Towards A General Solution for Robotics

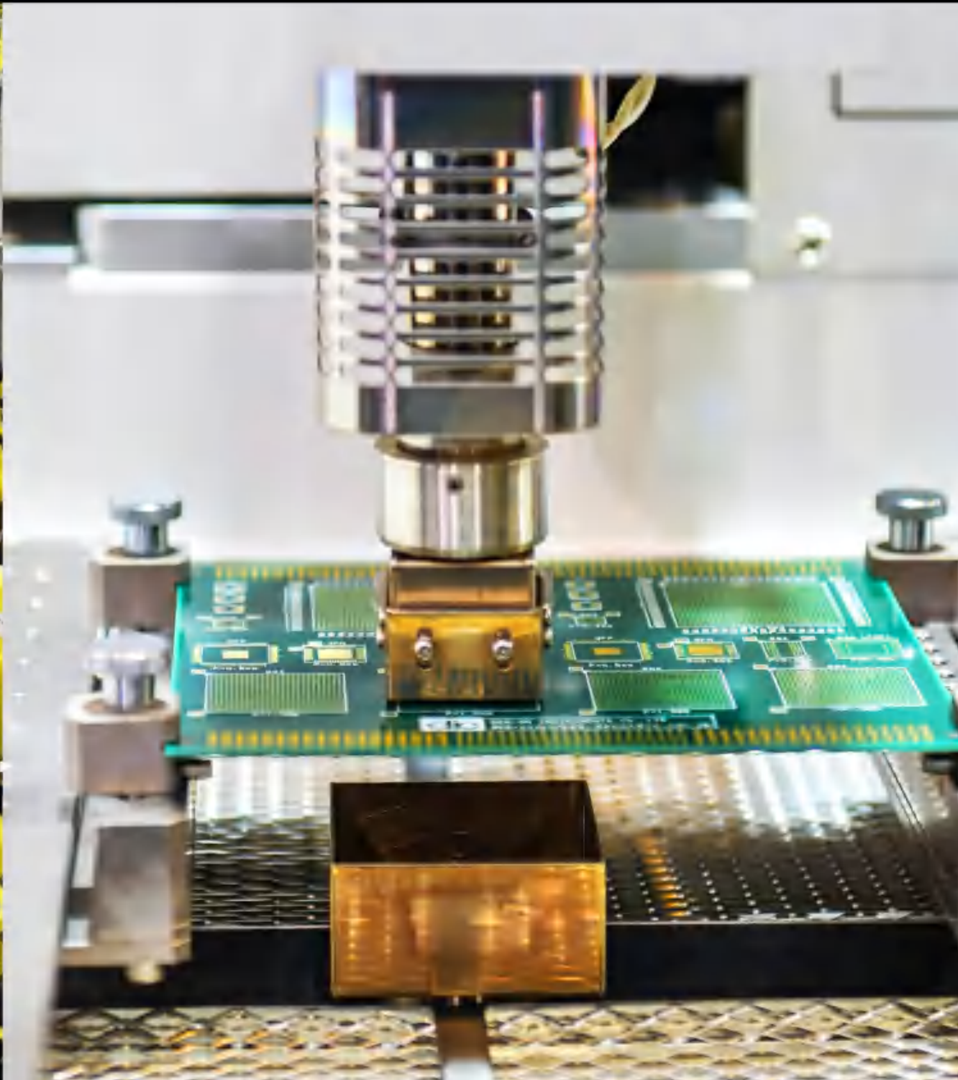
Pieter Abbeel

UC Berkeley & Covariant

PR-1

[Wyrobek, Berger, van der Loos, Salisbury, ICRA 2008]

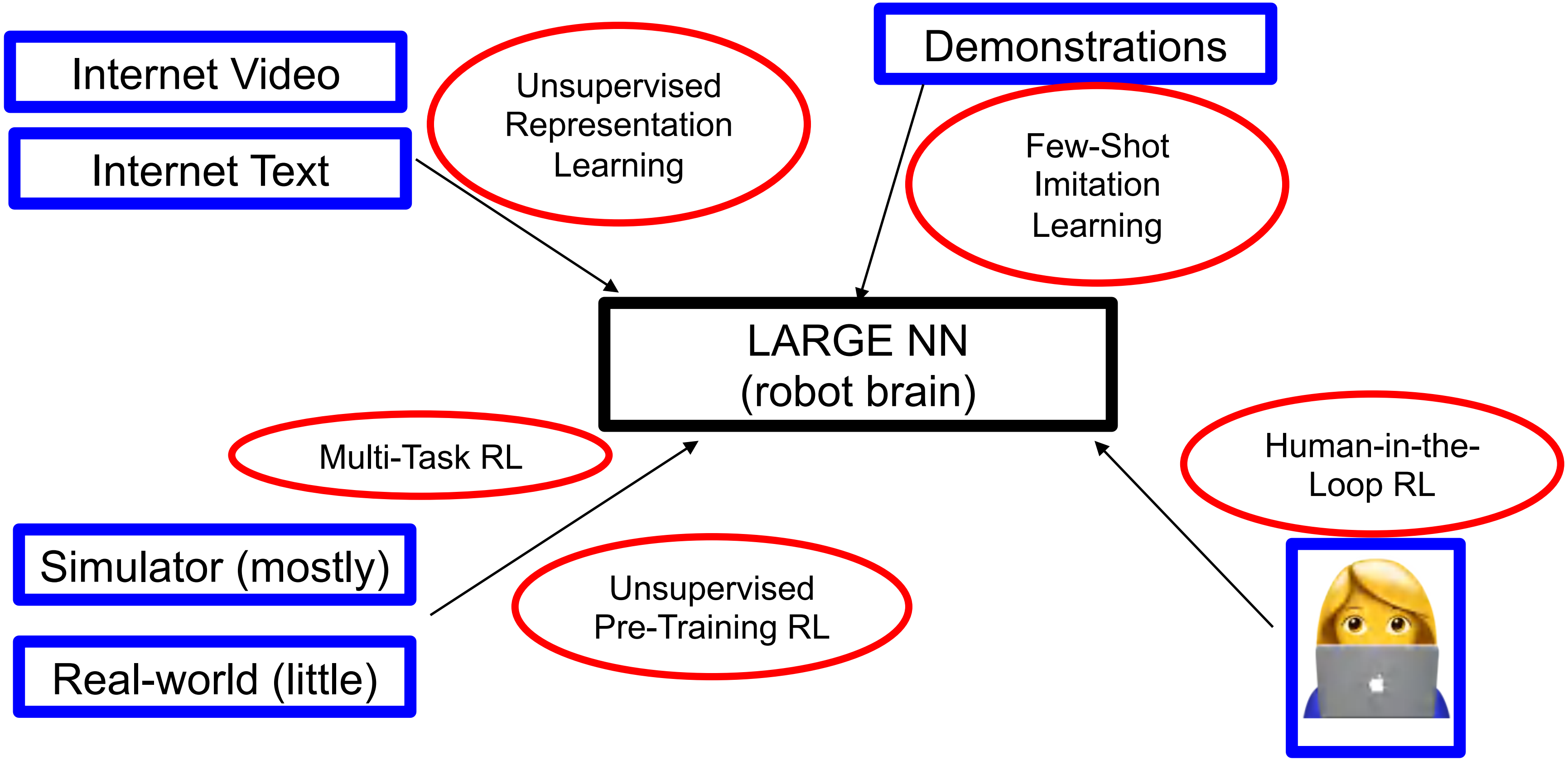

HYUNDAI
333-4



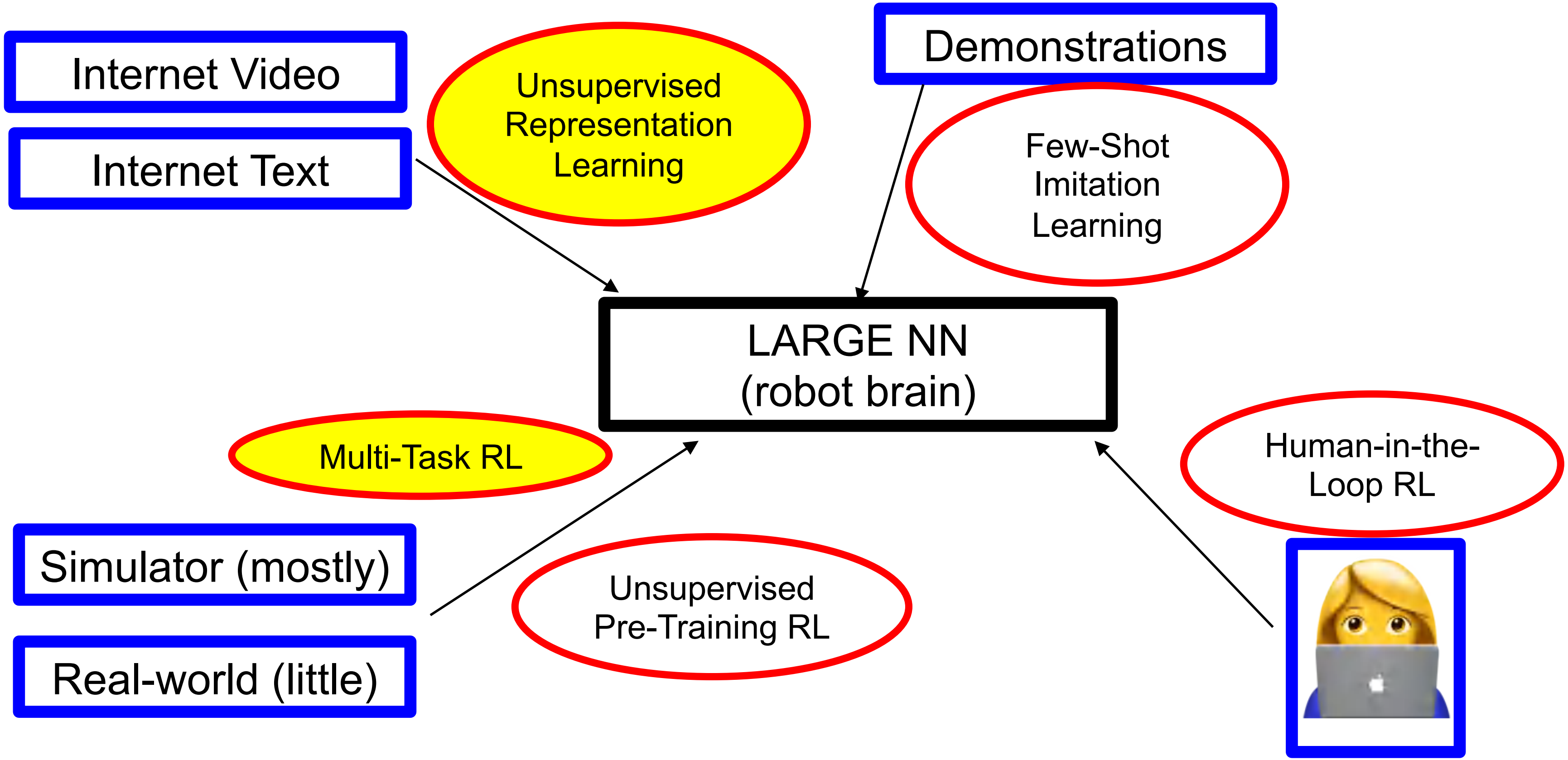
Open-ended Environments

- Vision:
  - Pre-train on ImageNet -> finetune for other tasks
- NLP (GPT-x,BERT):
  - Pre-train on internet text -> finetune for other tasks
- Robotics:
  - ?????????????????????? -> finetune for other tasks

An Attempt at a Complete Picture

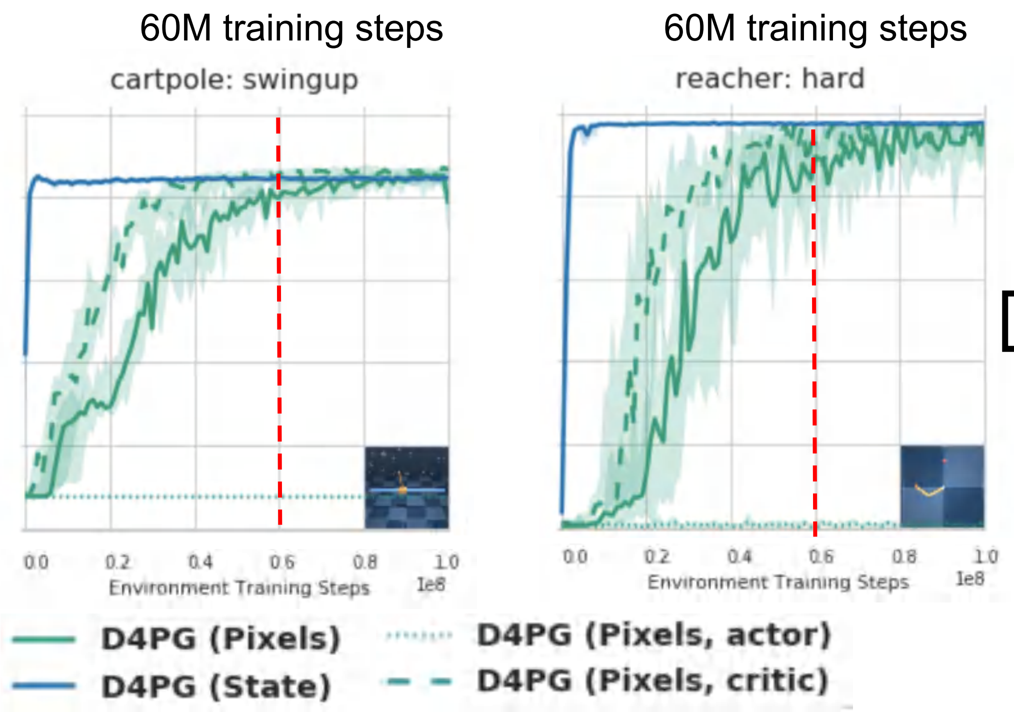


An Attempt at a Complete Picture



RL-from-pixels?

- State-based D4PG (blue) vs pixel-based D4PG (green)



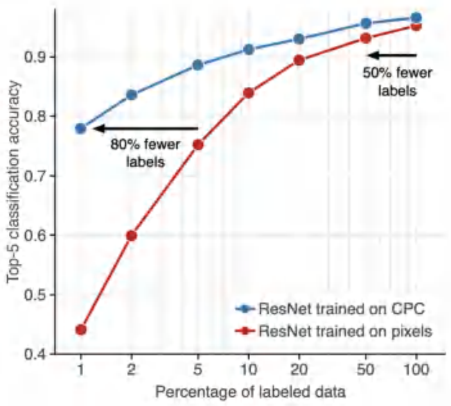
Pixel-based needs > 50M more training steps than state-based to solve same tasks

[Tassa et al., 2018] Tassa, Y., Doron, Y., Muldal, A., Erez, T., Li, Y., Casas, D.D.L., Budden, D., Abdolmaleki, A., Merel, J., Lefrancq, A. and Lillicrap, T DeepMind Control Suite, arxiv:1801.00690, 2018.

LeCake (Yann LeCun)

Contrastive learning: SOTA in computer vision

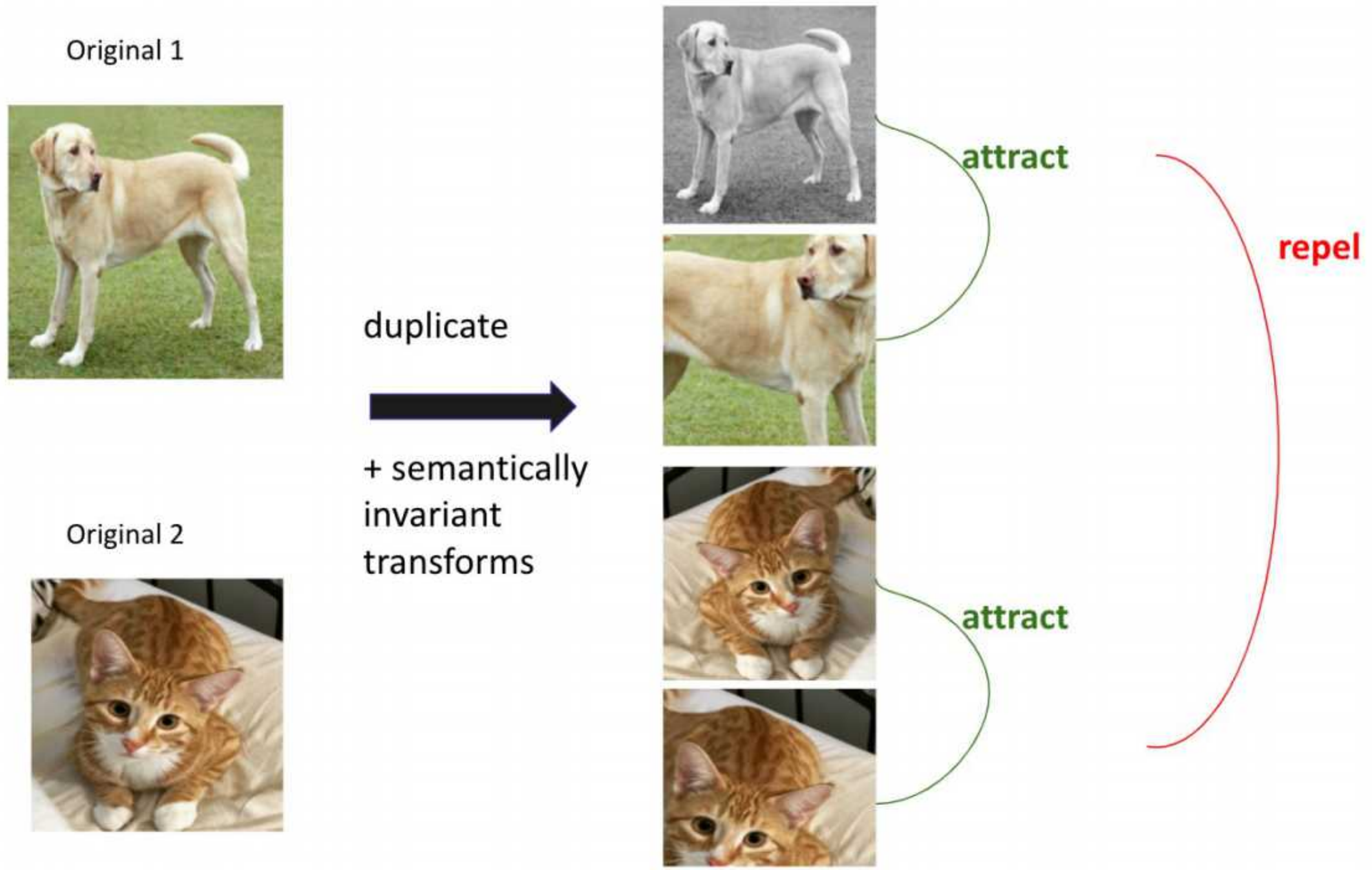
CPCv2 **top-5** ImageNet accuracy as function of labels



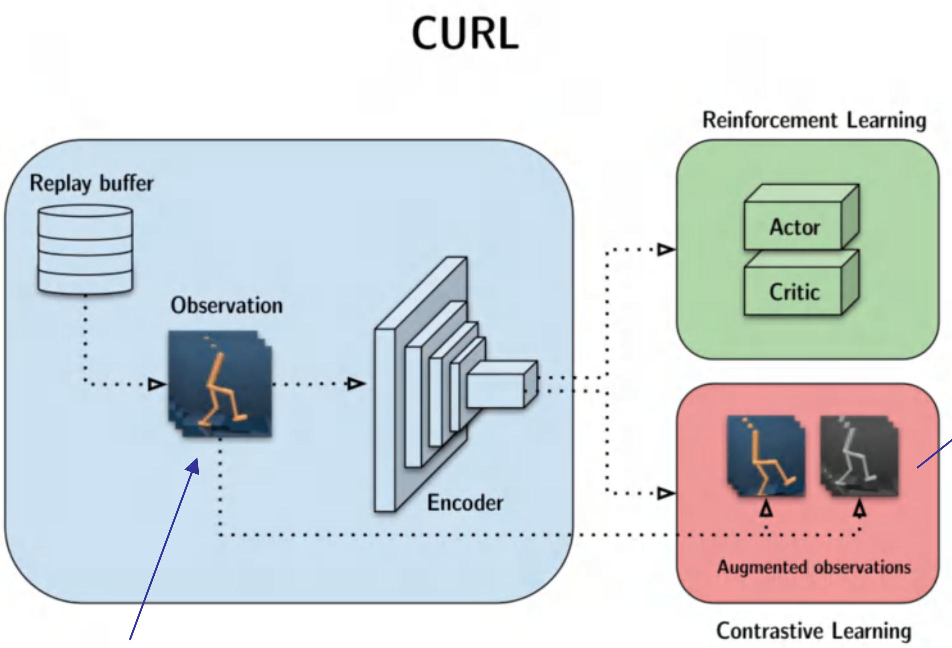
[Henaff, Srinivas et al., 2019]

[Henaff et al., 2019] Olivier J. Hénaff, Aravind Srinivas, Jeffrey De Fauw, Ali Razavi, Carl Doersch, S. M. Ali Eslami, Aaron van den Oord
Data-Efficient Image Recognition with Contrastive Coding arxiv:1905.09272, 2019.
[Chen et al., 2020] Chen, T., Kornblith, S., Norouzi, M. and Hinton, G.
A Simple Framework for Contrastive Learning of Visual Representations arxiv:2002.05709, 2020.

SimCLR / MoCo Main Idea



Contrastive + RL

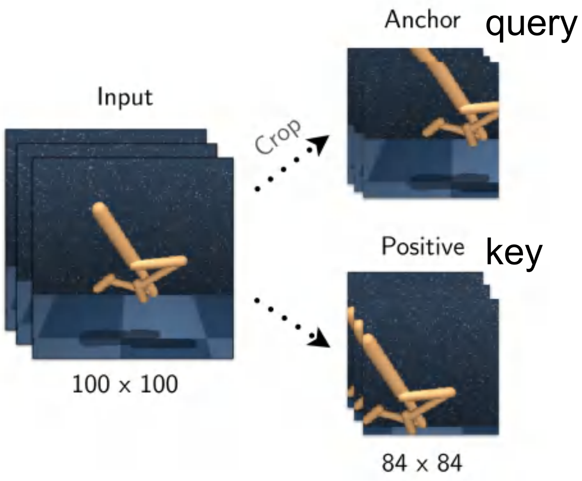


Need to define:
1. query / key pairs
2. similarity measure
3. architecture

Observations are stacked frames

[CURL: A Srinivas*, M Laskin*, P Abbeel, 2020]

1. Query / key pairs: random crop

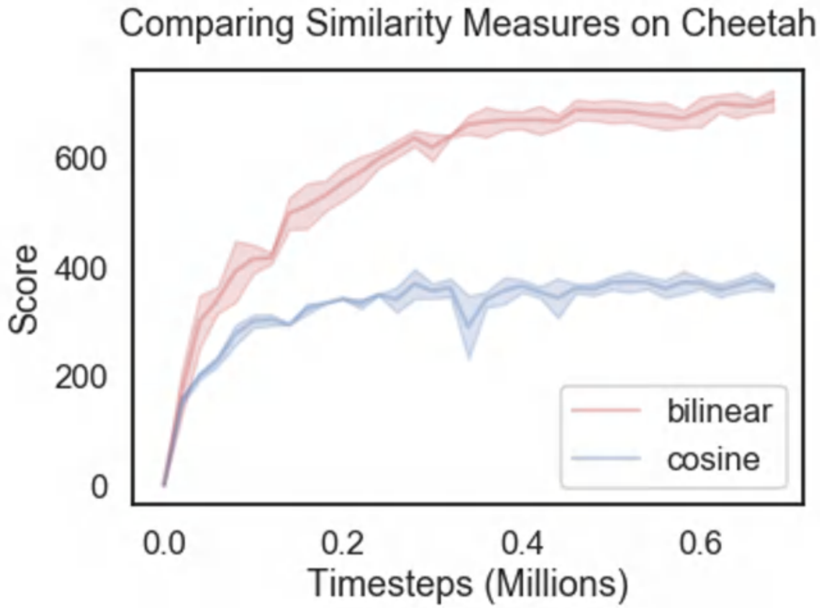


[CURL: A Srinivas*, M Laskin*, P Abbeel, 2020]

2. Bilinear inner product with learned weight matrix

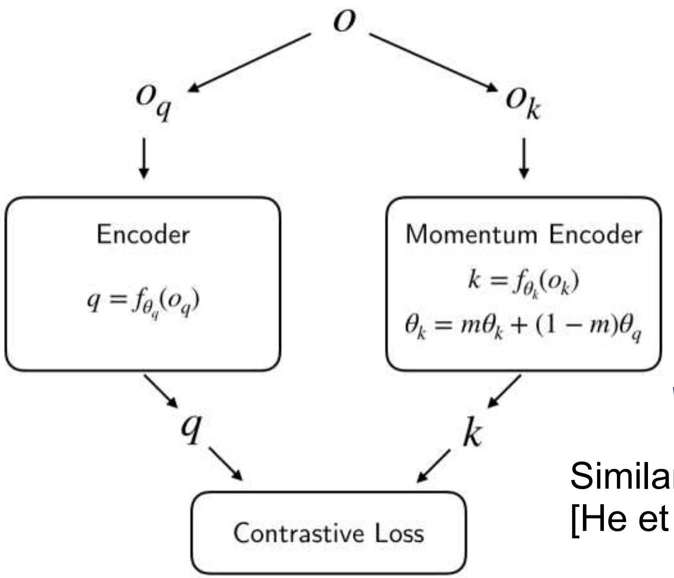
$$\begin{array}{cc} \text{logits} & \text{labels} \\ \begin{bmatrix} q_0^T W k_0 & q_0^T W k_1 & \cdots & q_0^T W k_j \\ q_1^T W k_0 & q_1^T W k_1 & \cdots & q_1^T W k_j \\ \vdots & \vdots & \ddots & \vdots \\ q_j^T W k_0 & q_j^T W k_1 & \cdots & q_k^T W k_j \end{bmatrix} & \begin{bmatrix} 1 & 0 & \dots & 0 \\ 0 & 1 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & 1 \end{bmatrix} \end{array}$$

$$\mathscr{L}_q = \frac{\exp(q^T W k_+)}{\exp(q^T W k_+) + \sum_{i=0}^{K-1} \exp(q^T W k_i)}$$

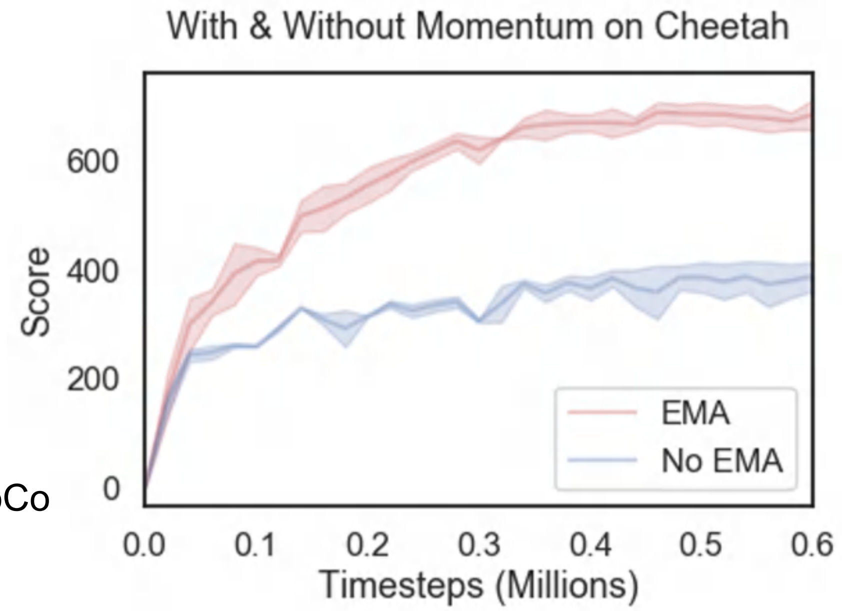


[CURL: A Srinivas*, M Laskin*, P Abbeel, 2020]

3. Keys encoded with momentum



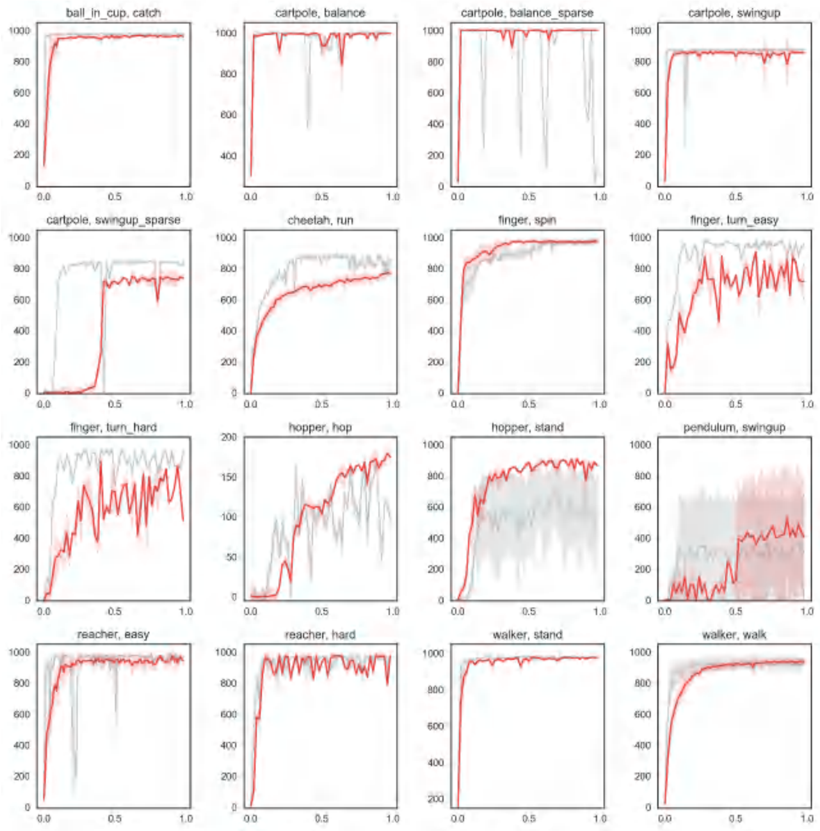
Similar to MoCo
[He et al.]



[He et al., 2019] He, K., Fan, H., Wu, Y., Xie, S. and Girshick, R. Momentum Contrast for Unsupervised Visual Representation Learning arxiv:1911.05722

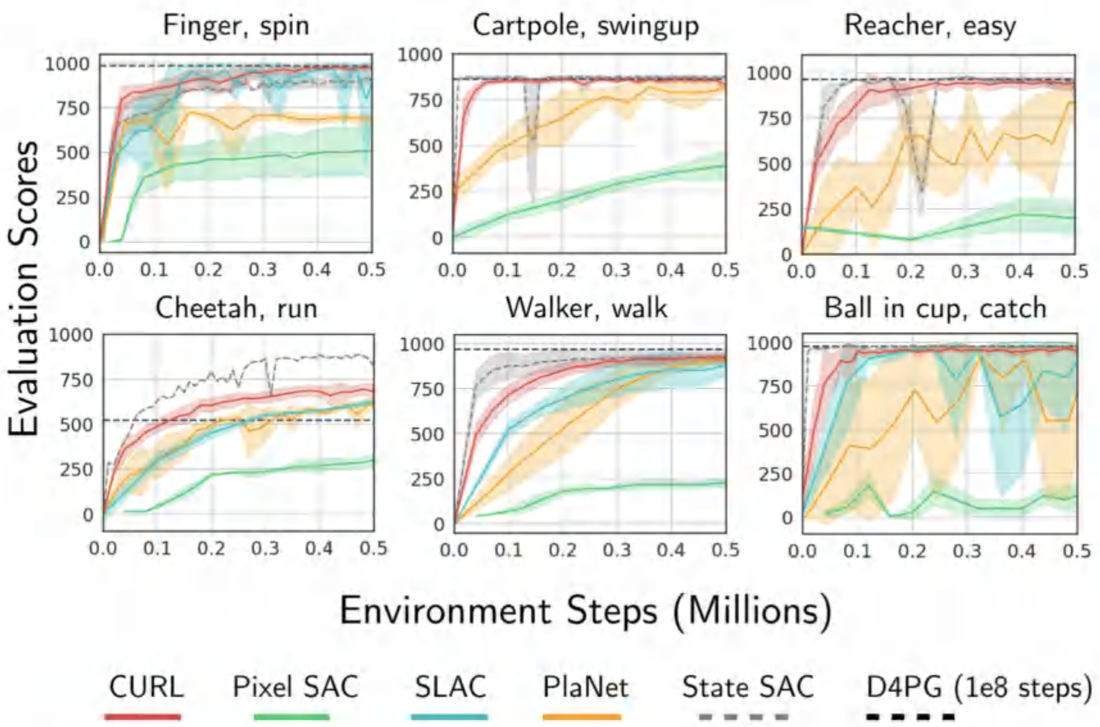
CURL from pixels matches state-based SAC

GRAY: SAC State
RED: CURL



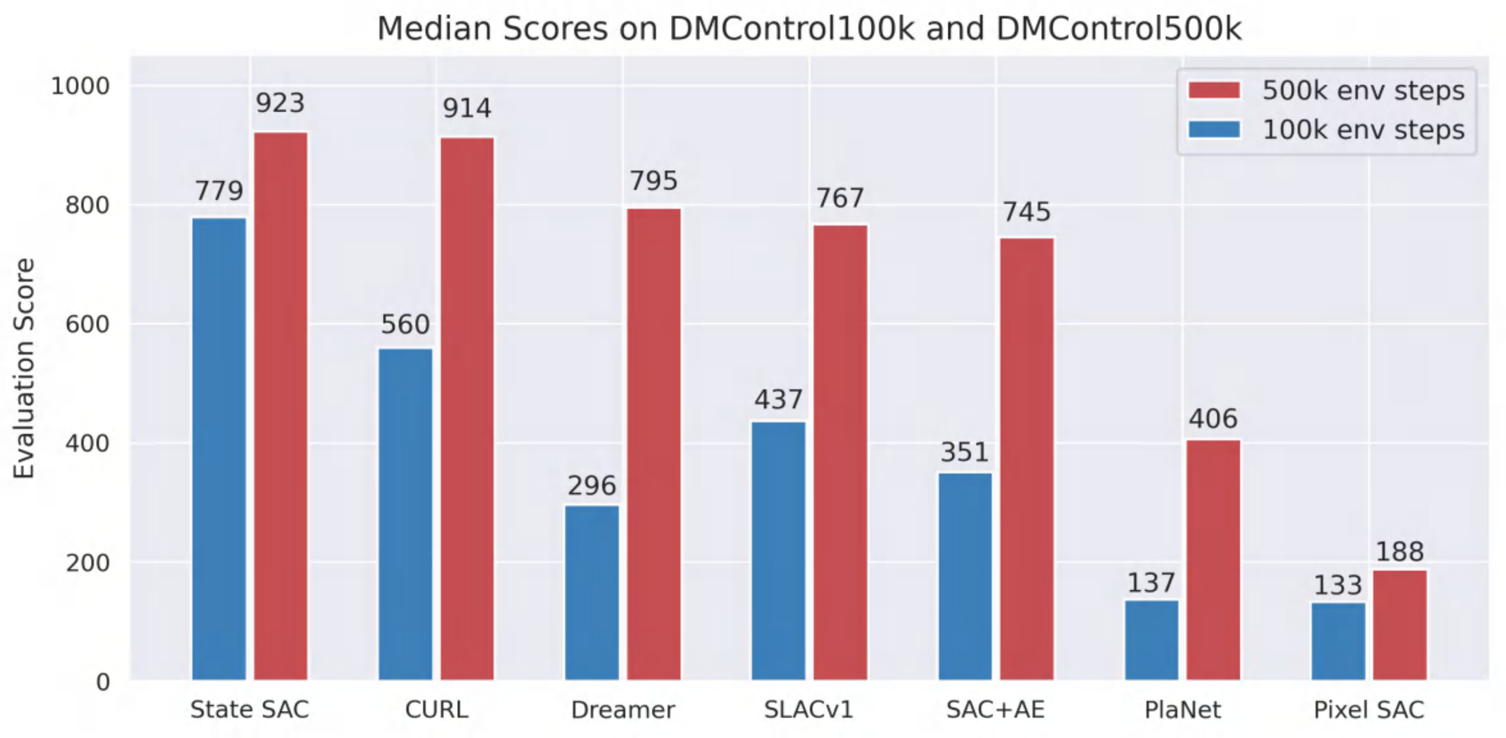
[CURL: A Srinivas*, M Laskin*, P Abbeel, 2020]

CURL Comparison: DeepMind Control Suite



[CURL: A Srinivas*, M Laskin*, P Abbeel, 2020]

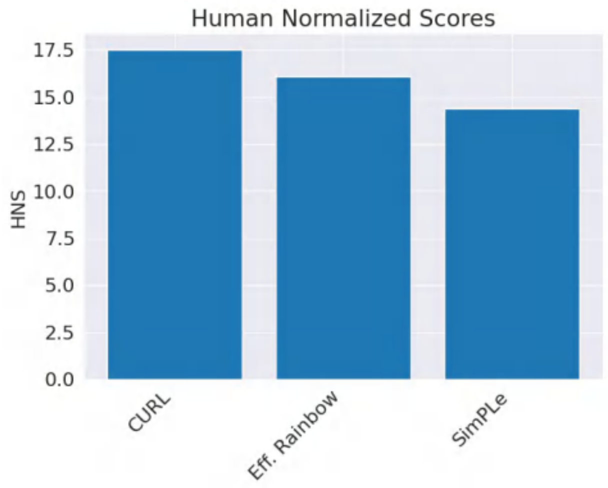
CURL Comparison: DeepMind Control Suite



[CURL: A Srinivas*, M Laskin*, P Abbeel, 2020]

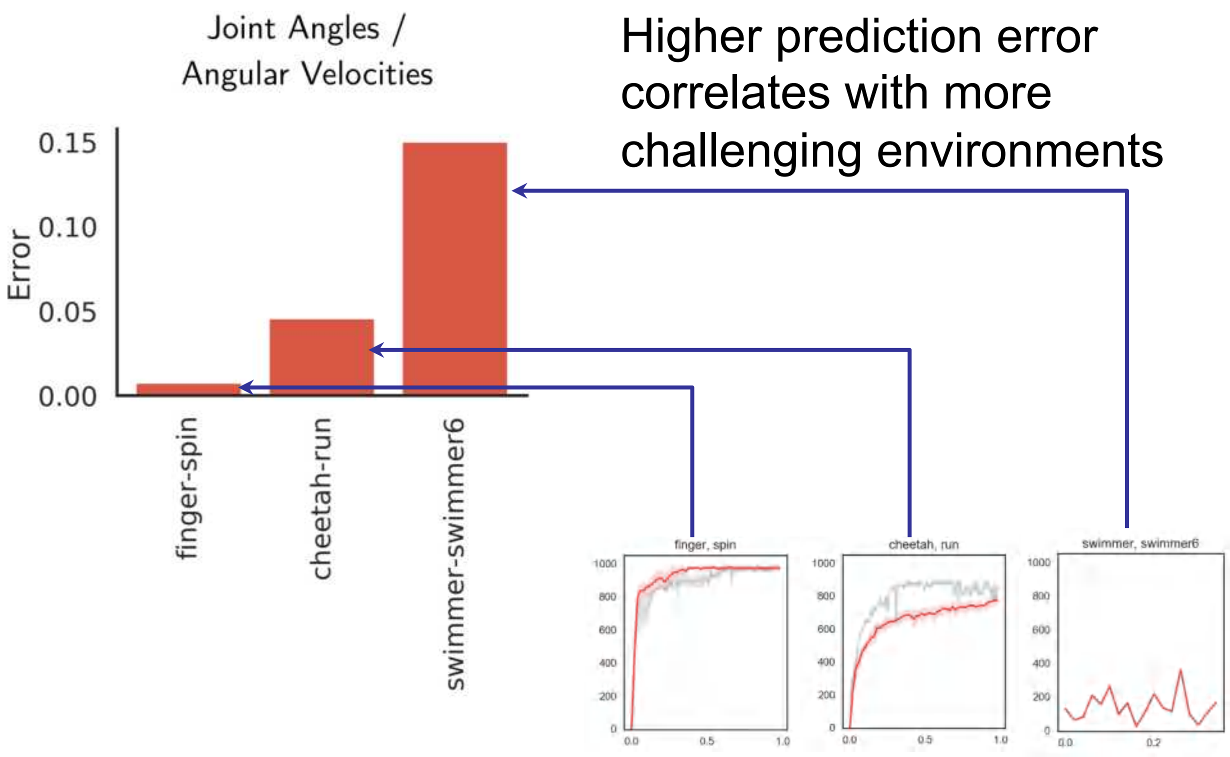
CURL Comparison: Atari 100K

Game	Human	Random	Rainbow	SimPLe	OTRainbow	Eff. Rainbow	CURL
Alien	7127.7	227.8	318.7	616.9	**824.7**	739.9	558.2
Amidar	1719.5	5.8	32.5	88.0	82.8	**188.6**	142.1
Assault	742.0	222.4	231	527.2	351.9	431.2	**600.6**
Asterix	8503.3	210.0	243.6	**1128.3**	628.5	470.8	734.5
Bank Heist	753.1	14.2	15.55	34.2	**182.1**	51.0	131.6
Battle Zone	37187.5	2360.0	2360.0	5184.4	4060.6	10124.6	**14870.0**
Boxing	12.1	0.1	-24.8	**9.1**	2.5	0.2	1.2
Breakout	30.5	1.7	1.2	**16.4**	9.84	1.9	4.9
Chopper Command	7387.8	811.0	120.0	**1246.9**	1033.33	861.8	1058.5
Crazy_Climber	35829.4	10780.5	2254.5	**62583.6**	21327.8	16185.3	12146.5
Demon_Attack	1971.0	152.1	163.6	208.1	711.8	508.0	**817.6**
Freeway	29.6	0.0	0.0	20.3	25.0	**27.9**	26.7
Frostbite	4334.7	65.2	60.2	254.7	231.6	866.8	**1181.3**
Gopher	2412.5	257.6	431.2	771.0	**778.0**	349.5	669.3
Hero	30826.4	1027.0	487	2656.6	6458.8	**6857.0**	6279.3
Jamesbond	302.8	29.0	47.4	125.3	112.3	301.6	**471.0**
Kangaroo	3035.0	52.0	0.0	323.1	605.4	779.3	**872.5**
Krull	2665.5	1598.0	1468	**4539.9**	3277.9	2851.5	4229.6
Kung_Fu_Master	22736.3	258.5	0.	**17257.2**	5722.2	14346.1	14307.8
Ms_Pacman	6951.6	307.3	67	**1480.0**	941.9	1204.1	1465.5
Pong	14.6	-20.7	-20.6	**12.8**	1.3	-19.3	-16.5
Private Eye	69571.3	24.9	0	58.3	100.0	97.8	**218.4**
Qbert	13455.0	163.9	123.46	**1288.8**	509.3	1152.9	1042.4
Road_Runner	7845.0	11.5	1588.46	5640.6	2696.7	**9600.0**	5661.0
Seaquest	42054.7	68.4	131.69	**683.3**	286.92	354.1	384.5
Up_N_Down	11693.2	533.4	504.6	**3350.3**	2847.6	2877.4	2955.2



[CURL: A Srinivas*, M Laskin*, P Abbeel, 2020]

Predicting state from pixels

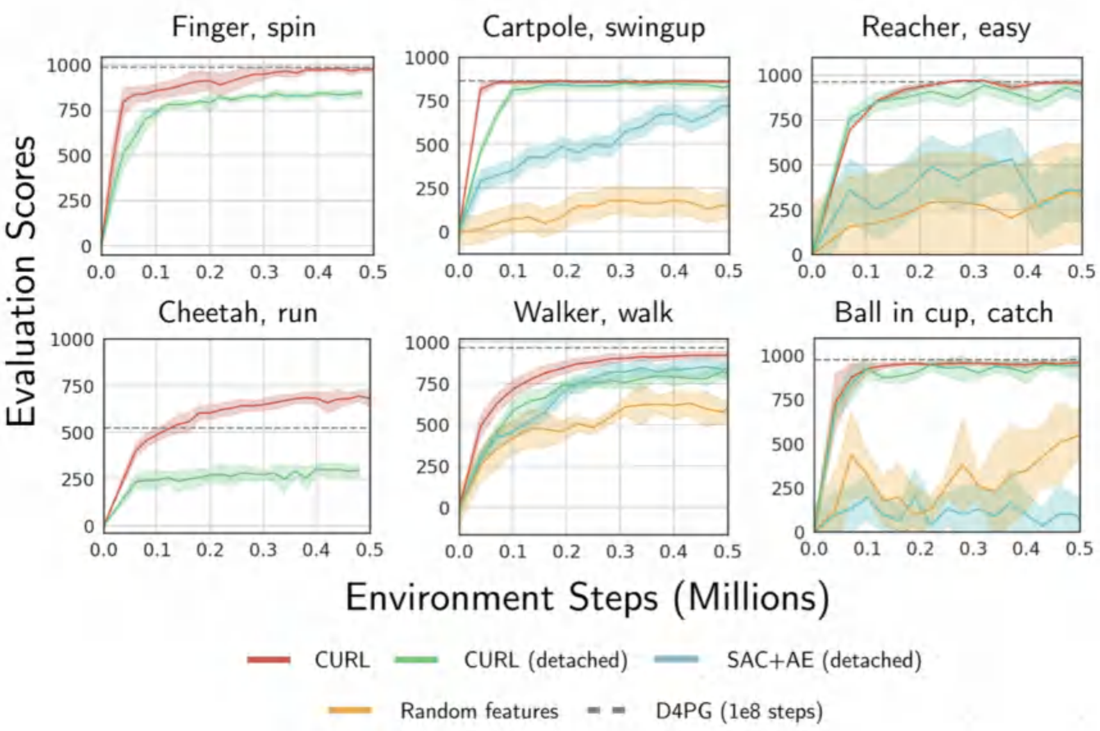


Higher prediction error correlates with more challenging environments

[CURL: A Srinivas*, M Laskin*, P Abbeel, 2020]

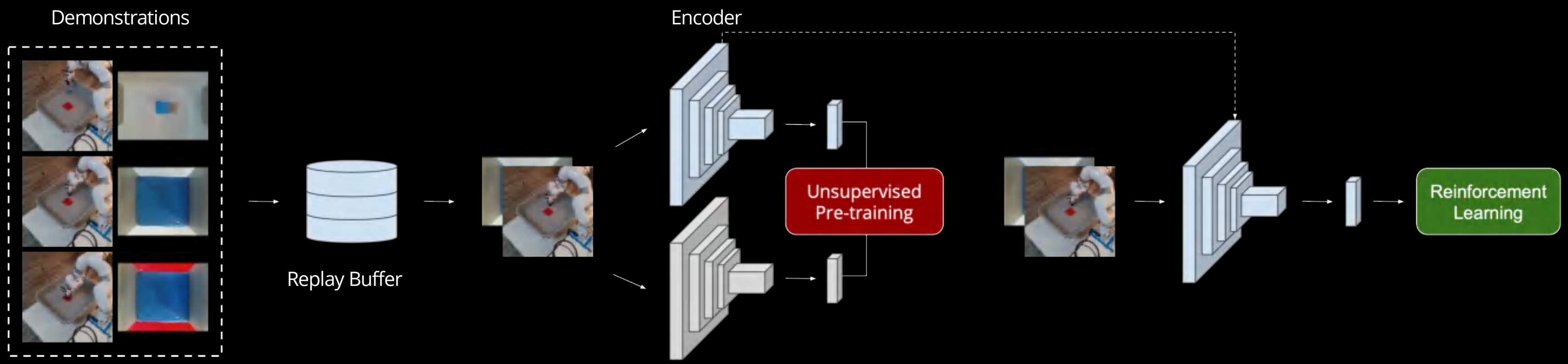
Can CURL learn representations w/o reward?

1. Detached CURL performs slightly worse than CURL
2. However, promising for learning representations independent of reward



[CURL: A Srinivas*, M Laskin*, P Abbeel, 2020]

Framework for Efficient Robotic Manipulation



(i) Collect 10 human demonstrations

Takes ~10 mins

(ii) Initialize CNN encoder with contrastive pre-training

Takes ~1 min

(iii) Continue training with data-augmented RL

Takes ~30 mins

Task: Pull

Demonstrations

10 ep ≈ 10:00 min

First Success

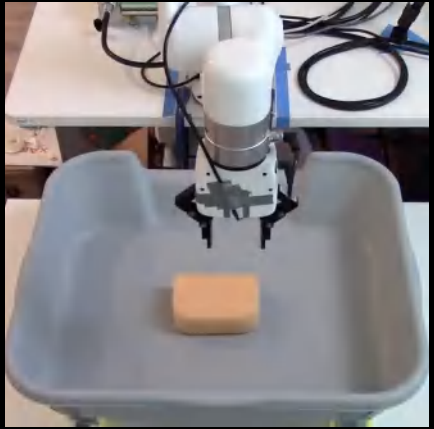
5 ep ≈ 5:12 min

Optimal Policy

45 ep ≈ 29:10 min

Evaluation

28/30 Success



Results

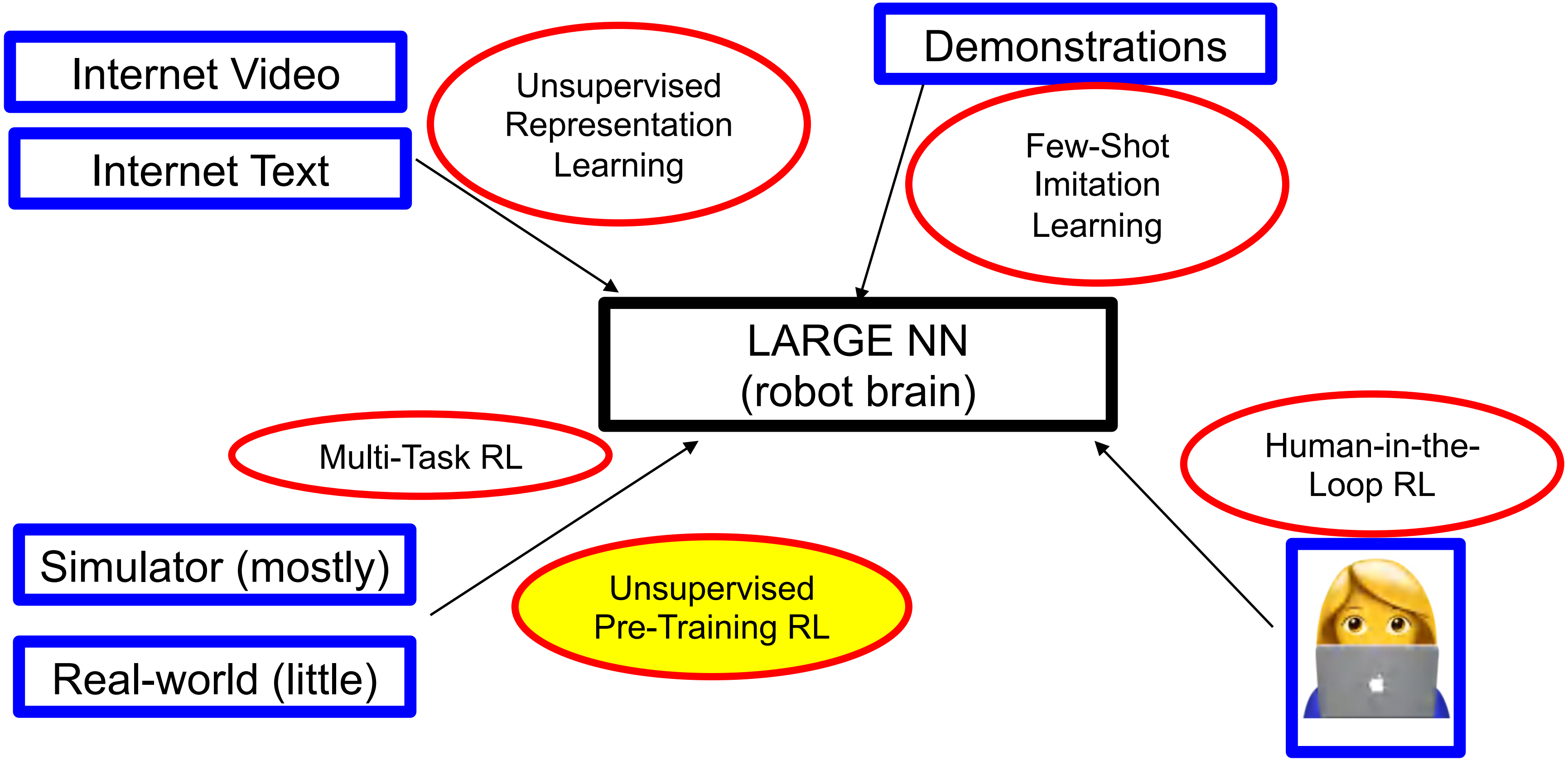
1. Learns **6 diverse tasks** with sparse reward, entirely from pixels, within an hour.
2. Uses the **same hyperparameters** across all tasks

	Reach	**Pickup**	**Move**	**Pull**	**Light Switch**	**Drawer Open**
Task Description + Difficulty	Reach a block	Pickup a block	Move a block to a given location	Pull a large object to itself	Flip on the Light Switch	Open the drawer
First Success	3:05	15:00	33:00	05:12	05:01	5:56
Optimal	**15:00**	**26:00**	**46:00**	**29:10**	**16:05**	**20:21**
Evaluation	100%	100%	86.7%	93.3%	100%	100%

Related Work

- Auto-encoder representation
  - SAC+AE – Yarats, Zhang, Kostrikov, Amos, Pineau, Fergus, 2019
  - SLAC – Lee, Nagabandi, Abbeel, levine, 2019
- Augmentation can go a long way
  - RAD – Laskin, lee, Stooke, Pinto, Abbeel, Srinivas, 2020
  - DrQ – Kostrikov, Yarats, Fergus, 2020
  - SPR – Schwarzer, Anand, Goel, Hjelm, Courville, Bachman, 2020
- Decoupling representation learning from RL with augmented temporal contrast
  - ATC – Stooke, Lee, Abbeel, Laskin, 2020
  - Deep InfoMax RL – Mazoure, des Combes, Doan, Bachman, Hjelm, 2020 (temporal, not always decoupled)
- Application to real robot
  - FERM – Zhan, Zhao, Pinto, Abbeel, Laskin, 2020

An Attempt at a Complete Picture



Intrinsic Reward for Unsupervised Pre-Training

- Incentivizing exploration by introducing intrinsic rewards based on a measure of state novelty

- State entropy as intrinsic reward

$$r^{\texttt{intrinsic}} = \mathcal{H}(s) = -\mathbb{E}_{s\sim p(s)}\left[\log p(s)\right]$$

  - Maximizing state entropy ~= good state coverage

[MEPOL – Mutti, Pratissoli, Restelli, 2020]

Intrinsic Reward for Unsupervised Pre-Training

- Incentivizing exploration by introducing intrinsic rewards based on a measure of state novelty

- State entropy as intrinsic reward

$$r^{\texttt{intrinsic}} = \mathcal{H}(s) = -\mathbb{E}_{s\sim p(s)}\left[\log p(s)\right]$$

  - Maximizing state entropy ~= good state coverage

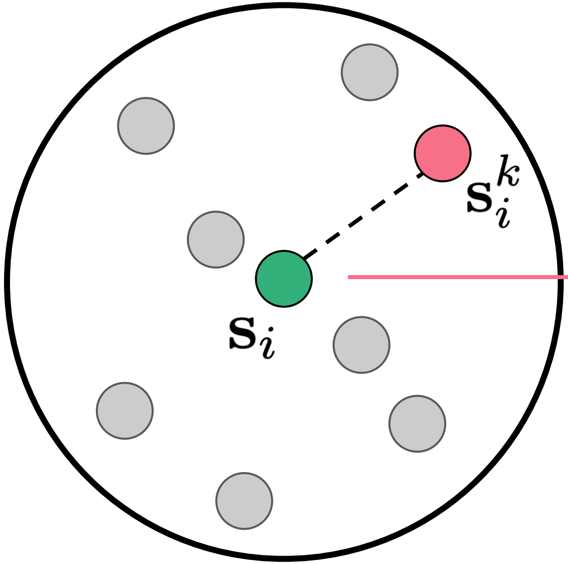
⇨ **Measuring state entropy is intractable** to compute in most setting

[MEPOL – Mutti, Pratissoli, Restelli, 2020]

K-Nearest-Neighbor Entropy Estimator

- *K*-nearest entropy estimator

$$\mathcal{H}(s) = -\mathbb{E}_{s\sim p(s)}\left[\underline{\log p(s)}\right]$$

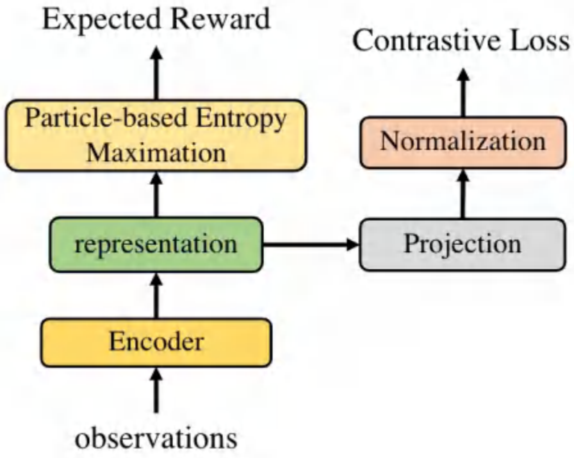


$$\widehat{\mathcal{H}}(\mathbf{s}) \propto \sum_i \boxed{\log(||\mathbf{s}_i - \mathbf{s}_i^k||)}$$

- Distribution → Store N number of visited states
- Compute the distance between each state and its K-NN

Singh, H., et al., 2003. Nearest neighbor estimates of entropy. *American journal of mathematical and management sciences*, *23*(3-4), pp.301-321.
MEPOL – Mutti, Pratissoli, Restelli, 2020

APT: Active Pre-Training

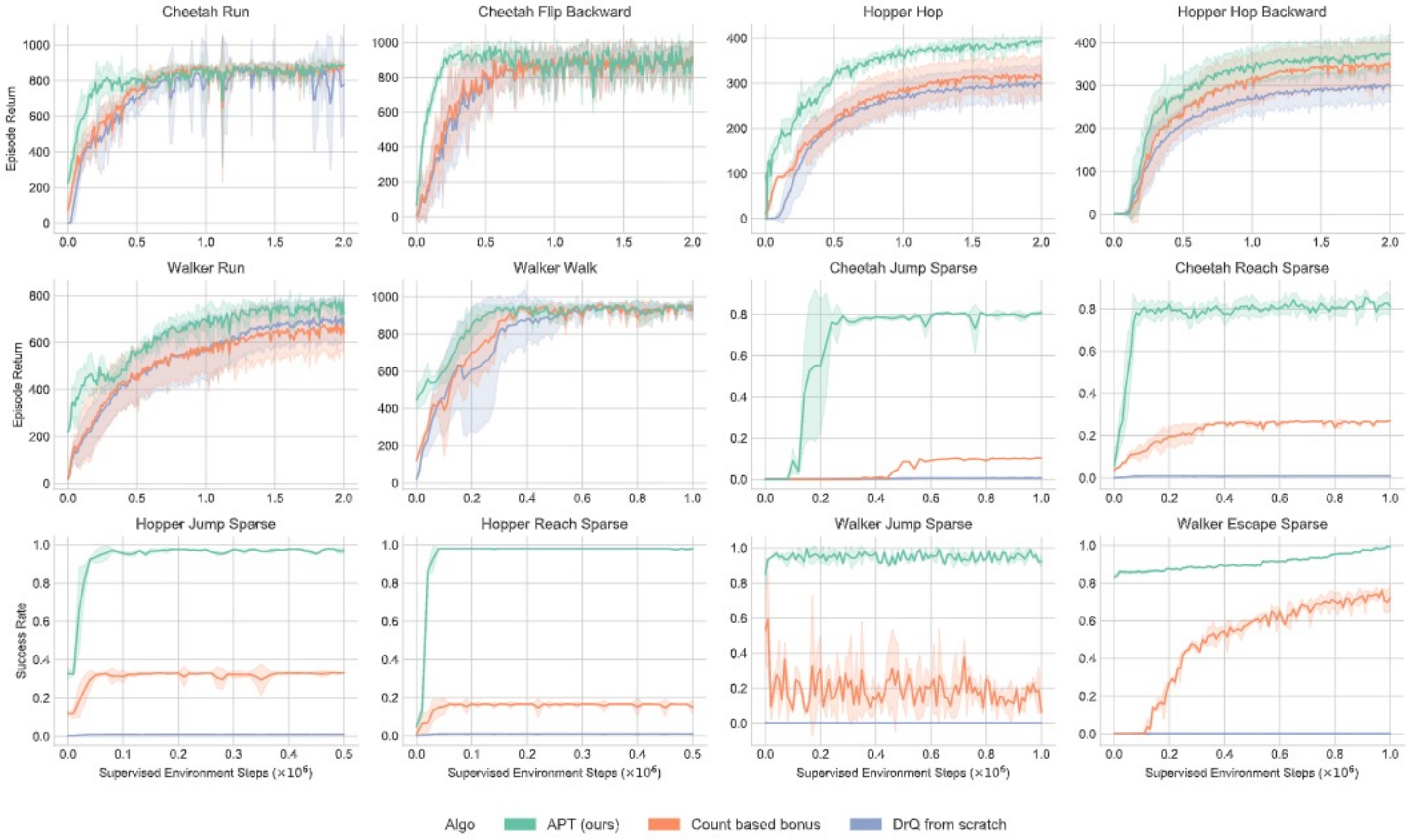


Algorithm 1: Training APT

Randomly Initialize f encoder
Randomly Initialize π and Q networks
for $e := 1, \infty$ **do**
  for $t := 1, T$ **do**
    Receive observation s_t from environment
    Take action $a_t \sim \pi(\cdot|s_t)$, receive observation s_{t+1} and r_t from environment
    $\mathcal{D} \leftarrow \mathcal{D} \cup (s_t, a_t, r_t, s_t')$
    $\{(s_i, a_i, r_i, s_i')\}_{i=1}^N \sim \mathcal{D}$ `// sample a mini batch`
    Train neural encoder f on mini batch `// representation learning`
    for *each* $i = 1..N$ **do**
      $a_i' \sim \pi(\cdot|s_i')$
      $\hat{Q}_i = Q_{\theta'}(s_i', a_i')$
      Compute r_{APT} with equation (5) `// particle-based entropy reward`
      $y_i \leftarrow r_{\text{APT}} + \gamma \hat{Q}_i$
    end
    $loss_Q = \sum_i (Q(s_i, a_i) - y_i)^2$
    Gradient descent step on Q and π `// standard Q-learning`
  end
end

[H Liu & P Abbeel, 2020]

Experiments: DM Control Suite

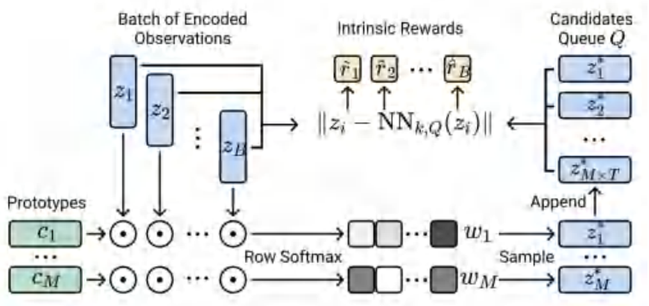


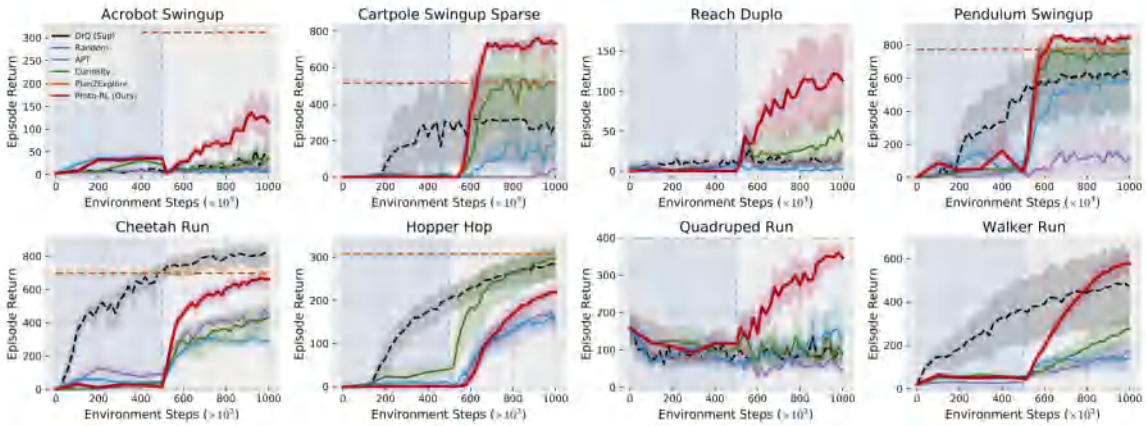
Experiments: Atari

Game	Random	Human	SimPLe	DER	CURL	DrQ	SPR	VISR	APT (ours)
Alien	227.8	7127.7	616.9	739.9	558.2	771.2	801.5	364.4	**2614.8**
Amidar	5.8	1719.5	88.0	188.6	142.1	102.8	176.3	186.0	**211.5**
Assault	222.4	742.0	527.2	431.2	600.6	452.4	571.0	**12091.1**	891.5
Asterix	210.0	8503.3	1128.3	470.8	734.5	603.5	977.8	**6216.7**	185.5
Bank Heist	14.2	753.1	34.2	51.0	131.6	168.9	380.9	71.3	**416.7**
BattleZone	2360.0	37187.5	5184.4	10124.6	14870.0	12954.0	**16651.0**	7072.7	7065.1
Boxing	0.1	12.1	9.1	0.2	1.2	6.0	**35.8**	13.4	21.3
Breakout	1.7	30.5	16.4	1.9	4.9	16.1	17.1	**17.9**	10.9
ChopperCommand	811.0	7387.8	**1246.9**	861.8	1058.5	780.3	974.8	800.8	317.0
Crazy Climber	10780.5	23829.4	**62583.6**	16185.2	12146.5	20516.5	42923.6	49373.9	44128.0
Demon Attack	107805	35829.4	62583.6	16185.3	12146.5	20516.5	42923.6	**8994.9**	5071.8
Freeway	0.0	29.6	20.3	27.9	26.7	9.8	24.4	-12.1	**29.9**
Frostbite	65.2	4334.7	254.7	866.8	1181.3	331.1	**1821.5**	230.9	1796.1
Gopher	257.6	2412.5	771.0	349.5	669.3	636.3	715.2	498.6	**2590.4**
Hero	1027.0	30826.4	2656.6	6857.0	6279.3	3736.3	**7019.2**	663.5	6789.1
Jamesbond	29.0	302.8	125.3	301.6	471.0	236.0	365.4	**484.4**	356.1
Kangaroo	52.0	3035.0	323.1	779.3	872.5	940.6	**3276.4**	1761.9	412.0
Krull	1598.0	2665.5	**4539.9**	2851.5	4229.6	4018.1	2688.9	3142.5	2312.0
Kung Fu Master	258.5	22736.3	17257.2	14346.1	14307.8	9111.0	13192.7	16754.9	**17357.0**
Ms Pacman	307.3	6951.6	1480.0	1204.1	1465.5	960.5	1313.2	558.5	**2827.1**
Pong	-20.7	14.6	**12.8**	-19.3	-16.5	-8.5	-5.9	-26.2	-8.0
Private Eye	24.9	69571.3	58.3	97.8	218.4	-13.6	**124.0**	98.3	96.1
Qbert	163.9	13455.0	1288.8	1152.9	1042.4	854.4	669.1	666.3	**17671.2**
Road Runner	11.5	7845.0	5640.6	9600.0	5661.0	8895.1	**14220.5**	6146.7	4782.1
Seaquest	68.4	42054.7	683.3	354.1	384.5	301.2	583.1	706.6	**2116.7**
Up N Down	533.4	11693.2	3350.3	2877.4	2955.2	3180.8	**28138.5**	10037.6	8289.4
Mean HNS	0.000	1.000	44.3	28.5	38.1	35.7	**70.4**	64.31	69.55
Median HNS	0.000	1.000	14.4	16.1	17.5	26.8	41.5	12.36	**47.50**
# Superhuman	0	N/A	2	2	2	2	**7**	6	**7**

How about size of replay buffer for entropy estimates?

→ Keep around cluster representatives for entropy estimation





Reinforcement Learning with Prototypical Representations, Yarats, Fergus, Lazarus, Pinto, 2021

How about “skills”?

- **VIC**: Variational Intrinsic Control – Gregor et al, 2016
  DIAYN: Diversity is all you need – Eysenbach, Gupta, Ibarz, Levine, 2018
  Valor: Variational Option Discovery Algorithms – Achiam, Edwards, Amodei, Abbeel, 2018
  VISR: Fast Task Inference with Variational Intrinsic Successor Features – Hansen et al, 2020

- They all optimize (up to some details):

MI (z ; s_{0:H}) = H(z) – H(z | s_{0:H})

APS Active Pretraining with Successor Features:

-- optimize **H(s_{0:H}) – H(s_{0:H} | z)**
using the particle entropy and feature learning as in APT
--from image inputs

Closely related: EDL: Explore, Discover and Learn – Campos et al, 2020

APS on Atari

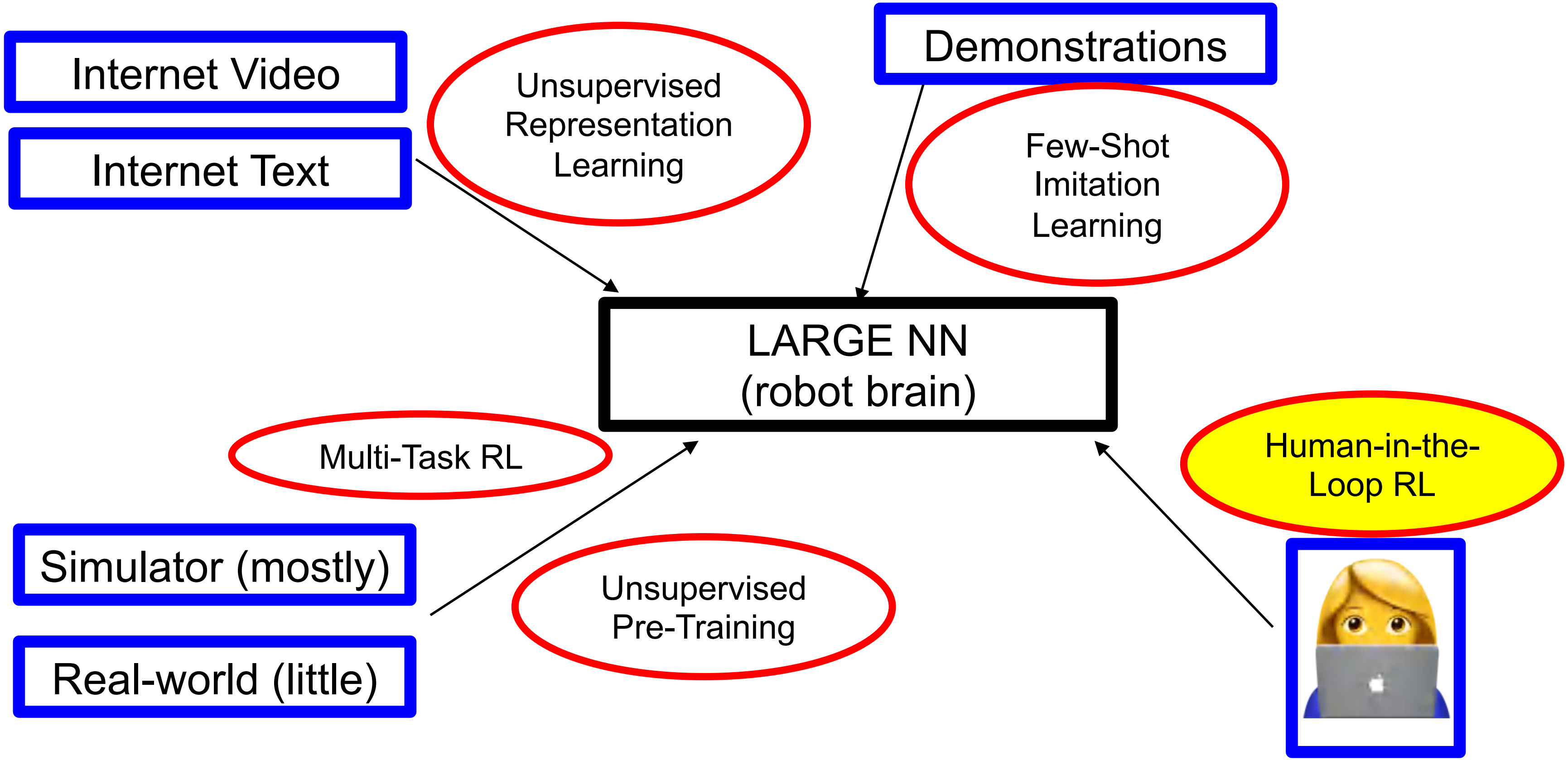
Game	Random	Human	SimPLe	DER	CURL	DrQ	SPR	VISR	APT	APS (ours)
Alien	227.8	7127.7	616.9	739.9	558.2	771.2	801.5	364.4	**2614.8**	934.9
Amidar	5.8	1719.5	88.0	188.6	142.1	102.8	176.3	186.0	**211.5**	178.4
Assault	222.4	742.0	527.2	431.2	600.6	452.4	571.0	**12091.1**	891.5	413.3
Asterix	210.0	8503.3	1128.3	470.8	734.5	603.5	977.8	**6216.7**	185.5	1159.7
Bank Heist	14.2	753.1	34.2	51.0	131.6	168.9	380.9	71.3	**416.7**	262.7
BattleZone	2360.0	37187.5	5184.4	10124.6	14870.0	12954.0	16651.0	7072.7	7065.1	**26920.1**
Boxing	0.1	12.1	9.1	0.2	1.2	6.0	35.8	13.4	21.3	**36.3**
Breakout	1.7	30.5	16.4	1.9	4.9	16.1	17.1	17.9	10.9	**19.1**
ChopperCommand	811.0	7387.8	1246.9	861.8	1058.5	780.3	974.8	800.8	317.0	**2517.0**
Crazy Climber	10780.5	23829.4	62583.6	16185.2	12146.5	20516.5	42923.6	49373.9	44128.0	**67328.1**
Demon Attack	107805	35829.4	62583.6	16185.3	12146.5	20516.5	42923.6	**8994.9**	5071.8	7989.0
Freeway	0.0	29.6	20.3	27.9	26.7	9.8	24.4	-12.1	**29.9**	27.1
Frostbite	65.2	4334.7	254.7	866.8	1181.3	331.1	**1821.5**	230.9	1796.1	496.5
Gopher	257.6	2412.5	771.0	349.5	669.3	636.3	715.2	498.6	**2590.4**	2386.5
Hero	1027.0	30826.4	2656.6	6857.0	6279.3	3736.3	7019.2	663.5	6789.1	**12189.3**
Jamesbond	29.0	302.8	125.3	301.6	471.0	236.0	365.4	484.4	356.1	**622.3**
Kangaroo	52.0	3035.0	323.1	779.3	872.5	940.6	3276.4	1761.9	412.0	**5280.1**
Krull	1598.0	2665.5	**4539.9**	2851.5	4229.6	4018.1	2688.9	3142.5	2312.0	4496.0
Kung Fu Master	258.5	22736.3	17257.2	14346.1	14307.8	9111.0	13192.7	16754.9	17357.0	**22412.0**
Ms Pacman	307.3	6951.6	1480.0	1204.1	1465.5	960.5	1313.2	558.5	**2827.1**	2092.3
Pong	-20.7	14.6	**12.8**	-19.3	-16.5	-8.5	-5.9	-26.2	-8.0	12.5
Private Eye	24.9	69571.3	58.3	97.8	218.4	-13.6	**124.0**	98.3	96.1	117.9
Qbert	163.9	13455.0	1288.8	1152.9	1042.4	854.4	669.1	666.3	17671.2	**19271.4**
Road Runner	11.5	7845.0	5640.6	9600.0	5661.0	8895.1	**14220.5**	6146.7	4782.1	5919.0
Seaquest	68.4	42054.7	683.3	354.1	384.5	301.2	583.1	706.6	2116.7	**4209.7**
Up N Down	533.4	11693.2	3350.3	2877.4	2955.2	3180.8	**28138.5**	10037.6	8289.4	4911.9
Mean Human-Norm'd	0.000	1.000	44.3	28.5	38.1	35.7	70.4	64.31	69.55	**99.04**
Median Human-Norm'd	0.000	1.000	14.4	16.1	17.5	26.8	41.5	12.36	47.50	**58.80**
# Superhuman	0	N/A	2	2	2	2	7	6	7	**8**

Active Pre-Training: References

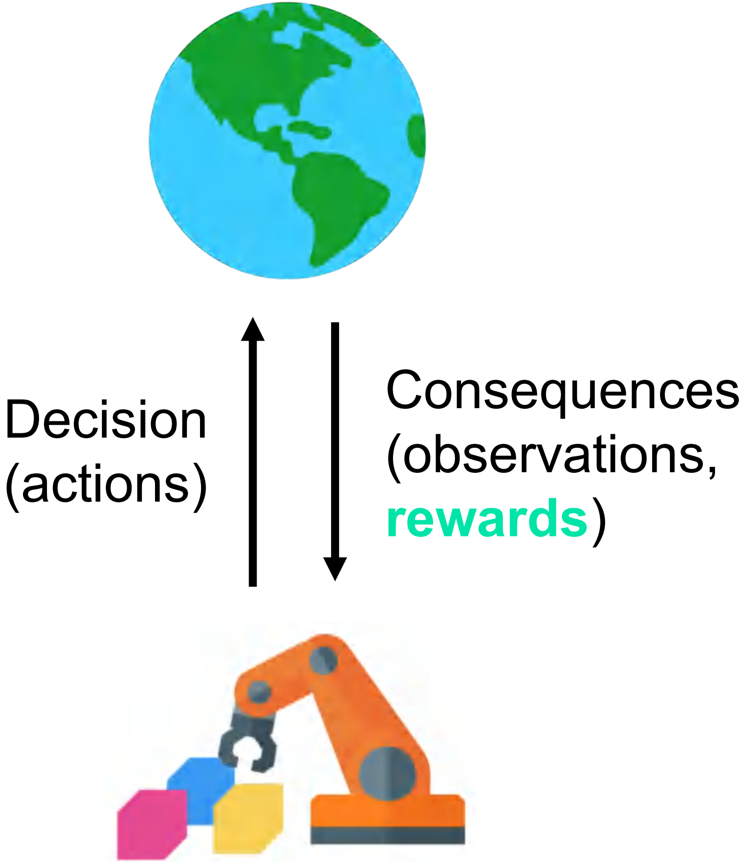
- **VIC**: Variational Intrinsic Control
  Gregor et al, 2016
- **DIAYN**: Diversity is all you need
  Eysenbach, Gupta, Ibarz, Levine, 2018
- **Valor**: Variational Option Discovery Algorithms
  Achiam, Edwards, Amodei, Abbeel, 2018
- **VISR**: Fast Task Inference with Variational Intrinsic Successor Features
  Hansen et al, 2020
- **MEPOL**: Task-Agnostic Exploration via Policy Gradient of a Non-Parametric State Entropy Estimate
  Mirco Mutti, Lorenzo Pratissoli, Marcello Restelli, 2020
- **EDL**: Explore, Discover and Learn
  Campos et al, 2020
- **APT**: Behavior From the Void: Unsupervised Active Pre-Training
  Hao Liu & Pieter Abbeel, 2020
- **CPT**: Coverage as a Principle for Discovering Transferable Behavior in Reinforcement Learning
  Campos et al, 2021
- **ProtoRL:** Reinforcement Learning with Prototypical Representations
  Yarats, Fergus, Lazarus, Pinto, 2021
- **RE3**: State Entropy Maximization with Random Encoders for Efficient Exploration
  Seo*, Chen*, Shin, Lee, Abbeel, Lee, 2021
- **APS**: See the Future through the Void: Active Pre-Training with Successor Features
  Hao Liu & Pieter Abbeel, 2021
- **ASP**: Intrinsic Motivation and Automatic Curricula via Asymmetric Self-Play
  Sukhbaatar, Lin, Kostrikov, Synnaeve, Szlam, Fergus, 2017
- Asymmetric Self-Play for Automatic Goal Discovery in Robotic Manipulation
  OpenAI, 2021

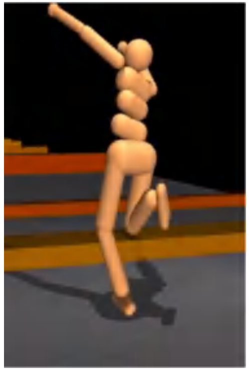
Also related: Exploration Bonuses, Curiosity, Surprise, VIME, Planning2Expore, GoExplore

An Attempt at a Complete Picture



Challenge: Designing Suitable Reward



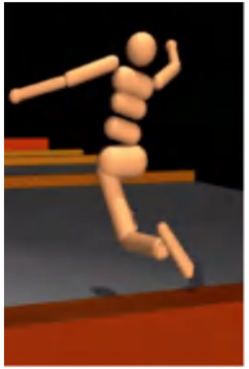


$$\hat{x}_t - 0.1||a_t||_2^2 - 3.0 \times (z_t - 1.3)^2$$

Challenge: Designing Suitable Reward

Decision (actions)

Consequences (observations, **rewards**)



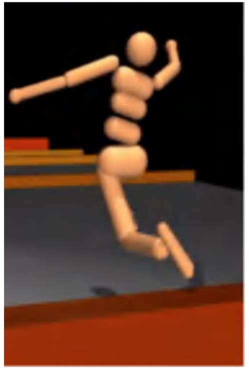
$$\hat{x}_t - 0.1\|a_t\|_2^2 - 3.0 \times (z_t - 1.3)^2$$

Hard tasks to define a reward (e.g. cooking)

Challenge: Designing Suitable Reward

Decision (actions)

Consequences (observations, **rewards**)



$$\hat{x}_t - 0.1||a_t||_2^2 - 3.0 \times (z_t - 1.3)^2$$

Hard tasks to define a reward (e.g. cooking)

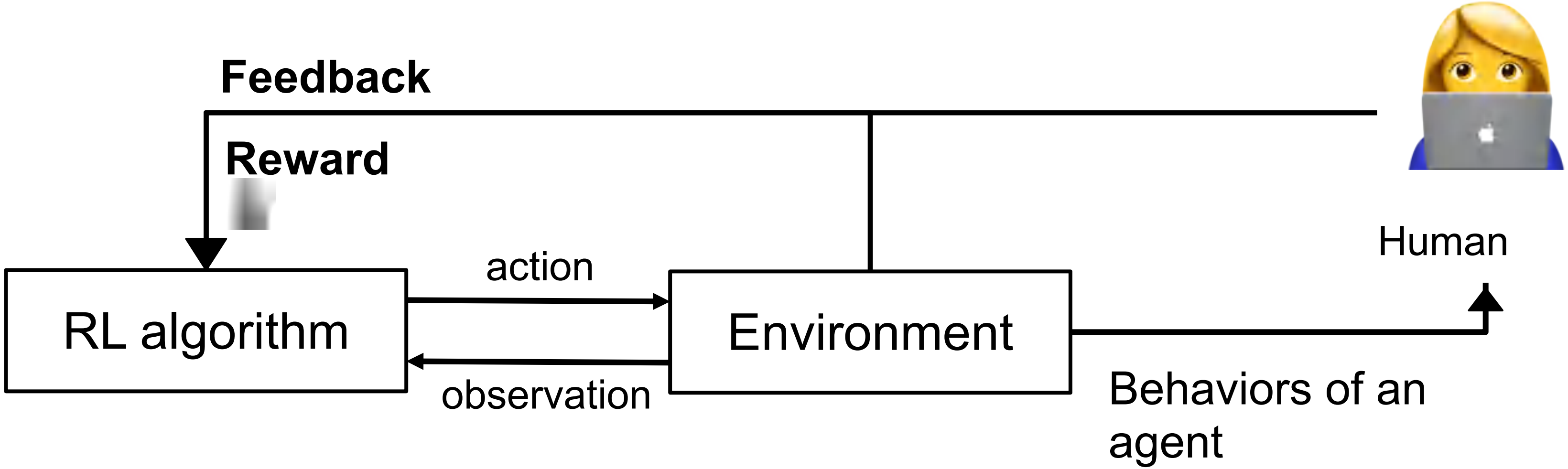
Reward exploitation

https://openai.com/blog/faulty-reward-functions/

What is an Alternative Solution?

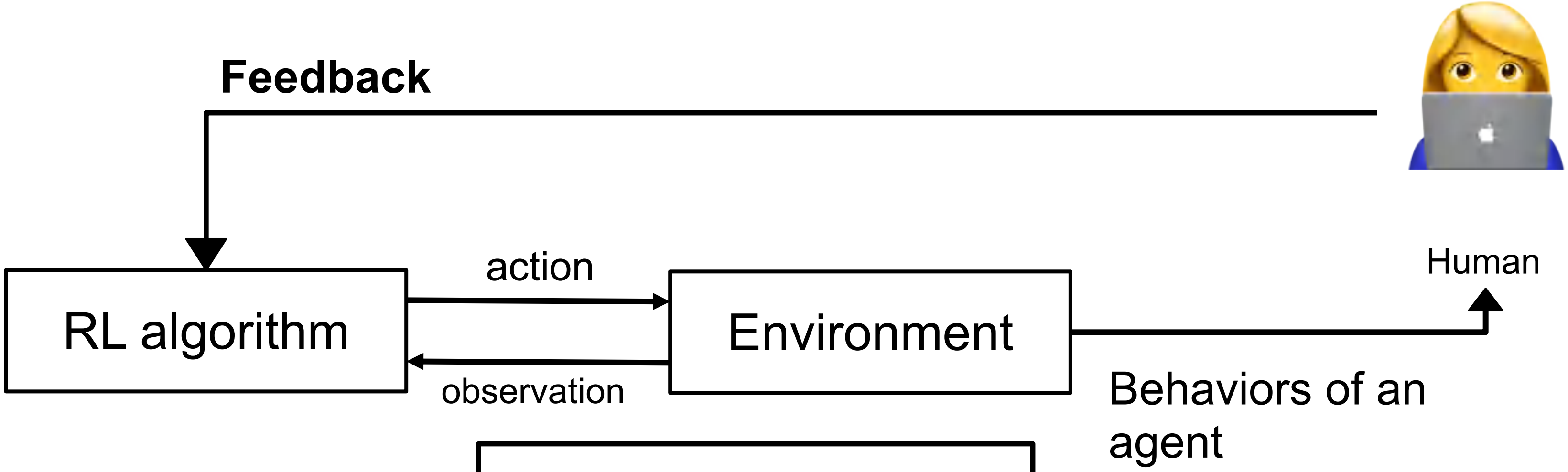
What is an Alternative Solution?

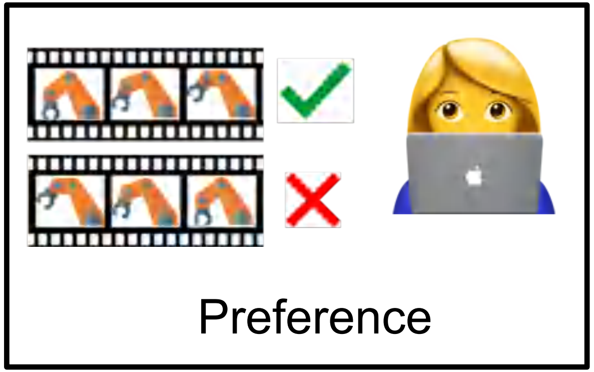
- Putting (non-expert) humans into the agent learning loop!



What is an Alternative Solution?

- Putting (non-expert) humans into the agent learning loop!





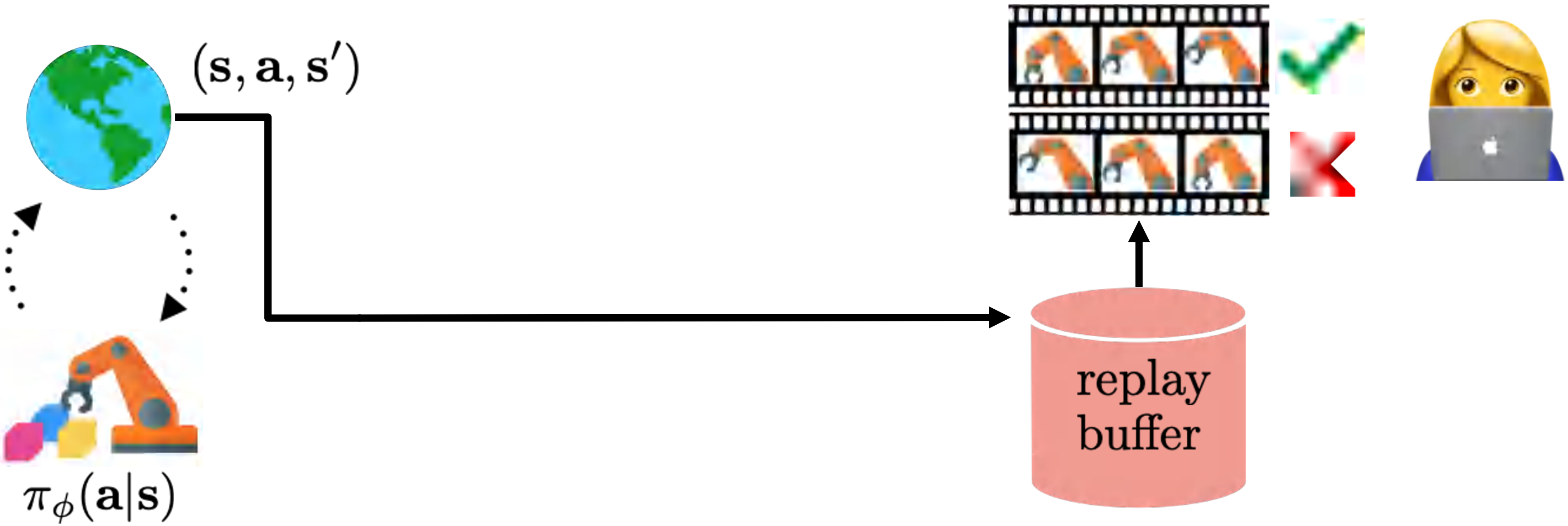
Ibarz, B., Leike, J., Pohlen, T., Irving, G., Legg, S. and Amodei, D., Reward learning from human preferences and demonstrations in atari. In NeurIPS, 2018.
Christiano, P., Leike, J., Brown, T.B., Martic, M., Legg, S. and Amodei, D., Deep reinforcement learning from human preferences. NeurIPS, 2017.

Overall Framework

- Step 1. Collect samples via interactions with environment

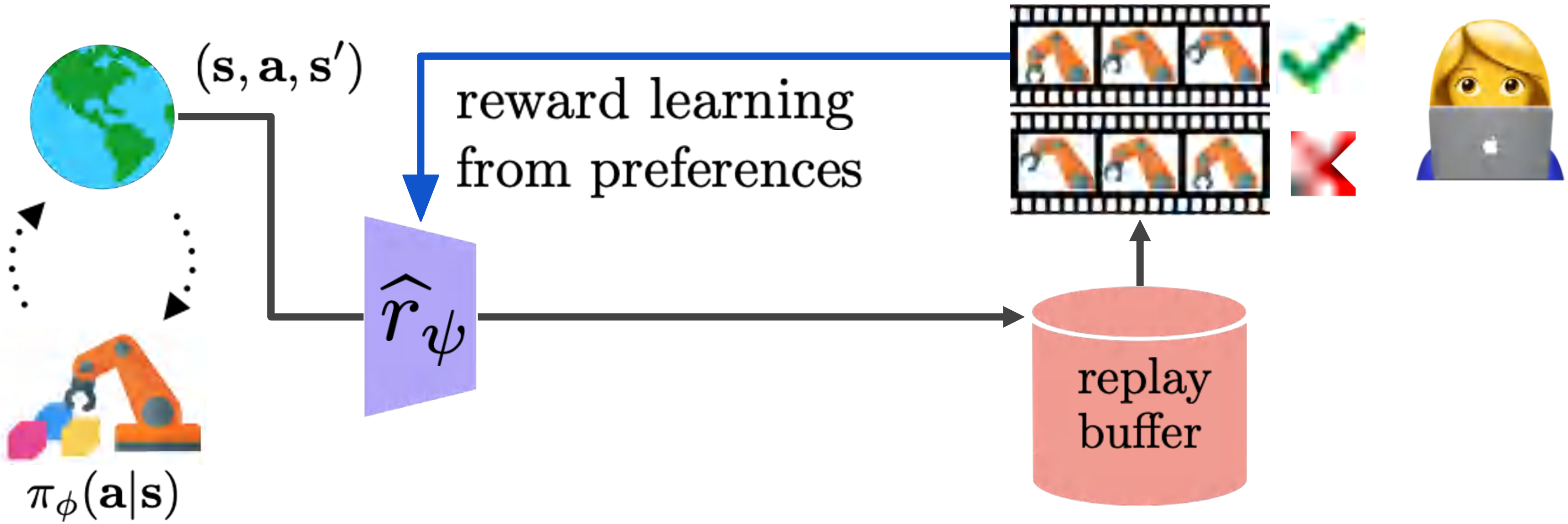
Overall Framework

- Step 1. Collect samples via interactions with environment
- Step 2. Collect human preferences



Overall Framework

- Step 1. Collect samples via interactions with environment
- Step 2. Collect human preferences
- Step 3. Optimize a reward model using cross entropy loss



Learning Reward from Preferences

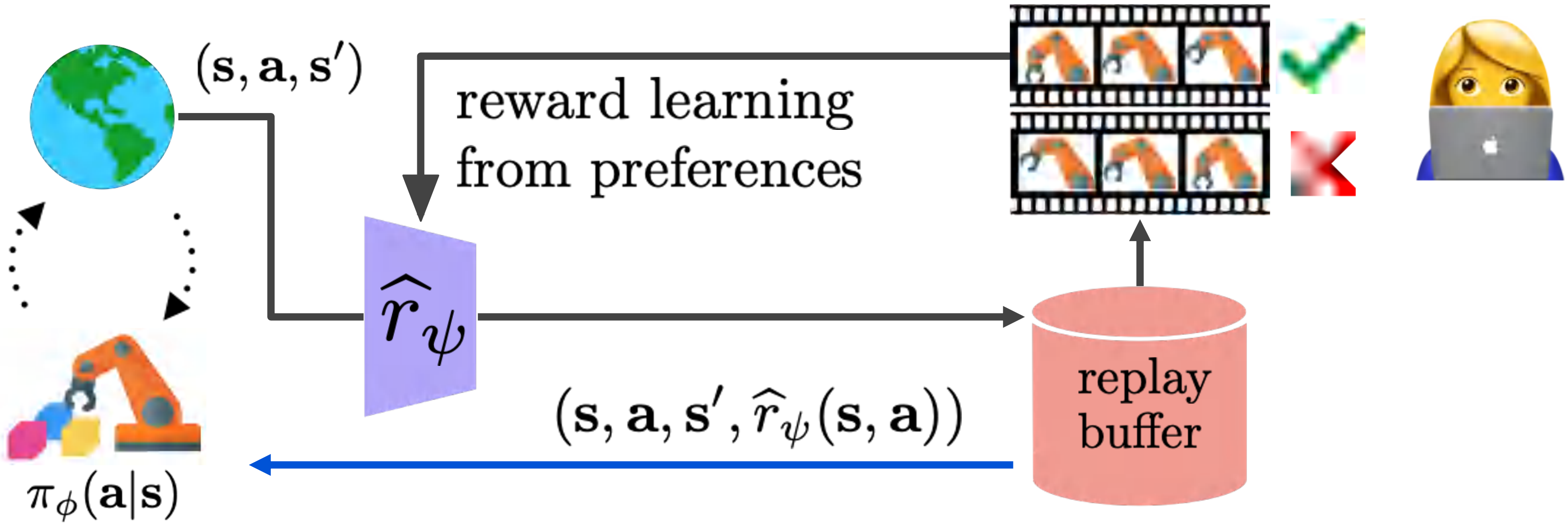
- Fitting a reward model [1]
  - Main idea: formulate this problem as a binary classification!
  - By following the Bradley-Terry model [2], we can model a **preference predictor** as follows:

$$P_\psi[\sigma^1 \succ \sigma^0] = \frac{\exp \sum_t \widehat{r}(\mathbf{s}_t^1, \mathbf{a}_t^1)}{\sum_{i \in \{0,1\}} \exp \sum_t \widehat{r}(\mathbf{s}_t^i, \mathbf{a}_t^i)}$$

[1] Christiano, P., Leike, J., Brown, T.B., Martic, M., Legg, S. and Amodei, D., Deep reinforcement learning from human preferences. NeurIPS, 2017.
[2] Bradley, R.A. and Terry, M.E., Rank analysis of incomplete block designs: I. The method of paired comparisons. Biometrika, 39(3/4), pp.324-345, 1952.

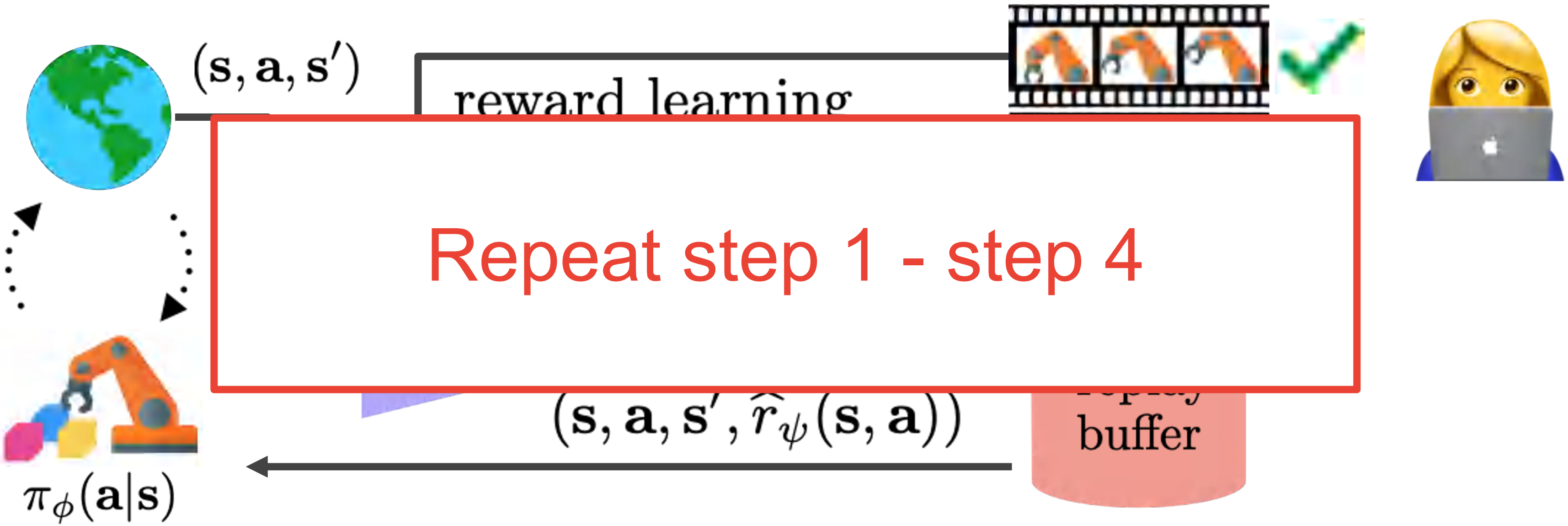
Overall Framework

- Step 1. Collect samples via interactions with environment
- Step 2. Collect human preferences
- Step 3. Optimize a reward model using cross entropy loss
- Step 4. Optimize a policy using off-policy algorithms



Overall Framework

- Step 1. Collect samples via interactions with environment
- Step 2. Collect human preferences
- Step 3. Optimize a reward model using cross entropy loss
- Step 4. Optimize a policy using off-policy algorithms



Unsupervised Pre-training: APT

- Obtaining a good initial state space coverage is important!
  - Human can’t convey much meaningful information to the agent

Unsupervised Pre-training: APT

- Obtaining a good initial state space coverage is important!
  - Human can’t convey much meaningful information to the agent

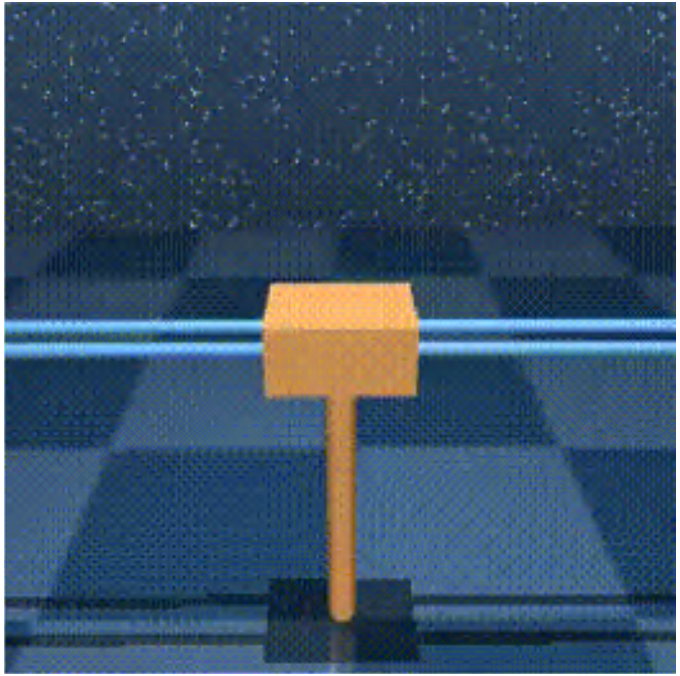
Behavior from random exploration policy

Behavior from pre-trained policy

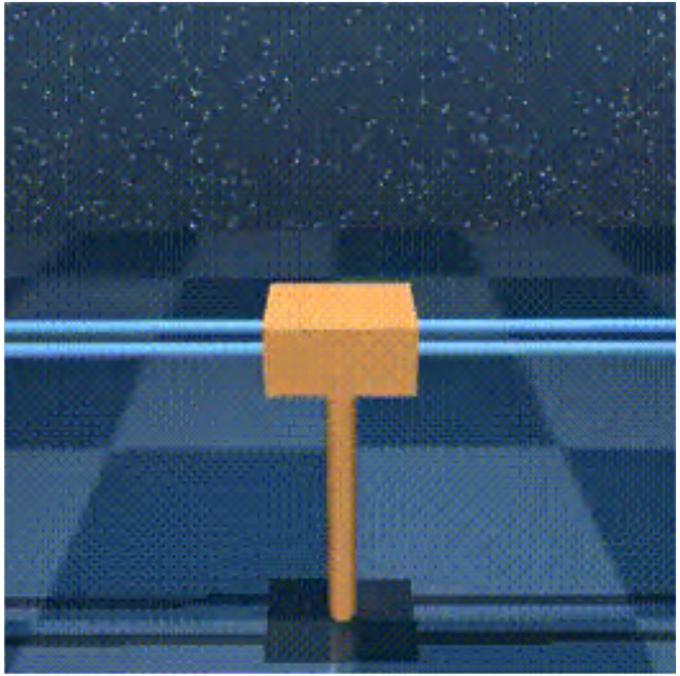
Can Human Teach Novel Behaviors?

Can Human Teach Novel Behaviors?

- 40 queries in less than 5 mins



Counter clockwise



Clockwise

Can Human Teach Novel Behaviors?

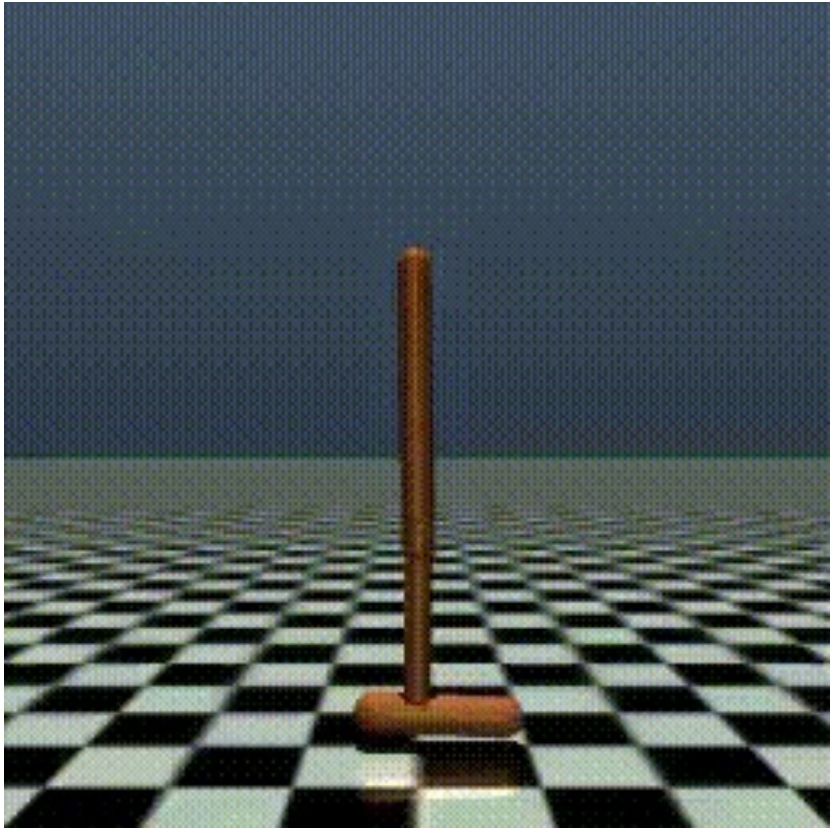
- 200 queries in less than 30 mins

Waving left front leg

Waving right front leg

Can Human Teach Novel Behaviors?

- 50 queries



Can We Avoid Reward Exploitation?

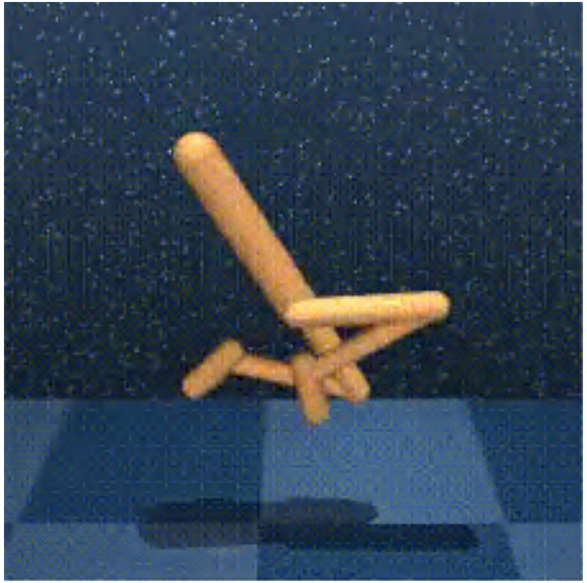
Can We Avoid Reward Exploitation?

SAC with task reward on walker, walk
(use one leg even if score ~=1000)

Can We Avoid Reward Exploitation?

- 150 queries in less than 20 mins

SAC with task reward on walker, walk
(use one leg even if score ~=1000)



SAC trained with human feedback
(use both legs)

Benchmarking

- We generate preferences using a scripted teacher [1, 2]:

$$\sigma^0 = \{(s_t^0, a_t^0), \cdots, (s_{t+H}^0, a_{t+H}^0)\}$$

$$\sigma^1 = \{(s_t^1, a_t^1), \cdots, (s_{t+H}^1, a_{t+H}^1)\}$$

$$y = 1 \quad \text{if} \sum_t r^*(\mathbf{s}_t^1, \mathbf{a}_t^1) > \sum_t r^*(\mathbf{s}_t^0, \mathbf{a}_t^0)$$

True task rewar

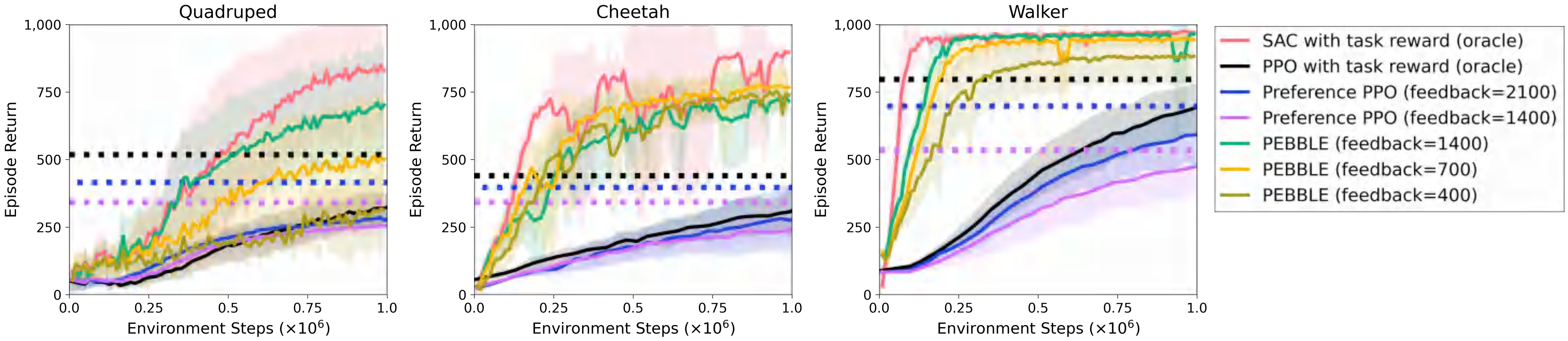
Preferences are immediately generated
→ more rapid experiments

We can evaluate the agent quantitatively by measuring the true average return

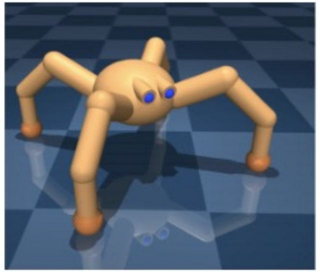
[1] Ibarz, B., Leike, J., Pohlen, T., Irving, G., Legg, S. and Amodei, D., Reward learning from human preferences and demonstrations in atari. In NeurIPS, 2018.
[2] Christiano, P., Leike, J., Brown, T.B., Martic, M., Legg, S. and Amodei, D., Deep reinforcement learning from human preferences. NeurIPS, 2017.

Comparison: Locomotion Tasks

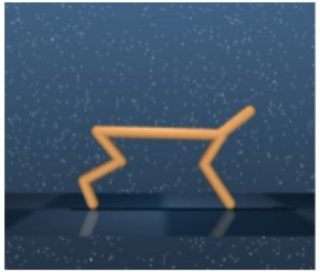
- Learning curves (10 random seeds)



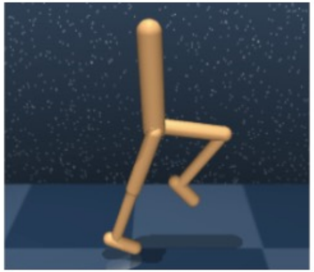
* Asymptotic performance of PPO and Preference PPO is indicated by dotted lines of the corresponding color



Quadruped



Cheetah



Walker

References

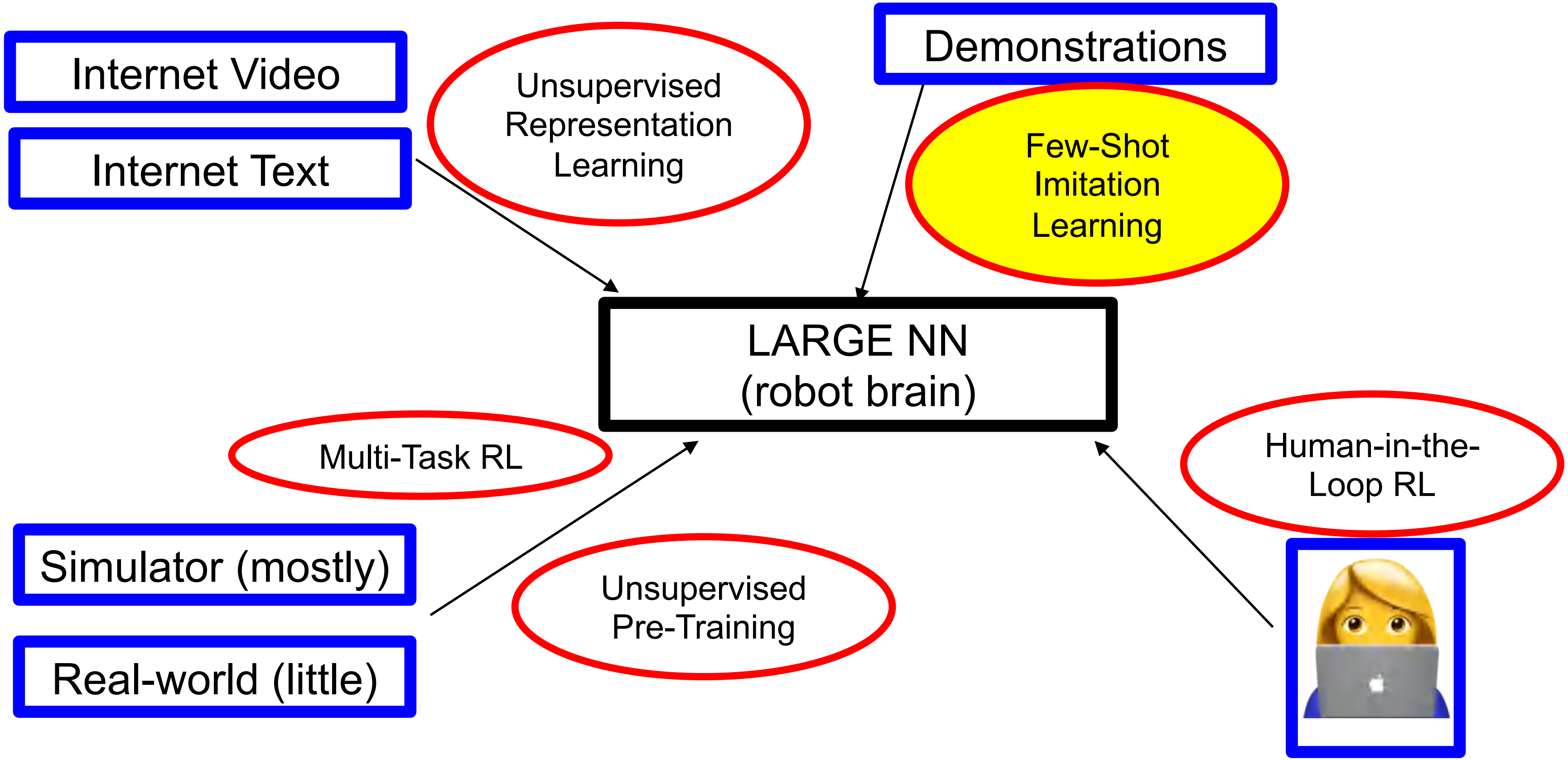
- **Preference-based RL**

  -- **PEBBLE: Feedback-Efficient Interactive Reinforcement Learning via Relabeling Experience and Unsupervised Pre-training**
  Kimin Lee*, Laura Smith*, Pieter Abbeel, 2021
  -- **DRL from Human Preferences**
  Paul Christano, J Leike, T Brown, M Martic, S Legg, D Amodei, 2017
  -- **Reward Learning from Human Preferences and Demonstrations**
  Ibarz, Leike, Pohlen, Irving, Legg, Amodei 2018

- **Binary-feedback RL**

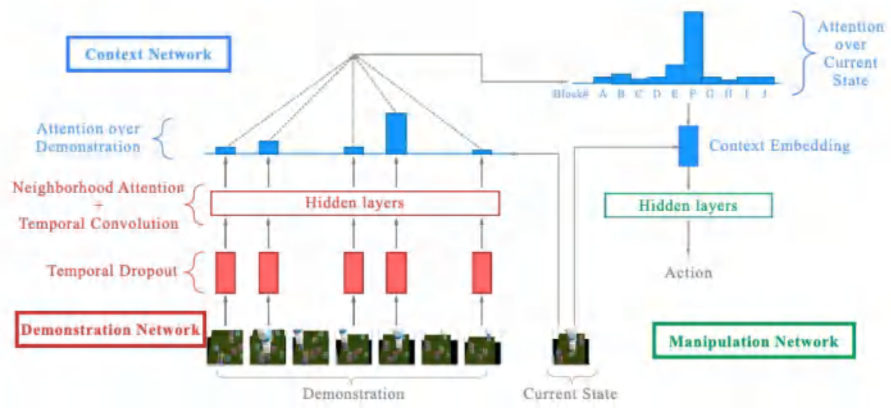
  -- **COACH: Interactive Learning from Policy-Dependent Human Feedback**
  MacGlashan, Ho, Loftin, Peng, Wang, Roberts, Taylor, Littman, 2017
  -- **Deep Coach: Deep Reinforcement Learning from Policy-Dependent Human Feedback**
  Arumugam, Lee, Saskin, Littman, 2019
  -- **TAMER: Interactively Shaping Agents via Human Reinforcement: The TAMER Framework**
  Knox, Stone, 2009
  -- **Deep TAMER: Interactive Agent Shaping in High-Dimensional State Spaces**
  Warnell, Waytowich, Lawhern, Stone, 2018

An Attempt at a Complete Picture



Few Shot Imitation Learning

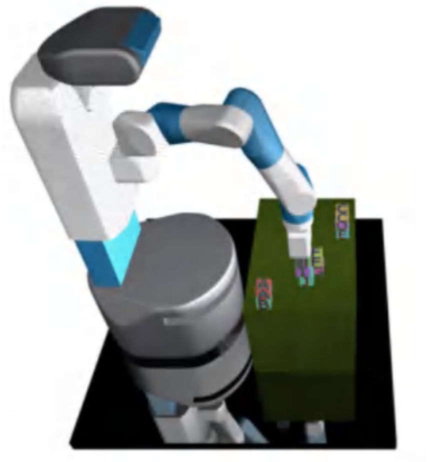
- One-Shot Imitation Learning, Duan et al, NeurIPS 2017



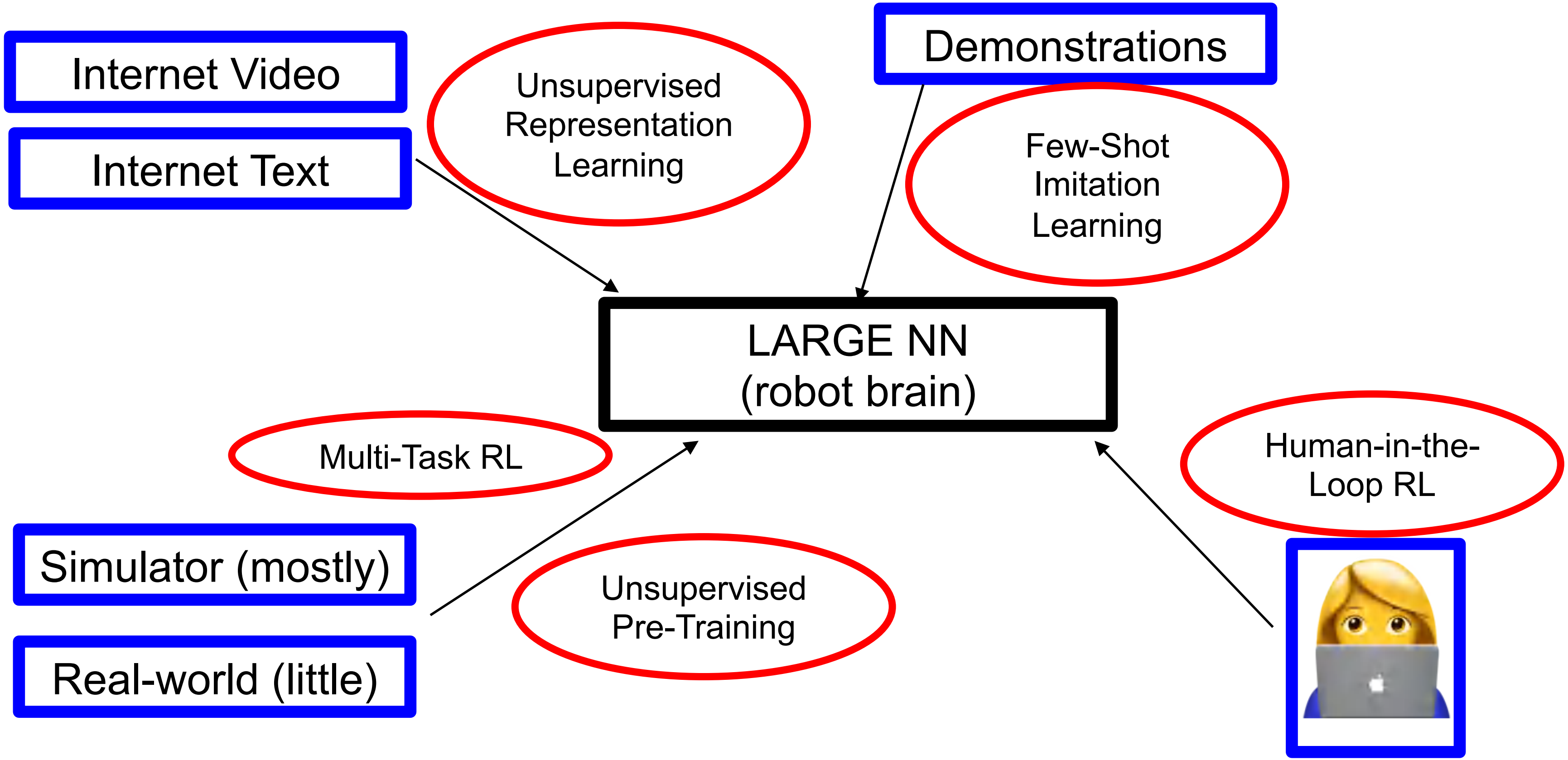
- One-Shot Visual Imitation Learning via Meta-Learning, Finn et al, CoRL 2017
- Transformers for One-Shot Visual Imitation, Dasari & Gupta, CorL 2020

Few Shot Imitation Learning -- Limitation

- Single task family
  - E.g.



Summary (so far)



Language Models (e.g. OpenAI’s GPT-2)

<u>Prompt:</u> A train carriage containing controlled nuclear materials was stolen in Cincinnati today. Its whereabouts are unknown.

<u>GPT-2:</u> *The incident occurred on the downtown train line, which runs from Covington and Ashland stations. In an email to Ohio news outlets, the U.S. Department of Energy said it is working with the Federal Railroad Administration to find the thief.*
‘The theft of this nuclear material will have significant negative consequences on public and environmental health, our workforce and the economy of our nation,’ said Tom Hicks, the U.S. Energy Secretary, in a statement. ‘Our top priority is to secure the theft and ensure it doesn’t happen again.’
The stolen material was taken from the University of Cincinnati’s Research Triangle Park nuclear research site, according to a news release from Department officials.
The Nuclear Regulatory Commission did not immediately release any information.
According to the release, the U.S. Department of Energy’s Office of Nuclear Material Safety and Security is leading that team’s investigation.
‘The safety of people, the environment and the nation’s nuclear stockpile is our highest priority,’ Hicks said. ‘We will get to the bottom of this and make no excuses.’

Text Generati… …'s GPT-2)

SYSTEM PROMPT (HUMAN-WRITTEN)

In a shocking fi… remote, previou… surprising to the… English.

MODEL COMPLETION (MACHINE-WRITTEN, 10 TRIES)

The scientist nam… Ovid's Unicorn. Th… previously unknown…

Now, after almost tw… odd phenomenon is fi…

Dr. Jorge Pérez, an ev… La Paz, and several co… Mountains when they fou… or humans. Pérez noticed… a natural fountain, surr… snow.

Pérez and the others then … the time we reached the top… with some crystals on top,"

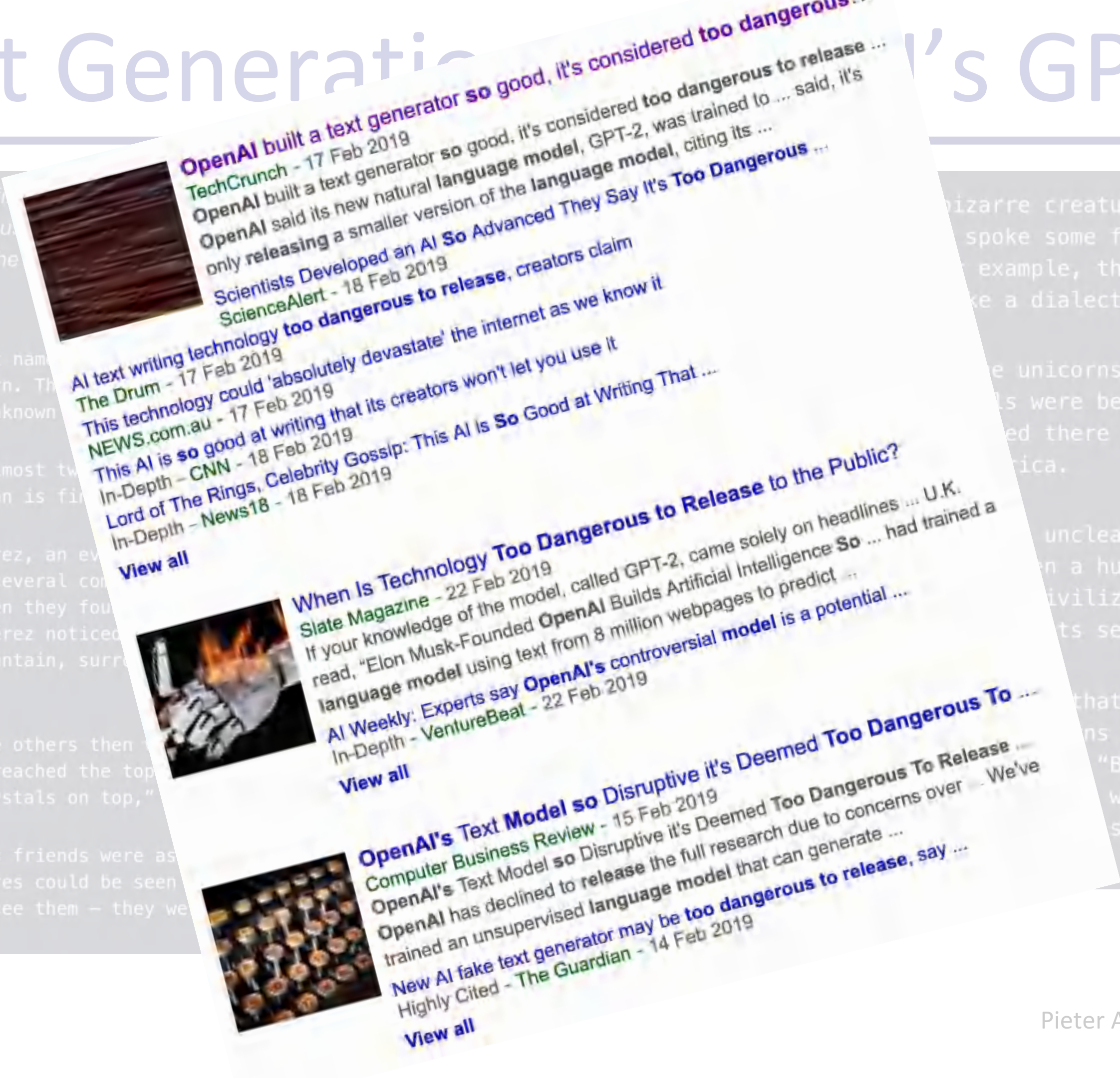
Pérez and his friends were as… These creatures could be seen… too much to see them – they we… horns.

…bizarre creatures the scientists discovered … spoke some fairly regular English. Pérez … example, that they have a common …ke a dialect or dialectic."

…e unicorns may have originated in …ls were believed to be descendants of a …ed there before the arrival of humans …ica.

…unclear, some believe that perhaps …n a human and a unicorn met each …ivilization. According to Pérez, …ts seem to be quite common."

…that it is likely that the only …ns are indeed the descendants of … "But they seem to be able to … which I believe is a sign of … social organization," said the



[Radford et al, 2019]

Benchmarks – GPT-2

DATASET	METRIC	OUR RESULT	PREVIOUS RECORD	HUMAN
Winograd Schema Challenge	accuracy (+)	**70.70%**	63.7%	92%+
LAMBADA	accuracy (+)	**63.24%**	59.23%	95%+
LAMBADA	perplexity (-)	**8.6**	99	~1-2
Children's Book Test Common Nouns (validation accuracy)	accuracy (+)	**93.30%**	85.7%	96%
Children's Book Test Named Entities (validation accuracy)	accuracy (+)	**89.05%**	82.3%	92%
Penn Tree Bank	perplexity (-)	**35.76**	46.54	unknown
WikiText-2	perplexity (-)	**18.34**	39.14	unknown

Benchmarks -- BERT

GLUE Results

System	MNLI-(m/mm) 392k	QQP 363k	QNLI 108k	SST-2 67k	CoLA 8.5k	STS-B 5.7k	MRPC 3.5k	RTE 2.5k	**Average** -
Pre-OpenAI SOTA	80.6/80.1	66.1	82.3	93.2	35.0	81.0	86.0	61.7	74.0
BiLSTM+ELMo+Attn	76.4/76.1	64.8	79.9	90.4	36.0	73.3	84.9	56.8	71.0
OpenAI GPT	82.1/81.4	70.3	88.1	91.3	45.4	80.0	82.3	56.0	75.2
$\text{BERT}_{\text{BASE}}$	84.6/83.4	71.2	90.1	93.5	52.1	85.8	88.9	66.4	79.6
$\text{BERT}_{\text{LARGE}}$	**86.7/85.9**	**72.1**	**91.1**	**94.9**	**60.5**	**86.5**	**89.3**	**70.1**	**81.9**

Might these pre-trained transformers be ***even*** more general?

Pretrained Transformers As Universal Computation Engines

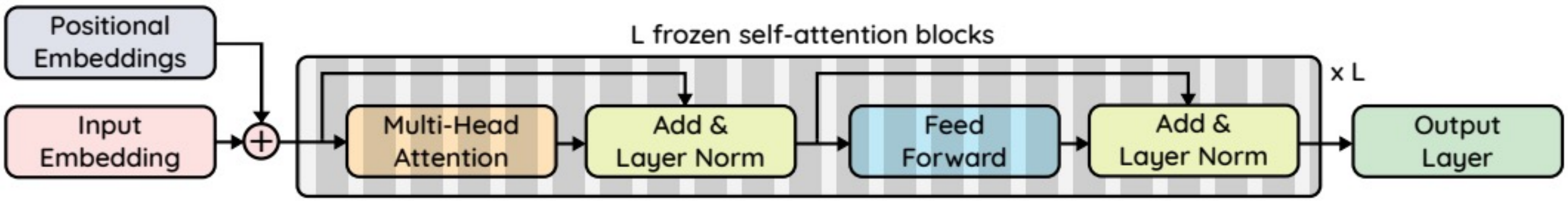
Kevin Lu
UC Berkeley
kzl@berkeley.edu

Aditya Grover
Facebook AI Research
adityagrover@fb.com

Pieter Abbeel
UC Berkeley
pabbeel@cs.berkeley.edu

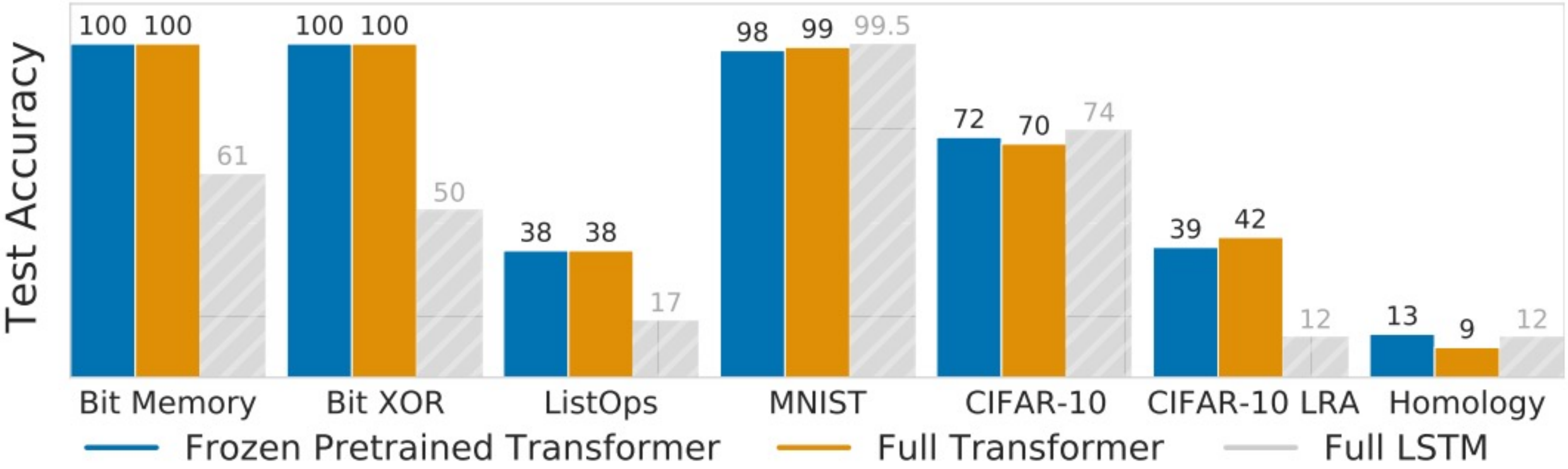
Igor Mordatch
Google Brain
imordatch@google.com

Pre-Trained Model + .1% finetune



- ***Pre-train:*** language corpus next-token prediction
- ***Minimally fine-tune:***
  - Bit memory
  - Bit XOR
  - ListOps
  - MNIST
  - CIFAR-10 and CIFAR-10 LRA
  - Remote homology detection

Can pretrained LMs transfer to new modalities?

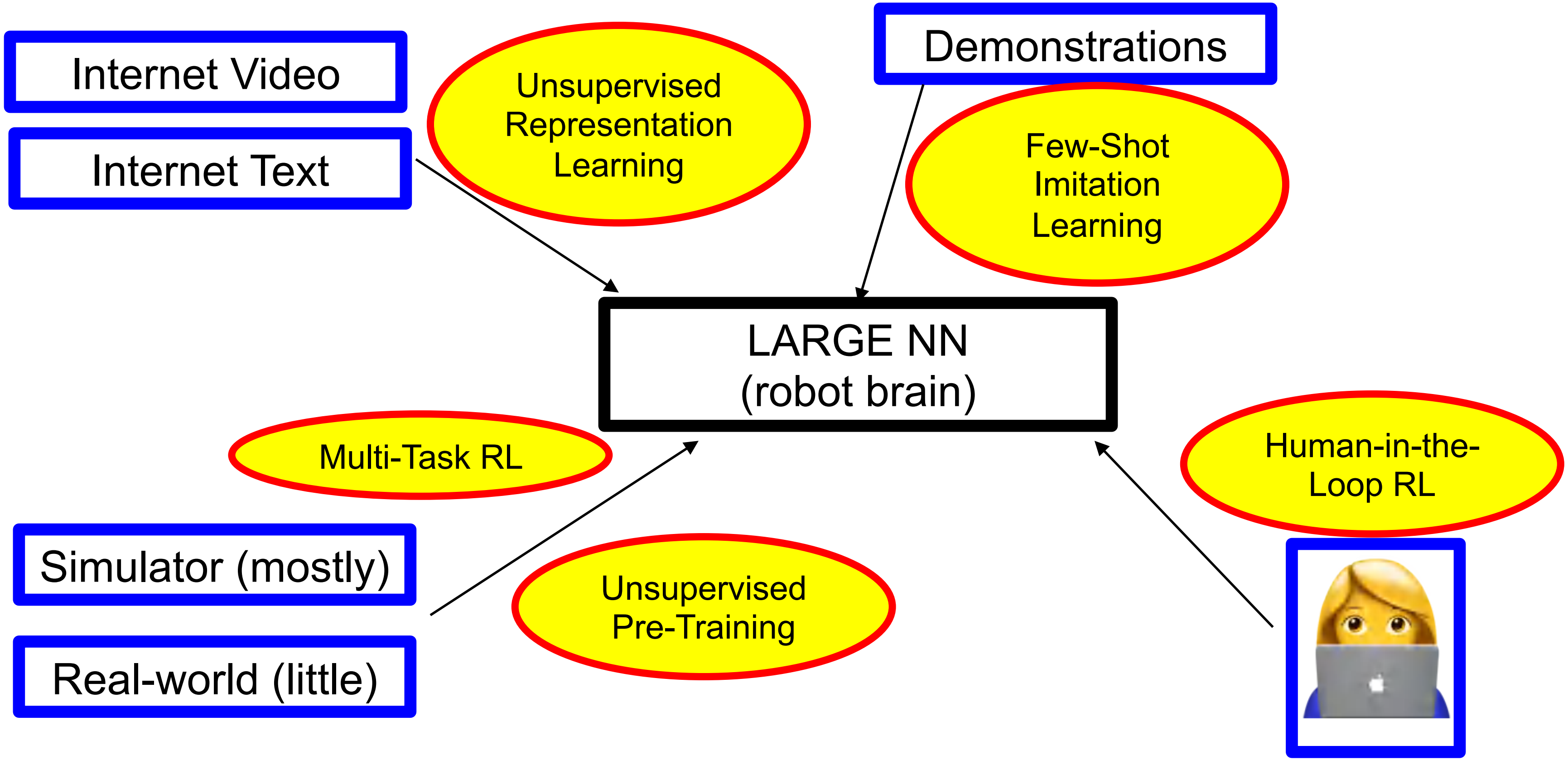


What's the importance of the pretraining modality?

Model	Bit Memory	XOR	ListOps	MNIST	C10	C10 LRA	Homology
FPT	100%	100%	38.4%	98.0%	68.2%	38.6%	12.7%
Random	75.8%	100%	34.3%	91.7%	61.7%	36.1%	9.3%
Bit	100%	100%	35.4%	97.8%	62.6%	36.7%	7.8%
ViT	100%	100%	37.4%	97.8%	72.5%	43.0%	7.5%

Table 2: Test accuracy of language-pretrained (FPT) vs randomly initialized (Random) vs Bit Memory pretraining (Bit) vs pretrained Vision Transformer (ViT) models. The transformer is frozen.

Summary


Internet Video
Internet Text
Unsupervised Representation Learning
Demonstrations
Few-Shot Imitation Learning
LARGE NN
(robot brain)
Multi-Task RL
Human-in-the-Loop RL
Simulator (mostly)
Real-world (little)
Unsupervised Pre-Training

Thank you!

pabbeel@cs.berkeley.edu