

Exploration + action-less

18 october 2025

О чём сегодня поговорим?

- Классический online RL:
 - ◆ Долго
 - ◆ Трудно
 - ◆ Дорого
- Обучение с помощью исследования
 - ◆ RND
 - ◆ ICM
- Обучение по видео
 - ◆ IDM
 - ◆ LAPO



Часть 1. Обучение через исследование

Self-supervision in Deep Learning

Когда нет разметки для обучения с учителем, придумываем какой-то таск, который можно учить без разметки!

А потом дотрениваемся на разметке чуть-чуть

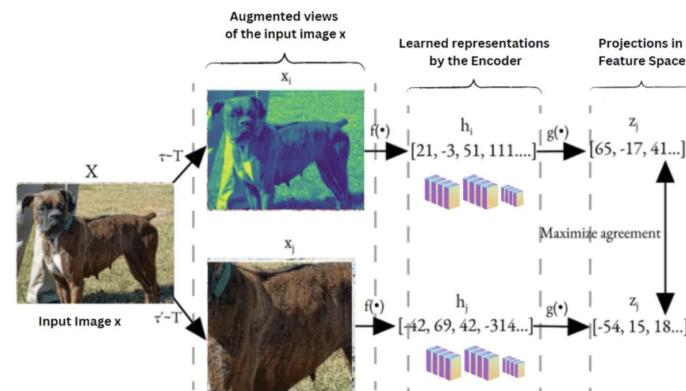
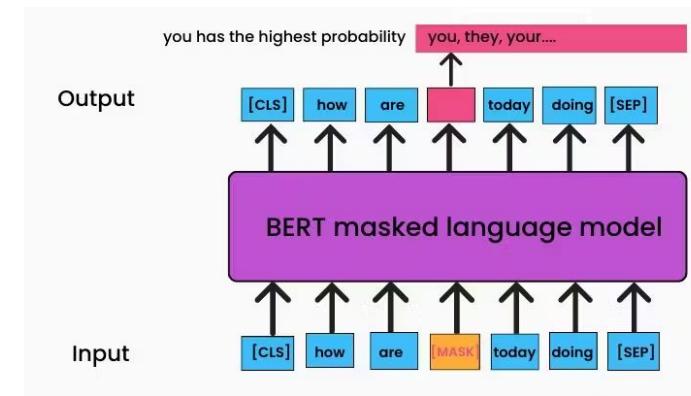
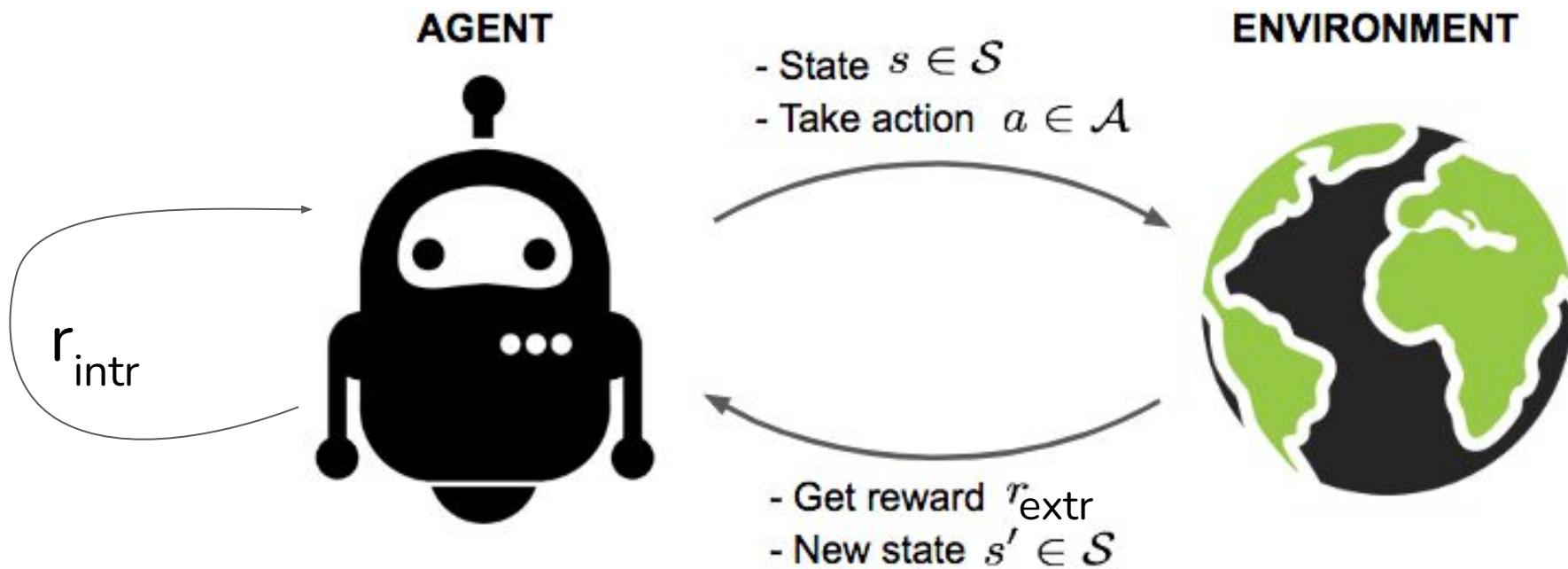


Fig 5. SimCLR's Contrastive Learning Process



Self-supervision в RL?



MOTIVATION

Intrinsic motivation



"I love this so much,
I can't think of anything
else while doing this!"

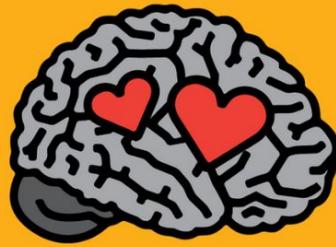
Extrinsic motivation



"Never mind, I have
to complete this before
next week to get a reward."

imgflip.com

INTRINSIC



Behavior that is triggered
from within a person. In
other words, the person
is **rewarding herself**.

vs.

EXTRINSIC



Behavior that is driven by
external rewards (given
by others), such as money,
grades, and praise.



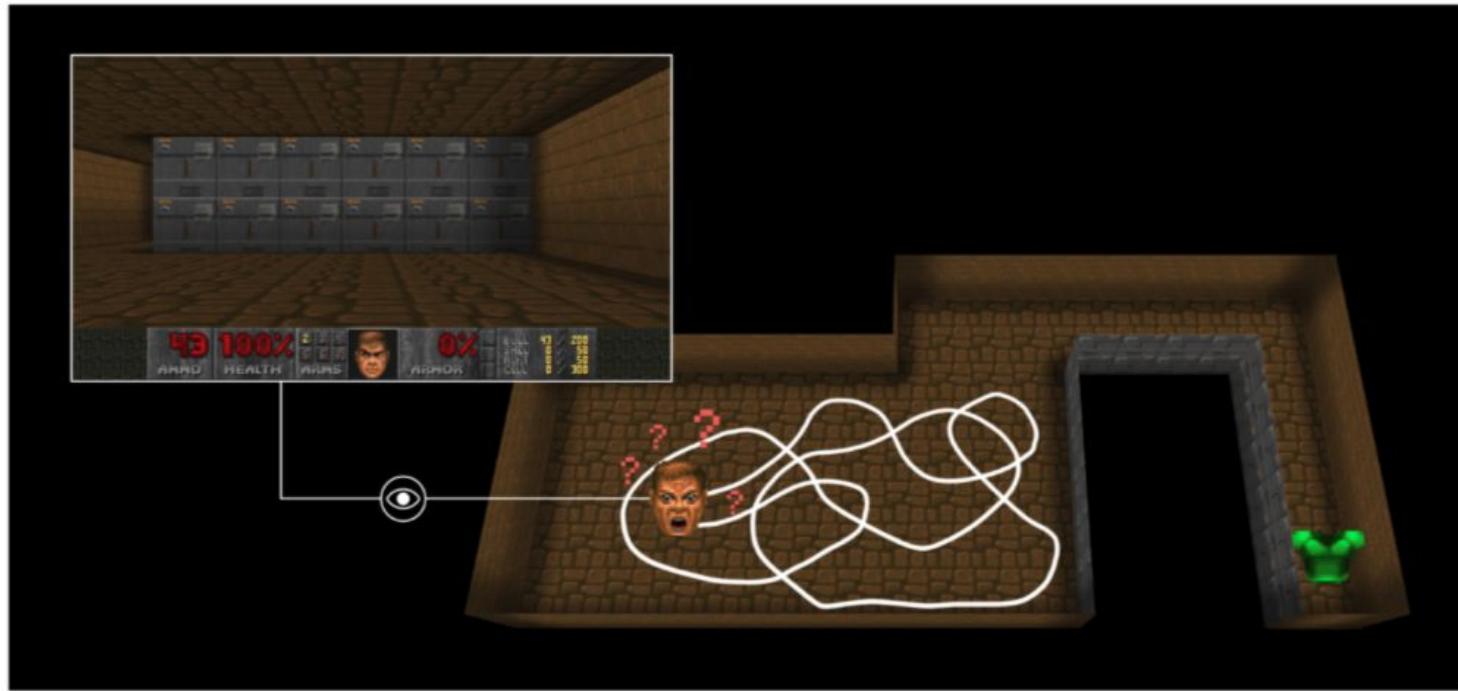
Intrinsic VS Extrinsic Motivation

Extrinsic motivation — provided by environment, rare

Intrinsic motivation — helps to develop broad set of skills

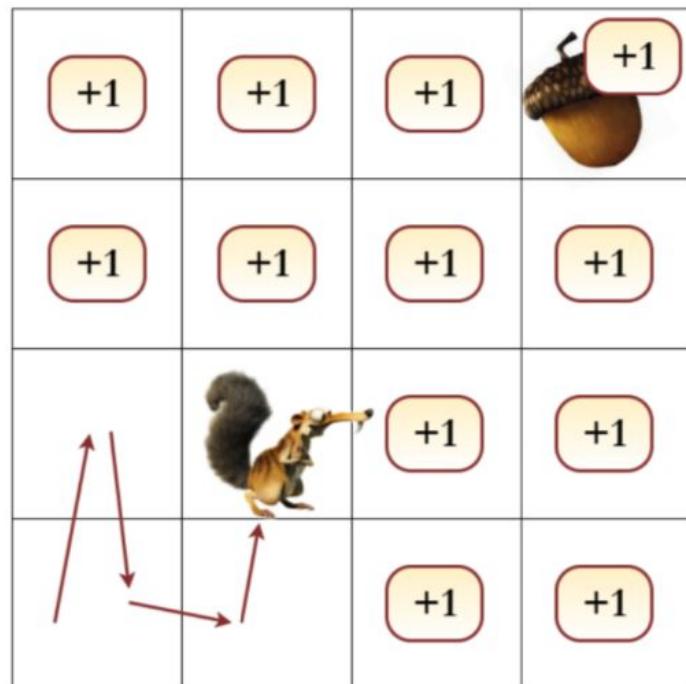
- Hindsight Experience Replay:
агент получает на вход не только исходный стейт, но и целевой. Если агент сможет достичь целевое состояние, то он получает positive intrinsic motivation. Смысл: набрать положительных примеров
- Empowerment:
функционал, который помогает агенту попасть в состояния, из которых, он лучше всего контролирует, что будет дальше (какие состояния придут впоследствии)
- Chaos minimization:
avoiding surprises, состояния, из которых легче всего предсказывать (тетрис: состояние в котором легче всего “минимизировать хаос”, т.е. предсказывать будущее, — это пустой экран. Так что в этой игре при такой intrinsic motivation агент обучится решать среду даже без extinsic reward)
- Exploration
Исследование/любопытство как внутренняя мотивация

Exploration is hard



Добавим +1 за исследование

а) на каждом эпизоде обновлять бонусы

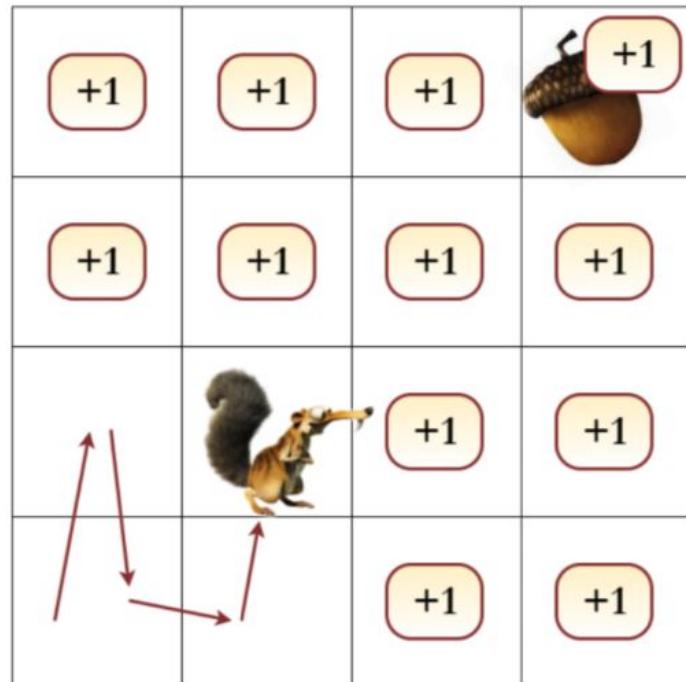


Добавим +1 за исследование

а) ~~на каждом эпизоде обновлять бонусы~~

б) $+1/n(s)$, n – кол-во посещений

- затухнет потихоньку



Добавим +1 за исследование

а) ~~на каждом эпизоде обновлять бонусы~~

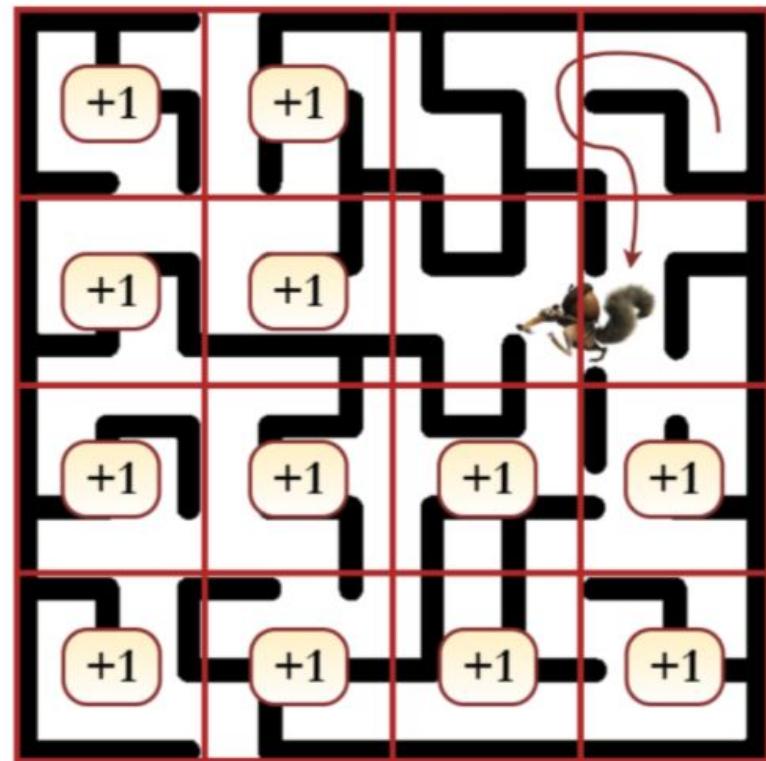
б) $+1/n(s)$, n – кол-во посещений

- затухнет потихоньку



Добавим +1 за исследование

- а) ~~на каждом эпизоде обновлять бонусы~~
- б) $+1/n(s)$, n – кол-во посещений
 - затухнет потихоньку



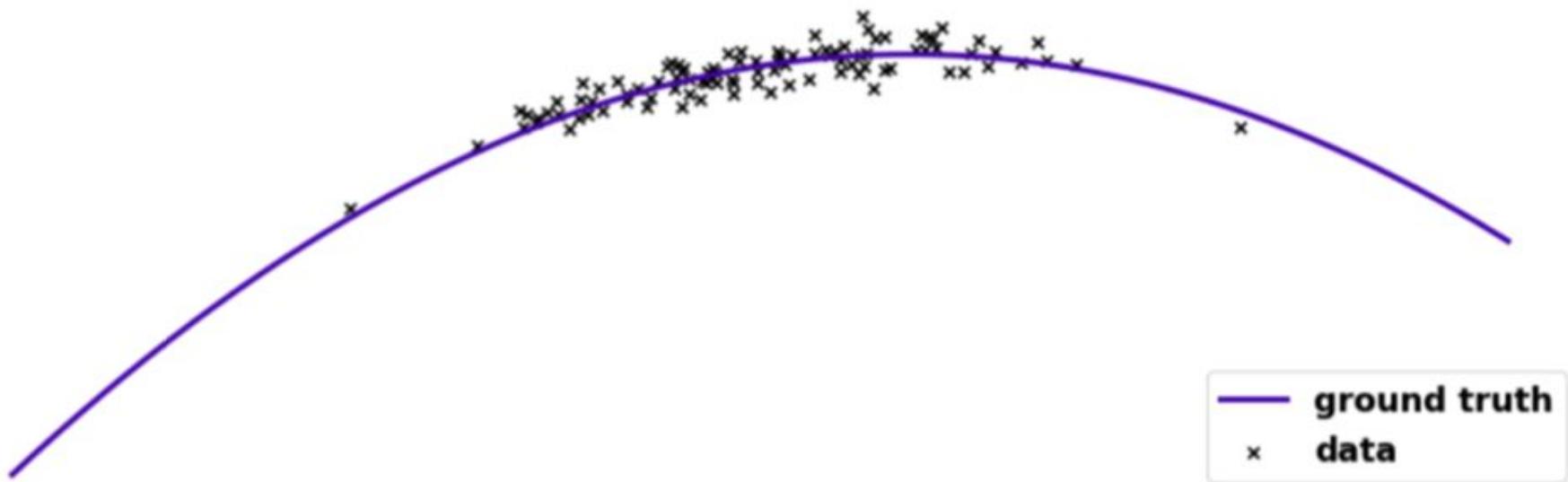
В Марио исследование = продвижение вправо



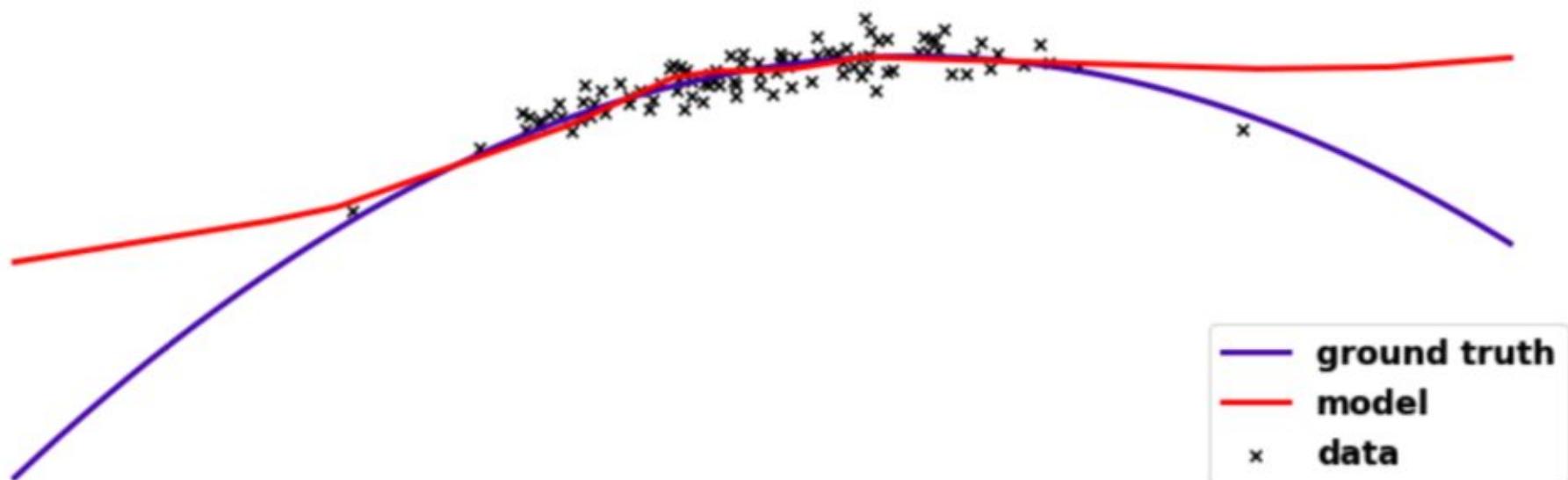
В Марио исследование != продвижение вправо



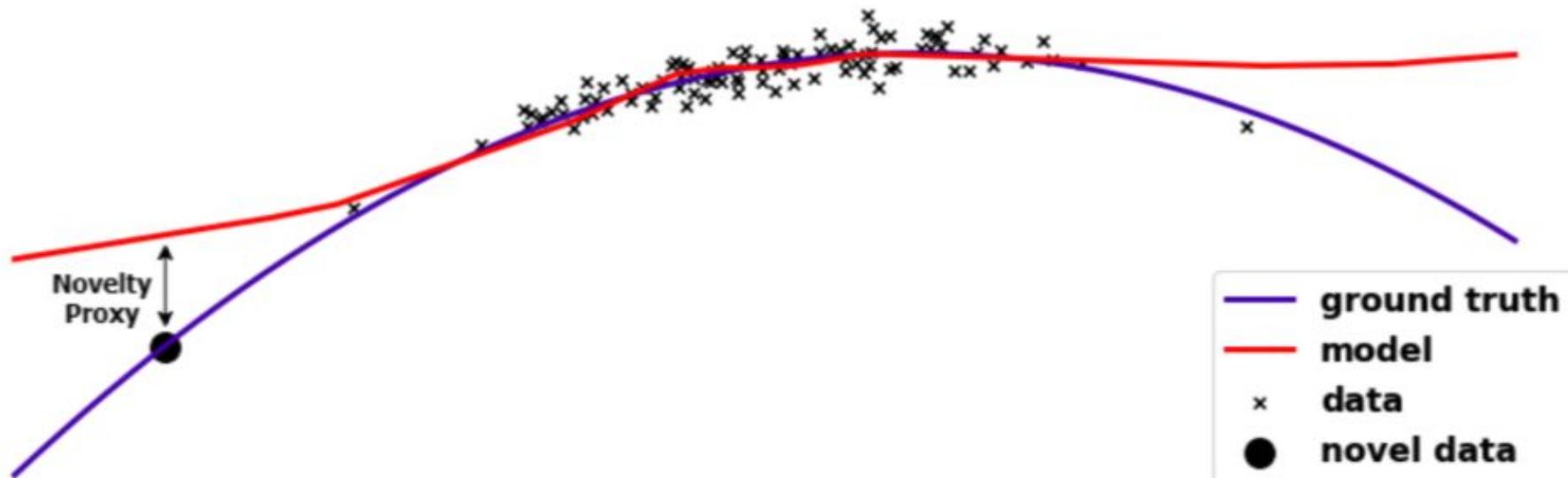
Попробуем формализовать Novelty detection



Попробуем формализовать Novelty detection



Попробуем формализовать Novelty detection



Новизна в RL

Взяли какую-то $y(s)$,

учить будем $f(s)$: $f(s) \approx y(s)$

тогда $r_{intr} = error(s) = ||f(s) - y(s)||^2$

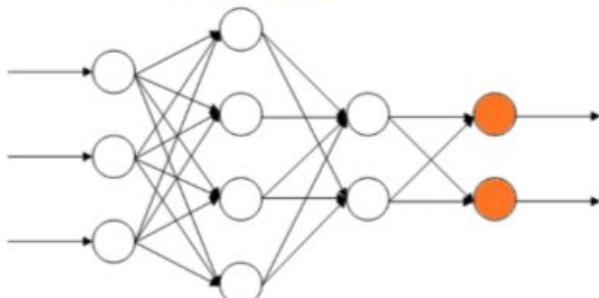
Нужна функция $y(s)$, которая:

- зависит от состояний
- детерминирована
- не меняется со временем

RND - Random Network Distillation



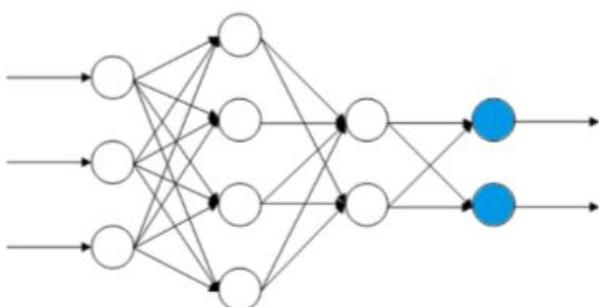
Target Network $y(s)$



$$r^{\text{intr}}(s) := \frac{1}{2} \|\phi^{\text{predictor}}(s) - \phi^{\text{target}}(s)\|_2^2$$



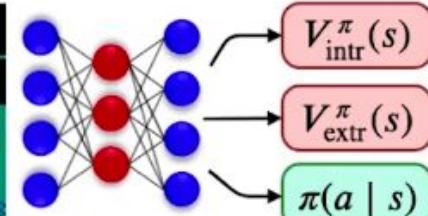
→ MSE



$$r = r_{\text{intr}} + r_{\text{extr}}$$

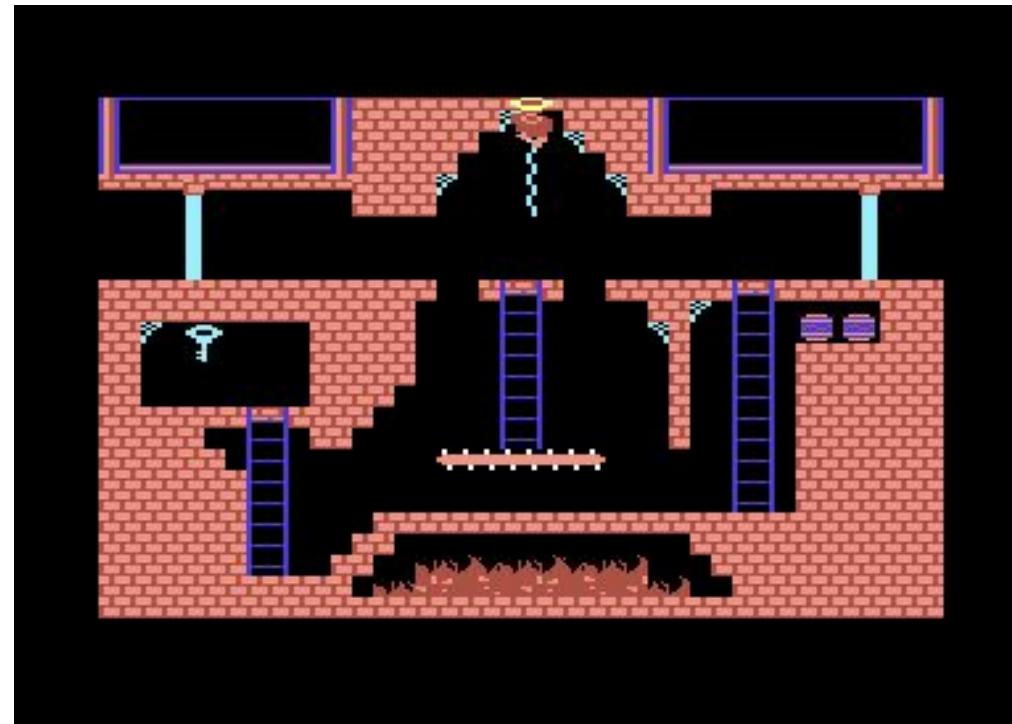
Но важно:

- масштабировать компоненты
- нормализовать внутренний ревард
- сначала немного обучаем предиктор без РЛ, только потом РЛ



https://www.c64-wiki.com/wiki/Montezuma%27s_Revenge

Montezuma's Revenge



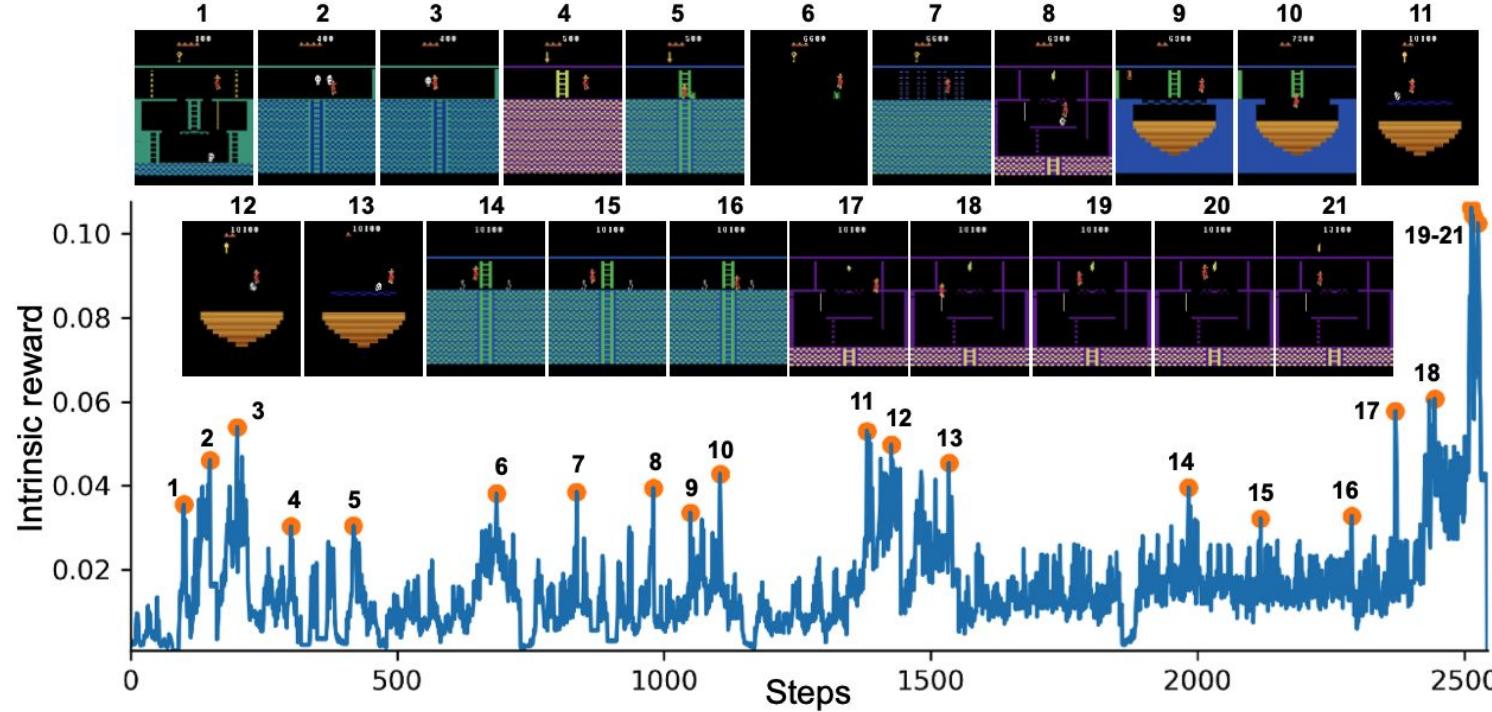


Рисунок 1. Бонус за исследование в методе RND в течение первого эпизода, в котором агент подбирает факел (шаги 19–21). Чтобы добраться до факела, агент проходит через 17 комнат и собирает драгоценные камни, ключи, меч, амулет, а также открывает две двери. Многие всплески бонуса за исследование соответствуют значимым событиям: потеря жизни (шаги 2, 8, 10, 21), узкое избежание врага (3, 5, 6, 11, 12, 13, 14, 15), преодоление сложного препятствия (7, 9, 18) или подбор предмета (20, 21). Крупный всплеск в конце связан с новым опытом взаимодействия с факелом, тогда как более мелкие всплески отражают относительно редкие, но уже встречавшиеся ранее события.

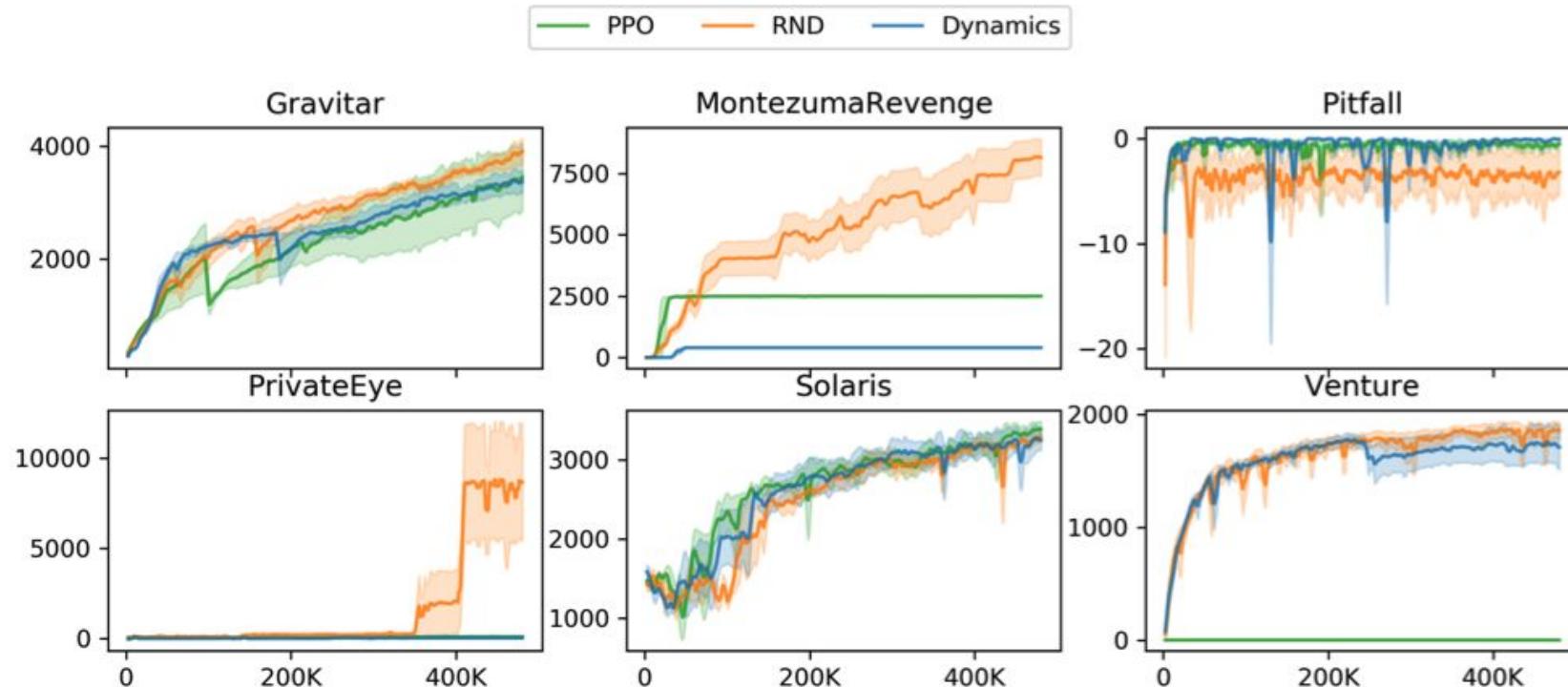


Figure 7: Mean episodic return of RNN-based policies: RND, dynamics-based exploration method, and PPO with extrinsic reward only on 6 hard exploration Atari games. RND achieves state of the art performance on Gravitar, Montezuma's Revenge, and Venture, significantly outperforming PPO on the latter two.

Можно использовать наоборот!

RND в офлайн-рл

Anti-Exploration by Random Network Distillation

Alexander Nikulin¹ Vladislav Kurenkov¹ Denis Tarasov¹ Sergey Kolesnikov¹

Abstract

Despite the success of Random Network Distillation (RND) in various domains, it was shown as not discriminative enough to be used as an uncertainty estimator for penalizing out-of-distribution actions in offline reinforcement learning. In this paper, we revisit these results and show that, with a naive choice of conditioning for the RND prior, it becomes infeasible for the actor to effectively minimize the anti-exploration bonus and discriminativity is not an issue. We show that this limitation can be avoided with conditioning based on Feature-wise Linear Modulation (FiLM), resulting in a simple and efficient ensemble-free algorithm based on Soft Actor-Critic. We evaluate it on the D4RL benchmark, showing that it is capable of achieving performance comparable to ensemble-based methods and outperforming ensemble-free approaches by a wide margin.

1 Introduction

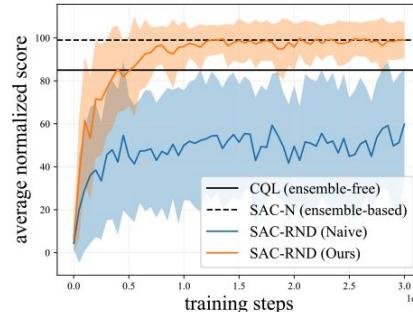


Figure 1. Mean performance of SAC-RND variants on walker and hopper medium-* datasets, each averaged over 3 seeds. We plot performance for the naive version, which uses concatenation conditioning, and our final version, which is described in Section 5. We also plot the final scores for the ensemble-free CQL (Kumar et al., 2020) and the ensemble-based SAC-N (An et al., 2021). It can be seen that our version is a significant improvement over the naive version, achieving performance comparable to ensembles.

Curiosity (любопытство)

У нас есть агент

Он строит свою модель мира $f(s,a)$

$$\text{Error} = \|f(s,a) - s'\|^2$$

Там где модель ошибается, у нас
плохие знания о мире: там надо
данных собрать

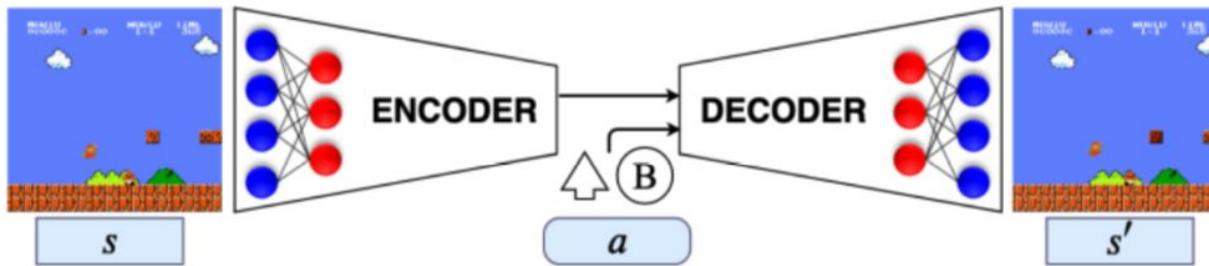
Агент мотивирован исследовать
те части среды, где его текущая
модель мира даёт наибольшую
ошибку предсказания.*



*Juergen Schmidhuber (1991)



Curiosity: forward dynamics model



$$\hat{s}_{t+1} = f(s_t, a_t)$$

$$r_t^{\text{intr}} = \|s_{t+1} - \hat{s}_{t+1}\|_2^2$$

Проблема:

- Прямое предсказание пикселей вычислительно дорого (можно обучить VAE, но:)
- Модель вынуждена учить всё: фон, мигающие огни, частицы — даже если они не связаны с задачей.
- Это приводит к переобучению на нерелевантных деталях.
- Модель у нас детерминированная, а функция перехода — нет

Noisy TV problem

Noisy TV = что-то что модель прямой динамики не сможет выучить никогда



Агент мотивирован исследовать те
части среды, где **его текущая**
модель мира даёт наибольшую
ошибку предсказания.*

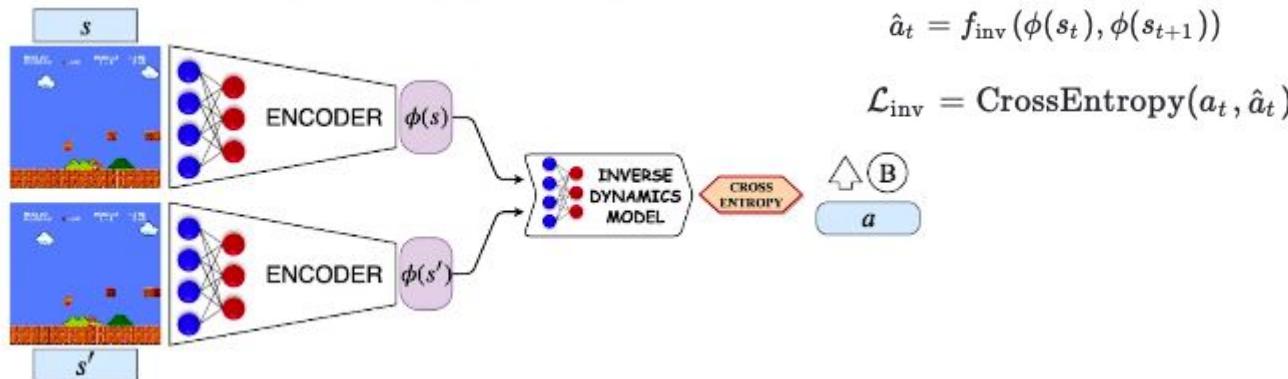


*Juergen Schmidhuber (1991)

Идея: отфильтровать noisy TV энкодером

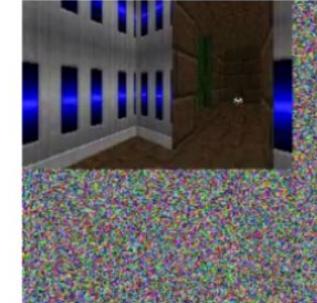
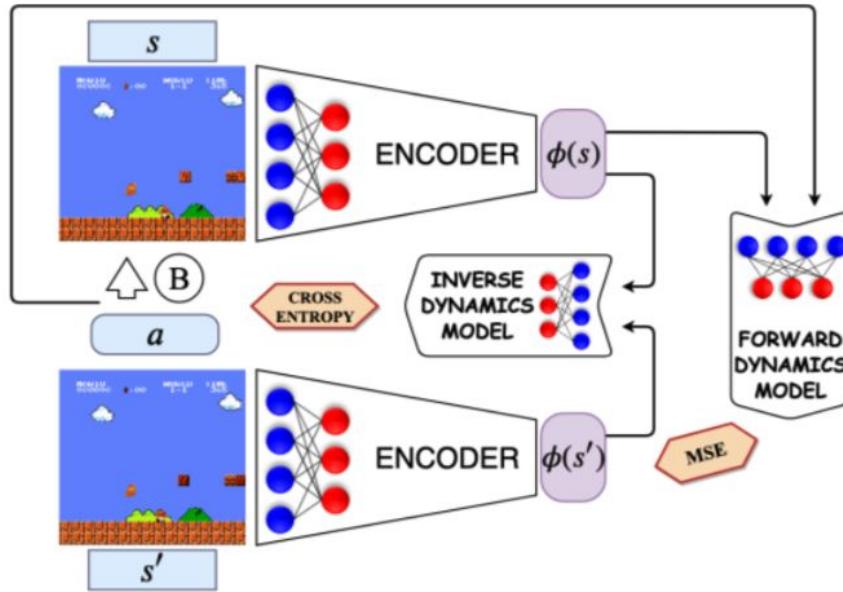
Идея: обучить энкодер ϕ так, чтобы его выходы содержали только ту информацию, которая необходима для предсказания действия, приведшего к переходу между состояниями.

Curiosity: inverse dynamics model

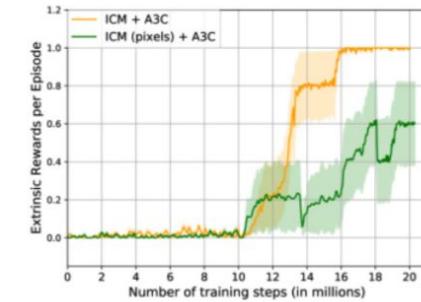


Теперь мы учим только контролируемую часть среды!

Intrinsic Curiosity Module (ICM)



40% of observation image is augmented with noise (VizDoom environment).



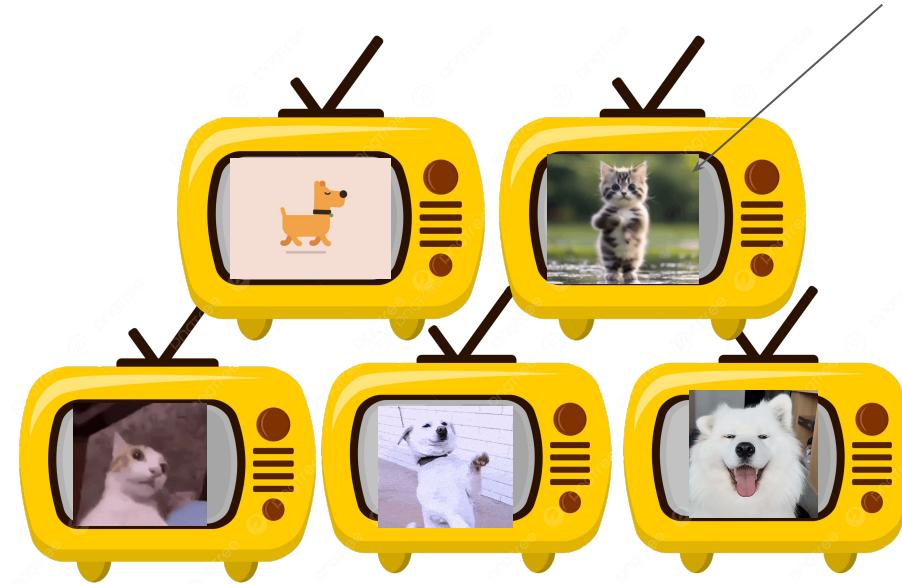
ICM still performs well!

<https://www.youtube.com/watch?v=l1FqtAHfJLI>

непредсказуемый шум

Procrastination

Noisy TV к которому у вас есть пульт



ICM не поможет



Часть 2. Обучение без действий

Recap. Online RL –

- Долго
- Трудно
- Дорого

Так что в основном на практике используется Offline RL...

Данные для RL

Хотим обучить foundation VLA на большом и разнообразном датасете, чтобы у нас наконец появились робо-дворецкие.

Откуда взять эти разнообразные и большие данные?
Какие есть варианты?

- Телеоперация
- Симуляция

Телеоперация

Основной способ получения данных сейчас. Требует дополнительного оборудования и специфических навыков. Масштабируется линейно, то есть плохо.



Симуляция

Может быть быстрой и разнообразной, но подвержена плохой sim2real генерализации. Дорого и сложно моделировать многие важные физические процессы.



А что в других областях?

"Our pre-training data is the equivalent of fossil fuel, and that data is running out.."

- LLM и языковым моделям достался весь интернет, триллионы токенов
- VLM и генеративные моделям достался весь визуальный контент в интернете
- Голосовым моделям достался как минимум звук со всех когда-либо записанных видео + Музыка



Pre-training as we know it will end

Compute is growing:

- Better hardware
- Better algorithms
- Larger clusters

Data is not growing:

- We have but one internet
- The fossil fuel of AI

Internet. We have, but one Internet. You could even say you can even go as far as to say. That data is the fossil fuel of AI. It was like, created somehow. And now we use it.



А что в других областях?



wait, you guys had
the data all this time?

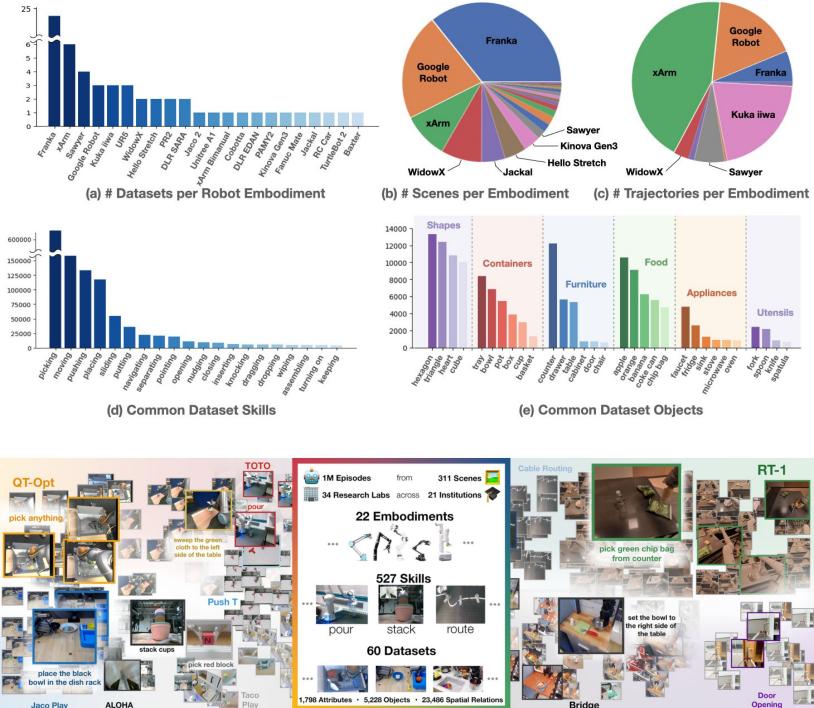
Реакция человека
из rl/imitation/robotics

Потом и кровью

Самый популярный open-source датасет для роботики - Open X-Embodiment. Большая коллаборация ученых со всего мира.

- Всем миром собирали несколько лет, участвовало 21 институт
- 22 различных робота
- Всего ~512 различных задач, и всего ~2M эпизодов

Покрывает ли этот датасет все разнообразие реального мира? Вопрос риторический.



Нас спасут данные без действий?

Все существующие видео в интернете + содержащие человека. Много ли это?

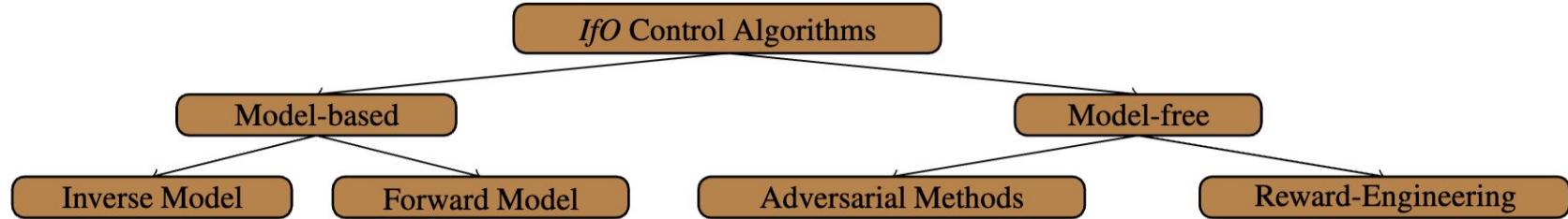
- Крайне разнообразные данные, от бытовых вещей, до крайне специфических
 - Появляется все больше академических датасетов с эгоцентричными видео
 - Очень легко и дешево масштабировать. Достаточно одной нагрудной камеры.

Но не содержат действий!!!!
Как их использовать?



Это большая область сама по себе!

Область существует уже давно и имеет множество названий: reinforcement learning from observations, imitation learning from observations, learning from videos, learning from passive data, etc..



Начнем с простого

Хотим решить Minecraft. Обучить VLA, которая смогла бы добыть алмаз.

- Симулятор из MineRL крайне медленный.
За 4 года соревнований на NeurIPS нет успеха.
- Есть маленький датасет где действия есть.
- Есть бесконечные запасы видео из YouTube и Twitch. Без действий.

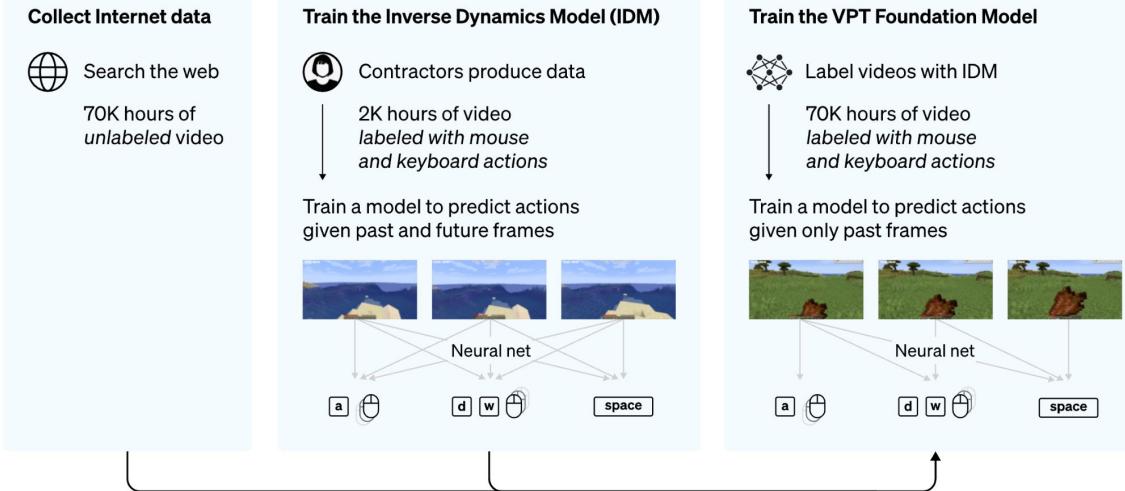
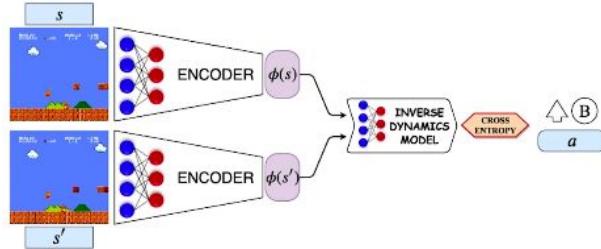
Как быть? Ваши предложения?



Inverse dynamics model

Идея!

На данных с действиями обучим модель предсказывать настоящие действия и разметим ей весь YouTube.
Да, это будут шумные данные, но мы умеем с таким работать (см. noisy labels)!



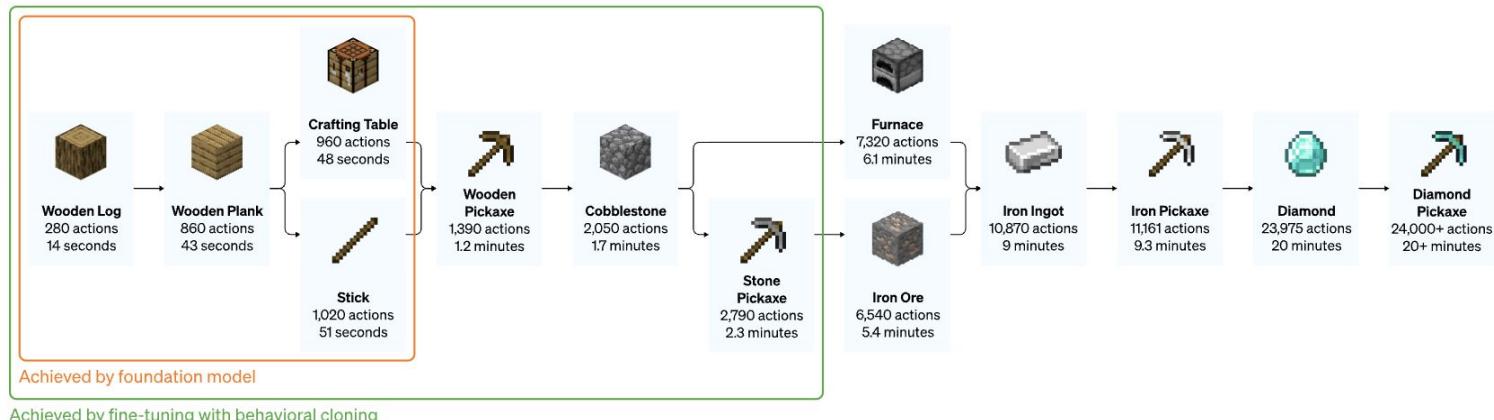
Open AI
так и сделали!

Обучаем

Сначала: пред-обучение на большом количестве шумных, но разнообразных данных

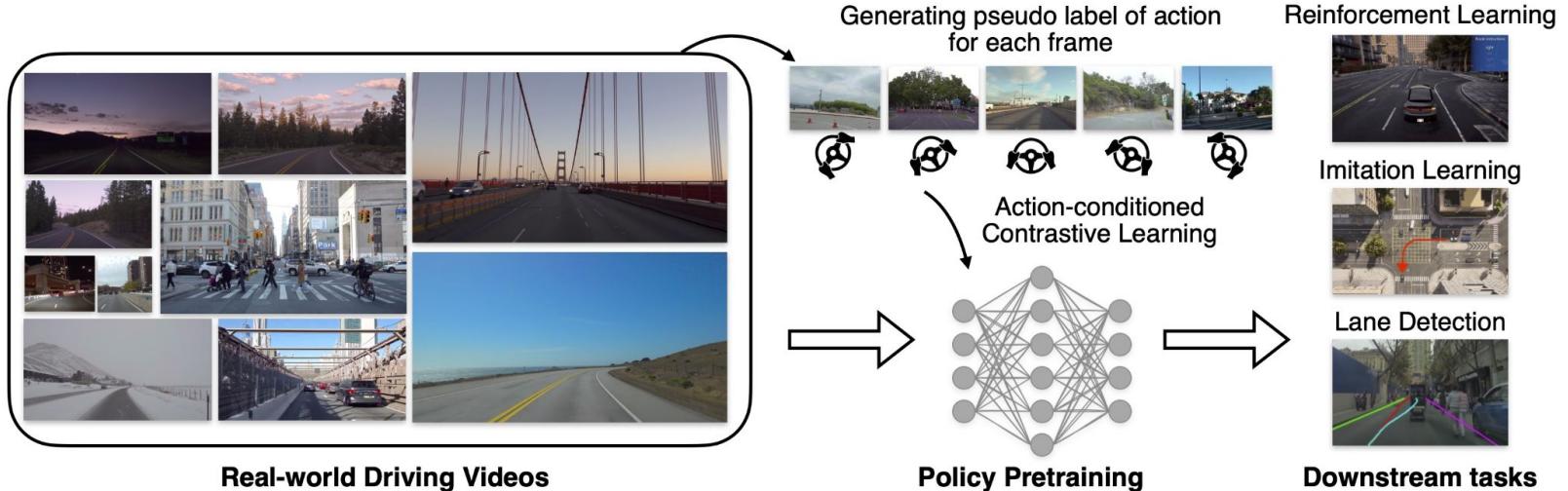
Далее: можно дообучить на маленьком количестве крайне хороших данных.

Далее: RL



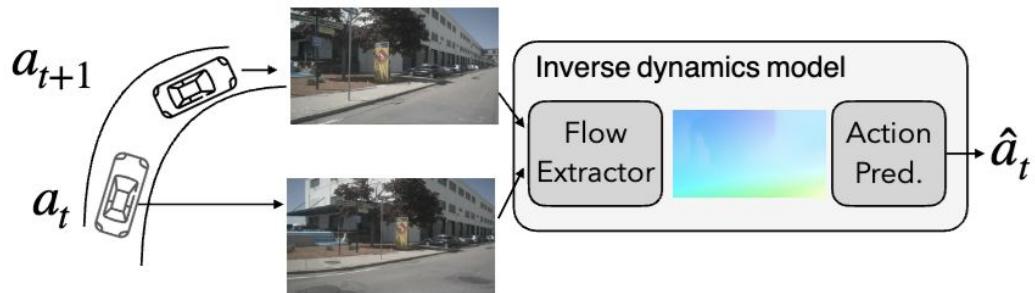
Работает не только на Minecraft

Можно использовать и для self-driving, точно так же обучаясь по видео с YouTube. В разы увеличивает доступные данные для пред-обучения.

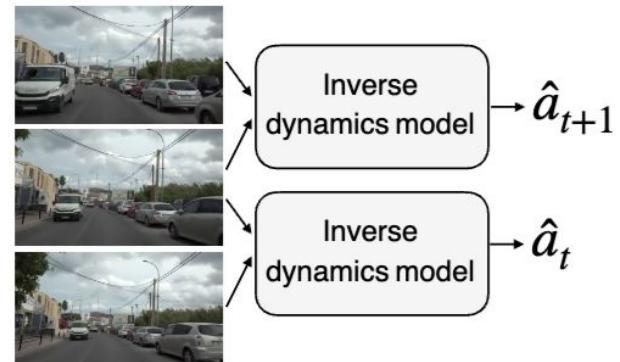


Хорошо работает не только на Minecraft

Можно использовать и для self-driving, точно так же обучаясь по видео с YouTube. В разы увеличивает доступные данные для пред-обучения.



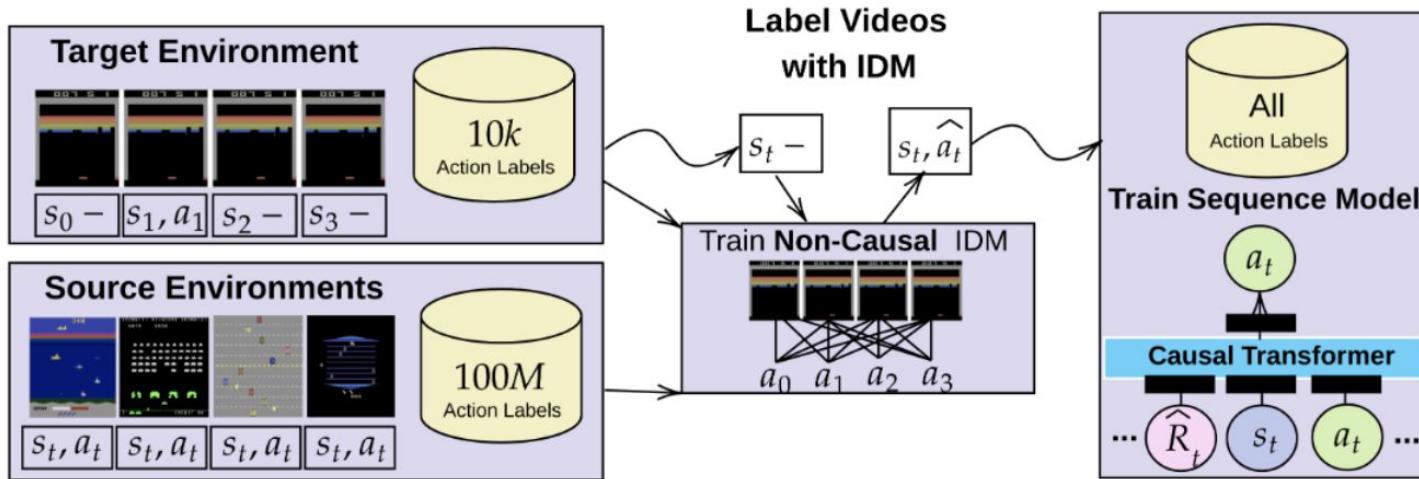
A. Inverse dynamics model training



B. Labeling in-the-wild videos

Чем больше данных, тем лучше

Что если данных с действиями в нужном домене
ну совсем мало? Оказывается, можно докинуть
из других и станет лучше. Тем не менее, домены
должны быть концептуально схожи, даже если
имеют разный набор действий.

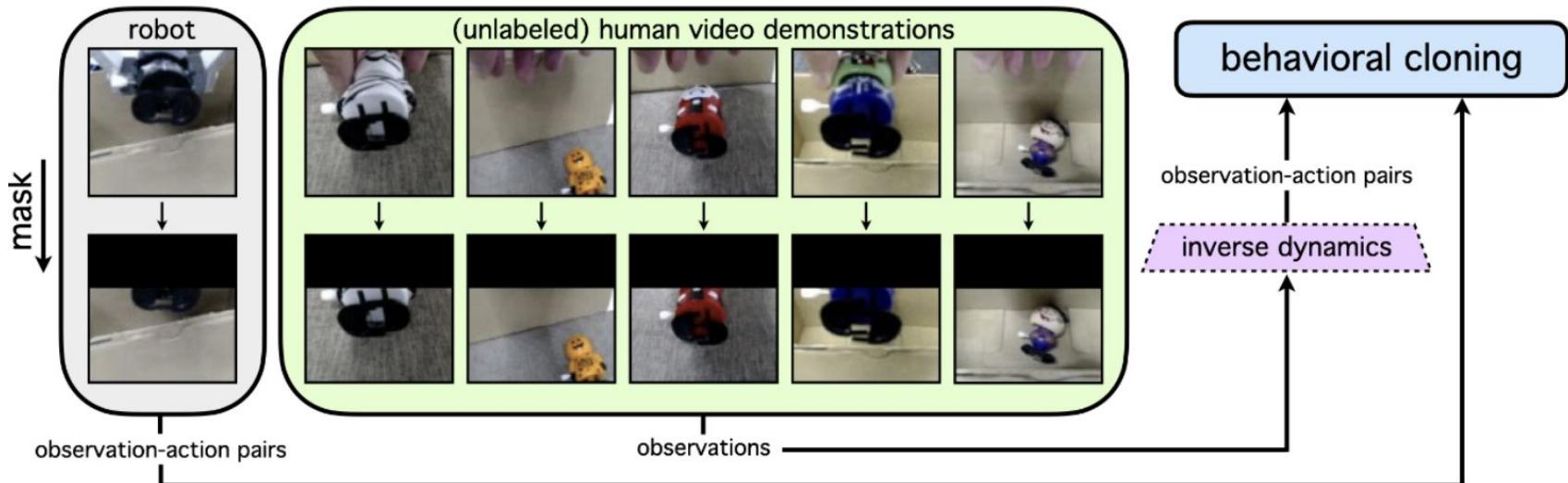


Что делать с эгоцентричными данными?



Можно попробовать применить...

С некоторой долей успеха можно разметить с помощью IDM обученной на данных с робота. Масштабируется ли это?



Ограничения простого подхода

Несмотря на то, что разметка с помощью IDM крайне простой и эффективный метод, у него есть ряд ограничений.

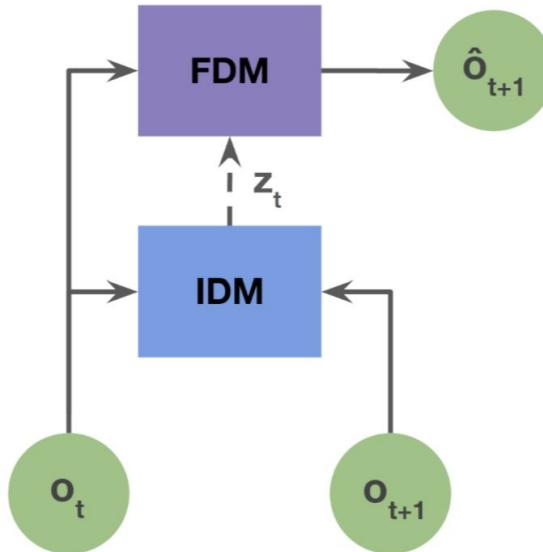
- Подходит только для доменов где уже есть достаточно количество данных с действиями
- Плохо генерализуется и обобщается на несколько доменов с разными пространствами действий
- Генерализация и точность ограничены разнообразием обучающего датасета с действиями
- Что, если в нашем домене в принципе нельзя достать настоящие действия?

Вот бы существовал метод, обобщающий IDM, так чтобы обучать можно было в unsupervised режиме.....

Latent Action Policy Optimization (LAPO)

Learning to act without actions — статья с которой все началось.

- Предложили способ обучать латентные действия без разметки (почти)
- Позволяет сразу пред-обучать агентов (а не только энкодеры) на большом количестве данных
- Требует совсем чуть чуть разметки для адаптации под настоящие действия



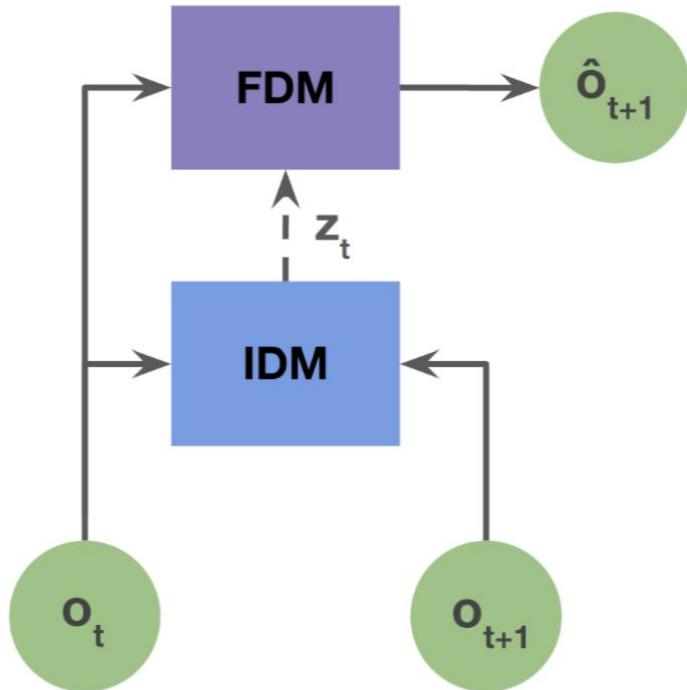
Почему это работает?

IDM выдает нам латентное действие по двум соседним состояниям.

Как приблизить его к настоящим действиям?
Вспомним динамику среды:

$$p(s' | s, a)$$

Если мы зафиксируем состояния, то каким свойством обладают настоящие действия по определению?



Почему это работает?

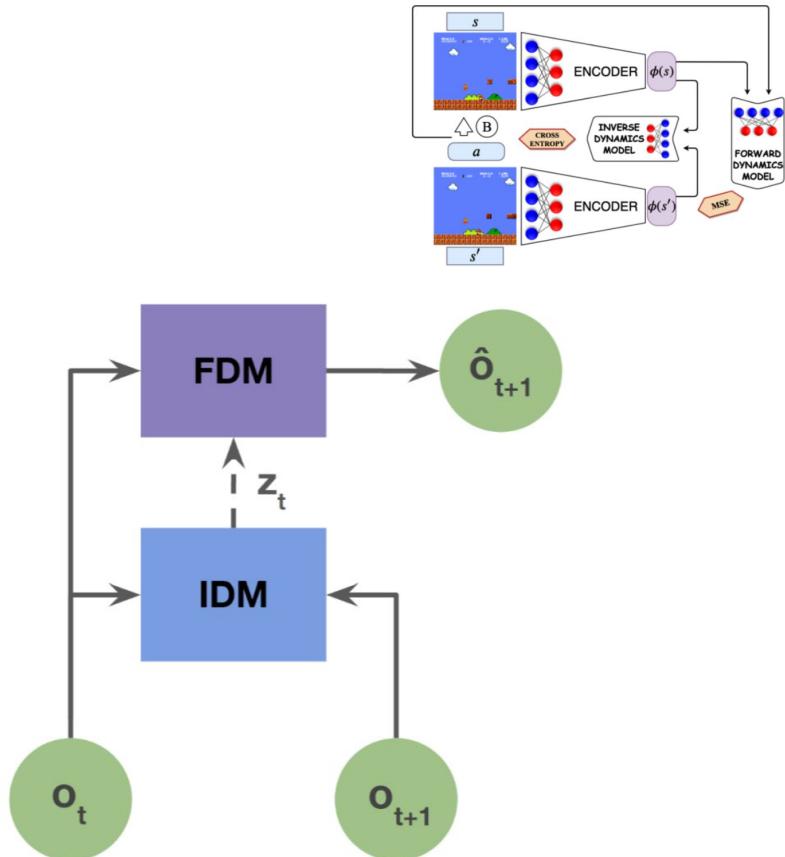
IDM выдает нам латентное действие по двум соседним состояниям.

Как приблизить его к настоящим действиям?
Вспомним динамику среды:

$$p(s'|s, a)$$

Если мы зафиксируем состояния, то каким свойством обладают настоящие действия по определению?

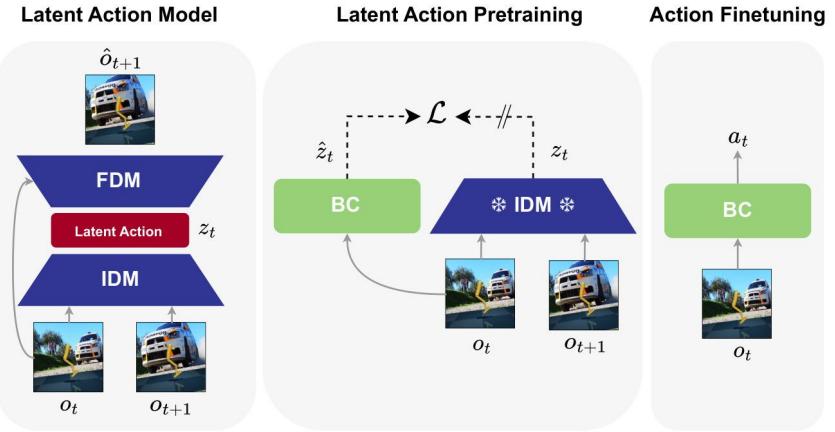
Правильно, они максимально информативны относительно следующего состояния — то есть **максимизируют правдоподобие!**



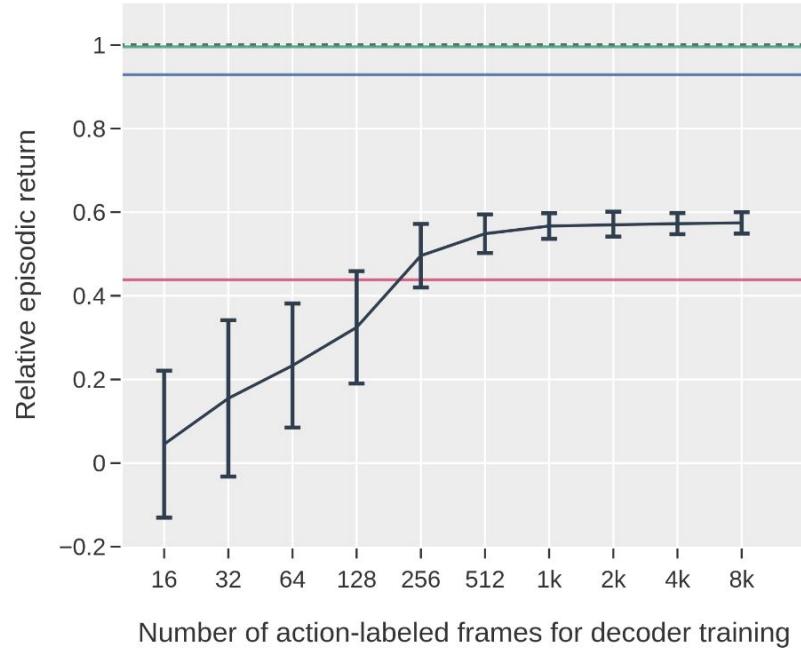
Что с этим делать?

Этапы почти идентичны тому, что мы разбирали с IDM:

- Предобучаем IDM с латентными действиями
- Размечаем ими весь датасет
- Обучаем агента предсказывать латентные действия на всем датасете
- Под конкретный домен до-обучаем предсказывать реальные действия на маленьком количестве реальных действий

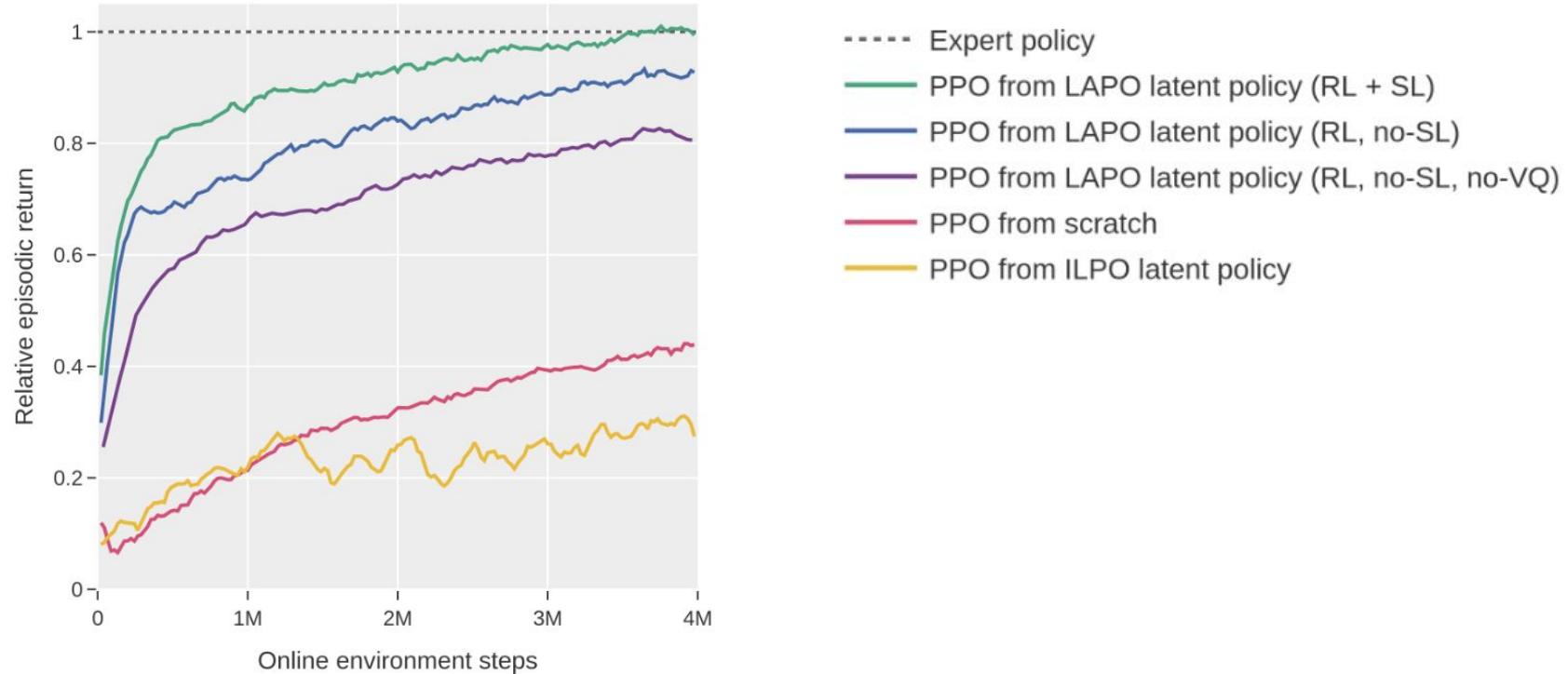


Работает



- Expert policy
- PPO from LAPO latent policy (RL + SL)
- PPO from LAPO latent policy (RL, no-SL)
- PPO from scratch
- Latent policy + decoder head (offline)

Можно до-обучать с помощью RL



Игрушки это круто, как на счет реальных данных?

LAPA повторяет успех LAPO но уже на более приближенных к реальным задачам данных. Берет датасет с эгоцентричными видео людей, использует трансформеры и выучивает латентные действия.

Large-Scale Robot Datasets



- Expensive to collect ✗
- Requires robot hardware ✗
- Contains robot actions ✓

Internet-scale Video Data

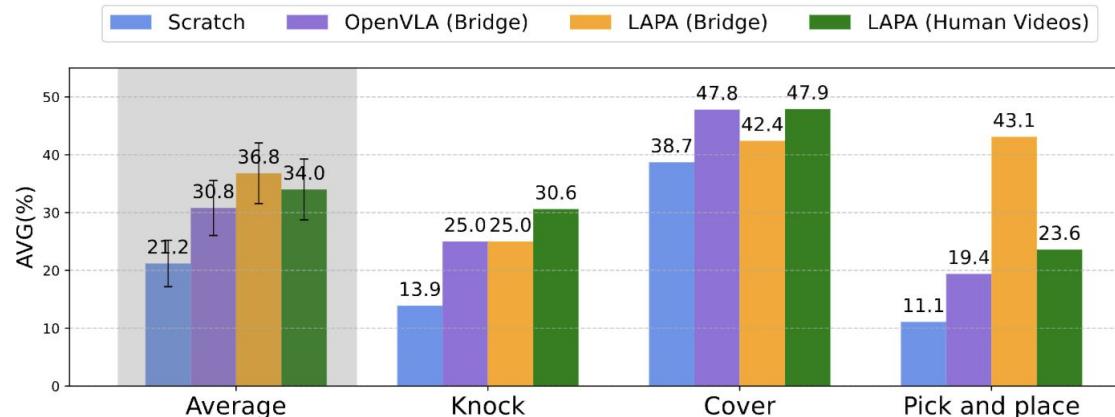


- Widely available ✓
- Human ↔ robot gap ✗
- No robot actions ✗

 Robotic Foundation Model

Игрушки это круто, как на счет реальных данных?

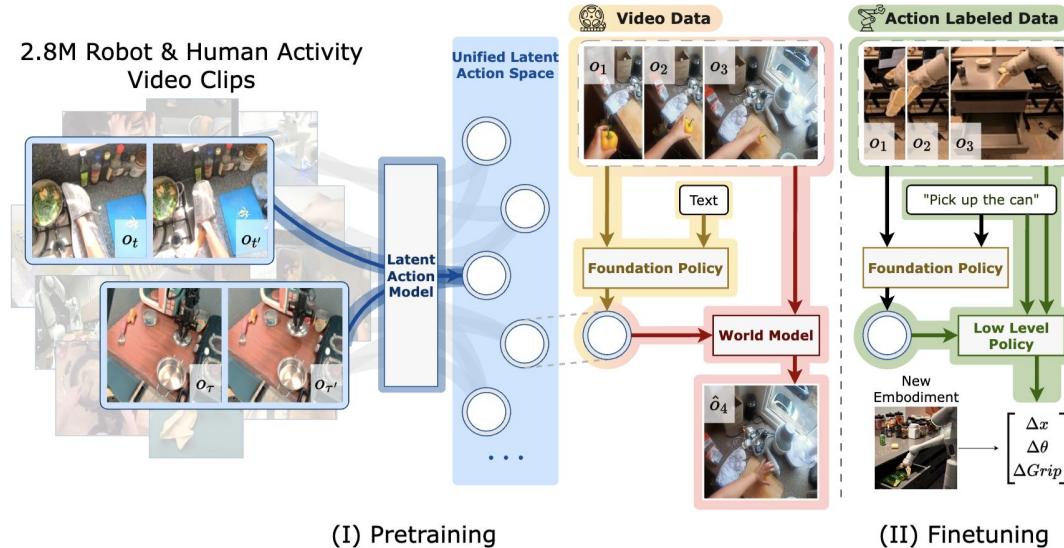
LAPA повторяет успех LAPO но уже на более приближенных к реальным задачам данных. Берет датасет с эгоцентричными видео людей, использует трансформеры и выучивает латентные действия.



(b) Real-world Tabletop Manipulation Robot Results

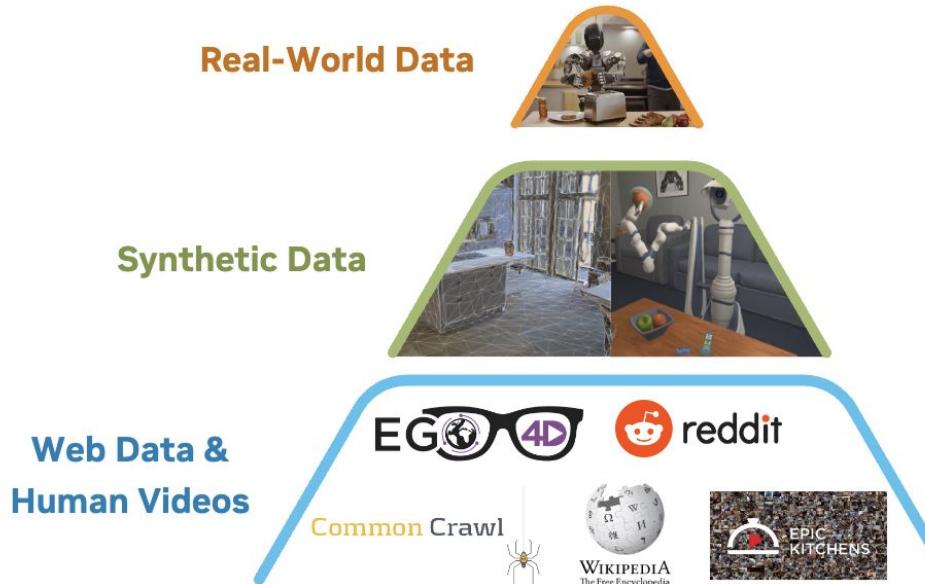
Теперь это мейнстрим

Вышло и продолжает выходить огромное множество статей,
которые используют латентные действия для робототехники:
LAPA, MOPO, IGOR, GROOT N1, AgiBot, UniVLA, AdaWorld, CLAM, CoMo, etc



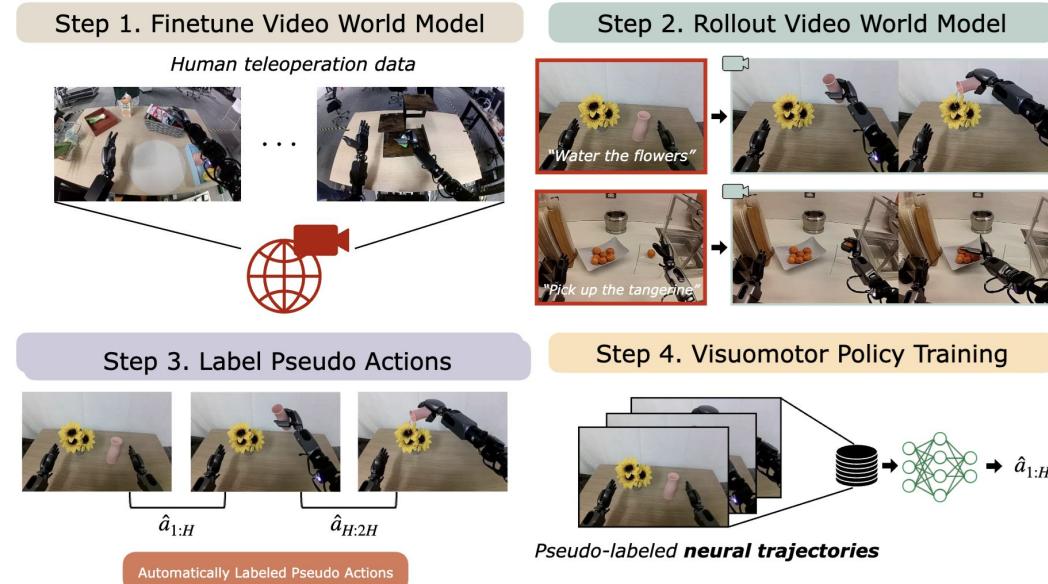
Теперь это мейнстрим

Например VLA модель GR00T N1 от Nvidia использует латентные действия как первый этап пре-трейна

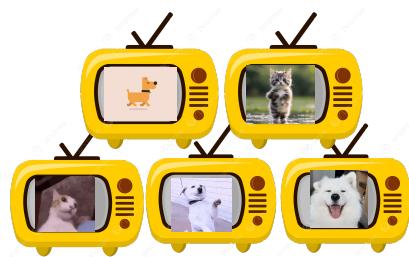


Универсальные модели мира

DreamGen использует латентные действия как универсальную репрезентацию для модели мира



Что если наши данные содержат шум?



Допустим шум скореллирован с настоящими действиями.

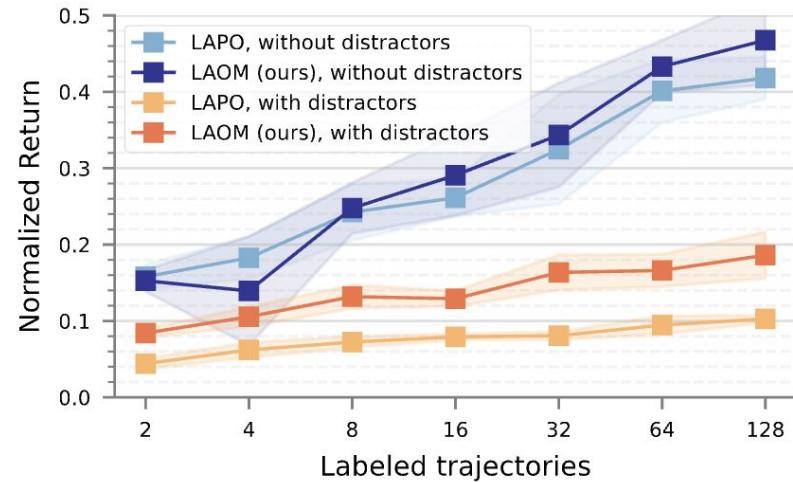
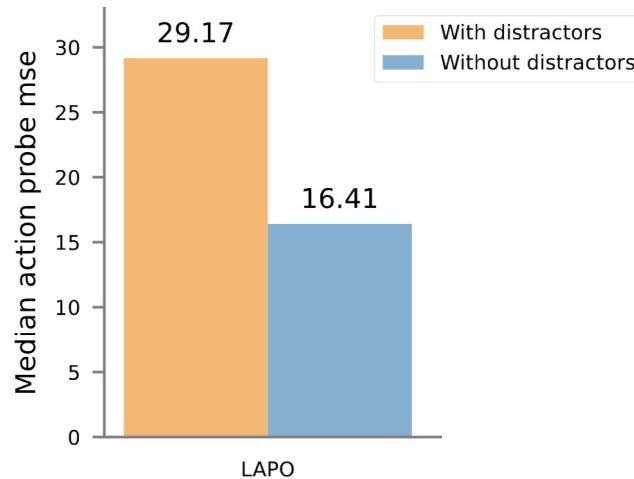
Будут ли реальные действия на самом деле максимизировать правдоподобие?

Distractor Control Suite



Что будет если обучить на таких данных?

LAPO полностью сломается!



Итоги лекции

- RL – тяжело
- Для упрощения online RL
 - улучшим exploration:
 - ◆ Random Network Distillation (RND)
 - ◆ Inverse Curiosity Model (ICM) = IDM + FDM
- Для упрощения offline RL
 - научимся использовать на action-less данные:
 - ◆ Inverse Dynamics Model (IDM)
 - ◆ Latent Action Policy Optimization (LAPO) = IDM + FDM
- ICM и LAPO страдают от дистракторов :(

Итоги курса

- Увидели как формулируются задачи на языке RL
- Model-based:
 - ◆ Имеем модель: Value/Policy Iteration
 - ◆ Нет модели: либо обучаем модель, либо используем model-free
- Model-free:
 - ◆ Monte-Carlo
 - ◆ Q-learning, Sarsa, Deep Q-learning, ...
 - ◆ Policy-based: PPO, SAC, DDPG, ...
- Not exactly RL:
 - ◆ Imitation learning: BC, DAGGER, action-less imitation
 - ◆ Offline RL
 - ◆ Inverse RL
 - ◆ Evolution (CE)
- Complicated RL: Multi-agent RL (MARL), Model-based RL (MCTS)
- RL applications: LLM, Robotics, Self-driving, ADDs, ...

Вопросы?



Credits. Lot's of slides taken from:

1. Exploration <https://www.youtube.com/watch?v=8zZrciFXJM8&list=PLt1IfGj6--eXjZDFBfnAhAJmCyX227ir>
2. Latent actions <https://www.youtube.com/watch?v=sAojfhu-XUs>