

Bab 18: Reinforcement Learning

Tujuan Bab

Bab ini memperkenalkan konsep dasar Reinforcement Learning (RL), yaitu pendekatan pembelajaran mesin di mana agen belajar membuat keputusan melalui interaksi dengan lingkungan untuk memaksimalkan reward jangka panjang.

Konsep Utama

1. Reinforcement Learning Framework

- Agen: Entitas pembelajar (misalnya, robot, bot game)
- Lingkungan (Environment): Dunia tempat agen beroperasi
- Tindakan (Action): Keputusan yang dapat diambil oleh agen
- Keadaan (State): Representasi lingkungan saat ini
- Reward: Nilai numerik yang menunjukkan seberapa baik suatu tindakan
- Kebijakan (Policy): Strategi untuk memilih tindakan berdasarkan keadaan

2. OpenAI Gym

OpenAI Gym menyediakan lingkungan RL standar untuk eksperimen.

Contoh:

```
import gym  
  
env = gym.make("CartPole-v1")  
  
state = env.reset()
```

3. Q-Learning

- Teknik RL yang menyimpan nilai-nilai Q (state-action value) dalam tabel
- Menggunakan persamaan Bellman:

$$Q(s, a) \leftarrow Q(s, a) + \alpha [r + \gamma \max_{a'} Q(s', a') - Q(s, a)]$$

- Cocok untuk lingkungan diskret dan kecil

4. Deep Q-Network (DQN)

- Menggunakan neural network sebagai aproksimasi fungsi Q
- Arsitektur:
 - Input: state
 - Output: Q-value untuk semua aksi
- Melatih dengan experience replay:
 - Simpan pengalaman (state, action, reward, next_state)
 - Sampling batch secara acak dari buffer replay untuk pelatihan
- Target network digunakan untuk stabilitas (diperbarui secara berkala)

5. Strategi Eksplorasi

- Epsilon-Greedy: Dengan probabilitas ϵ , pilih aksi acak; sisanya, pilih aksi terbaik.
- Epsilon menurun seiring waktu (explore \rightarrow exploit)

6. Tantangan RL

- Kurangnya data terlabel langsung (belajar dari trial and error)
- Lingkungan bisa stokastik (hasil dari aksi tidak selalu sama)
- Trade-off antara eksplorasi dan eksploitasi

Isi Colab Notebook:

1. CartPole dengan Q-Table

- Implementasi sederhana Q-learning berbasis tabel
- Visualisasi hasil pelatihan

2. CartPole dengan DQN

- Bangun jaringan neural Q-Network menggunakan Keras
- Implementasi buffer replay
- Pelatihan DQN untuk menyetabilkan tiang di atas kereta selama mungkin
- Visualisasi metrik reward selama pelatihan

Inti Pelajaran

Konsep	Penjelasan
RL	Agen belajar dari trial and error melalui reward
Q-Learning	Menyimpan nilai aksi dalam tabel
DQN	Neural network yang memetakan state ke Q-values
Replay Buffer	Mengurangi korelasi temporal dengan pengalaman acak
Epsilon-Greedy	Kombinasi eksplorasi dan eksploitasi
Gym	Platform simulasi lingkungan RL

Kesimpulan

Bab ini menunjukkan bahwa RL adalah bidang yang sangat berbeda dari supervised learning. Tidak ada label eksplisit, hanya feedback dari reward. DQN memanfaatkan deep learning untuk menyelesaikan tugas-tugas RL kompleks seperti menjaga tiang tetap tegak di CartPole. Teknik seperti experience replay dan target network penting untuk stabilitas pelatihan.