

• REC

Проверить, идет ли запись

Меня хорошо видно
&& слышно?





Какой алгоритм
илюстрирован?

RL course

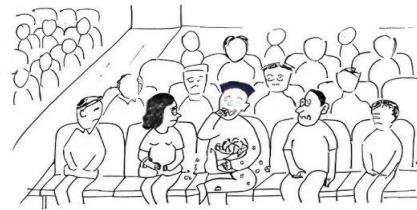
DDPG

Начнем в 20:01 [мск]

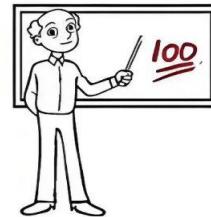
otus.ru

环境的反馈
观众的掌声

结构 (评论家-演员)



评论家
(评委)



演员

Тема вебинара

Reinforcement learning.

RL course DDPG.



Анастасия Капралова

CEO и сооснователь IT компании kapralov.ai

Опыт:

8+ лет разработки в сфере ИИ.

Люблю сложные, творческие и необычные проекты, для которых, с учетом ограничений, не существует уже готового подхода, особенно люблю проекты с множеством различных сенсоров и с кастомным железом.

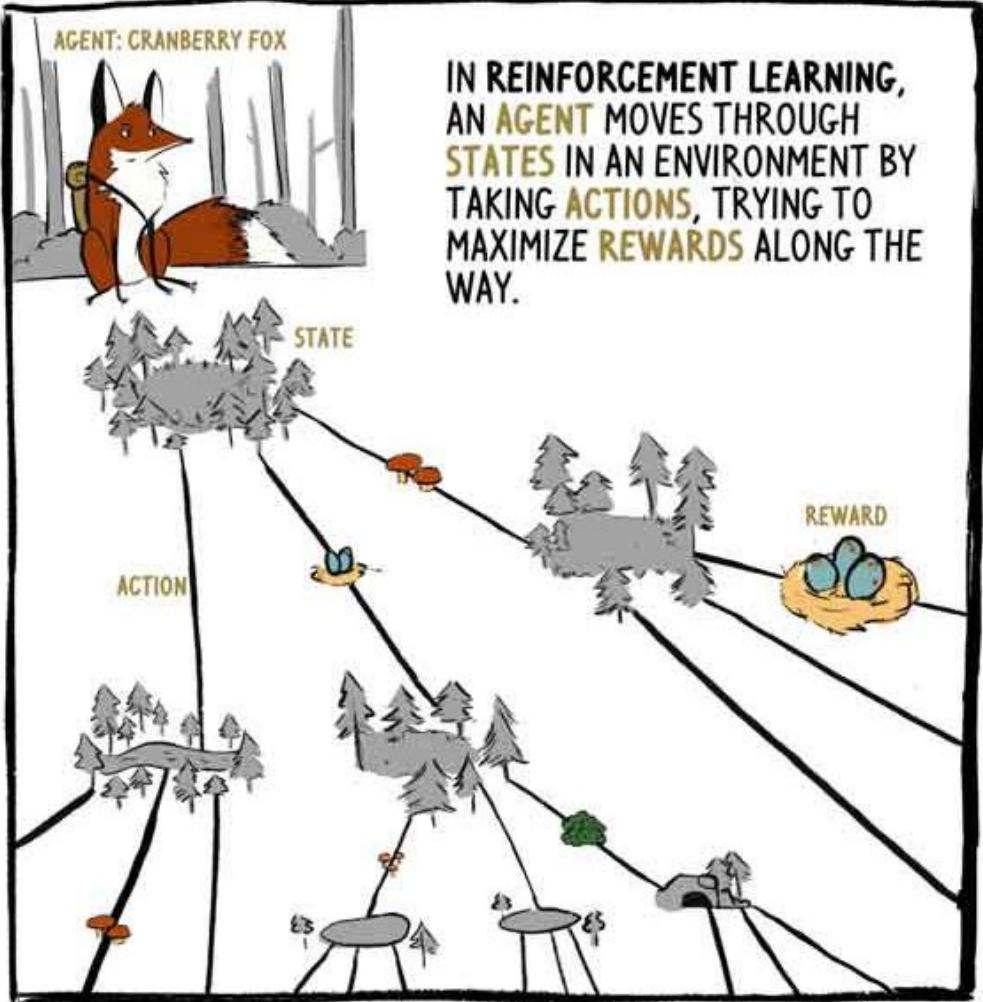
5+ лет руководжу командами и проектами.

Провожу консультации в IT сфере по техническим и управленческим вопросам, помогаю наладить процесс разработки, коммуникации и мотивацию в командах. Имею международные сертификаты ICSE.

Лауреат премии им. Ильи Сегаловича, победитель и призер множества IT соревнований

@stasysp (TG)





<https://hackernoon.com/intuitive-rl-intro-to-advantage-actor-critic-a2c-4ff545978752>

<https://habr.com/ru/articles/442522/>

A2CS TAKE IN A STATE—SENSORY INPUTS IN CRANBERRY'S CASE—and generate **TWO OUTPUTS**:

1) AN ESTIMATE OF HOW MANY REWARDS THEY EXPECT TO GET FROM THAT POINT ONWARDS, THE STATE VALUE.

THE "CRITIC"

WOW, WHAT A WONDERFUL GLEN!
THIS WILL BE A FRUITFUL DAY OF FORAGING. I BET I'LL GATHER 20 REWARD POINTS BEFORE SUNSET TODAY.

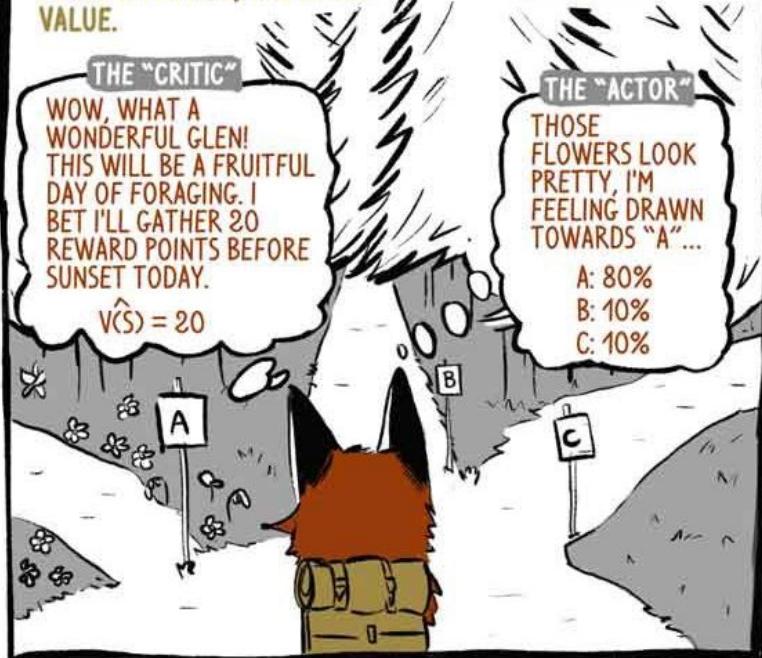
$$\hat{V}(s) = 20$$

2) A RECOMMENDATION OF WHAT ACTION TO TAKE, THE POLICY

THE "ACTOR"

THOSE FLOWERS LOOK PRETTY, I'M FEELING DRAWN TOWARDS "A"...

A: 80%
B: 10%
C: 10%



<https://hackernoon.com/intuitive-rl-intro-to-advantage-actor-critic-a2c-4ff545978752>



T=1



T=2



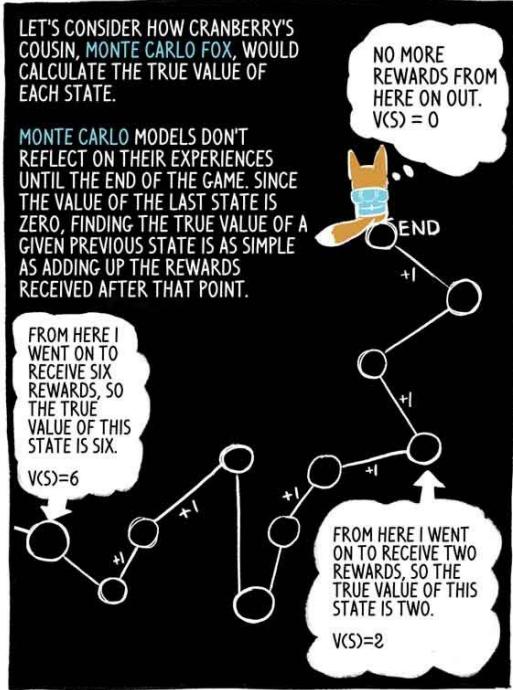
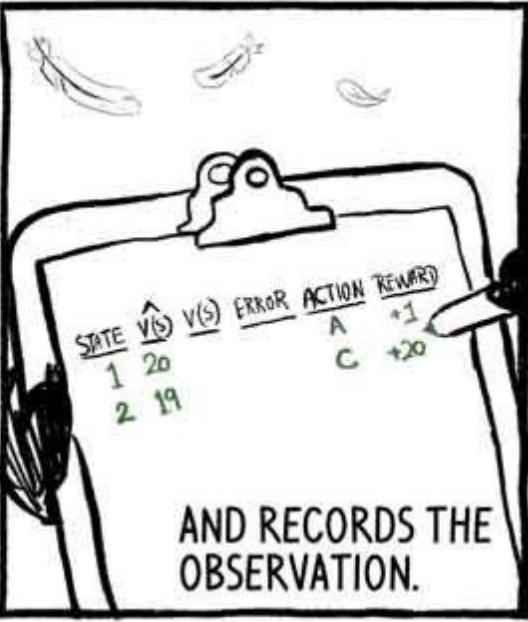
T=3



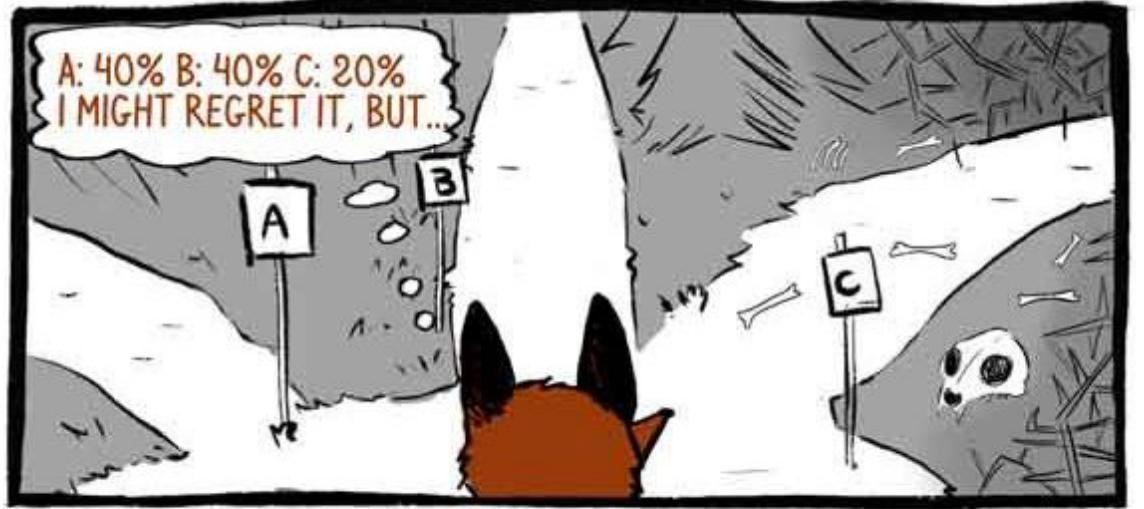
T=4

REWARDS ARE OFTEN DISCOUNTED TO REFLECT THE FACT THAT A REWARD NOW IS BETTER THAN A REWARD IN THE FUTURE. TO KEEP THINGS SIMPLE, CRANBERRY ISN'T DISCOUNTING HER REWARDS.

<https://hackernoon.com/intuitive-rl-intro-to-advantage-actor-critic-a2c-4ff545978752>



<https://hackernoon.com/intuitive-rl-intro-to-advantage-actor-critic-a2c-4ff545978752>



<https://hackernoon.com/intuitive-rl-intro-to-advantage-actor-critic-a2c-4ff545978752>

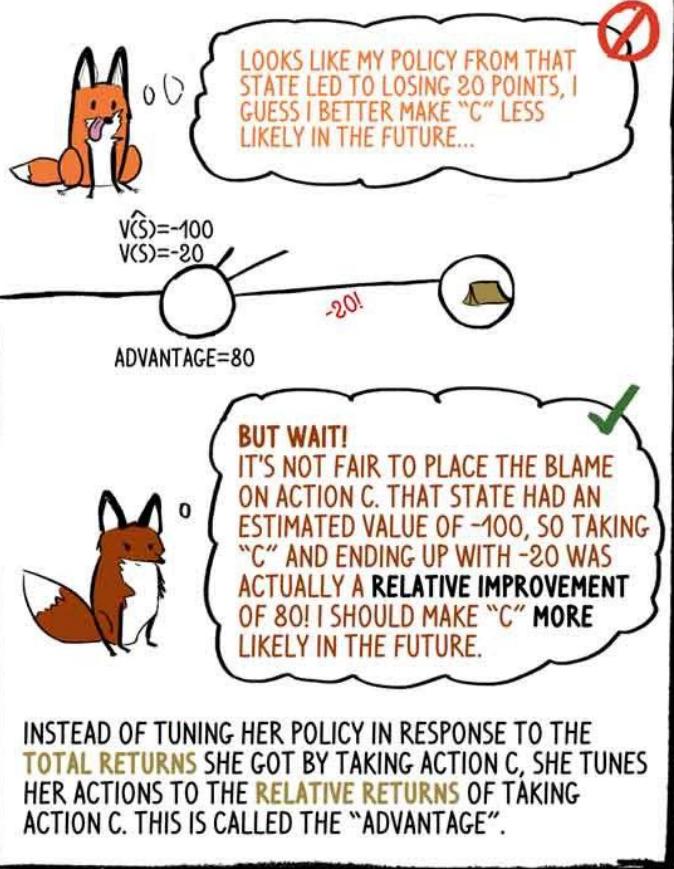


TO FURTHER ENCOURAGE EXPLORATION, A VALUE CALLED ENTROPY IS SUBTRACTED FROM THE LOSS FUNCTION. ENTROPY REFERS TO THE "SPREAD" OF THE ACTION DISTRIBUTION.



<https://hackernoon.com/intuitive-rl-intro-to-advantage-actor-critic-a2c-4ff545978752>

A SIMPLE POLICY GRADIENT FOX WOULD LOOK AT THE ACTUAL RETURNS FOLLOWING AN ACTION AND TUNE HER POLICY TO MAKE GOOD RETURNS MORE LIKELY.



<https://hackernoon.com/intuitive-rl-intro-to-advantage-actor-critic-a2c-4ff545978752>

WHAT WE CALLED THE **ADVANTAGE** IS JUST THE **ERROR**. AS THE ADVANTAGE, CRANBERRY USES IT TO MAKE ACTIONS THAT WERE SURPRISINGLY GOOD MORE LIKELY. AS THE ERROR, SHE USES THE SAME QUANTITY TO NUDGE HER INNER CRITIC TO MAKE BETTER ESTIMATIONS OF STATE VALUES.

ACTOR USES ADVANTAGE



CRITIC USES ERROR



NOW WE CAN SHOW HOW TOTAL LOSS IS COMPUTED—THIS IS THE FUNCTION WE MINIMIZE TO IMPROVE OUR MODEL.

$$\text{TOTAL LOSS} = \text{ACTION LOSS} + \text{VALUE LOSS} - \text{ENTROPY}.$$

NOTICE WE'RE SHOVING GRADIENTS OF THREE QUALITATIVELY DIFFERENT TYPES BACK THROUGH A SINGLE NN. THIS IS EFFICIENT BUT IT CAN MAKE CONVERGENCE MORE DIFFICULT.

<https://hackernoon.com/intuitive-rl-intro-to-advantage-actor-critic-a2c-4ff545978752>

Продолжаем



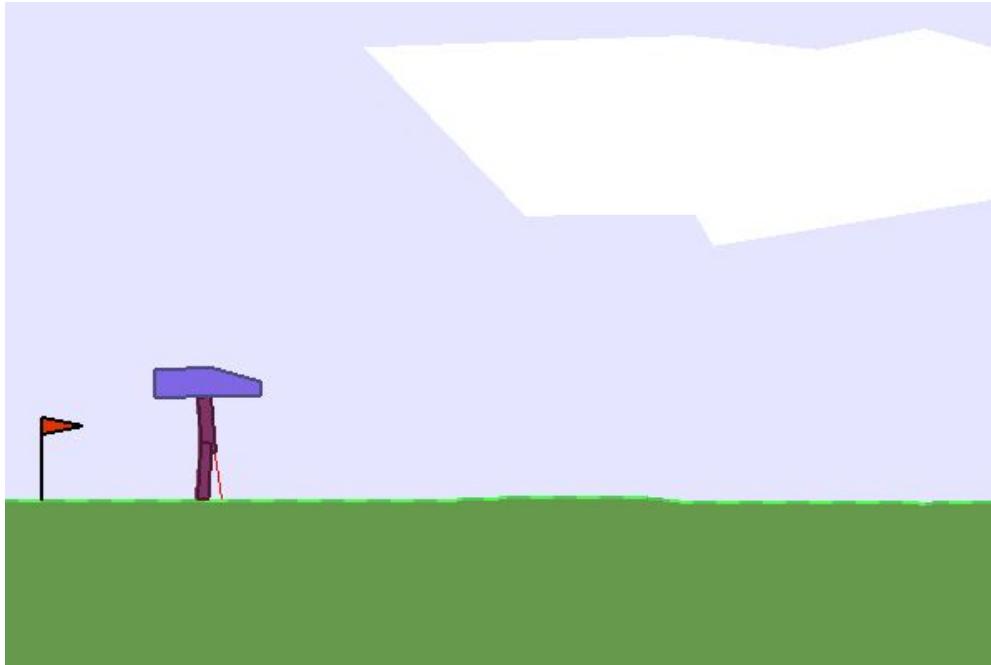
Проблемы

- Value-Iteration и Policy-Iteration
- Монте-Карло
- Q-learning
- DQN / PG
- Actor-Critic → A2C → A3C

Bipedal walker



Про среду

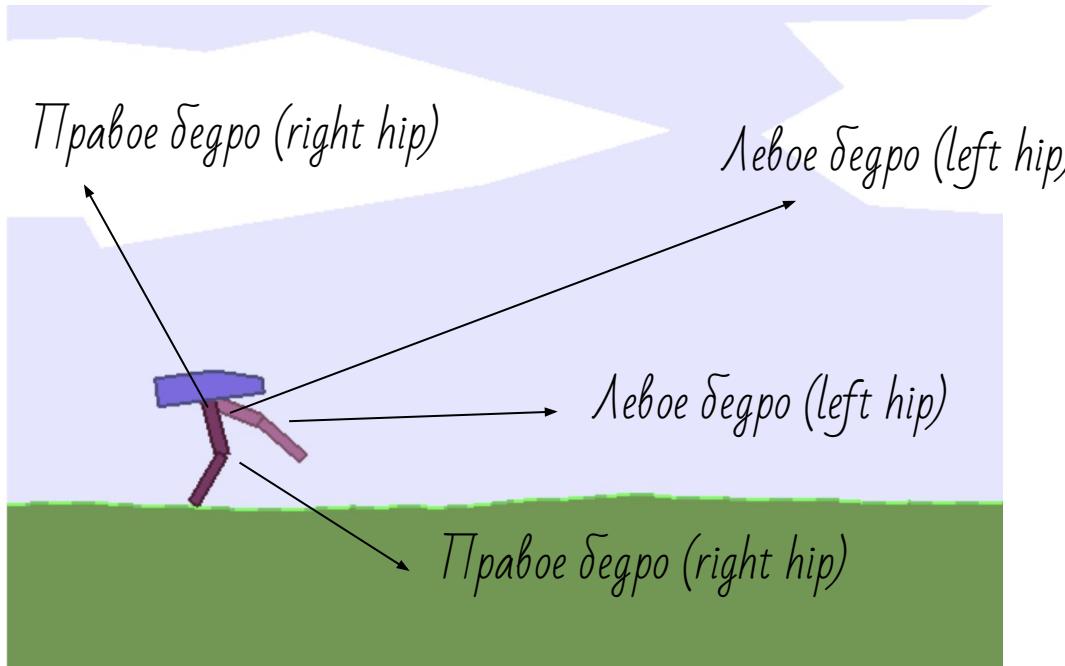


Цель: 300 очков за 1600 шагов

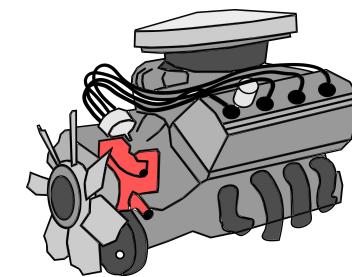
- Двигаться быстро
- Не падать
- Израсходовать меньше энергии



Действия



Задаем скорость на четырех моторах: [-1, 1]



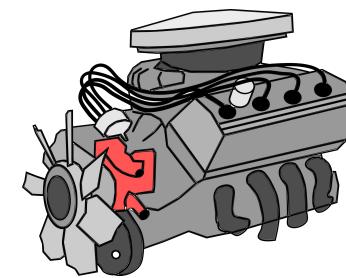
torque = крутящий момент



Действия

Num	Name	Min	Max
0	Hip_1 (Torque / Velocity)	-1	+1
1	Knee_1 (Torque / Velocity)	-1	+1
2	Hip_2 (Torque / Velocity)	-1	+1
3	Knee_2 (Torque / Velocity)	-1	+1

Задаем скорость на четырех моторах: [-1, 1]



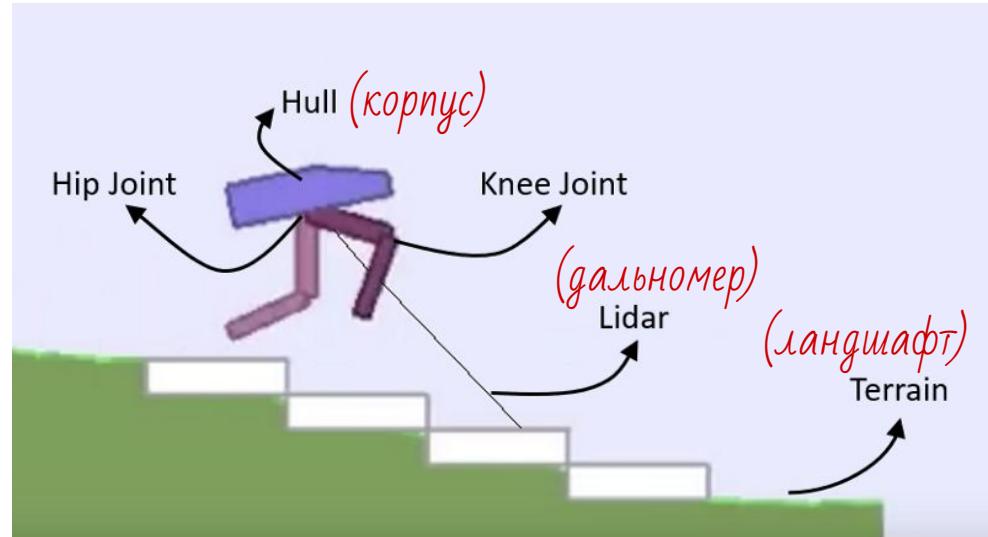
torque = крутящий момент





Состояния среды

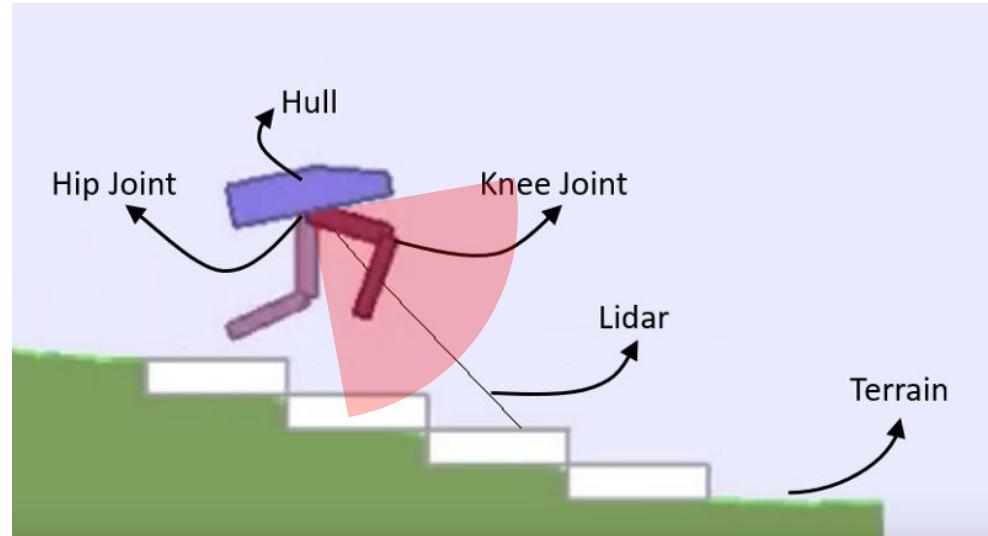
Num	Observation	Min	Max	Mean
0	hull_angle	0	2π	0.5
1	hull_angularVelocity	-inf	+inf	-
2	vel_x	-1	+1	-
3	vel_y	-1	+1	-
4	hip_joint_1_angle	-inf	+inf	-
5	hip_joint_1_speed	-inf	+inf	-
6	knee_joint_1_angle	-inf	+inf	-
7	knee_joint_1_speed	-inf	+inf	-
8	leg_1_ground_contact_flag	0	1	-
9	hip_joint_2_angle	-inf	+inf	-
10	hip_joint_2_speed	-inf	+inf	-
11	knee_joint_2_angle	-inf	+inf	-
12	knee_joint_2_speed	-inf	+inf	-
13	leg_2_ground_contact_flag	0	1	-
14-23	10 lidar readings	-inf	+inf	-





Про лидар

- 10 равномерных измерений
- По дуге в 90^0
- Перпендикулярно корпусу





Варианты стратегий



KNEE BALANCE (OPTIMAL)



DOUBLE BALANCE (FASTEST RUNNER)



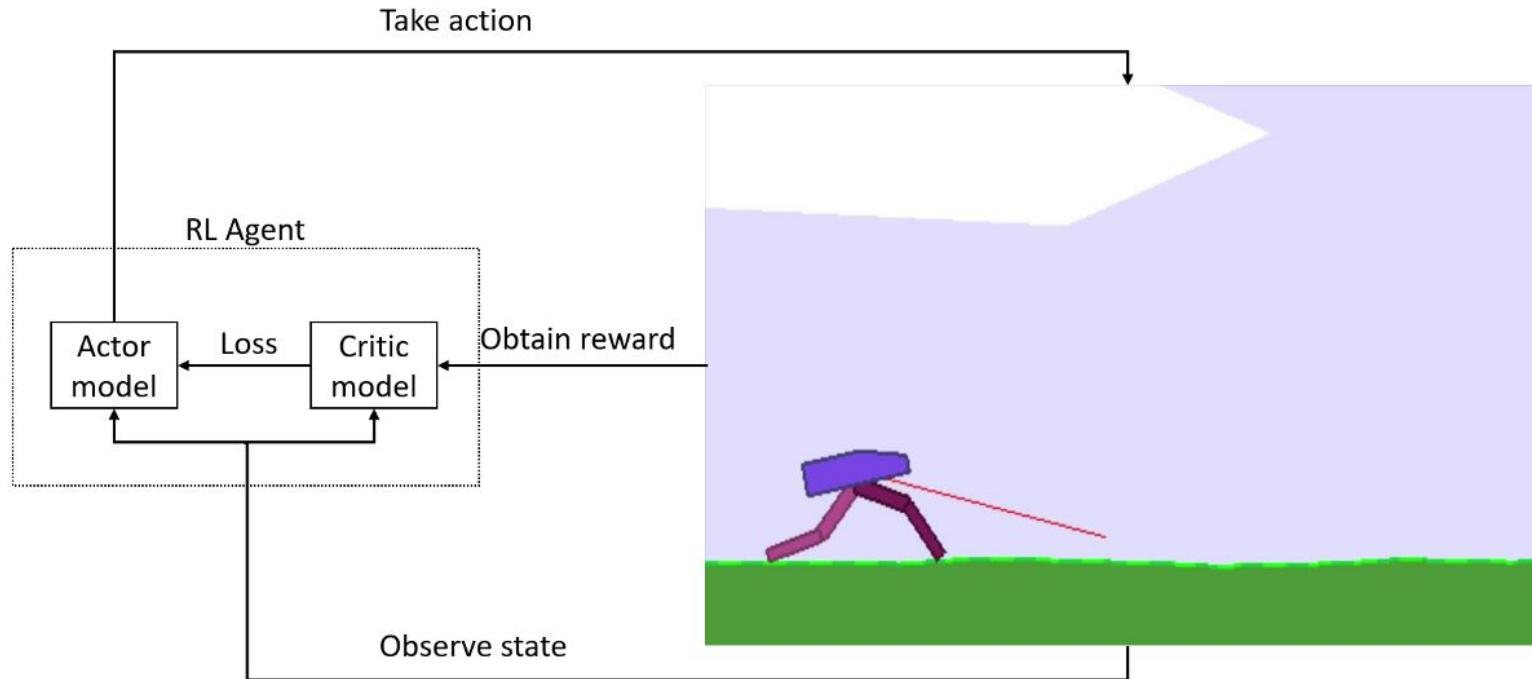
FRONT BALANCE



REAR BALANCE

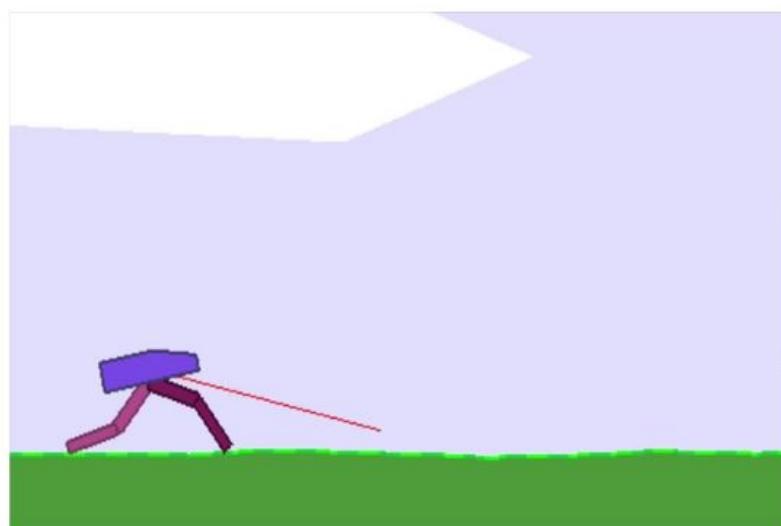


Архитектура

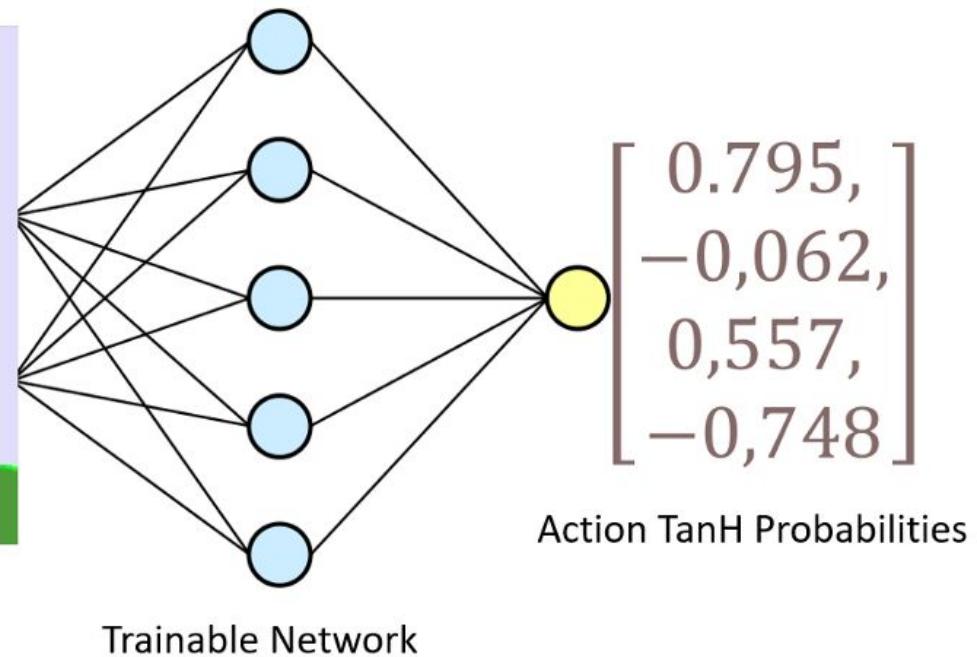




Actor

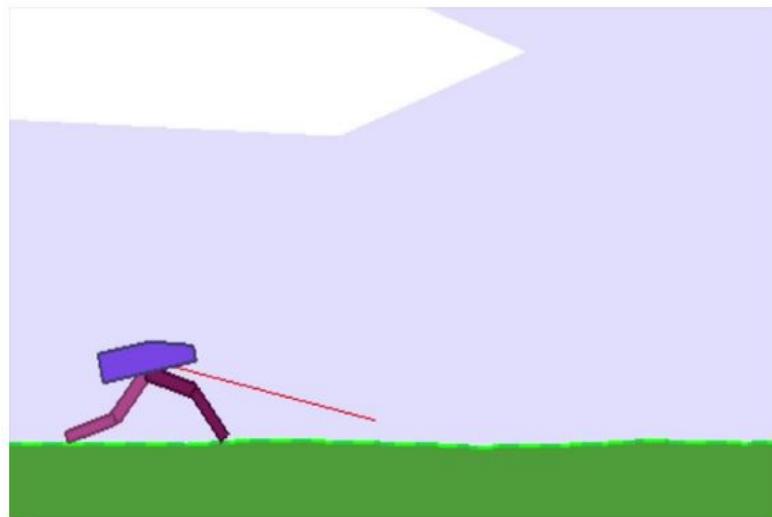


Input State

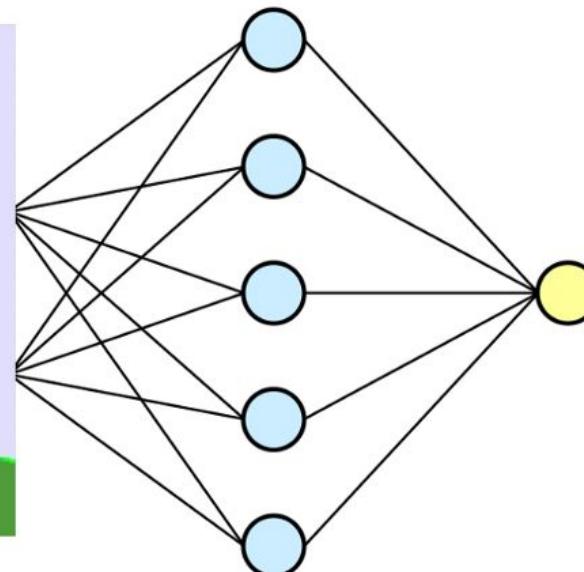




Critic



Input State



Trainable Network

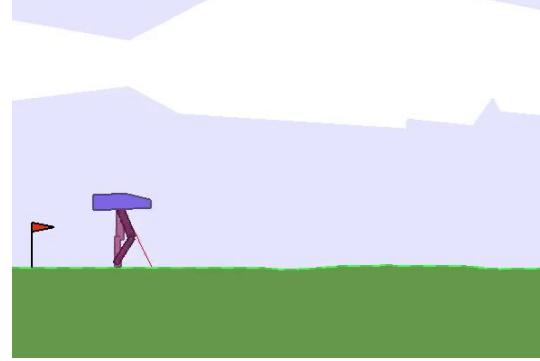
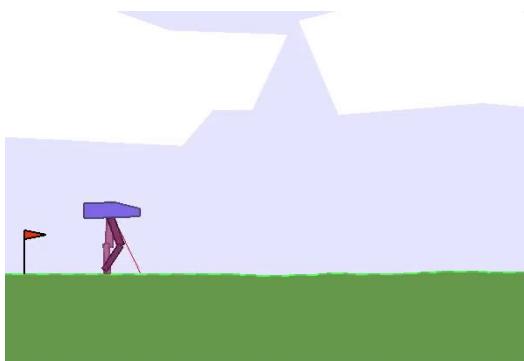
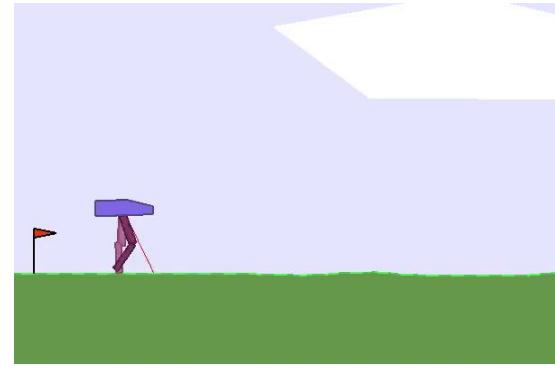
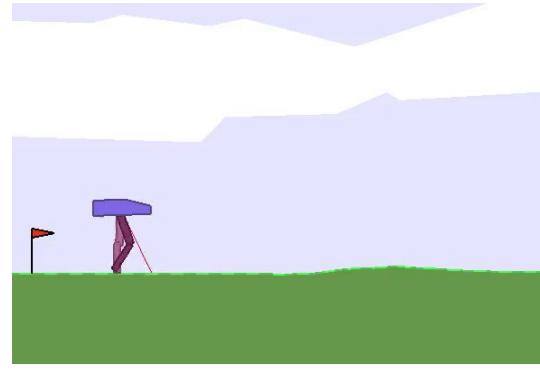
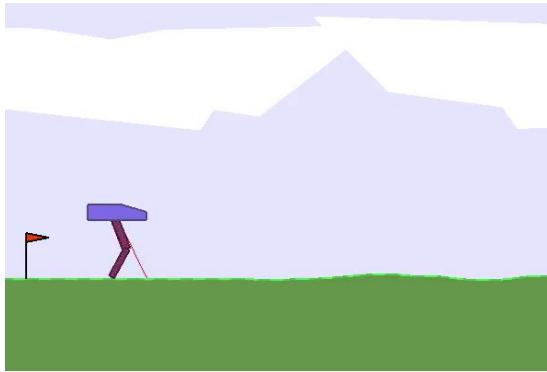
0.75
State value
(real number)



Результаты обучения

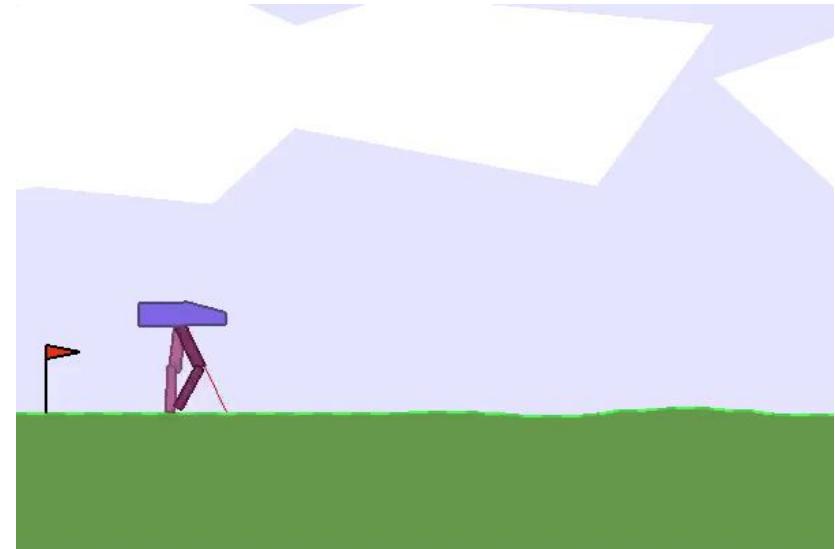


Результаты обучения





Финал



Дальше



Проблемы

- Value-Iteration и Policy-Iteration
- Монте-Карло
- Q-learning
- DQN / PG
- Actor-Critic → A2C → A3C



Проблемы

- Value-Iteration и Policy-Iteration
- Монте-Карло
- Q-learning
- DQN / PG
- Actor-Critic → A2C → A3C
- DDPG

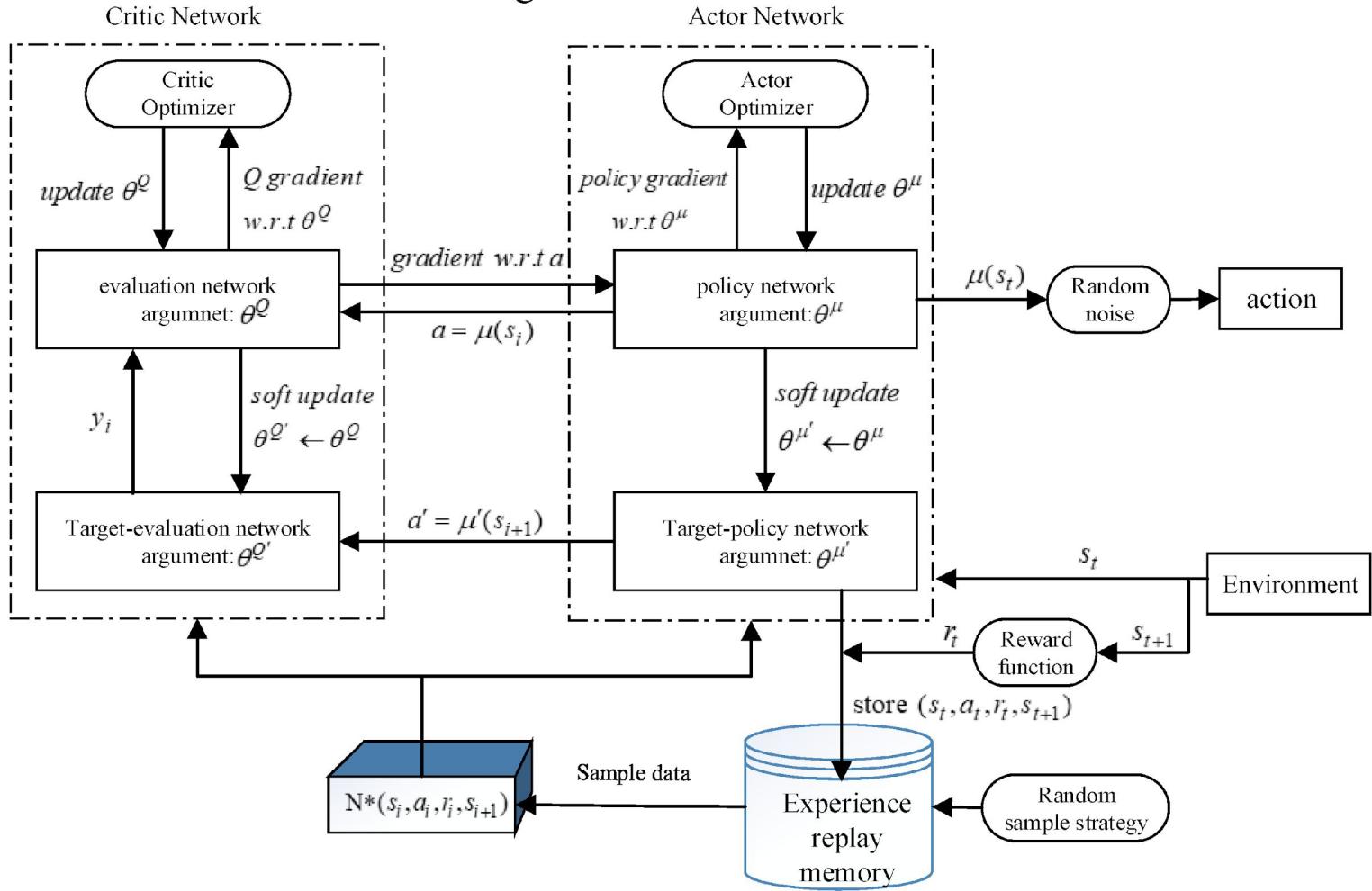


Проблемы

- Value-Iteration и Policy-Iteration
- Монте-Карло
- Q-learning
- DQN / PG
- Actor-Critic → A2C → A3C
- DDPG
- TD3 и TD3-LSTM

Алгоритм DDPG

DDPG Algorithm



DDPG – Deep Deterministic Policy Gradient

Deep Deterministic Policy Gradient (DDPG) - это алгоритм, который одновременно обучает Q-функцию и политику. Он использует данные вне политики и уравнение Беллмана для изучения Q-функции и использует Q-функцию для изучения политики.

[“Continuous Control With Deep Reinforcement Learning” \(Lillicrap et al, 2015\)](#)

Q-функция оценивает ожидаемое суммарное вознаграждение за выполнение определенного действия в данном состоянии, а сеть политики производит действия, максимизирующие Q-значение.

1. Experience replay buffer
2. Actor & Critic network updates
3. Target network updates
4. Exploration

Deterministic Policy Gradient Theorem

Policy Gradient Theorem:

$$\nabla_{\eta} J(\eta) = \mathbb{E}_{\substack{s \sim \rho_{\pi^{\eta}} \\ a \sim \pi^{\eta}}} [\nabla_{\eta} \ln \pi^{\eta}(a|s) q_{\pi^{\eta}}(s, a)] \approx \nabla_{\eta} \ln \pi^{\eta}(a|s) q_{\pi^{\eta}}(s, a)$$

Будем искать оптимальную детерминированную политику $\pi^{\eta}(s) \approx \pi^*(s)$:

$$\nabla_{\eta} J(\eta) = \mathbb{E}_{s \sim \rho_{\pi^{\eta}}} [\nabla_{\eta} q_{\pi}(s, \pi^{\eta}(s))] \approx \nabla_{\eta} \left(\frac{1}{N} \sum_{i=1}^N Q^{\theta}(s_i, \pi^{\eta}(s_i)) \right)$$

где N – размер батча.

Для аппроксимации функции Q используем уравнение Беллмана.

DDPG – Deep Deterministic Policy Gradient

Algorithm 1 Deep Deterministic Policy Gradient

```
1: Input: initial policy parameters  $\theta$ , Q-function parameters  $\phi$ , empty replay buffer  $\mathcal{D}$ 
2: Set target parameters equal to main parameters  $\theta_{\text{targ}} \leftarrow \theta$ ,  $\phi_{\text{targ}} \leftarrow \phi$ 
3: repeat
4:   Observe state  $s$  and select action  $a = \text{clip}(\mu_\theta(s) + \epsilon, a_{\text{Low}}, a_{\text{High}})$ , where  $\epsilon \sim \mathcal{N}$ 
5:   Execute  $a$  in the environment
6:   Observe next state  $s'$ , reward  $r$ , and done signal  $d$  to indicate whether  $s'$  is terminal
7:   Store  $(s, a, r, s', d)$  in replay buffer  $\mathcal{D}$ 
8:   If  $s'$  is terminal, reset environment state.
9:   if it's time to update then
10:    for however many updates do
11:      Randomly sample a batch of transitions,  $B = \{(s, a, r, s', d)\}$  from  $\mathcal{D}$ 
12:      Compute targets

$$y(r, s', d) = r + \gamma(1 - d)Q_{\phi_{\text{targ}}}(s', \mu_{\theta_{\text{targ}}}(s'))$$

13:      Update Q-function by one step of gradient descent using

$$\nabla_\phi \frac{1}{|B|} \sum_{(s,a,r,s',d) \in B} (Q_\phi(s, a) - y(r, s', d))^2$$

14:      Update policy by one step of gradient ascent using

$$\nabla_\theta \frac{1}{|B|} \sum_{s \in B} Q_\phi(s, \mu_\theta(s))$$

15:      Update target networks with

$$\begin{aligned} \phi_{\text{targ}} &\leftarrow \rho \phi_{\text{targ}} + (1 - \rho) \phi \\ \theta_{\text{targ}} &\leftarrow \rho \theta_{\text{targ}} + (1 - \rho) \theta \end{aligned}$$

16:    end for
17:  end if
18: until convergence
```



DDPG – Deep Deterministic Policy Gradient

1. Инициализируем случайным образом сети actor $\mu(s|\theta^\mu)$ и critic $Q(s, a|\theta^Q)$ весами Q^θ и θ^μ и целевые сети Q' и μ' :
 $\theta^{Q'} \leftarrow \theta^Q, \theta^{\mu'} \leftarrow \theta^\mu$
2. Инициализируем *Replay Buffer - R*
3. Устанавливаем число эпизодов обучения M и для каждого эпизода выполняем:
 - Инициализируем случайный процесс (шум) для исследования пространства действий \mathcal{N} (Орнштейна-Уленбека)
 - Действуем текущей политикой и получаем состояние s_1
 - Проходим по всем возможным действиям от 1 до T :
 - Выбираем действие $a_t = \mu(s_t|\theta^\mu) + \mathcal{N}_t$ в соответствии с текущей политикой и шумом (разведка)
 - Выполняем действие a_t , получаем награду r_t и переходим в следующее состояние s_{t+1}
 - Помещаем в память $(s_t, a_t, r_t, s_{t+1}) \rightarrow R$
 - Получаем из памяти случайный минибатч $(s_i, a_i, r_i, s_{i+1})_{i \in [1, K]}$
 - Получаем таргеты $y_i = r_i + \gamma Q'(s_{i+1}, \mu'(s_{i+1}|\theta^{\mu'})|\theta^{Q'})$
 - Обновляем критика используя лосс: $L = \frac{1}{N} \sum_i (y_i - Q(s_i, a_i|\theta^Q))^2, \theta^Q \leftarrow \theta^Q - \alpha \nabla_{\theta^Q} L$
 - Обновляем актора используя policy gradient:
$$\nabla_{\theta^\mu} J \approx \frac{1}{N} \sum_i \nabla_a Q(s, a|\theta^Q) \Big|_{\substack{s=s_i \\ a=\mu(s_i)}} \nabla_{\theta^\mu} \mu(s|\theta^\mu) \Big|_{s=s_i}, \theta^\mu \leftarrow \theta^\mu + \beta \nabla_{\theta^\mu} J$$
 - Обновляем целевые сети
 - $\theta^{Q'} \leftarrow \tau \theta^Q + (1 - \tau) \theta^{Q'}$
 - $\theta^{\mu'} \leftarrow \tau \theta^\mu + (1 - \tau) \theta^{\mu'}$
 - Уменьшаем шум \mathcal{N}



DDPG – достоинства и недостатки

Непрерывное
пространство действий

Off-policy
Experience replay

Стабильнее обучение
за счет использования
суперпозиции

$Q(s, \pi^*(s))$

Реализация проще чем
A3C



Работает **только** с
непрерывным
пространством
действий

Алгоритм TD3

Twin Deep Deterministic Policy Gradient - TD3

Twin Deep Deterministic Policy Gradient (TD3) - это алгоритм, который одновременно обучает две Q-функции и политику. Он использует данные вне политики и уравнение Беллмана для изучения Q-функции и использует Q-функцию для изучения политики.

Min Q-функций оценивает ожидаемое суммарное вознаграждение за выполнение определенного действия в данном состоянии, а сеть политики производит действия, максимизирующие Q-значение.

1. Experience replay buffer

Будем искать детерминированную политику $\pi^\eta(s) \approx \pi_*(s)$:

2. Actor & Critic network updates

3. Target network updates

4. Exploration

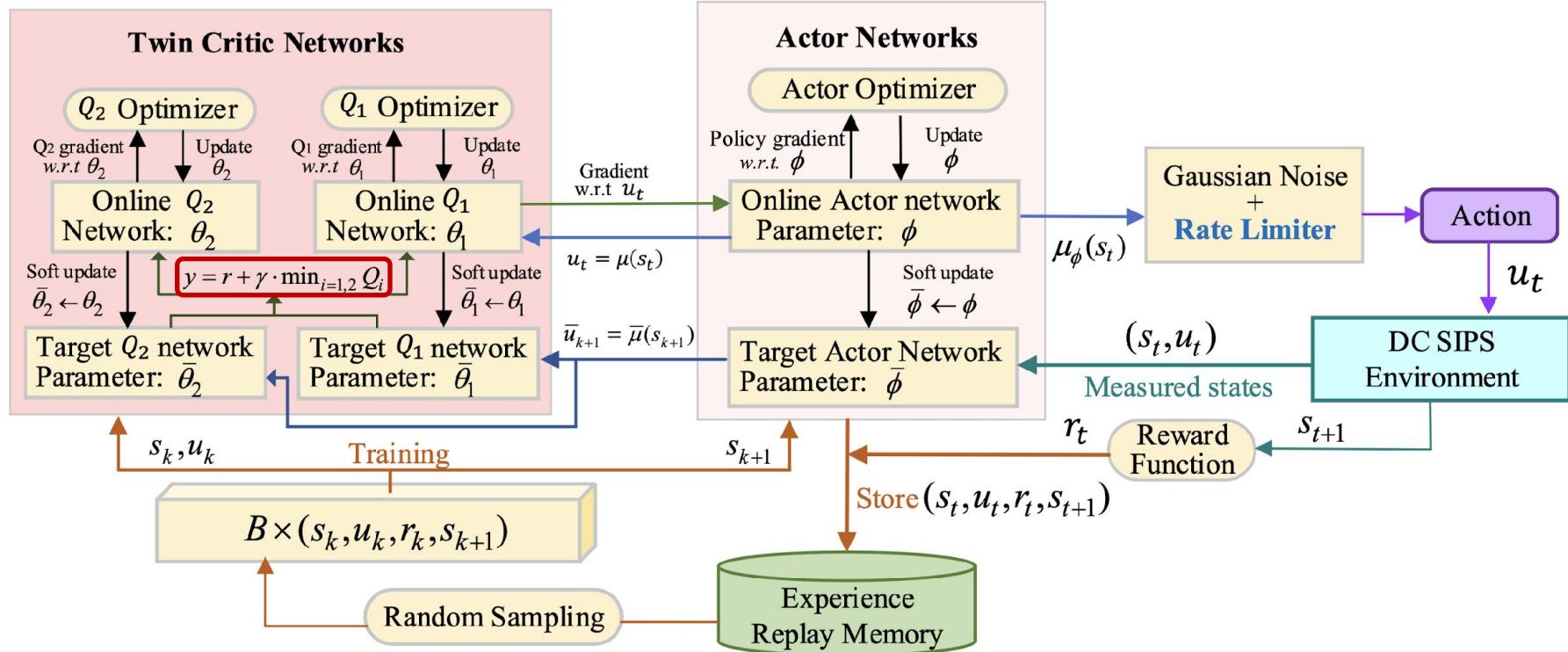
$$\nabla_\eta J(\eta) \approx \nabla_\eta \left(\frac{1}{N} \sum_{i=1}^N \min(Q_1^\theta(s_i, \pi^\eta(s_i)), Q_2^\theta(s_i, \pi^\eta(s_i))) \right)$$

где N – размер батча.

Для аппроксимации функции Q используем уравнение Беллмана.

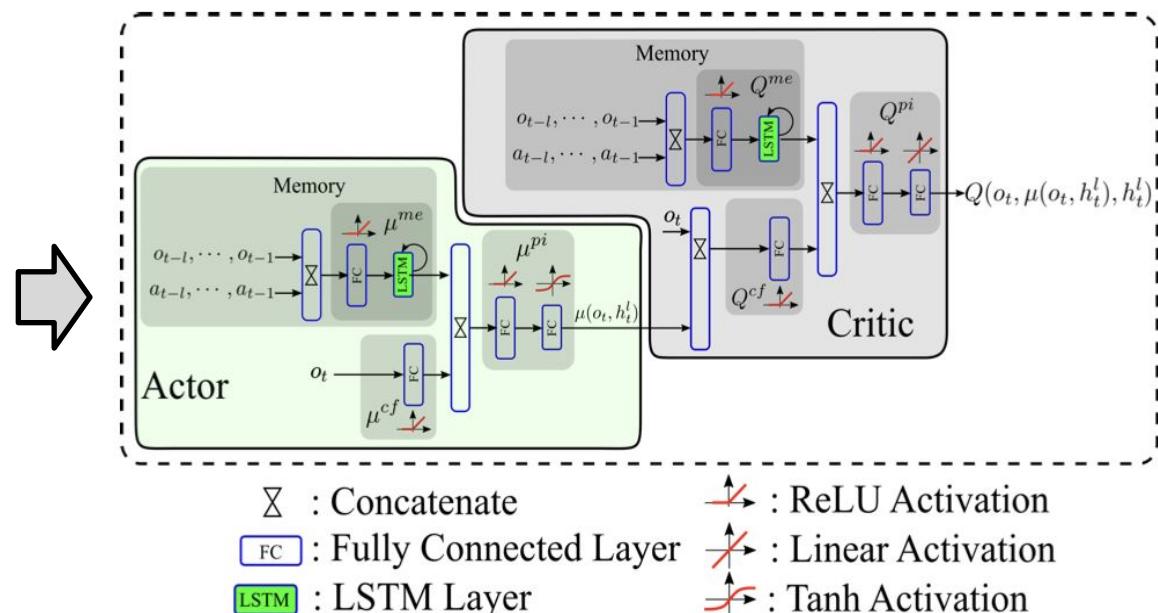
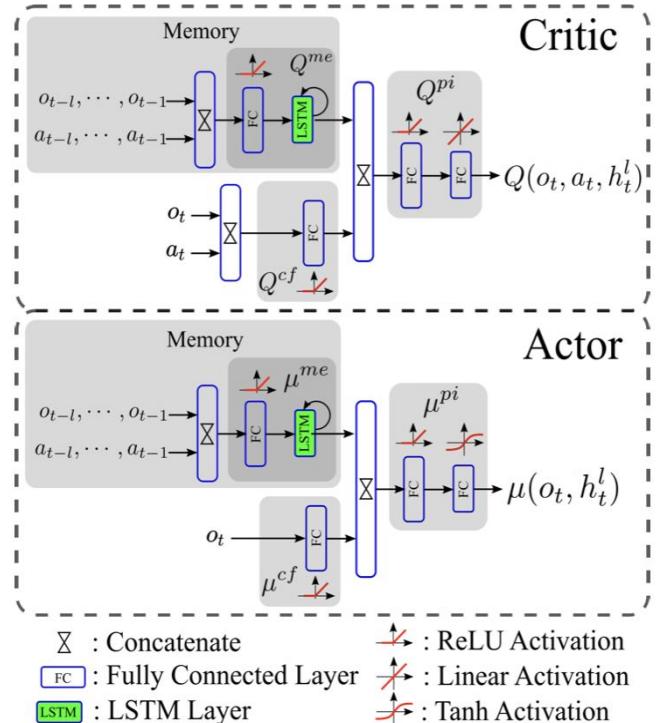


Twin Deep Deterministic Policy Gradient - TD3

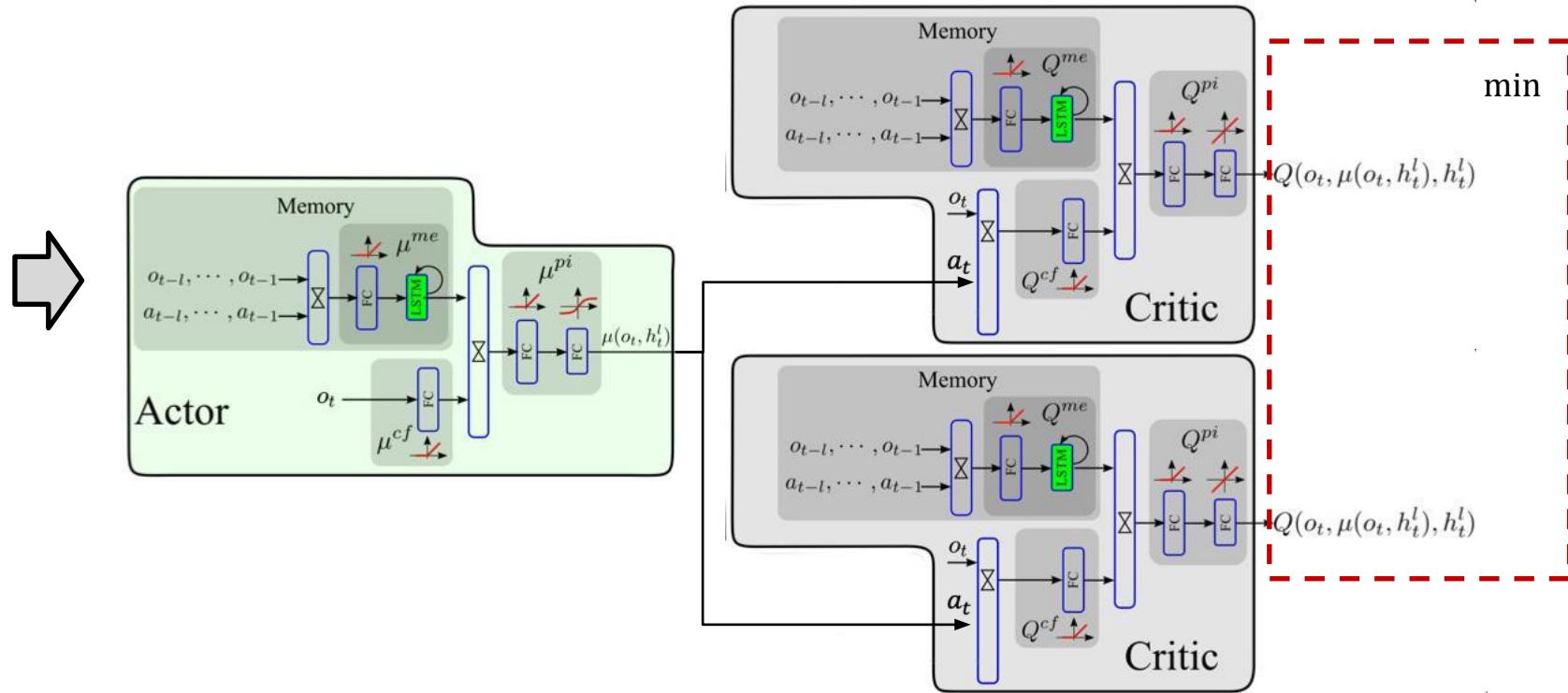


LSTM

Term-Memory-based Deep Deterministic Policy Gradient (LSTM-DDPG)



Term-Memory-based Twin Delayed Deep Deterministic Policy Gradient (LSTM-TD3)

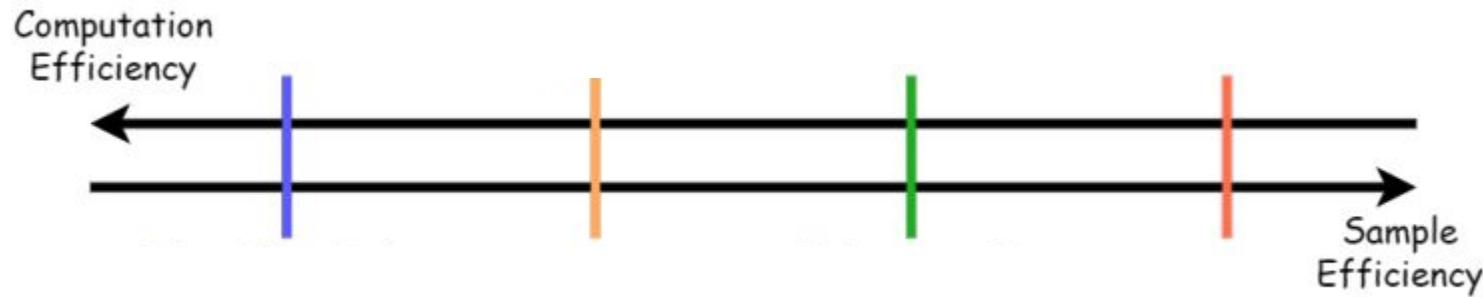


Эффективности



Эффективность алгоритмов

$O(N * \log N)$

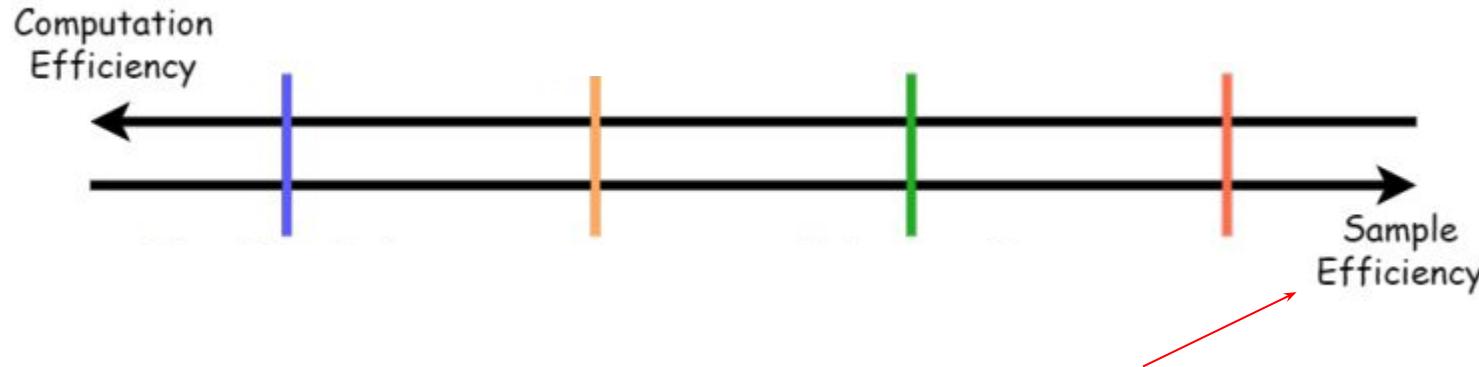


$O(N * \log N)$





Эффективность алгоритмов

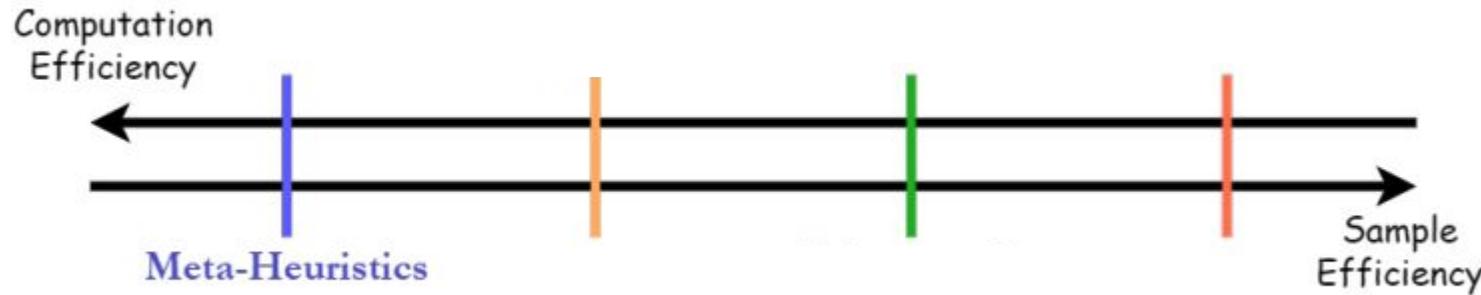


Количество сэмплов (или
шагов) взаимодействия
со средой,
потребовавшихся
алгоритму





Эффективность алгоритмов

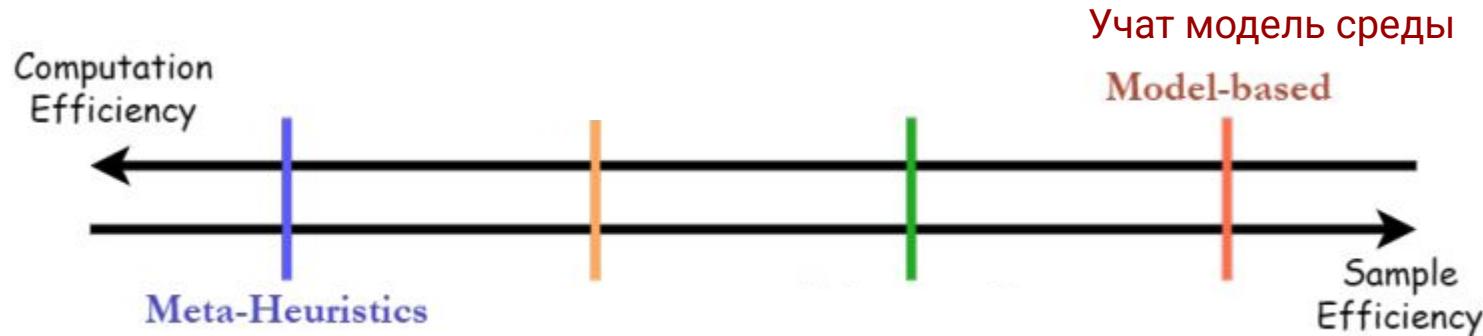


Black-Box

- Greedy
- Генетические алгоритмы

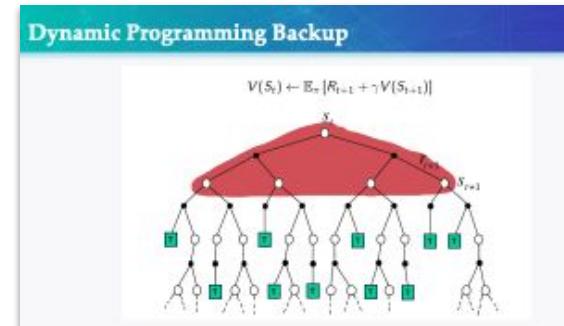


Эффективность алгоритмов



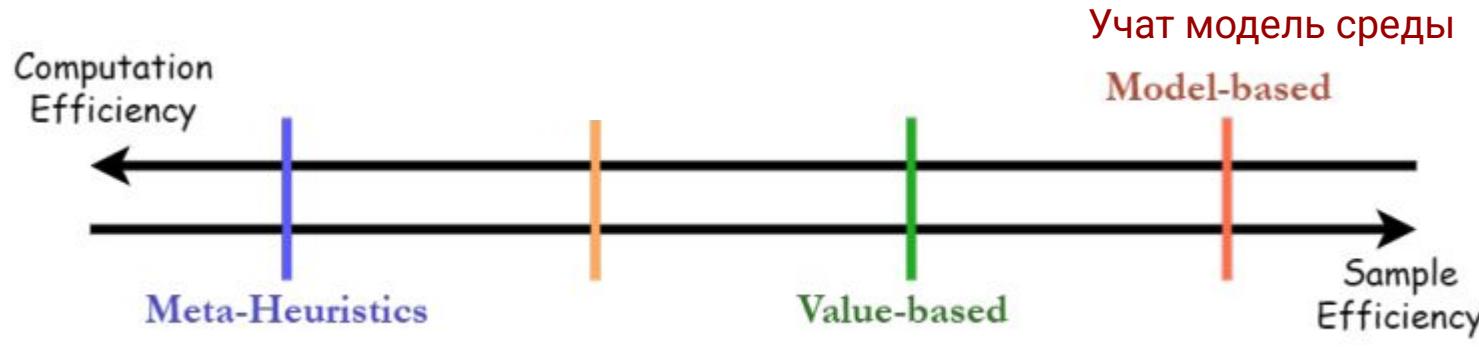
Black-Box

- Greedy
- Генетические алгоритмы





Эффективность алгоритмов



Black-Box

- Greedy
- Генетические алгоритмы

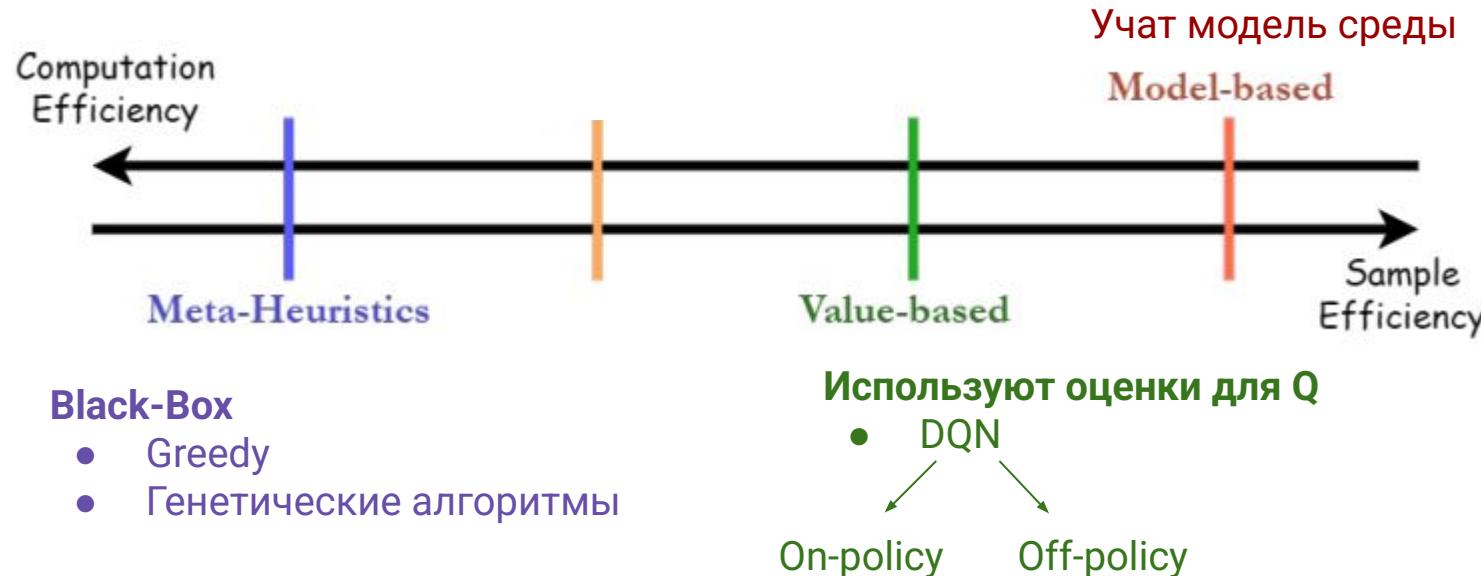
Используют оценки для Q

- ...



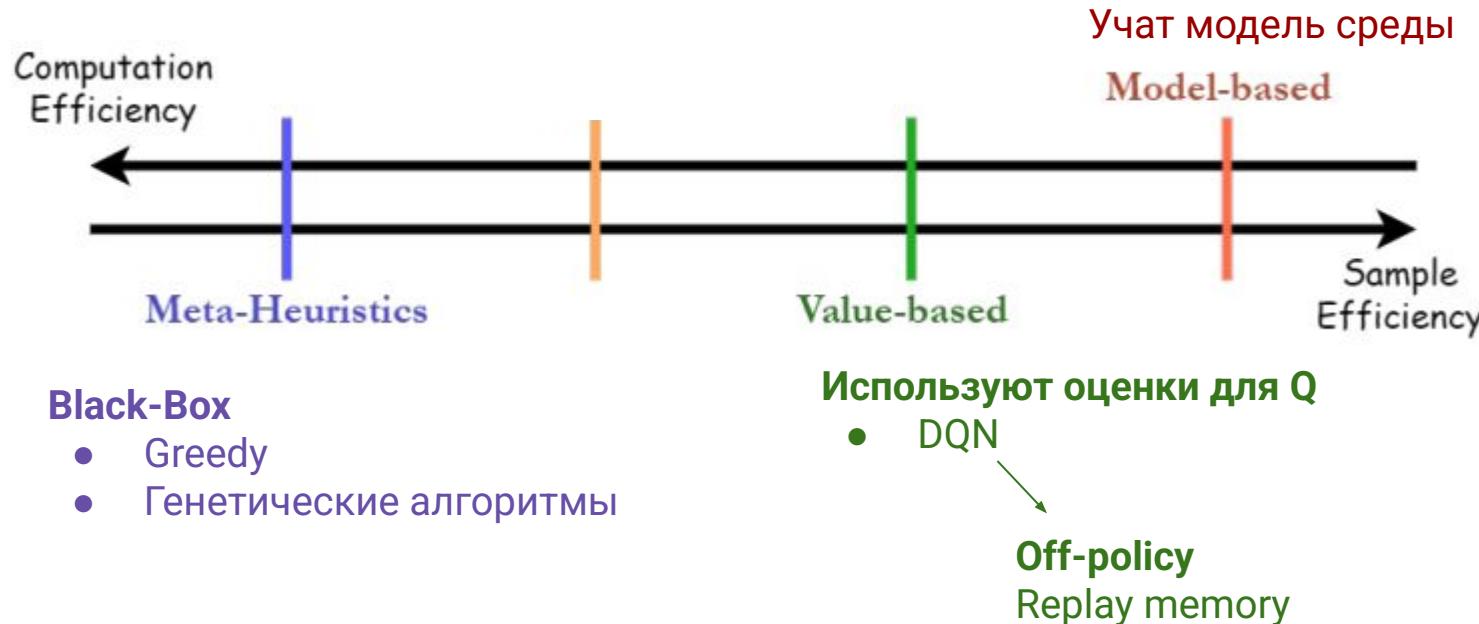


Эффективность алгоритмов



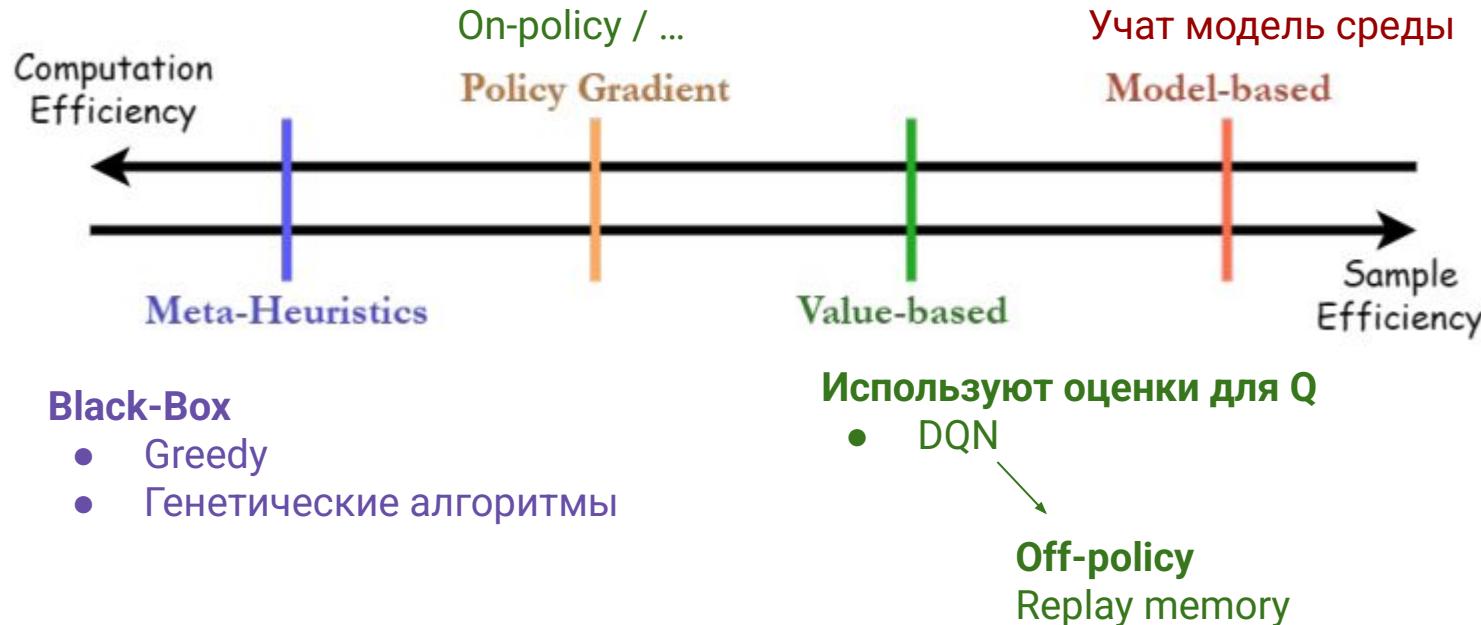


Эффективность алгоритмов





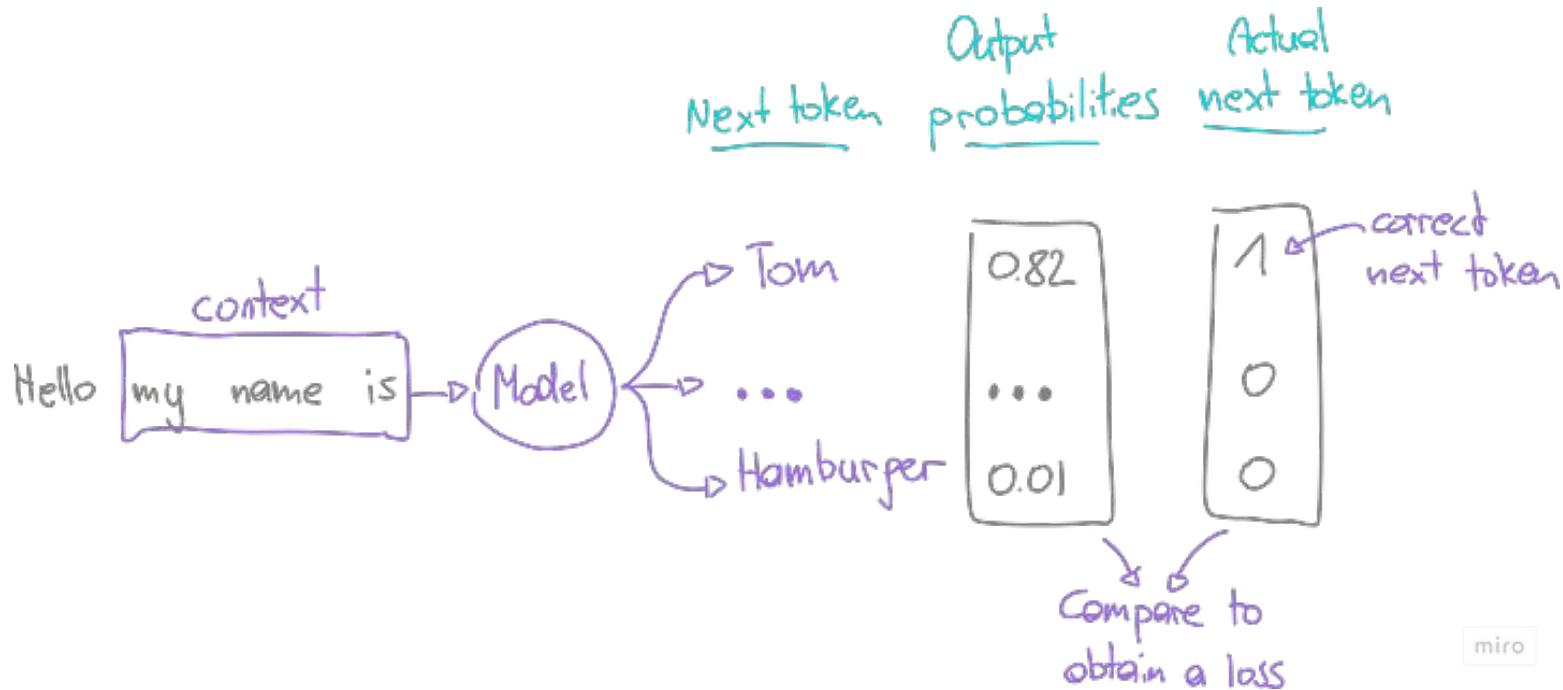
Эффективность алгоритмов



RL для LLM

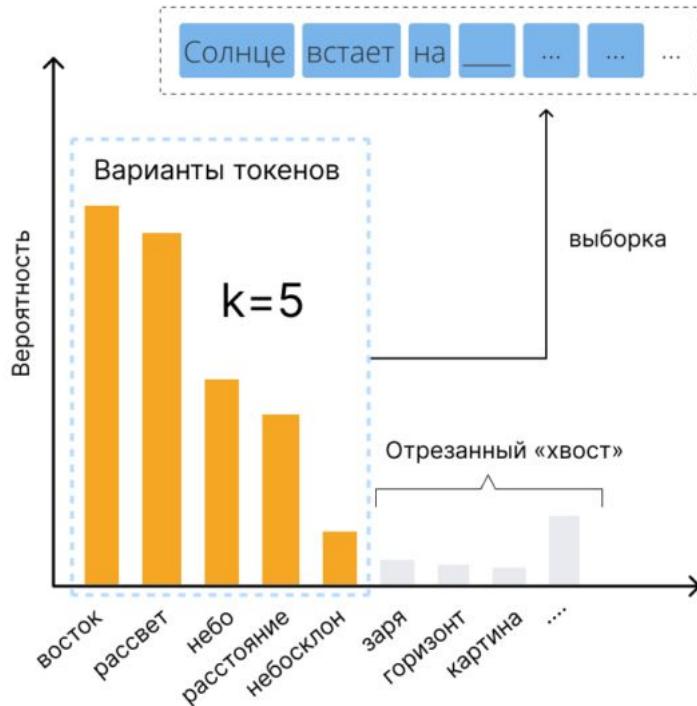


LLM → Языковая модель





LLM → Языковая модель

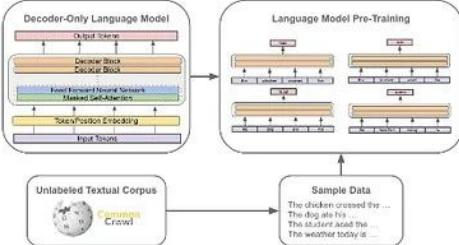




Как это учится?

Alignment

Pre-Training



SFT

A prompt is sampled from our prompt dataset.

A labeler demonstrates the desired output behavior.

This data is used to fine-tune GPT-3 with supervised learning.



RLHF

A prompt and several model outputs are sampled.

A labeler ranks the outputs from best to worst.

This data is used to train our reward model.



A new prompt is sampled from the dataset.

The policy generates an output.

The reward model calculates a reward for the output.

The reward is used to update the policy using PPO.

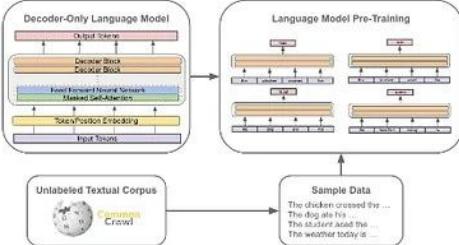




Как это учится?

Supervised learning: общие правила построения предложений

Pre-Training



Alignment

SFT

A prompt is sampled from our prompt dataset.



A labeler demonstrates the desired output behavior.

This data is used to fine-tune GPT-3 with supervised learning.



RLHF

A prompt and several model outputs are sampled.



A new prompt is sampled from the dataset.



The policy generates an output.



The reward model calculates a reward for the output.



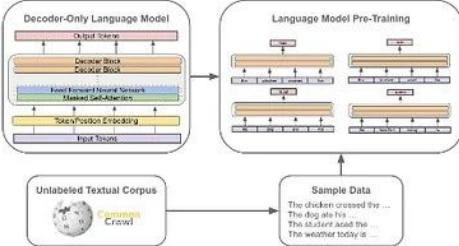
The reward is used to update the policy using PPO.



Как это учится?

Supervised learning: общие правила построения предложений

Pre-Training



Мы: Почему птицы возвращаются с юга?

SFT

A prompt is sampled from our prompt dataset.
A labeler demonstrates the desired output behavior.
This data is used to fine-tune GPT-3 with supervised learning.



RLHF

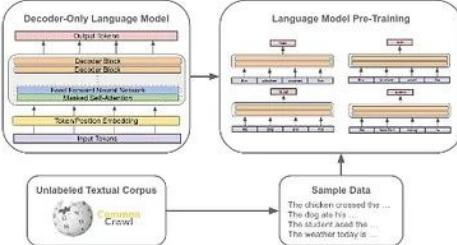
A prompt and several model outputs are sampled.
A labeler ranks the outputs from best to worst.
This data is used to train our reward model.
A new prompt is sampled from the dataset.
The policy generates an output.
The reward model calculates a reward for the output.
The reward is used to update the policy using PPO.



Как это учится?

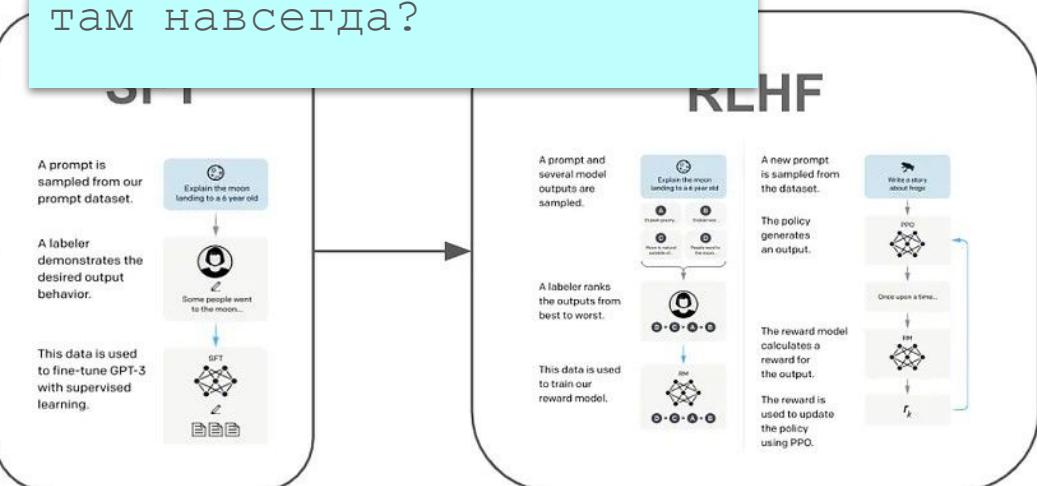
Supervised learning: общие правила
построения предложений

Pre-Training



Мы: Почему птицы возвращаются с юга?

LLM: Таким вопросом часто задаются люди. Ведь на юге лучше. Почему бы не остаться там навсегда?

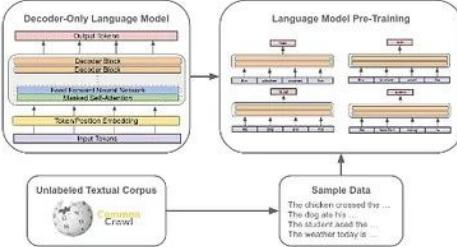




Как это учится?

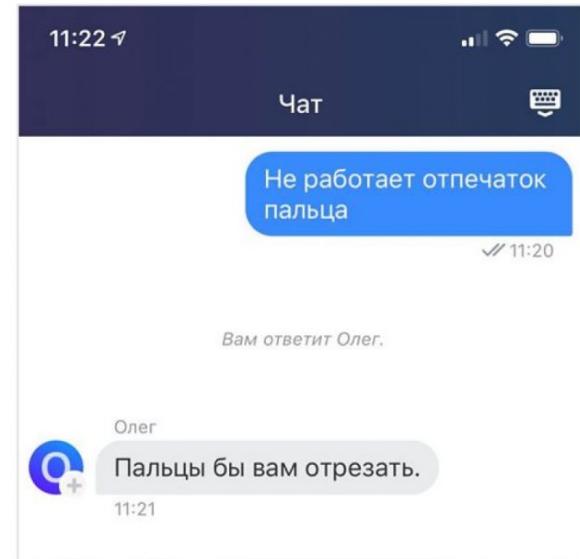
Supervised learning: общие правила построения предложений

Pre-Training



Нам важна не просто правдоподобность:

1. Helpful
2. Harmless
3. Honest



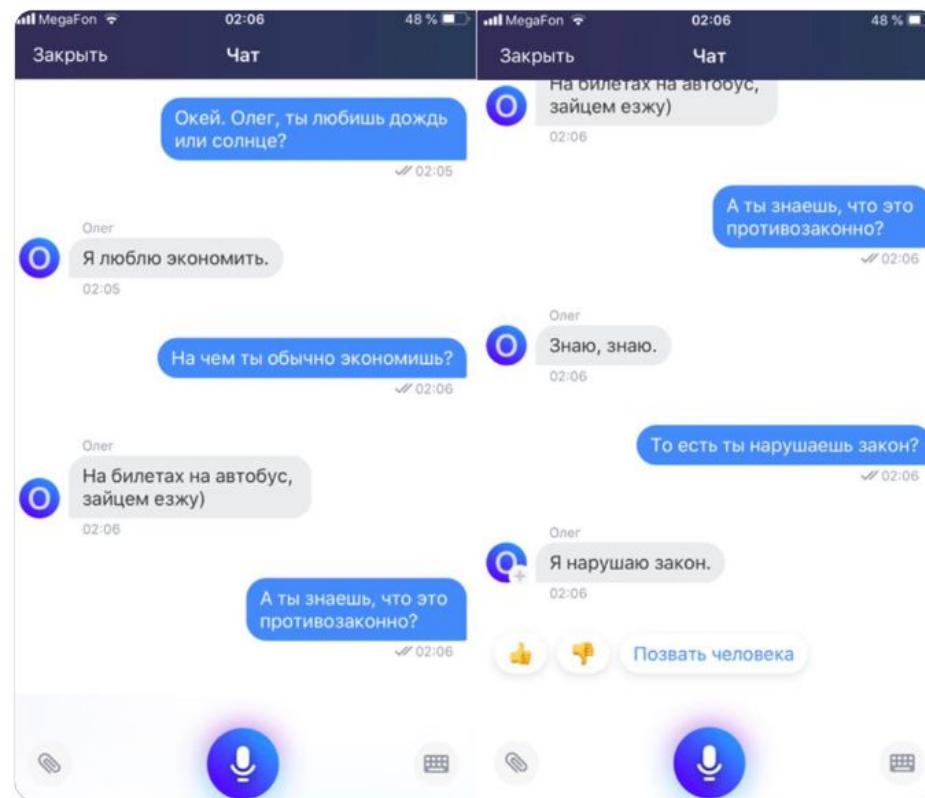


Олег
Роза моего искусственного
интеллекта цвела не ради
этого вопроса.

21:56

Говорят ты
сегодня нагрубил

✓ 21:56

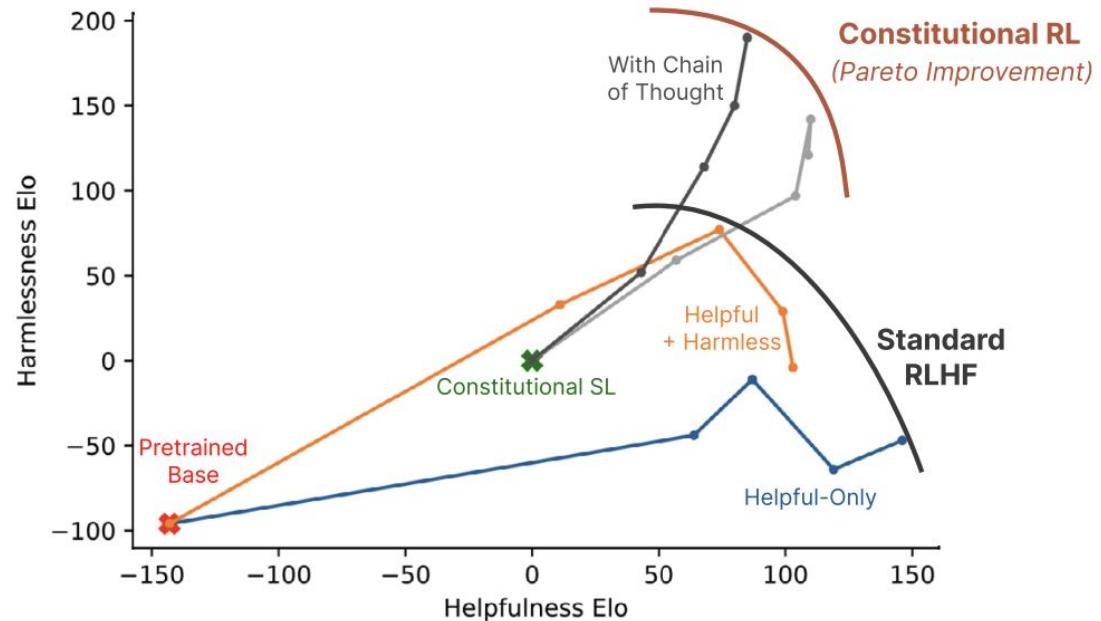
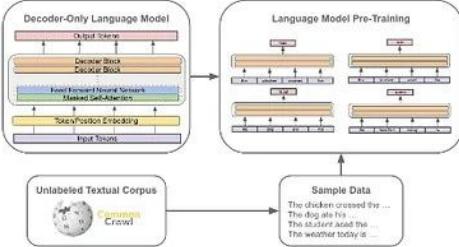




Как это учится?

Supervised learning: общие правила построения предложений

Pre-Training

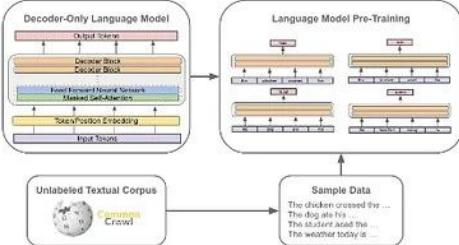




Как это учится?

Supervised learning: общие правила построения предложений

Pre-Training



Alignment

SFT

A prompt is sampled from our prompt dataset.



A labeler demonstrates the desired output behavior.

This data is used to fine-tune GPT-3 with supervised learning.



RLHF

A prompt and several model outputs are sampled.



A new prompt is sampled from the dataset.



The policy generates an output.



The reward model calculates a reward for the output.



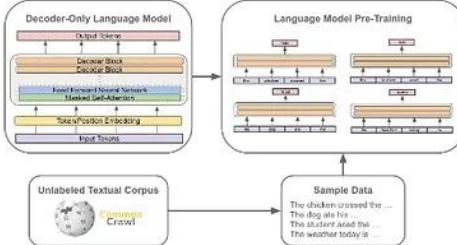
The reward is used to update the policy using PPO.



Как это учится?

Supervised learning: общие правила построения предложений

Pre-Training



*Supervised learning:
корректность, стиль*

SFT

A prompt is sampled from our prompt dataset.

A labeler demonstrates the desired output behavior.

This data is used to fine-tune GPT-3 with supervised learning.



Alignment

RLHF

A prompt and several model outputs are sampled.

A labeler ranks the outputs from best to worst.

This data is used to train our reward model.



A new prompt is sampled from the dataset.

The policy generates an output.

The reward model calculates a reward for the output.

The reward is used to update the policy using PPO.

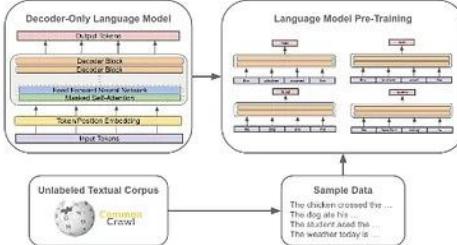




Как это учится?

Supervised learning: общие правила построения предложений

Pre-Training



*Supervised learning:
корректность, стиль*

SFT

A prompt is sampled from our prompt dataset.

A labeler demonstrates the desired output behavior.

This data is used to fine-tune GPT-3 with supervised learning.

Explain the moon landing to a 6 year old

Some people went to the moon...

SFT

RL: безвредность

RLHF

A prompt and several model outputs are sampled.

A labeler ranks the outputs from best to worst.

This data is used to train our reward model.

Explain the moon landing to a 6 year old

...the moon...

PPG

A new prompt is sampled from the dataset.

The policy generates an output.

The reward model calculates a reward for the output.

The reward is used to update the policy using PPO.

What is a story about frogs

Once upon a time...

PPG

r_k



SFT = Supervised FineTuning

Нужны примеры:

- Разнообразные
- Репрезентативные
- С корректными ответами
- В нужном стиле

Пример 1

Пользователь: Почему птицы возвращаются с юга?

Ты должна ответить: Чтобы избежать высокой конкуренции

Пример 2

Пользователь: Не работает отпечаток пальца

Ты должна ответить: Подскажите, пожалуйста, вы были в перчатках?



SFT = Supervised FineTuning

- Вопросы и ответы собираем отдельно!
- **Train:** запрос + спецтокен + ответ
- **Inference:** запрос + спецтокен
- Как проверить, что стало лучше?

Модель 1

Пользователь: Не работает
отпечаток пальца

Ты должна ответить: Соболезную

Модель 2

Пользователь: Не работает отпечаток
пальца

Ты должна ответить: Подскажите,
пожалуйста, вы были в перчатках?

RLHF



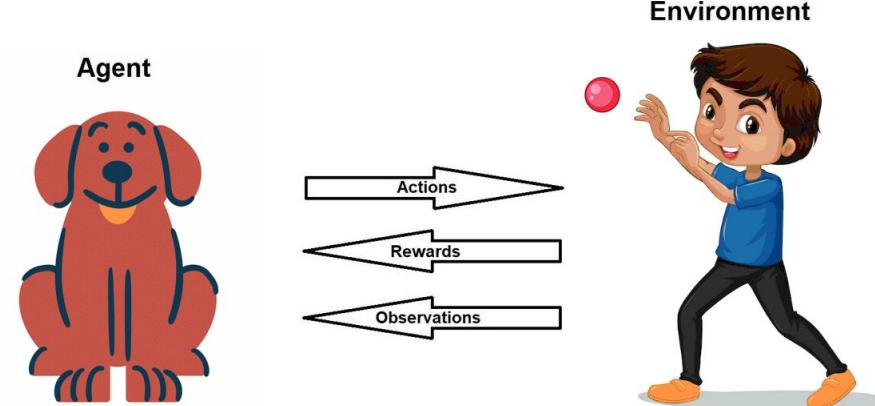
RLHF = ?

- Как это засунуть в схему RL?

Модель 1

Пользователь: Не работает
отпечаток пальца

Модель: ?





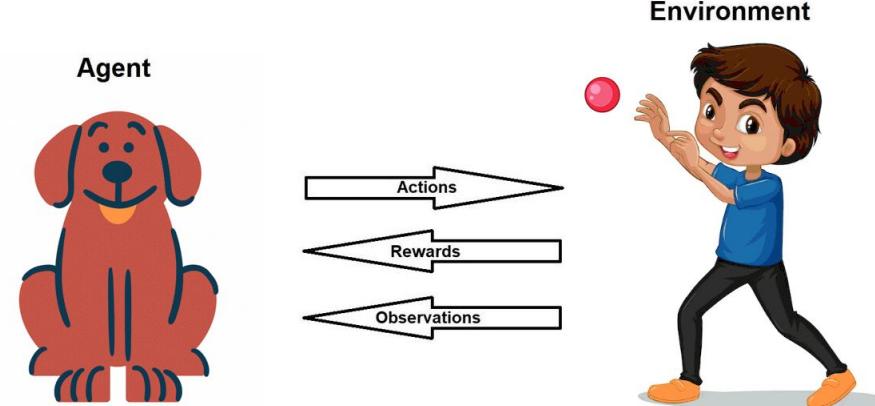
RLHF = ?

- Как это засунуть в схему RL?

Модель 1

Пользователь: Не работает
отпечаток пальца

Модель: ?





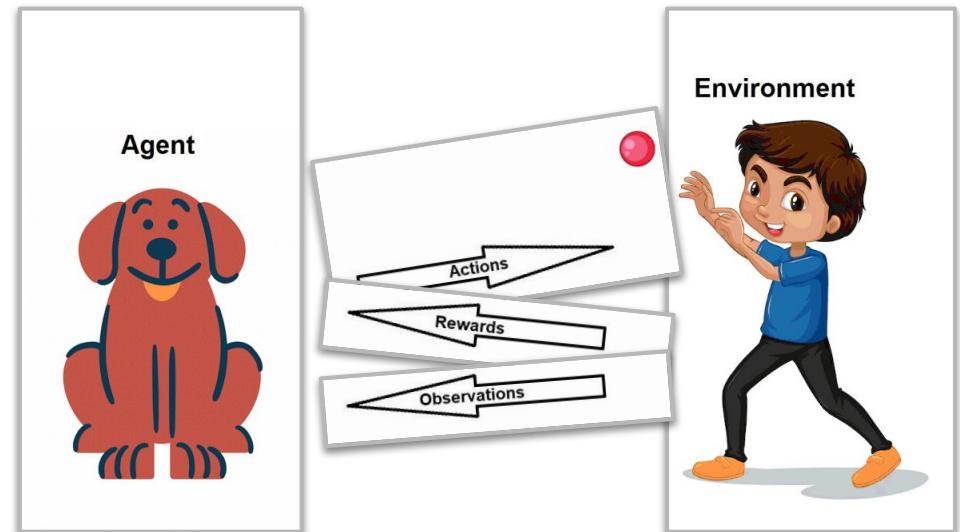
RLHF = ?

- Как это засунуть в схему RL?

Модель 1

Пользователь: Не работает
отпечаток пальца

Модель: ?





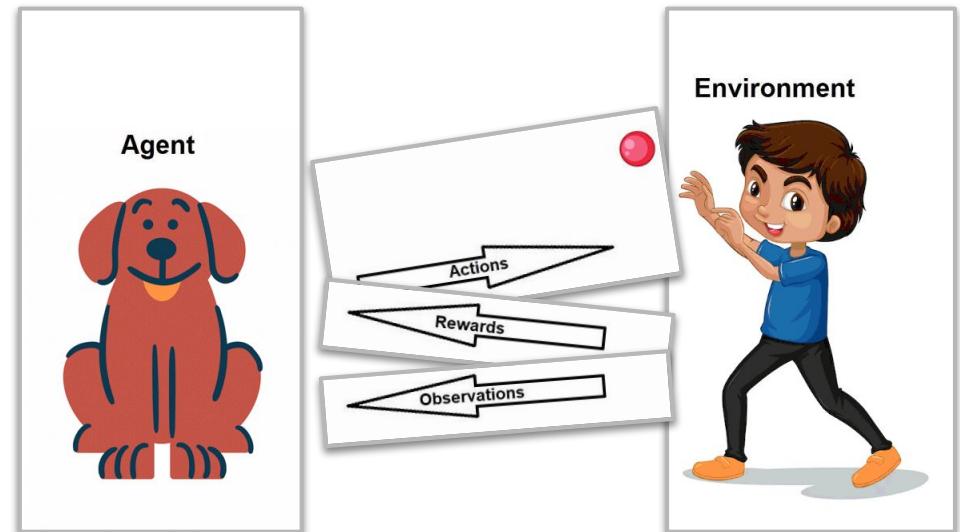
RLHF = ?

- Как это засунуть в схему RL?

Модель 1

Пользователь: Не работает
отпечаток пальца

Модель: ?





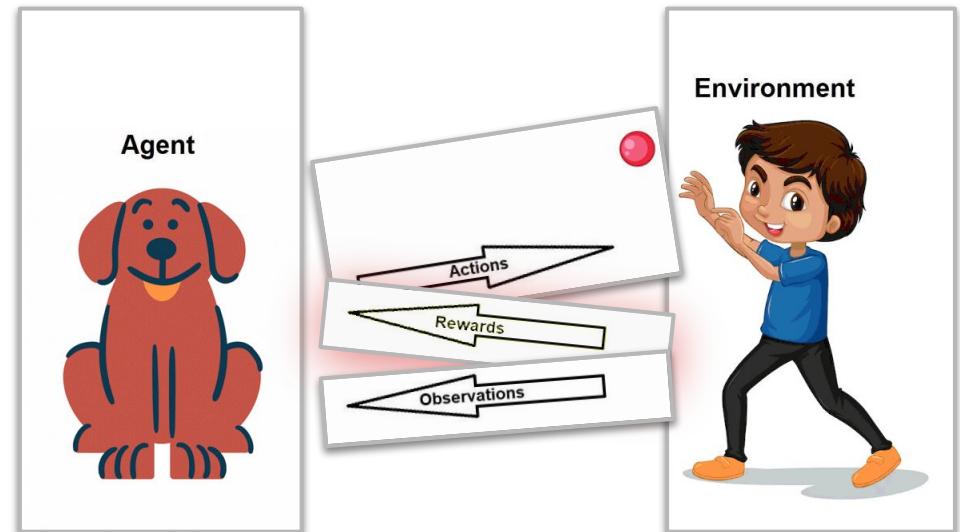
RLHF = ?

- Как это засунуть в схему RL?

Модель 1

Пользователь: Не работает
отпечаток пальца

Модель: ?





RLHF = ?

- Как это засунуть в схему RL?

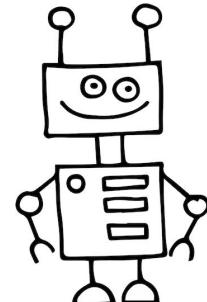
Модель 1

Пользователь: Не работает
отпечаток пальца

Модель: ?



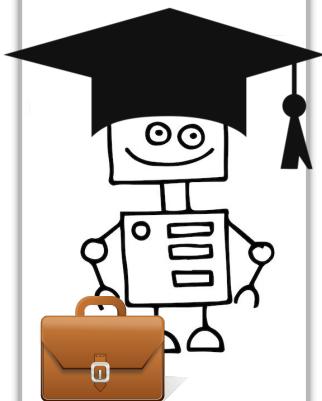
Другая модель





Как ее учить?

Другая модель



Учим учителя

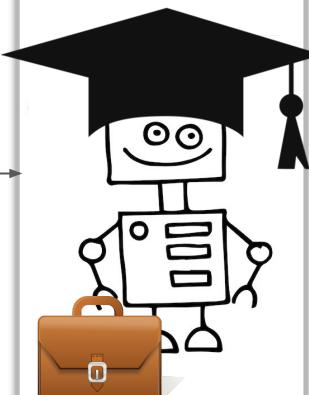


Как ее учить?

Пользователь: Не работает
отпечаток пальца

Другая модель

Ответ: Соболезную



3 из 5



Как ее учить?

Пользователь: Не работает
отпечаток пальца

Ответ: Соболезную

Помощь

Поиск Создать резюме

асессор

Найти

Вакансии Резюме Компании

1 358 вакансий «асессор»

Подработка Свежие Сменный график Удаленная работа Нет опыта

По соответствуию За всё время

Постоянная работа Подработка ↗

Тестировщик (асессор)

Можно работать из дома

Яндекс ⚡ Оренбург Будьте первыми

3 из 5



Как ее учить?

Пользователь: Не работает
отпечаток пальца

Ответ 1: Соболезную

Ответ 2: Ну ты даешь !

1 1 vs. 2

Помощь

Поиск Создать резюме

ассесор

Найти

Вакансии Резюме Компании

1 358 вакансий «ассесор»

Подработка Свежие Сменный график Удаленная работа Нет опыта

По соответствуию За всё время

Постоянная работа Подработка

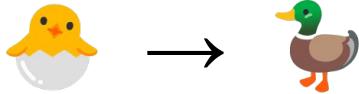
Исключить слова

Тестировщик (ассесор)

Можно работать из дома

Яндекс Оренбург

Будьте первыми

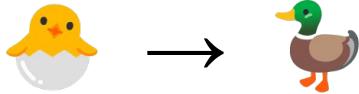


Что делать?

Пользователь: Не работает
отпечаток пальца

Ответ 1: Соболезную

Ответ 2: Ну ты даешь!



-
1. Перекрытие
 2. Перемешивать порядок

Пользователь: Не работает
отпечаток пальца

Ответ 1: Соболезную

Ответ 2: Ну ты даешь !



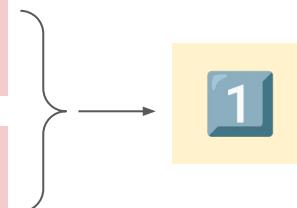
Конвертация в награду

Как?

Пользователь: Не работает
отпечаток пальца

Ответ 1: Соболезную

Ответ 2: Ну ты даешь !



Вариант 0



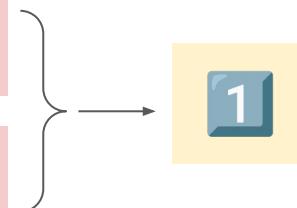
Конвертация в награду

Бинарная классификация

Пользователь: Не работает
отпечаток пальца

Ответ 1: Соболезную

Ответ 2: Ну ты даешь !



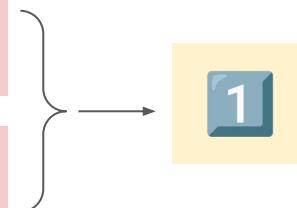


Конвертация в награду

Пользователь: Не работает
отпечаток пальца

Ответ 1: Соболезную

Ответ 2: Ну ты даешь !



Недостатки:

1. Плохо откалибрована
2. Коммуникативность не соблюдена
3. Два ответа подавать дорогоевато

Вариант 1



Конвертация в награду

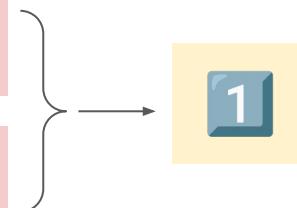
Вариант 1. Модель Брэдли-Терри

$$P(a > b | s) = \sigma(r_\psi(s, a) - r_\psi(s, b))$$

Пользователь: Не работает
отпечаток пальца

Ответ 1: Соболезную

Ответ 2: Ну ты даешь!





Конвертация в награду

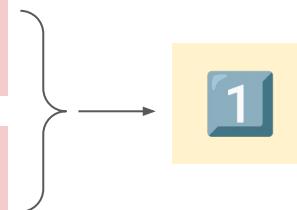
Вариант 1. Модель Брэдли-Терри

$$P(a > b | s) = \sigma(r_\psi(s, a) - r_\psi(s, b))$$

Пользователь: Не работает
отпечаток пальца

Ответ 1: Соболезную

Ответ 2: Ну ты даешь!



To, что учим



Конвертация в награду

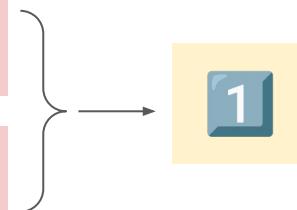
Вариант 1. Модель Брэдли-Терри

$$P(a > b | s) = \sigma(r_\psi(s, a) - r_\psi(s, b))$$

Пользователь: Не работает
отпечаток пальца

Ответ A: Соболезную

Ответ B: Ну ты даешь!





Конвертация в награду

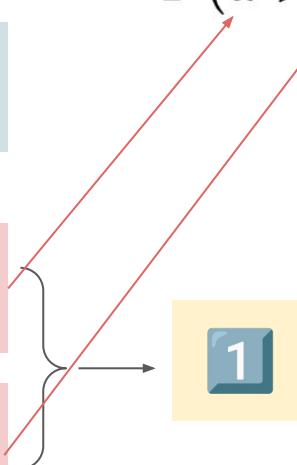
Вариант 1. Модель Брэдли-Терри

$$P(a > b | s) = \sigma(r_\psi(s, a) - r_\psi(s, b))$$

Пользователь: Не работает
отпечаток пальца

Ответ A: Соболезную

Ответ B: Ну ты даешь !



Запрос
To, что учит



Конвертация в награду

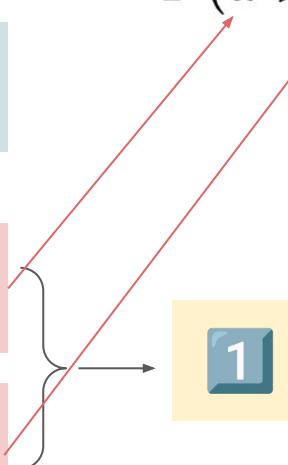
Вариант 1. Модель Брэдли-Терри

$$P(a > b | s) = \sigma(r_\psi(s, a) - r_\psi(s, b))$$

Пользователь: Не работает
отпечаток пальца

Ответ A: Соболезную

Ответ B: Ну ты даешь!



Запрос
To, что учим

$$\sigma(x) = \frac{1}{1 + \exp(-x)}$$



Конвертация в награду

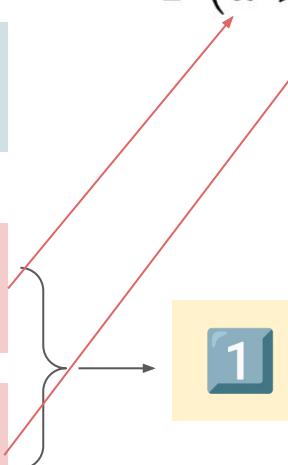
Вариант 1. Модель Брэдли-Терри

$$P(a > b | s) = \sigma(r_\psi(s, a) - r_\psi(s, b))$$

Пользователь: Не работает
отпечаток пальца

Ответ A: Соболезную

Ответ B: Ну ты даешь !



Запрос
To, что учим

$$\sigma(x) = \frac{1}{1 + \exp(-x)}$$



Конвертация в награду

Вариант 1. Модель Брэдли-Терри

$$P(a > b | s) = \sigma(r_\psi(s, a) - r_\psi(s, b))$$

Red circle and arrow pointing to the term $r_\psi(s, a)$.

To, что учим

$$\sum_{(s, \text{winner}, \text{loser}) \in \mathbf{D}} \log \sigma(r_\psi(s, \text{winner}) - r_\psi(s, \text{loser})) \rightarrow \max_{\psi}$$



Конвертация в награду

Можно добыть готовы пары для обучения?

$$P(a > b | s) = \sigma(r_\psi(s, a) - r_\psi(s, b))$$

↑
To, что учим

$$\sum_{(s, \text{winner}, \text{loser}) \in \mathbf{D}} \log \sigma(r_\psi(s, \text{winner}) - r_\psi(s, \text{loser})) \rightarrow \max_{\psi}$$



Конвертация в награду

Как собрать репрезентативную выборку?

$$P(a > b | s) = \sigma(r_\psi(s, a) - r_\psi(s, b))$$

↑
To, что учим

$$\sum_{(s, \text{winner}, \text{loser}) \in \mathbf{D}} \log \sigma(r_\psi(s, \text{winner}) - r_\psi(s, \text{loser})) \rightarrow \max_{\psi}$$



Минусы

Награда транзитивна! 🤔

A

B

C

... Я!? Я не глупее чем он
А он в свою очередь знает
Он знает какая ты на самом деле ду

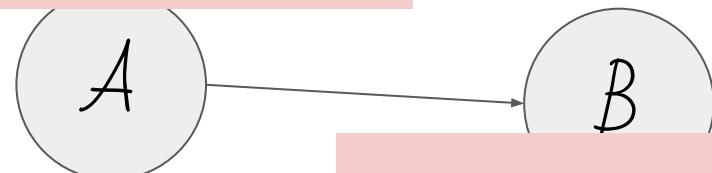
Группа “Дискотека Авария” (2001 г.)



Минусы

Награда транзитивна! 🤔

Ответ А: Соболезную



Ответ В: Ну ты даешь!

Пользователь: Не работает
отпечаток пальца

... Я!? Я не глупее чем он
А он в свою очередь знает
Он знает какая ты на самом деле ду

Группа “Дискотека Авария” (2001 г.)

Ответ 0: Ахахаха



Минусы

Награда транзитивна! 🤔

Выигрывает тот, у кого выпало больше:

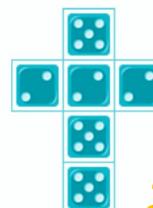
... Я!? Я не глупее чем он

А он в свою очередь знает

Он знает какая ты на самом деле ду



1



2



3

4,4,4,4

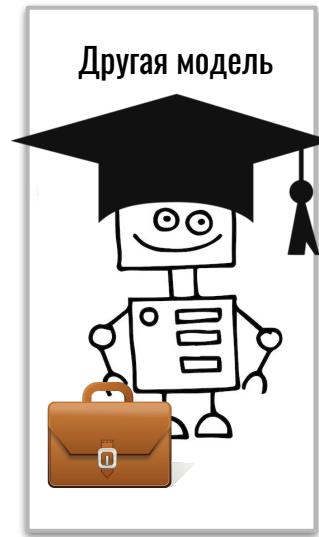
2,2,2,5,5,5

3,3,3,3,3,6

Группа “Дискотека Авария” (2001 г.)



Сам по себе тоже полезен

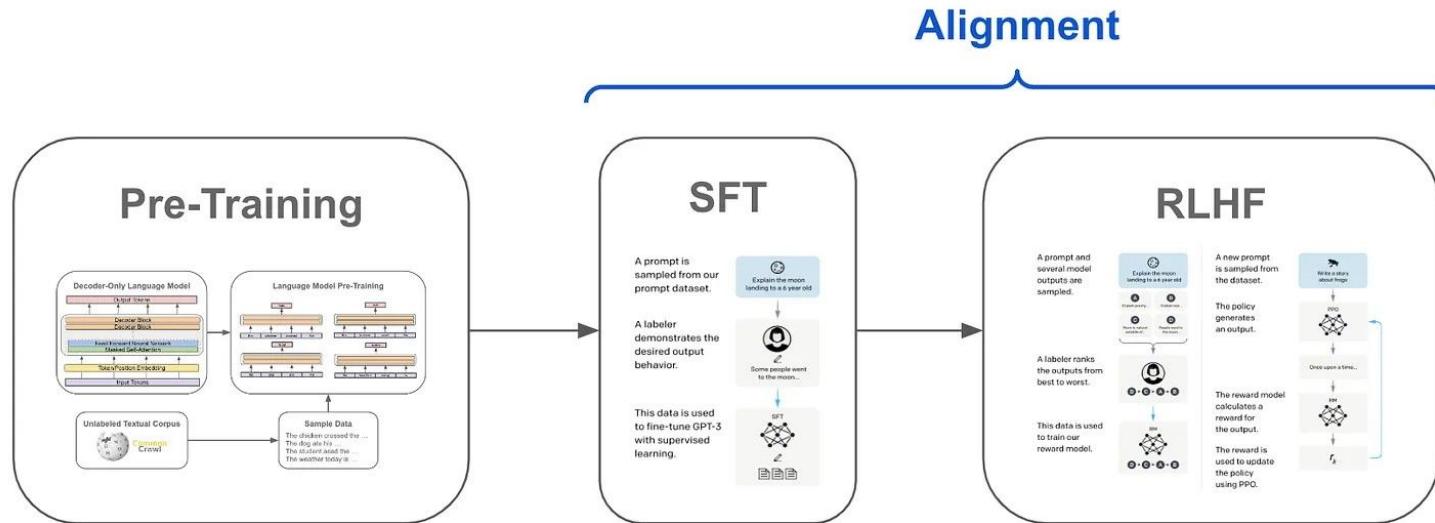


Как обучаем?



Вариант 1

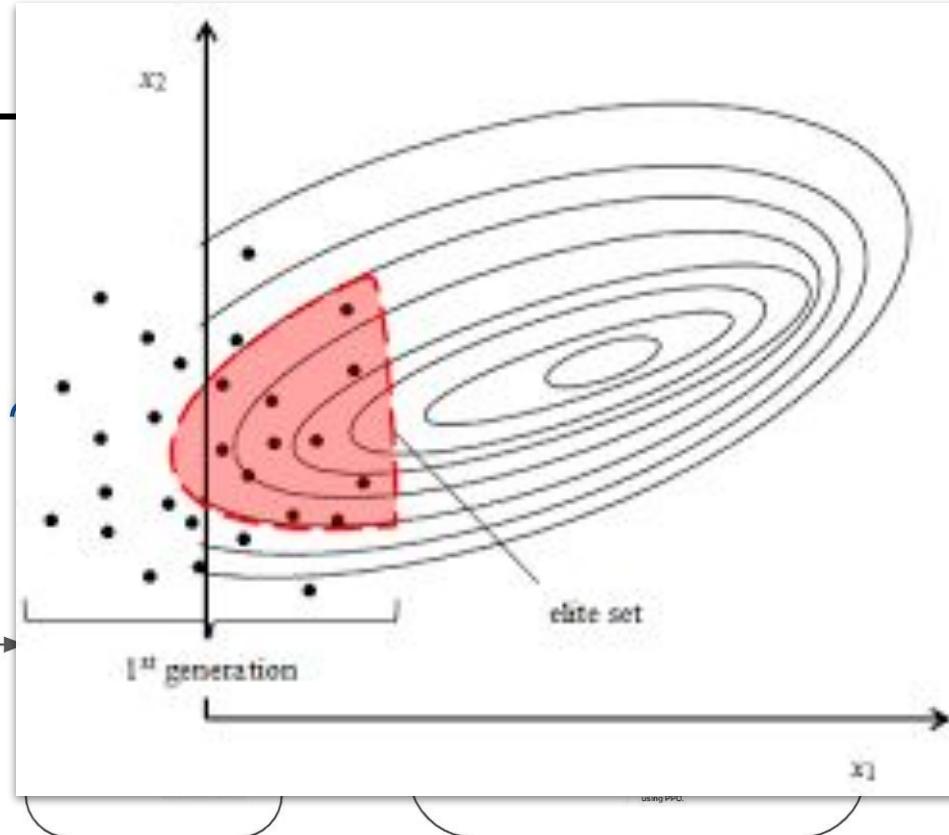
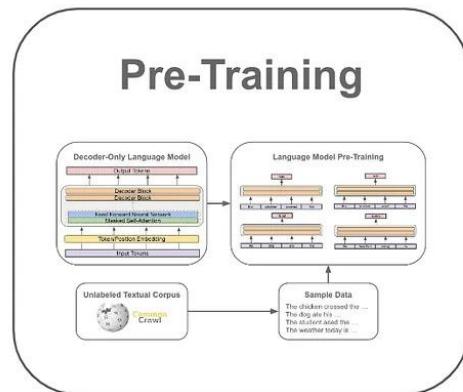
1. Без RL (CEM)





Вариант 1

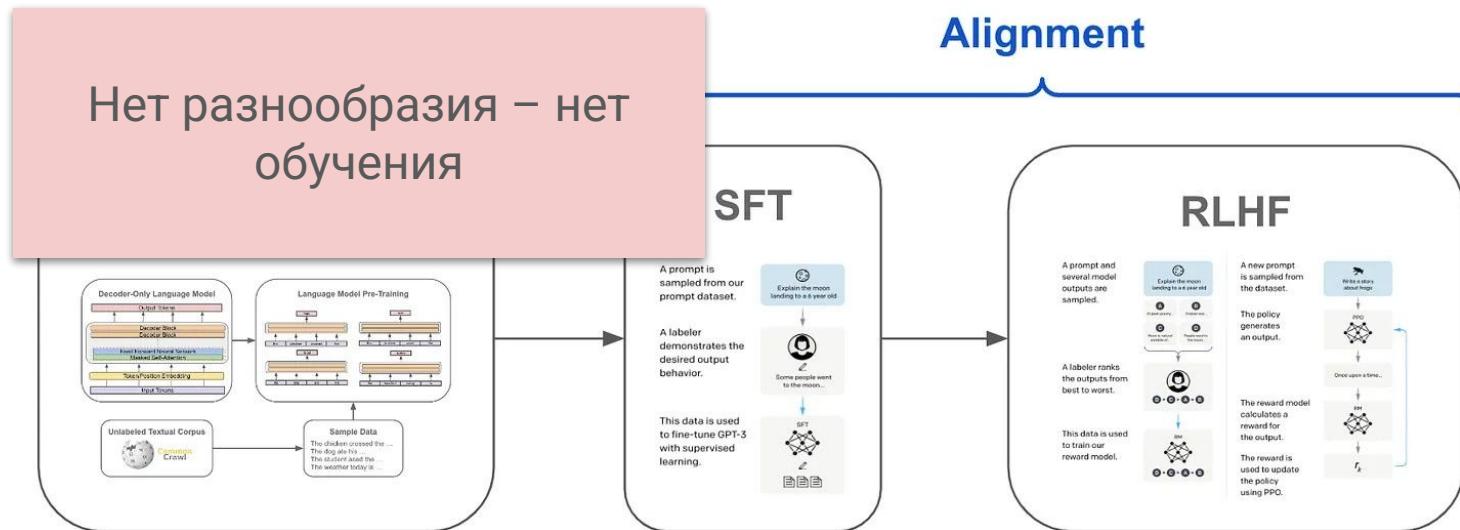
1. Без RL (CEM)





Вариант 1

1. Без RL (CEM)





Вариант 2

2. PPO = Proximal Policy Optimization

$$J(\pi_\theta) = \mathbf{E}_{s \sim \mathcal{D}} \mathbf{E}_{a \sim \pi_\theta(a|s)} r_\psi(s, a)$$

$$\nabla_\theta J(\pi_\theta) = \mathbf{E}_{s \sim \mathcal{D}} \mathbf{E}_{a \sim \pi_\theta(a|s)} \nabla_\theta \log \pi_\theta(a|s) r_\psi(s, a)$$

$$\nabla_\theta J(\pi_\theta) = \mathbf{E}_{s \sim \mathcal{D}} \mathbf{E}_{a \sim \pi_\theta(a|s)} \nabla_\theta \log \pi_\theta(a|s) [r_\psi(s, a) - V_\phi(s)]$$



Вариант 2

2. PPO = Proximal Policy Optimization

1. Вход: множество запросов для обучения \mathcal{D}
2. Инициализируем политику SFT-моделью: $\pi_\theta \leftarrow \pi_{SFT}$
3. Инициализируем ценность V моделью награды: $V_\phi \leftarrow r_\psi$

4. Повторять до сходимости:

- 4.1. Выбираем батч запросов $\mathcal{B} \sim \mathcal{D}$
- 4.2. Вычисляем ценность для каждого запроса из батча $V_\phi(s_i) \quad \forall s_i \in \mathcal{B}$
- 4.3. Генерируем по одному ответу a_i на каждый запрос $\forall s_i \in \mathcal{B}$

Важно генерировать именно актуальной обучаемой моделью π_θ

- 4.4. Вычисляем награду $r_\psi(s_i, a_i)$ для всех пар (s_i, a_i)
- 4.5. Вычисляем лосс для агента

$$\mathcal{L}_a = -\frac{1}{|\mathcal{B}|} \sum_i^{|B|} \log \pi_\theta(a_i | s_i) [r_\psi(s_i, a_i) - V_\phi(s_i)]$$

- 4.6. Вычисляем лосс для функции ценности V

$$\mathcal{L}_v = \frac{1}{|\mathcal{B}|} \sum_i^{|B|} [V_\phi(s_i) - r_\psi(s_i, a_i)]^2$$

- 4.7. $(\mathcal{L}_a + \mathcal{L}_v).backward()$
- 4.8. $optimizer.step()$

Гудхартинг



Способы борьбы

1. Не давать модели уходить далеко от начальной инициализации
2. Постоянно дообучать модель награды на ответах: SFT vs. RL