International Journal of Research Studies in Science, Engineering and Technology
Volume 7, Issue 12, 2020, PP 52-62
ISSN 2349-476X
DOI: https://doi.org/10.22259/2349-476X.0712006

A Succinct Overview to Reinforcement Learning

Sirisha Maddigapu[1], Akhila Maguluri[1], Dr. Pathan Basha[2]

[1]CSE Department, Malineni Lakshmaiah Women”s Engineering College, Guntur, AP

[2]Associate. Professor, CSE Department, Malineni Lakshmaiah Women”s Engineering College, Guntur, AP

ABSTRACT

This study will introduce, review, and summarise many Reinforcement Learning works and research publications. Artificial Intelligence's Reinforcement Learning is a branch of the field has proven to be a useful technique for artificially constructing structures. addressing sequential decision making with intelligent systems issues. Reinforcement Learning has had a lot of success. It has made significant advancements in recent years and has been able to In many domains, it outperforms humans; it can play and win. a variety of games Reinforcement learning has a long history of being effective. in the solution of several control system issues It now has a resurgence. The number of applications is increasing. In addition to the core concepts, this book contains an introduction to reinforcement learning that explains the intuition behind Reinforcement Learning. Then there's the amazing The advantages of reinforcement learning are emphasized. As a result, methods and procedures for tackling reinforcement problems have emerged. The issues with learning are summarized. After that, data from a a large number of studies in the field of reinforcement learning The applications were scrutinized. Finally, there are the prospects and the opportunities. The advantages and disadvantages of reinforcement learning are examined.

Keywords: *Reinforcement Learning, Artificial Intelligence, Machine Learning*

INTRODUCTION

In recent years, Artificial Intelligence (AI) has become a trendy issue. Many articles, books, and films have been written in response to the issue "can machines think?" "Can artificial intelligence surpass human intelligence?" and "Will Machines Replace Humans?" "How harmful is AI?" "How does AI differ from humans?" in addition to the enslavement problem [1], from AI?" These issues are not unnoticed by researchers and scientific groups. Some of these questions were discussed by Alan M. Turing. [2] As a result, the Turing test was created to evaluate a machine's intelligence. ability to behave intelligently in a way that is indistinguishable from [3] that of a human Some of these questions, however, remain unanswered. a point of contention among the most powerful CEOs (i.e. Mark Zuckerberg) Facebook's Mark Zuckerberg and Elon Musk, as well as the world's best AI experts) researchers. Discussing such issues necessitates a thorough examination. appreciation for reinforcement learning (RL). I recommend to the reader [1] is a reference to Stuart J. Russell's work.

Deep Learning advances in recent years have been linked to the rise of Artificial Intelligence. Deep Learning is a collection of neural networks with several layers. they are linked to one another Deep learning algorithms, on the other hand, are Deep learning is similar to what was employed in the late 1980s [4]. The advancement of computational power is the driving force behind progress. as well as the massive rise in both generated and gathered data Shifting from CPU (Central Processing Unit) to GPU (Graphical Processing Unit) data [5]. (Graphics Processing Unit) [6], and then to TPU (Tensor Processing Unit). Processing Unit) [7] increased processing speed and opened up new possibilities. More successes are on the way. Computing, on the other hand, Moore's law [8] limits capabilities, which may cause a slowdown. down constructing powerful AI systems [9]. Learning through interaction with reinforcement is referred to as reinforcement learning. a setting through engaging in a variety of activities and encountering new things many accomplishments and mistakes while attempting to maximise the received monetary compensation The agent isn't told which action he or she should take.

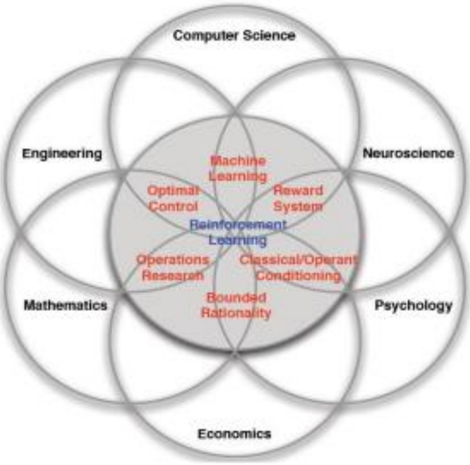

Fig. 1 Reinforcement Learning faces [10]

Reinforcement Computer science, engineering, neurology, mathematics, psychology, and economics are just a few of the domains where learning intersects. These are shown in Figure 1.intersections (n.d.) (n.d.) (n.d.) (Reinforcement learning is distinct from other types of learning. Both supervised and unsupervised learning are possible with machine learning. The most well-known and researched method of learning is supervised learning. Machine learning is a well-studied branch of the field. When it comes to guided learning,The computer learns from a set of labelled data that it has been given as a training set.offered by an outside teacher or supervisor who makes the final decision the appropriate actions for each that the system should take example. The objective of the system is to generalise its reactions in order to act. correctly in circumstances where the training isn't provided examples. The supervised learning system's performance By increasing the number of training samples, it improves. The following are some examples of supervised learning issues: regression, object detection, image captioning, categorizationas well as labelling Although this form of education is critical, it is also time consuming. insufficient for interactive. The goal of unsupervised learning is to uncover organisation in a set of unlabeled data. Clustering, feature learning, and unsupervised learning are some instances of unsupervised learning. Density estimation and dimensionality reduction Regardless, It may appear that reinforcement learning is a form of unsupervised learning. It is distinct in that it does not learn from labelled data; Instead of maximising the rewards, reinforcement learning tries to maximise the rewards. than the discovery of hidden structure [9]. Reinforcement learning is the third paradigm of learning. Along with unsupervised learning and machine learning, learning under supervision However, other options are available. [9] paradigms.

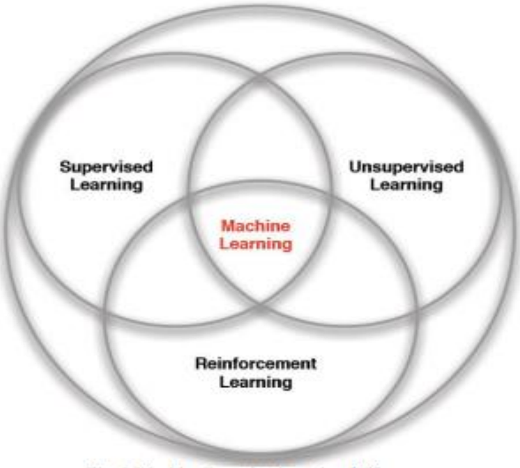

Fig. 2 Machine Learning branches [10]

"A Concise Introduction to Reinforcement Learning," by A. Hammoudeh, 2018.

The standard model and basic components of a reinforcement learning system will be discussed in the following sections. In the following section, we'll look at some amazing reinforcement learning examples. The accomplishments are emphasised. Then came the concept of Markov Decisions. The equations of process and Bellman optimality, which express the The following elements are essential for developing reinforcement learning problems: discussed. Following that, certain problem-solving algorithms are shown. The problem of reinforcement is examined, beginning with the general case. Approximate methods and traditional tabular methods are two examples of approaches. solution approaches, next the Monte Carlo method, and finally the Starting with the Temporal Difference approach and finishing with a policy-based method approaches, as well as the deep Q-network method. Eventually, some Applications for reinforcement learning are highlighted.

A. The Basic Model

The most important elements of a reinforcement learning system are: [9], [10] are policy, reward signal, value function, and model. The policy () describes how the agent (something) acts. that sees and responds to its surroundings [1]) will behave in specific circumstances Simply

$$V(s) = \mathbb{E}(G_t \mid S_t = s) \qquad (1)$$

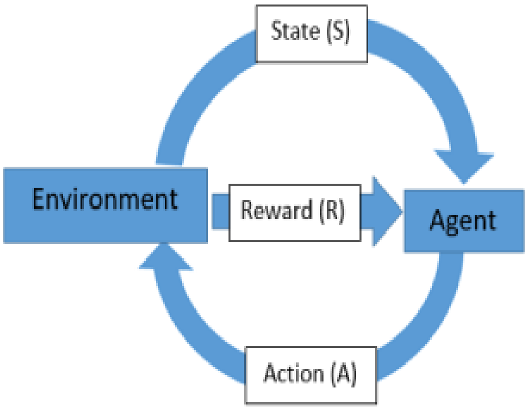

Fig. 3 Reinforcement Learning standard diagram

Fig. 4 Screen shots from five Atari 2600 Games: (Left-to-right) Pong, Breakout, Space Invaders, Seaquest, Beam Rider [11]

The total rewards R from time-step t are equal to the sum of the immediate reward and discounted future reward [9], [10].

$$G_t = R_{t+1} + \gamma R_{t+2} + \gamma^2 R_{t+3} + \cdots$$
$$= \sum_{m=0}^{\infty} \gamma^m R_{t+m+1} \qquad (2)$$

The price reduction varies between 0 and 1 and represents the deterioration factor of future rewards when they are evaluated now. Discounting is, nevertheless, occasionally necessary. controversial. The expected return is represented by the action-value function q(s,a).vwhen beginning from a state s and performing an action a [9], [10] shows the mathematical formulation.

$$q(s,a) = \mathbb{E}(G_t \mid S_t = s, A_t = a) \qquad (3)$$
$$= \mathbb{E}\left(R_{t+1} + \sum_{m=1}^{\infty} \gamma^m R_{t+m+1} \mid S_t = s, A_t = a\right)$$

Predictions regarding the behaviour of the environment can be made using the environment model. Model, on the other hand, is an optional component of reinforcement learning. Model-based methods are those that make use of models and planning. On the other hand, there are ways that do not require a model. if the agent doesn't have a model for the situation model-free approaches, on the other hand, are expressly trial-and-error learners [9], [10].

"A Concise Introduction to Reinforcement Learning," by A. Hammoudeh, 2018.

Implementation of Outstanding Successes

Reinforcement learning is a concept that has been around for a long time. The rise, on the other hand, The most achievements and the most reinforcement learning are recent. Deepmind outperformed humans in Atari games by a factor of ten. It is a blend of reinforcement and deep learning.

Without tuning to a specific game, the same algorithm was used to learn to play 49 various games from self-play and the score of the game. Raw screen pixels were mapped to a Deep Q. Network. Deep Mind revealed the details of a programme that can predict the future. In 2013, you can play Atari at a professional level [11]. Later in the game, Google bought DeepMind in the beginning of 2014 [12].

IBM's Gerald Tesauro created a programme that plays Reinforcement learning is used in backgammon. Gerald's show is on. was able to outperform human opponents [13]. However, scaling this up is a challenge. It was tough to achieve success in more difficult games until recently. Deep Mind's AlphaGo software was first trained in 2016 utilising Go's universe was defeated thanks to reinforcement learning.

Lee Sedol, champion Go is a traditional Chinese board game. It was significantly more difficult for computers to master [14]. One of the more notable accomplishments in the field of is AlphaZero is a reinforcement learning system that has reached a high level of success. in many ways than one superhuman level

Fig. 5 Reinforcement learning was implemented to make strategic decisions in Jeopardy! (IBM's Watson 2011) [18]

Markov Decision Process (Mdp)

The Markov Decision Process problem entails making a series of decisions in which an action (A) must be done. each state (S) that the agent travels through. Decision made by Markov A process can be described as a series of states, actions, and outcomes. as indicated in the next line [10]

$$\ldots\ldots S_t, A_t, R_t, S_{t+1}, A_{t+1}, R_{t+1}, S_{t+2}, A_{t+2}, R_{t+2}\ldots\ldots\ldots$$

The primary property of the Markov process is that given the present, the future is independent of the past. Markov state is defined by Equation 4 as the probability of the following state. based solely on the present situation, independent of the past [10],,,,,,,,,,

$$Prob(S_{t+1}|S_t) = Prob(S_{t+1}|\, S_t, S_{t-1}, \ldots\ldots S_2, S_1) \quad (4)$$

A. Bellman optimality equation

MDPs have been extensively researched in control theory, and their origins may be traced back to Richard Bellman's pioneering work. Bellman's The key contribution was to demonstrate that MDP may be solved utilizing Dynamic programming (DP) is a technique for reducing computing time.[19], [20], [21], [22], [23], [24], [25], [26], [An agent's goal is to take measures that optimise the return on investment. This is essentially an optimality issue in which the overall rewards are maximised. The reinforcement learning agent strives to take activities that result in positive reinforcement. maximum benefits that the action can represent (s,) value function [9], [10]The maximum action value function is the optimal action value function. over all policies, an action-value function

$$q_*(s,a) = \max_\pi (q_\pi(s,a)) \qquad (5)$$

The recursive form of the action value function q(s,) which may be recast in the recursive form as follows is used in the

Additional Learning Algorithms

It is possible to find the best policy for reinforcement learning. whether to use exact or approximate solution approaches (Approximation of a function)

Bellman optimality equation (equation 7). (equation 6) [9] and [10]

$$q(s,a) = R(s,a) + \gamma\, q(s',a') \qquad (6)$$

Bellman optimality equation:

$$q_*(s,a) = R(s,a) + \gamma \sum\nolimits_{s'} T(s,a,s') \max\nolimits_{a'} (q_\pi(s',a')) \qquad (7)$$

Where:

R(s,ɑ) is the immediate reward given the current state s and the current action a

"A Concise Introduction to Reinforcement Learning," by A. Hammoudeh, 2018.

$$R(s,a) = \mathbb{E}(R_{t+1} \,|\, S_t = s, A_t = a) \qquad (8)$$

is the action value function given the state (next possible state) and the action (next possible action) is the probability of a transition to state given the current state s and the current action a

$$T(s,a,s') = \mathbb{E}(S_{t+1} = s' \,|\, S_t = s, A_t = a) \qquad (9)$$

When it comes to large-scale and sophisticated MDPs, traditional DP approaches such as policy iteration and value iteration may fail; there are two barriers that may prevent MDPs from succeeding.1) The curse of modelling, which is the difficulty of scaling. 2) the curse of computing transition probabilities. When altering the elements of a scene, the dimensionality of the scene increases. MDP gets difficult. Adaptive functions, on the other hand Approximations [23] and learning-based approaches [24] have a good track record. in the search for a near-optimal solution for large-scale MDPs

A Tabular techniques

Q-learning is a straightforward reinforcement learning method. that, in the absence of any model, learns long-term optimal behavior [25] the environment The Q-learning algorithm's concept is as follows: that the present value of our Q estimate can help us increase our performance (bootstrapping) approximated solution [21]

$$\Delta Q(S_t,A_t) = \alpha\delta$$
$$= \alpha\left(R_{t+1} + \gamma \max_a (Q(S_{t+1},a) - Q(S_t,A_t)\right) \quad (10)$$

Where is the learning rate and what is the difference in time? (TD) error Q-learning is non-policy; it assesses only one policy (the target). policy) while adhering to a different policy (behavior policy). Finding an ideal action value function q* under these conditions, on the other hand, is difficult. Policy can be used to establish arbitrary behaviour policy iteration. Q eventually converges on the best action value function and its opportunistic policy converges on an optimal approach.

Iteration of policy necessitates policy improvement and review. The latter improves the action value function estimate by reducing temporal difference errors (TD) in the action value function estimate. By adhering to the policy, you can get experience on different paths. According to the estimation As the policy evolves, it is usually possible to improve it by selecting better options. According to the most recent value function, greedy actions Q. What if instead of separating the preceding steps (as in the old method) By allowing for policy iteration, the process can be sped up. As in extended policy iteration [26], interleaved steps are used.

B. Methods of approximate solution

Exact value functions and exact value functions are represented by tabular approaches. Tables of policies Because of the size and complexity of the project, As the complexity of the environment grows, so does the amount of computing power required. Increases rapidly. Approximations, on the other hand, are a type of approximation. a powerful notion that incorporates the problem of hidden states and The issue of scalability [9]. A parametrized action value function is a type of action value function. a function approximator with the parameter A linear weighting of features or a deep approximator can be used. Neural

$$\Delta\theta_t = \alpha\left(R_{t+1} + \gamma\, \hat{q}(S_{t+1},A_{t+1},\theta_t) - \hat{q}(S_t,A_t,\theta_t)\right)\frac{\partial \hat{q}(S_t,A_t,\theta_t)}{\partial \theta_t} \quad (11)$$

SARSA is a method that analyses and follows a single policy, with the behaviour policy being the same as the target policy. The SARSA algorithm is a method of approximation. As a result, the policy should not be completely avaricious; instead, it should be balanced. when the policy is most greedy () of the time, but random selection has a low likelihood. [22] Take actions On-policy methods outperform off-policy methods in general, but they uncover less effective policies. A method that is not in compliance with the rules is used to locate Although there is a superior policy (goal policy), it is not followed. The situation has deteriorated [9]. Given linear function on-policy approaches Many published works reported that the approximator was accurate. [27]–[29] and [27]–[29] assured convergence for prediction issues Control issues with no divergence [30]. Off-policy can benefit from function approximation. Q-learning and a semi-gradient Q-learning update are examples of such approaches. The approximate function's parameters are displayed in 12 [25] equation.

$$\Delta\theta_t = \alpha\left(R_{t+1} + \gamma \max_a \hat{q}(S_{t+1},a,\theta_t) - \hat{q}(S_t,A_t,\theta_t)\right)\frac{\partial \hat{q}(S_t,A_t,\theta_t)}{\partial \theta_t} \quad (12)$$

Hammoudeh, “A Concise Introduction to Reinforcement Learning,” 2018

However, Q-learning with function approximation may experience instability; the main source of instability is not learning or sampling, because dynamic learning and sampling are not possible. Divergence with function is a problem in programming. Approximation; neither greed, exploration, nor control are possible is the primary reason, because policy evaluation cannot deliver results on its own. instabilities, as well as the function's complexity Since linear functions are the fundamental cause, approximation is not the root cause. It is possible for approximation to diverge [9]. When using function approximation, bootstrapping, and other techniques, There is a risk of instability and divergence off-policy [28]. Choosing two out of three, on the other hand, is difficult; For generalisation and scalability, approximation is crucial. For both computational and data purposes, bootstrapping is critical. Off-policy learning aids in identifying a better solution. policy by separating behaviour and target policies [9]. Attempts to achieve stability and survive the crisis have been reported. The fatal trinity (combining function approximation, combining function approximation, and combining function approximation Experience Replay appears to be bootstrapping and off-policy. [31], as well as more stable targets such as Double Q-learning [32],

Temporal-Difference Learning Method and Monte Carlo Method

Both the Monte Carlo Method and the Temporal-Difference Method are used in this study. With no prior knowledge of MDP, the learning methods are model-free. rewards or transitions They gain knowledge straight through the episodes of The MC return can be estimated by averaging experience. the aftereffects of several rollouts Monte, on the other hand, Carlo, who learns from full episodes through sampling; Temporal-Difference is a machine learning algorithm that learns from incomplete episodes. bootstrapping and sampling It is, however, feasible to obtain the as in the finest of both Monte Carlo approach and TD learning Interpolates between Temporal Difference (=0) and MC (=1) [0] using the TD() technique.

D. RL depending on policy

The search for an effective policy-based reinforcement learning method As a result, it is an optimal policy with no value function. It has proven to be useful in continuous and high-dimensional spaces. Although it has stronger convergence properties, it usually converges to a optimum in the immediate vicinity

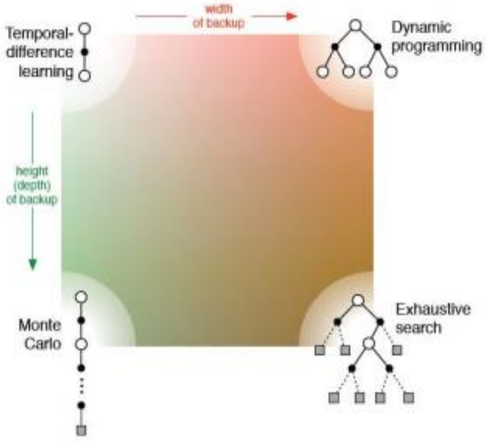

Fig. 6 A slice through the space of reinforcement learning methods, showing the most important dimensions. At the extremes of these two dimensions are: dynamic programming, exhaustive search, TD learning and Monte Carlo [9]

Both gradient-based [45]–[48] and gradient-free [49]–[51] approaches are used to successfully train neural networks to estimate policies. While gradient based approaches dominate, gradient free methods perform well with a small number of parameters and can optimise non differentiable policies. Algorithms for deep reinforcement learning [52] Actor-critic approaches are used to iterate generalised policies. alternating between policy improvement and policy implementation evaluation. Because it selects, a policy is referred to as an actor. A critic is a function that estimates the value of something. because it raises questions about the actor's conduct Actorcritic approaches trade off policy gradient variance reduction for variance reduction. using value function approaches to introduce bias [53]–[55].[48], [49], [50], [51], [52], [53], [54], [55], [56], [57], [58], [59], [[56], [57]

E. DQN

The Deep Q-Network algorithm combines Reinforcement Learning with deep neural network training. DQN was an acronym for illustrated to work straight from unprocessed visual inputs and on a computer a wide range of contexts and was successful in achieving Atari 2600 games at a superhuman level [11], [58]. DQN solves the core problem of instability in DQN. combining approximation of functions and reinforcement learning by employing two techniques: experience replay [59] and target-based learning [60] networks Both the experience replay and the target networks are important. have been utilised in deep reinforcement learning in the future[61]–[63] works.

The reinforcement learning agent can train on and sample cyclic transitions of the type (current state, current action, next state, reward) stored in the experience replay memory. from past experiences with others This effective by learning with it numerous times, you can make use of previous experience. When teaching a dog, this often results in improved converging behaviour.A. Hammoudeh,

In addition to a large reduction in the number of contacts required with the environment, the function approximat or reduces the variance of learning updates [64], [65].While Deep Mind's original DQN algorithm employed standardised sampling [58], Deep mind reported a similar result in a later paper. A more effective learning system that prioritises samples depending on their importance[64] on TD mistakes Despite the fact that experience replay is commonly used,It is a simple model that is known as model-free technique[65].The target network is the second stabilising strategy. By freezing the Q-network and the target, the relationships between them are broken instead of updating the policy network for a period of time. The TD error is based on a shaky approximation of the Q-values. The fixed target network is used by the policy network[60].MDP presume that the agent has complete knowledge of the current situation. This is a non-realistic state. Nonetheless, Partially Observable is a term that is used to describe something that is only partially visible. The Markov Decision Process (POMDP) posits that the agent makes decisions based on a set of probabilities.

APPLICATIONS

Electric power systems, healthcare, banking, robotics, and other sectors have all used reinforcement learning. marketing, natural language processing, and transportation systems are just a few examples. as well as games Games provide an excellent reinforcement environment. Because the agent can investigate different trials in a single session, it is a learning agent. Since the cost of exploration is low, the virtual world is a viable option [71]. In games, there have been some noteworthy results with reinforcement learning. Section II delves into these topics. [72]–[73]–[74]–[75]–[76]–[77]–[[75], as well as a survey of video reinforcement learning [76] games More applications can be found in the sub-sections below. are being debated.

Neural Network Hyper parameter Selection

Identifying the architecture of neural networks and The selection of hyperparameters is an iterative process. Of experimentation and evaluation As a result, it can be expressed as a The issue of reinforcement learning. Zoph created a recurrent neural network RNN (recurrent neural network) that The RNN creates hyperparameters for neural networks. By searching in, he was trained with reinforcement learning.

A. Hammoudeh, “A Concise Introduction to Reinforcement Learning,” 2018.

Results were compared to those obtained using state-of-the-art technologies [77]. Later, the approach was expanded to find optimization methods for deep neural networks [78], with greater results than before. Stochastic Gradient, for example, is a standard optimization strategy. SGD (Stochastic Gradient Descent) [4], Stochastic Gradient Descent with Root Mean Square Propagation [4], Momentum (RMSProp) [79] and Adam's adaptive moment estimation (80].

Intelligent Transportation Systems

Intelligent Transportation Systems make use of the most up-to-date technology. Information technology for traffic management and facilitation Networks of transportation [81] ATSC (adaptive traffic signal control) can help reduce traffic congestion. congestion by altering signal timing plans in real time in reaction to changes in traffic Reinforcement using many agents To tackle ATSC, a learning strategy was developed in [82]. a problem in which each controller (agent) is in charge of the Control of traffic lights in the vicinity of a single traffic intersection. Multiagent reinforcement learning integrates game theory and reinforcement learning. Reinforcement learning with a single agent Multi-agent systems confront a number of issues. The exploration exploitation tradeoff, curse, and reinforcement learning are three approaches to reinforcement learning.

Natural Language Processing

Reinforcement of Natural Language Processing Learning was used in a variety of natural processing activities, including text production [85], [86], and machine learning. [87]–[89], as well as conversational systems. within the Dialogue systems are covered in the subsections that follow.

1) Systems for Dialogue

Programs that communicate with natural environments are known as dialogue systems language. In general, they are divided into two groups: Task-oriented conversational agents and chatbots the task-focused Dialog agents converse in short bursts in a specific environment. domain to aid in the completion of specific activities The, on the other hand, Chatbots are made to manage standard discussions replicating human-to-human interactions [90]. The evolution of dialogue systems has been classified. In terms of generations, the first is a template-based system. The second is based on guidelines created by human specialists generation is a data-driven system that includes a few "light" machines strategies for learning The third generation is currently data-driven and uses deep neural networks. Recently, combining the third generation of reinforcement learning has begun.

Many modules make up a typical task-oriented dialogue system pipeline [92], [93]:

a) Natural language understanding (NLU): NLU is a term that refers to the ability to understand natural language. assigns a semantic representation to the user's responses.

b) the tracker of dialogue state (DST) gathers the turn's user input as well as the dialogue . The current condition of the dialogue is determined by history and the current state of the dialogue is determined by history.

c) the policy of dialogue based on the current status of the discourse, chooses the next action

d) creation of natural language.

The process of constructing natural language is known as natural language generation. Computer systems that generate natural language language messages derived from underlying information representations [94]. Designing dialogue systems can be done in two ways: The modular method and the end-to-end approach are two approaches that can be used. The distinction The difference between them is that the latter technique replaces some of the former. Figure 7 shows independent modules with a single end-to-end connection. [93] model in learning dialogue, reinforcement learning is common.

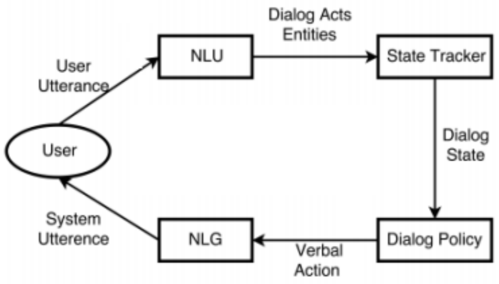

Fig. 7 Typical pipeline of task oriented dialogue systems [93]

The Alexa Prize is a 2.5-million-dollar competition in which university teams compete to construct conservation bots that communicate with people through text and sound [102]. Even though the 2017 competition winner (Sounding Board by University of Michigan) Reinforcement learning was not used by Washington [103]. The highest average user score was MILABOT, with a score of Reinforcement was used 3.15 out of 5 times in the competition. learning. The MILABOT system was created at the University of Montreal.

It's made up of 22 different response models, including Long-term and short-term user happiness must be balanced [104]. A selection of works that used reinforcement learning to solve problems. DeepMind's work on natural language processing includes the use of To calculate representations of natural phenomena, reinforcement learning is used.

Tree-structured neural networks are used to learn language sentences.[105].

Objectives and Challenges

In RL, there are huge uncharted territory as well as numerous unsolved questions in the ones that have been explored; among of the typical questions are: Evaluative feedback, nonstationarity, and delayed rewards are all issues in RL [9]. Multi-task learning is required for general AI [106]. when an agent is capable of doing a wide range of duties rather than focusing on a small number of similar jobs. Multi-task Learning is one of the issues that RL seeks to address. When it comes to driving, processing power is crucial. Research on reinforcement learning has progressed. Furthermore, designing efficient methods, and contemporary parallel processing Throughput is increased as a result of the hardware. For instance, the final AlphaGo had 40 search threads, 48 CPUs, and 8 GPUs. [14] Graphics processing units. However, computational power may be a stumbling block for some tabular approaches, exhaustive search, and other strategies Monte Carlo is a resort town in Monaco.Pure model-free RL, according to Yann LeCun [71], necessitates. It takes a lot of trials to learn anything. While both trial and error are part of the process.

References

Reinforcement learning is a vast subject with a lengthy history, numerous applications, and an excellent theoretical foundation. Distinguished accomplishments, new algorithms, and a large number of unresolved issues issues. More artificial intelligence research is needed. learning, decision-making, and search principles in general in addition to integrating a diverse range of domains knowledge. Reinforcement learning research is a driving factor. compel artificial intelligence ideas to be simpler and less general [9] Intelligence.

Hammoudeh, “A Concise Introduction to Reinforcement Learning,” 2018.

[1] S. Russell and P. Norvig, Artificial Intelligence: A Modern Approach, 3rd ed. Pearson, 2009.

[2] A. M. Turing, “Computing machinery and intelligence,” in Parsing the Turing Test: Philosophical and Methodological Issues in the Quest for the Thinking Computer, 1950, pp.

[3] C. J. C. H. Watkins and P. Dayan, “Q-learning,” Mach. Learn., vol. 8, no. 3–4, pp. 279–292, 1992. [22] G. A. Rummery and M. Niranjan, “On-line Q-learning using Connectionist Systems,” University of Cambridge, 1994.

[4] P. J. Werbos, “Building and Understanding Adaptive Systems: A Statistical/Numerical Approach to Factory Automation and Brain Research,” IEEE Trans. Syst. Man Cybern., vol. 17, no. 1, pp. 7–20, 1987.

[5] A. G. Barto, R. S. Sutton, and C. W. Anderson, “Neuronlike Adaptive Elements That Can Solve Difficult Learning Control Problems,” IEEE Trans. Syst. Man Cybern., vol. SMC-13, no. 5, pp. 834–846, 1983.

[6] C. J. C. H. Watkins, “Learning from delayed rewards,” University of Cambridge, 1989.

[7] K. Arulkumaran, M. P. Deisenroth, M. Brundage, and A. A. Bharath, “A Brief Survey of Deep Reinforcement Learning,” IEEE Signal Process. Mag. Spec. Issue Deep Learn. Image Underst., pp. 1–14, 2017.

[8] P. Dayan, “The Convergence of TD(λ) for General λ,” Mach. Learn., vol. 8, no. 3, pp. 341–362, 1992.

[9] J. N. Tsitsiklis and B. Van Roy, “An analysis of temporal-difference learning with function approximation,” IEEE Trans. Automat. Contr., vol. 42, no. 5, pp. 674–690, 1997.

[10] R. S. Sutton, “Learning to Predict by the Methods of Temporal Differences,” Mach. Learn., vol. 3, no. 1, pp. 9–44, 1988.

[11] G. J. Gordon, “Stable Function Approximation in Dynamic Programming,” Proc. 12th Int. Conf. Mach. Learn., no. January, pp. 261–268, 1995.

[12] L. Lin, “Reinforcement Learning for Robots Using Neural Networks,” Report, C., pp. 1–155, 1993.

[13] H. Van Hasselt, A. C. Group, and C. Wiskunde, “Double Qlearning,” Nips, pp. 1–9, 2010. [33] H. Yu, “Convergence of Least Squares Temporal Difference Methods Under General Conditions,” in International Conference on Machine Learning, 2010, pp. 1207–1214.

[14] A. R. Mahmood and R. S. Sutton, “Off-policy learning based on weighted importance sampling with linear computational complexity,” Proc. Thirty-First Conf. Uncertain. Artif. Intell. {UAI} 2015, July 12-16, 2015, Amsterdam, Netherlands, pp. 552–561, 2015.

[15] H. R. Maei, “Gradient Temporal-Difference Learning Algorithms,” Mach. Learn., 2011. [36] S. J. Bradtke and A. G. Barto, “Linear Least-Squares algorithms for temporal difference learning,” Mach. Learn., vol. 22, no. 1–3, pp. 33–57, 1996.

[16] H. R. Maei and R. S. Sutton, "GQ( ): A general gradient algorithm for temporal-difference prediction learning with eligibility traces," in Proceedings of the 3d Conference on Artificial General Intelligence (AGI-10), 2010.

[17] P. Abbeel and J. Schulman, "Deep Reinforcement Learning through Policy Optimization," in Neural Information Processing Systems, 2016.

[18] M. Minsky, "Steps toward Artificial Intelligence," Proc. IRE, vol. 49, no. 1, pp. 8–30, 1961. [40] J. A. Nelder and R. Mead, "A Simplex Method for Function.

[19] F. Gomez, J. Koutník, and J. Schmidhuber, "Compressed network complexity search," in Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), 2012, vol. 7491 LNCS, no. PART 1, pp. 316–326.

[20] A. Fraser, "Simulation of Genetic Systems by Automatic Digital Computers I. Introduction," Aust. J. Biol. Sci., vol. 10, no. 4, p. 484, 1957.

[21] M. P. Deisenroth, "A Survey on Policy Search for Robotics," Found. Trends Robot., vol. 2, no. 1–2, pp. 1–142, 2011.

[22] T. Salimans, J. Ho, X. Chen, and I. Sutskever, "Evolution Strategies as a Scalable Alternative to Reinforcement Learning," 2017.

[23] R. J. Willia, "Simple Statistical Gradient-Following Algorithms for Connectionist Reinforcement Learning," Mach. Learn., vol. 8, no. 3, pp. 229–256, 1992.

[24] D. Wierstra, A. Förster, J. Peters, and J. Schmidhuber, "Recurrent policy gradients," Log. J. IGPL, vol. 18, no. 5, pp. 620–634, 2009.

[25] J. Schulman, S. Levine, P. Abbeel, M. Jordan, and P. Moritz, "Trust Region Policy Optimization," in International Conference on Machine Learning, 2015.

[26] J. Schulman, P. Moritz, S. Levine, M. Jordan, and P. Abbeel, "HighDimensional Continuous Control using Generalized Advantage Estimation," in International Conference on Learning Representations, 2016.

[27] G. Cuccu, M. Luciw, J. Schmidhuber, and F. Gomez, "Intrinsically motivated neuro evolution for vision-based reinforcement learning," in IEEE International Conference on Development and Learning, ICDL, 2011.

[28] F. Gomez and J. Schmidhuber, "Evolving Modular Fast-Weight Networks for Control," in ICANN, 2005.

[29] J. Koutník, G. Cuccu, J. Schmidhuber, and F. Gomez, "Evolving large-scale neural networks for vision-based reinforcement learning," in Proceeding of the fifteenth annual conference on Genetic and evolutionary computation conference - GECCO '13, 2013, p. 1061.

[30] Y. Lecun, Y. Bengio, and G. Hinton, "Deep learning," Nature, vol. 521, no. 7553. pp. 436–444, 2015.

[31] R. S. Sutton, D. Mcallester, S. Singh, and Y. Mansour, "Policy Gradient Methods for Reinforcement Learning with Function Approximation," Adv. Neural Inf. Process. Syst. 12, pp. 1057–1063, 1999.

[32] J. Peters and S. Schaal, "Natural Actor-Critic," Neurocomputing, vol. 71, no. 7–9, pp. 1180–1190, 2008.

[33] V. R. Konda and J. N. Tsitsiklis, "Actor-Critic Algorithms," Control Optim, vol. 42, no. 4, pp. 1143–1166, 2003.

[34] V. Mnih et al., "Asynchronous methods for deep reinforcement learning," in International Conference of Machine Learning, 2016.

[35] S. Gu, T. Lillicrap, Z. Ghahramani, R. E. Turner, and S. Levine, "QProp: Sample-Efficient Policy Gradient with an Off-Policy Critic," in International Conference on Learning Representations, 2017.

[36] V. Mnih et al., "Human-level control through deep reinforcement learning," Nature, vol. 518, no. 7540, pp. 529–533, 2015.

[37] L. J. Lin, "Self-Improving Reactive Agents Based on Reinforcement Learning, Planning and Teaching," Mach. Learn., vol. 8, no. 3, pp.

Citation: *Sirisha Maddigapu et al, "A Succinct Overview to Reinforcement Learning", International Journal of Research Studies in Science, Engineering and Technology. 2020; 7(12): 52-62. DOI: https://doi.org/ 10.22259/2349-476X.0712006*