Uprising E-sports Industry: machine learning/AI improve in-game performance using deep reinforcement learning

Xianzuo Du[1, a, †], Xiwei Fuqian[2, b, †], Jiaxi Hu[3, c, †], Zechen Wang[4, d, †], Dongju Yang[5, e, †]
[1]School of Jilin University, Jilin 130000, China
[2]University of Toronto- St. George Campus, Toronto M5S, Canada
[3]Shanghai Jiao Tong University, Shanghai, China
[4]Harbin Institute of Technology, Weihai 264200, China
[5]School of Jilin University, Jilin 130000, China
[a]duxz5518@mails.jlu.edu.cn,[b]peter.fuqian@mail.utoronto.ca, [c]Westbrook0@sjtu.edu.cn, [d]181110414@stu.hit.edu.cn,
[e]yangdj5518@mails.jlu.edu.cn
[†]These authors contributed equally.

Abstract. **With the quick development of machine learning, deep reinforcement learning will have a big influence on E-sports. It can be considered machine learning will help us train E-sports players easily and effectively and give coaches and players some new ideas to train and win the games. Flappy Bird is a game where the players try to keep the bird alive as long as possible and get a high score. Flappy Bird is a good example to prove that the thought is feasible. In this project, a flappy bird training AI is developed based on Q-learning and DQN. The game is played by the models obtained from deep reinforcement learning, and the game is also played by humans. Then get experimental data from these two ways and compare them. For the two ways of playing the game (by AI or manually), there are many similarities in the increased rate of scores as training sessions increase, which means AI can "teach" players how to train to get a higher score. It can be applied to skills and experience-based games and help us to train top players. Maybe it can also be applied to other fields, such as helping engineers escape from potential errors and accidents.**

Keywords: Deep reinforcement learning, game players, e-sports

I.Introduction

With the development of video games, they attract an increasing number of people and are changing people's lifestyles. Electric sports began to replace traditional sports in people's hearts. According to the official data, fewer than 8million people were watching the NBA finals live in 2020. However, approximately 46 million people watched the LOL finals live in the same year. Therefore, E-sports are catching a growing number of people. Due to the popularity of video games, video game markets are expanding quickly, and the video game industry's revenue is increasing rapidly. The scale of the E-sports market exceeded 100 billion in 2019 in China after rapid growth. And video games are becoming more specialized. For example, E-sports players are very good at playing one or more games and have professional qualifications. And there are more and more E-sports events worldwide. Machine learning has become mature in the past decade, and it has some application in the field of games. Takano mentioned that Deep Q Network (DQN) had been successfully applied to many types of games [1]. Silver, in his paper, used Go as an example to show that a computer can beat the human master of a game [2]. Therefore, machine learning has more potential in the field of E-sports.

In the traditional e-sports industry, the game players always train in the following ways:

A. Train by playing against other teammates. It is the most traditional way of training, which costs a lot of time and human resources. But it is also one of the most inefficient ways because the teammates are so familiar that they cannot train the strain capacity.

B. Train by playing against strange players who are usually from other teams or society. But there are some problems. The first one is that it is impossible to play with other teams frequently. Because it will cost lots of money to let them help with training. Playing with the players from society maybe is cheaper. But there are not so many non-professional players who can play so well to train the professional players.

C. The coach conducts intensive special training. This way is usually useful. But the human has a stereotype and so the coach cannot give a better processing mode in many cases.

Above all, the training ways of game players are just play with a human because the computer players in most games cannot play so great to train the human players.

The current solution to this problem is limited, including repetition training, asking for professional guidance, and learning from previous game videos. And they are widely used in today's e-sports. However, according to our survey of papers in the last seven years (since reinforcement learning flourished), no one has suggested guiding players to train using machine learning (reinforcement learning). There are only examples of using machine learning to play games. For example, Cunha invented an Artificial Intelligence (AI) system, which can be considered an experienced player that plays alongside the user to improve users' experience [3]. Another example is that Pang et al. show an example of Reinforcement learning in games by investigating a set of techniques of reinforcement learning for the full-length game of StarCraft II [4]. Nevertheless, no one has ever proposed that computers can help players training. In this paper, our group firstly proposes to guide players to train using machine learning from some papers that have inspired us. Horn's research shows that bots encapsulating idealized player strategies can help us create a richer model of level difficulty that reveals useful information about player struggles and learning across level progressions

978-1-6654-1736-5/21/$31.00 ©2021 IEEE
DOI 10.1109/MLISE54096.2021.00112

[5]. Kim first presented “Performance Evaluation Gaps” in the game between human and computer (AI) players [6].

The goal of the research is to give an analysis to suggest using machine reinforcement learning to solve the currently existing problems in the e-sports industry and help players improve their gaming performances in a way that the current solutions cannot provide. Taking League of Legends as an example, a recent article states that top-ranking players in LOL have better gaming skills related to cognitive abilities than average players [7]. Cognitive ability is the mental process of how people understand the world and act in it, which in our case is the players’ thoughts on approaching the game. Machine learning can be used to learn different players’ cognitive abilities, such as decision-making playmaking opportunities in the game. It will be useful when the coaches and analysts want to give feedback after competitive matches or scrim games. The machine will simulate similar situations where the machine can exploit other alternative options that human brains have not thought of. Thus, the coaches can guide players from the feedback by having solo targeted training and provide strategical instructions from the choice of the machine as it usually represents the best action to take under the situation. Another article shows that practicing team-focused activities will significantly positively affect competitive e-sports performance [8]. During individual LOL regional series, professional teams usually make appointments for scrim games every week at a set time. When it comes to the LOL World Championship, where teams from different regions will be split up into groups to compete, it will be harder to do scrim with teams within the group or from different regions as it might leak strategies. Thus, machine learning can come into play to simulate different teams to scrim with and practice teamwork. It can reduce human costs and time costs because the scrims can be done at any time of the day, and the team management can spend less time communicating with other teams to focus more on its players. Overall, the use of machine learning can train professional players and improve their levels of gaming to provide better performances for viewers in appreciation.

Our main contributions are summarized as follows.

1.Introduce the idea of using deep reinforcement learning to guide game players to train, thus improving their in-game performance.

2.Improve the "Flappy Bird" game to explore more possibilities by using deep reinforcement learning to guide game players through training. Specifically, four versions of the "Flappy Bird" game are designed, each with different levels of difficulty.

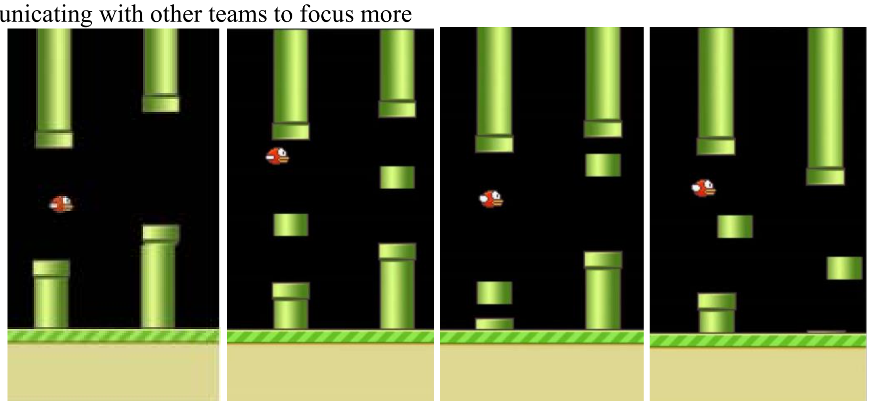
3.Conduct experiments to play games in two situations (by AI and manually) and compare the obtained data. The result shows a great number of similarities in the rate of increase of scores as training sessions increased, which means AI can "teach" players how to train to get a higher score.

II. Methods

A.Improvements

Flappy Bird is a very simple game in which the player tries to keep the bird alive as long as possible. The bird automatically flies forward and falls. If the bird hits the ground or pipes, it dies. Therefore, the player must time jump properly to keep the bird alive. The game score is measured by how many pipes the bird successfully passes through. So, the player must keep the bird alive as long as possible to get a high score.

The game is very simple, so it is a good idea to add some improvements to the game and make it more challenging. In flappy bird 1.0, there are more gaps. Compared with the original game, which only has one gap and two pipes in the vertical direction, flappy bird 1.0 has two gaps and three pipes. Both of the gaps allow the bird to pass through.



flappy bird   flappy bird 1.0   flappy bird 2.0   flappy bird 3.0

Figure 1. Different versions of our game

In flappy bird 2.0, neither of the gaps allow the bird to pass through, and the width of gaps needs to be changed. In some cases, only one of the gaps allows the bird to pass. In some other cases, both gaps allow the bird to pass just like flappy bird 1.0. the player must plan the movement route ahead of time because sometimes there will be only one way to pass through pipes.

In flappy bird 3.0, based on flappy bird 2.0, the pipes distribute randomly in the horizontal direction. The game will look very complex. The player must plan the route, or it would be very easy to hit the pipes.

B. Algorithm

The reinforcement learning used in the game is an iterative process that solves a given policy evaluation function and updates the policy according to the value function. The value function is derived from the Bellman equation and represents the expected return such that it will choose the option with the highest value as the final decision. Thus, an input state (S) will present the output Q (S, A) by the Ɛ-greedy strategy. In this case, the state S is the flappy bird game interface, action A is the set of actions of doing nothing or jump, and the quality function Q will calculate the maximum reward after acting. The reward will be either positive or negative considering the actions it takes.

In Q-learning, it uses Ɛ-greedy strategy to find action A, but also use the greedy strategy to find the action that maximizes the next Q (S', A') function and calculate the target formula. It would make the Q function from consecutive iterations become very dependent on each other and lead to an adverse impact as it converges. Thus, the DQN is introduced to approximate the action-value function and allow generalizations for unknown states. This algorithm is model-free in that the model of its environment is unknown and requires sampling. The input state observation will go through about 3 convolution layers: the first layer is of size 8 x 8 with stride 4 and consists of 32 filters, the second layer is of size 4 x 4 with stride 2 and consists of 64 filters, the third layer is of size 3 x 3 with stride 1 and consists of 64 filters, and between each layer would be a 2 x 2 max pooling. After the convolution layers, there will be a fully connected dense layer of 256 outputs, followed by another matrix multiplier dense layer with 2 outputs for the action in the final readout. In the training process of DQN, the Convolutional Neural Network will first randomly initialize and input a state S. Then, the Ɛ-greedy strategy is used to find the optimal action, alongside the use of experience replays to store experiences in memory during the process. The bird will not train at first but just store data for replay memory. As the game progresses after each frame, the probability Ɛ will slowly decrease to balance out exploration and exploitation. In the end, the game wants to stabilize the probability Ɛ and converge the Q function so that the game will reach convergence or complete a certain number of iterations.

C. Evaluation method

Q-Learning and DQN use the Ɛ-greedy method to predict Q values by randomly choosing between exploration and exploitation that has nothing to do with the action-value estimates. For exploration, it will explore randomly to improve the current knowledge of each action. For exploitation, it will be greedy and choose the action that gives the most reward despite some potential negative feedbacks. In this game, the initial probability of Ɛ is set to be small for some explorations at the start, because the bird will not know the correct thing to do at the start. As training goes on by a lot of random actions, it will get a sense of the correct actions for different situations. It means that as the game runs in every frame, this probability will reach closer to an equilibrium with the probability of 1- Ɛ for exploitation. Hence, the game goes from random exploration to set exploitation.

DQN will also use the technique of experience replay to solve the strong correlation in the training of Q-Learning. That makes it inefficient and negatively impacts the game because, during the running game, the experiences from consecutive frames are closely related to each other and affect the training. At the start of the game, a replay memory of set size is created to store the most recent experiences of every frame during the run. Thus, the game will not immediately train with the emptiness of the replay memory but store some data, to begin with. As the game progresses and takes action to compute the Q function, the replay memory will also update to ensure that it will always contain the latest experience created. Then it will uniformly sample a mini batch of experiences as input to discover the maximum Q values and update the DQN.

III. Experiments

A.Model training:

Models are trained for each of the four levels of difficulty of "Flappy Bird," respectively. It has been found that for games with some difficulty levels, birds can survive for a long time with a model that requires only a little time of training. While for games with other difficulty levels, birds can only survive with a model that requires a large time of training or even cannot survive. It is believed that this will help game players choose the right level of difficulty in games where humans can choose the level of difficulty to not waste time on the impossible levels of difficulty. For example, in a game with selectable difficulty levels, the players always want to complete as high a level of difficulty as possible. Still, some very high difficulty levels may have been made impossible to complete at all. This is where deep reinforcement learning comes in. Try different levels of difficulty and give feedback to the player. With the information provided by AI, game players can then choose the highest difficulty level of the game they are likely to complete and continue to train in anticipation of completing the game.

B.Method of comparison

Experiments are designed to show that deep reinforcement learning can guide the training of game players to improve their in-game performance. It is a comparison experiment. The "Flappy Bird" game is played by AI (the model being trained) and by humans (manually). Flappy bird 1.0 version is chosen because the original version is too easy, and flappy bird 2.0 and flappy bird 3.0 are too difficult. For playing the game by AI, 8 models are obtained, and each is used for playing games. The first model is trained for 50000 timesteps (one timestep is one action of the bird), the second model is trained for 100000 timesteps, and so on until the eighth model is trained for 400000 timesteps. For playing games by humans manually, 8 groups of people are invited. Before playing games, each group of people did some practice. The first group practiced playing the game 50 times (one time in one game), the second 100 times, and so on until the eighth group practiced playing the game 400 times. With the model and the practice experience, our AI and human players play the game, respectively. Based on some important values in our codes, such as "epsilon", "reward", and "goal", an average score for the AI-controlled birds is calculated. Based on the average time the birds survived and the average number of pillars they passed, average score of human-controlled birds is calculated. We try to compare the increase of computer scores as model training time increases with the increase of human scores as human practice time increases. If there is some relationship between them, it can be concluded that there are some similarities between the models trained by deep reinforcement learning and practice made by human players. It indicates that players can learn how many more times they need to practice getting a particular higher score from the AI's performance in the game. Thus, game players can make training plans better.

Some important values in our codes:

• Epsilon: The value of the parameter ϵ.

• Goal: The sum of all values between two adjacent -1's in" reward".

• Reward: The evaluation value returned each time.

IV. Results

A.Result of Models

In the original version of the flappy bird, the bird can survive for a very long time with a model trained for 200000 timesteps. In flappy bird 1.0, the bird can survive for a very long time with a model trained for 300000 timesteps. However, in flappy bird 2.0 and flappy bird 3.0, it is too difficult for the bird to survive for a long time, even with a trained model for a huge number of timesteps. It can be concluded that this is because the pillars are too irregular in these versions of the game, so the birds can not learn enough through deep reinforcement learning to help them survive. Thus, if I am a "Flappy Bird" game player, I would stick with the flappy bird 1.0 difficulty for training. Because I know there is a huge possibility that I can never play flappy bird 2.0 or flappy bird 3.0 well because they are too difficult. This is a simple but perfect example of how deep reinforcement learning can help game players make training plans.

B.Result of Comparisons

As shown in Figure 2, the practice time (for the human player) is from 50 to 400, and the timestep to train the models (for the AI player) is 1000 times the above numbers. The result shows human scores from 39 to 81 and AI scores from 250 to 350. And as shown in Figure 3, our group make a Pearson correlation analysis using SPSS and find that the Pearson correlation between human and AI scores was very strong. Although the horizontal and vertical axes are both different, the two curves almost coincide, which indicates that the overall trends of the two curves are the same. It can be found that the increased rate of computer scores as model training time increases and the increased rate of human's scores as human's practice time increases are very similar. This is also a good example of how deep reinforcement learning can help game players make training plans. Because if AI can increase 20% of its score by 50% more timesteps of model training, human game players will know that they can also increase 20% of their scores by 50% more times of practice.

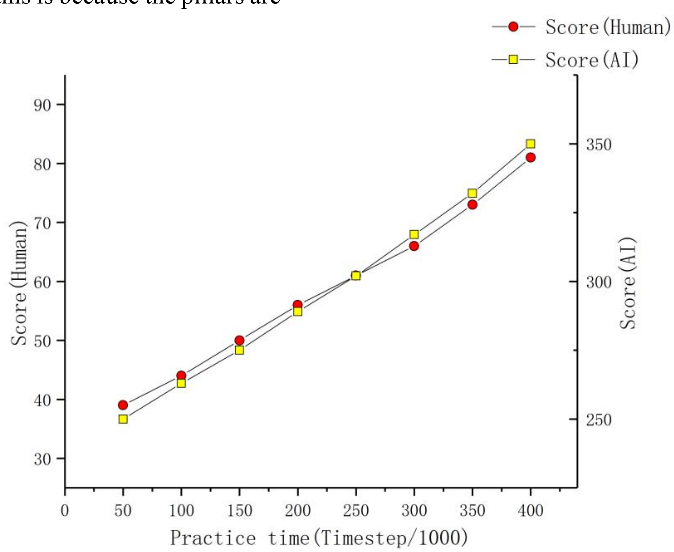


Figure 2. Line chart of practice time(timestep/1000) and score

Correlations

		Score_Human	Score_AI
Score_Human	Pearson Correlation	1	.999**
	Sig. (2-tailed)		.000
	N	8	8
Score_AI	Pearson Correlation	.999**	1
	Sig. (2-tailed)	.000	
	N	8	8

**. Correlation is significant at the 0.01 level (2-tailed).

Figure 3. Correlations between Human's score and AI's score

V. Discussion

A. Discoveries:

The discoveries of this paper are as follows:

a) From the result, it is easy to find that the general scores also increased with the increase of training timesteps. But at certain stages, scores also fluctuate.

b) It shows that the scores increase linearly with training timesteps. But it increases slowly, which can only become visible after every 5,000 training timesteps.

c) In the different difficulty of games, the speeds of the score increasing are different. The more difficult the game is, the slower the score increase.

B. Comparison:

Compared with the previous study, which just used the original design, it improved the difficulty of the game, which helps in finding more precise details. It can be found about how the computer changes the ways of playing the game when it becomes more difficult. It also shows what difficulties are impossible to complete and how fast game players can improve their scores. This will help human players to learn more from AI systems and become more professional.

C. Inferences:

The inferences of this paper are as follows:

a) The difficulty will not change as time goes by, such that in one level of a game, the scores will increase linearly with training times.

b) It can be found that the number of rewarding, being -0.1, decreases over time, which means the bird dies less and less. It can be estimated that after so many training timesteps, the bird will not die anymore.

D. Future works:

In reality, there is no such thing as a flappy bird competition. Thus, this project cannot be directly applied to the current e-sports industry. Its purpose is to test the feasibility of deep learning in the field of coaching game players and then used it to refine the project for different games to produce real value.

VI. Conclusions

In the traditional e-sports industry, players are usually trained by other people, which has many problems. To improve the training effect of e-sports players, it is proposed to use reinforcement learning to help players train. This is tried in a small game called the flappy bird to verify its feasibility. This research develops a flappy bird training AI based on reinforcement learning. It will become stronger as the number of training increases. When enough many times are being trained, it got a very hard score for humans to get. That means it can help players to learn how to get more scores. It can be used to determine what difficulties are impossible to complete and how fast game players can improve their scores. Thus, game players can make training plans better. In this paper, there is only a simple example of the "Flappy Bird" game being used as preliminary proof of this idea. And the findings in the flappy bird game suggest that it can be applied to the gaming industry by simulating different situations to help train professional players that were originally done by coaches, such as in MOBA or FPS. It will show the significance of using machine learning to train players by demonstrating a positive and impactful increase in gaming performance over a set amount of time. The algorithm can also be used in other skills and experience-based fields, which require teaching and training from coaches. Taking examples of modern industry, how to choose the correct building structure in civil engineering, or the choice of weld spot in welding engineering all require teaching from experienced professionals. It could potentially be replaced using machine learning to prevent errors or accidents from happening. In future research, more data should be collected or more algorithms should be developed in the gaming industry to increase the effect and efficiency that machine learning has on training professional players and increase their gaming performance. It can be shown by going through targeted training or team-focused activities to improve players' decision-making or choice of making plays during a game to find an alternative way to win. It should achieve a state where machine learning can fully replace the role of coaches to instruct players. Furthermore, it can reduce the human costs and time costs of coaches that are redundant and inefficient in the gaming industry when sometimes coaches cannot provide a correct way to win games.

References

[1] Takano Y, H. Inoue, Thawonmas R and Harada T 2019 Self-play for training general fighting game AI *Proc. - 2019 NICOGRAPH Int. NicoInt* **362** 6419 p 120

[2] Silver D et al 2016 Mastering the game of Go with deep neural networks and tree search *Nature* **529** 7587 pp 484-9

[3] Cunha L de Freitas R and Chaimowicz L 2010 An Artificial Intelligence system to help the player of Real-Time Strategy games *Proc. - 2010 Brazilian Symp. Games Digit. Entertain. SBGames 2010* pp 71–81

[4] Pang Z J, Liu R Z, Meng Z Y, Zhang Y, Yu Y and Lu T 2019 On reinforcement learning for full-length game of starcraft *33rd AAAI Conf. Artif. Intell. AAAI 2019, 31st Innov. Appl. Artif. Intell. Conf. IAAI 2019 9th AAAI Symp. Educ. Adv. Artif. Intell. EAAI 2019* pp 4691–98

[5] Horn B, Hoover A K, Folajimi Y, Barnes J, Harteveld C and Smith G 2017 AI-assisted analysis of player strategy across level progressions in a puzzle game *ACM Int. Conf. Proceeding Ser.* **F1301**

[6] Kim M J, Kim K J, Kim S and Dey A K 2018 Performance Evaluation Gaps in a Real-Time Strategy Game between Human and Artificial Intelligence Players *IEEE Access* **6** pp 13575–86

[7] Li X, Huang L, Li B, Wang H and Han C 2020 Time for a true display of skill: Top players in League of Legends have better executive control *Acta Psychologica.* **204** p 103007

[8] Gerber, H., Sweeney, K. and Pasquini, E. 2019 Using API data to understand learning in League of Legends: a mixed methods study *Educational Media International.* **56(2)** pp 93–115