

LECTURER: TAI LE QUY

ARTIFICIAL INTELLIGENCE

TOPIC OUTLINE

History of AI

1

Modern AI Systems

2

Reinforcement Learning

3

Natural Language Processing – Part 1

4

Natural Language Processing – Part 2

5

Computer Vision

6

UNIT 3

REINFORCEMENT LEARNING

STUDY GOALS



- Understand the basic principles of reinforcement learning.
- Utilize Markov decision processes.
- Apply the Q-learning algorithm.



1. How does reinforcement learning work?
2. What is temporal difference learning?
3. How does the Q-learning algorithm work?

MAIN MACHINE LEARNING TASKS

Based on the feedback we have on the data:

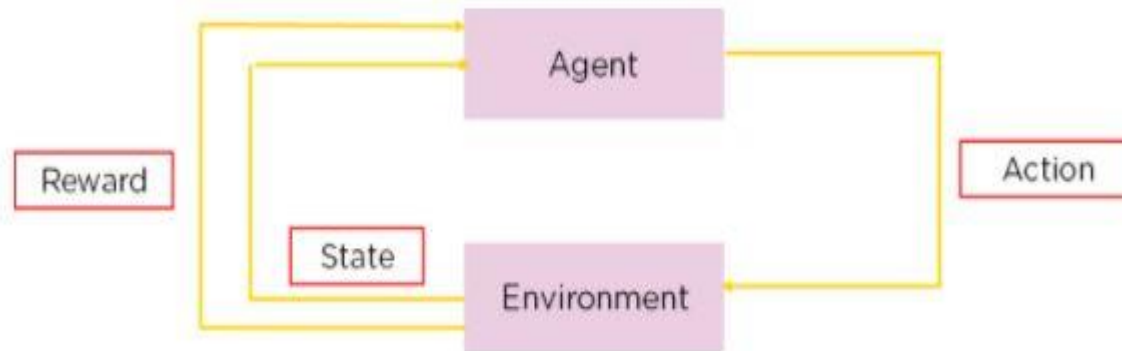
- **Direct-feedback** instances **Supervised learning**
 - the correct response /label is provided for each instance by the “teacher”
 - e.g., good or bad product
- **No-feedback** instances **Unsupervised learning**
 - no evaluation/label of the instance is provided, since there is no “teacher”
 - e.g., no information on whether a product is good or bad, just the description of the product/instance
- **Indirect-feedback** instances **Reinforcement learning**
 - less feedback is given, since not the proper action, but only an evaluation of the chosen action is given by the “teacher”

fruit	length	width	weight	label
fruit 1	165	38	172	Banana
fruit 2	218	39	230	Banana
fruit 3	76	80	145	Orange
fruit 4	145	35	150	Banana
fruit 5	90	88	160	Orange
...				
fruit n

fruit	length	width	weight
fruit 1	165	38	172
fruit 2	218	39	230
fruit 3	76	80	145
fruit 4	145	35	150
fruit 5	90	88	160
...			
fruit n

AGENT AND ENVIRONMENT

- Reinforcement learning is a type of machine learning technique that enables an **agent** to learn in an interactive **environment** by trial and error using **feedback** from its own **actions** and **experiences**.



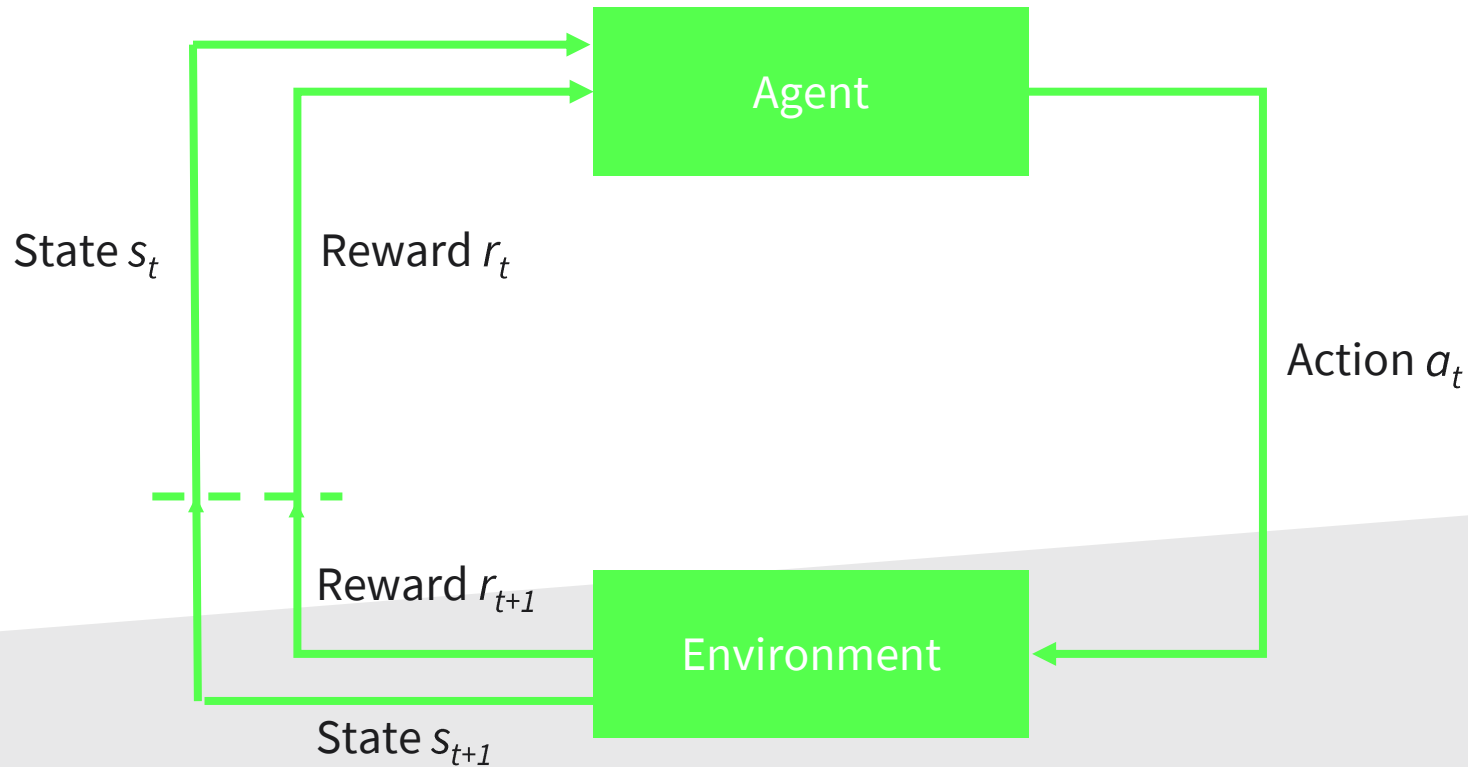
In the terminology of psychology,
reward is reinforcement

- The goal of the agent is to learn to choose actions so as to **maximize** the sum of rewards

BASIC TERMS OF REINFORCEMENT LEARNING

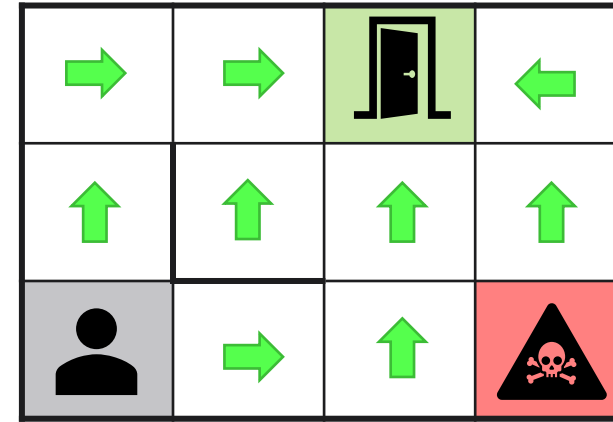
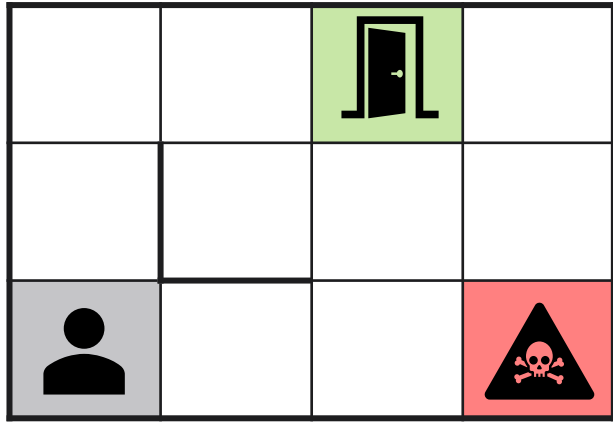
Terms	Description
Agent	Performs actions in an environment and receives reward for doing so
Action (A)	The set of all possible actions the agent can perform
State (S)	The current state of the agent in the given environment
Reward (R)	Immediate feedback from the environment to reward an agent's action
Policy (π)	The policy the agent applies to determine the next action based on the current state
Value (V)	The long-term value of the current state S using the policy π

THE PROCESS OF REINFORCEMENT LEARNING



- The agent starts in a certain state $s_t \in S$ and applies an action $a_t \in A(s_t)$ to the environment E , where $A(s_t)$ is the set of actions available at state s_t .
- The environment reacts by returning a new state s_{t+1} and a reward r_{t+1} to the agent.
- In the next step the agent will apply the next action a_{t+1} to the environment which will again return a new state and a reward.

REINFORCEMENT LEARNING EXAMPLE



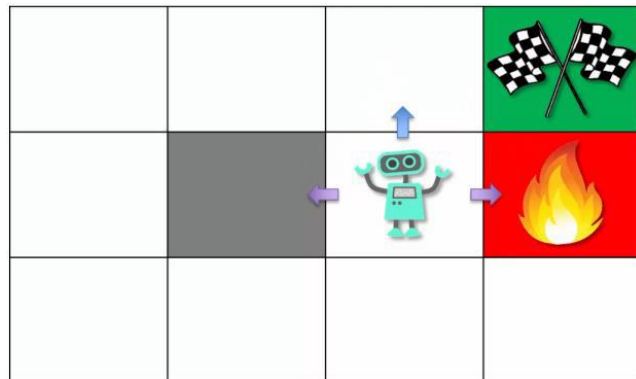
- State: the location of the person at a specific time
- Action: go left, go right, go up, go down
- Reward: 1 if the person manages to get out of the door and 0 if go to a toxic room
- The idea is the person “remembers” which way he/she should go to get the reward → maximize the rewards over time → maximize the (expected) return
 - Trying out actions and observing the outcomes ← data

HOW THE AGENT LEARN?

- Model-based learning
 - Learn an approximate model of the environment based on experiences/data
 - All decisions are based on a **value function**. The value function is based on the current state and the future state where the agent will end
 - Use the model to solve a (conventional) MDP (Markov Decision Process)
- Model-free learning
 - Analyze the quality of an action to evaluate their actions
 - Temporal difference (TD) learning
 - Q-learning

MARKOV DECISION PROCESS

- MDPs are used to estimate the probability of a future event based on a sequence of possible events.
- A stochastic process has the **Markov property** if the conditional probability distribution of future states of the process (conditional on both past and present states) depends only upon the present state, not on the sequence of events preceded it.
- A process with Markov property is called a **Markov decision process**



The state that the agent is in now does not depend on how it gets there. The future only depends on where the agent is and the actions it will take.

COMPONENTS OF THE MARKOV DECISION PROCESS

- States: S
- Actions: A
- Rewards for an action a at a state s : $r_a = R(s, a, s')$
- Transition probabilities for the actions to move from one state to the next state: $T_a(s, s')$
- Transition function: $T_{a_t}(s, s') = P(s_{t+1} | s_t, a_t)$
- Policy: $\pi(s, a) = p(a_t = a | s_t = s)$

WHAT IS MARKOV ABOUT MDPS?

- “Markov” generally means “The future is independent of the past given the present”
- For Markov decision processes (MDPs), “Markov” means action outcomes depend only on the current state

$$P(s_{t+1}|s_t, a_t, s_{t-1}, a_{t-1}, \dots) = P(s_{t+1}|s_t, a_t) = T_a(s, s')$$

- The policy π describes the action which is picked in a certain state:

$$\pi(s, a) = p(a_t = a | s_t = s)$$



Russian
mathematician
Andrey Markov
(1856-1922)

THE VALUE FUNCTION

- The total reward R after a time T

$$R_t = r_{t+1} + r_{t+2} + \cdots + R_T$$

- The reward is also referred to as the value $V_{\pi}(s)$ in the state s using the strategy (policy) π
- The value function (or V-value)
 - It is a prediction of future reward
 - Used to evaluate the goodness/badness of states

- Value function

$$V_{\pi}(s) = \mathbb{E}_{\pi}\{r_{t+1} + \gamma r_{t+2} + \cdots + \gamma^{T-1} r_T\}$$

$$= \mathbb{E}_{\pi} \left\{ \sum_{k=0}^{\infty} \gamma^k r_{t+k+1} \mid s_t = s \right\}$$

- $0 \leq \gamma \leq 1$ is the **discount** rate
 - Determines the importance of future rewards at the present state
 - Indicates the likelihood to reach a reward state in the future
 - Helps agent select actions more precisely according to the expected reward
 - It is important to set $\gamma < 1$ as otherwise the value function will not converge.
- An action a_{t+1} will be chosen to maximize the expected discounted return

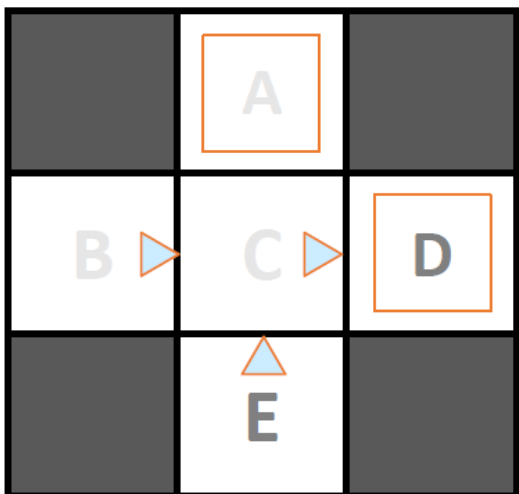
MODEL-BASED LEARNING

- Model-based learning idea
 - (Step 1) Learn an approximate model of T , R based on experiences/data
 - Count outcomes for each pair of (state, action)
 - (Step 2) Solve the MDP based on the learned T , R
 - Use value iteration or policy iteration

EXAMPLE: MODEL-BASED LEARNING

Learn the empirical model

Input Policy π



Assume: $\gamma = 1$

Observed Episodes (Training)

Episode 1

B, east, C, -1
C, east, D, -1
D, exit, x, +10

Episode 2

B, east, C, -1
C, east, D, -1
D, exit, x, +10

Episode 3

E, north, C, -1
C, east, D, -1
D, exit, x, +10

Episode 4

E, north, C, -1
C, east, A, -1
A, exit, x, -10

Learned Model

$$\hat{T}(s, a, s')$$

$T(B, \text{east}, C) = 1.00$
 $T(C, \text{east}, D) = 0.75$
 $T(C, \text{east}, A) = 0.25$
...

$$\hat{R}(s, a, s')$$

$R(B, \text{east}, C) = -1$
 $R(C, \text{east}, D) = -1$
 $R(D, \text{exit}, x) = +10$
...

Assumption: the reward is deterministic

MODEL-FREE LEARNING

- Model-based learning
 - Agent tries to understand the model of the environment
 - All decisions are based on a **value function**.
- Model-free learning
 - Analyze the quality of an action
 - Learn the V-values and Q-values of state-action pairs directly, without constructing a model of the rewards and transitions in the MDP

TEMPORAL DIFFERENCE LEARNING

- Main idea of TD learning
 - Learning from every experience
 - TD learning makes prediction based on the fact that there is often a correlation between subsequent predictions (considers the temporal difference between subsequent predictions)

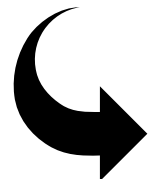
Q-LEARNING

- Main idea
 - Initialization: set all values in the Q-table to 0
 - Iteration: evaluate the outcome of action, then the agent will adapt its behavior for the subsequent actions.
- Goals
 - Maximize the quality function $Q(s,a)$: accumulative reward
- Bellman equation
 - The Bellman equation computes the expected reward in an MDP of taking an action in a certain state. The reward is broken into the immediate and the total future expected reward.

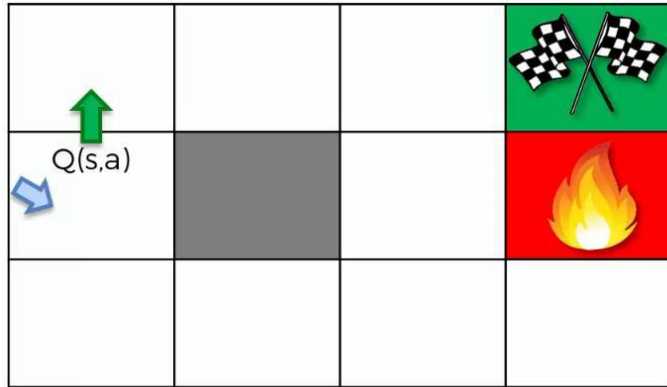
- Model-free approach → learning from experience

$$\underbrace{Q(s, a)}_{\text{Original state (t-1)}} = \underbrace{r}_{\text{Reward}} + \gamma \underbrace{\max_{a'} Q(s', a')}_{\text{New state (t)}}$$

$$TD(s, a) = r + \gamma \max_{a'} Q(s', a') - Q(s, a)$$

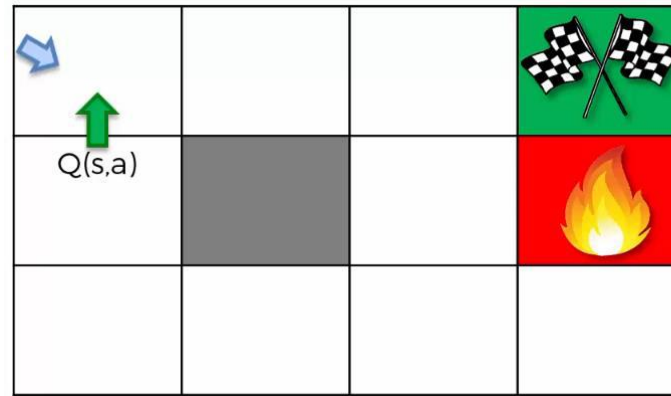

$$Q_t(s, a) = Q_{t-1}(s, a) + \underbrace{\alpha}_{\text{Learning rate}} TD_t(s, a)$$

TEMPORAL DIFFERENCE LEARNING



Before:

$$Q(s, a)$$



Before:

$$Q(s, a)$$

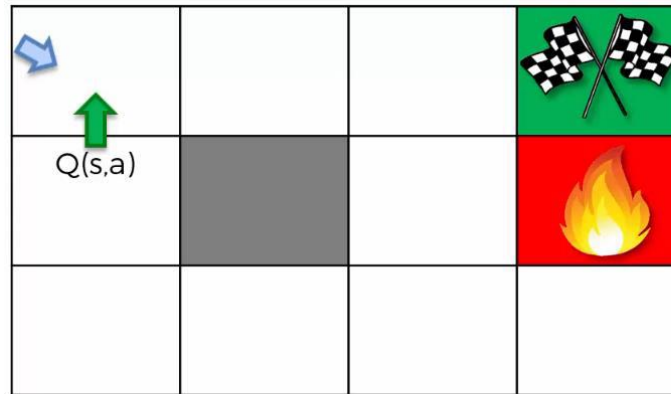
After:

$$R(s, a) + \gamma \max_{a'} Q(s', a')$$

- We have Q-values calculated by the agent walking around.
- The agent is sitting in the blue-arrow cell and it needs to make a choice

$$TD(a, s) = R(s, a) + \gamma \max_{a'} Q(s', a') - Q_{t-1}(s, a)$$

$$Q_t(s, a) = Q_{t-1}(s, a) + \alpha TD_t(a, s)$$



Before:

$$Q(s, a)$$

After:

$$R(s, a) + \gamma \max_{a'} Q(s', a')$$

$$TD(a, s) = \overbrace{R(s, a) + \gamma \max_{a'} Q(s', a')} - \underbrace{Q(s, a)}$$

Based on temporal difference learning

Q-Learning Algorithm:

1. Choose an action for the current state. Possible strategies:
 - **Exploration:** perform random actions in order to find out more about the environment
 - **Exploitation:** perform actions based on known information
2. Perform chosen action
3. Evaluate the outcome and get the value of the reward, update Q-table

Q-LEARNING: EXAMPLE

Initialized

Q-Table		Actions					
		South (0)	North (1)	East (2)	West (3)	Pickup (4)	Dropoff (5)
States	0	0	0	0	0	0	0

	327	0	0	0	0	0	0

	499	0	0	0	0	0	0

Training

Q-Table		Actions					
		South (0)	North (1)	East (2)	West (3)	Pickup (4)	Dropoff (5)
States	0	0	0	0	0	0	0

	328	-2.30108105	-1.97092096	-2.30357004	-2.20591839	-10.3607344	-8.5583017

	499	9.96984239	4.02706992	12.96022777	29	3.32877873	3.38230603



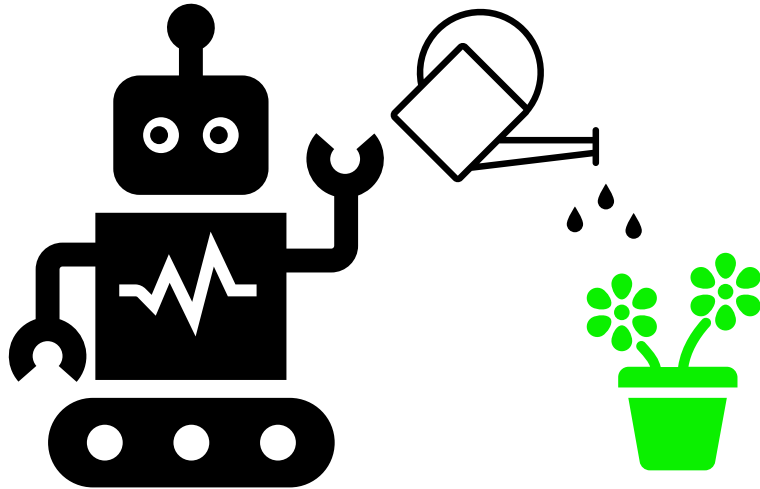
- Understand the basic principles of reinforcement learning.
- Utilize Markov decision processes.
- Apply the Q-learning algorithm.

SESSION 3

TRANSFER TASK

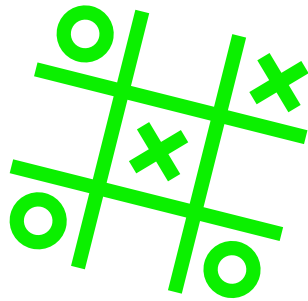
TRANSFER TASKS

1) Imagine you have a gardening bot to water your plants. How could the bot apply reinforcement learning to learn how to perfectly water the plants?



TRANSFER TASK

2) You have a computer that learns a game by playing against a random opponent. How would the learning process change if the computer played against another computer using the same algorithm?



3) In Q-Learning exploration and exploitation play an important role. What is more important at what stage of the learning process and why?

TRANSFER TASK
PRESENTATION OF THE RESULTS

Please present your
results.

The results will be
discussed in plenary.



TRANSFER TASK
PRESENTATION OF THE RESULTS

Please present your
results.

The results will be
discussed in plenary.





1. Who performs the actions in reinforcement learning?
 - a. The agent
 - b. The policy
 - c. The present state
 - d. A model-free approach



2. What component describes which action is picked in a certain state?

- a. The policy
- b. The environment
- c. The value function
- d. The agent



3. What do the decisions in MDPs depend on?
- a. The history of states
 - b. The future state
 - c. The final state
 - d. The present state



4. What kind of approach is used in temporal difference learning?
- a. A model-based approach
 - b. A model-free approach
 - c. A data-driven approach
 - d. A supervised approach

LIST OF SOURCES

- Ntoutsj, Eirini, “Reinforcement learning”, Machine learning for Data science, FU Berlin, 2021
- Nghia Duong-Trung, Artificial Intelligence, IU University of Applied Sciences, 2022

© 2022 IU Internationale Hochschule GmbH

This content is protected by copyright. All rights reserved.

This content may not be reproduced and/or electronically edited, duplicated, or distributed in any kind of form without written permission by the IU Internationale Hochschule GmbH.