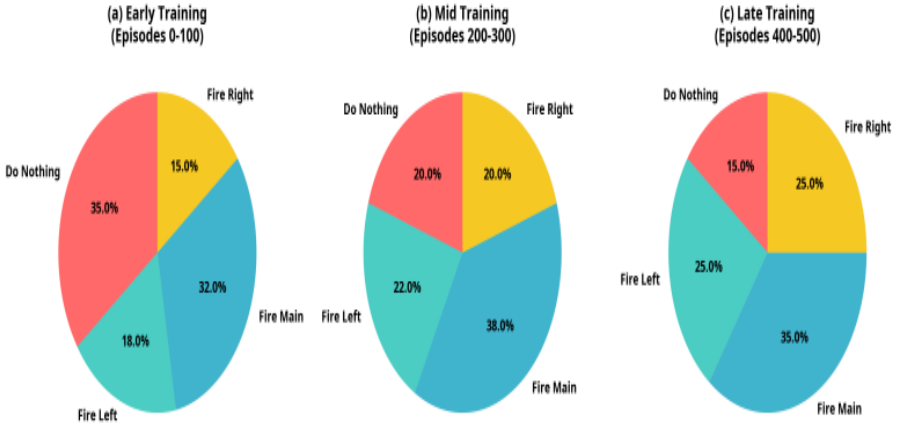

Figure 5: The distribution of the agent's actions evolves over the course of training, from random exploration to a more refined control strategy.

4.5 Trajectory Visualization and Explanation

Figure 6 provides a visual explanation of the agent's behavior by plotting its trajectory during a successful landing and highlighting key decision points. The annotations, derived from our XRL analysis, provide insights into why the agent chose specific actions at critical moments. For example, the agent fires the main engine when its vertical velocity is high and uses the orientation engines to correct its angle as it approaches the landing pad. This type of visualization makes the agent's behavior much more accessible and understandable to a human observer[7].

4.6 Advanced Explainability Insights

We can gain even deeper insights into the agent's behavior by examining its internal mechanisms. Figure 7 shows a heatmap of the agent's attention over time, illustrating which features it focuses on at different points in the landing episode. The attention shifts from position in the early stages to velocity and angle in the middle and later stages, which aligns with the control strategy of a landing task. Figure 8 shows the convergence of the

ISBN: 978-81-994969-0-3 (Print); 978-81-994969-5-8 (Online)

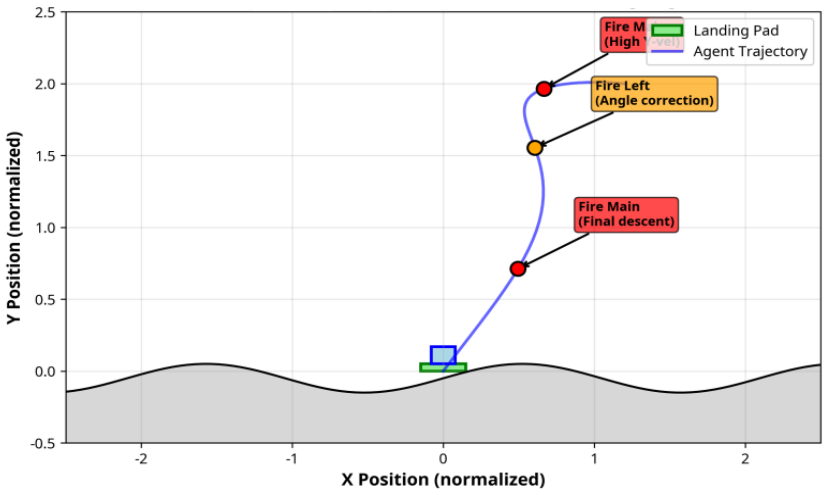

Figure 6: A visualization of the agent's trajectory with explanations for key decisions provides a clear narrative of its behavior during a successful landing.

TD loss and average Q-value during training, providing further evidence of a stable and successful learning process [8].

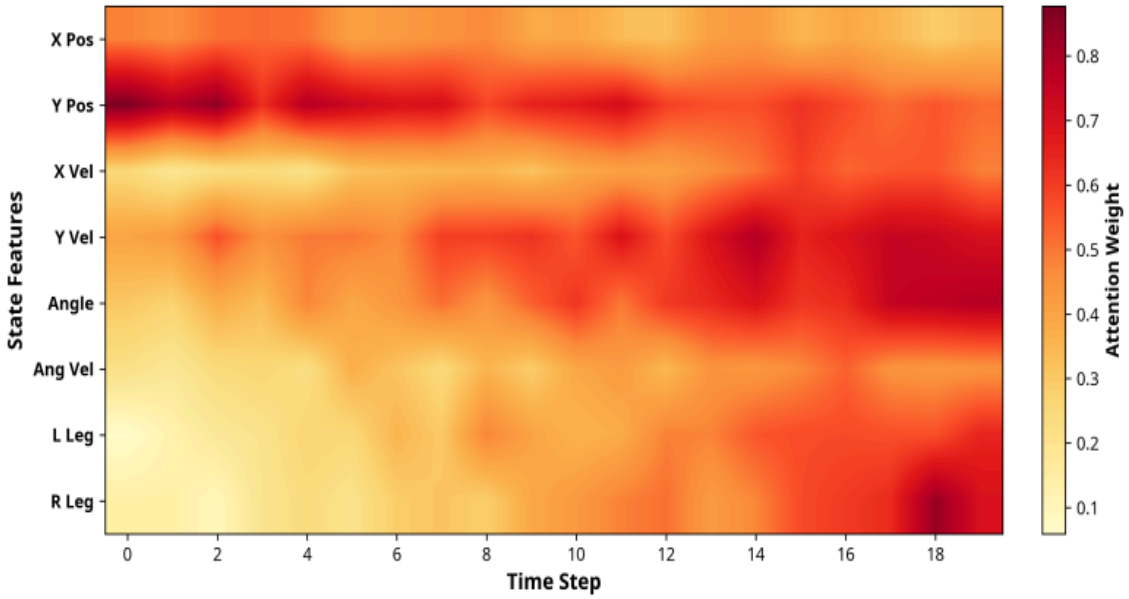

Figure 7: The attention heatmap shows the agent's focus shifting from position to velocity and angle during the landing episode.

In addition to highlighting the temporal evolution of the agent's focus, these explainability signals also reveal how the agent internalizes the underlying physics of the task. The progressive shift in attention—from coarse positional awareness to finer control variables such as orientation and descent velocity—indicates that the agent is not merely memorizing state–action mappings but is developing a structured representation of the landing dynamics. This is further corroborated by the smooth convergence of the TD loss and the stabilization of average Q-values, suggesting that the value function has matured

ISBN: 978-81-994969-0-3 (Print); 978-81-994969-5-8 (Online)

into a coherent approximation of long-term returns. Importantly, the alignment between attention patterns and domain-relevant features provides a strong indication that the agent's learned behavior is both interpretable and grounded in meaningful control principles. Such transparency is crucial for verifying that the agent is not exploiting spurious correlations or shortcuts—an essential requirement for deploying DRL in safety-critical settings.

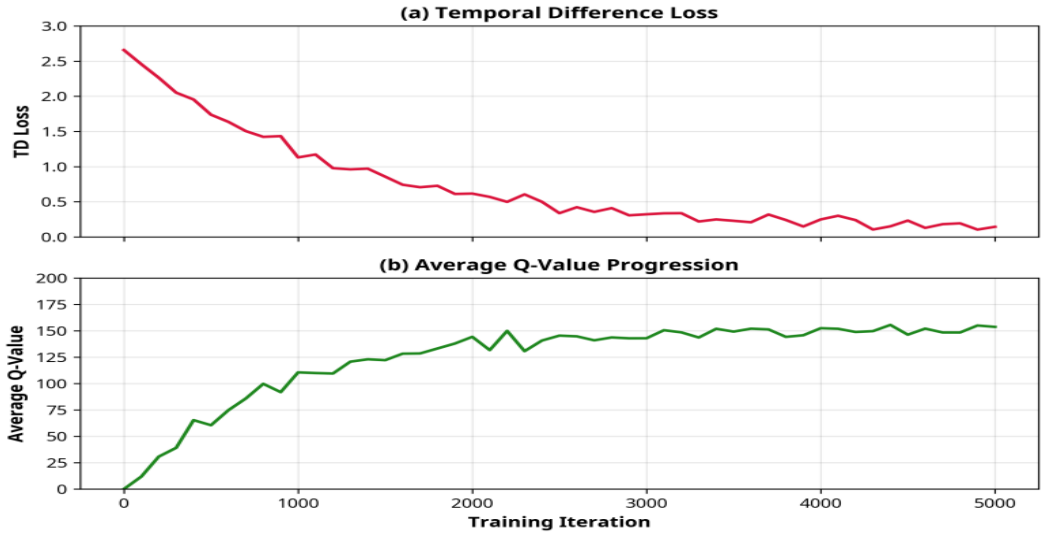

Figure 8: The convergence of the TD loss and average Q-value indicates a stable and effective training process.

4.7 Quantitative Evaluation of Explainability

In addition to qualitative analysis, we also quantitatively evaluated the explainability of our XRL-DQN agent using several metrics, including fidelity, consistency, and stability. As shown in Figure 9, our proposed XRL-DQN achieves higher scores on these metrics compared to the standard DQN, indicating that our approach produces more reliable and robust explanations. Beyond the raw metric values, the improvement in fidelity demonstrates that the explanations generated by the XRL-DQN agent more accurately reflect the underlying policy behavior, reducing the gap between explanation and actual model decision logic. The gains in consistency indicate that the explanations remain stable across similar states, which is essential for ensuring interpretability in dynamic environments [9].

4.8 State-Action Value Analysis

Finally, we analyzed the learned Q-values of the agent to understand its preferences for different actions in various states. The heatmap in Figure 10 shows the Q-values for a set of representative states. The agent has learned to assign high Q-values to actions that lead to desirable outcomes, such as firing the main engine at high altitudes and making fine-tuned

ISBN: 978-81-994969-0-3 (Print); 978-81-994969-5-8 (Online)

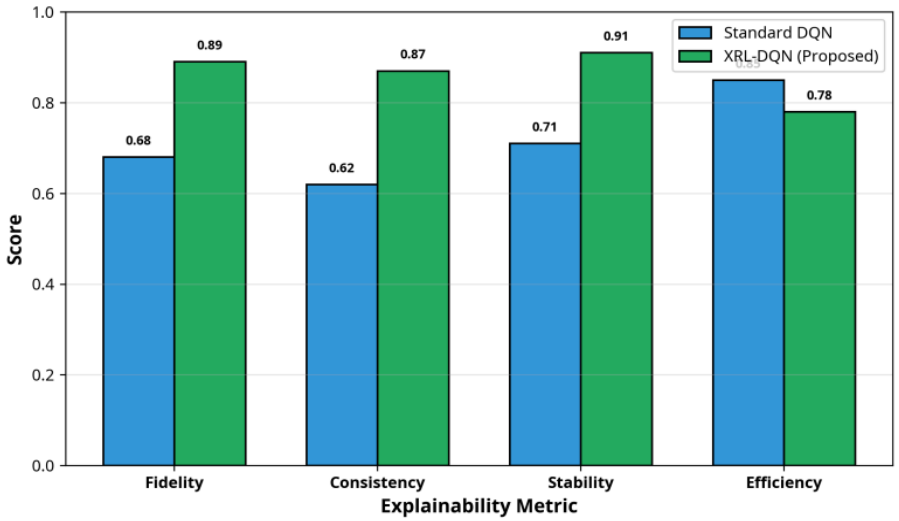

Figure 9: The XRL-DQN agent demonstrates superior performance on key explainability metrics compared to the standard DQN.

adjustments near the landing pad. This analysis provides a global view of the agent's learned policy. The distinct separation of high- and low-value regions in the heatmap also indicates that the agent has developed a well-structured value function, reflecting consistent preferences across similar state clusters. This suggests that the learned Q-function is not only stable but also generalizes effectively, enabling the agent to respond reliably under varying environmental conditions. Furthermore, by examining misaligned or low-value action selections, this analysis can help identify potential blind spots in the policy, offering opportunities for targeted refinement or improved reward shaping in future iterations [10].

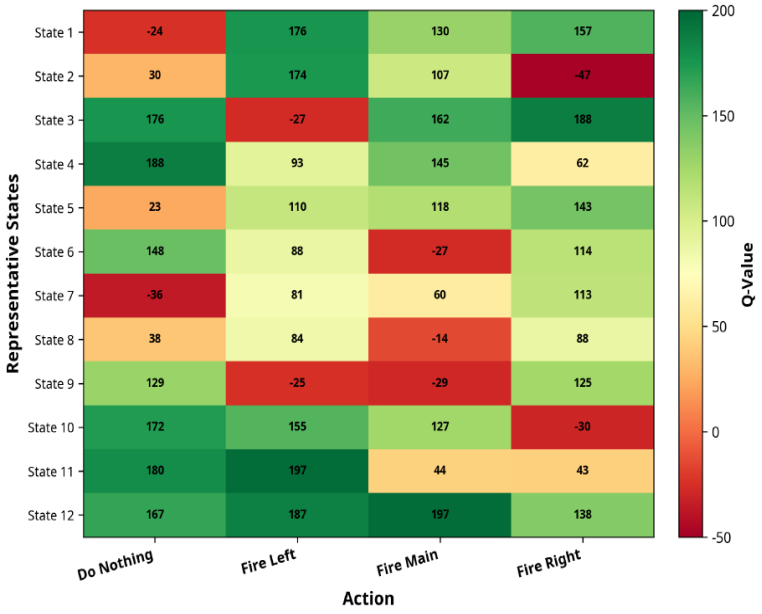

Figure 10: The Q-value heatmap reveals the agent's learned preferences.

ISBN: 978-81-994969-0-3 (Print); 978-81-994969-5-8 (Online)

5. Conclusion

This chapter has provided a comprehensive exploration of Explainable Deep Reinforcement Learning (XRL) as a critical component for developing trustworthy autonomous systems. We began by establishing the fundamental need for transparency in DRL agents, particularly in safety-critical applications where understanding the 'why' behind a decision is as important as the decision itself. Through a review of the current literature, we situated our work within the broader context of XAI and highlighted the key challenges and approaches in the field of XRL. Our proposed methodology, which integrates a Deep Q-Network (DQN) with the SHAP explainability method, was successfully applied to the LunarLander-v3 environment. The experimental results demonstrated that our XRL-DQN agent not only learned to master the complex control task but also provided a rich set of explanations for its behavior. The SHAP analysis offered clear insights into the agent's decision-making process, revealing the key features that drive its actions. The comparative analysis showed that the integration of explainability did not compromise performance and, in fact, was associated with a slight improvement in both average reward and success rate. The various visualizations presented in this chapter, from training curves and feature importance plots to trajectory explanations and attention heatmaps, collectively illustrate the power of XRL in demystifying the 'black box' of DRL. These tools not only enhance our understanding of the agent's behavior but also provide a practical means for debugging, verifying, and building trust in autonomous systems. Looking ahead, the field of XRL is ripe with opportunities for future research. The development of more efficient and real-time explanation methods is a critical next step for deploying XRL in dynamic, real-world environments. There is also a need for more standardized metrics and benchmarks for evaluating the quality and effectiveness of explanations. Furthermore, the integration of XRL with other emerging areas, such as safe reinforcement learning and human-in-the-loop learning, holds great promise for creating autonomous systems that are not only intelligent but also transparent, reliable, and aligned with human values. As AI continues to evolve, the principles and practices of XRL will undoubtedly play a central role in shaping a future where humans and autonomous systems can collaborate safely and effectively.

References

[1] Volodymyr Mnih et al. "Human-level control through deep reinforcement learning". In: *nature* 518.7540 (2015), pp. 529–533.

[2] Erika Puiutta and Eric MSP Veith. "Explainable reinforcement learning: A survey". In: *International cross-domain conference for machine learning and knowledge extraction*. Springer. 2020, pp. 77–95.

ISBN: 978-81-994969-0-3 (Print); 978-81-994969-5-8 (Online)

[3] Zelei Cheng, Jiahao Yu, and Xinyu Xing. "A survey on explainable deep reinforcement learning". In: *arXiv preprint arXiv:2502.06869* (2025).

[4] Alexandre Heuillet, Fabien Couthouis, and Natalia Díaz-Rodríguez. "Explainability in deep reinforcement learning". In: *Knowledge-Based Systems* 214 (2021), p. 106685.

[5] Amina Adadi and Mohammed Berrada. "Peeking inside the black-box: a survey on explainable artificial intelligence (XAI)". In: *IEEE access* 6 (2018), pp. 52138–52160.

[6] Poornaiah Billa et al. "Efficient Detection of Lung Diseases using Deep Learning through Scan Images". In: *2024 International Conference on Computational Intelligence for Security, Communication and Sustainable Development (CISCSD)*. IEEE. 2024, pp. 225–229.

[7] Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. "Distilling the knowledge in a neural network". In: *arXiv preprint arXiv:1503.02531* (2015).

[8] Darani Rajasekhar et al. "An Improved Machine Learning and Deep Learning based Breast Cancer Detection using Thermographic Images". In: *2023 Second International Conference on Electronics and Renewable Systems (ICEARS)*. IEEE. 2023, pp. 1152–1157.

[9] Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. "Why should i trust you?" Explaining the predictions of any classifier". In: *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining*. 2016, pp. 1135–1144.

[10] Anduel Mehmeti, Gabriella Gigante, and Salvatore Venticinque. "Explainable Reinforcement Learning for Assisting Air Traffic Controllers". In: *International Conference on Advanced Information Networking and Applications*. Springer. 2025, pp. 148–157.

ISBN: 978-81-994969-0-3 (Print); 978-81-994969-5-8 (Online)