Title: Machine learning in video games

URL: https://en.wikipedia.org/wiki/Machine_learning_in_video_games

PageID: 60951296

Categories: Category:Game artificial intelligence, Category:Machine learning

-----

Artificial intelligence and machine learning techniques are used in video games for a wide variety of applications such as non-player character (NPC) control, procedural content generation (PCG) and deep learning-based content generation. Machine learning is a subset of artificial intelligence that uses historical data to build predictive and analytical models. This is in sharp contrast to traditional methods of artificial intelligence such as search trees and expert systems .

Information on machine learning techniques in the field of games is mostly known to public through research projects as most gaming companies choose not to publish specific information about their intellectual property . The most publicly known application of machine learning in games is likely the use of deep learning agents that compete with professional human players in complex strategy games . There has been a significant application of machine learning on games such as Atari /ALE, Doom , Minecraft , StarCraft , and car racing. [1] Other games that did not originally exists as video games, such as chess and Go have also been affected by the machine learning. [2]

Overview of relevant machine learning techniques

Deep learning

Deep learning is a subset of machine learning which focuses heavily on the use of artificial neural networks (ANN) that learn to solve complex tasks. Deep learning uses multiple layers of ANN and other techniques to progressively extract information from an input. Due to this complex layered approach, deep learning models often require powerful machines to train and run on.

Convolutional neural networks

Convolutional neural networks (CNN) are specialized ANNs that are often used to analyze image data. These types of networks are able to learn translation invariant patterns, which are patterns that are not dependent on location. CNNs are able to learn these patterns in a hierarchy, meaning that earlier convolutional layers will learn smaller local patterns while later layers will learn larger patterns based on the previous patterns. [3] A CNN's ability to learn visual data has made it a commonly used tool for deep learning in games. [4] [5]

Recurrent neural network

Recurrent neural networks are a type of ANN that are designed to process sequences of data in order, one part at a time rather than all at once. An RNN runs over each part of a sequence, using the current part of the sequence along with memory of previous parts of the current sequence to produce an output. These types of ANN are highly effective at tasks such as speech recognition and other problems that depend heavily on temporal order. There are several types of RNNs with different internal configurations; the basic implementation suffers from a lack of long term memory due to the vanishing gradient problem , thus it is rarely used over newer implementations. [3]

Long short-term memory

A long short-term memory (LSTM) network is a specific implementation of a RNN that is designed to deal with the vanishing gradient problem seen in simple RNNs, which would lead to them gradually "forgetting" about previous parts of an inputted sequence when calculating the output of a current part. LSTMs solve this problem with the addition of an elaborate system that uses an additional input/output to keep track of long term data. [3] LSTMs have achieved very strong results across various fields, and were used by several monumental deep learning agents in games. [6] [4]

Reinforcement learning

Reinforcement learning is the process of training an agent using rewards and/or punishments. The way an agent is rewarded or punished depends heavily on the problem; such as giving an agent a positive reward for winning a game or a negative one for losing. Reinforcement learning is used heavily in the field of machine learning and can be seen in methods such as Q-learning , policy search , Deep Q-networks and others. It has seen strong performance in both the field of games and robotics . [7]

Neuroevolution

Neuroevolution involves the use of both neural networks and evolutionary algorithms . Instead of using gradient descent like most neural networks, neuroevolution models make use of evolutionary algorithms to update neurons in the network. Researchers claim that this process is less likely to get stuck in a local minimum and is potentially faster than state of the art deep learning techniques. [8]

Deep learning agents

Machine learning agents have been used to take the place of a human player rather than function as NPCs, which are deliberately added into video games as part of designed gameplay . Deep learning agents have achieved impressive results when used in competition with both humans and other artificial intelligence agents. [2] [9]

Chess

Chess is a turn-based strategy game that is considered a difficult AI problem due to the computational complexity of its board space. Similar strategy games are often solved with some form of a Minimax Tree Search. These types of AI agents have been known to beat professional human players, such as the historic 1997 Deep Blue versus Garry Kasparov match. Since then, machine learning agents have shown ever greater success than previous AI agents.

Go

Go is another turn-based strategy game which is considered an even more difficult AI problem than chess. The state space of is Go is around 10^170 possible board states compared to the 10^120 board states for Chess. Prior to recent deep learning models, AI Go agents were only able to play at the level of a human amateur. [5]

AlphaGo

Google's 2015 AlphaGo was the first AI agent to beat a professional Go player. [5] AlphaGo used a deep learning model to train the weights of a Monte Carlo tree search (MCTS). The deep learning model consisted of 2 ANN, a policy network to predict the probabilities of potential moves by opponents, and a value network to predict the win chance of a given state. The deep learning model allows the agent to explore potential game states more efficiently than a vanilla MCTS. The network were initially trained on games of humans players and then were further trained by games against itself.

AlphaGo Zero

AlphaGo Zero , another implementation of AlphaGo, was able to train entirely by playing against itself. It was able to quickly train up to the capabilities of the previous agent. [10]

StarCraft series

StarCraft and its sequel StarCraft II are real-time strategy (RTS) video games that have become popular environments for AI research. Blizzard and DeepMind have worked together to release a public StarCraft 2 environment for AI research to be done on. [11] Various deep learning methods have been tested on both games, though most agents usually have trouble outperforming the default AI with cheats enabled or skilled players of the game. [1]

Alphastar

Alphastar was the first AI agent to beat professional StarCraft 2 players without any in-game advantages. The deep learning network of the agent initially received input from a simplified zoomed out version of the gamestate, but was later updated to play using a camera like other human players. The developers have not publicly released the code or architecture of their model, but have listed several state of the art machine learning techniques such as relational deep reinforcement learning, long short-term memory , auto-regressive policy heads, pointer networks, and centralized value baseline. [4] Alphastar was initially trained with supervised learning, it watched replays of many human games in order to learn basic strategies. It then trained against different versions of itself and was improved through reinforcement learning. The final version was hugely successful, but only trained to play on a specific map in a protoss mirror matchup.

Dota 2

Dota 2 is a multiplayer online battle arena (MOBA) game. Like other complex games, traditional AI agents have not been able to compete on the same level as professional human player. The only widely published information on AI agents attempted on Dota 2 is OpenAI 's deep learning Five agent.

OpenAI Five

OpenAI Five utilized separate long short-term memory networks to learn each hero. It trained using a reinforcement learning technique known as Proximal Policy Learning running on a system containing 256 GPUs and 128,000 CPU cores . [6] Five trained for months, accumulating 180 years of game experience each day, before facing off with professional players. [12] [13] It was eventually able to beat the 2018 Dota 2 esports champion team in a 2019 series of games.

Planetary Annihilation

Planetary Annihilation is a real-time strategy game which focuses on massive scale war. The developers use ANNs in their default AI agent. [14]

Supreme Commander 2

Supreme Commander 2 is a real-time strategy (RTS) video game.

The game uses Multilayer Perceptrons (MLPs) to control a platoon’s reaction to encountered enemy units. Total of four MLPs are used, one for each platoon type: land, naval, bomber, and fighter. [15]

Gran Turismo

Gran Turismo is a PlayStation game franchise that simulates realistic racing and driving experiences. In 2022, Sony AI researchers presented Sophy, an agent which can play Gran Turismo with performance on par with or superior to the world’s best e-sports drivers. [16] [17] The implemented solution is based on model-free deep reinforcement learning .

Generalized games

There have been attempts to make machine learning agents that are able to play more than one game. These "general" gaming agents are trained to understand games based on shared properties between them.

AlphaZero

AlphaZero is a modified version of AlphaGo Zero which is able to play Shogi , chess , and Go . The modified agent starts with only basic rules of the game, and is also trained entirely through self-learning. DeepMind was able to train this generalized agent to be competitive with previous versions of itself on Go, as well as top agents in the other two games. [2]

Strengths and weaknesses of deep learning agents

Machine learning agents are often not covered in many game design courses. Previous use of machine learning agents in games may not have been very practical, as even the 2015 version of AlphaGo took hundreds of CPUs and GPUs to train to a strong level. [2] This potentially limits the creation of highly effective deep learning agents to large corporations or extremely wealthy individuals. The extensive training time of neural network based approaches can also take weeks on these powerful machines. [4]

The problem of effectively training ANN based models extends beyond powerful hardware environments; finding a good way to represent data and learn meaningful things from it is also often a difficult problem. ANN models often overfit to very specific data and perform poorly in more generalized cases. AlphaStar shows this weakness, despite being able to beat professional players, it is only able to do so on a single map when playing a mirror protoss matchup. [4] OpenAI Five also shows this weakness, it was only able to beat professional player when facing a very limited hero pool out of the entire game. [13] This example show how difficult it can be to train a deep learning agent to perform in more generalized situations.

Machine learning agents have shown great success in a variety of different games. [12] [2] [4] However, agents that are too competent also risk making games too difficult for new or casual players. Research has shown that challenge that is too far above a player's skill level will ruin lower player enjoyment. [18] These highly trained agents are likely only desirable against very skilled human players who have many of hours of experience in a given game. Given these factors, highly effective deep learning agents are likely only a desired choice in games that have a large competitive scene, where they can function as an alternative practice option to a skilled human player.

Computer vision-based players

Computer vision focuses on training computers to gain a high-level understanding of digital images or videos. Many computer vision techniques also incorporate forms of machine learning, and have been applied on various video games. This application of computer vision focuses on interpreting game events using visual data. In some cases, artificial intelligence agents have used model-free techniques to learn to play games without any direct connection to internal game logic, solely using video data as input.

Pong

Andrej Karpathy has demonstrated that relatively trivial neural network with just one hidden layer is capable of being trained to play Pong based on screen data alone. [19] [20]

Atari games

In 2013, a team at DeepMind demonstrated the use of deep Q-learning to play a variety of Atari video games — Beamrider , Breakout , Enduro , Pong , Q*bert , Seaquest , and Space Invaders — from screen data. [21] The team expanded their work to create a learning algorithm called MuZero that was able to "learn" the rules and develop winning strategies for over 50 different Atari games based on screen data. [22] [23]

Doom

Doom (1993) is a first-person shooter (FPS) game. Student researchers from Carnegie Mellon University used computer vision techniques to create an agent that could play the game using only image pixel input from the game. The students used convolutional neural network (CNN) layers to interpret incoming image data and output valid information to a recurrent neural network which was responsible for outputting game moves. [24]

Super Mario

Other uses of vision-based deep learning techniques for playing games have included playing Super Mario Bros. only using image input, using deep Q-learning for training. [19]

Minecraft

Researchers with OpenAI created about 2000 hours of video plays of Minecraft coded with the necessary human inputs, and then trained a machine learning model to comprehend the video feedback from the input. The researchers then used that model with 70,000 hours of Minecraft playthroughs offered on YouTube to see how well the model could create the input to match that behavior and learn further from it, such as being able to learn the steps and process of creating a diamond pickaxe tool. [25] [26]

Machine learning for procedural content generation in games

Machine learning has seen research for use in content recommendation and generation. Procedural content generation is the process of creating data algorithmically rather than manually. This type of content is used to add replayability to games without relying on constant additions by human developers. PCG has been used in various games for different types of content generation, examples of which include weapons in Borderlands 2 , [27] all world layouts in Minecraft [28] and entire universes in No Man's Sky . [29] Common approaches to PCG include techniques that involve grammars , search-based algorithms , and logic programming . [30] These approaches require humans to manually define the range of content possible, meaning that a human developer decides what features make up a valid piece of generated content. Machine learning is theoretically capable of learning these features when given examples to train off of, thus greatly reducing the complicated step of developers specifying the details of content design. [31] Machine learning techniques used for content generation include Long Short-Term Memory (LSTM) Recurrent Neural Networks (RNN), Generative Adversarial networks (GAN), and K-means clustering . Not all of these techniques make use of ANNs, but the rapid development of deep learning has greatly increased the potential of techniques that do. [31]

Galactic Arms Race

Galactic Arms Race is a space shooter video game that uses neuroevolution powered PCG to generate unique weapons for the player. This game was a finalist in the 2010 Indie Game Challenge and its related research paper won the Best Paper Award at the 2009 IEEE Conference on Computational Intelligence and Games. The developers use a form of neuroevolution called cgNEAT to generate new content based on each player's personal preferences. [32]

Each generated item is represented by a special ANN known as a Compositional Pattern Producing Network (CPPNs). During the evolutionary phase of the game cgNEAT calculates the fitness of current items based on player usage and other gameplay metrics, this fitness score is then used decide which CPPNs will reproduce to create a new item. The ending result is the generation of new weapon effects based on the player's preference.

Super Mario Bros.

Super Mario Bros. has been used by several researchers to simulate PCG level creation. Various attempts having used different methods. A version in 2014 used n-grams to generate levels similar to the ones it trained on, which was later improved by making use of MCTS to guide generation. [33] These generations were often not optimal when taking gameplay metrics such as player movement into account, a separate research project in 2017 tried to resolve this problem by generating levels based on player movement using Markov Chains. [34] These projects were not subjected to human testing and may not meet human playability standards.

The Legend of Zelda

PCG level creation for The Legend of Zelda has been attempted by researchers at the University of California, Santa Cruz. This attempt made use of a Bayesian Network to learn high level knowledge from existing levels, while Principal Component Analysis (PCA) was used to represent the different low level features of these levels. [35] The researchers used PCA to compare generated levels to human made levels and found that they were considered very similar. This test did not include playability or human testing of the generated levels.

Deep learning for content generation in games

The introduction of Generative Adversarial Networks first, and then of diffusion models allows for generating in-game content at runtime using non-procedural approaches. Examples include:

The 3D printer available in InZOI (available in early access), a life simulation game developed by InZOI Studio and published by Krafton . Given a 2D image file provided by the user, the 3D printer generates the corresponding 3D object, which the user can then place in the game or use to decorate its avatar. [36]

The terrain generation solution implemented in Prologue: Go Wayback! (early access: to be announced, with play tests undergoing in early 2025), a survival game developed by P layerUnknown Productions . The approach allows to generate a new game map every time the game is launched. [37]

Music generation

Music is often seen in video games and can be a crucial element for influencing the mood of different situations and story points. Machine learning has seen use in the experimental field of music generation; it is uniquely suited to processing raw unstructured data and forming high level representations that could be applied to the diverse field of music. [38] Most attempted methods have involved the use of ANN in some form. Methods include the use of basic feedforward neural networks , autoencoders , restricted boltzmann machines , recurrent neural networks , convolutional neural networks , generative adversarial networks (GANs), and compound architectures that use multiple methods. [38]

VRAE video game melody symbolic music generation system

The 2014 research paper on "Variational Recurrent Auto-Encoders" attempted to generate music based on songs from 8 different video games. This project is one of the few conducted purely on video game music. The neural network in the project was able to generate data that was very similar to the data of the games it trained off of. [39] The generated data did not translate into good quality music.

References

External links

v

t

e

Differentiable programming

Information geometry

Statistical manifold

Automatic differentiation

Neuromorphic computing

Pattern recognition

Ricci calculus

Computational learning theory

Inductive bias

IPU

TPU

VPU

Memristor

SpiNNaker

TensorFlow

PyTorch

Keras

scikit-learn

Theano

JAX

Flux.jl

MindSpore

Portals Computer programming Technology

Computer programming

Technology