Proceedings of the 2nd International Conference on Software Engineering and Machine Learning
DOI: 10.54254/2755-2721/73/20240386

An exploration of reinforcement learning and deep reinforcement learning

Zisheng Fan

Hefei New Channel School, Hefei, 230041, China

1919213554@qq.com

Abstract. Today, machine learning is evolving so quickly that new algorithms are always appearing. Deep neural networks in particular have shown positive outcomes in a variety of areas, including computer vision, natural language processing, and time series prediction. Its development moves at a very sluggish pace due to the high threshold. Therefore, a thorough examination of the reinforcement learning field should be required. This essay examines both the deep learning algorithm and the reinforcement learning operational procedure. The study identifies information retrieval, data mining, intelligent speech, natural language processing, and reinforcement learning as key technologies. The scientific study of reinforcement learning has advanced remarkably quickly, and it is now being used to tackle important decision optimization issues at academic conferences and journal research work in computer networks, computer graphics, etc. Brief introductions and reviews of both types of models are provided in this paper, along with an understanding of some of the most cutting-edge reinforcement learning applications and approaches.

Keywords: Reinforcement learning, deep reinforcement learning, machine learning.

1. Introduction

Sequential tasks, a significant class of problems with similarities to real decisions, are used in machine learning. Tasks involving prediction and decision-making are distinct. The decision-maker must take responsibility for the future and make more decisions at a later time since decisions frequently have "consequences." This book explores the reinforcement learning method for sequential machine learning. Simply defined, forecasting creates a signal for input data and anticipates that it will match the observed signal in the future. The situation in the future is unaffected by this. This article focuses mostly on the fundamental ideas and approaches to understanding reinforcement learning and deep reinforcement learning.

2. Reinforcement learning

Reinforcement learning, in general, is a computational technique in which machines engage with their surroundings to accomplish objectives. A round of interaction between the machine and the environment is when the machine decides what to do in response to a state of the environment, does the action, the environment changes as a result, and then the machine receives the reward feedback and the next round of state. The machine aims to maximize the expectation of cumulative reward

DOI: 10.54254/2755-2721/73/20240386

earned across several rounds of interaction during this iterative engagement [1]. The idea of an agent is used in reinforcement learning to depict a machine that makes choices.

In contrast to the "model" in supervised learning, the "agent" in reinforcement learning stresses that the machine may actively alter the environment by making decisions rather than merely providing certain prediction signals [2].

Here, the agent has three key elements, namely perception, decision-making and reward.

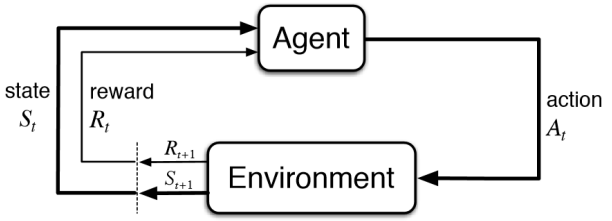

Figure 1. Basic principle of RL

In terms of perception, the agent has a general perception of the status of the environment and is aware of its present state. Examples include a Go agent that perceives the current state of the chessboard; an unmanned vehicle that senses nearby traffic lights, vehicles, and pedestrians; and a robot dog that uses a camera to sense what is in front of it as well as mechanical sensors on its soles to sense the friction and tilt of the ground. level, etc.

Decision-making is the process by which an agent determines the steps it must take in light of its present condition in order to attain its goal. For instance, the unmanned vehicle calculates the angle of the steering wheel and the strength of the brake and accelerator based on the current road conditions; the robot dog provides 4 based on the currently collected visual and force signals; and so forth. the leg of the gear's angle of rotational speed. The main distinction between various agents is their strategy, which is the type of intelligence that an agent eventually embodies.

In terms of award, the environment produces a scalar signal as reward feedback based on the agent's condition and actions [3]. This scalar signal evaluates how well the agent performed in this round. The outcome of a game of Go, the unmanned vehicle's ability to drive fast, safely, and smoothly, or even whether the robot dog is able to go forward without falling are a few examples. The objective of the agent's improvement strategy is to maximize the cumulative reward expectation, which is also a crucial metric for assessing how effective the approach is.

There are significant form differences between supervised learning for prediction tasks and reinforcement learning for decision-making tasks. First, prediction tasks are always single-round independent tasks, but decision-making tasks frequently entail many rounds of interaction [4], i.e., sequential decision-making. Decision-making may be turned into a prediction problem of "discriminating the optimal action" if it also occurs in a single round. Second, because the decision-making job involves several rounds, the agent must take into account the corresponding changes in the future environment while making decisions in each round. As a result, the action that generates the most positive reward feedback in the immediate round may not be the best option over time.

As can be seen, a reinforcement learning agent interacts with a dynamic environment to fulfill consecutive judgments. When the author remarks that a situation is dynamic, we suggest that it will keep changing as other circumstances change. In both mathematics and science, stochastic processes are frequently used to describe this. In actuality, virtually every aspect of existence is in a state of evolution, including a galaxy, a football game, a city's traffic, a lake's ecosystem, etc. The state and the conditional probability distribution of state transition are the two components that matter most for a stochastic process.Similar to how a particle moving through water may be described by its initial location and the conditional probability distribution of its position related to its present position at a later time.

DOI: 10.54254/2755-2721/73/20240386

If an external interference factor—that is, the agent's action—is added to a random process of self-evolution like the environment, the probability distribution of the state of the environment at the next instant will be jointly determined by the present state and the agent's action. The formula indicates using the most simple case.

Next state~P(current state. agent's action)

Tthe trial-and-error learning model of reinforcement learning focuses on learning as the agent interacts with the environment. The paper needs to comprehend the multi-armed bandit issue, which can be thought of as a condensed version of the reinforcement learning problem, before people can properly learn reinforcement learning. The most basic example of "learning in interaction with the environment" is the multi-armed slot machine, which, unlike reinforcement learning, just has actions and rewards. Understanding it can help us learn reinforcement learning. The dilemma of exploration vs. exploitation in multi-armed bandits has always been a particularly classic topic.

There is a multi-armed bandit machine with K levers in the issue. Each lever's pull corresponds to a R about the reward's probability distribution. Every time people pull one of the levers, we can receive a reward (r) from the lever-specific reward probability distribution. When the reward probability distribution for each lever is unknown, the author starts over with the intention of achieving the highest cumulative reward achievable after operating T levers. The author must choose between "exploring the winning probability of the levers" and "selecting the most awarded levers based on experience" because the probability distribution of rewards is unknown. The multi-armed slot machine problem is "What type of operation strategy can be used to obtain the highest cumulative reward." What would you do if it were you?

A tuple can be used to represent the multi-armed bandit issue, where A represents a series of actions, one of which is pushing a lever. The set a1,...,ax is what use at A to indicate any action if the multi-armed slot machine has K total levers;

The action of pulling each lever correlates to a reward probability distribution R(ra), and the reward distribution of several levers is often different. R is the reward probability distribution.

The objective of the multi-armed bandit machine is to maximize the reward accrued over a period of time step T: max Zirt, rt R(lat), presuming that only one lever may be pulled in each time step. At signifies pushing a lever among them.

The objective of the multi-armed bandit machine is to maximize the reward accrued over a period of time step T: max Zirt, rt R(lat), presuming that only one lever may be pulled in each time step. At the seventh time step, at symbolizes the action of pulling a lever, and rt represents the reward at received.

For each action a, the paper defines its expected reward as Q(a) = Er~R(la)[r]. Therefore, there is at least one pull rod whose expected reward is not less than pulling any other pull rod. We express the optimal expected reward as Q*= maxaeAQ(a). In order to more intuitively and conveniently observe the difference between the expected reward of pulling a lever and the expected reward of the optimal lever, the paper introduces the concept of regret. Regret is defined as the difference between the action a of pulling the current lever and the expected reward of the optimal lever, that is, R(a) = Q* - Q(a). Cumulative regret is the total amount of regret accumulated after operating the lever T times. For a complete T-step decision {a1, a2,...,aT}, the cumulative regret is or = $\sum$R(at). The goal of the MAB problem is to maximize the cumulative reward, which is equivalent to minimizing the cumulative regret.

In order to know which lever will get a higher reward by pulling it, the paper needs to estimate the expected reward of pulling this lever. Due to the randomness of the reward obtained by pulling the lever only once, it is necessary to pull a lever multiple times, and then calculate the expectation of the multiple rewards obtained.

The algorithm flow is as follows.

For $\forall a \in A$, initialize the counter N(a) = 0 and the expected reward estimate Q(a) = 0

Select a pull rod and this action is recorded as at get rewarded rt

DOI: 10.54254/2755-2721/73/20240386

$$\text{Update counter: } N(at) = N(at) + 1 \tag{1}$$

Update the expected reward estimate: $Q(at) = Q(at) + N(a_1)\,[re - Q(a_1)]$

The fourth step in the for loop above updates the valuation in this way because it allows for incremental expectation updates. The formula is as follows.

$$Qk = \frac{1}{k}\sum_{i=1}^{k} r\,i = \frac{1}{k}(rk + \sum_{i=1}^{k-1} r\,i) = Qk - 1 + \frac{1}{k}[rk - Qk - 1] \tag{2}$$

In the trial-and-error approach to reinforcement learning, exploration and utilization are crucial technologies and critical challenges in interactive learning with the environment. The finest setting for understanding the theory of technological exploration and application is the multi-armed bandit dilemma. It will be highly beneficial for us to understand about the exploration of reinforcement learning settings if can comprehend the exploration and utilization problems of multi-armed bandits. Students who are interested in learning more about the theoretical analysis of cumulative regret of different algorithms for multi-armed bandits can do it on their own. -In multi-armed bandit issues, the greedy method, the higher confidence bound algorithm, and the Thompson sampling algorithm are all often employed.Among them, the upper confidence bound algorithm and the Thompson sampling method can guarantee the asymptotically optimal cumulative regret of the logarithm.

The multi-armed bandit issue and reinforcement learning vary significantly in that the former's interaction with the environment does not alter the latter. That is, stateless reinforcement learning may be used to describe the multi-armed bandit machine since the outcome of each encounter has nothing to do with past actions.

3. Deep Reinforcement learning

To hold the Q values of all activities in each state, the study created a table in the shape of a matrix. The expected return that may be attained by selecting action an in state s and continuing to employ a certain strategy is represented by each action value Q(s,a) in the table[5]. This way of employing tables to record action values, however, is only appropriate when the environment's state and actions are discrete and the available space is modest. This is the case in a number of settings where we have really engaged in code battle (like cliff walking). When there are a huge number of states or actions, this strategy is not appropriate. For example, when the state is an RGB image, assuming the image size is 210×160×3, there are a total of 256 (210×60×3) states. It is unrealistic to store a Q-value table of this order of magnitude in the computer. . What's more, when states or actions are continuous, there are infinite state-action pairs, and cannot use this table form to record the Q value of each state-action pair. For this situation, the paper needs to use the function fitting method to estimate the Q value, that is, treat this complex Q value table as data, and use a parameterized function Q to fit these data. Obviously, this method of function fitting has a certain loss of precision, so it is called an approximate method. The DQN algorithm are going to introduce today can be used to solve the problem of discrete actions in a continuous state [6].

Now the paper wants to get the action value function Q(s,a) in an environment similar to the car pole. Since the value of each dimension of the state is continuous, it cannot be recorded in a table, so a common solution is to use the function fitting (function approximation) idea. Since the neural network has a powerful expressive ability, a neural network can be used to represent the function Q. If the action is continuous (infinite), the input of the neural network is the state s and the action a, and then outputs a scalar, indicating the value that can be obtained by taking an action in the state. If the action is discrete (finite), in addition to the continuous action, the paper can also only input the state s into the neural network, so that it can output the Q value of each action at the same time. Usually, DQN (and Q-learning) can only handle the case of discrete actions because there is max in the update process of function Q. This operation. Assume that the parameters used by the neural network to fit the function are, that is, the Q values of all possible actions in each state s as Qw(sa). We call the neural network used to fit the function Q function Q network [7-8].

DOI: 10.54254/2755-2721/73/20240386

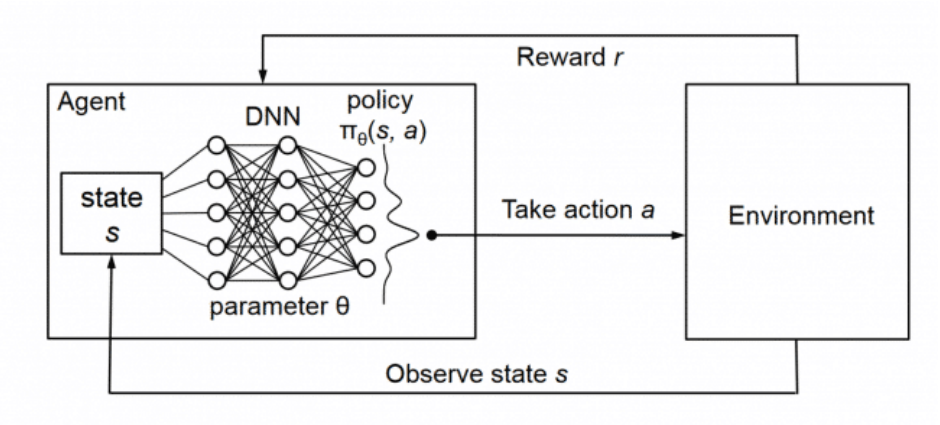

Figure 2. DQN networks function

Of course, there are many classic algorithms in reinforcement learning, such as the DQN algorithm. The main idea is to use a neural network to represent the function of the optimal policy, and then use the idea of Q-learning to update parameters. In order to ensure the stability and efficiency of training, the DQN algorithm introduces two modules: experience playback and target network, so that the algorithm can achieve better results in practical applications [9]. At the 2013 NIPS Deep Learning Symposium, the DeepMind research team published the DQN paper, demonstrating for the first time this reinforcement learning that directly accepts pixel input through a convolutional neural network to play various Atari games. algorithm, thus kicking off the field of deep reinforcement learning. DQN is the basis of deep reinforcement learning. Only by mastering this algorithm can you truly enter the field of deep reinforcement learning.

4. Discussion

Science's understanding of reinforcement learning has advanced quite quickly in recent years. Reinforcement learning, computer vision, intelligent voice, natural language processing, data mining, and information retrieval are topics covered by more than one-fifth of the papers presented at the major academic conferences on machine learning and artificial intelligence in general. Reinforcement learning is being used to address important decision optimization issues in computer networks, computer graphics, and other domains in an increasing number of academic conferences and journal research articles. Some domestic and foreign start-up companies with reinforcement learning as their core technology have started to emerge in the industry as more and more businesses start using reinforcement learning technology in actual business to make their decision-making systems more and more intelligent. The most recent advancements in scientific research and the commercial use of reinforcement learning technology almost every day from a variety of sources, and many of the findings are impressive.

5. Conclusion

No matter how advanced the technology, it must be used to benefit people in order to be truly valuable. The development of reinforcement learning technology has as its overarching objective its efficient application to a variety of decision-making activities. based on a limited understanding Because reinforcement learning is interactive, there is no assurance that the strategy or value function can be improved from the interactive data. As a result, reinforcement learning algorithms—especially deep reinforcement learning—always struggle with low sample efficiency. The paper has reviewed the most often used techniques to increase the effectiveness of reinforcement learning samples from a variety of angles in the third chapter of this book, including imitation learning, model-based policy optimization, goal-oriented reinforcement learning,etc.These techniques represent the state-of-the-art in reinforcement learning research, yet each has significant drawbacks. The paper has assumed that the

DOI: 10.54254/2755-2721/73/20240386

sample efficiency of reinforcement learning algorithms will advance in the coming years of study, and that eventually the demands on computer resources and data sampling will be significantly decreased.

References

[1] AUER P, CESA-BIANCHI N,FISCHER P. Finite-time analysis of the multiarmed bandit problem[J]. Machine learning,(2002), 47 (2) : 235-256.
[2] AUER P. Using confidence bounds for exploitation-exploration trade-offs[J]. Journal of Machine Learning Research,(2002), 3(3): 397-422.
[3] GITTINS J, GLAZEBROOK K,WEBER R. Multi-armed bandit allocation indices[M]. 2nd ed. America: John Wiley & Sons, (2011).
[4] CHAPELLE O, LI L. An empirical evaluation of thompson sampling [J]. Advances in neural information processing systems, (2011), 24: 2249-2257.
[5] VOLODYMYR M, KAVUKCUOGLU K, SILVER D, et al. Human-level control through deep reinforcement learning [J]. Nature, (2015), 518(7540): 529-533.
[6] VOLODYMYR M, KAVUKCUOGLU K, SILVER D, et al. Playing atari with deep reinforcement learning [C]//NIPS Deep Learning Workshop, 2013.
[7] HASSELT V H, GUEZ A, SILVER D. Deep reinforcement learning with double q-learning [C]// Proceedings of the AAAI conference on artificial intelligence. 2016, 30(1).
[8] WANG Z, SCHAUL T, HESSEL M, et al. Dueling network architectures for deep reinforcement learning [C]// International conference on machine learning, PMLR, 2016: 1995-2003.
[9] HESSEL M, MODAYIL J, HASSELT V H, et al. Rainbow: Combining improvements in deep reinforcement learning [C]// Thirty-second AAAI conference on artificial intelligence, 2018.