Comparison of AI Strategies for the Taxi-v3 Environment

INFO-H410 Techniques of artificial intelligence
*Michele LEGGIERI, Thienlong LY, Hongdu WEI

***Abstract*—This report provides a comprehensive comparison of three AI strategies—reinforcement learning (RL), deep learning, and a genetic algorithm combined with reinforcement learning (GA + RL)—within the Taxi-v3 environment from OpenAI Gym. Over 1000 generations, we analyze each approach's performance, examining factors such as learning stability, generalization, and hyperparameter impact. The differences in performance are discussed, and the best-performing model's simulation is visualized to demonstrate its effectiveness in navigating the Taxi-v3 environment efficiently.**

***Index Terms*—Reinforcement learning, deep learning, genetic algorithm, genetic algorithm conbined with reinforcement learning, Taxi-v3 environment, gymnasium, simulation.**

I. Introduction

Reinforcement learning (RL) and deep learning are two prominent and powerful methods in artificial intelligence, each bringing unique strengths to problem-solving. RL excels in environments where an agent learns to make sequences of decisions by receiving rewards or penalties, effectively optimizing its actions to maximize cumulative rewards over time. Deep learning, on the other hand, leverages neural networks to model complex patterns and representations, making it particularly effective in handling large and high-dimensional datasets.

In this project, we conduct a detailed comparison of traditional RL, deep learning, and a hybrid approach that combines a genetic algorithm (GA) with RL. The goal is to solve the Taxi-v3 problem, a well-known challenge within the OpenAI Gym framework. The Taxi-v3 environment presents a discrete, grid-based scenario where an autonomous agent (the taxi) must navigate to pick up passengers from various locations and efficiently drop them off at designated destinations. The task requires the agent to plan its moves strategically, considering the optimal path and timing to minimize travel distance and avoid penalties.

By implementing and evaluating these three approaches—traditional RL, deep learning, and GA + RL—we aim to uncover their relative advantages and potential synergies. Traditional RL methods, such as Q-learning, provide a straightforward way to learn optimal policies through interaction with the environment. Deep learning approaches enhance this by using neural networks to approximate value functions or policies, allowing for greater flexibility and scalability. The hybrid GA + RL method introduces evolutionary strategies to the learning process, using genetic algorithms to explore a broader solution space and refine policies over generations.

Through rigorous experimentation and analysis, we assess the performance of each method in terms of efficiency, learning speed, and robustness. The comparison highlights not only which method performs best in the Taxi-v3 environment but also provides insights into the mechanisms behind their success. This comprehensive evaluation helps in understanding how different AI techniques can be effectively applied to solve complex, dynamic problems in discrete environments.

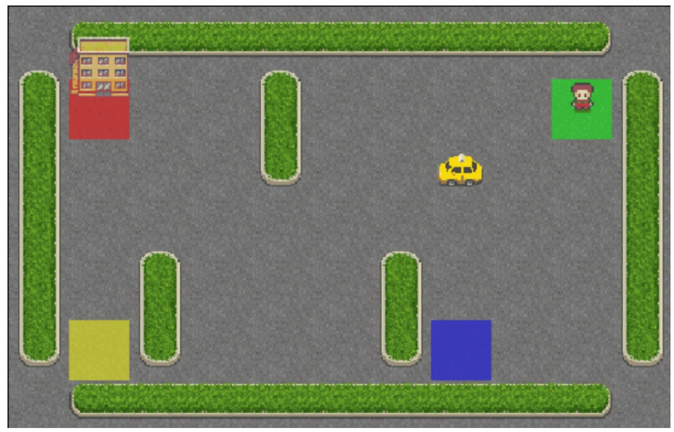

Fig. 1. Taxi-v3 environment

II. Methodology

We implemented three different methods to train the taxi agent, with each method representing an increase in algorithmic complexity. We started with "basic reinforcement learning," which uses traditional techniques like Q-learning to help the agent learn optimal actions through trial and error. Next, we progressed to a "deep learning algorithm using a neural network," which allowed the agent to handle more complex decision-making processes by approximating value functions with neural networks. Finally, we employed a "deep learning algorithm with a neural network supported by a genetic algorithm," combining evolutionary strategies with deep learning to enhance the agent's ability to explore and optimize its actions effectively.

A. Reinforcement Learning Model

The reinforcement learning model aims to apply traditional Q-learning techniques to optimize the taxi's performance in the Taxi-v3 environment. The hyperparameters for this model include a learning rate of 0.01, a discount rate of 0.99, and an initial epsilon value of 1.0, which decays to a minimum of 0.01. The training spans 1000 episodes. The Q-learning approach involves maintaining a Q-table that records the expected future rewards for each action taken in a given state. As the agent explores the environment, it updates the Q-values based on the maximum expected future rewards. This iterative process helps the agent to progressively improve its policy, ultimately learning to make decisions that maximize cumulative rewards.

B. Deep Learning Model

The objective of the deep learning model is to train a neural network to optimize the taxi's performance in the Taxi-v3 environment. The hyperparameters used for this model include a learning rate of 0.01, a discount rate of 0.99, and an initial epsilon value of 1.0, which decays to a minimum of 0.01. The total number of episodes for training is set to 1000. The approach involves using the neural network to approximate the Q-values, which represent the expected future rewards for each action taken in a given state. As the agent interacts with the environment, the neural network updates its weights based on the rewards received, thereby improving the policy that dictates the agent's actions over time. The aim is to enable the neural network to learn the optimal actions that maximize cumulative rewards.

C. Genetic Algorithm with Reinforcement Learning

The genetic algorithm combined with reinforcement learning aims to enhance the optimization of Q-learning parameters by leveraging the evolutionary processes of genetic algorithms. The key parameters optimized by the genetic algorithm include the learning rate, discount rate, and epsilon decay. The genetic algorithm is configured with 1000 generations, a population size of 20, and a mutation rate of 0.1. The approach involves evolving a population of candidate solutions over successive generations. Each candidate's performance is evaluated based on its fitness, which reflects its ability to achieve high rewards in the Taxi-v3 environment. Through selection, crossover, and mutation operations, the genetic algorithm refines the parameter values, striving to find an optimal balance that significantly enhances the agent's performance. This evolutionary process aims to discover a set of hyperparameters that enable the agent to learn a more effective policy for navigating the taxi tasks.

III. Results and Analysis

A. Reinforcement Learning Results

The reinforcement learning model, utilizing traditional Q-learning, demonstrated robust performance improvements over 1000 episodes. Initially, at episode 100, the scores varied widely, ranging from approximately -1200 to -200, indicating a phase of instability as the model was still exploring the environment. However, by episode 500, the model exhibited a significant upward trend, with scores rapidly improving and moving closer to 0. By episode 1000, the scores had stabilized around 0, signaling that the model had effectively converged to an optimal policy. This rapid improvement and convergence underscore the efficiency of traditional Q-learning techniques in discrete action spaces like the Taxi-v3 environment. The model's capacity to efficiently explore and exploit state-action pairs enables it to quickly learn and adopt optimal behaviors.

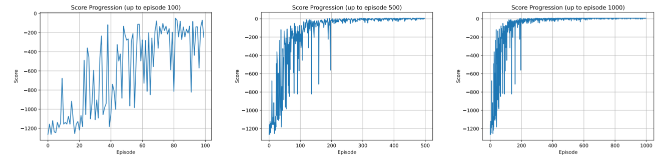

Fig. 2. RL with 100, 500 and 1000 generations

B. Deep Learning Results

The performance of the deep learning model was evaluated over 1000 episodes. At episode 100, the scores varied significantly, ranging from approximately -450 to -250, reflecting initial instability as the model began learning. By episode 500, the scores demonstrated an upward trend, with noticeable improvements and reduced variability, stabilizing around -100. By episode 1000, the model continued to improve and showed convergence, consistently achieving scores around -100. This gradual improvement and eventual convergence indicate that while deep learning can effectively learn policies for the Taxi-v3 environment, it does so at a slower pace compared to traditional reinforcement learning methods. The complexity of neural networks, while powerful for handling high-dimensional data, may not provide substantial advantages in this simpler, discrete environment, leading to a slower convergence rate. This suggests that traditional methods like Q-learning might be more efficient for such tasks, where the problem space is well-defined and less complex.

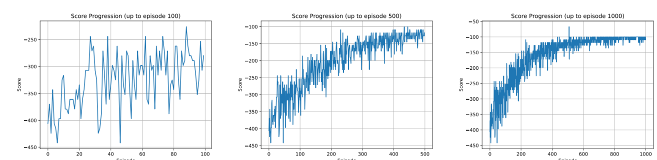

Fig. 3. Deep learning with 100, 500 and 1000 generations

C. Genetic Algorithm with Reinforcement Learning Results

The genetic algorithm combined with reinforcement learning exhibited a distinct performance trajectory over 1000 episodes. Initially, at episode 100, the model showed quick stabilization, with scores converging around -127, indicating that the genetic algorithm effectively identified a set of hyperparameters that provided a stable performance baseline. However, by episode 500, the scores remained around -127, suggesting that the optimization process had plateaued. At episode 1000, the scores still hovered around -127, demonstrating that the hybrid approach struggled to refine the policy further. This performance trajectory suggests that while the genetic algorithm can efficiently identify a reasonably effective set of hyperparameters and stabilize performance quickly, it may not be as effective in fine-tuning these parameters for optimal performance in discrete environments like Taxi-v3. The suboptimal long-term performance compared to traditional reinforcement learning highlights the need for further tuning and possibly more sophisticated integration of the genetic algorithm with reinforcement learning techniques to achieve better results. This observation underscores the importance of adaptive strategies that can evolve not just hyperparameters but also the policy itself, ensuring continuous improvement beyond the initial stabilization phase.

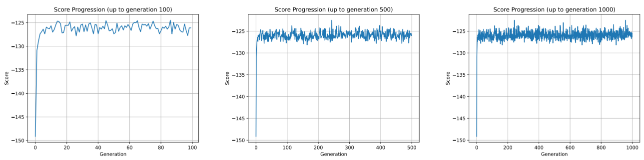

Fig. 4. GA + RL with 100, 500 and 1000 generations

D. Comparative Analysis Across Generations

At generation 100, the performance of the different models varied significantly. The reinforcement learning model showed wide score variations, ranging from -1200 to -200, indicating initial instability as the agent was still exploring the environment. The deep learning model exhibited scores between -450 and -250, reflecting a more stable but still inconsistent learning process. The genetic algorithm combined with reinforcement learning quickly stabilized around -127, demonstrating early convergence and suggesting that the genetic algorithm effectively identified a set of hyperparameters that provided a stable performance baseline.

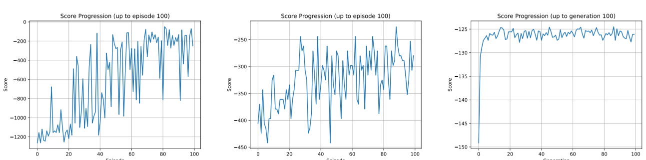

Fig. 5. Three strategies with 100 generations

By generation 500, reinforcement learning showed significant improvement, with scores nearing 0, indicating that the model was learning more effective policies and getting closer to optimal performance. The deep learning scores stabilized around -100, showing steady but slower progress compared to reinforcement learning. The genetic algorithm's performance remained around -127, indicating that it had reached a plateau and was not improving further, suggesting limitations in the hybrid approach's ability to fine-tune beyond initial stabilization.

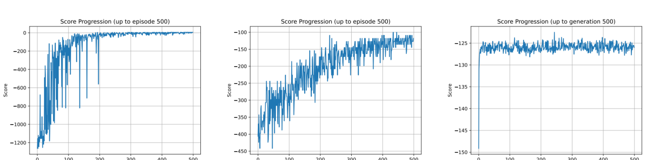

Fig. 6. Three strategies with 500 generations

At generation 1000, reinforcement learning achieved optimal performance with scores consistently around 0, demonstrating its effectiveness in quickly converging to an optimal policy. In contrast, the deep learning model consistently scored around -100, showing that while it could learn effective policies, it did so at a slower pace and did not reach the same level of performance as reinforcement learning. The genetic algorithm's scores still hovered around -127, indicating that it struggled to refine the policy further after the initial stabilization phase.

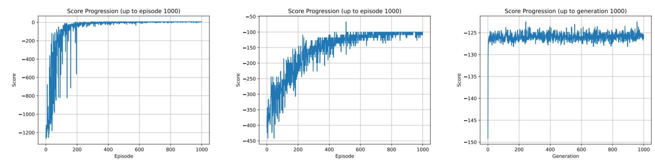

Fig. 7. Three strategies with 1000 generation

These trends illustrate that reinforcement learning provides the most effective and rapid learning in the Taxi-v3 environment, achieving optimal performance efficiently. In comparison, deep learning and the genetic algorithm with reinforcement learning show slower and less effective improvements, highlighting the need for further refinement and more sophisticated integration to enhance their performance to match the efficiency of traditional reinforcement learning techniques.

IV. Conclusion and Visualization

A. Conclusion

The comparative analysis reveals that traditional reinforcement learning (RL) outperforms both deep learning and the hybrid genetic algorithm with reinforcement learning (GA + RL) in the Taxi-v3 environment. The RL model's scores converged around 0, indicating optimal performance and showcasing its ability to effectively learn and apply optimal policies. In contrast, the deep learning model exhibited slower improvement, with its scores stabilizing around -100, reflecting a less efficient learning process in this particular setting.

The hybrid GA + RL approach showed a different performance trajectory. While it quickly stabilized to -127 by generation 100, it failed to show significant improvement beyond that point, highlighting its limitations in fine-tuning policies within this discrete environment. This plateau suggests that while the genetic algorithm component is effective at rapidly identifying a stable set of parameters, it does not provide the same level of incremental improvement as traditional RL techniques.

These results suggest that traditional Q-learning techniques are particularly well-suited for simple, discrete environments like Taxi-v3. Their ability to efficiently explore and exploit state-action pairs allows them to quickly converge to optimal policies, outperforming more complex models like deep learning and hybrid GA + RL in these contexts. This efficiency makes traditional reinforcement learning a robust choice for tasks where the problem space is well-defined and less complex.

B. Visualization

To illustrate the effectiveness of the best-performing model, we created a video demonstrating the simulation of the reinforcement learning model at its optimal performance. This visualization showcases the agent's ability to navigate the environment efficiently, pick up passengers, and drop them off at designated locations, highlighting the practical success of the learned policy.

Fig. 8. Reinforcement learning visualization

V. Future Work

Further research could focus on several areas to enhance the performance and effectiveness of these AI models. One approach could involve fine-tuning the parameters of the genetic algorithm (GA) to achieve better optimization results. Adjusting factors such as population size, mutation rates, and crossover strategies might lead to improved performance by enabling more effective exploration and exploitation of the solution space.

Additionally, exploring hybrid models that integrate the strengths of deep learning and reinforcement learning could be beneficial. By combining the powerful pattern recognition capabilities of deep learning with the strategic decision-making processes of reinforcement learning, these hybrid models might offer superior performance in complex environments. This could involve developing novel architectures or training methodologies that leverage the advantages of both approaches.

Another important avenue for research is extending the number of training episodes for all models to investigate long-term performance improvements. By running the models for a more extended period, it would be possible to observe whether they continue to learn and refine their strategies or if they plateau. This could provide valuable insights into the scalability and robustness of each approach over time.

Overall, these research directions aim to build on the current findings and push the boundaries of what these AI strategies can achieve in dynamic and discrete environments like Taxi-v3.