

# Improvement of Deep Reinforcement Learning Using Curriculum in Game Environment

Mohammadreza Mohammadnejad<sup>1</sup>, Morteza Dorri-Giv<sup>2</sup>, Farzin Yaghmaee<sup>3</sup>

## Original Article

### Abstract

**Introduction:** Training deep curriculum learning is a kind of smart agent training in which, first the simple acts, and then, the difficult acts are trained to smart agent. In this study, we proposed a new framework for training deep curriculum learning to defense-based game in particular Dragon Cave.

**Materials and Methods:** Deep reinforcement learning approach with curriculum learning was used to train an intelligent agent in the game Dragon Cave. Curriculum learning paradigm started from simple tasks, and then gradually tried harder ones. Using Proximal Policy Optimization, the intelligent agents were trained in various environments, once in a curriculum-learning environment, and once in an environment without curriculum learning. Then, they started the game in the same environment.

**Results:** The improvement of the agent was observed with deep curriculum reinforcement learning.

**Conclusion:** It seems that the deep curriculum reinforcement learning increases the rate and the quality of intelligent agent training in complex environment of strategic games.

**Keywords:** Intelligent agent, Deep reinforcement with curriculum learning, Machine learning, Neural network

**Citation:** Mohammadnejad M, Dorri-Giv M, Yaghmaee F. **Improvement of Deep Reinforcement Learning Using Curriculum in Game Environment.** J Res Rehabil Sci 2019; 15(1): 50-7.

Received: 09.02.2019

Accepted: 11.03.2019

Published: 04.04.2019

### Introduction

High-quality artificial intelligence has had a significant impact on the sales of computer games in recent years, but in games with a complex environment, the game's artificial intelligence is far behind human intelligence (1). The manual implementation of artificial intelligence behaviors is time consuming and difficult, and the use of machine learning techniques is a promising alternative to produce intelligent behavior. Deep reinforcement learning (DRL), which focuses on learning based on trial and error, is a type of artificial intelligence training that has been used successfully in the gaming space (2).

When the game's actions are very complex and the rewards for the actions are sparse, reinforcement learning takes a long time to converge, and direct agent training in a complex game environment has a poor performance (3). In order to improve agent training in the present study, DRL with curriculum learning was used, which trains the agent with a

sequence of environments that gradually become more difficult. Curriculum learning is a type of learning in which the agent begins the training with simple samples of work and then gradually increases the difficulty of the task (4). This type of training is applied in various fields such as image classification and has a better performance compared to other machine learning approaches (5).

Some actions in the game environment are so large that they can be divided into several sub-tasks with varying degrees of difficulty, and then after decomposing the action, they can be taught with the curriculum learning approach. Thus, the agent encounters a simple problem at each stage, learns it, and then enters the next stage of training (3). In fact, this is the way humans learn to do something (for example, walking). First they roll over, then they experience crawling and standing, and finally, they learn to walk, and with this type of learning algorithm training, they can use the basic concepts they have

1- PhD Student, Department of Artificial Intelligence, School of Electrical and Computer Engineering, University of Semnan, Semnana, Iran

2- Assistant Professor, Department of Software Engineering, School of Electrical and Computer Engineering, University of Semnan, Semnana, Iran

3- Assistant Professor, Department of Software Engineering, School of Electrical and Computer Engineering, University of Semnan, Semnana, Iran

**Corresponding Author:** Morteza Dorri-Giv, Email: dorrigiv@semnan.ac.ir

learned from the previous step to learn the higher level operations.

DRL is the use of the neural network as an estimating function of reinforcement learning (6). The idea of combining neural network and reinforcement learning has a long history and was developed by Tesauro in the early 1990 using the neural network in the game as an estimating function and was displayed at the level of the best human player (7). Neural network has long been used in identification and control systems (8). It should be noted that two decades after the results of the investigation by Tesauro (7), reinforcement learning with the nonlinear estimating function is still somewhat ambiguous. The sudden progress in the use of DRL was followed by the success of the study by Mnih et al., which showed that computers could learn Atari games by entering images (9). After that, numerous interesting studies were initiated in this field.

In curriculum learning, the training samples should be divided into several sets based on difficulty, which requires knowledge in the field of research (4). The “Baby Step” learning method was suggested in a study by Bengio et al. in order to improve the curriculum learning. However, curriculum learning and Baby Step learning require the manual processing of the training samples before training (4).

Another study used the Convolutional Neural Network (CNN) along with a curriculum learning strategy in the classification of mammography images (10). In particular, first the classifier was taught on the images of the lesions on mammograms, and then a scan-based model was presented using the features learned. Additionally, the curriculum learning and algorithm used in the study by AlphaGo was used (11) to run the Gomoku game on a human level (12). The use of Actor-Critic reinforcement learning with the curriculum learning approach in the game environment had promising results in the first-person shooter game; So that the smart agent was able to use new tactics in the three-dimensional environment of the game (13). The aim of the present study is to use DRL with a curriculum learning approach in the environment of the strategic games.

### Materials and Methods

The deep learning technique has shown very promising results in object detection (14) and speech recognition. One of the challenges in the reinforcement learning is the lack of statistical access to the function being optimized (total agent reward predicted). The agent goal depends on the dynamic model, which is

sometimes unknown in the problem space. Moreover, the input data to the algorithm depends on the behavior of the agent in the environment, and therefore, it is not possible to provide an algorithm with a uniform improvement. In more complex problems, sometimes instead of one function, there are a number of different approximate functions. The agent DRL with the curriculum learning can help the agent in teaching complex actions.

A curriculum can be a weighted sequence of the courses learned. Initially, the weight of the training samples is such that it makes the lessons easier. In other words, samples are taught that have the simplest concepts and can be easily learned. At the end of the sequence, the weight of the samples is uniform and the training is conducted with all the samples. It is assumed that  $z$  is a random variable of the set of the training samples [pair  $(x, y)$  for supervised learning] and  $P(z)$  is the distribution of the training samples that the agent must eventually learn a function of. Moreover, if  $0 \leq W_\lambda(z) \leq 1$  is a weight that is applied for sample  $z$  in step  $\lambda$  of the curriculum, with  $0 \leq \lambda \leq 1$  and  $W_\lambda(z) = 1$ , relation 1 is established.

$$\forall z \cdot Q_\lambda(z) \propto W_\lambda(z)P(z) \quad \text{Relation 1}$$

And if  $\int Q_\lambda(z)dz = 1$ , relation 2 is established.

$$\forall z \cdot Q_1(z) = P(z) \quad \text{Relation 2}$$

In a sequence with a uniform increase in which the value of  $\lambda$  increases from zero to 1, entropy should be increased to increase the diversity of the training samples and the weight of the samples should be increased by adding to the training set. In each problem space, the ease and difficulty definition of the training sample is different (4).

**Proximal Policy Optimization (PPO):** The PPO algorithm was used in DRL (15). This method is the modified form of the trust region method, and the goal of both methods is to maximize the substitute function with a limit on the policy updating rate. PPO uses a clipped objective to initiatively restrict the Kullback-Leibler divergence (KL-divergence).

In relation 3,  $\rho_t = \frac{\pi_{\theta}(a_t|s_t)}{\pi_{\theta_{old}}(a_t|s_t)}$  and  $\epsilon$  are hyperparameters. When  $A_t$  is positive, the function is clipped with  $1 + \epsilon$ , and when  $A_t$  is negative, the function is clipped with  $1 - \epsilon$ . Besides,  $L^{PPO}$  does not include changes that improve the objective and changes that make the goal worse (15).

$$\max_{\theta} L^{PPO}(\theta), L^{PPO}(\theta) = E [\min(\rho_t A_t, \text{clip}(\rho_t, 1-\epsilon, 1+\epsilon)A_t)] \quad \text{Relation 3}$$

**Test platform**

**Dragon Cave:** Dragon Cave is a two-dimensional strategic game developed using the Unity game engine and by one of the authors of this study and can be downloaded from the “Bazaar” app store (16). In this game, 4 different dragons with different abilities confront their 9 enemies to defend their cave. Each dragon has its own spell, of which it can only place six spells in its deck. The game consists of 5 routes through which the enemies move towards the cave and the dragons, by moving in these routes, throw their spells towards them and destroy them. The player will win the game if he can withstand the enemies for 80 seconds. Figure 1 demonstrates a view of the Dragon Cave game environment.

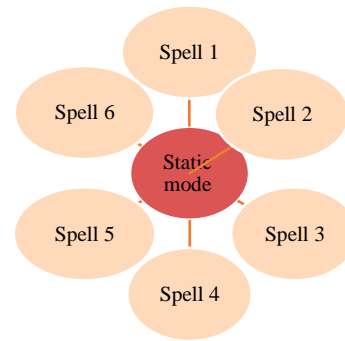


**Figure 1.** View of the Dragon Cave game environment

**Dragon’s abilities:** The dragon in the game can perform certain actions by having forces (for example, reducing the spells of its enemies or increasing its own spells, etc.); these forces are named spells in the game. Each dragon can use a spell at any time. In other words, the outline of the dragon system

is as figure 2.

Each spell has different characteristics; the common feature of the spells is having the preparation time; So that after using each spell, it will take some time to prepare and reuse. The complete list of dragon spells is indicated in table 1.



**Figure 2.** General view of the spells of dragons

The different spells of each dragon have a different effect on the game environment (including enemies and dragons). For example, the use of spell can damage enemies, slow them down, increase the hit point and power of dragons, or create fortified towers in the game. Each player in the game can move their character in different directions. As figure 3 shows a view of the five paths, each game map contains 5 horizontal paths that the player can go to each of them by touching it without taking time. Enemies move randomly from each direction to the player, and in order to use most of its spells, the dragon must move in the direction in which the enemy is present. Being on the path in which the enemy is present is one of the basic rules of the game.

**Table 1.** Dragon Cave game spells










| Spell name       | Icon | Amount of damage   | Amount of elixir used | Preparation time |
|------------------|------|--|-----------------------|------------------|
| Fireball         |      | Can harm the enemy as much as 120% of the dragon’s power is.   | 20                    | 1.5              |
| Lava             |      | Can harm the enemy as much as 180% of the dragon’s power is.   | 40                    | 5                |
| Fire blast       |      | Can harm the enemy as much as 150% of the dragon’s power is.   | 50                    | 30               |
| Hearth           |      | Saves 1000 deadly impacts of the two forces of fireball and lava, and after hitting the enemy, damages it in the same level. | 80                    | 0                |
| Dragon storm     |      | After hitting the enemy, kills it and damages the enemies behind itself by as much as 10% of its spell.                      | 80                    | 30               |
| Fire tower       |      | The dragon appears 5 fire towers that protect the dragon cave for 15 seconds.  | 120                   | 120              |
| Presence of mind |      | It reduces the preparation time of all dragon forces to zero.  | 60                    | 90               |



**Figure 3.** View of the five routes

**Enemies:** The enemies encounter the dragons in 9 different modes and accidentally move towards them in 5 different directions, and they differ from each other in the rate of damage per second, the speed of movement, and the rate of spells. Enemies are the main observation of the player during the game. Table 2 lists the enemies and their features.

**Table 2.** Enemies in Dragon Cave game

| Name of enemies in the game | Icon  | Spell rate | Amount of damage per second | Speed of movement per second |
|-----------------------------|---|------------|-----------------------------|------------------------------|
| Insidious                   |  | 600        | 41                          | 20                           |
| Small shielded              |  | 1500       | 71                          | 24                           |
| Hunter                      |  | 1050       | 101                         | 8                            |
| Lancer                      |  | 1200       | 101                         | 13                           |
| Catapult                    |  | 900        | 251                         | 3                            |
| Large shielded              |  | 2400       | 131                         | 15                           |
| Mercenary                   |  | 1350       | 91                          | 28                           |
| Chubby                      |  | 3000       | 121                         | 13                           |
| Magician                    |  | 900        | 121                         | 12                           |

**Online Competition:** In the online part of the game, to which the player enters from stage 30, two dragons compete with each other, and whichever defends his cave better will win this part. The two dragon characters, one controlled by the intelligent agent and the other by the human agent, confront each other. Actions that can be performed by players include vertical movement in lines and a practical action called force. In the meantime, the machine must decide how to move on the paths and to which mode of its enemies throw each spell so that it has both real and human-like behavior and its movements seem reasonable.

## Results

**Agent training algorithm:** The agent observes and decides in the game space. The task of the agent is to

use the spells of the dragon against the enemy. The agent training is performed for one type of dragon, and it is considered to simplify the training of the dragon with 4 forces. The path to success in DRL is not clear, and the algorithm of variables has many changing components that make it difficult to remove errors, and a lot of effort is required to set these components to achieve good results.

PPO is the balance between ease of execution, sample complexity, and ease of adjustment and effort to calculate an update at each stage that minimizes the cost function; while the confidence of deviation from the previous policy is relatively small. Therefore, this algorithm has been used to train the agent. Furthermore, the curriculum learning was used to train the agent; in such a way that the agent first learns simple operations and then more difficult operations are trained to it.

**Observations:** The agent observation vector contains 79 features. The observations included: in which path the enemy is present? For all four dragon spells: Is the force ready for use? the amount of the normalized force damage in the range [0, 1] (the maximum amount that the spell can apply is 2000 and the damage of all spells in the range [0, 1] is normalized based on the maximum damage), for 6 enemies close to the dragon: the amount of spell of the enemy normalized in the range [0, 1], the type of enemy (there are 9 different enemies in the game), whether the enemy is hitting or not? The agent is responsible for using the dragon spells; the actions include the use of force 1, the use of force 2, the use of force 3, the use of force 4, and the lack of use of force.

**Reward function:** The reward function is designed in such a way that the agent is driven to hit the enemies. In addition, it will be rewarded by the overall goal of the system, which is to win the game. Algorithm 1 shows the reward function.

```

Algorithm 1: Reward function
if Player Cast Spell
    if !any Enemy Exis in Same Lane
        Punish()
if Player Select Not Ready Spell
    Punish()
if Die
    Punish()
if Win
    Punish()
if Spell hit Enemy
    Punish()
if Enemy Attacking
    Punish()

```



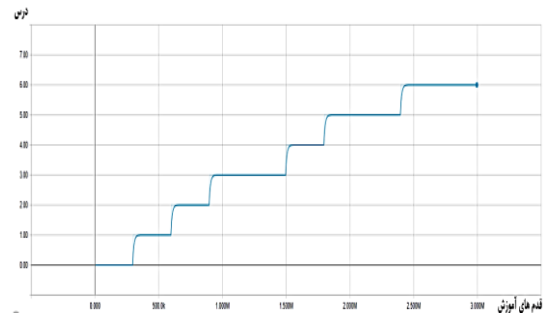
If the player wins or the spell reduces the enemy's life, he will be rewarded. If the player is not on the enemy's path when using the spell, he will be punished. Moreover, if the player loses or the enemy is damaging, the agent will be punished.

**Curriculum design:** The agent is first trained in a simple environment. The simple environment in the Dragon Cave game means more limited encounters with enemies. The agent must learn how to confront 9 different enemies. In other words, it must have a proper mapping between each enemy and its spells, and gaining the knowledge of what spell to use against each enemy is a key factor in a player's superiority. For example, the dragon's storm spell can destroy the first enemy on the path. If this spell is used against the enemy with the most spells, it is the best use of that spell, and if it is used against the weakest enemy, it may cause the player to lose.

Learning this requires the agent to first learn how to confront a weaker enemy. This low level of knowledge leads the agent towards the local optimum and facilitates the training path towards learning the higher level operations. The training was implemented in 3 million steps and the curriculum was designed as follows. At first, the enemies were organized in order of power, and 7 curricula were designed. The design of these steps was such that the agent gradually confronted the enemies with different forces and faced the new enemies with the weights learned from the previous curriculum.

Figure 4 illustrates the curriculum increase stages in the training steps.

First 10% of training steps: Facing of the agent with two types of enemies (weaker enemies)



**Figure 4.** Curriculum increase stages in the training steps

Second 10% of training: Facing of the agent with three types of enemies

Third 10% of training: Facing of the agent with four types of enemies

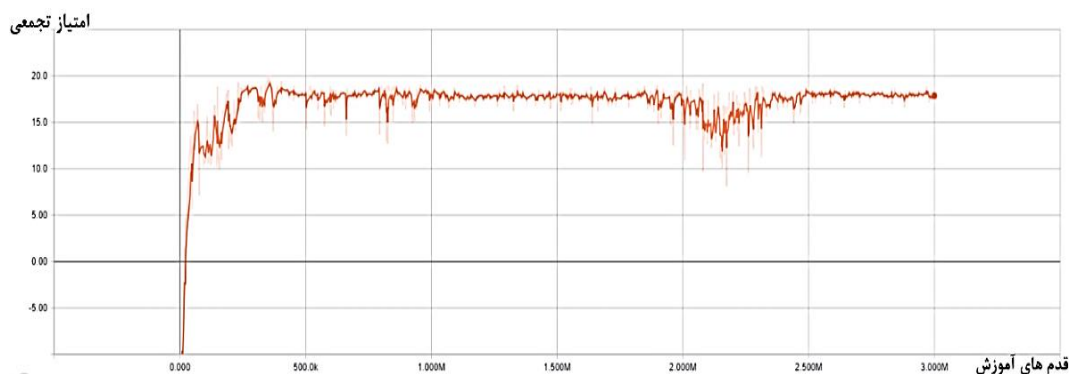
Fourth 20% of training: Facing of the agent with five types of enemies

Fifth 10% of training: Facing of the agent with six types of enemies

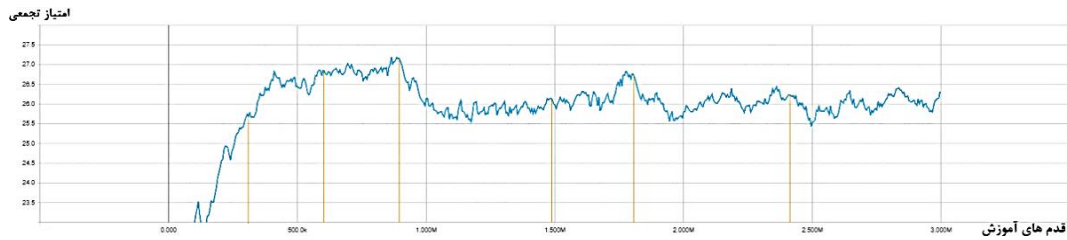
Sixth 20% of training: Facing of the agent with seven types of enemies

Final 20% of training: Facing of the agent with nine types of enemies

**Intelligent agent training:** Figures 5 and 6 show the cumulative reward of the agent during training. Two agents are trained with the same characteristics and algorithms (one in the complex game environment and the other with the curriculum). As can be seen, the agent without a curriculum was able to score 18 points at the end of the training, which increased to 26 points for the agent with the curriculum. The findings suggested that this method increased the speed and quality of learning of the agent. In fact, the agent with the curriculum was able to score more points in the game space and move faster towards the local optimum.



**Figure 5.** Collective rewards during training without a curriculum



**Figure 6.** Cumulative reward of the learning agent with the curriculum stages (orange lines show the time of facing of the agent with the new curriculum)

### Discussion

The present study was accomplished with the aim to investigate the success of DRL with a curriculum learning approach to train the intelligent agent in the Dragon Cave game environment. This method seems to increase the speed and quality of the agent learning. In fact, the agent with curriculum was able to score more points in the game space and move faster towards the local optimum. In the training steps, the agent score drops when a new curriculum is added to the training space; because the agent has no knowledge of the new problem. Using the DRL method with curriculum learning, the agent learns the new action very quickly using the knowledge gained in the previous step. However, without a curriculum, it is very difficult for the agent to learn the actions trained.

The topic of the present study is somewhat new and innovative, and has received less attention compared to other subjects. For example, Sukhbaatar et al. proposed a framework for automatically learning of curriculum through asymmetric self-play. Two agents named Alice and Bob had the same tasks with different purposes. Alice challenged Bob to perform an action, and Bob tried to do it as soon as possible. Thus, the interaction between Alice and Bob automatically created a curriculum that included more difficult and challenging tasks (17).

Justesen et al. employed DRL in game content designing and showed that DRL in some special games can be over-fitted, but with the help of the curriculum, one can generalize the training policy to design the game content on a human level (18).

In the study by Wu and Tian, two intelligent agents were trained in different environments in the First-Person Shooter game using PPO; One in an environment with a curriculum and the other in an environment without a curriculum. Then they both started playing in the same environment and the results exhibited a better quality of the agent with the curriculum in terms of scores. In these cases, the experiment was mainly focused on games with fixed environments or in environments other than the game

(3). However, the present study was conducted by the research team on strategic games, which had a more complex environment in comparison to the game with a fixed environment.

### Limitations

The weak computer system was one of the limitations of the present study, so the agent training time was limited to 3 million steps. Since the present study was the first to be performed on the Dragon Cave game, it was not possible to compare it with other studies.

### Recommendations

It is suggested that the present study be conducted on a more powerful computer system, more steps, and smart agent training, in addition to passing through each curriculum step in the training process by obtaining the desired score, in which the desired score is the criterion for learning that action in that course. Therefore, the curriculum changes as the agent reaches the score. In the present study, the curriculum increase is along with the number of training steps which may not be a good criterion in some subjects. Additionally, the idea presented in this study can be used in the space of other strategic games.

### Conclusion

The present study examined the effect of using DRL with the curriculum learning approach to train the intelligent agent in strategic games. For this purpose, the Dragon Cave strategy game space was used. The algorithm used trained curriculum using simple activities and gradually taught the more difficult activities. Using PPO, an intelligent agent was trained in an environment with a curriculum and another in an environment without a curriculum. Then they both started playing in the same environment. Given the findings, the first hypothesis was confirmed as DRL with the curriculum in the complex space of strategic games helps the intelligent agent to have better and faster generalization in learning.

### Acknowledgments

Aseman Omid Entertainment Company is appreciated for providing the game.

The present study was one of the articles submitted to the Secretariat of the Fifth International Conference on "Computer Games; Opportunities and Challenges" with a special focus on therapeutic games (February 2017, Isfahan, Iran), which was praised by the editorial board of the Journal of Research in Rehabilitation Sciences. The authors would like to appreciate the Cyberspace Research Institute of National Cyberspace Center for supporting the publication of this study. The Innovation Center for Entertainment Industries of the University of Isfahan, which played an important role in collecting data and achieving this project is also appreciated.

### Authors' Contribution

Mohammadreza Mohammadnejad: Study design and ideation, supportive, executive, and scientific services of the study, analysis and interpretation of results, manuscript preparation, specialized manual evaluation in scientific terms, confirmation of the final manuscript for submission to the journal office, responsibility to maintain the study integrity from beginning to publishing, and responding to the referees' comments; Morteza Dorri-Giv: Analysis and interpretation of results, manuscript preparation, specialized manual

evaluation in scientific terms, confirmation of the final manuscript for submission to the journal office, responsibility to maintain the study integrity from beginning to publishing, and responding to the referees' comments; Farzin Yaghmaee: Analysis and interpretation of results, manuscript preparation, specialized manual evaluation in scientific terms, confirmation of the final manuscript for submission to the journal office, responsibility to maintain the study integrity from beginning to publishing, and responding to the referees' comments.

### Funding

The present study was conducted with the financial support of Semnan University. The university did not comment on the data collection, analysis and reporting, manuscript preparation, and final approval of the paper for publication. This study was published in the Journal of Research in Rehabilitation Sciences, with the financial support of the Cyberspace Research Institute of the National Cyberspace Center, the sponsor of the Fifth International Conference on Computer Games with an Approach to Therapeutic Games. This research institute did not contribute to designing, compiling, and reporting this study.

### Conflict of Interest

The authors declare no conflict of interest.

### References

1. Arulraj JP. Adaptive agent generation using machine learning for dynamic difficulty adjustment. Proceedings of the 2010 International Conference on Computer and Communication Technology (ICCT). 2010 Sep 17-19; Allahabad, Uttar Pradesh, India. p. 746-51.
2. Mohammadnejad M, Yaghmaee F. Design of Intelligent agent with deep reinforcement learning in game environment. Proceedings of the 4<sup>th</sup> National and 2<sup>nd</sup> International Conference on Computer Games, Challenge and Opportunities; 2019 Feb 21; Kashan, Iran. p. 1-16. [In Persian].
3. Wu Y, Tian Y. Training Agent for First-Person Shooter Game with Actor-Critic Curriculum Learning. Proceedings of the International Conference on Learning Representations, ICLR 2017; 2017 Apr 24-26; Toulon, France. p. 1-10.
4. Bengio Y, Louradour J, Collobert R, Weston J. Curriculum learning. Proceedings of the 26<sup>th</sup> Annual International Conference on Machine Learning (ICML 2009); 2009 Jun 14-18; Montreal, Canada. p. 41-8.
5. Gong C, Tao D, Maybank SJ, Liu W, Kang G, Yang J. Multi-modal curriculum learning for semi-supervised image classification. IEEE T Image Process 2016; 25(7): 3249-60.
6. Francois-Lavet V, Henderson P, Islam R, Bellemare MG, Pineau J. An introduction to deep reinforcement learning. Foundations and Trends in Machine Learning 2018; 11(3-4): 219-354.
7. Tesauro G. Temporal difference learning and TD-Gammon. Communications of the ACM 1995; 38(3): 58-68.
8. Narendra KS, Parthasarathy K. Identification and control of dynamical systems using neural networks. IEEE Transactions on Neural Networks 1990; 1(1): 4-27.
9. Mnih V, Kavukcuoglu K, Silver D, Graves A, Antonoglou I, Wierstra D, Riedmiller M. Playing Atari with deep reinforcement learning. arXiv preprint arXiv:1312.5602. 2013.
10. Lotter W, Sorensen G, Cox D. A Multi-scale CNN and Curriculum Learning Strategy for Mammogram Classification. Cham, Switzerland: Springer International Publishing; 2017 p. 169-77.
11. Silver D, Huang A, Maddison CJ, Guez A, Sifre L, van den Driessche G, et al. Mastering the game of Go with deep neural networks and tree search. Nature 2016; 529(7587): 484-9.
12. Xie Z, Fu X, Yu J. AlphaGomoku: An AlphaGo-based Gomoku Artificial Intelligence using Curriculum Learning. arXiv, abs/1809.10595. 2018

13. Krizhevsky A, Sutskever I, Hinton GE. ImageNet classification with deep convolutional neural networks. *Commun ACM* 2012; 60 (6): 1097–1105.
14. Dahl GE, Yu D, Deng L, Acero A. Context-dependent pre-trained deep neural networks for large-vocabulary speech recognition. *IEEE Transactions on Audio, Speech, and Language Processing* 2012; 20(1): 30-42.
15. Tuan YL, Zhang J, Li Y, Lee HY. Proximal policy optimization and its dynamic version for sequence generation. *arXiv: 1808.07982*. 2018.
16. Mohammadnejad M. Dragon Cave, a strategy game [Online]. [cited 2020 Feb 20]; Available from: URL: [https://cafebazaar.ir/app/ir.sinsin.DragonCave.v\\_0/?l=en](https://cafebazaar.ir/app/ir.sinsin.DragonCave.v_0/?l=en), developed by M. Mohammadnejad
17. Sukhbaatar S, Lin Z, Kostrikov I, Synnaeve G, Szlam A, Fergus R. Intrinsic motivation and automatic curricula via asymmetric self-play. 2018. *Proceedings of the 6<sup>th</sup> International Conference on Learning Representations, ICLR 2018*; 2018 Apr 30-May 3; Vancouver, Canada.
18. Justesen N, Torrado RR, Bontrager P, Khalifa A, Togelius J, Risi S. Illuminating Generalization in Deep Reinforcement Learning through Procedural Level Generation. *arXiv: 1806.10729 [cs.LG]*. 2018.