ISSN: 2472-3703 DOI: 10.6911/WSRJ.202312_9(12).0015

Machine Learning in Video Games: Current Status and Future Prospects

Ruizhe Wan[1, a]

[1]International Business School Suzhou, Xi'an Jiaotong-Liverpool University, Suzhou, Jiangsu, 215123, China

[a]ruizhe.wan21@student.xjtlu.edu.cn

Abstract

One of the main challenges for video games in using artificial intelligence has always been how to balance controllability and intelligence while maintaining game playability. With the breakthroughs of new artificial intelligence technologies such as large language models in recent years, the topic of how to integrate these emerging technologies into the latest game genres has stood out. This paper reviews the application and challenges of AI and its subfield machine learning in the game field, as well as the potential future development trends.

Keywords

Artificial intelligence, Machine learning, Video games.

1. INTRODUCTION

Video games are an important application domain for artificial intelligence, as they provide an environment where the performance of AI algorithms can be objectively evaluated and compared with human players, thus demonstrating the capabilities and potentials of AI.[1] Machine learning is a branch of AI, whose core goal is to learn the relationship between inputs and outputs without explicitly specifying it. Some methods in machine learning have achieved remarkable results in video games, such as AlphaGo, AlphaZero, DQN, etc., which have surpassed human level in various types of video games.[2] This article will review the applications and challenges of machine learning in video games, and analyze its advantages and limitations in different game genres, as well as the future trends and possibilities.

2. ARTIFICIAL INTELLIGENCE AND MACHINE LEARNING IN CURRENT COMMERCIAL VIDEO GAMES

While artificial intelligence has advanced in the research field, machine learning has been barely applied to electronic games, which is because video games need to have playability first. It can be said that, in video games, the goal of introducing artificial intelligence is not to generate an invincible entity for players to compete against, but to let players participate and enjoy the game for a long time to the maximum extent. Therefore, unpredictable actions made by artificial intelligence are more valuable in the research field, but not necessarily interesting for players. Therefore, compared with uncontrollable artificial intelligence, game developers prefer controllable "narrow" artificial intelligence that can optimize the game experience.

2.1. Artificial Intelligence in Modern Video Game Development

Generally speaking, modern video game development can be divided into two parts, engine and assets. The game engine focuses on the basic logic of the game, and it calls the game assets.

ISSN: 2472-3703 DOI: 10.6911/WSRJ.202312_9(12).0015

According to the rules designed by the developers, to achieve the game functions and effects. Assets refer to the materials that make up the game graphics and sound, such as images, audio, animation, etc.

In the current commercial video game development process, artificial intelligence is mainly reflected in the non-assets part. This part of artificial intelligence can be briefly divided into two types, game AI optimization, behavior strategy level and game interaction feedback level. In game AI optimization, behavior strategy level, artificial intelligence is more inclined to the "narrow artificial intelligence" mentioned earlier, specifically in dealing with the behavior of computer controlled non-player characters (NPCs) in game. Here are some common methods in current developing.

2.1.1. Finite State Machine. Finite state machine (FSM) is a model that represents a number of finite states and the transitions between them. It abstracts complex behaviors into a few simple core states, which switch under certain triggering conditions, forming the feedback of NPCs. The programmers pre-set a series of rules or algorithms for the NPC's behavior, allowing the NPC to react accordingly to different situations. As Laura E Shummon Maass illustrates in her article with the example of Wolfenstein 3D, game developers considered all the possible scenarios that the enemy army could encounter.[3] And based on these scenarios, they compiled a list that allowed the NPC to respond accordingly in the game. This can make the NPC show a certain degree of intelligence and adaptability, but it also has drawbacks such as monotonous behavior and fixed logic. A more advanced method is based on this is to use Monte Carlo tree search.

2.1.2. Monte Carlo Tree Search. Monte Carlo tree search (MCTS) is a heuristic search algorithm. Its asymmetric search tree will randomly be generated with the development of the program running and response gradually following the operator's action. Specifically in the game domain, apart from the 1997 chess artificial intelligence Deep Blue, a more recent example is Civilization, a turn-based game where players develop their own civilization through various operations and compete with artificial intelligence players to achieve specific victory conditions. [4] In this case, it is impossible for the computer to pre-program all the action for the artificial intelligence. Therefore, the MCTS AI will evaluate some possible next actions, then calculate the total return of every action, and select the highest value one.

2.1.3. Artificial Neural Network. To make the NPCs in the video game more realistic, there is another method, namely artificial neural network (ANN), which can be used for both game development and play game program development. Given sufficient parameters, any continuous real-valued function can be approximated by an ANN with its distinctive network structure and the weights defined by each edge. This feature can help large-scale action role-playing games have more challenging and unpredictable hostile NPCs that will have real-time feedback. Chollakorn Nimpattanavong showed us how to use this technique to predict the behavior of computer NPCs in fighting games.[5,6] Similarly, this technique can also be used to improve the intelligence of NPC behavior. NPCs can learn the specific style of the player during the initial several rounds of the game, and adjust their actions in the subsequent games based on these data.

2.2. Artificial Intelligence in Playing Modern Video Game

Compared to game development, play game program development have more freedom to operate artificial intelligence. These games have objective criteria for victory or defeat, which can intuitively compare with humans and serve as a platform for AI research verification.[1] From the initial Deep Blue, to the research on playing Atari 2600 games in 2013, artificial intelligence has shown its strength. Based on deep reinforcement learning, AlphaGo, developed by Google DeepMind, successfully swept humans in the field of Go.[7] Based on the statement of the AlphaGo Zero team, the latest work of AlphaGo, AlphaGo Zero, skipped the supervised

ISSN: 2472-3703 DOI: 10.6911/WSRJ.202312_9(12).0015

learning step of artificial intelligence training, and started playing itself from using a neural network trained by self-play reinforcement learning. In StarCraft 2, a more complex real-time strategy game, the reinforcement learning program AlphaStar based on human player playback also achieved similar achievements.[8]

In this paper, games are divided into two categories: self-play and group play, and they are explained separately. Self-play was first used in the early education field, which describes the process of freely choosing, initiating, and playing games according to one's own needs and interests, and spontaneously communicating and interacting in them. Group play refers to activities that involve multiple people, have certain rules and goals, are competitive and changeable, and are for entertainment purposes.

2.2.1. Self-play. In the field of self-play, besides the famous examples in Go and chess domain, the research on playing Atari 2600 games by Volodymyr Mnih's team is also pioneering.[7] They used the raw video data as the basis, and successfully achieved the goal by jointly training a convolutional neural network and a Q-learning based variant algorithm. This enabled them to outperform all previous methods in six games, and in three games beat a game well-trained player.

2.2.2. Group play. In group play, two types of games that are more favored by researchers are MOBA games and real-time strategy games (RTS), which exhibit aspects such as multiple-agent, vast state-action range, and multiple action management. Generally speaking, these two types of games include multiple stages, and each stage has different strategies, and the choice of each stage affects the subsequent stages. Therefore, the design of artificial intelligence for playing these two types of games is usually a combination of a series of machine learning methods.

Deheng Ye et al. proposed a learning method for supporting an entire MOBA game battle through deep reinforcement learning.[9] For different stages of MOBA games, like different hero gameplay or hero actions, they designed an actor-critic network. For problems where the game is too complex, they processed it by first fixing the lineup to train the teacher model, then using multi-teacher policy distillation to transfer knowledge to the student model, and finally using random lineup for mixed training. This enables deep reinforcement learning to handle complete MOBA games.

In RTS games, besides AlphaStar developed by Google DeepMind, Peng Peng et al. also conducted research using a similar stage-wise approach as in MOBA games.[10] This includes modeling the micromanagement tasks in the game as a zero-sum stochastic game problem, setting unique state range, action domain and reward function for each agent. This aim to maximize the cumulative discounted reward, and assigning different local and individual reward functions for different agents and their interactions with the opponents.

When dealing with the task of coordinating multiple teams to fight and defeat the enemy, they introduced a multi-agent bidirectional coordination network BiCNet with an actor-critic formula that contains a vectorized extension. This enables it to handle battles of arbitrary numbers of agents of different types on both sides, and allows BiCNet to learn various advanced coordination strategies that experienced game players often use. In addition, Jakob N. Foerster et al. also have similar characteristics in their research on artificial intelligence for StarCraft games.[11]

From the above research, it can be seen that different researchers have some commonalities in their research on complex group games, which include stage-wise processing of tasks and using actor-critic networks.

ISSN: 2472-3703 DOI: 10.6911/WSRJ.202312_9(12).0015

3. FUTURE PROSPECTS

Generally speaking, the plot of a game, although it has some differences in the way of expression, but overall, the plot output needs to balance the rationality and uniqueness of the story, so that the players can be immersed and attracted by the game content. As a result, it can be regarded as a kind of interactive novel or visual novel.

In recent years, with the emergence of the latest research achievements such as ChatGPT and new Bing, the field of artificial intelligence text generation has also gained attention. People hope that artificial intelligence text generation can provide corresponding feedback according to the different responses made by the players, and ultimately make the game have a similar personalized experience effect. An earlier study was an agent competition held by Timothy Atkinson et al. between 2016 and 2018 IEEE conferences, which can play classic text adventure games.[12] The competition provided challenges for natural language understanding and generation for game AI and tried to improve and optimize them. In terms of fully artificial intelligence-generated creative game text, there is Nick Walton's AI Dungeon. It initially used the smallest GPT-2 model and later improved it with a new machine learning model. Its automatically generated text has more complete logic and more realistic emotions. Although its gameplay is still controversial, it has also made a breakthrough.[13]

Although there is no corresponding research, in the game Mount & Blade II: Bannerlord, the players developed a MOD Inworld Calradia based on the API provided by the ChatGPT developers, which can make the NPCs in the game respond accordingly according to the different text content entered by the players, combined with the NPC's character background.

In the field of reality, some researches that combine external tool interfaces to achieve more capabilities and domains have also emerged. According to Meta AI Research, they built a model called toolformer, which trained a language model to use external APIs, to achieve more abilities and functions without sacrificing its basic language skills.[14] The open-source project langchain on GitHub also hopes to combine large language models with external sources of computation and knowledge, to develop truly usable applications.[15] In addition, Sai Vemprala et al. from Microsoft also brought language models to reality by introducing a similar method.[16] They used ChatGPT to generate corresponding operation functions or classes by calling the given external APIs, which enabled them to control robots. In the research, ChatGPT could also create new functions and methods by itself according to the task instructions. This could also be a good example, such as using fully artificial intelligence-generated methods, only by giving and constructing a rough framework, the game content could be completely randomly or personalized fed back through large language models, which could give players more real-time and closely related feedback similar to real life. This may become a new direction and future of using artificial intelligence in game development when it combined with AR or VR.

For the aspects of level design and environmental scenes in game development, due to the richness of game content and the limitation of human resources, automatically and/or assisted creation of assets is becoming an increasingly important part of game development. Linus Gisslén et al. proposed a novel method, ARLPCG, an adversarial reinforcement learning approach based on procedural content generation.it can automatically generate and test previously unseen environments or models according to auxiliary input as control variables.[17] This can replace some or all of the manual operations in the development of some large games.

For the aspects of art in game development, including textures, maps and music, etc., there are also some breakthroughs at present. Artificial intelligence programs such as Amper Music, AIVA and MuseNet can create completely unique music or use machine learning algorithms to evaluate existing music and produce new works in a similar style. Artificial intelligence can also be used to generate textures, maps and paintings. For example, OpenAI's DALL-E model can generate images based on text descriptions. Some of these products have already entered the

ISSN: 2472-3703 DOI: 10.6911/WSRJ.202312_9(12).0015

market, such as Adobe Firefly by Adobe, which includes functions such as Text to image, Generative fill, Text effects and Generative recolor.

Since the launch of the open development platform Early Access by Steam in 2013, one of the largest online video game distribution website in the world, it has provided a trial-and-error opportunity for many independent game developers and small companies. At the same time, this open game development that allows real-time interaction between developers and players also provides more possibilities for game improvement.[18] These developers can use the artificial intelligence programs mentioned above to assist their design, such as making prototypes, reducing development costs, etc.

4. CONCLUSION

This paper reviews the application of machine learning in the field of video games as well as the future development trends and possibilities. The main challenge for commercial games in using artificial intelligence in game development is how to balance controllability and intelligence, as well as how to improve the playability and innovation of games. This makes the current artificial intelligence in commercial games still in the early stage, compared to the rapid development in the research field. In recent years, the emergence of new artificial intelligence technologies has made the future development direction of artificial intelligence in the game field clearer, which includes resource-oriented exploration, player-oriented personalization and so on. It can be said that the combination of machine learning and video games is the trend of the future game field, with great potential.

REFERENCES

[1] Chen,Y. (2022). Application of Artificial Intelligence Technology in Computer Game Software. Computer Engineering & Software, 43(10), 92–94.

[2] Arulkumaran, K., Deisenroth, M. P., Brundage, M., & Bharath, A. A. (2017). Deep Reinforcement Learning: A Brief Survey. IEEE Signal Processing Magazine, 34(6), 26–38.

[3] Shummon Maass, L. E., & Luc, A. (2019, July 2). Artificial intelligence in video games. Towards Data Science. https://towardsdatascience.com/artificial-intelligence-in-video-games-3e2566d59c22 (Accessed on September 8, 2023)

[4] Lou, H., & Maass, L. E. S. (2017, June 27). AI in video games: Toward a more intelligent game. Science in the News. https://sitn.hms.harvard.edu/flash/2017/ai-video-games-toward-intelligent-game/ (Accessed on September 8, 2023)

[5] Oh, I., Rho, S., Moon, S., Son, S., Lee, H., & Chung, J. (2022). Creating Pro-Level AI for a Real-Time Fighting Game Using Deep Reinforcement Learning. IEEE Transactions on Games, 14(2), 212–220.

[6] Nimpattanavong, C., Choensawat, W., & Sookhanaphibarn, K. (2022). Action Prediction of AI Bot in FightingICE by using Deep Learning Model. 2022 IEEE 4th Global Conference on Life Sciences and Technologies (LifeTech), 259–262.

[7] Mnih, V., Kavukcuoglu, K., Silver, D., Graves, A., Antonoglou, I., Wierstra, D., & Riedmiller, M. (2013). Playing Atari with Deep Reinforcement Learning (arXiv:1312.5602). arXiv. http://arxiv.org/abs/1312.5602

[8] Silver, D., Schrittwieser, J., Simonyan, K., Antonoglou, I., Huang, A., Guez, A., Hubert, T., Baker, L., Lai, M., Bolton, A., Chen, Y., Lillicrap, T., Hui, F., Sifre, L., van den Driessche, G., Graepel, T., & Hassabis, D. (2017). Mastering the game of Go without human knowledge. Nature, 550(7676), 354.

ISSN: 2472-3703 DOI: 10.6911/WSRJ.202312_9(12).0015

[9] Ye, D., Chen, G., Zhang, W., Chen, S., Yuan, B., Liu, B., Chen, J., Liu, Z., Qiu, F., Yu, H., Yin, Y., Shi, B., Wang, L., Shi, T., Fu, Q., Yang, W., Huang, L., & Liu, W. (2020). Towards Playing Full MOBA Games with Deep Reinforcement Learning (arXiv:2011.12692). arXiv. http://arxiv.org/abs/2011.12692

[10] Peng, P., Wen, Y., Yang, Y., Yuan, Q., Tang, Z., Long, H., & Wang, J. (2017). Multiagent Bidirectionally-Coordinated Nets: Emergence of Human-level Coordination in Learning to Play StarCraft Combat Games (arXiv:1703.10069). arXiv. http://arxiv.org/abs/1703.10069

[11] Foerster, J., Farquhar, G., Afouras, T., Nardelli, N., & Whiteson, S. (2017). Counterfactual Multi-Agent Policy Gradients (arXiv:1705.08926). arXiv. http://arxiv.org/abs/1705.08926

[12] Atkinson, T., Baier, H., Copplestone, T., Devlin, S., & Swan, J. (2019). The Text-Based Adventure AI Competition. IEEE Transactions on Games, 11(3), 260–266.

[13] Hua, M., & Raley, R. (n.d.). Playing With Unicorns: AI Dungeon and Citizen NLP.

[14] Schick, T., Dwivedi-Yu, J., Dessì, R., Raileanu, R., Lomeli, M., Zettlemoyer, L., Cancedda, N., & Scialom, T. (2023). Toolformer: Language Models Can Teach Themselves to Use Tools (arXiv:2302.04761). arXiv. http://arxiv.org/abs/2302.04761

[15] Chase, H. W., Maass, L. E. S., & Gan, L. (2023). LangChain: Building applications with LLMs through composability. GitHub. https://github.com/hwchase17/langchain (Accessed on September 8, 2023)

[16] Vemprala, S., Bonatti, R., Bucker, A., & Kapoor, A. (n.d.). ChatGPT for Robotics: Design Principles and Model Abilities.

[17] Gisslén, L., Eakins, A., Gordillo, C., Bergdahl, J., & Tollmar, K. (2021). Adversarial Reinforcement Learning for Procedural Content Generation (arXiv:2103.04847). arXiv. http://arxiv.org/abs/2103.04847

[18] Thominet, L. (2021). Open Video Game Development and Participatory Design. Technical Communication Quarterly, 30(4), 359–374.