Deep Reinforcement Learning

236504 : Anushree Abhay Bhuskute
236508 : Ekta Mohan Satpute
236523 : Sonali Mahesh Gawade

Table of contents

01 Definition

Reinforcement learning

+

Deep neural networks

Solve multi-level problems by trial and error

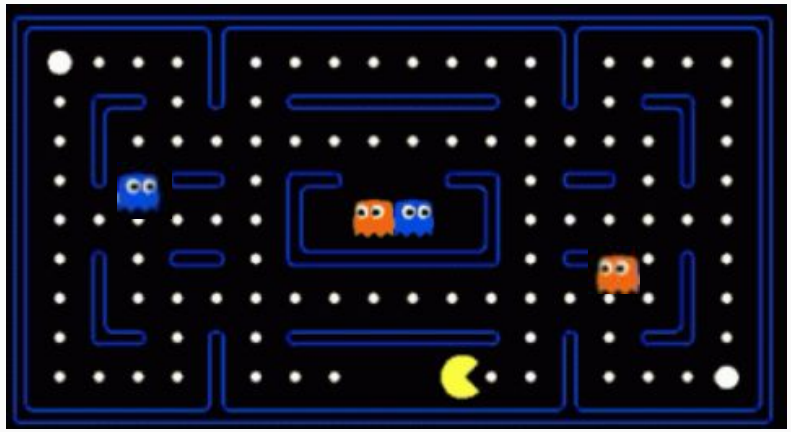
+

Combines multiple layer of ANN to replicate human brain

02 Example

Let’s understand with an example

PAC-MAN game



Components

1 State

Position of PAC-MAN, GHOST and COINS

2 Actions

Can move UP, DOWN, FRONT, BACK

3 Environment

Positive reward : when a coin is collected
Negative reward : when encountered with a ghost

03 Core Components

State

- Represents the current situation
- Agent bases its actions & decisions on the state

Action

- A decision made by the agent
- Alters the state of the system
- Guided by the agent's policy

Agent

- Decision maker
- Acts acc to its policy
- Gains experience over time

Environment

- The system outside of the agent
- Provides feedback to the agent (reward or punishment)

Policy

- Guides the agent's decisions by mapping states to actions
- Aims to find the optimal strategy

Value Function

- Estimates the expected reward from a state under a policy
- Helps evaluate and compare states and policies

Model

- Represents the environment's dynamics
- Simulates outcomes of actions for planning and prediction

Exploration - Exploitation Strategy

- Explores new actions to gain knowledge
- Exploited known actions to achieve immediate rewards

Learning Algorithm

- The process by which agent modifies its value function or policy.
- Various algorithms include Q-learning, policy gradient, etc.

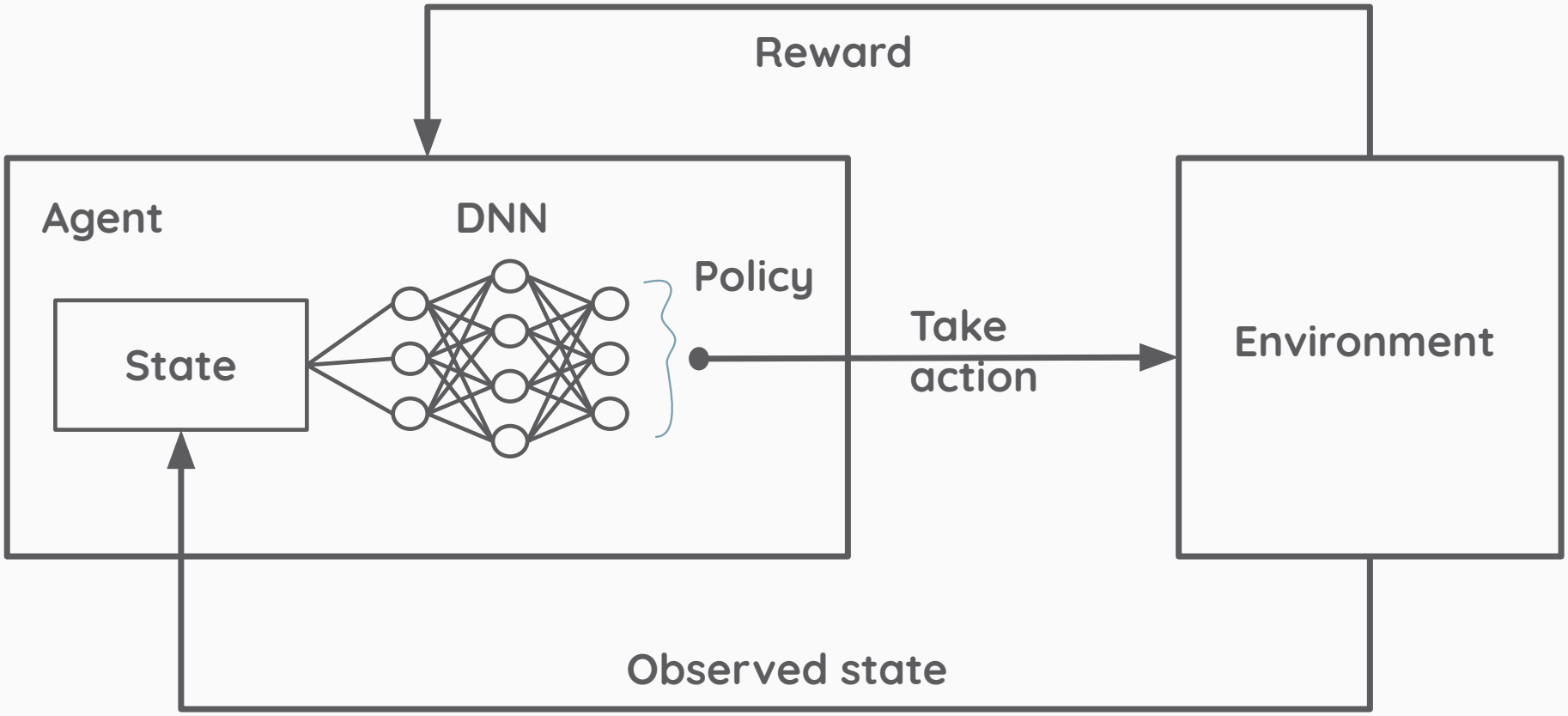
DNN

- Act as function approximators in DRL
- Handle high-dimensional state and action spaces
- Learn complex mappings from inputs to outputs

Experience Replay

- Randomly samples past experiences during training
- Experiences include state, action, reward and next state
- Enhances learning stability by reducing event correlations

04 DRL Architecture

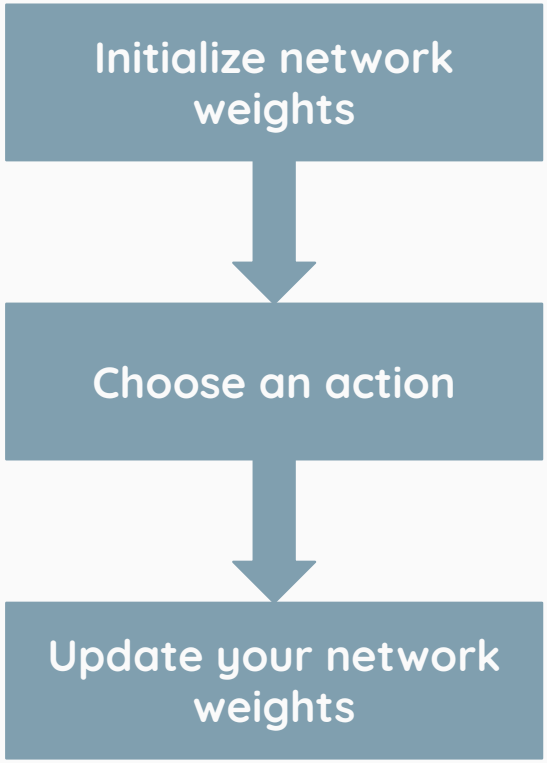


DRL Architecture

05 DRL Algorithms

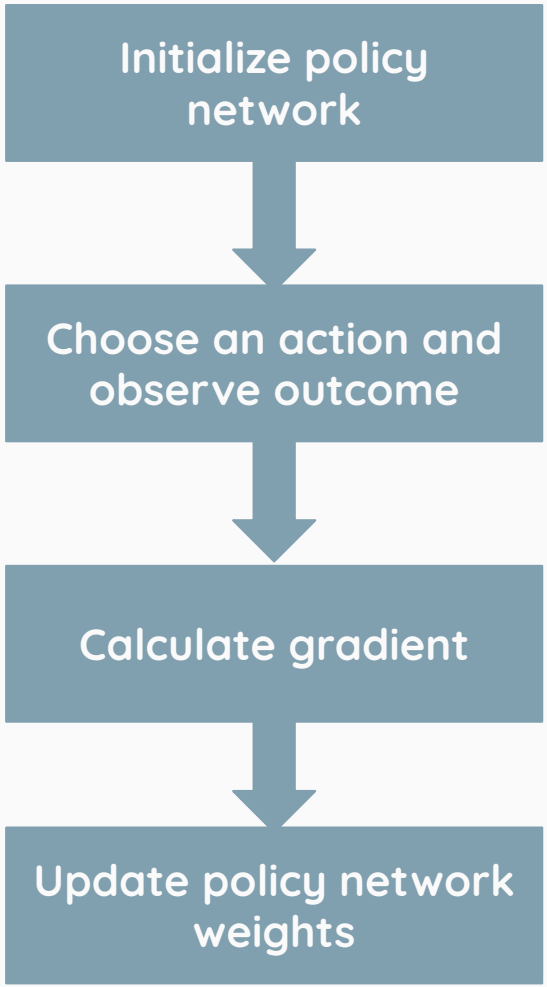
Deep Q Learning

- Q-Learning is a technique where an agent learns to take the best actions in different situations (or states) to maximize rewards.
- Contains a table having states as rows and actions as columns.
- Deep Q learning replaces the Q table to a neural network, which approximates Q-value for different actions given in each state.

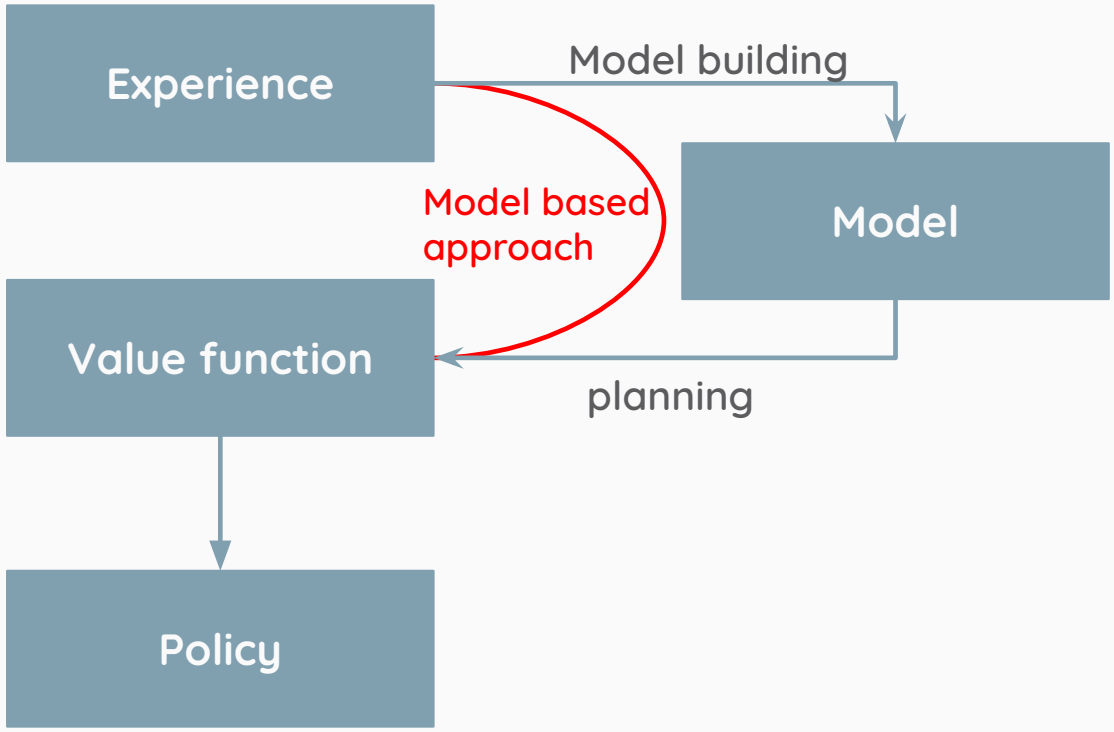


Policy Gradient Methods

- Goal is to directly learn the best strategy (policy) for an agent to follow.
- Instead of estimating the value of actions (like in Q-learning), policy gradient methods directly adjust the policy to maximize the expected reward.
- It uses the gradient (direction) of the rewards to make these adjustments.



Model-based vs Model-free algorithms

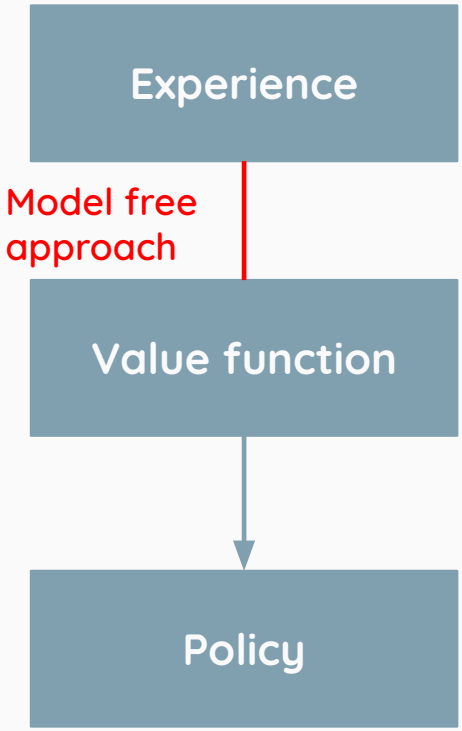


1

Model-based algorithm

- Used when we have complete knowledge of environment and how it reacts to different actions.
- Allow the reinforcement learning agent to plan ahead by thinking ahead.
- For static/fixed environments,Model-based Reinforcement Learning is more suitable.

Model-based vs Model-free algorithms



2

Model-free algorithm

- Estimate the optimal policy directly from experience i.e., interaction between agent and environment without having any hint of the reward function.
- Applied in scenarios involving incomplete information of the environment.
- Used in real-world dynamic environment.

06 Applications

Applications

Entertainment and Gaming

- Used to develop intelligent, realistic game AI
- Example: DRL has mastered games like chess, Go and Dota 2

Robotics and autonomous systems

- Robots pick up skills like navigation, object identification
- Development of autonomous vehicles, drones, etc.

Healthcare and medicine

- Used to develop individualised treatment plans, analyse medical images, identify diseases and perform robotically assisted procedures.

Thank You!!!