A

Seminar Report

on

“Reinforcement Learning and Deep Reinforcement Learning with applications”

Submitted to the

Savitribai Phule Pune University

In partial fulfillment for the award of the Degree of

Bachelor of Engineering

in

Information Technology

by

Aditya Kangune

(33323)

Under the guidance of

Mrs. Usha. A. Jogalekar

Department Of Information Technology

Pune Institute of Computer Technology College of Engineering
Sr. No 27, Pune-Satara Road, Dhankawadi, Pune - 411 043.

A. Y. 2021-2022

CERTIFICATE

This is to certify that the seminar report entitled **"Reinforcement Learning and Deep Reinforcement Learning with applications"** being submitted by **Aditya Kangune (33323)** is a record of bonafide work carried out by him under the supervision and guidance of **Mrs. Usha A. Jogalekar** in partial fulfillment of the requirement for **TE (Information Technology Engineering) – 2019 course** of Savitribai Phule Pune University, Pune in the academic year 2021-22.

Date: 14/11/2021

Place: Pune

Mrs. Usha. A. Jogalekar
Guide

Dr.A.M.Bagade
Head of the Department

Dr. R. Sreemathy
Principal

ACKNOWLEDGEMENT

"As our circle of knowledge expands, so does the circumference of darkness surrounding it."

The seminar has helped me a lot to discover various new things. I extend my gratitude to Pune Institute of Computer Technology for giving me an opportunity to enhance my knowledge through this seminar.

I am extremely grateful to Dr. R. Shreemathy, Principal, PICT, and Dr. Anant Bagade, Head of the Department (Information Technology), for providing all the required resources for the successful completion of my seminar.

My heartfelt gratitude to my seminar guide Mrs. Usha. A. Jogalekar, Department of Information Technology, for her valuable suggestions and guidance in the preparation of the seminar report.

I specially thank my friend Aagaaz for working on this report together with me and constantly being there for helping me with difficulties.

I express my thanks to all the staff members and friends for all the help and coordination extended in bringing out this seminar successfully in time.

I acknowledge with grateful thanks to the authors of the references and other pieces of literature referred to in this seminar.

Aditya Kangune
(33323)

ABSTRACT

Reinforcement learning (RL) is an area of machine learning concerned with how intelligent agents ought to take actions in an environment in order to maximize the notion of cumulative reward. Deep reinforcement learning is the combination of reinforcement learning (RL) and deep learning. DRL has applications in many fields like medicine, robotics, games, etc. RL works on the Markov Decision Process which leads to Q-learning. MDP provides a mechanism to maximize the reward in a given environment. Combining DL and RL leads to the formation of Deep Q-Networks. RL holds great importance in creating AI for games. Few AIs have managed to defeat world champions in a few games like chess. When we dive deeper into RL, we encounter more complex multi-agent environments. The main challenge in multi-agent environments is to understand the interaction between agents. There are certain techniques to deal with such environments including Graph Convolutional Reinforcement Learning. Particular focus is on exploring these concepts and generalizing DRL and exploring its use in more practical applications like personalized recommendations systems, etc.

Keywords: Reinforcement Learning, Deep Reinforcement Learning, Machine Learning, Artificial Intelligence, MDP, Neural Networks.

CONTENTS

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1

INTRODUCTION

1.1 History of RL:

Artificial Intelligence is a concept that has been discovered and tweaked since around the 1940s. Under AI we have Machine Learning where we work with statistics and algorithms. Finally, we have Deep Learning under ML where we deal with deep concepts like image and speech recognition.

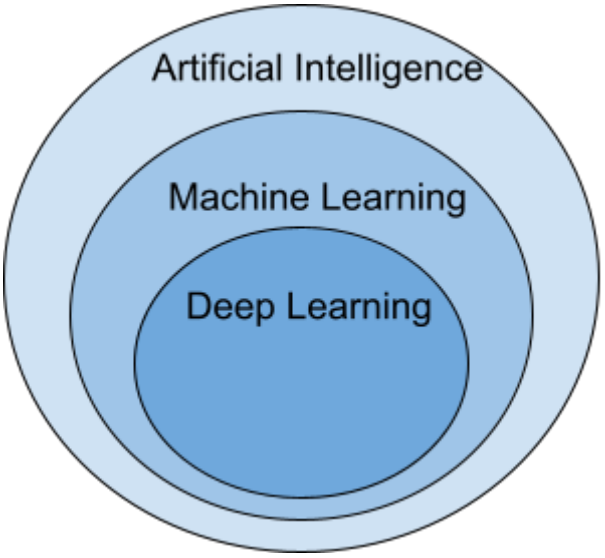


Fig 1- Branches of Automation

There are 3 branches in ML - Supervised Learning, Unsupervised Learning, and Reinforcement Learning.
In RL, the model keeps on developing its performance using Reward and Feedback to learn the behavior and the pattern. These algorithms are particular to a particular problem.

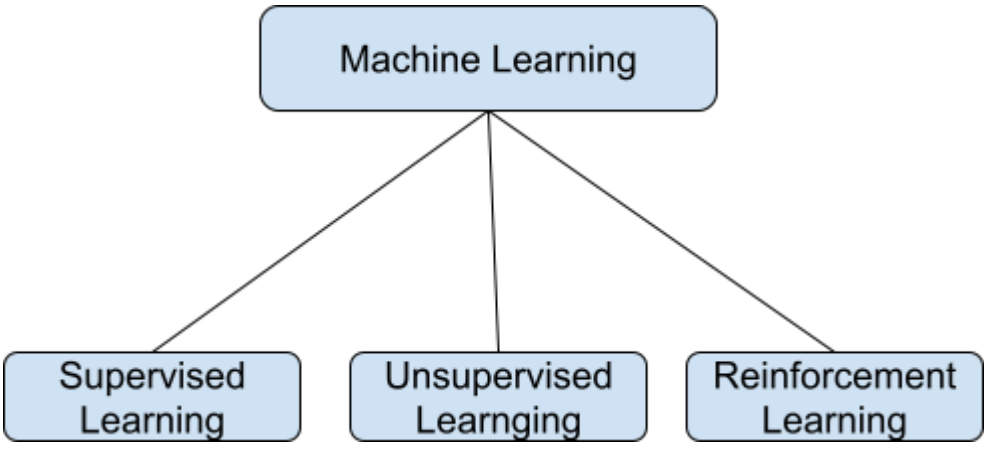


Fig 2 - ML Types

1.2 **Motivation**:

For new age problems in this decade, we need a solution that can create the perfect model to solve that particular problem. That's what the RL aims for.
RL has various interesting applications in the modern world viz medicine, healthcare, self-driving cars, robotics, gaming, marketing, advertising, etc.

1.3 Objective:

Aim:

The aim of this seminar is to understand the complex subject of RL and DRL and simplify them with studying about applications of the same.

Objectives:

- Studying basic concepts in Reinforcement Learning.
- Learning how to integrate Reinforcement Learning with Deep Learning or neural networks.
- Exploring the applications of single-agent environments.
- Exploring the techniques involved in multi-agent environments.

1.4 Introduction to RL and DRL:

Reinforcement Learning is a subfield of machine learning that teaches linked agents how to choose an action/movement from a set of options in a given environment. This could be one way to maximize rewards over time by mentoring.

Deep Reinforcement Learning is what happens when Deep Learning and Reinforcement Learning are coupled and linked. It can be found in a variety of real-world and funny applications in the United States that go unnoticed, such as games, recommendation engines for well-known e-commerce websites, video-streaming platforms, and so on.

This presentation will mostly cover the implementation of DRL in recommendation systems. From Google Maps recommendations to OTT services like Netflix and Amazon, recommendation algorithms are something we all come across on a daily basis.
Self-supervised recommendation systems may be used to increase watch time on video streaming platforms or sales on e-commerce websites. There are other elements to consider, such as the interactions between users and things that form a unit. By focusing on the interaction between the agent, i.e. the user, and the atmosphere (in this case, a website), RL can improve and increase recommendation systems, resulting in an increase in the cumulative and average reward for the agent supporting the interaction with the atmosphere.

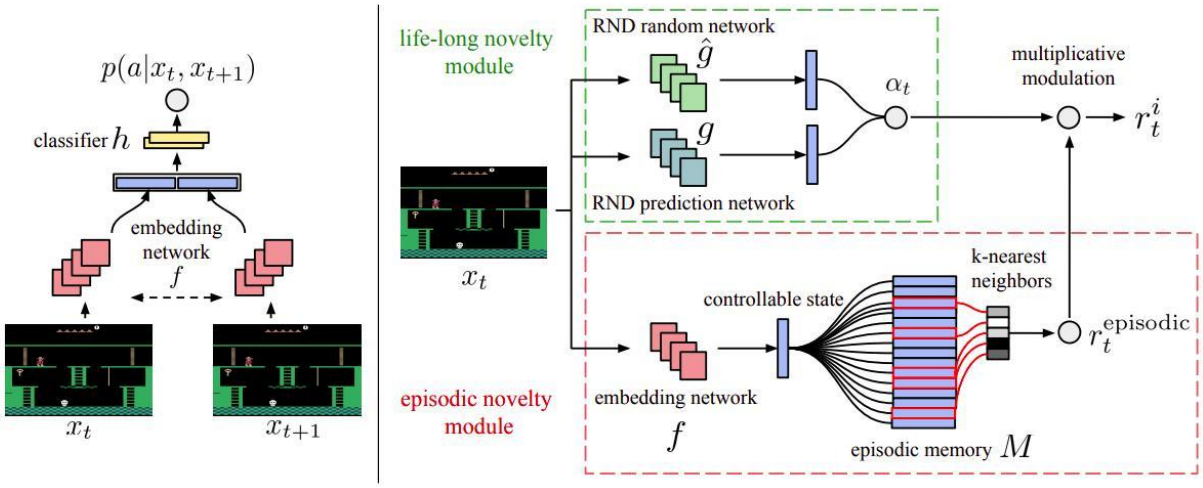
This seminar focuses on the work of reinforcement learning and deep reinforcement learning, as well as their application in two state-of-the-art recommendation frameworks, namely Self-Supervised Q-learning (SQN) and Self-Supervised Actor-Critic Learning (SAC) (SAC).

CHAPTER 2

LITERATURE SURVEY OF RL and DRL

[1]

By understanding directed explorative paths, an RL agent is employed to expose exploration games. To teach the explorative policies, memory-based intrinsic reward is used to maximize the use of knn over fresh experiences. The agent goes over all of the states in the environment once more. The embeddings of the highest neighbor are taught using a self-supervised inverse dynamics model, which biases the new signal towards what the agent can handle. UVFA trains many directed exploration strategies with similar NN in real time. Transfer is no match for major explorative policies leading to effective exploitation since it punishes equivalent NN for n number of degrees of exploration/exploitation. The strategies are implemented using distributed RL agents that collect massive amounts of data from a large number of actors operating in parallel on distinct setting instances. In all, the policy doubles the minimal agent's performance.



[7] Fig 3 - Atari game RL, official - "Never provide Up: Learning Directed Exploration Strategies"

[2]

A CNN is taught to transfer raw pixels from a single front-facing camera onto steering commands. The system learns to drive in traffic on local roads with or without lane markings, as well as on highways, with minimal human intervention. It can be used in areas where visual direction is hazy. Due to the work signal, it automatically learns internal representations of the stated procedure stages such as useful road choices. The model was never explicitly taught to look for things like road definitions. In comparison to particular matter decomposition, this end-to-end approach optimises all procedure steps in real time. Internal pieces may self-optimize to maximise overall system performance, resulting in higher performance. Such criteria are understandably elite for straightforward human interpretation, but they do not ensure the performance of most systems. As a result of the system, smaller networks are possible.

[3]
Virtual assistants are taught with human-annotated lines to learn different language commands. This research describes a toolset for managing novel chemical lines with minimal human intervention. The goal is to translate language into VAPL code using a VAPL and a language programme. A realistic set of input for the brain model is desired by the disembodied spirit. Disembodied spirits use data augmentation, whereas coders use snippets to get information. It is proposed to coach a linguistics programme style principles that build VAPL languages that are adaptable to language translation on the side of the synthesis information. These concepts are being utilised to modify ThingTalk, the language used by the virtual assistant Almond. The disembodied spirit is used to create a primary languages programme that will make compound virtual assistant commands with unquoted free-form parameters easier to execute. On realistic human inputs, it achieves a 62 percent accuracy. Genie's generality is no match by showing a 19 and 31st improvement over the previous state of the art on a music talent, mixture functions, and access management.

[4]
On the internet, where the number of options is vast, there is a need to efficiently transmit relevant material so that the problem of data overload is alleviated. This has resulted in a believable fall for numerous internet users. Recommender systems overcome this annoyance by sifting through a vast amount of constantly generated data. Recommendation systems play a critical role in providing users with personalised content and services. This paper examines the various potentials of the numerous prediction approaches used in recommendation systems. The systems act as a compass for analysis, providing the user with the counsel that he or she desires or anticipates based on data from all other users.
[5]
In recent years, chatbots have advanced rapidly in a variety of industries. Education, Health Care, Cultural Heritage, and Recreation are all examples of the same. The study provides a natural overview of the evolution of human interest in chatbots. The reason why one of chatbots is processed, as well as chatbots' utility in a very specific sector. Furthermore, the influence of societal stereotypes on chatbot style is discussed. The study then carries on to a chatbot classification based on a variety of factors, such as the world of information they communicate with, the requirement they serve, and so on. Furthermore, the overall design of contemporary chatbots is imparted, as well as the most common platforms for their construction.

[6]
The focus of this video series is on reinforcement learning (RL). It is covered how to comprehend the intuition, math, and code involved in RL. An introduction to RL is provided first, followed by a discussion of Markov Decision Processes (MDPs) and Q-learning. Then policy gradients and deep RL deep Q-networks (DQNs) are investigated. There are some great RL projects implemented in code utilising Python, PyTorch, and OpenAI Gym.

CHAPTER 3

Reinforcement Learning & Deep Reinforcement Learning

Reinforcement Learning is a branch of Machine Learning that investigates how software agents should behave in a given environment and assigns appropriate rewards.

Reinforcement Learning is a type of deep learning that allows you to maximise a percentage of the total/average reward. This neural network learning method helps to understand how to optimise a single dimension or how to achieve a complex goal across a number of steps.

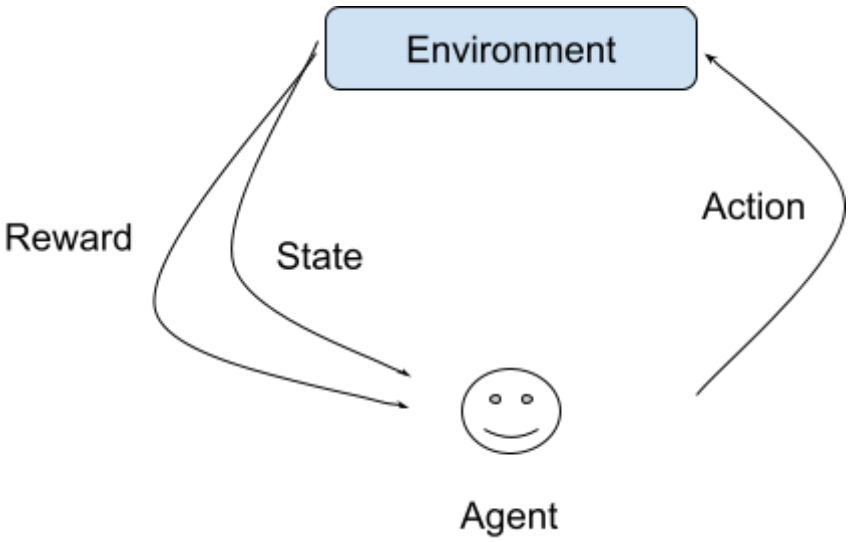


Fig 4 - RL Terms

Consider a self-driving car, in which the car is the agent and the track it must traverse is the environment. Positive benefits include breaking or turning at the proper angle, stopping at a red light, and so on, whereas negative rewards include colliding with another car/person, and so on. In this situation, the agent's clear purpose would be to arrive at the destination with the highest possible payout. The agent has been programmed to make the correct decisions at the appropriate times. We can use Markov decision processes to formalise sequential decision-making.

Terminologies:

- **Agent:** It is an entity which performs actions in an environment in order to gain some reward.
- **State (s):** State is the current situation in the environment by the environment.
- **Policy (π):** It is a strategy which is applied by the agent to decide the next action according to the current state.
- **Environment (e):** A scenario that an agent has to face/surroundings of the assumed agent.
- **Reward (R):** An immediate acknowledgement given to an agent when it performs a particular action or task.
- **Value (V):** In comparison to the short-term reward, it is an expected long-term return with discount.
- **Model of the environment:** This mimics or represents the behavior of the environment. It helps to determine how the environment will behave and make conclusions.
- **Model based methods:** Method for solving RL problems which use model-based methods.
- **Value Function:** Agent which should be expected beginning from that state.
- **Q value or action value (Q):** The only difference between Q value and value is that Q-value takes an additional parameter as a current action.

Working of Reinforcement Learning:

- Consider the situation of training a dog new tricks.
- We can't tell a dog what to do because he doesn't comprehend English or any other human language. Instead, we use a different approach.
- We simulate a situation, and the dog attempts a variety of responses. We will feed the dog if he responds in the desired manner.
- When the dog is exposed to the same situation again, he does a similar activity with even more zeal in the hopes of receiving a larger reward (food).
- That's similar to discovering that positive experiences teach a dog "what to do."
- Simultaneously, the dog learns what not to do when confronted with unpleasant situations.

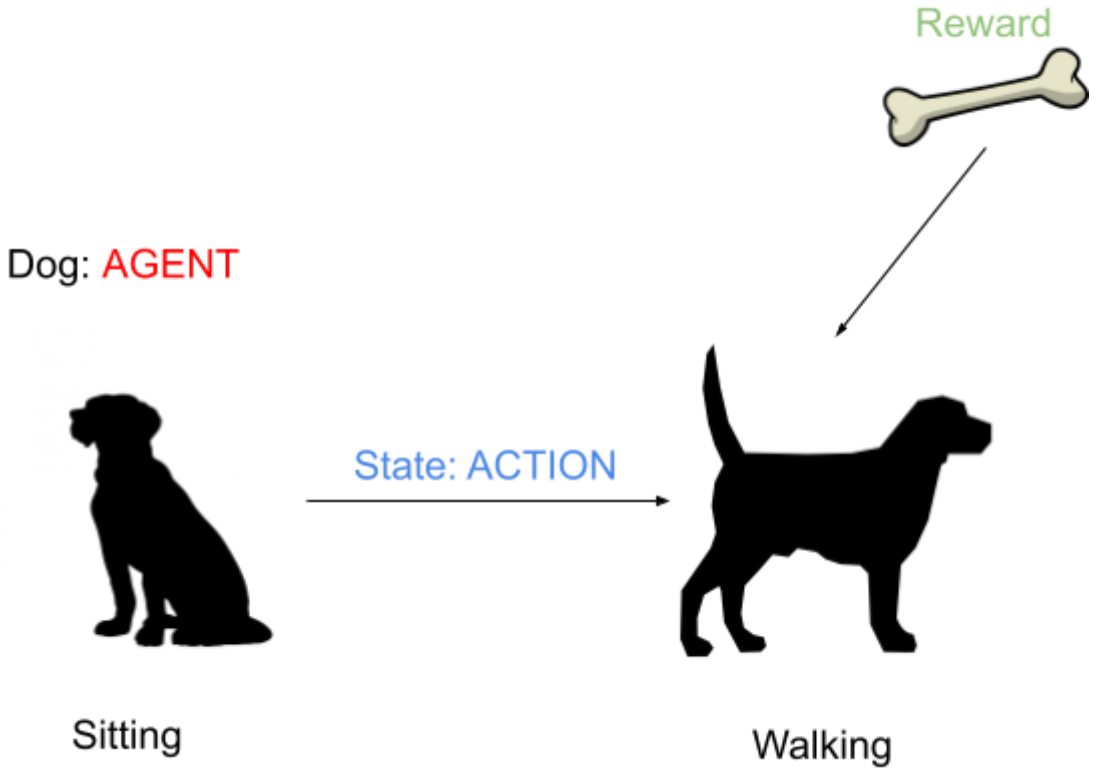


Fig 5 - RL dog example

Reinforcement Learning Algorithms:

There are three approaches to implement a Reinforcement Learning algorithm.

Value-Based:

A value function V should be maximised in a value-based Reinforcement Learning approach (s). In this strategy, the agent anticipates a long-term return of the current policy states.

Policy-based:

In a policy-based RL technique, the goal is to create a policy in which every action taken in each state helps you obtain the most reward in the future.
Two types of policy-based methods are:

- Deterministic: The policy produces the same action for any state.
- Stochastic: Every action has a probability, which may be calculated using the equation below. Stochastic Policy :

n{a\s) = P\A, = a\S, =S]

Model-Based:

You must develop a virtual model for each environment in this Reinforcement Learning approach.

The agent learns how to perform in that particular setting.

Important traits of Reinforcement Learning:

- There is no supervisor, simply a number or a signal of reward.
- Making decisions in a sequential order
- In Reinforcement problems, time is critical.
- Feedback is never immediate; it is always delayed.
- The data that the agent receives is determined by its actions.

Types of Reinforcement Learning

Two kinds of reinforcement learning methods are:

Positive:
It is described as an occurrence that occurs as a result of certain actions. It improves the strength and frequency of the behaviour and has a beneficial impact on the agent's actions.
This type of reinforcement aids in maximising performance and maintaining change for a longer period of time. However, too much reinforcement might lead to state over-optimization, which can have an impact on the outcome.

Negative:
Negative Reinforcement is defined as behaviour strengthening that occurs as a result of a negative condition that should have been avoided or halted. It assists you in establishing a minimal level of performance. The disadvantage of this strategy is that it just gives enough to meet the basic behaviour requirements.

Learning Models of Reinforcement

There are two important learning models in reinforcement learning:

- Markov Decision Process
- Q learning

Markov decision processes

There is an agent in MDP that interacts with the environment in which it is put. The agent successively receives a representation of the current state of the environment through time. The agent picks a specific action based on the state that results in a change in the environmental state, and the agent gets rewarded as a result of the action it performed.

There are 5 major components involved in an MaDP:

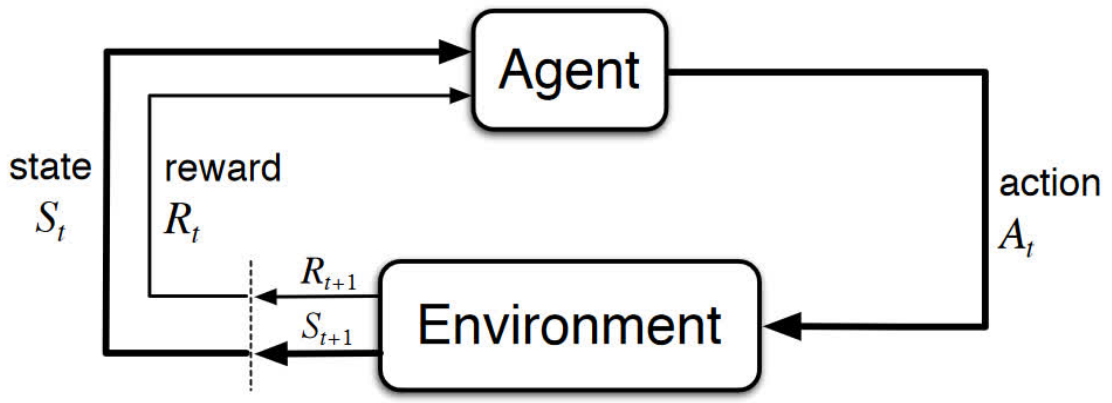
1. Agent
2. Environment
3. State Space (S)
4. Action Space (A)
5. Reward Space (R)

The process of transitioning from one state to another occurs through a series of steps. The agent's purpose throughout this process is to maximise not only the immediate benefit, but also the cumulative rewards it receives over time.

The agent obtains a representation of the current state of the environment St S at each time step t. The agent chooses an action At A based on this state. This results in a state-action pair for time t as follows: (St, At).

The environment is converted to a new state St+1 S, and time is increased to the following time step t+1. The agent is given a numerical reward Rt+1 R for the state-action pair at this point (St, At).

The following diagram illustrates the above-mentioned algorithm.



[8] Fig 6 - Markov decision processes

Expected Return

The agent's purpose is to maximise total rewards, which are calculated as the sum of future benefits up to time step T. This is known as the expected return, which is denoted by G.

$$G = R_{t+1} + a\,R_{t+1} + a^{2}R_{t+1} + \ldots$$

$$G = \sum_{k=0}^{\infty} a^{k} R_{t+k+1}$$

Where R_t is the return at time *t* and *a* is a number ranging from 0 to 1.

Policies and Value Functions

A policy is a function that maps a given state to probabilities of selecting each possible action from that state. It is denoted by $\boldsymbol{\pi}$. If an agent follows policy $\boldsymbol{\pi}$
at time t, then $\boldsymbol{\pi}$(a|s) is the probability that A_t= a if S_t, = s. An optimal policy is a policy that gives the highest expected reward among all the policies.

Value functions give a measure of how good a particular action is. This is achieved by using an action-value function which is denoted by q_{π}. The value of action *A* in state *S* under policy $\boldsymbol{\pi}$ is the expected return from starting from state *S* at time *t*, taking action *A*, and following policy $\boldsymbol{\pi}$ thereafter is given by

$$q_{\pi}(S, A) = E_{\pi}[\sum_{k=0}^{\infty} a^{k} R_{t+k+1} \mid S_t, = S, A_t = A]$$

An optimal state value function is denoted by q_* and is defined as

$$q_* = max\, q_{\pi}(S, A)$$

q_*must always satisfy an equation called as Bellman Optimality Equation which is given as

$$q_*(S, A) = E[R_{t+1} + a * max\, q_*(S', A')]$$

S' and A' denote the next state and action respectively. This equation ensures that the agent is following an optimal policy and the next state S' will be the state from which best possible action A' can be taken at time $t + 1$.

Q-Learning

Q Learning keeps a lookup table of values Q(s, a) with one entry for every state-action pair [2]. The Q-learning algorithm makes use of the Bellman equation for the Q-value function.

Deep Q-Learning

The use of a deep neural network to estimate the Q-values for each state-action pair in a given environment is called deep Q-Learning and the resultant network is called a deep Q-Network. For each given state input, the network outputs the Q-values for each action that can be taken from that state. The Bellman equation is used for this purpose.

$$q_*(S,\ A)\ =\ E[R_{t+1}\ +\ a\ *\ max\ q_*(S',\ A')]$$

The loss associated with the neural network is calculated by comparing the output Q-values with the target Q-values. This loss is then used for backpropagation in order to update the weights in the

Reinforcement Learning vs. Supervised Learning:

Parameters	Reinforcement Learning	Supervised Learning
Decision style	Reinforcement learning enables you to make judgments in a sequential manner.	The input given at the start is used to make a choice in this technique.
Works on	focuses on Interacting with the environment is a focus.	Works with examples or data provided as a sample.
Dependency on decision	The decision is dependent on the RL method of learning. As a result, all of the dependent decisions should be labelled.	Supervised learning of judgments that are unrelated to one another, with labels assigned to each decision.
Best suited	Supports and operates better in AI where there is a lot of human involvement.	It is generally controlled by a software system or programmes that are interactive.
Example	Chess game	Object recognition

CHAPTER 4

APPLICATIONS OF RL and DRL

- OTT platforms like Netflix, Amazon prime, etc.
- Social Media (Eg: Facebook suggestions).
- Gaming.
- Google Maps.
- CNN.
- Virtual Assistants.
- Self-driving cars.
- Robotics for industrial automation.
- Business strategy planning.
- Machine learning and data processing.
- It helps you to create training systems that provide custom instruction and materials according to the requirements of students.
- Aircraft control and robot motion control.

CHAPTER 5

Advantages and Disadvantages

Advantages of RL:

- It aids in determining which situations require action.
- Aids you in determining which activity delivers the highest benefit over time.
- Reinforcement Learning also includes a reward function for the learning agent.
- It also enables it to determine the most efficient means of collecting significant rewards.

Disadvantages of RL:

- When enough data is available to solve the problem using a supervised learning method, the reinforcement learning model cannot be used.
- Reinforcement Learning is time-consuming and computationally intensive, especially when the action space is huge.
- Design of the feature/reward, which should be highly involved.
- The pace with which you learn can be influenced by a variety of factors.
- Partially observable environments are possible in realistic settings.
- Too much reinforcement might result in an overabundance of states, lowering the quality of the output.
- Non-stationary situations are possible in realistic settings.

CHAPTER 6

CONCLUSION

- Reinforcement Learning is a form of Machine Learning.
- Aids you in determining which activity delivers the highest benefit over time.
- Three methods for reinforcement learning are 1) Value-based 2) Policy-based and Model based learning.
- Value function, Agent, State, Reward, Environment The terms "model of the environment" and "model-based approaches" are used frequently in the RL learning method.
- Your cat is an agent that is exposed to the environment, which is an example of reinforcement learning.
- The most notable feature of this system is that there is no supervisor involved; instead, a genuine number or incentive signal is used.
- Two types of reinforcement learning are 1) Positive 2) Negative.
- Two widely used learning models are 1) Markov Decision Process 2) Q learning.
- The supervised learning approach works on given sample data or example, but the reinforcement learning method works on interacting with the environment.
- Robotics for industrial automation and business strategy planning are two examples of applications or reinforcement learning methodologies.
- When there is enough data to answer the problem, this method should not be employed.
- The most significant disadvantage of this strategy is that factors can influence learning speed.

CHAPTER 7

REFERENCES

[1]- Adrià Puigdomènech Badia, "**Never Give Up: Learning Directed Exploration Strategies**" arXiv:2002.06038 [cs.LG], 14 Feb 2020.

[2] Urs Muller, "**End to End Learning for Self-Driving Cars**" arXiv:1604.07316 [cs.CV], Apr 2016.

[3] Giovanni Campagna, "**Genie: A Generator of Natural Language Semantic Parsers for Virtual Assistant Commands**" arXiv:1904.09020 [cs.CL], Apr 2019.

[4] F.O. Isinkaye, Y.O. Folajimi, B.A. Ojokoh, "**Recommendation systems: Principles, methods and evaluation"**, Egyptian Informatics Journal, Volume 16, Issue 3, 2015, Pages 261-273.

[5] Adamopoulou E., Moussiades L. (2020) "**An Overview of Chatbot Technology"**. In: Maglogiannis I., Iliadis L., Pimenidis E. (eds) Artificial Intelligence Applications and Innovations. AIAI 2020. IFIP Advances in Information and Communication Technology, vol 584. Springer, Cham. https://doi.org/10.1007/978-3-030-49186-4_31

[6] Chris and Mandy. "Reinforcement Learning - Goal Oriented Intelligence". ***deeplearning.ai,*** Sep 2018

[7] Fig 3 - Atari game RL, official - "Never provide Up: Learning Directed Exploration Strategies", https://vitalab.github.io/article/2020/05/28/NGU.html

[8] Fig 6 - Markov decision processes, https://towardsdatascience.com/introduction-to-reinforcement-learning-markov-decision-process-44c533ebf8da

CHAPTER 8

PLAGIARISM REPORT