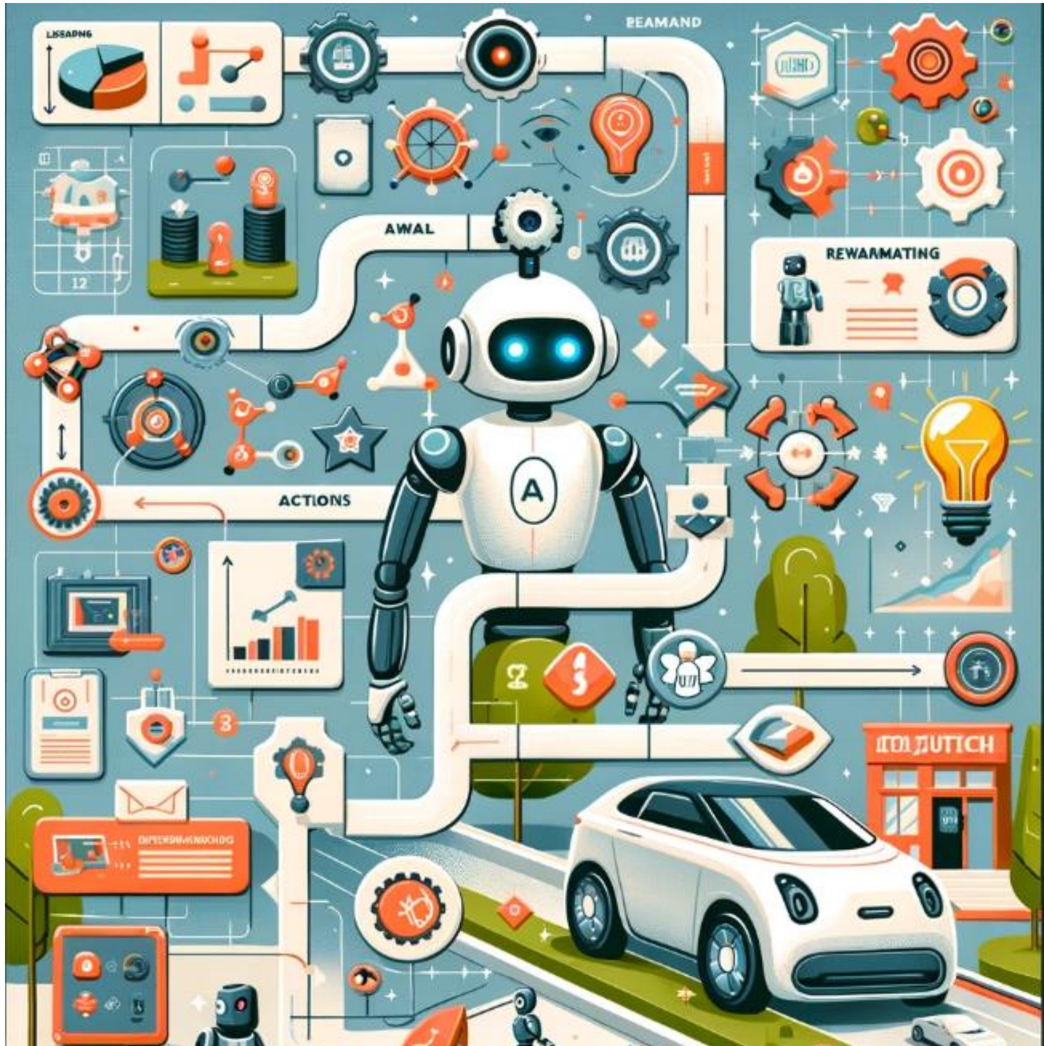
Machine Learning

| Reinforcement Learning

Prof. Dr. Mostafa Elhosseini

Professor of Smart Systems Engineering

https://youtube.com/drmelhosseini



Agenda

- Reinforcement Learning (RL) is a type of machine learning where an agent learns to make decisions by performing actions in an environment to achieve a goal.
- The fundamental concept of RL is the agent's interaction with the environment and learning from the consequences of its actions through rewards or penalties. This process is similar to how humans learn from experiences in the real world

Reinforcement Learning

- Reinforcement Learning (RL) is a type of machine learning where an agent learns to make decisions by performing actions in an environment to achieve a goal.
- The fundamental concept of RL is the agent's interaction with the environment and learning from the consequences of its actions through rewards or penalties. This process is similar to how humans learn from experiences in the real world

Reinforcement Learning

Key Components

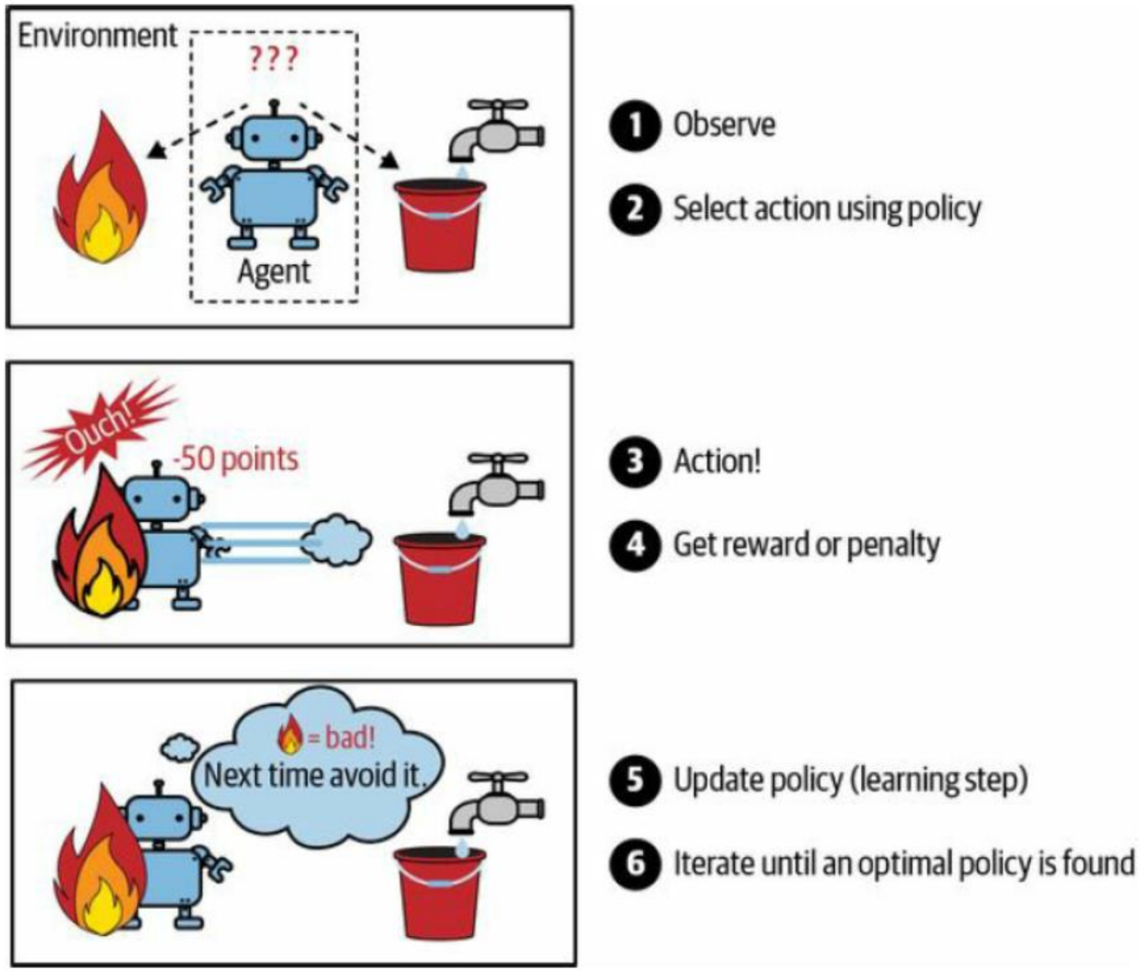
- **Agent**: The learner or decision-maker.
- **Environment**: The space in which the agent operates and interacts.
- **State**: A representation of the current situation or condition in which the agent is.
- **Actions**: What the agent can do or the choices it can make.
- **Reward**: Feedback from the environment in response to the agent's actions. Rewards can be positive (reinforcing the action) or negative (discouraging the action).
- **Policy**: A strategy used by the agent to determine its actions based on the current state.

Reinforcement Learning

How Reinforcement Learning Works:

- The agent observes the current state of the environment.
- Based on this observation, the agent takes an action.
- The environment transitions to a new state and gives feedback to the agent in the form of a reward.
- The agent learns a policy over time that maximizes the cumulative reward it receives.

Reinforcement Learning



Reinforcement Learning

DeepMind's AlphaGo:

- It made the headlines in May 2017 when it beat Ke Jie, the number one ranked player in the world at the time, at the game of Go
- Deep Learning and Reinforcement Learning: AlphaGo combines deep neural networks with reinforcement learning techniques to handle the complex game of Go.
- It learned its winning policy by analyzing millions of games, and then playing many games against itself

Reinforcement Learning

DeepMind's AlphaGo:

- AlphaGo is built on the principles of both deep learning and reinforcement learning. It uses deep neural networks to evaluate Go board positions and predict moves. These neural networks act as function approximators within the reinforcement learning framework, enabling the system to handle the extremely high dimensionality of Go
- Training with Human Expert Games (Supervised Learning): Initially, AlphaGo was trained on a large dataset of expert human games. This phase used supervised learning to teach the neural networks to predict human expert moves.

Reinforcement Learning

DeepMind's AlphaGo:

- Policy and Value Networks: AlphaGo uses two main neural networks — a policy network that selects the next move and a value network that predicts the winner of the game from each position. These networks reduce the breadth and depth of the search tree compared to traditional methods.
- Self-Play and Reinforcement Learning: After the supervised learning phase, AlphaGo improved its capabilities through reinforcement learning, particularly by playing games against itself (self-play). In this phase, AlphaGo continually updated its neural networks based on the outcome of each self-play game, learning novel strategies and refining its understanding of Go strategy beyond what it learned from human games.

Reinforcement Learning

DeepMind's AlphaGo:

- Monte Carlo Tree Search (MCTS): Alongside the neural networks, AlphaGo used a Monte Carlo Tree Search algorithm to explore possible future moves in a more structured way. MCTS helped in efficiently searching the vast space of possible Go moves, guided by the policy and value networks.
- AlphaGo Zero and Further Advancements: Subsequent versions, like AlphaGo Zero, took this further by relying solely on reinforcement learning without the need for human game data. AlphaGo Zero only used the rules of Go for its training, learning from scratch by playing games against itself, significantly outperforming the original AlphaGo.

Reinforcement Learning

Teaching robots how to walk:

- Reinforcement Learning (RL) is a crucial technique in teaching robots how to walk, a task often referred to as robotic locomotion. This process involves training the robot to make sequences of decisions that enable it to move effectively, which can be especially challenging given the complexity of physical movements and balance.

Reinforcement Learning

How RL is typically used in teaching robots how to walk :

- Modeling the Environment: In RL, the environment is where the robot operates. For robotic walking, this can be a simulation or a real-world setting where the robot needs to navigate.
- Defining the Agent: The robot is the agent in RL. The agent interacts with the environment and learns from these interactions.
- Action Space: This refers to the set of all possible movements the robot can make. In walking, these could be different ways of moving its legs, adjusting its balance, etc.

Reinforcement Learning

How RL is typically used in teaching robots how to walk :

- Modeling the Environment: In RL, the environment is where the robot operates. For robotic walking, this can be a simulation or a real-world setting where the robot needs to navigate.
- Defining the Agent: The robot is the agent in RL. The agent interacts with the environment and learns from these interactions.
- Action Space: This refers to the set of all possible movements the robot can make. In walking, these could be different ways of moving its legs, adjusting its balance, etc.

Reinforcement Learning

How RL is typically used in teaching robots how to walk :

- **State Space**: The state space includes all possible scenarios the robot might encounter. For walking, this includes the robot's position, orientation, speed, and the terrain it's walking on.
- **Reward Function**: Crucial in RL, the reward function provides feedback to the robot. For walking, a simple reward could be positive values for moving forward without falling and negative values for stumbling or falling.
- **Learning Algorithm**: This is where the robot uses RL algorithms (like Q-learning, Deep Q Networks, or Proximal Policy Optimization) to learn the best actions to take in different states to maximize its cumulative reward..

Reinforcement Learning

How RL is typically used in teaching robots how to walk :

- **Exploration vs. Exploitation**: The robot must balance between exploring new actions to find better strategies (exploration) and using known strategies that yield good rewards (exploitation).
- **Training and Testing**: Initially, the robot is trained in a controlled environment. Once it achieves a satisfactory level of performance, it can be tested in more varied and complex environments.
- **Challenges**: RL in robotic walking faces challenges like ensuring stability, dealing with the high dimensionality of movements, and transferring learning from simulations to real-world scenarios.