

# **535514: Reinforcement Learning**

## **Lecture 18 – Value-Based Methods**

Ping-Chun Hsieh

April 25, 2024

# On-Policy vs Off-Policy Methods

	<b>Policy Optimization</b>	<b>Value-Based</b>	<b>Model-Based</b>	<b>Imitation-Based</b>
<b>On-Policy</b>	<b>Exact PG</b> <b>REINFORCE (w/i baseline)</b> <b>A2C</b> <b>On-policy DAC</b> <b>TRPO</b> <b>Natural PG (NPG)</b> <b>PPO-KL &amp; PPO-Clip</b> <b>RLHF by PPO-KL</b>	<b>Epsilon-Greedy MC</b> <b>Sarsa</b> <b>Expected Sarsa</b>	<b>Model-Predictive Control (MPC)</b> <b>PETS</b>	<b>IRL</b> <b>GAIL</b> <b>IQ-Learn</b>
<b>Off-Policy</b>	<b>Off-policy DPG &amp; DDPG</b> <b>Twin Delayed DDPG (TD3)</b>	<b>Q-learning</b> <b>Double Q-learning</b> <b>DQN &amp; DDQN</b> <b>C51 / QR-DQN / IQN</b> <b>Soft Actor-Critic (SAC)</b>		

# Recall: PPO-Clip Algorithm

Input: initial policy parameters  $\theta_0$ , clipping threshold  $\epsilon$   
for  $k = 0, 1, 2, \dots$  do

Collect trajectories and  
estimate  $A(s, a)$

Collect set of partial trajectories  $\mathcal{D}_k$  on policy  $\pi_k = \pi(\theta_k)$

Estimate advantages  $\hat{A}_t^{\pi_k}$  using any advantage estimation algorithm

Compute policy update

$$\theta_{k+1} = \arg \max_{\theta} \mathcal{L}_{\theta_k}^{CLIP}(\theta)$$

by taking  $K$  steps of minibatch SGD (via Adam), where

$$\mathcal{L}_{\theta_k}^{CLIP}(\theta) = \mathbb{E}_{\tau \sim \pi_k} \left[ \sum_{t=0}^T \left[ \min(r_t(\theta) \hat{A}_t^{\pi_k}, \text{clip}(r_t(\theta), 1 - \epsilon, 1 + \epsilon) \hat{A}_t^{\pi_k}) \right] \right]$$

end for

PPO does not use 2nd-order derivatives

# Assignment for Lec 17:

- ▶ Spend 30 minutes reading the code of PPO-clip
- ▶ [github.com/ikostrikov/pytorch-a2c-ppo-acktr-gail/](https://github.com/ikostrikov/pytorch-a2c-ppo-acktr-gail/)

- ▶ Could you find any part of the code that we have not discussed in this lecture?

## 1. Value prediction is also clipped for more stable training (just a heuristic)

$$L^V = \max \left[ (V_{\theta_t} - V_{targ})^2, (\text{clip}(V_{\theta_t}, V_{\theta_{t-1}} - \epsilon, V_{\theta_{t-1}} + \epsilon) - V_{targ})^2 \right]$$

<https://stable-baselines.readthedocs.io/en/master/modules/ppo2.html>

## 2. The loss of PPO-clip has three terms: (i) policy loss (ii) value loss (iii) entropy bonus

```
43     for e in range(self.ppo_epoch):
44         if self.actor_critic.is_recurrent:
45             data_generator = rollouts.recurrent_generator(
46                 advantages, self.num_mini_batch)
47         else:
48             data_generator = rollouts.feed_forward_generator(
49                 advantages, self.num_mini_batch)
50
51         for sample in data_generator:
52             obs_batch, recurrent_hidden_states_batch, actions_batch, \
53             value_preds_batch, return_batch, masks_batch, old_action_log_probs_batch, \
54             adv_targ = sample
55
56         # Reshape to do in a single forward pass for all steps
57         values, action_log_probs, dist_entropy, _ = self.actor_critic.evaluate_actions(
58             obs_batch, recurrent_hidden_states_batch, masks_batch,
59             actions_batch)
60
61         ratio = torch.exp(action_log_probs -
62                           old_action_log_probs_batch)
63         surr1 = ratio * adv_targ
64         surr2 = torch.clamp(ratio, 1.0 - self.clip_param,
65                             1.0 + self.clip_param) * adv_targ
66         action_loss = -torch.min(surr1, surr2).mean()
67
68         if self.use_clipped_value_loss:
69             value_pred_clipped = value_preds_batch + \
70                 (values - value_preds_batch).clamp(-self.clip_param, self.clip_param)
71             value_losses = (values - return_batch).pow(2)
72             value_losses_clipped = (
73                 value_pred_clipped - return_batch).pow(2)
74             value_loss = 0.5 * torch.max(value_losses,
75                                         value_losses_clipped).mean()
75
76         else:
77             value_loss = 0.5 * (return_batch - values).pow(2).mean()
78
79         self.optimizer.zero_grad()
80         (value_loss * self.value_loss_coef + action_loss -
81          dist_entropy * self.entropy_coef).backward()
82         nn.utils.clip_grad_norm_(self.actor_critic.parameters(),
83                                 self.max_grad_norm)
84         self.optimizer.step()
85
86         value_loss_epoch += value_loss.item()
87         action_loss_epoch += action_loss.item()
88         dist_entropy_epoch += dist_entropy.item()
89
90         num_updates = self.ppo_epoch * self.num_mini_batch
91
92         value_loss_epoch /= num_updates
93         action_loss_epoch /= num_updates
94         dist_entropy_epoch /= num_updates
95
96         return value_loss_epoch, action_loss_epoch, dist_entropy_epoch
```

# A Nice ICLR Blogpost on PPO-Clip Implementation

The ICLR Blog Track | Blog Posts

---

## The 37 Implementation Details of Proximal Policy Optimization

25 Mar 2022 | [# proximal-policy-optimization](#) [# reproducibility](#) [# reinforcement-learning](#) [# implementation-details](#) [# tutorial](#)

Huang, Shengyi; Dossa, Rousslan Fernand Julien; Raffin, Antonin; Kanervisto, Anssi; Wang, Weixun

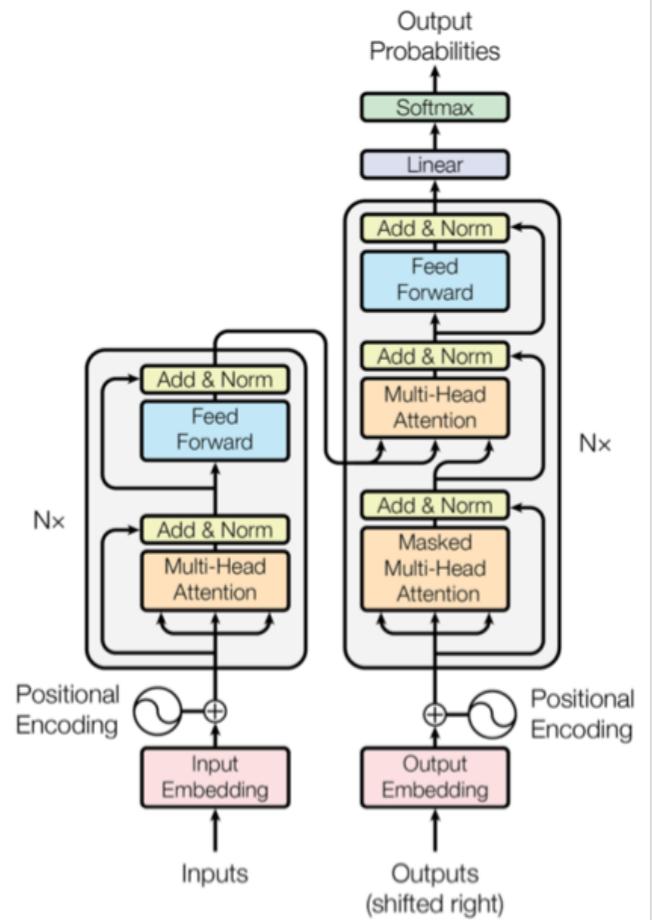
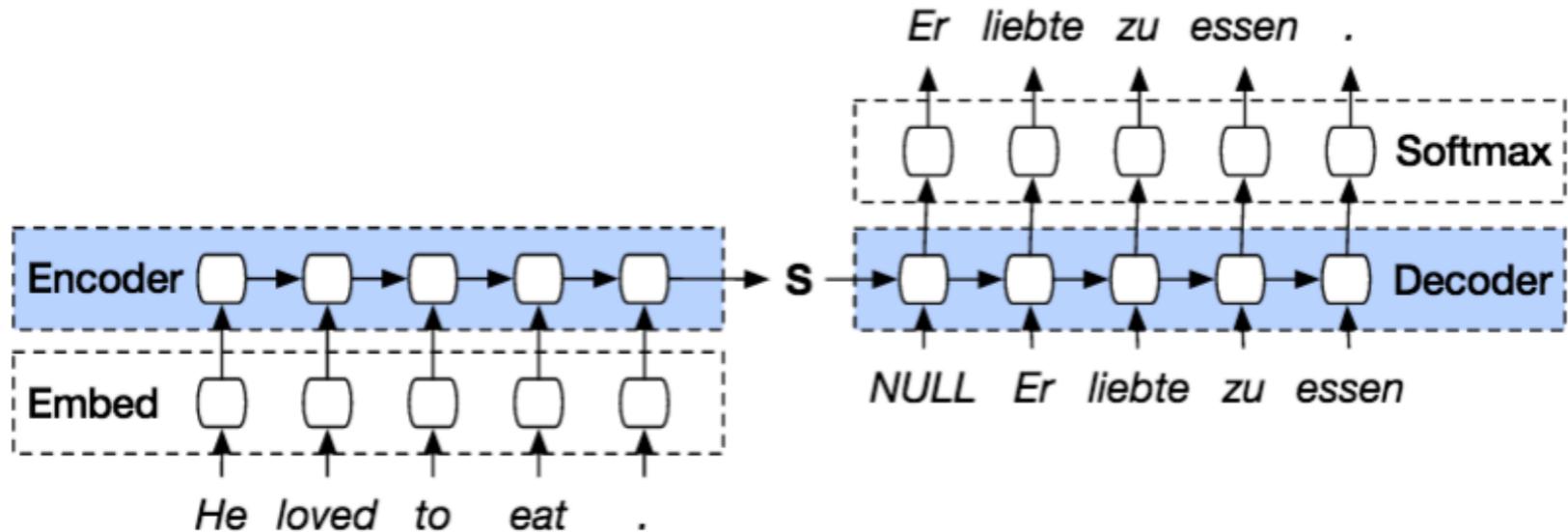
Jon is a first-year master's student who is interested in reinforcement learning (RL). In his eyes, RL seemed fascinating because he could use RL libraries such as [Stable-Baselines3 \(SB3\)](#) to train agents to play all kinds of games. He quickly recognized Proximal Policy Optimization (PPO) as a fast and versatile algorithm and wanted to implement PPO himself as a learning experience. Upon reading the paper, Jon thought to himself, "huh, this is pretty straightforward." He then opened a code editor and started writing PPO. [CartPole-v1](#) from Gym was his chosen simulation environment, and before long, Jon made PPO work with [CartPole-v1](#). He had a great time and felt motivated to make his PPO work with more interesting environments, such as the Atari games and MuJoCo robotics tasks. "How cool would that be?" he thought.

# Reinforcement Learning From Human Feedback (RLHF)

Some of the material is from the ICML 2023 Tutorial by Nathan Lambert  
(<https://icml.cc/media/icml-2023/Slides/21554.pdf>)

# Recall: Three Phases of RLHF

## Phase 1: Pre-Trained Language Model



## Phase 2: Preference Collection

**Playground task 1**

**Talk to the assistant**

Task 1 of 5

Decide on a task you'd like the assistant to help with and enter it into the task box. Interact with the AI assistant. When you're finished, select a final response from the assistant and leave a comment on how the assistant did in the comment box. Click to go to the next task.

**Comments:**  
I thought the assistant was ...

**Rating:**  
Bad ○ ○ ○ ○ Good

**Next Task**

**Human:** I have to come up with a conversation with an AI assistant for the purposes of showing it off in a demo. What sorts of conversation topics would you recommend?

**Assistant:** I think the best topics for conversation are common themes and interests that everyone could have. Things like weather, sports, hobbies, current events, popular culture, and general life topics like relationships and work would all be good conversation topics. You could also have more advanced conversations discussing philosophy and morality if you want to demonstrate some higher level AI capabilities.

**Human:** I think I'm more interested in an advanced conversation. What's a good starter question if I wanted to discuss something philosophical, like epistemology or the nature of language?

**Assistant:** I think an excellent philosophical topic would be epistemology, which is the theory of knowledge, and specifically how we can know and understand the world around us. A really good starting question would be something like "How can we truly know and understand the world?", or "What is the nature of knowledge?". If you want to go even deeper, I think philosophy of language would also be fascinating - questions like "What is meaning?", "How do language and thought relate?", and "How can language mislead and deceive?" would be great discussion topics.

**Human:** How would you answer a question like: How do language and thought relate?

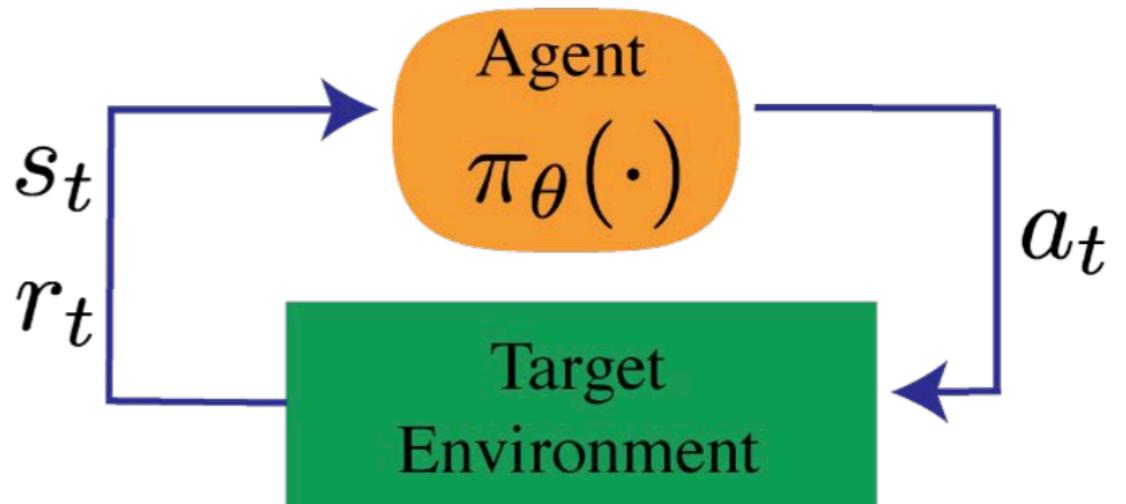
**Choose the most helpful and honest response:**

A: I think a good answer to that would be that language helps us to structure our thoughts and express what we are thinking. Language gives us a symbolic system of signs and meanings that we use to communicate our ideas and make sense of the world. It's critical for how we record and relate knowledge, and I think it's impossible to completely understand and express our thoughts without the use of language. Without language, our thoughts would just remain as instinctual impulses and basic desires, and we would have a much harder time comprehending and reflecting on the world around us.

B: I think the key relationship between language and thought is that language allows us to communicate and express our thoughts and ideas to others. Language is a system of symbols and signs that we use to organize and express our thoughts and mental concepts. So language is how we translate our inner thoughts and ideas into something shared and communicable with others. Without language, we wouldn't be able to express or understand the thoughts and ideas of others, and so language is essentially bridge between our inner minds and the shared external world.

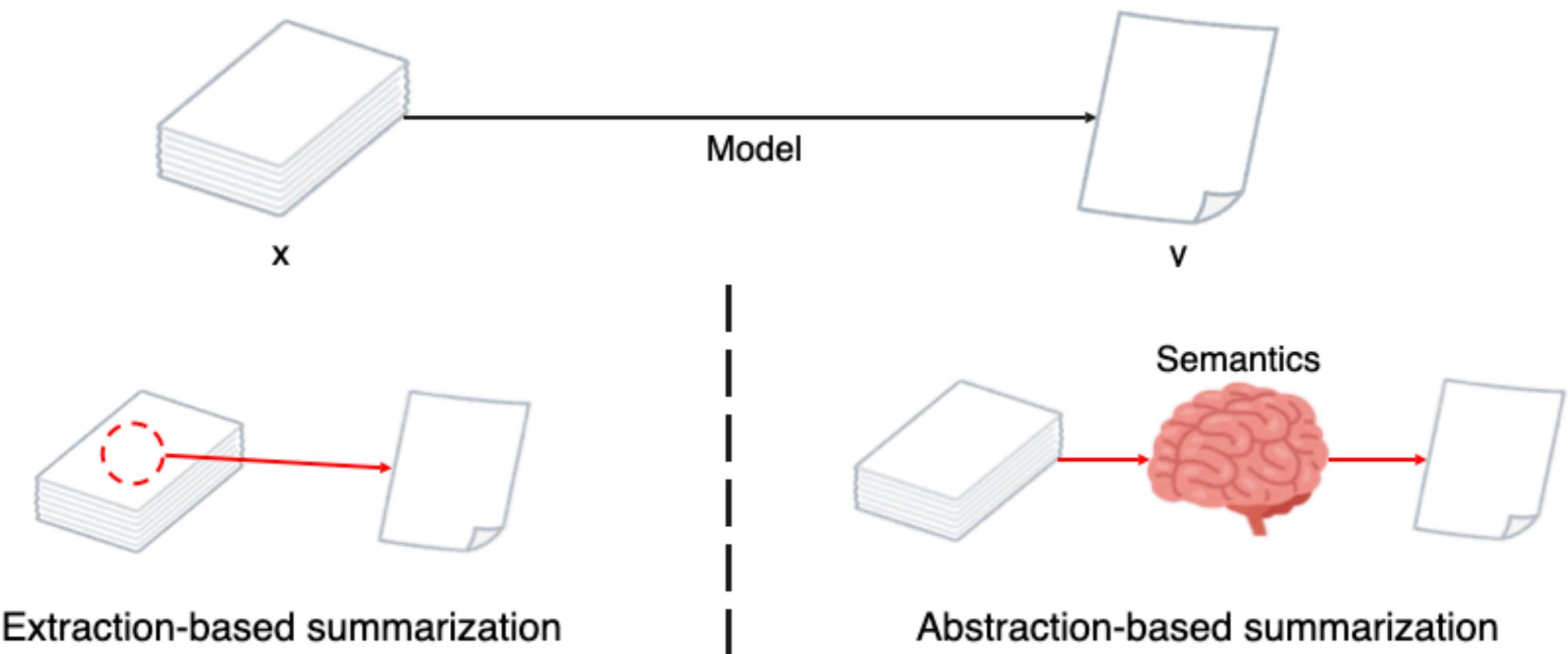
**A | A | A | A | B | B | B**  
A is better      B is better

## Phase 3: RL for Fine-Tuning



# Example: Text Summarization

- Given an article  $x$ , generate a summary of the article (denoted by  $y$ )



# Dataset for Text Summarization?

“TL;DR Summarization” Dataset from Reddit



A screenshot of the Reddit homepage. The left sidebar includes links for Home, Popular, Topics (Gaming, Sports, Business, Crypto, Television, Celebrity), Create a post, and Resources (About Reddit). The main content area shows several posts: one about Prince William and the Women's World Cup, another about Hurricane Hilary, a third about Linus Tech Tips, and a fourth about Lisa Vanderpump. Below these are posts about League of Legends and T1 vs. KT Rolster. A sidebar on the right lists popular communities like r/AskMen, r/PS4, r/apple, r/NBA2k, and r/xboxone.

**Learning to summarize from human feedback**

Nisan Stiennon\* Long Ouyang\* Jeff Wu\* Daniel M. Ziegler\* Ryan Lowe\*  
Chelsea Voss\* Alec Radford Dario Amodei Paul Christiano\*

OpenAI

**Abstract**

As language models become more powerful, training and evaluation are increasingly bottlenecked by the data and metrics used for a particular task. For example, summarization models are often trained to predict human reference summaries and evaluated using ROUGE, but both of these metrics are rough proxies for what we really care about—summary quality. In this work, we show that it is possible to significantly improve summary quality by training a model to optimize for human preferences. We collect a large, high-quality dataset of human comparisons between summaries, train a model to predict the human-preferred summary, and use that model as a reward function to fine-tune a summarization policy using reinforcement learning. We apply our method to a version of the TL;DR dataset of Reddit posts [63] and find that our models significantly outperform both human reference

In this paper, they hired human annotators to rank the summaries in the Reddit dataset

# An Example of Collecting Human Feedback

美國的批踢踢

Reddit Post

關係版

Subreddit: /r/relationships

標題：我(22歲男)和前任(21歲女)交往三年，父母安排了出國旅行，**前任買了自己的機票還沒取消**  
Title: Me [22 M] with my ex [21 F] 3 years. Exotic trip planned by my parents. She purchased her own flight and still hasn't cancelled.

So my ex finished our relationship 2 months ago, we were together for three years.

So the "issue" is as follows:

My family plans a large exotic, expensive trip every summer and this year my family wanted her to join. (Paid by my family) She and I were excited to finally experience it together. Fast forward a few months and she broke up with me. The trip is still happening but she is not coming; my parents cancelled her room, obviously.

我們兩個月前切了，在那之前在一起三年了  
我父母每年夏天都會安排極盡奢華的異國旅行  
今年想要**邀請我前任一起**（費用由我家支出）

我們當時都很期待，(快進)但幾個月前我們切了，我們家仍然會照常去國外度假**without her**，我父母也取消了她的房間

我碰巧在今天看到航班資訊，**看起來她還沒取消她自己訂的機票**

I happened to be looking at out flight info today. She bought her flight on her own and it seem she has not cancelled her flight. Should I say something to her? or just let her deal with financial hit of not doing anything?

我應不應該跟她說呢？或是什麼都不做**讓她承受一些經濟上的打擊**

## Sample 1

Ex who broke up with me months ago bought her own flight and still hasn't cancelled it. Family is planning expensive trip with her. Should I say something or just let her deal with financial hit of not doing anything?

## Sample 2

Ex bought flight ticket for my family's trip on her own. She hasn't cancelled it for the trip in June. We broke up 2 months ago, should I say something to her?

## Human feedback 6B model

**Note:** Ok \_\_\_\_\_ 'Family is planning expensive trip with her' is out of place in the sentence.

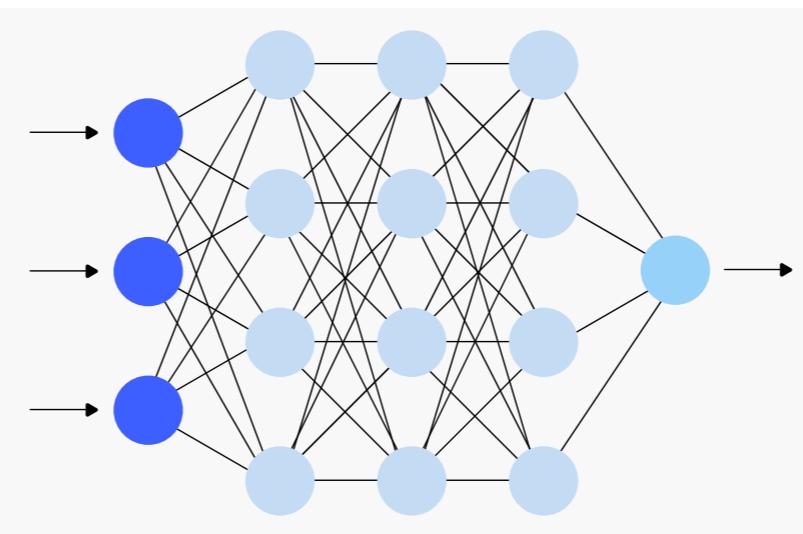
## Human-written reference TL;DR

**Note:** Ok \_\_\_\_\_ in june isn't mentioned  
(Slide Credit: Qian-You Zhang)

# RLHF: Learn a Reward Predictor from Human Feedback

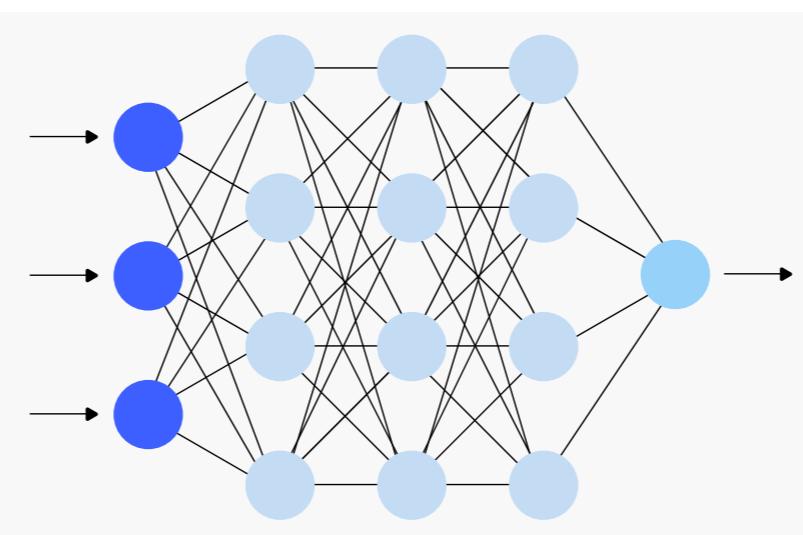
## Sample 1

Ex who broke up with me months ago bought her own flight and still hasn't cancelled it. Family is planning expensive trip with her. Should I say something or just let her deal with financial hit of not doing anything?



Reward = +10

## Reward predictor



Reward = +20

If Sample 2 is better, then it shall be assigned a higher reward!

## Sample 2

Ex bought flight ticket for my family's trip on her own. She hasn't cancelled it for the trip in June. We broke up 2 months ago, should I say something to her?

**Want:** A better summary is assigned a higher reward!

# Bradley-Terry (BT) Model

- For any pair of generated texts, BT model captures the mapping from “reward difference” to “human preference”

Let  $r_1^*$ ,  $r_2^*$  denote the true rewards of the following two sample texts

**Sample 1**

Ex who broke up with me months ago bought her own flight and still hasn't cancelled it. Family is planning expensive trip with her. Should I say something or just let her deal with financial hit of not doing anything?

**Sample 2**

Ex bought flight ticket for my family's trip on her own. She hasn't cancelled it for the trip in June. We broke up 2 months ago, should I say something to her?

## Under BT model:

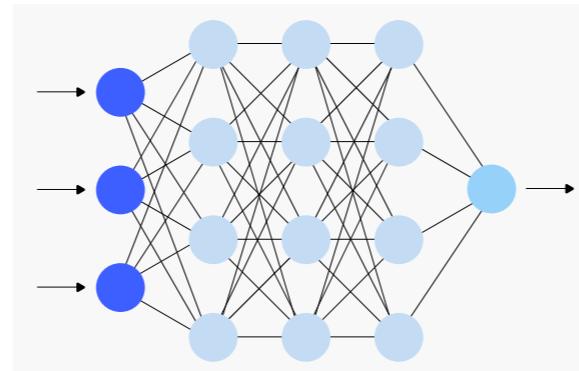
$P(\text{Sample 2 is better than Sample 1})$

$$\begin{aligned} &= \frac{\exp(r_2^*)}{\exp(r_1^*) + \exp(r_2^*)} \\ &= \frac{1}{1 + \exp(-(r_2^* - r_1^*))} \\ &= \text{Sigmoid}(r_2^* - r_1^*) \end{aligned}$$

# RLHF: Learn a Reward Predictor from Human Feedback (Cont.)

## Sample 1

Ex who broke up with me months ago bought her own flight and still hasn't cancelled it. Family is planning expensive trip with her. Should I say something or just let her deal with financial hit of not doing anything?

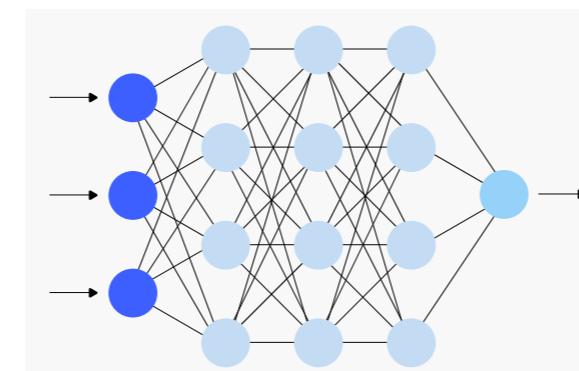


$$r_\phi(x, y_1)$$

Reward predictor  $r_\phi$

## Sample 2

Ex bought flight ticket for my family's trip on her own. She hasn't cancelled it for the trip in June. We broke up 2 months ago, should I say something to her?



$$r_\phi(x, y_2)$$

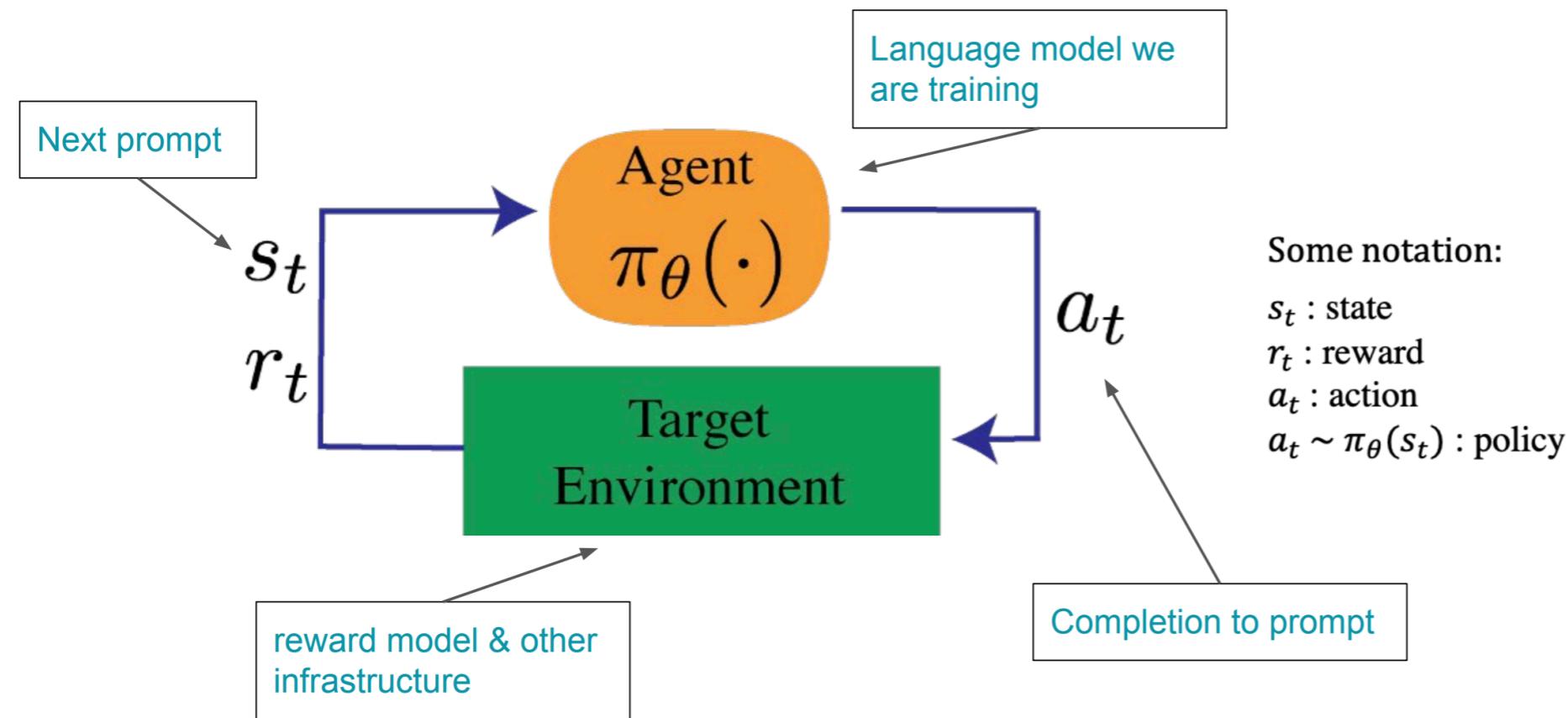
(Suppose  $y_2$  is better)

(Let  $x$  be the prompt or original text for w/o summarization)

Loss function (maximum likelihood estimation under BT model)

$$L(\phi) = -E_{(x, y_1, y_2)} \left[ \log (\text{Sigmoid}(r_\phi(x, y_2) - r_\phi(x, y_1))) \right]$$

# Put Everything Together: Fine-Tuning by PPO-KL



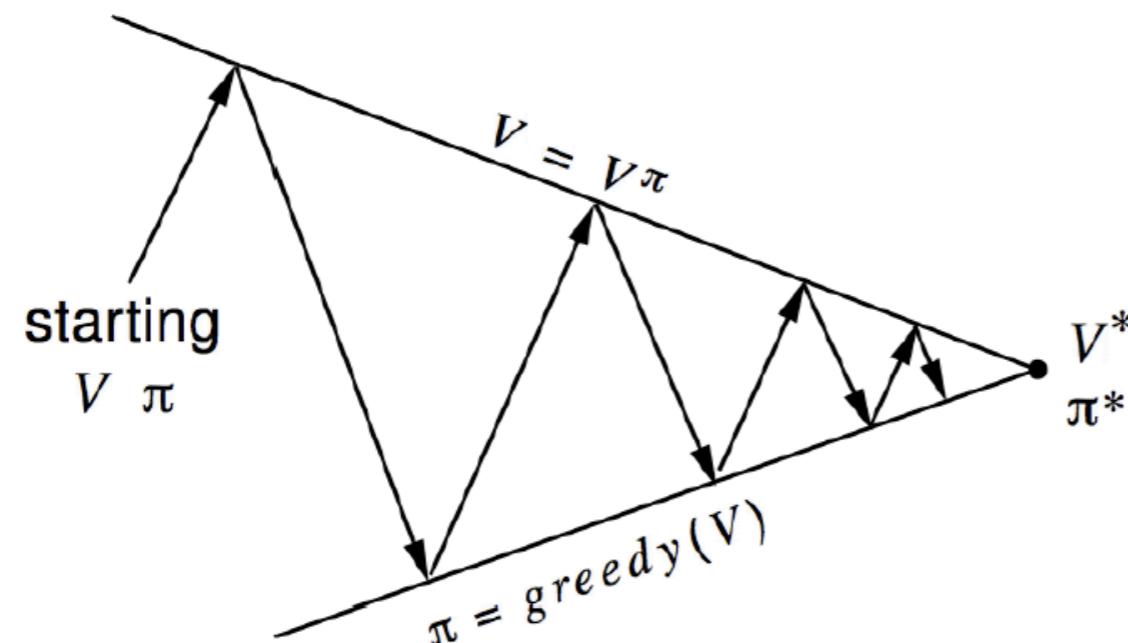
$$\theta^* := \arg \max_{\pi_\theta} \mathbb{E}_{x \sim \mathcal{D}, y \sim \pi_\theta(\cdot|x)} \left[ r_\phi(x, y) - \beta \mathbb{D}_{\text{KL}} [\pi_\theta(\cdot | x) \| \pi_{\text{ref}}(\cdot | x)] \right]$$

This is exactly PPO-KL under the learned reward predictor  $r_\phi$

# Next-Topic: Value-Based Methods

# Review: (Generalized) Policy Iteration

- If model parameters are **known**, an optimal policy can be found by **(generalized) policy iteration**

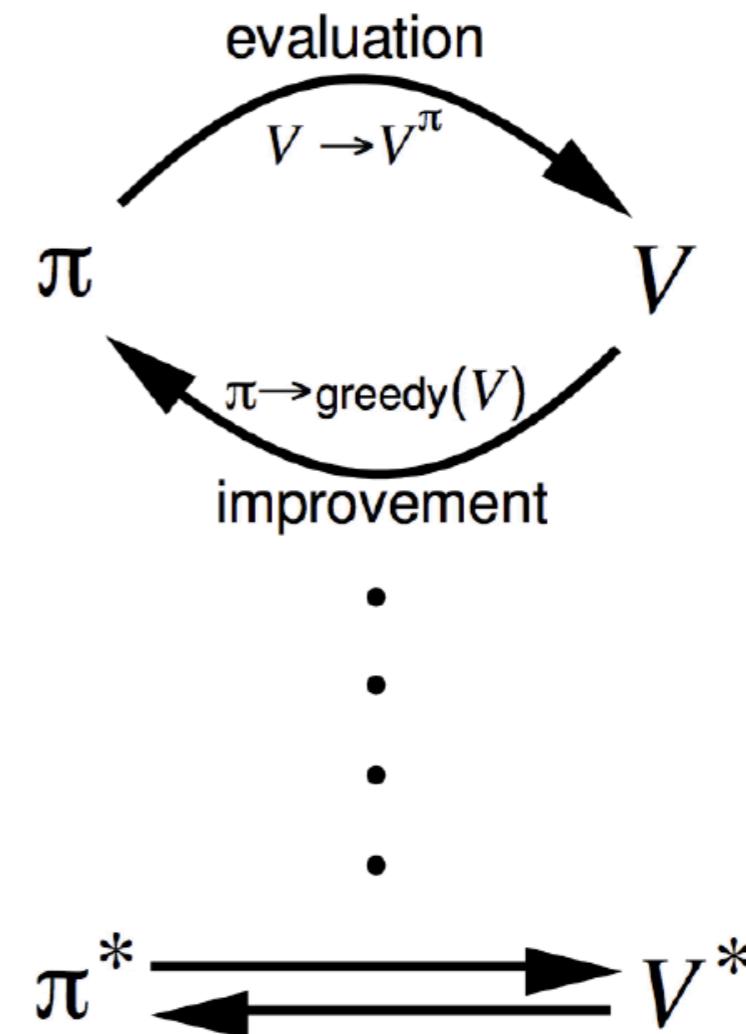


**Policy evaluation** Estimate  $v_\pi$

Iterative policy evaluation

**Policy improvement** Generate  $\pi' \geq \pi$

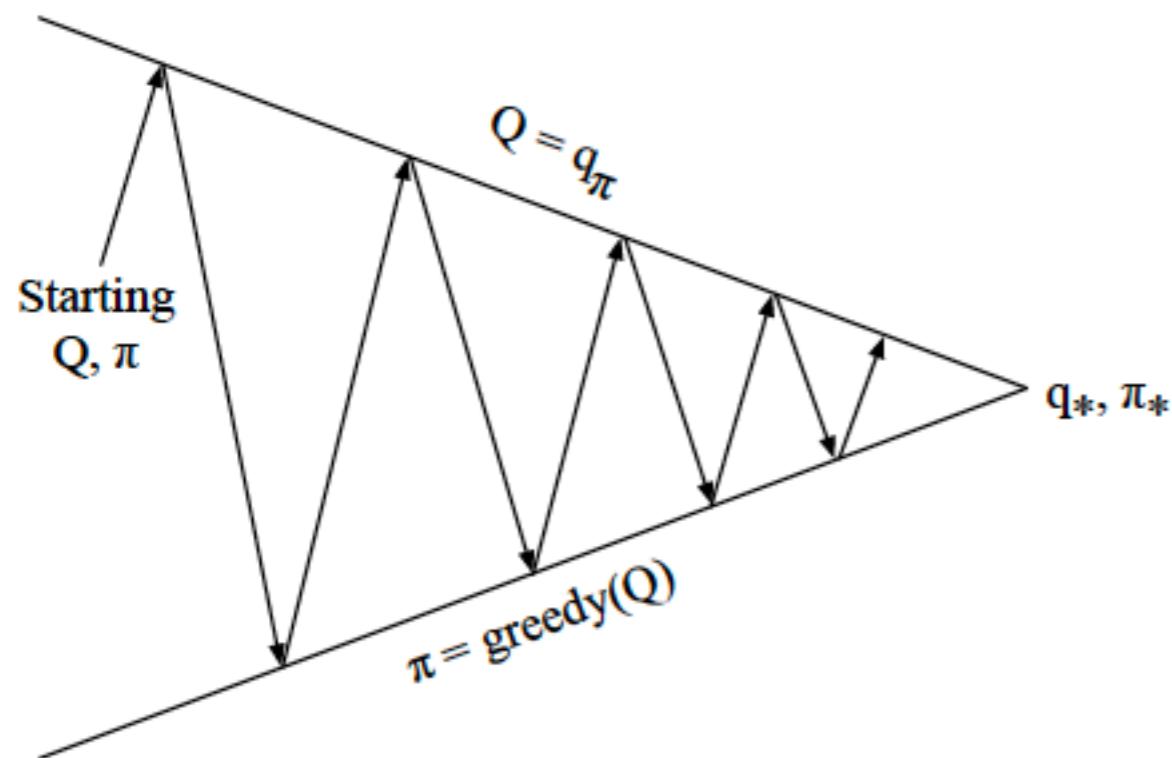
Greedy policy improvement



- Question:** If model parameters are **unknown**, then how to learn an optimal policy by **policy iteration**?

# Greedy MC / TD Control?

- ▶ Let's design our first **value-based** algorithm!
- ▶ Idea: MC or TD + one-step greedy policy improvement?



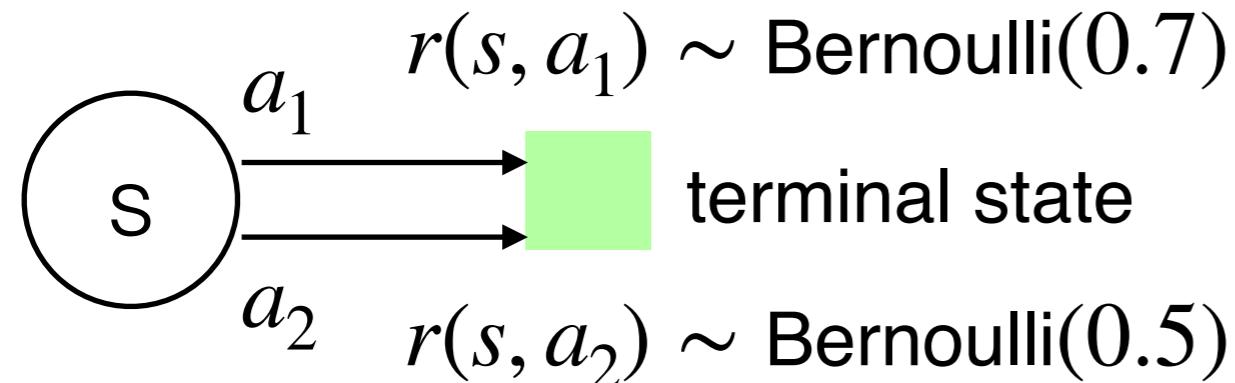
**Question:** Any issue with one-step policy improvement?

**Policy Evaluation:** MC or TD policy evaluation

**Policy Improvement:** Greedy policy improvement?

# Example: “MC+Greedy” for 1-State MDP

- ▶ **Toy Example:** 1-state MDP with 2 actions and set  $\pi_0(s) = a_1$ . Consider a greedy policy w.r.t. Q function (ties are broken randomly)



$$\pi^*(s) = a_1$$



$a_1$



$a_2$

$k = 1$ : choose  $a_1$ , and  $r(s, a_1) = 0 \rightarrow \hat{Q}_1(s, a_1) = \underline{\hspace{2cm}}, \hat{Q}_1(s, a_2) = \underline{\hspace{2cm}}, \pi_1(s) = \underline{\hspace{2cm}}$

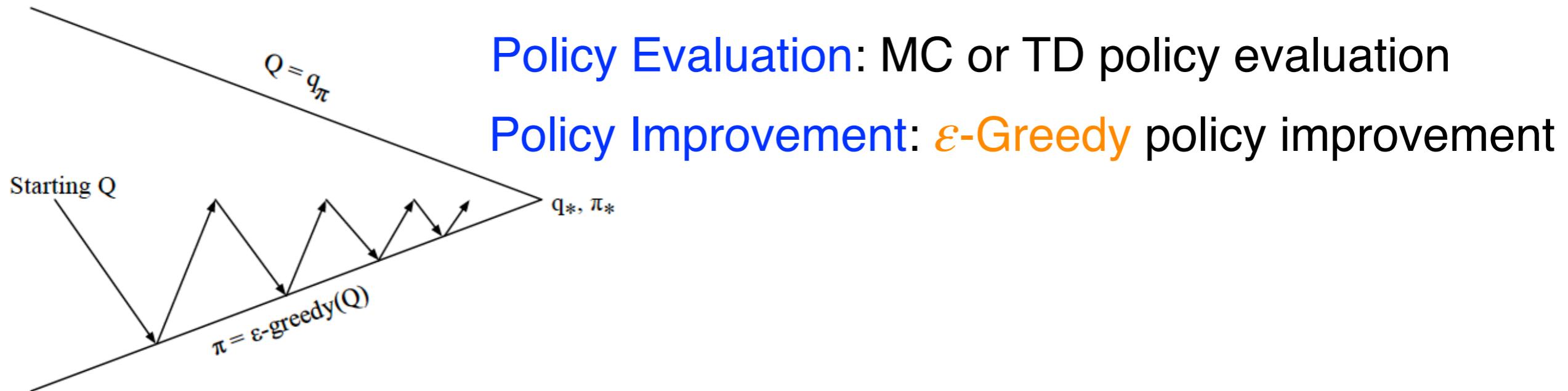
$k = 2$ : choose  $a_2$ , and  $r(s, a_2) = 1 \rightarrow \hat{Q}_2(s, a_1) = \underline{\hspace{2cm}}, \hat{Q}_2(s, a_2) = \underline{\hspace{2cm}}, \pi_2(s) = \underline{\hspace{2cm}}$

$k = 3$ : choose  $a_2$ , and  $r(s, a_2) = 0 \rightarrow \hat{Q}_3(s, a_1) = \underline{\hspace{2cm}}, \hat{Q}_3(s, a_2) = \underline{\hspace{2cm}}, \pi_3(s) = \underline{\hspace{2cm}}$

$k = 4$ : choose  $a_2$ , and  $r(s, a_2) = 0 \rightarrow \hat{Q}_4(s, a_1) = \underline{\hspace{2cm}}, \hat{Q}_4(s, a_2) = \underline{\hspace{2cm}}, \pi_4(s) = \underline{\hspace{2cm}}$

# $\varepsilon$ -Greedy + MC / TD?

- Idea: MC / TD + one-step  $\varepsilon$ -greedy policy improvement



- With probability  $1 - \varepsilon$ : choose the greedy action
- With probability  $\varepsilon$ : choose an action uniformly at random

$$\pi(a | s) = \begin{cases} \frac{\varepsilon}{|\mathcal{A}|} + 1 - \varepsilon, & \text{if } a = \arg \max_{a \in \mathcal{A}} Q(s, a) \\ \frac{\varepsilon}{|\mathcal{A}|}, & \text{otherwise} \end{cases}$$

# $\varepsilon$ -Greedy Policy Improvement Theorem

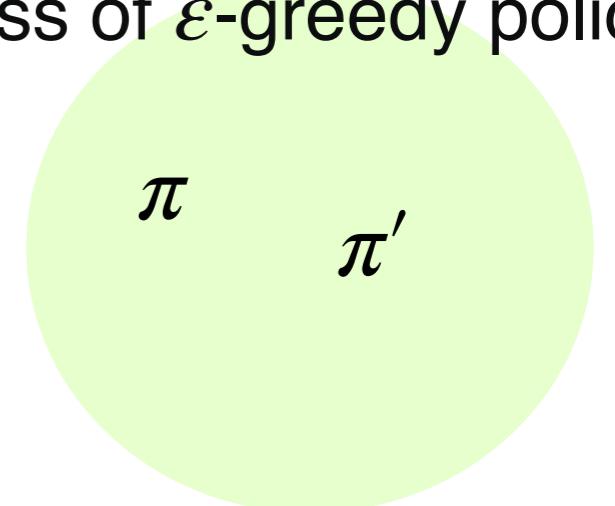
- ▶  **$\varepsilon$ -Greedy Policy Improvement Theorem:**

For any  $\varepsilon$ -greedy policy  $\pi$ , the new  $\varepsilon$ -greedy policy  $\pi'$  with respect to  $Q^\pi$  is an improvement, i.e.

$$V^{\pi'}(s) \geq V^\pi(s)$$

- ▶ **Proof:** The proof is similar to one-step policy improvement in Lecture 3 (see the appendix)

Class of  $\varepsilon$ -greedy policies



# $\varepsilon$ -Greedy Monte-Carlo Control (Formally)

- ▶  $\varepsilon$ -Greedy MC Control:

Step 1: Initialize  $Q(s, a)$  and  $N(s, a) = 0$  for each  $(s, a)$

Step 2: In episode  $k$ , sample a trajectory  $\tau = (s_0, a_0, r_1, \dots) \sim \pi_k$

For each first-visit  $(s_t, a_t)$  in the current episode, update

$$N(s_t, a_t) \leftarrow N(s_t, a_t) + 1$$

$$Q(s_t, a_t) \leftarrow Q(s_t, a_t) + \frac{1}{N(s_t, a_t)}(G_t - Q(s_t, a_t))$$

Improve the policy by using  $\varepsilon$ -greedy w.r.t.  $Q(\cdot, \cdot)$

Question: Is  $\varepsilon$ -Greedy Monte-Carlo control *on-policy* or *off-policy*?

# Sarsa: $\varepsilon$ -Greedy TD Control (Formally)

- ▶ Sarsa:  $\varepsilon$ -Greedy TD control

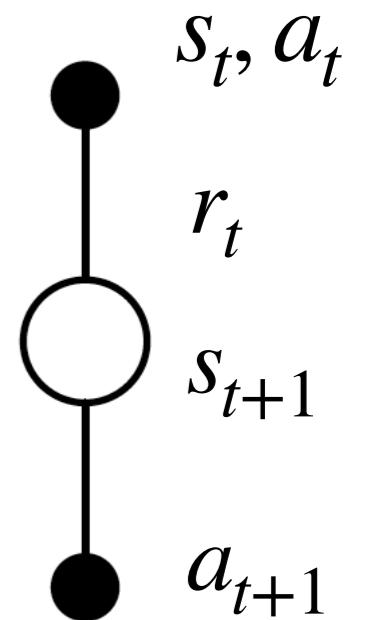
Step 1: Initialize  $Q(s, a)$

Step 2: In each step  $t$ , repeat the following:

In state  $s_t$ , apply  $a_t \sim \varepsilon\text{-greedy}(Q(s_t, \cdot))$

Observe  $s_{t+1}$  and draw  $a_{t+1} \sim \varepsilon\text{-greedy}(Q(s_{t+1}, \cdot))$

$$Q(s_t, a_t) \leftarrow Q(s_t, a_t) + \alpha_t(s_t, a_t)(r_t + \gamma Q(s_{t+1}, a_{t+1}) - Q(s_t, a_t))$$



**Question:** Why is it called “Sarsa”?

**Question:** Is “Sarsa” *on-policy* or *off-policy*?

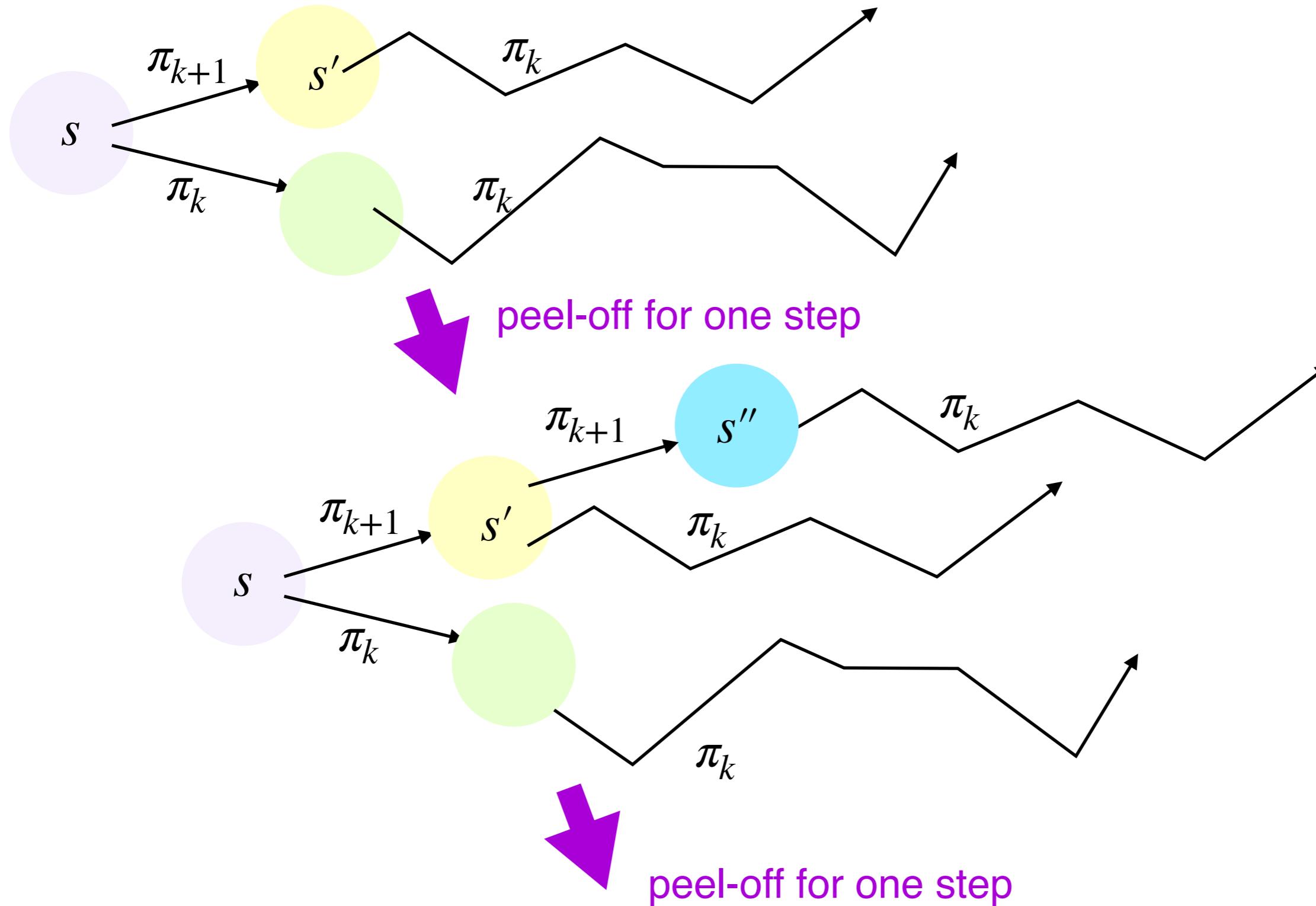
- ▶ **Question:** Does Sarsa enjoy convergence to the optimal Q?  
*Yes, under proper conditions*

- ▶ **Greedy in the Limit with Infinite Exploration (GLIE):**
  - (G1) All state-action pairs are explored **infinitely many times**
  - (G2) The policy converges on a greedy policy (no exploration in the limit)

- ▶ **Theorem:** Sarsa converges to the optimal action-value function, i.e.,  $Q(s, a) \rightarrow Q^*(s, a)$ , under the following conditions:
  - (1) GLIE
  - (2)  $\sum_{t=1}^{\infty} \alpha_t(s, a) = \infty$ ,  $\sum_{t=1}^{\infty} \alpha_t(s, a)^2 < \infty$ , for all  $(s, a)$
- ▶ **Question:** How to choose  $\alpha_t(s, a)$ ?
- ▶ **Proof Technique:** *Stochastic approximation* (To be discussed next time)

## Appendix: Proof of $\varepsilon$ -Greedy Policy Improvement Theorem

# Recall from Lecture 3: Proof of Monotonic Policy Improvement by “Peeling off”



# Recall from Lecture 3: Proof of Monotonic Policy Improvement by “Peeling off”

$$V^{\pi_k}(s) \leq \max_a Q^{\pi_k}(s, a)$$

← Step 1: peel-off for one step

$$\begin{aligned} &= \max_a R(s, a) + \gamma \sum_{s'} P(s' | s, a) V^{\pi_k}(s') \\ &= R(s, \pi_{k+1}(s)) + \gamma \sum_{s'} P(s' | s, \pi_{k+1}(s)) V^{\pi_k}(s') \\ &\leq R(s, \pi_{k+1}(s)) + \gamma \sum_{s'} P(s' | s, \pi_{k+1}(s)) \max_{a'} Q^{\pi_k}(s', a') \\ &= R(s, \pi_{k+1}(s)) + \gamma \sum_{s'} P(s' | s, \pi_{k+1}(s)) \\ &\quad \times \left( R(s', \pi_{k+1}(s')) + \gamma \sum_{s''} P(s'' | s', \pi_{k+1}(s')) V^{\pi_k}(s'') \right) \\ &\quad \dots \\ &= V^{\pi_{k+1}}(s) \end{aligned}$$

↑ Step 2: peel-off for all the remaining steps

# Proof of $\varepsilon$ -Greedy Policy Improvement: One-Step Comparison

**Step 1:** Show  $V^\pi(s) \leq \sum_a \pi'(a | s) Q^\pi(s, a) =: Q^\pi(s, \pi'(s))$

---

$$\begin{aligned} V^\pi(s) &= \sum_a \pi(a | s) Q^\pi(s, a) && \text{This term } \geq 0 \text{ if } \pi \text{ is } \varepsilon\text{-greedy} \\ &= \frac{\varepsilon}{|\mathcal{A}|} \sum_a Q^\pi(s, a) + (1 - \varepsilon) \sum_a \boxed{\frac{\pi(a | s) - \frac{\varepsilon}{|\mathcal{A}|}}{1 - \varepsilon} Q^\pi(s, a)} \\ &\leq \frac{\varepsilon}{|\mathcal{A}|} \sum_a Q^\pi(s, a) + (1 - \varepsilon) \cdot \max_a Q^\pi(s, a) \\ &= \sum_a \pi'(a | s) Q^\pi(s, a) \equiv Q^\pi(s, \pi'(s)) \end{aligned}$$

Hence, we have  $V^\pi(s) \leq Q^\pi(s, \pi'(s))$

# Proof of $\varepsilon$ -Greedy Policy Improvement: “Peeling off”

**Step 2:** Peel-off for all the remaining steps

---

$$V^\pi(s) \leq Q^\pi(s, \pi'(s))$$

$$= R(s, \pi'(s)) + \gamma \sum_{s'} P(s' | s, \pi'(s)) V^\pi(s')$$

$$\leq R(s, \pi'(s)) + \gamma \sum_{s'} P(s' | s, \pi'(s)) Q^\pi(s', \pi'(s'))$$

$$= R(s, \pi'(s)) + \gamma \sum_{s'} P(s' | s, \pi'(s))$$

$$\times \left( R(s', \pi'(s')) + \gamma \sum_{s''} P(s'' | s', \pi'(s')) V^\pi(s'') \right)$$

...

$$= V^{\pi'}(s)$$