

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 **Prof. 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

Acknowledgement	I
Abstract	II
List of Tables	IV
List of Figures	IV

Sr.	Chapter	Page No
1.	Introduction	
1.1	Introduction to RL	1
1.2	Motivation behind RL and DRL	2
1.3	Objective(s) of the work	2
1.4	Introduction to RL and DRL	2
2.	Literature Survey of Seminar Topic	
2.1	Explanations of various papers used as reference.	3
3.	Reinforcement Learning & Deep Reinforcement Learning	
3.1	Introduction	4
3.2	Terminologies	5
3.3	Working of RL	6
3.4	Algorithms	7
3.5	Important traits of RL	7
3.6	Types of RL	8
3.7	Learning models of RL	8
3.8	Markov decision process	9
3.9	Expected return	10
3.10	Policies and value functions	10
3.11	Q-Learning	11
	Deep Q-Learning	11

	Reinforcement Learning vs Supervised Learning	12
4	Applications of RL and DRL	13
4.1	State of Art applications	
4.2	Advantages and Disadvantages	
5.	Conclusion	14
6	References	15
	Appendix	
	Plagiarism Report	16

LIST OF FIGURES

Sr. No.	Figure Name	Page no.
1	Branches of Automation	1
2	ML Types	1
3	Atari game RL	3
4	RL Terms	4
5	RL Dog example	6
6	Markov decision processes	9

LIST OF TABLES

Sr. No.	Table Name	Page no.
1	Reinforcement Learning vs. Supervised Learning	12

CHAPTER 1

INTRODUCTION TO SEMINAR TOPIC

1.1 Introduction to RL and DRL:

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.

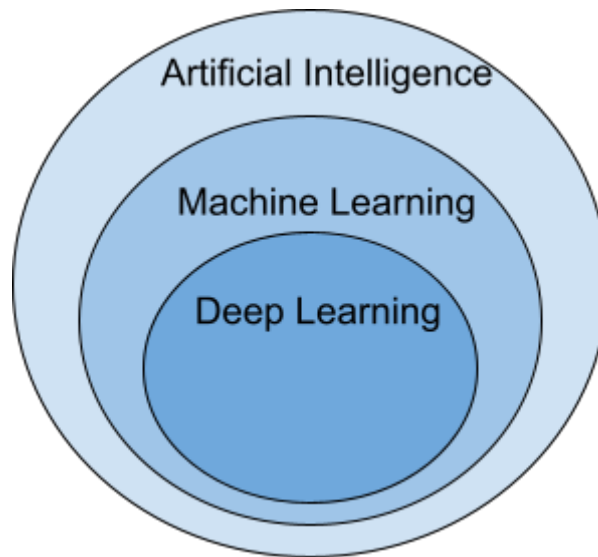


Fig 1- Branches of Automation

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.

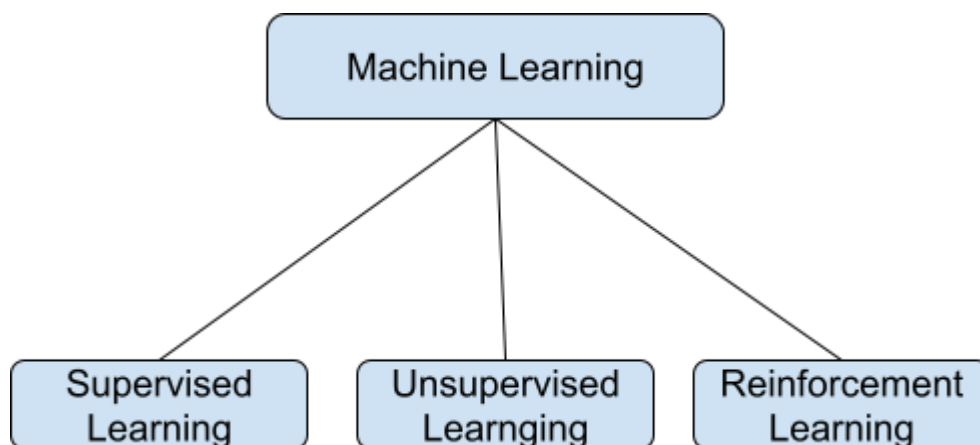


Fig 2 - ML Types

1.2 **Motivation behind RL and DRL:**

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 **Aim and Objective(s) of the work**

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 the machine learning domain that teaches and helps an agent in how to choose an action/movement from its action space, within a particular environment. This is one in order to maximize rewards by training over time.

When Deep Learning is combined and integrated with RL it is known as Deep Reinforcement Learning. It is seen in many real-world and amusing applications around us that might be unnoticed like games, recommendation engines for famous e-commerce websites, video-streaming platforms, etc.

This seminar will mostly cover the application of DRL which is used in recommendation systems. Recommendation systems are something that we all come across daily from recommendations on Google maps to OTT services like Netflix, Amazon. Increasing watch time for video streaming platforms or increasing sales for e-commerce websites can be done by self-supervised recommendation systems. There are multiple factors like the interactions between users and items that are taken into account. Recommendation systems can be improved and enhanced by RL by concentrating on the interaction between the agent i.e. the user and the environment(a website in this case). Which results in maximizing the cumulative and average reward for the agent based on the interaction with the environment.

This seminar focuses on how reinforcement learning and deep reinforcement learning work and their use in 2 state of art recommendations frameworks namely Self-Supervised Q-learning (SQN) and Self-Supervised Actor-Critic (SAC).

Objectives of this seminar include gaining a good understanding of MDP and applying this knowledge to understand the 2 recommendations frameworks mentioned above.

CHAPTER 2

LITERATURE SURVEY OF RL and DRL

[“Never Give Up: Learning Directed Exploration Strategies”](#).

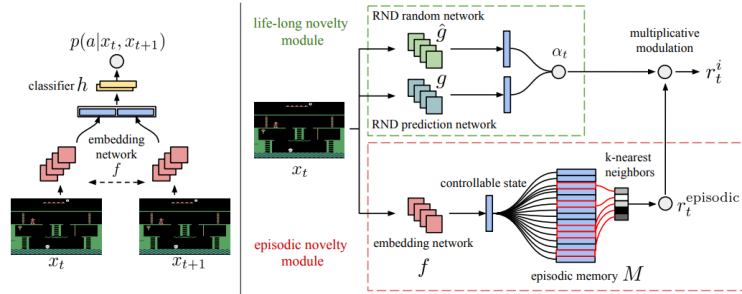


Fig 3 - Atari game RL, Ref - “Never Give Up: Learning Directed Exploration Strategies”

[“End to End Learning for Self-Driving Cars”](#)

[“Genie: A Generator of Natural Language Semantic Parsers for Virtual Assistant Commands”](#)

[“Recommendation systems: Principles, methods and evaluation”](#)

[“An Overview of Chatbot Technology”](#)

[“DynaMIT: a simulation-based system for traffic prediction”](#)

CHAPTER 3

Reinforcement Learning & Deep Reinforcement Learning

Reinforcement Learning is a part of Machine Learning that encounters how software agents should take actions in an environment and reward is given accordingly.

Reinforcement Learning is a part of the deep learning method, this helps to maximize some portion of the cumulative/average reward. Over many steps this neural network learning method helps to understand maximize a specific dimension or how to attain a complex objective.

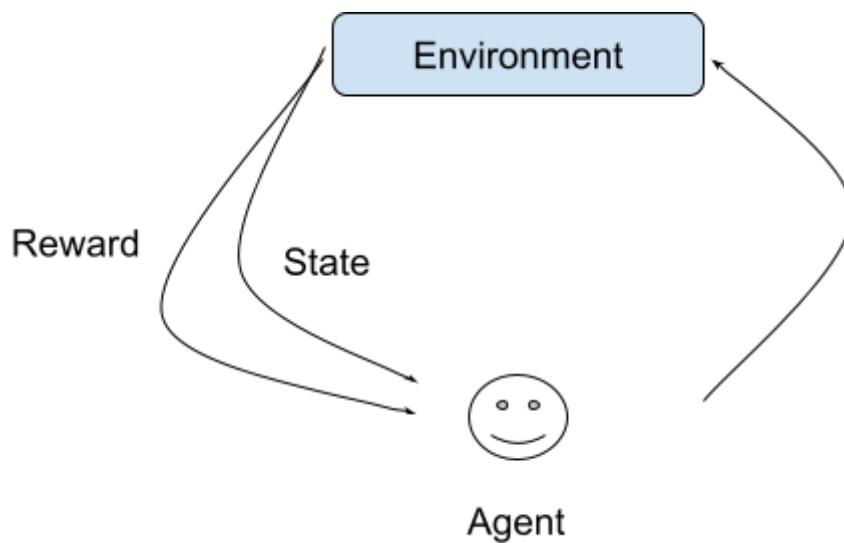


Fig 4 - RL Terms

Take the example of a self-driving car, where the car is the agent, and the track that it has to cover is the environment. Positive rewards may include breaking or turning at the right point, stopping at a red light, etc and negative rewards may include crashing with another car/person, etc. So the clear goal for the agent, in this case, would be to reach the destination with maximum reward. The agent is programmed to make the right decisions at the right time. Markov decision processes give us a way to formalize 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 scenario of teaching new tricks to a dog.

- As a dog doesn't understand English or any other human language, we can't tell him directly what to do. Instead, we follow a different strategy.
- We emulate a situation, and the dog tries to respond in many different ways. If the dog's response is the desired way, we will give him food.
- Now whenever the dog is exposed to the same situation, the dog executes a similar action with even more enthusiasm in expectation of getting more reward(food).
- That's like learning that a dog gets "what to do" from positive experiences.
- At the same time, the dog also learns what not to do when faced with negative experiences.

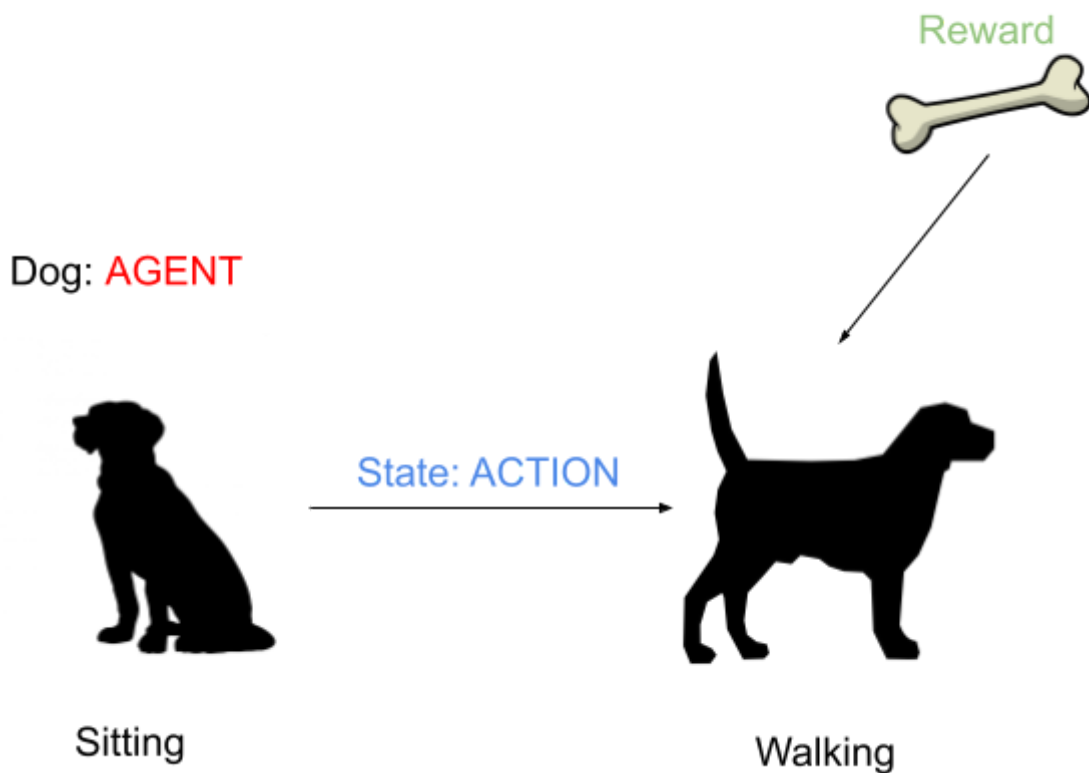


Fig 5 - RL dog example

Reinforcement Learning Algorithms:

There are three approaches to implement a Reinforcement Learning algorithm.

Value-Based:

In a value-based Reinforcement Learning method, one should try to maximize a value function $V(s)$. In this method, the agent is expecting a long-term return of the current states under policy π .

Policy-based:

In a policy-based RL method, one should try to come up with such a policy that the action performed in every state helps you to gain maximum reward in the future.

Two types of policy-based methods are:

- Deterministic: For any state, the same action is produced by the policy π .
- Stochastic: Every action has a certain probability, which is determined by the following equation. Stochastic Policy :

$$p(a|s) = P(A_t = a | S_t = s)$$

Model-Based:

In this Reinforcement Learning method, you need to create a virtual model for each environment.

The agent learns to perform in that specific environment.

Important traits of Reinforcement Learning:

- There is no supervisor, only a real number or reward signal
- Sequential decision making
- Time plays a crucial role in Reinforcement problems
- Feedback is always delayed, not instantaneous
- Agent's actions determine the subsequent data it receives

Types of Reinforcement Learning

Two kinds of reinforcement learning methods are:

Positive:

It is defined as an event that occurs because of specific behavior. It increases the strength and the frequency of the behavior and impacts positively on the action taken by the agent.

This type of Reinforcement helps you to maximize performance and sustain change for a more extended period. However, too much Reinforcement may lead to over-optimization of state, which can affect the results.

Negative:

Negative Reinforcement is defined as strengthening of behavior that occurs because of a negative condition which should have stopped or avoided. It helps you to define the minimum standard of performance. However, the drawback of this method is that it provides enough to meet up the minimum behavior.

Learning Models of Reinforcement

There are two important learning models in reinforcement learning:

- Markov Decision Process
- Q learning

Markov decision processes

In MDP, there is an agent that interacts with the environment that it is placed in. The agent gets a representation of the current state of the environment sequentially over time. Based on the state, the agent chooses a particular action which leads to a change in the environmental state, and the agent is rewarded as a cause of the action that it took.

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 happens sequentially over and over again. Throughout this process, it is the agent's goal to maximize not just the immediate reward, but the cumulative rewards it receives over time

At each time step t , the agent receives a representation of the environment's current state $S_t \in S$.

Based on this state, the agent selects an action $A_t \in A$. This gives a state-action pair for time t as (S_t, A_t) .

Time is incremented to the next time step $t+1$, and the environment is transitioned to a new state $S_{t+1} \in S$. At this time, the agent receives a numerical reward $R_{t+1} \in R$ for the state-action pair (S_t, A_t) .

The following diagram (fig 1) illustrates the above-mentioned algorithm.

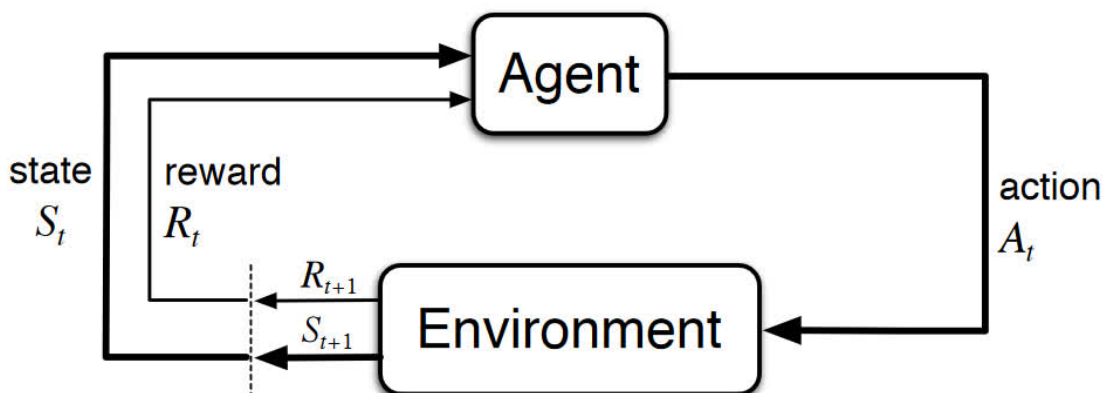


Fig 6 - Markov decision processes

Expected Return

The goal of the agent is to maximize the total rewards which can be represented as the sum of future rewards till the last time step T . This is called the expected return denoted by G . G is given by

$$G = R_{t+1} + a R_{t+1} + a^2 R_{t+1} + \dots$$
$$G = \sum_{k=0}^{\infty} a^k R_{t+k+1}$$

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 π . If an agent follows policy π at time t , then $\pi(a|s)$ 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 π is the expected return from starting from state S at time t , taking action A , and following policy π thereafter is given by

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

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

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

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

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

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.

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

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 helps you to take your decisions sequentially.	In this method, a decision is made on the input given at the beginning.
Works on	Works on interacting with the environment.	Works on examples or given sample data.
Dependency on decision	In RL method learning, decision is dependent. Therefore, you should give labels to all the dependent decisions.	Supervised learning the decisions which are independent of each other, so labels are given for every decision.
Best suited	Supports and works better in AI, where human interaction is prevalent.	It is mostly operated with an interactive software system or applications.
Example	Chess game	Object recognition

CHAPTER IV

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.

Advantages of RL:

- It helps to find which situation needs action.
- Helps you to discover which action yields the highest reward over the longer period.
- Reinforcement Learning also provides the learning agent with a reward function.
- It also allows it to figure out the best method for obtaining large rewards.

Disadvantages of RL:

- When there is enough data to solve the problem with a supervised learning method reinforcement learning model can't be applied.
- Reinforcement Learning is computing-heavy and time-consuming, in particular when the action space is large.
- Feature/reward design which should be very involved.
- Parameters may affect the speed of learning.
- Realistic environments can have partial observability.
- Too much Reinforcement may lead to an overload of states which can diminish the results.
- Realistic environments can be non-stationary.

CHAPTER V

CONCLUSION

- Reinforcement Learning is a Machine Learning method.
- Helps you to discover which action yields the highest reward over the longer period.
- Three methods for reinforcement learning are 1) Value-based 2) Policy-based and Model based learning.
- Agent, State, Reward, Environment, Value function Model of the environment, Model based methods, are some important terms using in RL learning method.
- The example of reinforcement learning is your cat is an agent that is exposed to the environment.
- The biggest characteristic of this method is that there is no supervisor, only a real number or reward signal.
- Two types of reinforcement learning are 1) Positive 2) Negative.
- Two widely used learning models are 1) Markov Decision Process 2) Q learning.
- Reinforcement Learning method works on interacting with the environment, whereas the supervised learning method works on given sample data or example.
- Application or reinforcement learning methods are: Robotics for industrial automation and business strategy planning.
- This method should not be used when there is enough data to solve the problem.
- The biggest challenge of this method is that parameters may affect the speed of learning.

REFERENCES

- [1] Mariusz Bojarski, Davide Del Testa, Daniel Dworakowski, Bernhard Firner, Beat Flepp, Praseen Goyal, Lawrence D. Jackel, Mathew Monfort, Urs Muller, Jiakai Zhang, Xin Zhang, Jake Zhao, Karol Zieba – **“End to End Learning for Self-Driving Cars”**.
- [2] Giovanni Campagna, Silei Xu, Mehrad Moradshahi, Richard Socher, Monica S. Lam - **“Genie: A Generator of Natural Language Semantic Parsers for Virtual Assistant Commands”**
- [3] Eleni AdamopoulouEmail authorLefteris Moussiades - **“Recommendation systems: Principles, methods and evaluation”**, Egyptian Informatics Journal, Volume 16, Issue 3, November 2015.
- [4] 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
- [5] Ben-Akiva, Moshe & Bierlaire, Michel & Koutsopoulos, Haris & Mishalani, Rabi. (2000). **“DynaMIT: a simulation-based system for traffic prediction”**.
- [6] Chris and Mandy. **“Reinforcement Learning - Goal Oriented Intelligence”**. deeplearning.ai, Sep 2018

PLAGIARISM REPORT