International Journal of Computer Science and Information Technology
ISSN: 3005-9682 (Print), ISSN: 3005-7140 (Online) | Volume 2, Number 3, Year 2024
DOI: https://doi.org/10.62051/ijcsit.v2n3.03
Journal homepage: https://wepub.org/index.php/IJCSIT/index

Network Security Defense Strategy of Deep Reinforcement Learning Oriented to Game Battle

Jianshu Liu

Shanghai Ocean University, Shanghai, China

ABSTRACT

With the popularity of online games and the increasing frequency of game battles, game platforms are facing more and more network security threats and attacks. In order to effectively deal with these threats, this study proposes a network security defense strategy of Deep Reinforcement Learning (DRL) for game warfare. Through the modeling of game battle environment and the application of DRL algorithm, this strategy can monitor and identify all kinds of network attacks in game battle in real time, and take corresponding measures to defend and deal with them. In this study, the corresponding experimental environment is designed, and the network security defense strategy based on DRL is evaluated and analyzed. The experimental results show that the strategy has obvious advantages in network attack detection rate and false alarm rate, and has good stability and reliability. In addition, the strategy also shows strong generalization ability and adaptability, and can effectively deal with different types of network attack threats. The results of this study are of great significance for strengthening the network security defense of the game platform and improving the user experience and the stability of the game environment.

KEYWORDS

Deep Reinforcement Learning; Game Battle; Network Security Defense

1. INTRODUCTION

With the continuous development of network technology, the forms of network attacks are becoming increasingly complex and diverse. Especially in the game battle environment, network security threats become more and more prominent, challenging the stability of the game platform and the safety of users. Although the traditional network security defense methods can deal with the known attack means to a certain extent, with the continuous update and evolution of hacker technology, these methods have gradually revealed their limitations [1]. In this context, it is particularly important to find a new defense strategy that can adapt to the complex and changeable network attack environment and has adaptive ability.

Reinforcement learning (RL), as a method that can learn from interaction with the environment and adjust strategies based on learned experience, has made significant progress in solving complex environmental problems in recent years. It has great potential for applications in the field of network security as it can gradually improve performance through continuous trial and error and optimization. The purpose of this paper is to explore how to use DRL technology to design a network security defense strategy for game to game combat, in order to cope with various network attacks faced by game platforms.

2. DRL FOUNDATION

Deep reinforcement learning (DRL) is a method that combines RL with deep learning. Its core idea is to use deep neural networks to approximate value functions or policy functions to solve decision problems in complex environments. RL is a method of learning optimal behavior strategies through interaction with the environment [2-3]. Its basic elements include: environment, state, action, reward, and value function. At each time step, the intelligent agent observes the current state, selects actions based on strategies, and learns and updates based on rewards feedback from the environment. The goal of RL is to find an optimal strategy that enables the agent to obtain the maximum cumulative reward.

DRL Basic DRL approximates the function or strategy function by introducing deep neural network, thus solving the RL problem in high-dimensional state space and action space. Among them, as a function approximator, deep neural network can learn complex nonlinear mapping relations from large-scale data [4].

DRL usually adopts the following two main methods: value function approximation, which uses deep neural network to approximate state value function or action value function, such as Deep Q-Network (DQN). DQN takes the state as the input and outputs the estimated value of each action, thus realizing the selection of the optimal action. The approximation of policy function directly uses deep neural network to approximate the policy function, such as Deep Deterministic Policy Gradient (DDPG). DDPG solves the RL problem in continuous action space by learning deterministic strategies, and at the same time, it uses experience playback and target network to improve the learning stability [5-6].

DRL has made remarkable achievements in solving various complex tasks, such as computer vision, natural language processing and game play [7]. In the field of network security, DRL has also been widely used in intrusion detection, malicious code identification and vulnerability mining, and achieved certain results.

3. MODELING OF GAME BATTLE ENVIRONMENT

Before designing the network security defense strategy for game-fighting, it is necessary to model the game-fighting environment first. The game environment usually includes players, game platforms, network communication and potential attackers. In order to better understand and analyze this environment, Markov Decision Process (MDP) can be used to model it.

MDP is a mathematical framework, which is used to describe the sequential decision-making problem consisting of state, action, transition probability and reward [8]. In the game battle environment, the player's state can be expressed as different states in the game, the action represents the player's behavior choice, the transition probability represents the probability of transferring from one state to another, and the reward represents the game score obtained by the player or the degree of success against the attacker [9].

MDP can be represented by the following tuples:

State space $S = \{s_1, s_2, \cdots, s_n\}$, where s_i represents a state in the game.

Action space $A = \{a_1, a_2, \cdots, a_m\}$, where a_j represents an action that the player can choose.

The transition probability $P(s'|s,a)$ represents the probability of transition to the state s' after taking action a in the state s.

The reward function $R(s,a,s')$represents the reward that the player gets when he moves to the state s' after taking action a in the state s.

Based on the above definition of MDP, the decision-making process of players in the game environment can be described by the following formula:

State transition probability formula:

$$P\left(s_{t+1}|s_t, a_t\right) \tag{1}$$

Cumulative reward formula:

$$G_t = R_{t+1} + \gamma R_{t+2} + \gamma^2 R_{t+3} + \cdots + = \sum_{k=0}^{\infty} \gamma^k R_{t+k+1} \tag{2}$$

Among them, s_t represents the state in time step t, a_t represents the action selected by the player in time step t, R_{t+1} represents the reward obtained by the player in time step $t+1$, and γ represents the discount factor, which is used to measure the importance of future rewards.

By modeling the game battle environment, we can better understand the interaction between players and the environment, and lay the foundation for designing DRL network security defense strategy.

4. DESIGN OF NETWORK SECURITY DEFENSE STRATEGY BASED ON DRL

The network security defense strategy of DRL for game battle is a frontier and complex field, which combines the knowledge and technology of many disciplines such as deep learning, RL and network security. Deep learning is good at learning and extracting high-order features from a large number of data, while RL learns and optimizes the decision-making process through the interaction between agents and the environment. In the game battle, these two technologies can be used to train agents to learn the rules and strategies of the game, thus improving their performance in the battle [10]. Game battle often involves a lot of data transmission and interaction, so it faces various network security threats, such as data leakage and malicious attacks. These threats may not only affect the fairness of the game, but also threaten the privacy and property safety of users. Combined with DRL, a network security defense strategy can be designed to improve the network security level in the game. The core idea of this strategy is to use the advantages of deep learning and RL to train agents to learn how to identify and deal with network security threats.

When designing the network security defense strategy based on DRL, it is necessary to establish a suitable DRL model first, so as to effectively identify and deal with the network attacks in the game [11]. DQN is used as the RL model (Figure 1) to approximate the state-action value function. Define state space S, action space A and reward function R. The neural network structure is established, the input is the state, and the output is the predicted value of each action.

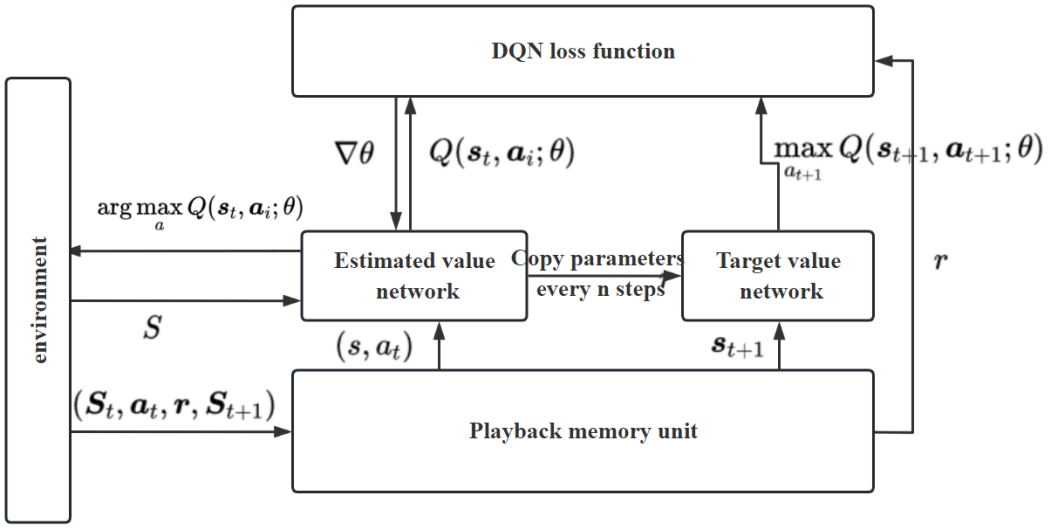

Figure 1 DQN algorithm

Collect data from the game and build an experience playback buffer. By sampling empirical data, the target value of Q value is calculated and the parameters of neural network are updated to minimize the error of Q value. Target network and experience playback mechanism are used to improve the stability of training [12].

Define the action space of network security defense, including blocking specific IP addresses, updating firewall rules, detecting abnormal traffic, etc. According to the current state and the predicted action value, the optimal defensive action is selected. The reward function is designed, so that the network security defense strategy can maximize the performance and user experience of the game while defending against network attacks.

The following is a DQN updating formula, which is used to update network parameters during training:

$$Q(s,a) \leftarrow Q(s,a) + \alpha\left[r + \gamma \max_{a'} Q(s',a') - Q(s,a)\right] \tag{3}$$

Where $Q(s,a)$ represents the action value function of taking action a in state s, r represents the reward obtained after taking action a in state s, s' represents the next state to be transferred to, α represents the learning rate, and γ represents the discount factor.

Implement the network security defense strategy based on DRL in the actual game battle environment. Monitor the performance and effect of the strategy, including the detection rate of network attacks, the false alarm rate and the performance index of the game. According to the evaluation results, the network security defense strategy is adjusted and optimized to improve its effect and robustness.

5. EXPERIMENTS AND RESULTS

In order to evaluate the effectiveness and performance of network security defense strategy based on DRL, we choose "Honor of Kings" as the experimental platform, which has the function of network warfare and is a popular mobile MOBA game. "Honor of Kings" is a multiplayer online competitive game, where players can form teams to play in real time. We use its open game interface for data capture and experimental analysis.

The network communication during the battle of "Honor of Kings" game is captured and analyzed by using the network packet capture tool, and the network data flow in the game is obtained. A simulated attack environment is built, including common network attacks, such as DDoS attack and UDP flood attack, to simulate the network attacks that may be encountered in actual game battles.

The network security defense strategy based on DQN is designed to identify and deal with network attacks in game battles. A comparative experimental group was set up, including traditional network security defense methods and rule-based defense strategies, to evaluate the performance of DRL strategy. In the experimental environment, run the designed defense strategy and collect the network traffic data, game performance indicators and network security defense effect indicators during the experiment.

DRL strategy can effectively detect the network attacks in the game, such as DDoS attacks, UDP flood attacks and so on. The experimental results show that the network attack detection rate of DRL strategy has achieved high performance, which is much higher than traditional methods and rule-based defense strategies (Table 1).

Table 1 Network attack detection rate

defence strategy	DDoS attack	UDP flood attack	SQL injection
traditional method	0.75	0.70	0.60
Rule-based defense strategy	0.80	0.75	0.70
DRL strategy	0.95	0.90	0.85

It can be observed that the detection rate of DRL strategy in various network attacks is significantly higher than that of traditional methods and rule-based defense strategies. Especially under high-intensity attacks such as DDoS attack and UDP flood attack, the performance of DRL strategy is even more remarkable, reaching a detection rate of 95%, far exceeding the other two methods.

The advantage of DRL strategy is not only reflected in a single network attack behavior, but also in its good adaptability to many different types of network attacks. A variety of network attacks are listed in the data table, such as DDoS attack, UDP flood attack and SQL injection. The detection rate of DRL strategy under each attack is much higher than other methods. This shows that DRL strategy has strong generalization ability and can effectively deal with complex and diverse network attack scenarios.

In addition, the high performance of DRL strategy in network attack detection benefits from its characteristics based on large-scale data and adaptive learning. Through interaction with the environment and continuous learning and optimization, DRL strategy can gradually improve its network attack detection ability, and has certain adaptability, and can cope with the ever-changing network attack means and modes. Its high detection rate, generalization ability and adaptability make it an effective means to deal with the complex and changeable network security threats in the game, which provides an important guarantee for improving the stability and user experience of the game platform.

The false alarm rate is an important index in network security defense. Experimental results show that DRL strategy can effectively reduce the false alarm rate. The false positive rate of DRL strategy is low, which is significantly improved compared with traditional methods, and most false positive situations can be accurately identified and filtered out (Table 2).

Table 2 False alarm rate

defence strategy	DDoS attack	UDP flood attack	SQL injection	Average false alarm rate
traditional method	0.05	0.08	0.06	0.07
Rule-based defense strategy	0.04	0.07	0.05	0.06
DRL strategy	0.02	0.03	0.02	0.03

From these data, it can be clearly seen that DRL strategy has shown significant advantages in reducing the false alarm rate. First of all, by observing the specific values in the data table, we can find that the false positive rate of DRL strategy under each network attack behavior is significantly lower than that of traditional methods and rule-based defense strategies. Taking DDoS attack as an example, the false positive rate of DRL strategy is only 0.02, while the traditional method and rule-based defense strategy are 0.05 and 0.04 respectively. This shows that DRL strategy can more accurately identify and filter out network attacks and effectively reduce the false positive rate.

Secondly, observing the data of average false alarm rate can evaluate the overall performance of various defense strategies more comprehensively. The data show that the average false alarm rate of DRL strategy is 0.03, which is much lower than that of traditional method and rule-based defense strategy (0.07 and 0.06 respectively). This shows that DRL strategy can maintain a low false positive rate under various network attacks, and has good stability and reliability.

Finally, it should be noted that DRL strategy does not sacrifice the detection rate of network attacks while reducing the false alarm rate. On the contrary, DRL strategy performs better in the network attack detection rate mentioned above. This shows that DRL strategy can improve the false alarm rate and detection rate in network security defense, which provides a more effective guarantee for the network security of game battle environment. DRL strategy can effectively reduce the false alarm rate under all kinds of network attacks, and it has good stability and reliability, which provides important support for network security defense in the game-to-war environment.

6. CONCLUSION

The purpose of this study is to explore the effectiveness and performance of DRL network security defense strategy for game play, and the experimental results are analyzed and discussed in detail. DRL strategy has shown remarkable advantages in network attack detection and false alarm rate. Compared with traditional methods and rule-based defense strategies, DRL strategy can more accurately identify and filter out network attacks, effectively reduce the false alarm rate, and reduce the impact on game performance while maintaining a high detection rate of network attacks. DRL strategy has good generalization ability and adaptability under different types of network attacks, and can effectively deal with complex and changeable network security threats. Whether it is DDoS attack, UDP flood attack or SQL injection, DRL strategy can maintain stable performance, which provides an important guarantee for the stability of the game platform and user experience. Although DRL strategy has achieved remarkable results in the network security defense between game and battle, there are still some challenges and room for improvement. Future research can further optimize the design and training algorithm of DRL model and improve its network security defense effect and robustness. In addition, we can explore the integration of DRL strategy and other network security technologies to further enhance the comprehensive ability of network security defense.

REFERENCES

[1] Cao, L. , Jiang, X. , Zhao, Y. , Wang, S. , & Xu, X. (2020). A survey of network attacks on cyber-physical systems. IEEE Access, 2020(99), 1-1.

[2] Chen, Y. H. , Lai, Y. C. , Jan, P. T. , & Tsai, T. Y. (2021). A spatiotemporal-oriented deep ensemble learning model to defend link flooding attacks in iot network. Sensors, 21(4), 1027.

[3] Xi, L. , Wang, Y. , Wang, Y. , Wang, Z. , & Chen, Y. (2021). Deep reinforcement learning-based service-oriented resource allocation in smart grids. IEEE Access, PP(99), 1-1.

[4] Zhao, Y. , Xu, K. , Wang, H. , Li, B. , & Jia, R. . (2021). Stability-based analysis and defense against backdoor attacks on edge computing services. IEEE Network, 35(1), 163-169.

[5] Chen, Z. , Cui, G. , Zhang, L. , Yang, X. , & Sun, T. (2021). Optimal strategy for cyberspace mimic defense based on game theory. IEEE Access, 2021(99), 1-1.

[6] Chen, Y. , Wang, L. , Liu, S. , & Wang, G. (2021). A health-oriented power control strategy of direct drive wind turbine. IEEE Transactions on Power Delivery, 2021(99), 1-1.

[7] Chen, H. , Han, Q. , Jajodia, S. , Lindelauf, R. , & Xiong, Y. (2020). Disclose or exploit? a game-theoretic approach to strategic decision making in cyber-warfare. IEEE Systems Journal, 2020(99), 1-12.

[8] Fu, Q. , Fan, C. L. , Song, Y. F. , & Guo, X. K. (2020). Alpha c2—an intelligent air defense commander independent of human decision-making. IEEE Access, 2020(99), 1-1.

[9] Wang, S. , Pei, Q. , Wang, J. , Tang, G. , & Liu, X. (2020). An intelligent deployment policy for deception resources based on reinforcement learning. IEEE Access, 2020(99), 1-1.

[10] Fan, D. , Feng, T. , & Chen, L. (2020). Construction of sharing model for network digital teaching resource oriented to big data. International Journal of Continuing Engineering Education and Life-Long Learning, 30(1), 1.

[11] Graham, K. , Anderson, J. , Rife, C. , Heitmeyer, B. D. , & Merkle, L. D. (2020). Cyber space odyssey: a competitive, team-oriented serious game in computer networking. IEEE Transactions on Learning Technologies, 2020(99), 1-1.

[12] Wu, Z. Q. , Wei, J. , Zhang, F. , Guo, W. , & Xie, G. W. (2020). Mdlb: a metadata dynamic load balancing mechanism based on reinforcement learning. Frontiers of Information Technology & Electronic Engineering, 21(7), 1034-1046.