**RESUME REINFORCEMENT LEARNING**

1. Pengertian Reinforcement Learning

Reinforcement Learning berbeda berbeda dengan supervised maupun unsupervised learning. Algoritma ini dimaksudkan untuk membuat komputer dapat belajar sendiri dari lingkungan (environtment) melalui sebuah agent. Jadi komputer akan melakukan pencarian sendiri (self discovery) dengan cara berinteraksi dengan environment.

Reinforcement-learning didefinisikan sebagai metode machine learning yang berkaitan dengan bagaimana agent perangkat lunak harus mengambil action di dalam environment. RL adalah bagian dari metode deep learning yang membantu Anda memaksimalkan sebagian dari reward kumulatif.

1. Karakteristik Reinforcement Learning

* Tidak ada supervisor, hanya ada bilangan real atau reward signal
* Pengambilan keputusan berurutan
* Waktu memainkan peran penting dalam masalah reinforcement
* Feedback balik selalu tertunda, tidak seketika
* Action dari sebuah agent menentukan data selanjutnya yang diterimanya

1. Kapan menggunakan Reinforcement Learning

* Untuk membantu menemukan situasi mana yang membutuhkan tindakan
* Membantu menemukan action mana yang menghasilkan reward tertinggi selama periode yang lebih lama.
* RL juga menyediakan fungsi reward bagi agent pembelajaran.
* RL memungkinkan untuk mengetahui metode terbaik untuk mendapatkan reward besar.
* Kita tidak dapat menerapkan RL dalam semua kasus. Berikut adalah beberapa kondisi ketika kita sebaiknya tidak menggunakan model reinforcement learning.
* Memiliki cukup data untuk menyelesaikan masalah dengan metode supervised learning
* Kita perlu ingat bahwa RL membutuhkan banyak komputasi dan memakan waktu terlbih jika ruang action nya besar.

1. Istilah dalam Reinforcement Learning

* Agent : Sebuah entitas yang diasumsikan melakukan aksi (action) di environment untuk mendapatkan beberapa reward.
* Environment (e) : Skenario yang harus dihadapi agent.
* Reward (R) : Feedback langsung yang diberikan kepada agent ketika dia melakukan action atau tugas tertentu.
* State (s) : Keadaan mengacu pada situasi saat ini yang dikembalikan oleh environment.
* Plicy (π) : Ini adalah strategi yang diterapkan oleh agent untuk memutuskan action selanjutnya berdasarkan state saat ini.
* Value (V) : Diharapkan feedback jangka panjang dengan diskon, dibandingkan dengan feedback jangka pendek.
* Value Function : Ini menentukan nilai state yang merupakan jumlah total reward.
* Environtment Model : Ini meniru perilaku lingkungan. Ini membantu kita membuat kesimpulan yang akan dibuat dan juga menentukan bagaimana environment akan berperilaku.
* Model based : Merupakan metode pemecahan masalah RL yang menggunakan metode berbasis model.
* Q Value / Action Value (Q) : Q value sangat mirip dengan Value. Satu-satunya perbedaan antara keduanya adalah bahwa dibutuhkan parameter tambahan sebagai tindakan saat ini.

1. Cara Kerja Reinforcement Learning

Selama proses training, komputer dituntun oleh algoritma untuk melakukan kegiatan trial and error, mirip seperti anak kecil yang belajar berjalan. Setiap kali percobaan trial and error dilakukan akan ada feedback untuk komputer. Feedback dari aksi (action) sebelumnya akan digunakan sebagai panduan sekaligus peta (guide and mapping) untuk melakukan aksi selanjutnya.

Mari kita lihat contoh sederhana yang membantu menggambarkan cara kerja RL berupa skenario mengajarkan trik baru kepada seekor kucing

Karena kucing tidak mengerti bahasa bahasa manusia, kita tidak dapat memberi tahu kucing secara langsung apa yang harus dilakukan. Sebaliknya, kita akan mengikuti strategi yang berbeda.

Kita meniru situasi, dan kucing mencoba merespons dengan berbagai cara. Jika respon kucing sesuai yang diinginkan, kita akan memberikan ikannya.

Sekarang setiap kali kucing dihadapkan pada situasi yang sama, kucing tersebut melakukan tindakan serupa dengan lebih antusias dengan harapan mendapatkan lebih banyak reward (makanan).

Ini seperti belajar bahwa kucing mendapat “apa yang harus dilakukan” dari pengalaman positif.

Pada saat yang sama, kucing juga belajar apa yang tidak boleh dilakukan saat dihadapkan pada pengalaman negatif.

1. Model Pembelajaran Reinforcement Learning

Ada dua model pembelajaran penting dalam reinforcement learning yaitu:

1. Markov Decision Process

Markov Decision Process lebih dikenal dengan MDP adalah suatu pendekatan dalam RL untuk mengambil keputusan dalam environment gridworld. Lingkungan gridworld terdiri dari state dalam bentuk grid.

MDP mencoba menangkap dunia dalam bentuk grid dengan membaginya menjadi state, action, model / model transition, dan reward. Solusi untuk MDP disebut policy dan tujuannya adalah menemukan policy yang optimal untuk tugas MDP tersebut. Oleh karenanya parameter berikut digunakan untuk mendapatkan solusi yang diharapkan:

Set of states -S

Set of actions- A(s), A

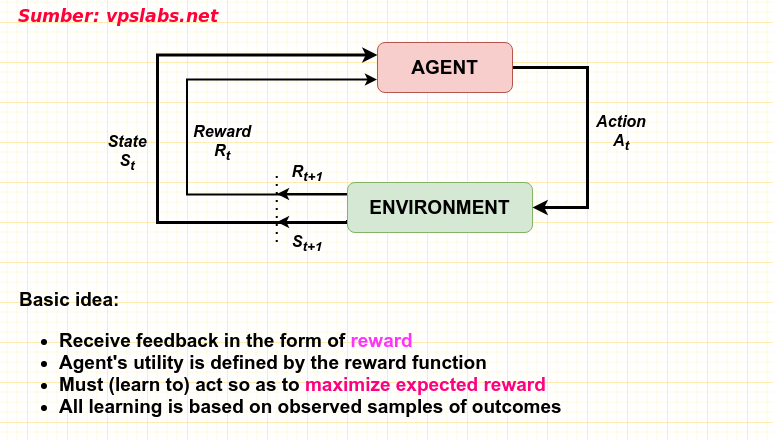
Transition- T(s,a,s’) ~ P(s’|s,a)

Reward- R(s), R(s,a), R(s,a,s’)

Policy- n

Value- V

MDP jika digambarkan kurang lebih seperti ini:

Cara Kerja Reinforcement Learning (MDP)

Algoritma RL akan mencoba berbagai pilihan dan kemungkinan yang berbeda, melakukan pengamatan (observation) dan evaluasi (evaluation) setiap pencapaian. Reinforcement learning dapat belajar dari pengalaman.

Agent di dalam environtment diharuskan mengambil tindakan yang didasarkan pada state saat ini. Jenis pembelajaran ini berbeda dengan supervised learning dalam artian data training pada model sebelumnya memiliki output mapping yang disediakan sedemikian rupa sehingga model mampu mempelajari jawaban yang benar. Sedangkan dalam hal ini RL tidak ada kunci jawaban yang disediakan kepada agent ketika harus melakukan action tertentu. Jika tidak ada set data pelatihan, ia belajar dari pengalamannya sendiri.

1. Q Learning

Q-learning (disebut sebagai model free algorithm) adalah algoritma RL tanpa model untuk mempelajari policy yang memberi tahu agen tindakan apa yang harus diambil dalam keadaan apa. Q-learning tidak memerlukan model dari environtment, dan dapat menangani masalah dengan transisi stokastik dan reward, tanpa memerlukan adaptasi.

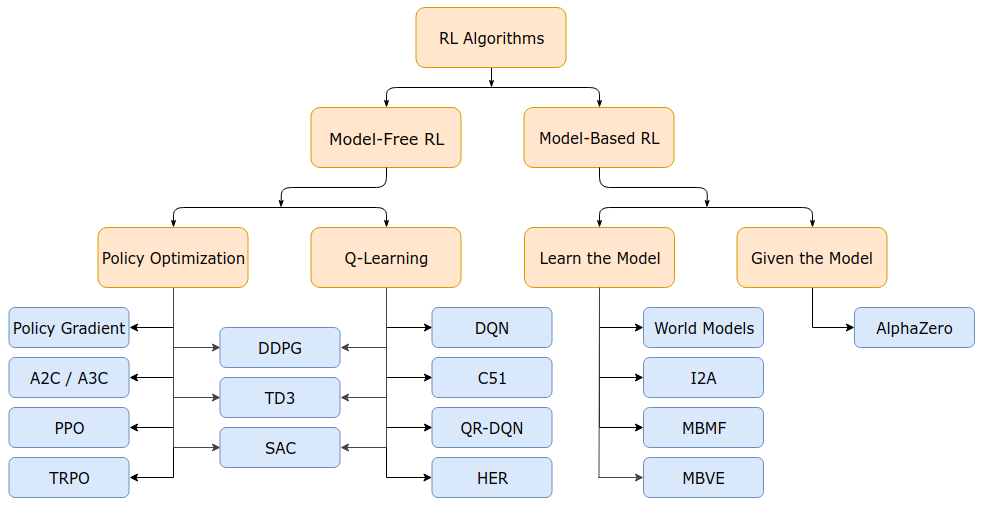
Reinforcement Learning Tidak Perlu Data Set?

Ada berbagai dokumentasi yang menyebutkan bahwa reinforcement learning tidak membutuhkan dataset. Pernyataan ini tidak sepenuhnya benar. Bahwa setiap algoritma machine learning memerlukan input untuk dipelajari selama proses training, namun jenis inputnya bisa saja berbeda-beda.

Pada RL tidak ada kunci jawaban yang diberikan kepada agent ketika harus melakukan tugas tertentu. Jika tidak ada set data pelatihan, ia belajar dari pengalamannya sendiri.

1. Algoritma Reinforcement Learning

Algoritma yang termasuk reinforcement learning: Q-Learning, State-Action-Reward-State-Action (SARSA), Deep Q Network (DQN), Deep Deterministic Policy Gradient (DDPG), Actor Critic, Monte Carlo Tree Search (MCTS) [[1]](https://spinningup.openai.com/en/latest/user/introduction.html).

Untuk lebih lengkapnya tentang algoritma RL bisa dilihat pada gambar berikut ini:

1. Penerapan Reinforcement Learning

Google telah mengimplementasikan penerapan reinforcement learning pada sistem Google’s Active Query Answering (AQA) mereka. Jadi chat bot ini akan melakukan formulasi ulang atas pertanyaan yang diketikkan oleh pengguna.

Sebagai contoh, jika anda menanyakan pertanyaan “Kapan hari kemerdekaan RI” maka AQA ajab mereformulasi pertanyaan tersebut menjadi beberapa pertanyaan berbeda misalnya “Tanggal berapa hari kemerdekaan RI”, “Kapan HUT RI”, “Ulang tahun Indonesia”, dll. Proses reformulasi ini telah mengutilisasi model sequence to sequence, tetapi Google telah mengintegrasikan reinforcement learning agar pengguna dapat berinteraksi dengan sistem menjadi lebih baik.

1. Area penerapan reinforcement learning meliputi:

* Robotika untuk otomasi industri.
* Perencanaan strategi bisnis
* Pembelajaran mesin dan pemrosesan data
* Membuat sistem pelatihan yang menyediakan instruksi dan materi khusus sesuai dengan kebutuhan siswa.
* Kontrol pesawat dan kontrol gerak robot

1. Pendekatan Implementasi

Ada tiga pendekatan untuk mengimplementasikan algoritma Reinforcement Learning (RL) yaitu:

1. Value Based

Dalam metode RL berbasis nilai (value based), Anda harus mencoba memaksimalkan fungsi nilai V (s). Dalam metode ini, agen mengharapkan pengembalian jangka panjang dari keadaan saat ini berdasarkan policy π.

1. Policy Based

Dalam metode RL berbasis policy, Anda mencoba menghasilkan aturan sedemikian rupa sehingga action yang dilakukan di setiap state membantu Anda mendapatkan reward maksimum di masa mendatang. Dua jenis metode berbasis policy based adalah Deterministik dan Stochastic.

1. Model Based

Dalam metode RL ini, Anda perlu membuat model virtual untuk setiap environtment. Agent belajar untuk bekerja di environment spesifik tersebut.

1. Metode monte carlo

Metode Monte Carlo adalah [algoritme](https://id.wikipedia.org/wiki/Algoritme" \o "Algoritme) [komputasi](https://id.wikipedia.org/wiki/Komputasi" \o "Komputasi) untuk [mensimulasikan](https://id.wikipedia.org/wiki/Simulasi" \o "Simulasi) berbagai perilaku sistem fisika dan matematika. Penggunaan klasik metode ini adalah untuk mengevaluasi [integral definit](https://id.wikipedia.org/w/index.php?title=Integral_definit&action=edit&redlink=1), terutama integral multidimensi dengan syarat dan batasan yang rumit.

Konsep Monte Carlo (MC)

* MC tidak mengambil pengetahuan lengkap dari environment
* MC belajar dari experience, episode per episode, baik itu experience aktual, maupun simulasi
* MC belajar dari episode-episode secara utuh dan independent, tidak bootstrapping.

Konsep Monte Carlo (MC) cont’d

* MC didefinisikan untuk jenis episodic environment
* Ide utama dari MC: Value didapatkan dari ratarata returns
* Dengan semakin banyak returns, nilai rata-ratanya diharapkan konvergen pada expected value.