



Applications of Artificial Intelligence in Game Algorithms: History, Current Status, and Future Prospects

Hongyu Jiang

School of Economics and Management, Beijing Jiaotong University, Weihai, China
23711091@bjtu.edu.cn

Abstract The rapid advancement of Artificial Intelligence (AI) technology has significantly influenced the gaming industry, enhancing both player experiences and developer efficiency. This review explores key issues such as improving game interactivity and challenge through AI, and addressing technical challenges in game development. It examines the history, current state, and future trends of AI in game algorithms and its impact on the industry. AI in gaming creates responsive, adaptive, and challenging experiences using techniques like pathfinding, decision-making, machine learning, and procedural content generation. Theoretical models such as Finite State Machines, Behavior Trees, and Neural Networks are crucial for these applications. Significant research breakthroughs include reinforcement learning with AlphaGo and OpenAI Five, advancements in visual processing with Deep Convolutional Neural Networks, and enhanced dialogue systems via Natural Language Processing. Adaptive game design, including Dynamic Difficulty Adjustment (DDA) and player behavior analysis, personalizes gaming experiences. Emerging trends point to further AI integration in complex AI opponents, cross-platform support, and improved virtual and augmented reality games. Real-time content generation and automated game design are expected to reduce development costs and foster innovation. AI's role in game testing and quality assurance is also expanding, ensuring stability and performance. Overall, AI in gaming is advancing towards greater intelligence, personalization, and automation, promising richer, more interactive, and immersive experiences for players.

Keywords: Applications, Artificial Intelligence, Game Algorithms.

1 Introduction

The rapid development of Artificial Intelligence (AI) technology has profoundly impacted the gaming industry, enhancing both player experiences and developer efficiency. AI technology enables games to create more complex and challenging environments, which significantly improves player engagement while simultaneously alleviating the developers' workload. This review addresses two primary issues: enhancing game interactivity and challenge through AI technology and solving

technical problems in game development with AI. AI in gaming involves the use of sophisticated algorithms to create responsive, adaptive, and intelligent behaviors in non-player characters (NPCs) and game environments. Techniques such as pathfinding, decision-making, machine learning, and procedural content generation play crucial roles in making gameplay more immersive and dynamic. Theoretical models like Finite State Machines, Behavior Trees, and Neural Networks are fundamental in implementing these AI-driven applications. Noteworthy research milestones include advancements in reinforcement learning demonstrated by AlphaGo and OpenAI Five, improvements in visual processing through Deep Convolutional Neural Networks, and the development of enhanced dialogue systems using Natural Language Processing. Moreover, adaptive game design, which includes mechanisms like Dynamic Difficulty Adjustment (DDA) and player behavior analysis, is crucial for personalizing gaming experiences. These technologies allow games to adjust their difficulty levels in real-time based on the player's performance, ensuring an engaging and tailored experience.

2 Applications of AI in Games

2.1 Definitions and Basic Concepts

Artificial Intelligence (AI) in games involves the use of intelligent algorithms to create responsive, adaptive, and challenging gameplay experiences. AI techniques enable non-player characters (NPCs) to exhibit human-like behavior, making games more immersive and dynamic. In games, AI is used for pathfinding, allowing NPCs to efficiently navigate complex environments. It also drives decision-making, allowing NPCs to respond strategically to player actions. Machine learning can enhance AI by allowing game systems to learn and adapt to individual player styles, providing personalized gaming experiences [1]. AI-driven procedural content generation creates diverse and unpredictable game worlds, enhancing replayability. AI is also used in game testing and quality assurance to identify bugs and optimize game performance. AI applications in gaming extend to player behavior analysis, which helps developers understand player preferences and improve game design. Overall, AI technologies are playing a key role in advancing the complexity and realism of modern video games [2].

2.2 Relevant Theories and Models

The application of artificial intelligence (AI) in games uses various theories and models to create intelligent, dynamic, and immersive game experiences. Here are some key theories and models used in game AI [3].

Finite State Machines (FSM). FSMs are simple yet powerful models used to control the behavior of non-player characters (NPCs). Each state represents a specific behavior, and transitions between states are triggered by specific conditions, allowing for predictable and manageable AI actions [4].

Behavior Trees (BT). BTs provide a structured approach to designing complex NPC behaviors. They use a hierarchical node structure to perform tasks, allowing for greater flexibility and adaptability in response to changing game environments [5].

Pathfinding. Algorithms such as A* and Dijkstra's are essential for pathfinding, helping NPCs efficiently navigate complex game environments. These algorithms find the shortest or most efficient paths between points, enhancing the realism of NPC movement [6].

Decision Trees. Decision trees help the AI make choices in various game situations. By branching based on conditions, AI can select the most appropriate course of action, enabling strategic and context-aware behavior [7].

Neural Networks. Deep Neural Networks (DNNs) and Convolutional Neural Networks (CNNs) are widely used for image and pattern recognition, enhancing visual processing and decision-making capabilities in games [8].

Genetic algorithms (GA). GAs simulate natural selection and genetic variation to optimize AI behavior and strategies. This approach is particularly useful for solving complex optimization problems in games [9].

Procedural Content Generation (PCG). PCG uses algorithms to automatically create diverse and unpredictable game content. This includes level design, character generation, and storyline creation, enhancing replay ability and game richness.

Fuzzy Logic. Fuzzy logic enables AI to handle uncertain and imprecise information and make more human-like decisions under ambiguity. This is especially useful in games that require nuanced NPC behavior [10].

2.3 Major Research Findings

In the gaming industry, the application of artificial intelligence algorithms has led to several significant research outcomes. Here are some of the current major research directions and achievements:

Reinforcement Learning. AlphaGo: Developed by DeepMind, AlphaGo used reinforcement learning and Monte Carlo Tree Search (MCTS) to defeat human champions in the game of Go, marking a breakthrough in complex strategy games.

OpenAI Five. OpenAI's system achieved notable success in the multiplayer online battle game Dota 2, showcasing AI's potential in complex real-time strategy games.

Neural Networks and Deep Learning. Deep Convolutional Neural Networks (DCNN): It is widely used for visual processing in real-time strategy games, such as movement detection and image recognition for each unit in StarCraft II.

Generative Adversarial Networks (GANs). It is used to generate more realistic game images and battle scenes, effectively enhancing the visual diversity of the game.

Natural Language Processing in Games. Dialogue Systems: This AI is mostly used to create more natural and realistic conversations between NPC and players in role-playing games (RPG) to enhance player immersion.

Text Generation. Convert the quests issued by NPC and the storyline during the game into text output to improve the playability of the game and make the game content richer.

Adaptive Game Design and Personalization. Dynamic Difficulty Adjustment (DDA): The AI algorithm adjusts the difficulty of the game in real time according to the performance of the player at different levels of difficulty in order to achieve the goal of balancing the player's skill level with the difficulty of the game. Enhance the playability and challenge of the game.

Player Behavior Analysis. Mostly used in role-playing games (RPG), where machine learning is used to analyze the player's behavior at various stages and provide the player with a personalized gaming experience.

Pathfinding and Motion Planning. A algorithm*: Used to solve pathfinding problems in games, including the production of miniature maps and wayfinding systems, widely applied in strategy and role-playing games.

Monte Carlo Tree Search (MCTS). Combined with reinforcement learning, MCTS is used for action decision-making in complex strategy and board games.

Game Testing and Quality Assurance. Automated Testing Tools: AI generates test cases and scenarios automatically, improving the efficiency and coverage of game testing.

Bug Detection and Fixing: AI helps identify and fix bugs and vulnerabilities in games, ensuring stability and reliability.

2.4 Emerging Trends and Developments

The application of AI algorithms in gaming has shown several major trends. Firstly, reinforcement learning is widely used to significantly improve AI performance in

complex strategy games, such as the success of AlphaGo in Go and OpenAI Five in Dota 2. Secondly, deep learning makes game images and scenes more realistic, and natural language processing technology makes in-game dialog more natural and realistic. Thirdly, personalized and adaptive gaming experience has become a trend, and it has become a mainstream trend for AI to adjust the difficulty of the game in real time according to the player's skill level. In terms of automated testing and quality assurance, AI-generated test cases to ensure game stability are widely used for game testing and post-production assurance.

Looking to the future, AI in gaming may develop more complex AI opponents and NPC, creating more challenging and interactive gaming experiences. Cross-platform and multi-device support will enhance player experience consistency, while AI applications in virtual and augmented reality games will make virtual worlds more realistic and interactive. Real-time content generation will make each gaming experience unique, increasing freshness and playability. AI will further automate game design and development processes, reducing development costs, enhancing creative efficiency, and driving the emergence of more innovative games. Social and multiplayer gaming experiences will also be optimized, improving game sociability and community feel by analyzing player behavior and social networks. Overall, AI applications in gaming will continue to evolve towards greater intelligence, personalization, and automation, delivering richer, more interactive, and immersive gaming experiences for players.

3 AI in Game Difficulty Adjustment

3.1 Summary of Relevant Research Findings

Dynamic Difficulty Adjustment (DDA) adjusts game difficulty in real-time based on the player's performance, which research indicates improves player engagement and satisfaction by keeping the game challenging but not frustrating. Methods for DDA include adjusting enemy AI behavior, resource availability, and game environment difficulty, as seen in Valve's AI Director in games such as "Left 4 Dead," which adapts enemy spawns and item placements based on player stress levels. Reinforcement Learning (RL) uses trial-and-error learning to optimize game difficulty, with algorithms like Q-learning and Deep Q-Networks (DQN) effectively personalizing game experiences and maintaining player interest by learning from player interactions. Bayesian Networks model player behavior and game states probabilistically to adjust difficulty, predicting player frustration or boredom to adjust difficulty levels accordingly, offering a probabilistic framework for handling uncertainty in player performance. REAs (Region-based Evolutionary Algorithms) are a class of evolutionary algorithms designed for multi-objective optimization problems, which improve search efficiency and diversity of settlement through mechanisms such as region partitioning and region competition.

3.2 Comparison and Contrast between Studies

REAs vs MOEA/D: REAs algorithm operates by dividing the search space into several regions, each corresponding to a problem, where selection, crossover, and mutation operations are performed within each region, and regions are evaluated, with resources reallocated based on performance, allowing superior regions to receive more resources. This enhances solution diversity and global search capability, making REAs particularly effective in complex and high-dimensional optimization problems. There is a flow chart of REAs algorithm. As shown in Fig 1.

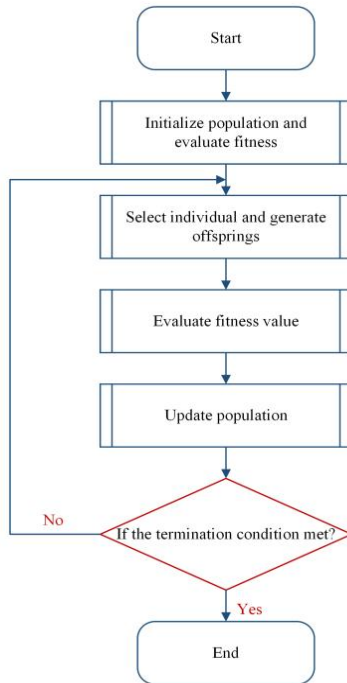


Fig. 1 Flow chart (Photo credit: Original).

On the other hand, RMOEA uses reinforcement learning methods like Q-learning and DQN to dynamically adjust control parameters and strategies during the evolutionary process, learning and dynamically adjusting the objective function through inverse reinforcement learning (IRL) to guide the search direction. This approach allows MOEA/D to dynamically adapt search strategies and parameters based on real-time feedback, improving search efficiency and exhibiting strong performance in multi-objective optimization problems by finding better Pareto frontiers. As shown in Fig 2. Here’s a summary diagram of the MOEA/D algorithmic process.

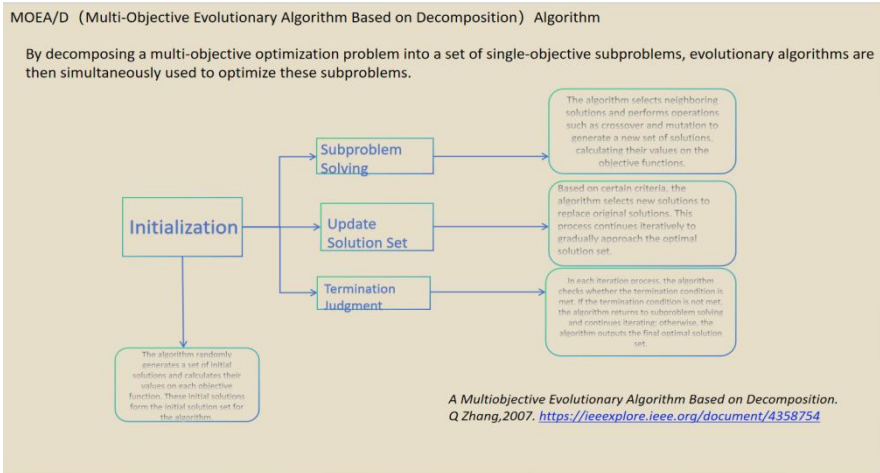


Fig. 2 Structure chart (Photo credit: Original).

3.3 Consistencies and Differences

Despite the variety of AI algorithms for game difficulty control and the different focuses of the research methods, they all consistently emphasize the importance of dynamically adjusting the game difficulty to improve the player experience. However, there are differences in processing complexity and efficiency between different algorithms. With this figure can clearly see the comparison of the three different algorithms that require a four-environment frame rate overhead in the simulated environment of the same game, Sinister Souls. The superiority of the MOEA/D algorithm is also demonstrated. Continuously improving algorithms to make them more efficient while using the same resources is also a hot research topic about AI algorithms today.

3.4 Limitations and Controversies

The application of AI models in actual game difficulty adjustment presents both promising potential and significant limitations. AI-driven dynamic difficulty adjustment (DDA) aims to enhance player engagement by maintaining an optimal balance between challenge and enjoyment. Real-time adaptability is crucial but poses challenges due to the high computational power required to process extensive gameplay data and make instantaneous adjustments. This can lead to latency issues, which can negatively impact the gaming experience. Additionally, AI models often struggle to accurately predict player behavior due to the dynamic and unpredictable nature of human actions, resulting in inappropriate difficulty adjustments that can frustrate players. Moreover, ensuring ethical use of AI is paramount to avoid manipulating difficulty settings for monetization purposes, which could erode player trust and lead to fairness concerns.

In conclusion, while AI models offer substantial benefits for dynamic difficulty adjustment in games, their application is constrained by real-time adaptability issues, predictive accuracy limitations, ethical considerations, resource demands,

generalization challenges, integration difficulties, and the need to preserve player autonomy. Overcoming these hurdles through continued innovation and ethical practices will be key to fully leveraging AI in creating more engaging and personalized gaming experiences

4 AI in MOBA Games

4.1 Case Studies of Practical Applications

OpenAI Five is a project developed by OpenAI with the goal of using reinforcement learning to train agents capable of competing against human players in the multiplayer online battle arena (MOBA) game Dota 2. This project demonstrates the potential of AI in handling complex, real-time strategy games that require teamwork and adaptability. It uses the Proximal Policy Optimization (PPO) algorithm, which is a self-playing game with extensive simulation training to deal with complex teamwork and confrontation.

AlphaStar is a project developed by DeepMind, aimed at using deep reinforcement learning to train agents to compete with human top players in StarCraft II, a complex real-time strategy (RTS) game. StarCraft II requires players to manage resources, build units, and devise strategies in a dynamic and partially observable environment, making it an ideal testbed for advanced AI techniques. After a period of learning, it managed to beat a number of top professional players, demonstrating the excellence of AI in complex implementation strategy games.

MOBA games are real-time confrontation games, and in most cases there are only two camps. Therefore, the AI of MOBA games are all trained by virtual self-gaming. This approach has many benefits, for example, it does not need to learn by collecting expensive expert data, and it can learn better strategies than human in the training process of self-gaming. Although these AIs are developed by different companies and are used in different MOBA games, the underlying algorithms are pretty much the same. Among the virtual self-play algorithms of current games, naive self-play is the most classic and fundamental one, which chooses the newest strategy as the opponent every time. The following is its sampling strategy distribution function and strategy set update function.

$$\omega(\pi' | \pi^0, \pi) = \begin{cases} 1 & \forall \pi' \in \pi': \pi' == \pi \\ 0 & \text{Other Situation} \end{cases} \quad (1)$$

Sampling strategy distribution function.

$$G(\pi^0, \pi) = \{\pi\} \quad (2)$$

Strategy set update function.

And the basic idea of the algorithm is shown in Fig 3.

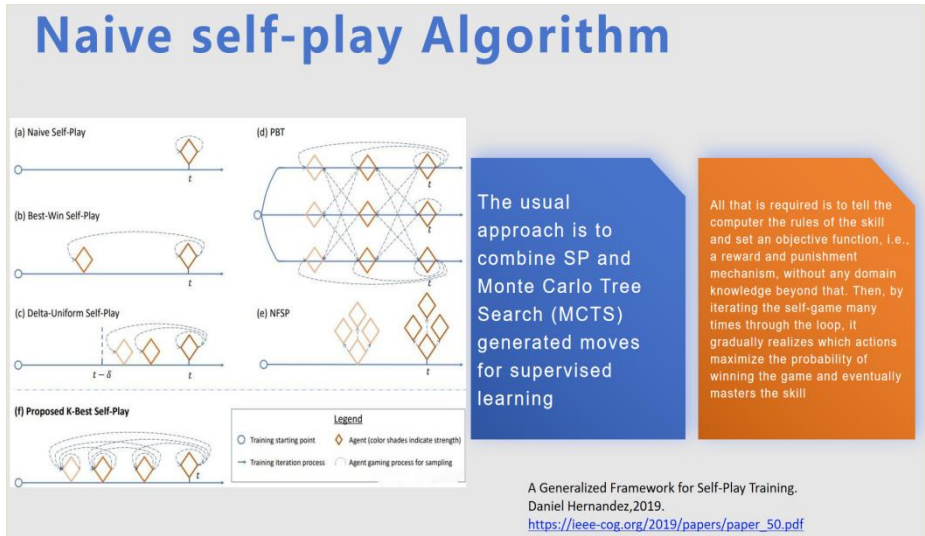


Fig. 3 Structural diagram (Photo credit: Original).

4.2 Impact of Research on Practice

The application of Artificial Intelligence (AI) in Multiplayer Online Battle Arena (MOBA) games has significantly influenced both the development and the playability of these games. AI technologies such as reinforcement learning, neural networks, and advanced decision-making algorithms have been integrated into MOBA games, enhancing their complexity and player engagement.

Enhanced Player Experience. AI has been pivotal in personalizing and enhancing the player experience. Through machine learning techniques, AI systems can adapt in real-time to individual player behaviors and skill levels. This adaptability ensures that players are continually challenged but not overwhelmed. For example, AI can adjust the difficulty of in-game opponents based on the player's performance, providing a tailored experience that keeps players engaged and motivated.

Game Development and Testing. AI has also streamlined game development and testing processes. AI-driven tools can automatically generate and test numerous scenarios, identifying bugs and balancing issues more efficiently than traditional methods. This not only accelerates the development cycle but also ensures a higher quality product. Automated testing can simulate thousands of matches, providing developers with insights into potential problems and allowing them to make data-driven decisions.

Real-time Strategy and Decision Making. The ability of Artificial Intelligence to process and analyze real-time data greatly impacts strategic decision-making in

MOBA games. AI systems can provide real-time analysis and recommendations to help players make informed decisions during gameplay. This help can be applied in other ways entirely. For example, AI can suggest optimal item builds, skill upgrades, and strategic moves based on the current state of the game. Such guidance is especially important in high-stakes professional eSports environments, where split-second decisions can determine the outcome of a match.

Challenges and Solutions in Practical Applications. Firstly, MOBA games like Dota 2 and League of Legends are characterized by their complex, real-time strategy environments where players must make split-second decisions based on a myriad of variables. This complexity poses a significant challenge for AI systems, which need to process vast amounts of data and make decisions quickly and accurately. Game makers can use advanced reinforcement learning techniques to solve this problem. Examples include Proximal Policy Optimization (PPO): an algorithm used to stabilize training and improve the AI's learning efficiency in complex environments. PPO helps manage the trade-off between exploration and exploitation, allowing the AI to adapt more effectively to dynamic game states.

Secondly, In MOBA games, the game state is constantly changing due to player actions, making the environment highly dynamic and non-stationary. AI systems must continuously adapt to these changes, requiring sophisticated learning algorithms that can handle real-time updates and strategy adjustments. Hierarchical and Modular Approaches can be a good solution to this problem. Its breaking down the AI decision-making process into smaller, more manageable modules helps address the complexity of MOBA games. Hierarchical reinforcement learning allows the AI to first learn high-level strategies and then focus on finer details, thus improving overall performance.

5 Research Gaps and Future Directions

For the game difficulty control AI algorithm, the MOEA/D algorithm described previously is a distributed algorithm parallelized with a population, which runs with a distribution of CPUs only. For the behavior of repeating the same operation recursively, GPUs are more appropriate. The algorithm can be subsequently parallelized with GPUs to improve the efficiency of the operation. The following figure shows the efficiency difference between running the same code (triple loop to realize matrix output) using CPU and GPU respectively, As shown in Fig 4.

```
~/C/vs $ g vector_mul.cpp -o vector_mul_cpu&&nvcc vector_mul.cu -o vector_mul1_gpu
~/C/vs $ ./vector_mul_gpu&&./vector_mul_cpu
GPU Time: 2616 microseconds
CPU Time: 12823628 microseconds
~/C/vs $ bc
bc 1.07.1
Copyright 1991-1994,1997,1998,2000,2004,2006,2008,2012-2017 Free Software Foundation, Inc/
```

Fig. 4 Sample code (Photo credit: Original).

In the naive self-play algorithm, the intelligent body is constantly playing against its current, most recent self. However, as learning proceeds, this approach can lead to a cyclic situation where the intelligent body is constantly playing against its past self without being exposed to new, stronger opponents or strategies. Because it cannot gain new learning opportunities from confrontations with stronger opponents, there will be a ceiling in the progress of the intelligence. To solve this problem, multiple intelligence can be used against each other, each using a different strategy, so that the intelligence can learn and evolve against different opponents. In addition, some heuristics can be used to avoid loops, such as periodically introducing new randomized strategies or periodically resetting the state of the intelligence.

Even for the increasingly sophisticated AI algorithms, there are gaps in their research. For example, the application of AI in game sentiment analysis can play a key role in role-playing games (RPG) such as *Zelda* and open-world games such as *Red Dead Redemption*. Nowadays, virtual reality and augmented reality technologies are developing rapidly, and how to apply AI algorithms to virtual reality and augmented reality games is an unprecedented research direction. Its research value and market potential are huge.

6 Conclusion

As a result of this review, it is evident that AI technology has become a key force driving innovation and development in the gaming industry. Whether it involves AI algorithms automatically controlling and adjusting game difficulty, continuously updating the intelligence of robot players in MOBA games, or other research areas, the application of AI in game design and development shows immense promise while presenting numerous challenges. In this context, it is crucial for game developers and researchers to deepen their understanding of AI technology and implement the most advanced models in practice. Additionally, they must consider the ethical implications and user experience to ensure that AI algorithms and games enhance each other. By doing so, the gaming industry can harness the full potential of AI, leading to more engaging, dynamic, and intelligent gaming experiences.

References

1. Hernandez, D., Denamganai, K., Gao, Y., York, P., Devlin, S., Samothrakis, S., Walker, J. A.: A Generalized Framework for Self-Play Training. In: 2019 IEEE Conference on Games (COG) (2019).
2. Oesterheld, C., Gleave, A.: Survey of Self-Play in Reinforcement Learning. (2021).
3. Zhang, Q.: A Multiobjective Evolutionary Algorithm Based on Decomposition. *IEEE Transactions on Evolutionary Computation* 11(6), 773-795 (2007).
4. Song, Y., et al.: Reinforcement Learning-assisted Evolutionary Algorithm: A Survey and Research Opportunities. *arXiv preprint* (2024).
5. Souchleris, K., et al.: Reinforcement Learning in Game Industry—Review, Prospects and Challenges. *Applied Sciences* 13(4), 2443 (2023).

6. Zhu, X., Xu, H., Zhao, Z., et al.: An Environmental Intrusion Detection Technology Based on WiFi. *Wireless Personal Communications* 119(2), 1425-1436 (2021).
7. Zhang, C., Lu, Y.: Study on Artificial Intelligence: The State of the Art and Future Prospects. *Journal of Industrial Information Integration* 23, 100224 (2021).
8. Sharma, S., Singh, S.: Vision-Based Hand Gesture Recognition Using Deep Learning for the Interpretation of Sign Language. *Expert Systems with Applications* 182, 115657 (2021).
9. Pan, Y.: Heading Toward Artificial Intelligence 2.0. *Engineering* 2(4), 409-413 (2016).
10. Manlhiot, C., et al.: A Primer on the Present State and Future Prospects for Machine Learning and Artificial Intelligence Applications in Cardiology. *Canadian Journal of Cardiology* 38(2), 169–184 (2022).

Open Access This chapter is licensed under the terms of the Creative Commons Attribution-NonCommercial 4.0 International License (<http://creativecommons.org/licenses/by-nc/4.0/>), which permits any noncommercial use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license and indicate if changes were made.

The images or other third party material in this chapter are included in the chapter's Creative Commons license, unless indicated otherwise in a credit line to the material. If material is not included in the chapter's Creative Commons license and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder.

