

REINFORCEMENT LEARNING

Team Teaching

PENDAHULUAN

- ***Reinforcement learning*** merupakan ilmu pengambilan keputusan dengan mempelajari perilaku optimal dalam sebuah lingkungan untuk mendapatkan *reward* yang maksimal.
- ***Reinforcement learning*** merupakan teknik pembelajaran mesin (*machine learning*) yang memungkinkan agen (subjek yang melakukan tindakan) untuk belajar dalam sebuah lingkungan yang interaktif melalui sebuah sistem *trial and error* dari *feedback* yang didapatkan dari tindakan dan pengalaman mereka sendiri.

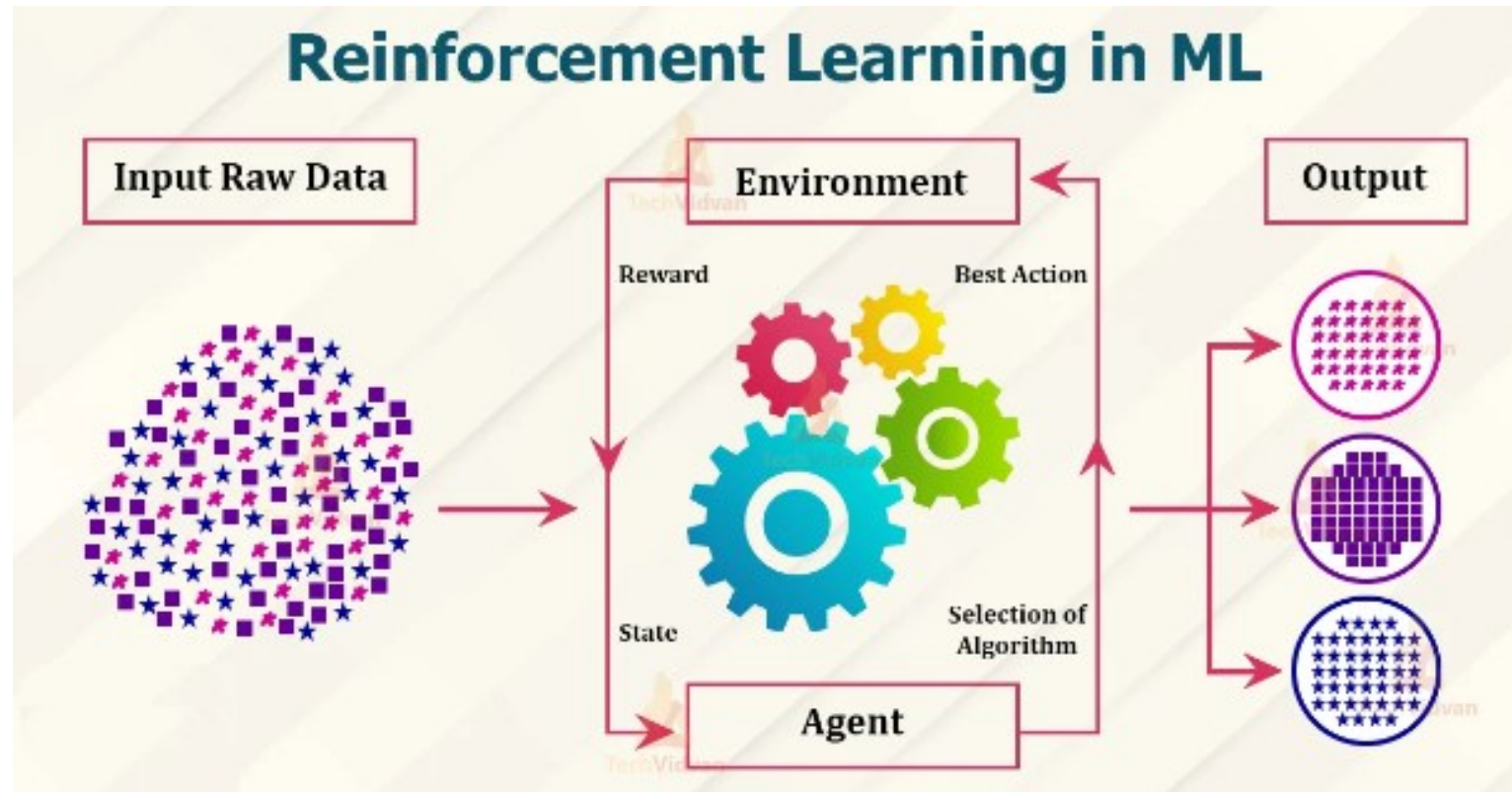
PENDAHULUAN

- ***Reinforcement learning*** menggunakan *reward* dan *punishment* sebagai sinyal untuk mengambil keputusan atau tindakan berikutnya.
- Keterlibatan manusia pada reinforcement learning terbatas pada perubahan lingkungan dan menyempurnakan sistem *rewards* dan *punishment*.

PENDAHULUAN

- Ketika komputer berusaha untuk memaksimalkan *rewards*, ia cenderung menemukan cara baru untuk mencapainya.
- Keterlibatan manusia ditujukan untuk mencegah komputer menyalahgunakan sistem dan menginspirasi untuk melakukan pekerjaan sebagaimana dimaksud.

PENDAHULUAN



ISTILAH PADA *REINFORCEMENT LEARNING*

- **Agent:** Entitas yang dapat melihat/menjelajahi lingkungan dan bertindak berdasarkan itu.
- **Environment:** Lingkungan atau situasi di mana agen hadir atau dikelilingi olehnya. Di RL, kita mengasumsikan lingkungan bersifat stokastik atau acak.
- **Action:** Action adalah gerakan yang diambil oleh agen dalam lingkungan.

ISTILAH PADA *REINFORCEMENT LEARNING*

State: State adalah situasi yang dikembalikan oleh lingkungan setelah setiap tindakan yang dilakukan oleh agen.

Reward: Umpan balik yang dikembalikan ke agen dari lingkungan untuk mengevaluasi tindakan agen.

Policy: Policy adalah strategi yang diterapkan oleh agen untuk tindakan selanjutnya berdasarkan keadaan saat ini.

ISTILAH PADA *REINFORCEMENT LEARNING*

Value: Imbalan jangka panjang yang diharapkan dengan faktor diskon dan kebalikan dari imbalan jangka pendek.

Q-value: Q-value sebagian besar mirip dengan value, tetapi dibutuhkan satu parameter tambahan sebagai tindakan saat ini (a).

KOMPONEN RL

- ***Policy*** (kebijakan)

Kebijakan (*policy*) adalah perilaku agen dalam waktu tertentu. Perilaku tersebut mencerminkan pengaruh kondisi lingkungan terhadap perilaku yang muncul. Kebijakan merupakan komponen inti *reinforcement learning* karena dapat mempengaruhi perilaku agen secara langsung.

KOMPONEN RL

- ***Reward signal*** (sinyal hadiah)

Reward signal adalah hadiah yang mungkin didapatkan adalah tujuan dari *reinforcement learning*. Dalam setiap kondisi, lingkungan akan mengirimkan *reward signal* secara langsung kepada agen. *Reward* yang akan didapatkan tentu saja ditentukan oleh perilaku agen itu sendiri. Tujuan utama agen dalam melakukan *reinforcement learning* adalah memperbanyak *reward* yang didapatkan.

KOMPONEN RL

- ***Value function*** (fungsi nilai)

Fungsi nilai adalah sumber informasi mengenai seberapa baik suatu situasi dan tindakan, serta berapa banyak *reward* yang mungkin didapatkan agen. Fungsi nilai sangat bergantung pada *reward* atau hadiah karena tanpa hadiah, sebuah tindakan atau situasi tidak akan memiliki nilai.

KOMPONEN RL

- ***Model of the environment*** (model lingkungan)

Model lingkungan adalah elemen terakhir, yang berarti bagaimana agen berperilaku sesuai dengan lingkungan mereka. Hal ini erat kaitannya dengan kemampuan agen dalam memprediksi kondisi lingkungan untuk merencanakan tindakan yang akan mereka lakukan nantinya.

KARAKTERISTIK RL

- Agen tidak mendapatkan intruksi mengenai lingkungan dan tindakan serta perilaku apa yang harus dilakukan dalam menghadapi lingkungan tersebut.
- *Reinforcement learning* didasarkan pada *trial and error* yang merupakan bagian dari pengalaman agen.
- Agen mengambil tindakan lanjutan dan mengubah situasi tergantung pada *feedback* yang diperoleh dari tindakan sebelumnya.

KARAKTERISTIK RL

- *Reward* yang didapatkan agen bisa saja tertunda.
- Lingkungan dalam *reinforcement learning* bersifat stokastik, sehingga agen perlu melakukan eksplorasi untuk mendapatkan *reward* yang maksimum.

FUNGSI DAN KELEBIHAN *REINFORCEMENT LEARNING* (RL)

1. Fokus pada masalah secara menyeluruh
2. Tidak memerlukan tahapan pengumpulan data terpisah
3. Berlangsung dalam lingkungan dinamis dan tidak pasti

TANTANGAN RL

- Menyiapkan lingkungan simulasi yang sangat bergantung pada pekerjaan yang ada.
- Penskalaan dan penyesuaian jaringan saraf yang mengontrol bot.
 1. Tidak ada metode lain untuk berkomunikasi dengan jaringan selain melalui sistem *reward* dan *punishment*.
 2. Hal ini dapat mengakibatkan risiko kelupaan, di mana pengetahuan baru menyebabkan beberapa pengetahuan lama hilang dari jaringan.

JENIS RL

- **Positive reinforcement**

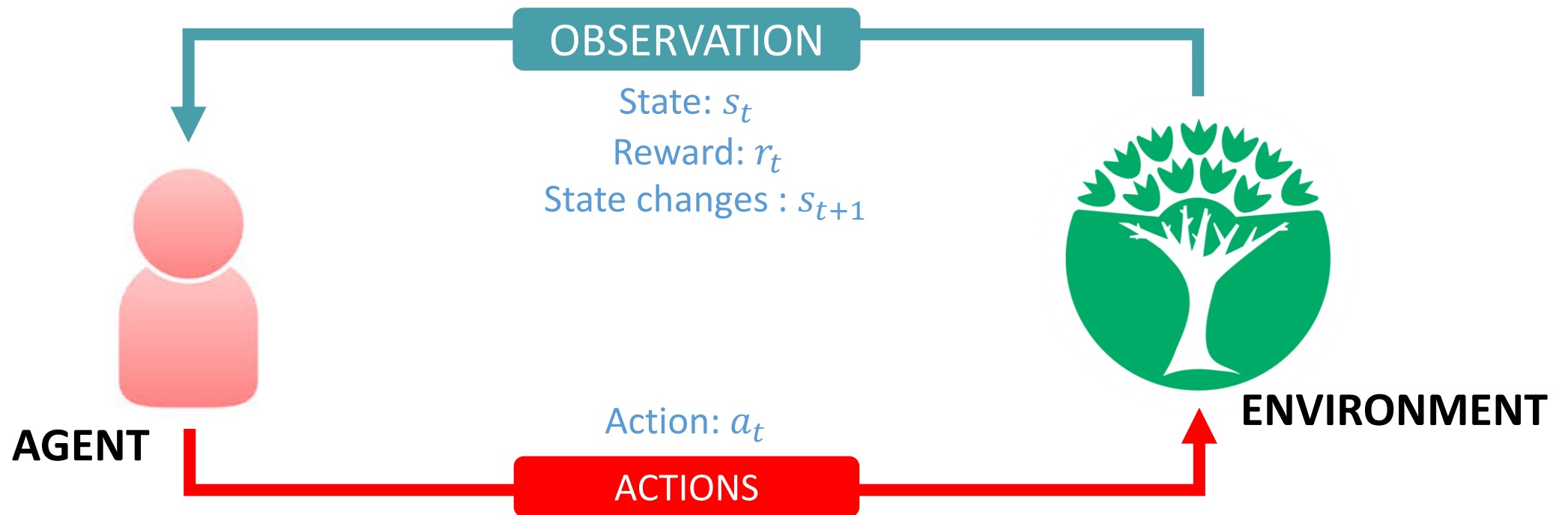
***Positive reinforcement* adalah** konsep yang menyatakan bahwa perilaku yang membawa dampak positif cenderung akan kembali dilakukan oleh agen.

JENIS RL

- **Negative reinforcement**

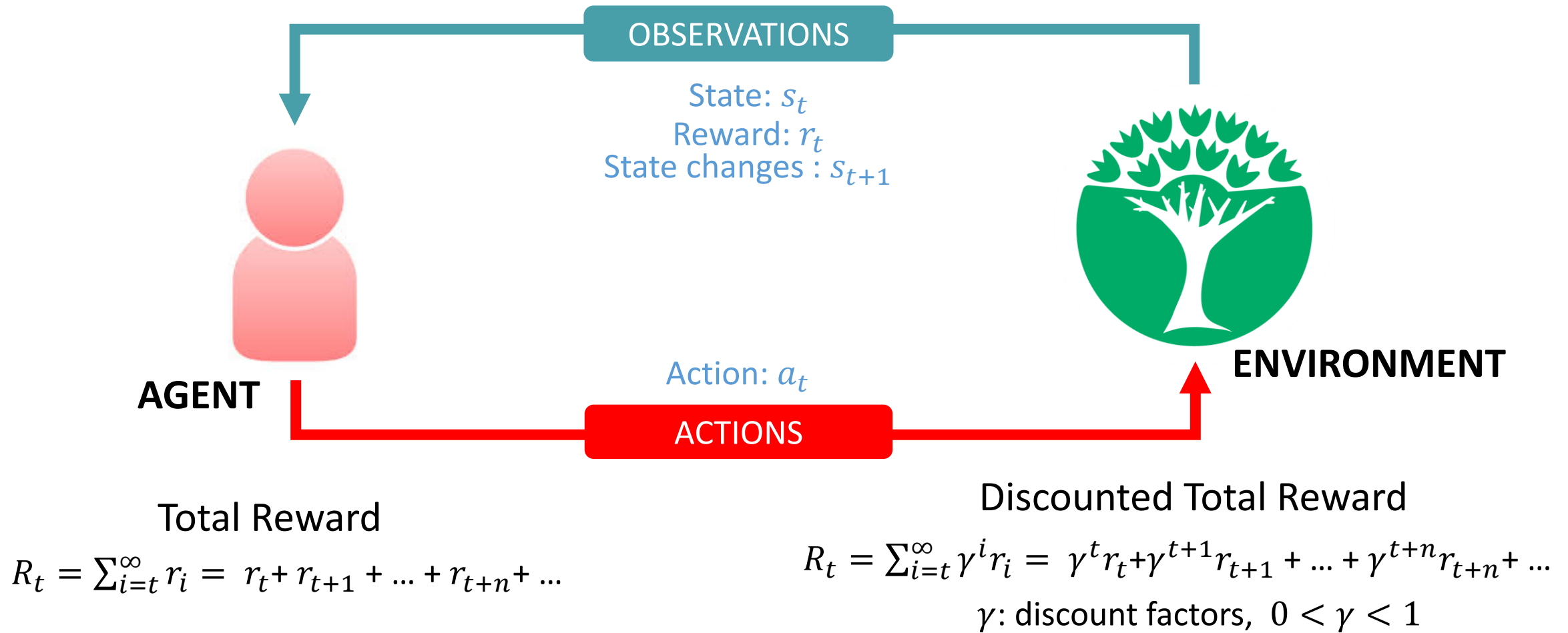
***Negative reinforcement* adalah** konsep yang menyatakan bahwa perilaku yang sama tidak akan diulangi kembali karena memberikan dampak yang cenderung negatif. Tergantung dari situasi dan perilaku agen, tipe negatif ini bisa lebih kuat dibanding positif, namun hanya bisa membantu agen untuk memenuhi persyaratan perilaku yang paling minimum.

Konsep Reinforcement Learning



- s_t : state pada waktu t
- a_t : aksi pada waktu t
- r_t : reward pada waktu t

Konsep Reinforcement Learning



Definisi Q-function

Total Reward, R_t , total semua reward dengan diskon dari waktu t

$$R_t = r_t + \gamma^1 r_{t+1} + \gamma^2 r_{t+2} + \dots$$

γ : discount factors, $0 < \gamma < 1$

Q-function menangkap ***expected total feature reward*** Agen pada state s , yang melakukan aksi a tertentu

$$Q(s_t, a_t) = E[R_t | s_t, a_t]$$

Q-function

- Bagaimana cara Agen memilih aksi dari Q-function?

$$Q(s_t, a_t) = E[R_t | s_t, a_t]$$

Agen memerlukan suatu **policy** $\pi(s)$ untuk memilih aksi terbaik pada state s

- **Strategi:** memilih aksi yang memaksimalkan **future reward**

$$\pi^*(s) = \operatorname{argmax}_a Q(s, a)$$

π^* optimal **policy** π

Pendekatan Reinforcement Learning

- **Policy-based RL**

- Memilih secara langsung *optimal policy* π^*
- Memilih *policy* yang mencapai *maximum future reward*

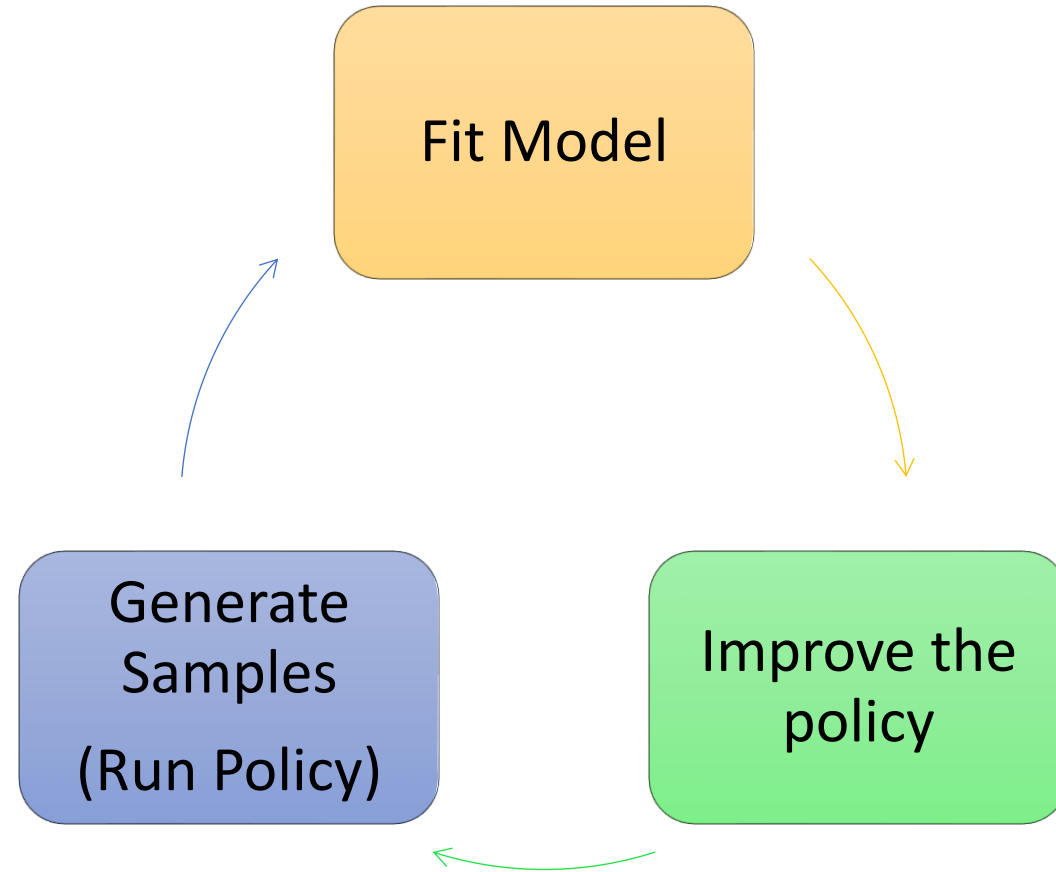
- **Value-based RL**

- Mengestimasi *optimal value function* $Q^*(s,a)$
- Maximum value yang tercapai oleh *policy*

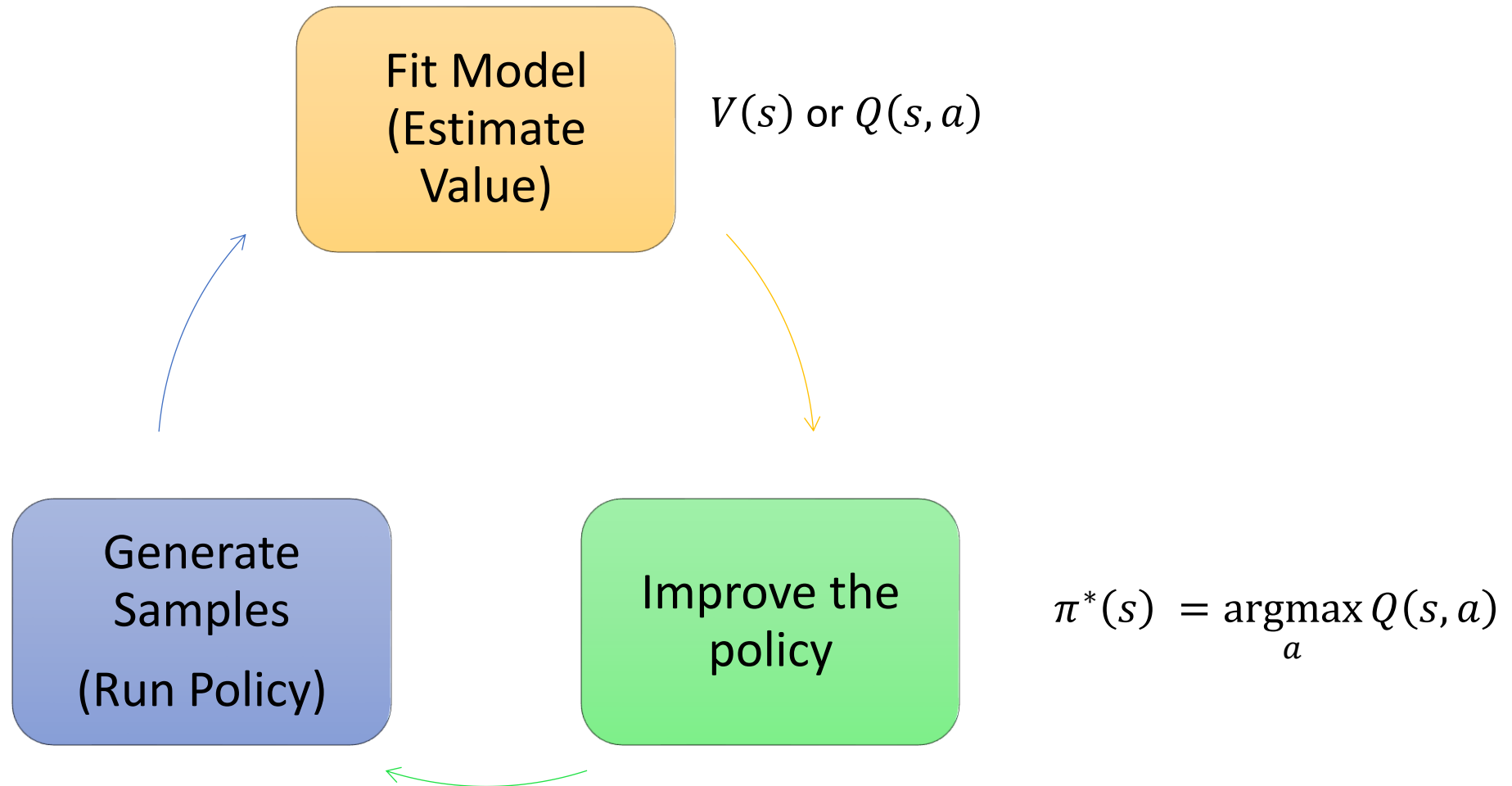
- **Model-based RL**

- Membangun *transition model* pada sebuah *environment*
- Perencanaan berbasis model

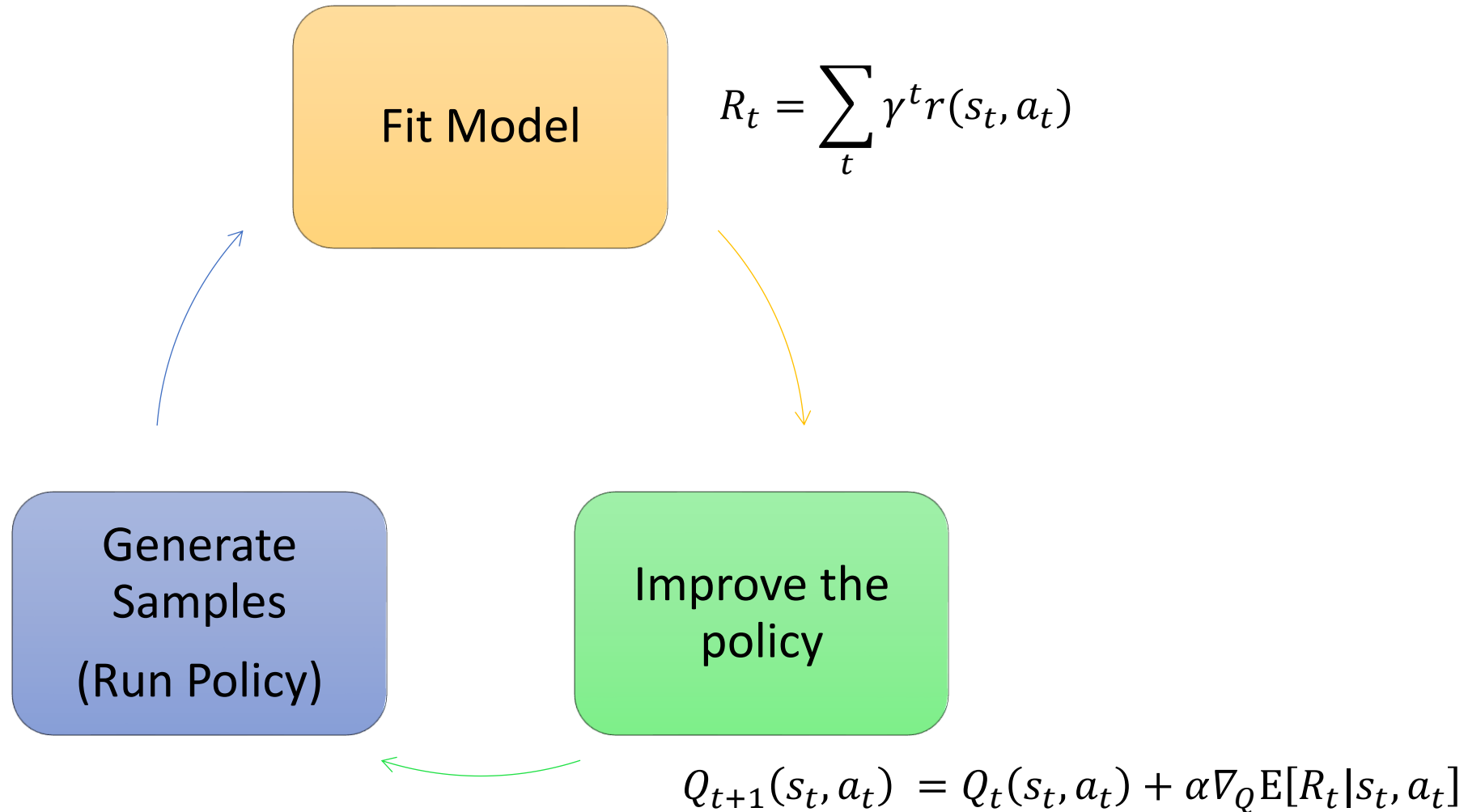
Desain Algoritma Reinforcement Learning



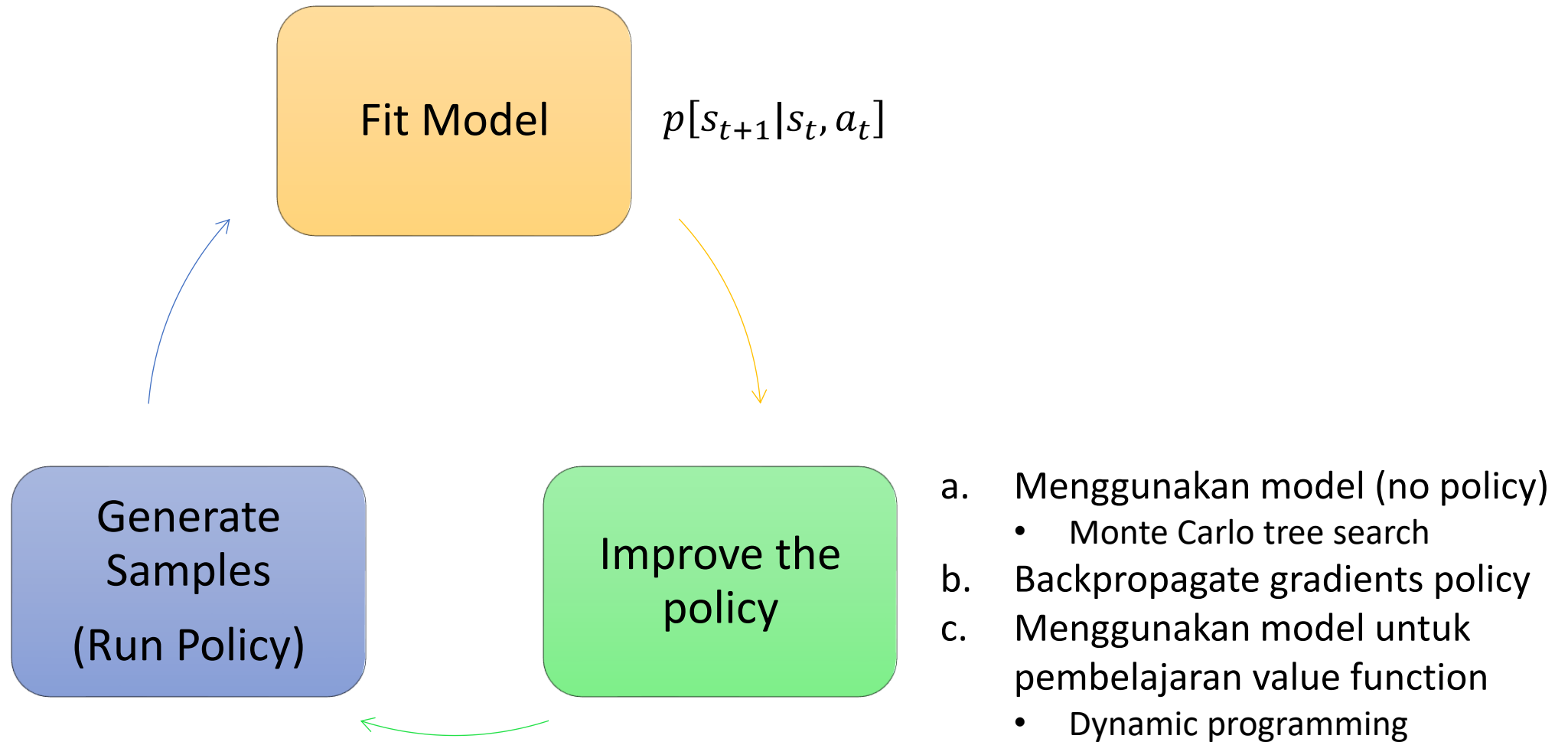
Policy-based Reinforcement Learning



Value-based Reinforcement Learning



Model-based Reinforcement Learning



Q-Learning

Menggunakan ***policy*** untuk mengestimasi Q yang memaksimalkan ***future reward***:

- Aproksimasi Q^* (persamaan *Bellman optimality*)
- Update setiap pasangan (s, a)

$$Q_{t+1}(s_t, a_t) = Q_t(s_t, a_t) + \alpha \left(R_{t+1} + \gamma \max_a Q_t(s_{t+1}, a) - Q_t(s_t, a_t) \right)$$

Diagram illustrating the Q-Learning update equation with labels:

- $Q_{t+1}(s_t, a_t)$: New state
- $Q_t(s_t, a_t)$: Old state
- R_{t+1} : Reward
- α : Learning rate
- γ : Discount factor
- $\max_a Q_t(s_{t+1}, a)$: Maximum Q-value for the next state

Q-Learning: Value Iteration

$$Q_{t+1}(s_t, a_t) = Q_t(s_t, a_t) + \alpha \left(R_{t+1} + \gamma \max_a Q_t(s_{t+1}, a) - Q_t(s_t, a_t) \right)$$

New state

Old state

Reward

	A1	A2	A3	A4
S1	+1	+2	-1	0
S2	+2	0	+1	-2
S3	-1	+1	0	-2
S4	-2	0	+1	+1

```
initialize  $Q[num\_states, num\_actions]$  arbitrarily
observe initial state  $s$ 
repeat
    select and carry out an action  $a$ 
    observe reward  $r$  and new state  $s'$ 
     $Q[s, a] = Q[s, a] + \alpha(r + \gamma \max_{a'} Q[s', a'] - Q[s, a])$ 
     $s = s'$ 
until terminated
```

Sumber: Lex Fridman, Deep Reinforcement Learning, MIT Course 2018 : Introduction to Deep Learning

Kelemahan Q-Learning

- *Value Iteration* tidak praktis:
 - State atau action yang terbatas
 - Tidak dapat menggeneralisir state yang belum diketahui

IMPLEMENTASI RL

- *Mobil Self Driving*

Beberapa kegiatan mengemudi otonom di mana RL dapat digunakan termasuk optimasi lintasan, perencanaan gerak, *pathing* dinamis, pengoptimalan pengontrol, dan aturan pembelajaran berbasis skenario untuk jalan raya.

- *Natural Language Processing (NLP)*

Peringkasan teks, penjawab pertanyaan, dan terjemahan mesin hanyalah beberapa dari aplikasi RL di NLP.

IMPLEMENTASI RL

- Sektor Kesehatan

Kebijakan yang dipelajari menggunakan sistem RL dapat membantu pasien dalam perawatan kesehatan. Contoh RL dalam perawatan kesehatan termasuk rejimen pengobatan dinamis atau *dynamic treatment regimens* (DTR) pada penyakit kronis atau perawatan kritis, diagnostik medis otomatis, dan aplikasi luas lainnya.

- Manipulasi Robotika

RL dapat digunakan untuk mengajar robot untuk memahami berbagai objek, termasuk yang tidak terlihat selama *training*.

IMPLEMENTASI RL

- Otomasi Industri

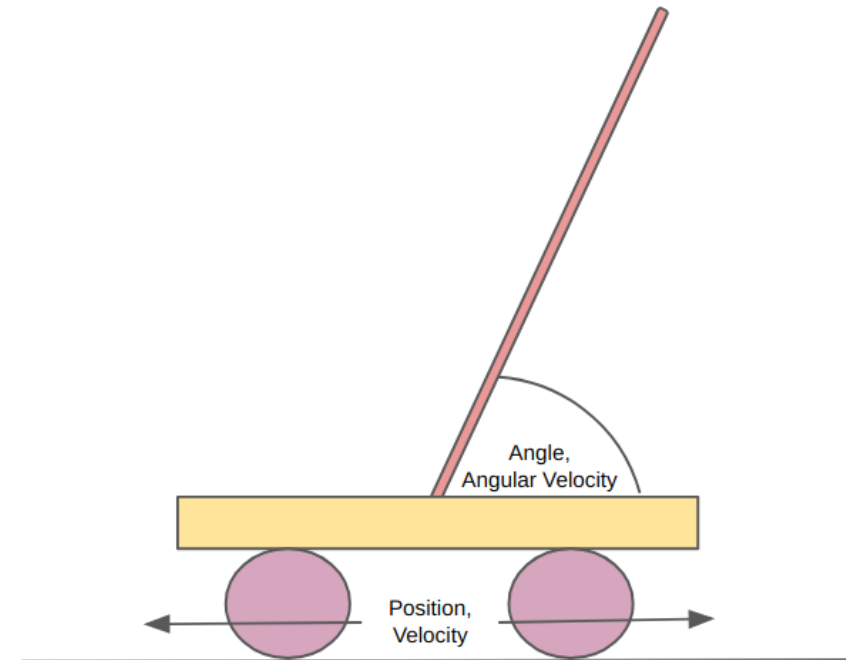
Robot berbasis pembelajaran digunakan untuk melakukan berbagai tugas dalam penguatan industri. Selain lebih efisien dari manusia, robot-robot ini juga dapat melakukan tugas-tugas yang mungkin dianggap tidak aman oleh manusia.

- Permainan Game
- Dan sebagainya

Contoh Reinforcement Learning

- **Cart-Pole Balancing**

- **Goal** - Menyeimbangkan tiang diatas gerobak agar tetap berdiri
- **State** - Sudut tiang, kecepatan sudut, posisi gerobak, kecepatan horizontal
- **Actions** - Gaya horisontal ke gerobak
- **Reward** – setiap step bernilai 1 jika tiang berdiri tegak



Contoh Reinforcement Learning

- **Bin Packing**

- **Goal** - Mengambil barang dalam kotak dan meletakkan ke kontainer
- **State** - Piksel-piksel pada gambar yang tertangkap kamera
- **Actions** – Aksi-aksi yang dilakukan robot, misalkan mengambil atau meletakkan barang
- **Reward** – Bernilai positif jika berhasil menempatkan barang dan bernilai negative jika sebaliknya



TERIMA KASIH