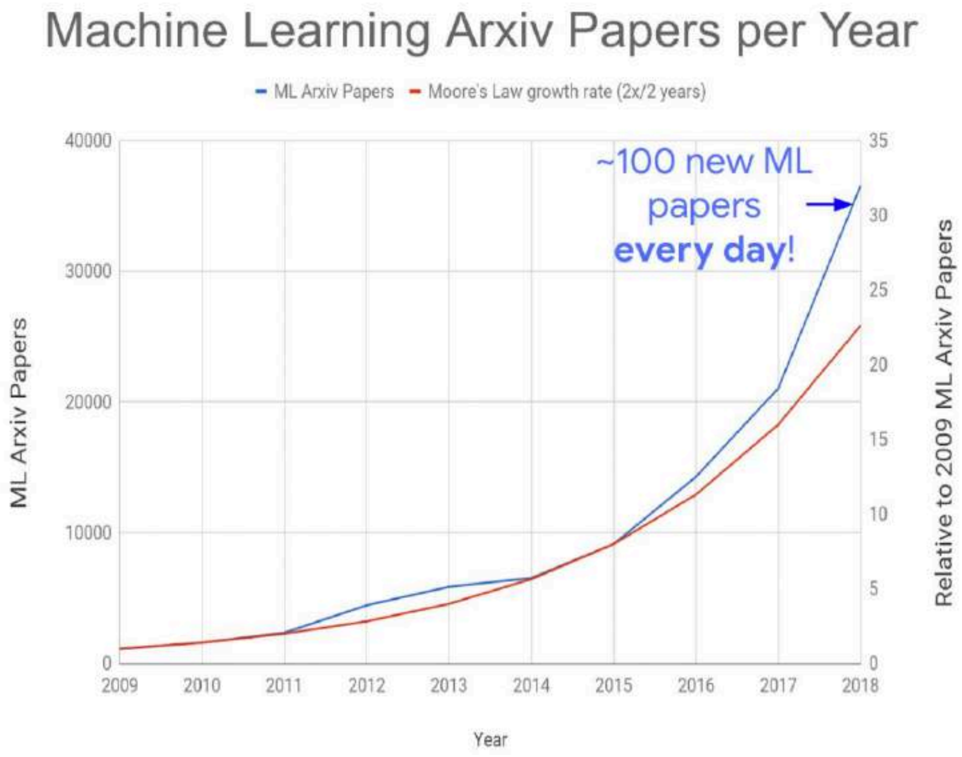
Full Stack Deep Learning

Research Directions

Pieter Abbeel, Sergey Karayev, Josh Tobin

Number of arxiv papers submitted in AI categories



[source: http://data-mining.philippe-fournier-viger.com/too-many-machine-learning-papers]

Many Exciting Directions in AI

- Few-Shot Learning
- Reinforcement Learning
- Imitation Learning
- Domain Randomization
- Architecture Search
- Unsupervised Learning
- Lifelong Learning
- Bias in ML (avoiding)
- Long Horizon Reasoning
- Safe Learning
- Value Alignment
- Planning + Learning
- …

Many Exciting Directions in AI

- Few-Shot Learning
- Reinforcement Learning
- Imitation Learning
- Domain Randomization
- Architecture Search
- Unsupervised Learning

- Lifelong Learning
- Bias in ML (avoiding!)
- Long Horizon Reasoning
- Safe Learning
- Value Alignment
- Planning + Learning
- …

Outline

- Sampling of research directions
- Overall research theme
- Research <> Real-World gap
- How to keep up

Many Exciting Directions in AI

- ***Few-Shot Learning***
- Reinforcement Learning
- Imitation Learning
- Domain Randomization
- Architecture Search
- Unsupervised Learning

- Lifelong Learning
- Bias in ML (avoiding)
- Long Horizon Reasoning
- Safe Learning
- Value Alignment
- Planning + Learning
- …

Supervised Learning

- Huge successes

- **But requirement: lots of labeled data**

In Contrast: Humans Need Only One Example

hoverboard

onewheelplus

What’s this one?

What’s this one?

What's this one?

What's this one?

Why Can We Do This?

- We have a strong prior notion of what object categories are like

How to equip a machine with such prior?


IMAGENET

Model-Agnostic Meta-Learning (MAML)

- Starting observation:
  - Computer vision practice:
    - Train on ImageNet [Deng et al. '09]
    - Fine-tune on actual task

    works really well! [Decaf: Donahue et al. '14; ...]

- Questions:
  - How to generalize this to behavior learning?
  - And can we explicitly train end-to-end for being maximally ready for efficient fine-tuning?

Model-Agnostic Meta-Learning (MAML)

<u>Key idea:</u> End-to-end learning of parameter vector θ that is good init for fine-tuning for many tasks

finetuned parameters

pretrained parameters

MAML test time: fine-tuning: $\theta' \leftarrow \theta - \alpha \nabla_\theta \mathcal{L}_{\mathrm{tr}}(\theta)$

train data for new task

MAML training: $$\min_\theta \sum_{\text{tasks } i} \mathcal{L}^{(i)}_{\mathrm{val}}(\underbrace{\theta - \alpha \nabla_\theta \mathcal{L}^{(i)}_{\mathrm{train}}(\theta)}_{\text{finetuned parameters } \theta^{(i)\prime}})$$

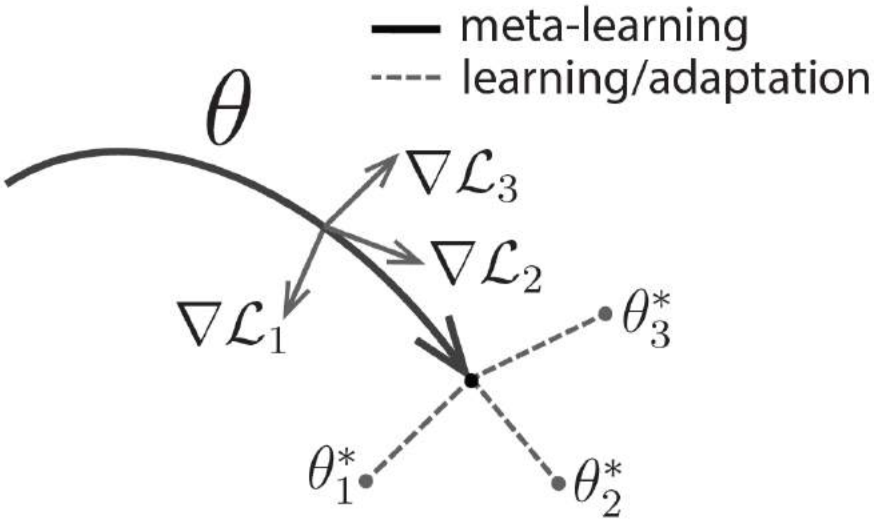
[Finn, Abbeel, Levine ICML 2017]

Model-Agnostic Meta-Learning (MAML)

$$\min_{\theta} \sum_{\text{tasks } i} \mathcal{L}^{(i)}_{\text{val}}(\theta - \alpha \nabla_\theta \mathcal{L}^{(i)}_{\text{train}}(\theta))$$

θ parameter vector being meta-learned

θ_i^* optimal parameter vector for task i



Few-Shot Learning: Classification

Omniglot dataset (Lake et al. ’15), Mini-Imagenet dataset (Vinyals et al. ‘16)

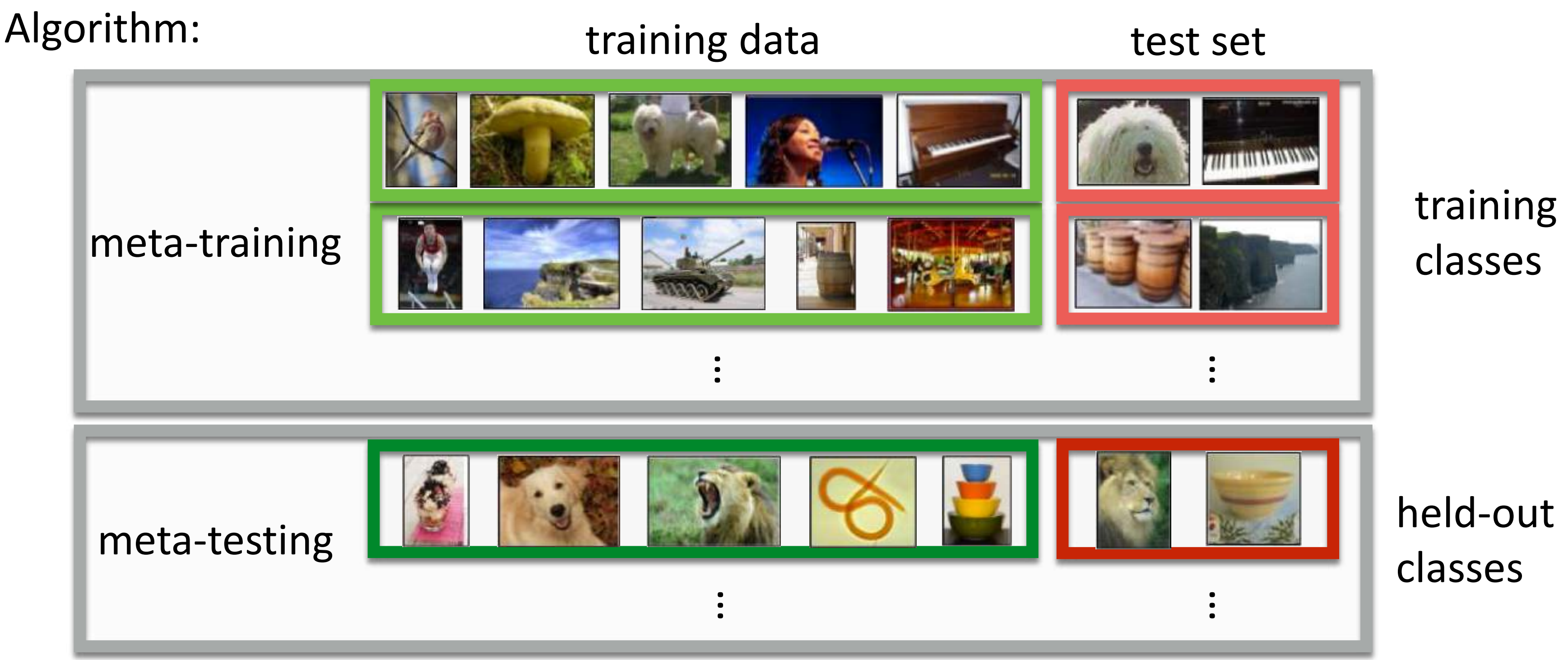
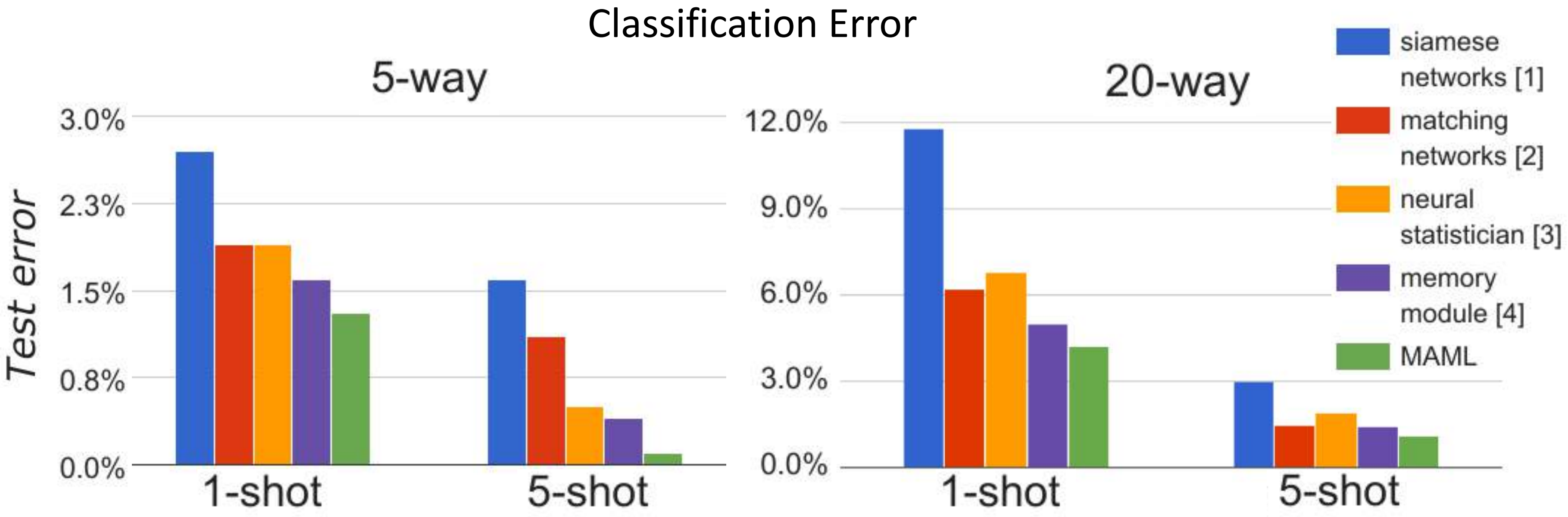


diagram adapted from Ravi & Larochelle ‘17

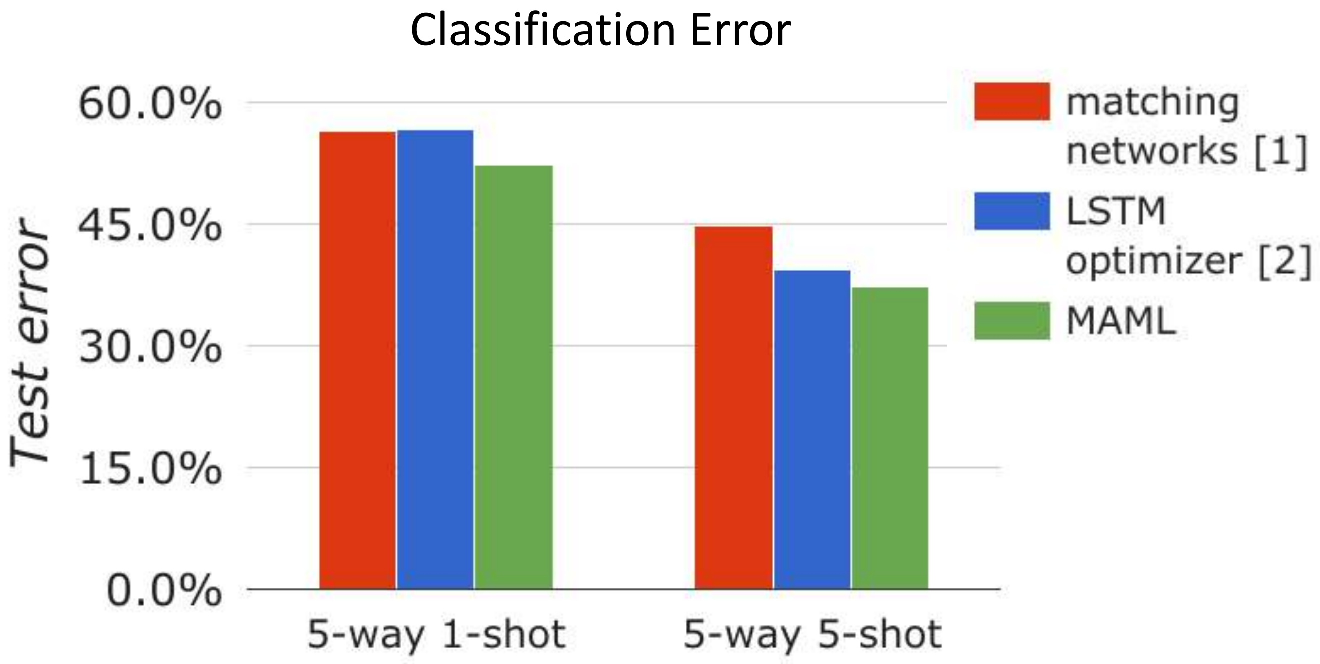
Omniglot Few-Shot Classification



Omniglot Dataset: 1200 training classes, 423 test classes

[1] Koch '15 [2] Vinyals et al. '16
[3] Edwards & Storkey '17 [4] Kaiser et al. '17

Mini-ImageNet Few-Shot Classification



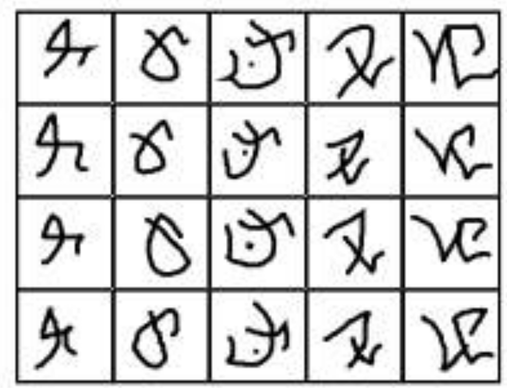
Mini-Imagenet Dataset: 64 training classes, 24 test classes

[1] Vinyals '16   [2] Ravi & Larochelle '17

Meta Learning for Classification

Task distribution: different classification datasets (input: images, output: class labels)

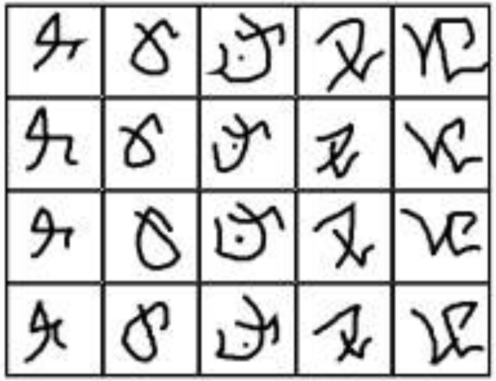
- Hochreiter et al., (2001) Learning to learn using gradient descent
- Younger et al., (2001), Meta learning with back propagation
- Koch et al., (2015) Siamese neural networks for one-shot image recognition
- Santoro et al., (2016) Meta-learning with memory-augmented neural networks
- Vinyals et al., (2016) Matching networks for one shot learning
- Edwards et al., (2016) Towards a Neural Statistician
- Ravi et al., (2017) Optimization as a model for few-shot learning
- Munkhdalai et al., (2017) Meta Networks
- Snell et al., (2017) Prototypical Networks for Few-shot Learning
- Shyam et al., (2017) Attentive Recurrent Comparators
- Finn et al., (2017) Model-Agnostic Meta-Learning for Fast Adaptation of Deep Netwo
- Mehrotra et al., (2017) Generative Adversarial Residual Pairwise Networks for One S
- Mishra et al., (2017) Meta-Learning with Temporal Convolutions
- Li et al., (2017) Meta-SGD: Learning to Learn Quickly for Few Shot Learning
- Finn and Levine, (2017) Meta-Learning and Universality: Deep Representations and Gradient Descent can Approximate any Learning Algorithm
- Grant et al., (2017) Recasting Gradient-Based Meta-Learning as Hierarchical Bayes



Meta Learning for Classification

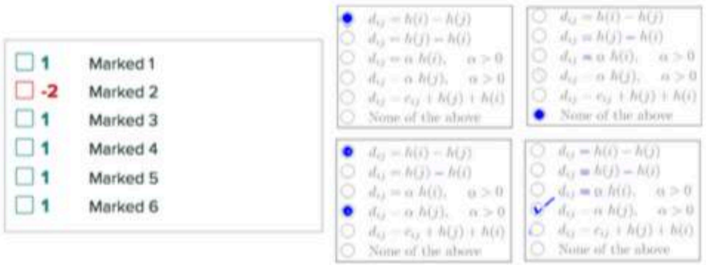
Task distribution: different classification datasets (input: images, output: class labels)

- Hochreiter et al., (2001) Learning to learn using gradient descent
- Younger et al., (2001), Meta learning with back propagation
- Koch et al., (2015) Siamese neural networks for one-shot image recognition
- Santoro et al., (2016) Meta-learning with memory-augmented neural networks
- Vinyals et al., (2016) Matching networks for one shot learning
- Edwards et al., (2016) Towards a Neural Statistician
- Ravi et al., (2017) Optimization as a model for few-shot learning
- Munkhdalai et al., (2017) Meta Networks
- Snell et al., (2017) Prototypical Networks for Few-shot Learning
- Shyam et al., (2017) Attentive Recurrent Comparators
- Finn et al., (2017) Model-Agnostic Meta-Learning for Fast Adaptation of Deep Networks
- Mehrotra et al., (2017) Generative Adversarial Residual Pairwise Networks for One Shot Learning
- Mishra et al., (2017) Meta-Learning with Temporal Convolutions
- Li et al., (2017) Meta-SGD: Learning to Learn Quickly for Few Shot Learning
- Finn and Levine, (2017) Meta-Learning and Universality: Deep Representations and Gradient Descent can A
- Grant et al., (2017) Recasting Gradient-Based Meta-Learning as Hierarchical Bayes
- Raghu et al, (2019) Rapid learning or feature reuse? towards understanding the effectiveness of maml
- Lake et al, (2019) The Omniglot Challenge: a 3-year progress report
- Rajeswaran et al (2019) Meta-learning with implicit gradients
- Wang et al (2019) SimpleShot: Revisiting Nearest-Neighbor Classification for Few-Shot Learning

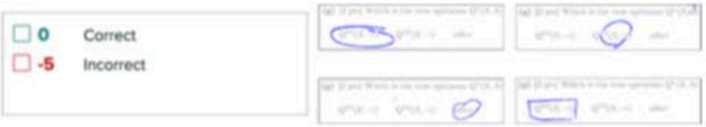


gradescope

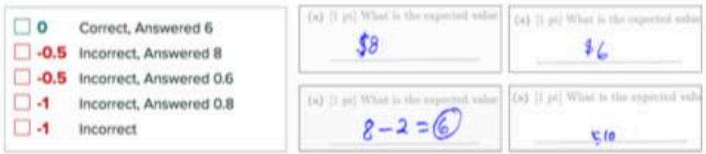
Few-Shot Learning for Grading



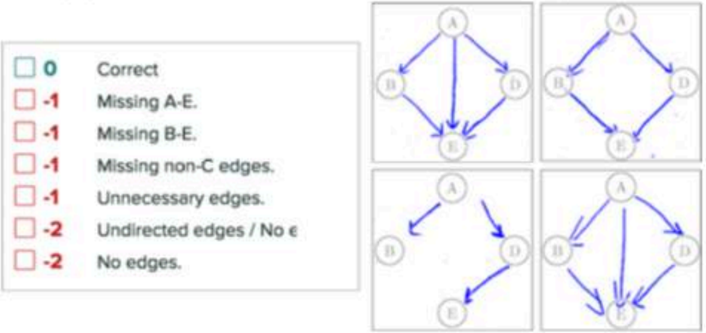
(a) Defined Multiple Choice (MC)



(b) Free-form Multiple Choice (MC)



(c) Fill-in-the-blank



(d) Drawing

[Tang*, Karayev* et al, 2018 (in prep.)]

Meta Learning for Optimization

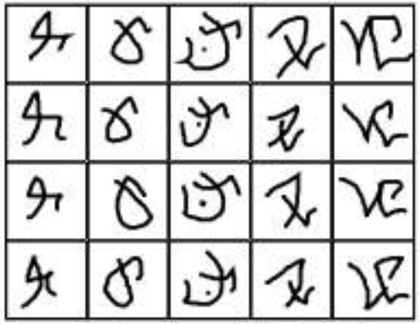
Task distribution: different neural networks, weight initializations, and/or different loss functions

- Bengio et al., (1990) Learning a synaptic learning rule
- Naik et al., (1992) Meta-neural networks that learn by learning
- Hochreiter et al., (2001) Learning to learn using gradient descent
- Younger et al., (2001), Meta learning with back propagation
- Andrychowicz et al., (2016) Learning to learn by gradient descent by gradient descent
- Chen et al., (2016) Learning to Learn for Global Optimization of Black Box Functions
- Wichrowska et al., (2017) Learned Optimizers that Scale and Generalize
- Ke et al., (2017) Learning to Optimize Neural Nets
- Wu et al., (2017) Understanding Short-Horizon Bias in Stochastic Meta-Optimization

Meta Learning for Generative Models

Task distribution: different unsupervised datasets (e.g. collection of images)

- Rezende et al., (2016) One-Shot Generalization in Deep Generative Models
- Edwards et al., (2016) Towards a Neural Statistician
- Bartunov et al., (2016) Fast Adaptation in Generative Models with Generative Matching Networks
- Bornschein et al., (2017) Variational Memory Addressing in Generative Models
- Reed et al., (2017) Few-shot Autoregressive Density Estimation: Towards Learning to Learn Distributions

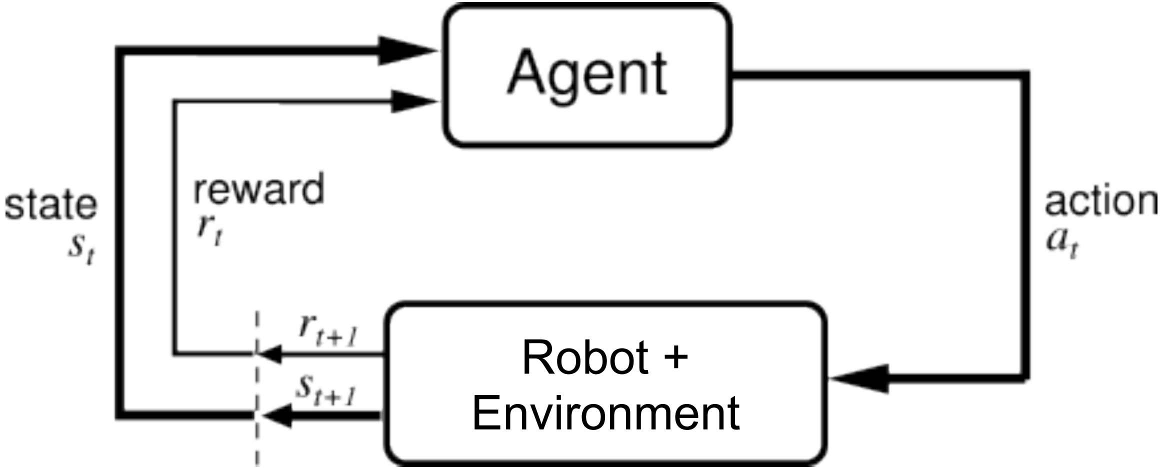


Many Exciting Directions in AI

- Few-Shot Learning
- ***Reinforcement Learning***
- Imitation Learning
- Domain Randomization
- Architecture Search
- Unsupervised Learning

- Lifelong Learning
- Bias in ML (avoiding)
- Long Horizon Reasoning
- Safe Learning
- Value Alignment
- Planning + Learning
- …

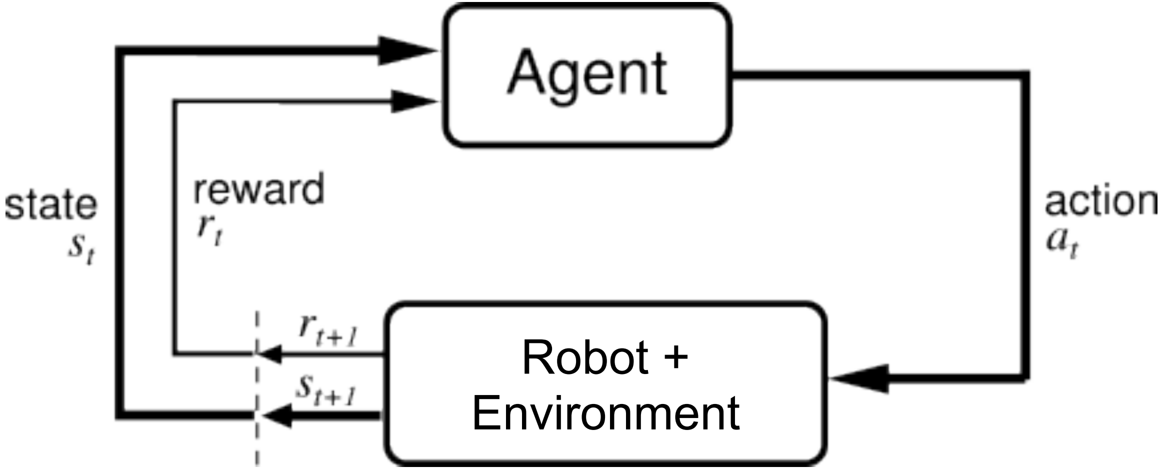
Reinforcement Learning (RL)



$$\max_{\theta} \quad \mathrm{E}[\sum_{t=0}^{H} R(s_t)|\pi_\theta]$$

[Image credit: Sutton and Barto 1998]

Reinforcement Learning (RL)

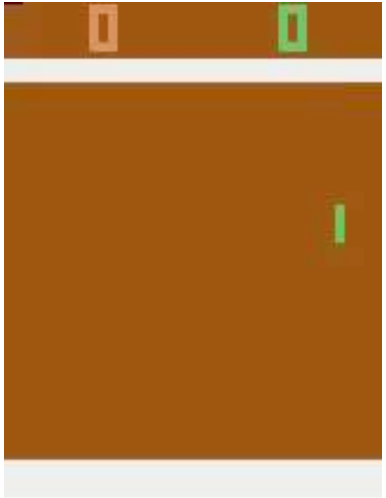


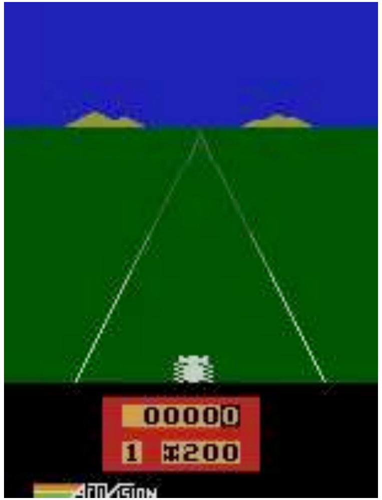
$$\max_{\theta} \quad \mathrm{E}[\sum_{t=0}^{H} R(s_t)|\pi_\theta]$$

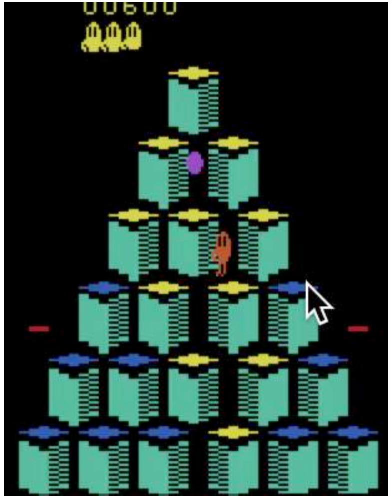
- **Compared to supervised learning, additional challenges:**
  - **Credit assignment**
  - **Stability**
  - **Exploration**

[Image credit: Sutton and Barto 1998]

Deep RL Success Stories







DQN Mnih et al, NIPS 2013 / Nature 2015

MCTS Guo et al, NIPS 2014; **TRPO** Schulman, Levine, Moritz, Jordan, Abbeel, ICML 2015; **A3C** Mnih et al, ICML 2016; **Dueling DQN** Wang et al ICML 2016; **Double DQN** van Hasselt et al, AAAI 2016; **Prioritized Experience Replay** Schaul et al, ICLR 2016; **Bootstrapped DQN** Osband et al, 2016; **Q-Ensembles** Chen et al, 2017; **Rainbow** Hessel et al, 2017; **Accelerated** Stooke and Abbeel, 2018; …

Deep RL Success Stories

AlphaGo Silver et al, Nature 2015

AlphaGoZero Silver et al, Nature 2017

AlphaZero Silver et al, 2017

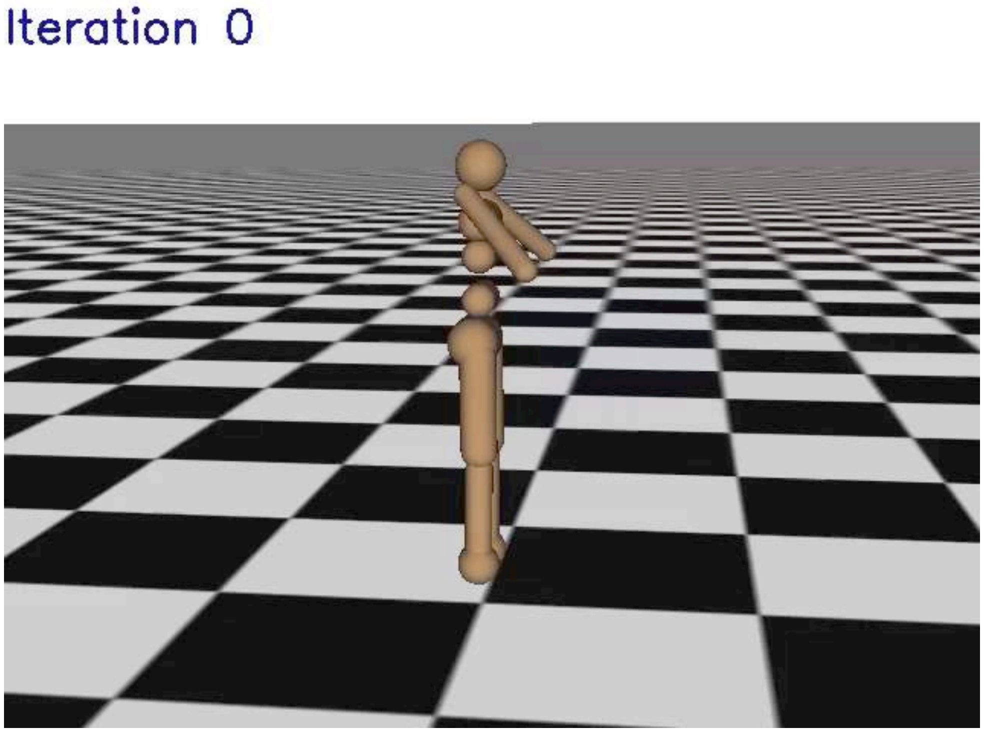
Tian et al, 2016; Maddison et al, 2014; Clark et al, 2015

Deep RL Success Stories

- Super-human agent on Dota2 1v1, enabled by
  - Reinforcement learning
  - Self-play
  - Enough computation

[OpenAI, 2017]

Deep RL Success Stories



^ TRPO Schulman et al, 2015 + **GAE** Schulman et al, 2016

See also: **DDPG** Lillicrap et al 2015; **SVG** Heess et al, 2015; **Q-Prop** Gu et al, 2016; **Scaling up ES** Salimans et al, 2017; **PPO** Schulman et al, 2017; **Parkour** Heess et al, 2017;

Deep RL Success Stories

[Peng, Abbeel, Levine, van de Panne, 2018]

Deep RL Success: Dynamic Animation

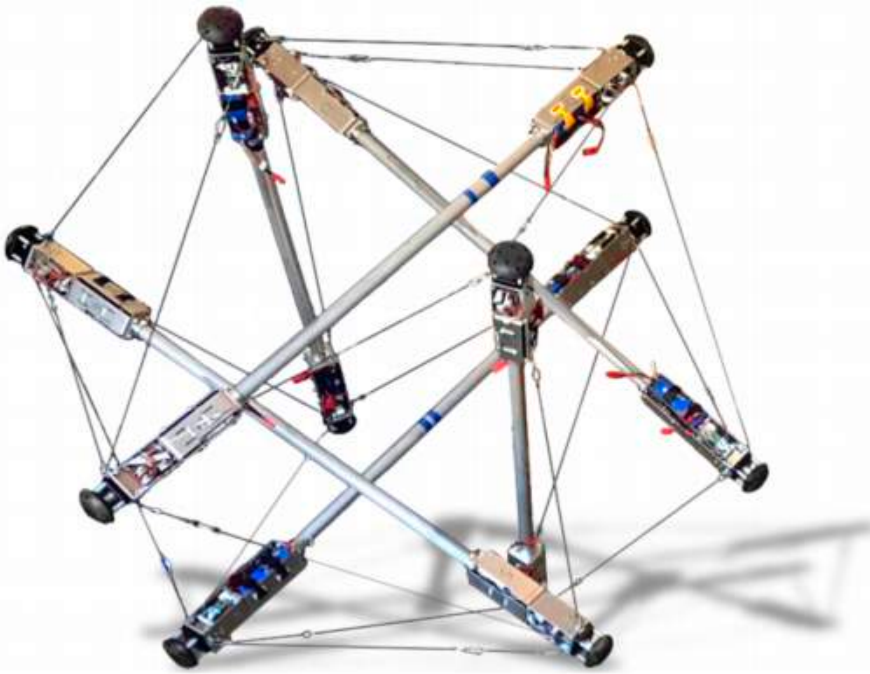
[Peng, Abbeel, Levine, van de Panne, 2018]

Deep RL Success Stories

Guided Policy Search Levine*, Finn*, Darrell, Abbeel, JMLR 2016

Tensegrity Robotics: NASA SuperBall

- Rigid rods connected by elastic cables
- Controlled by motors that extend / contract cables
- Properties:
  - Lightweight
  - Low cost
  - Capable of withstanding significant impact
- NASA investigates them for space exploration
- Major challenge: control



[Geng*, Zhang*, Bruce*, Caluwaerts, Vespignani, Sunspiral, Abbeel, Levine, ICRA 2017]

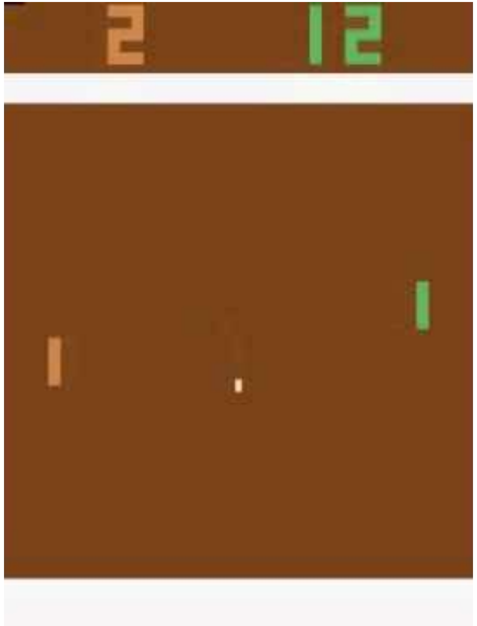
[Geng*, Zhang*, Bruce*, Caluwaerts, Vespignani, Sunspiral, Abbeel, Levine, ICRA 2017]

Mastery? Yes

Deep RL (DQN)

Score: **18.9**

vs.



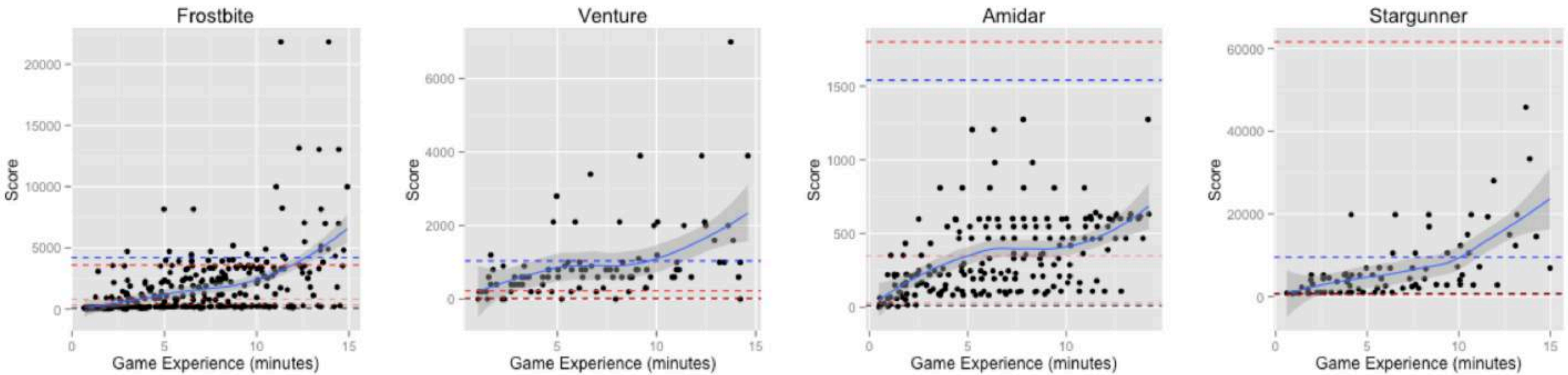
Human

Score: 9.3

But how good is the learning?

Humans vs. DDQN

Humans after 15 minutes tend to outperform DDQN after 115 hours



Black dots: human play

Blue curve: mean of human play

Blue dashed line: 'expert' human play

Red dashed lines:

DDQN after 10, 25, 200M frames

(~ 46, 115, 920 hours)

[Tsividis, Pouncy, Xu, Tenenbaum, Gershman, 2017]

How to bridge this gap?

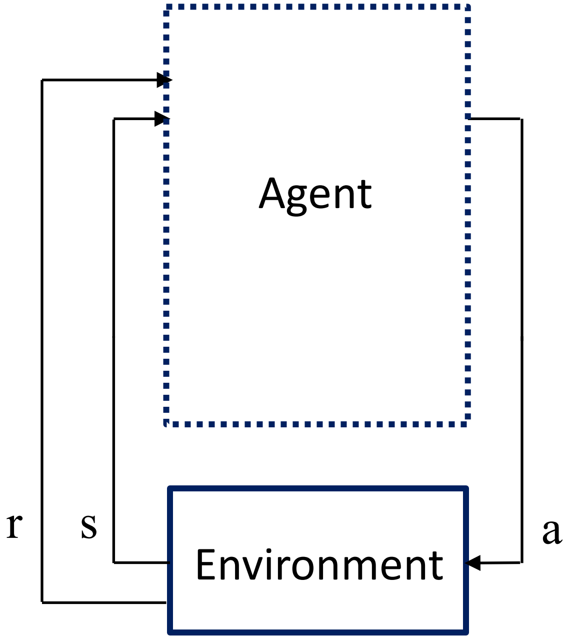
Starting Observations

- TRPO, DQN, A3C, DDPG, PPO, Rainbow, … are fully general RL algorithms
  - i.e., for any environment that can be mathematically defined, these algorithms are equally applicable

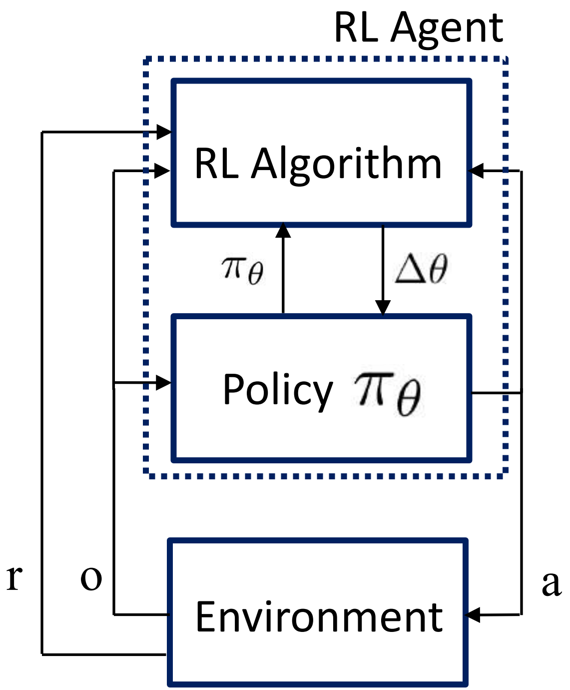
- Environments encountered in real world
  = tiny, tiny subset of all environments that could be defined
  (e.g. they all satisfy our universe’s physics)

Can we develop “fast” RL algorithms that take advantage of this?

Reinforcement Learning

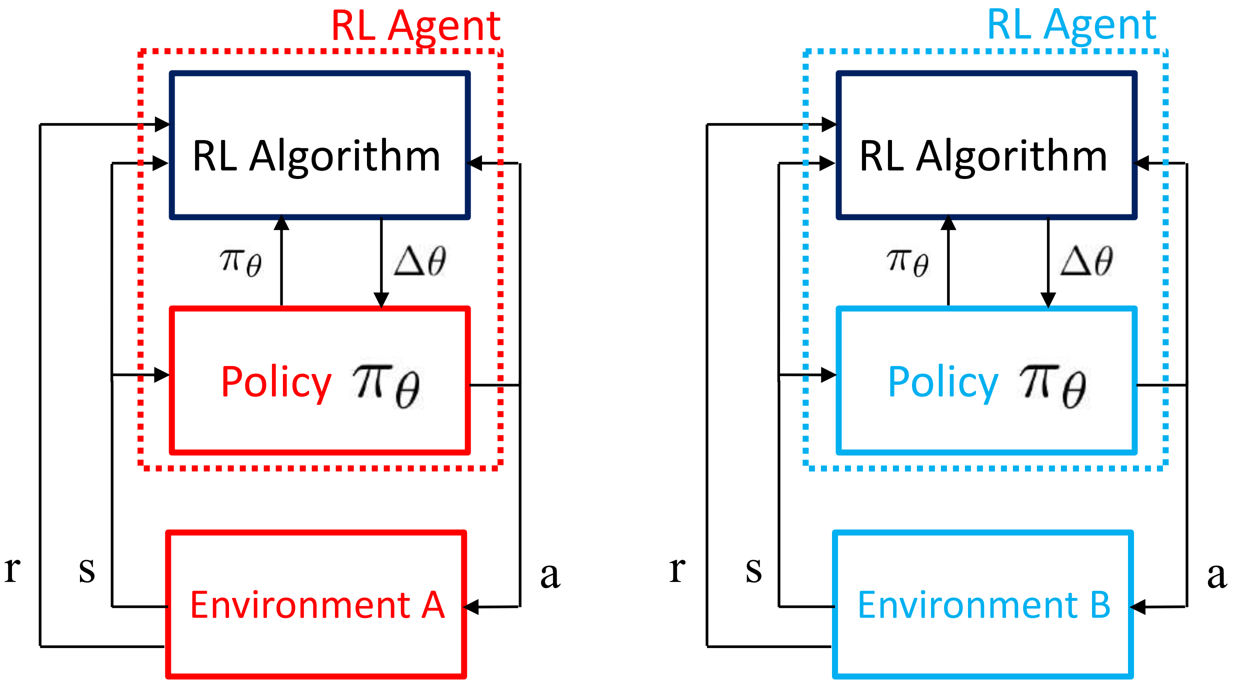


Reinforcement Learning



[Duan, Schulman, Chen, Bartlett, Sutskever, Abbeel, 2016] also: [Wang et al, 2016]

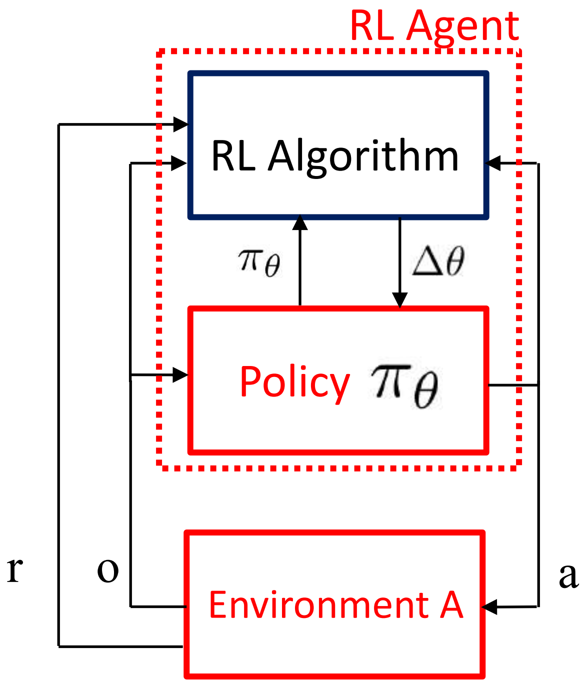
Reinforcement Learning

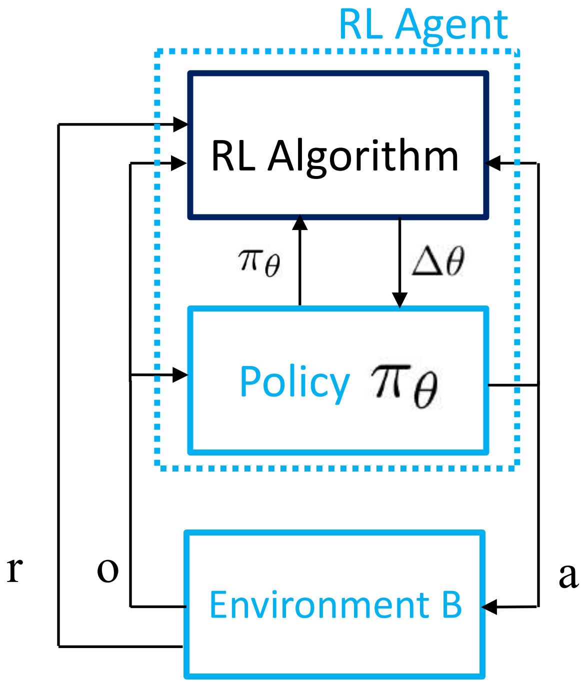


...

[Duan, Schulman, Chen, Bartlett, Sutskever, Abbeel, 2016] also: [Wang et al, 2016]

Reinforcement Learning





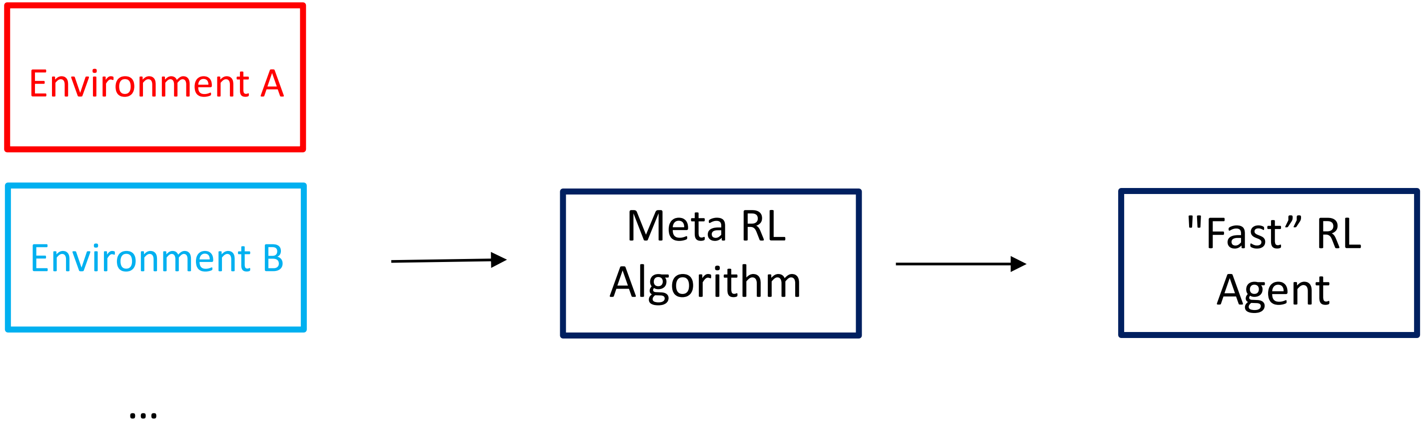
Traditional RL research:
- Human experts develop the RL algorithm
- After many years, still no RL algorithms nearly as good as humans…

Alternative:
- Could we learn a better RL algorithm?
- Or even learn a better entire agent?

[Duan, Schulman, Chen, Bartlett, Sutskever, Abbeel, 2016] also: [Wang et al, 2016]

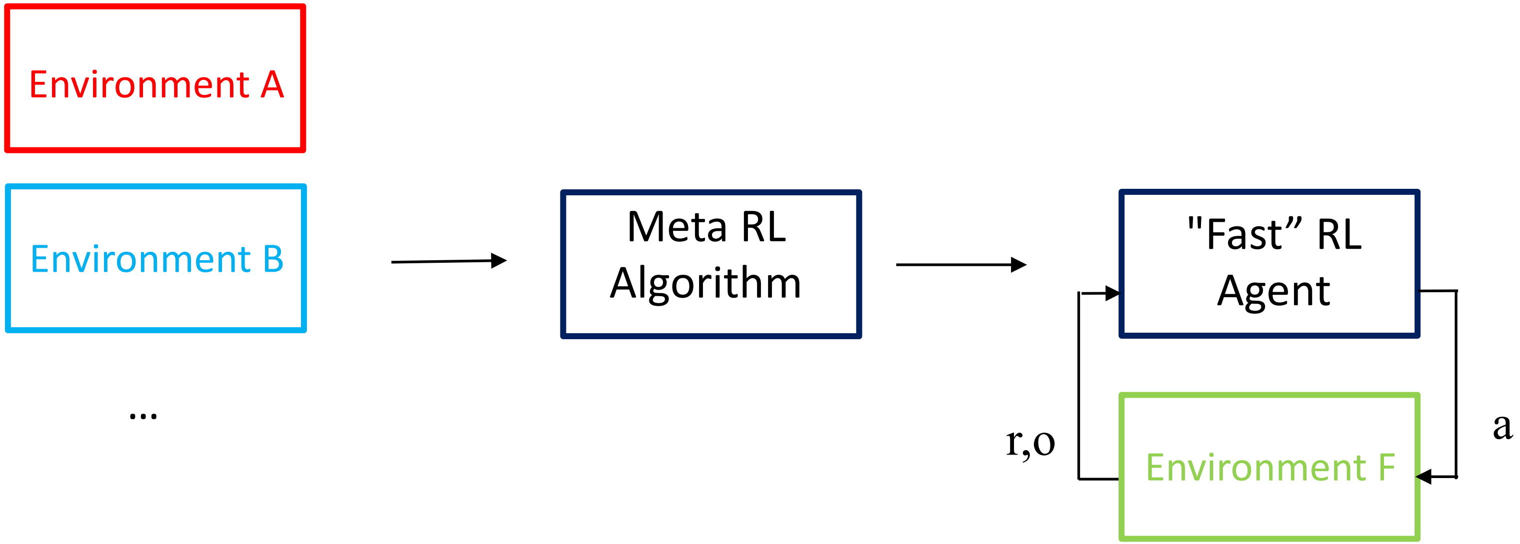
Meta-Reinforcement Learning

Meta-training environments



Meta-Reinforcement Learning

Meta-training environments

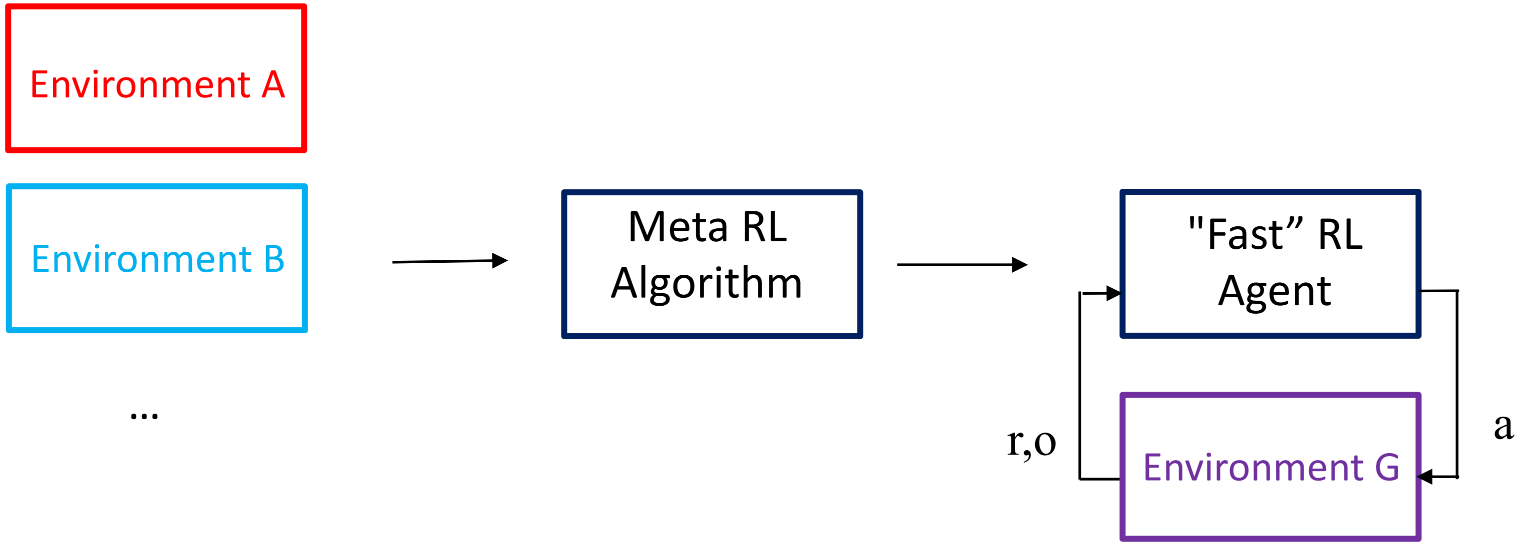


Testing environments

[Duan, Schulman, Chen, Bartlett, Sutskever, Abbeel, 2016] also: [Wang et al, 2016]

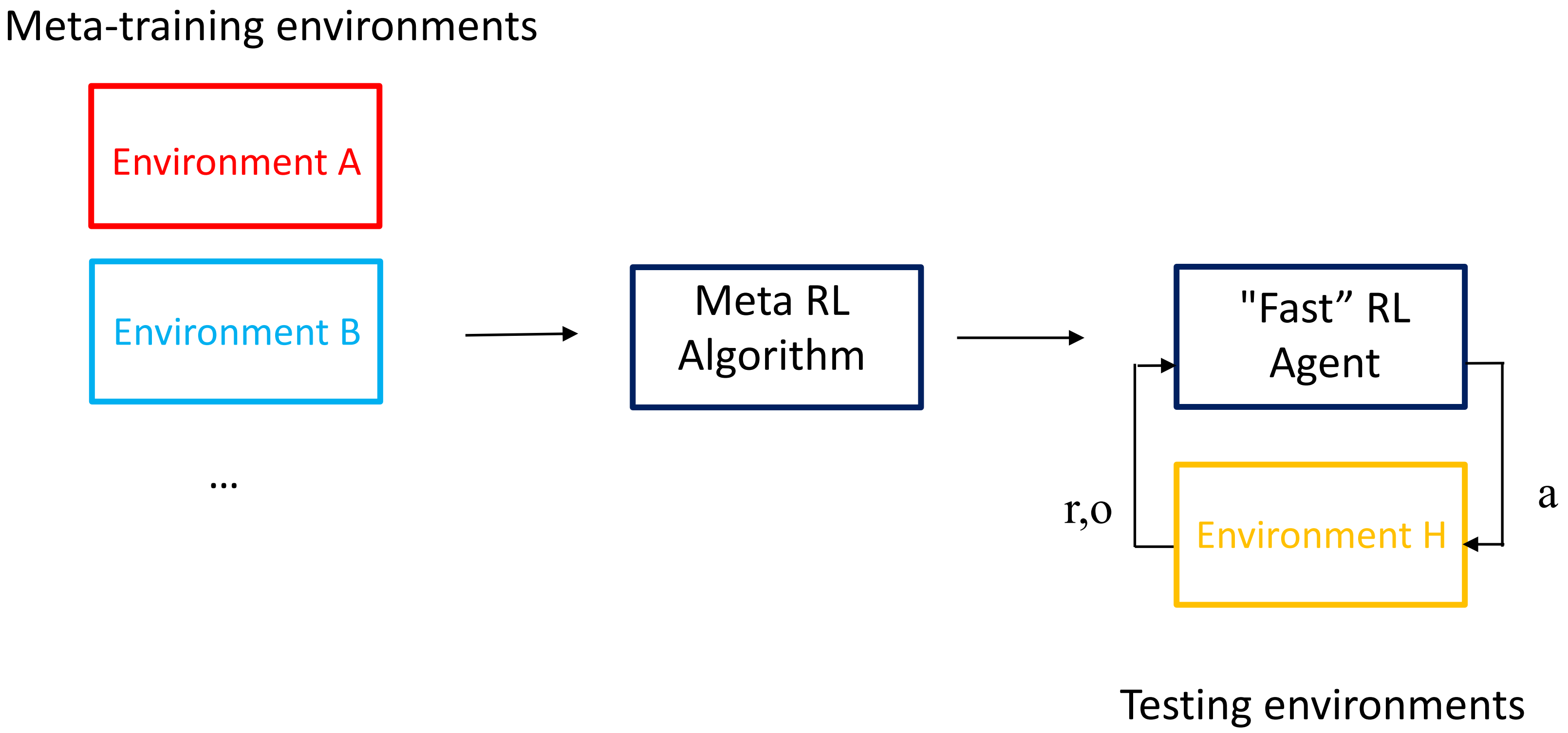
Meta-Reinforcement Learning

Meta-training environments



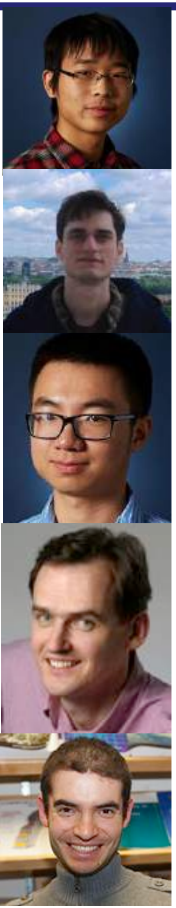
Testing environments

Meta-Reinforcement Learning



[Duan, Schulman, Chen, Bartlett, Sutskever, Abbeel, 2016] also: [Wang et al, 2016]

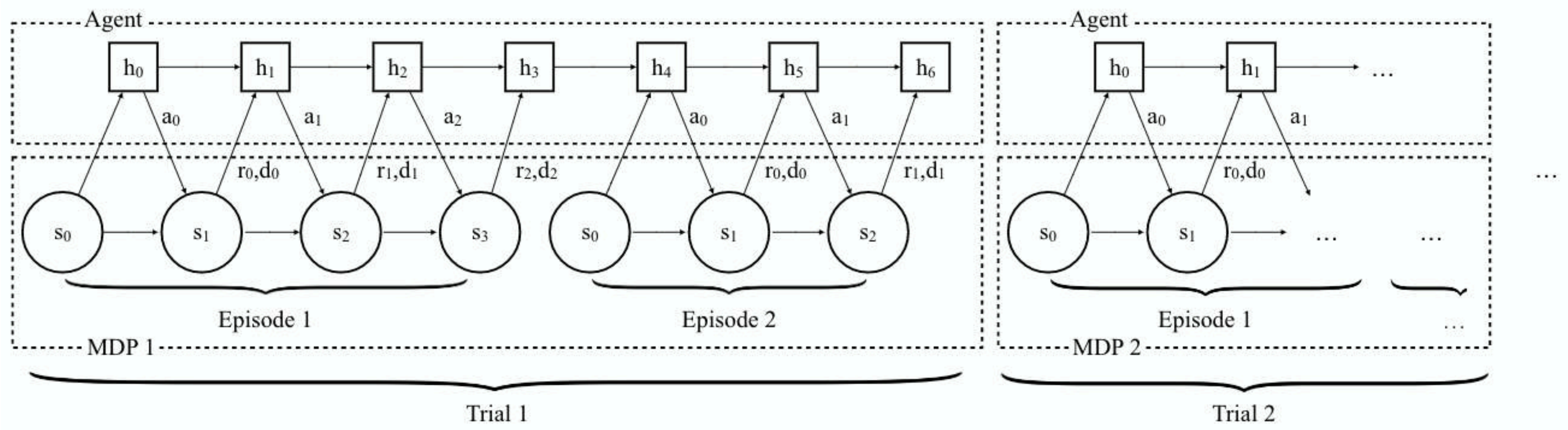
Formalizing Learning to Reinforcement Learn



$$\max_\theta \mathbb{E}_M \mathbb{E}_{\tau_M^{(k)}} \left[\sum_{k=1}^{K} R(\tau_M^{(k)}) \mid \text{RLagent}_\theta \right]$$

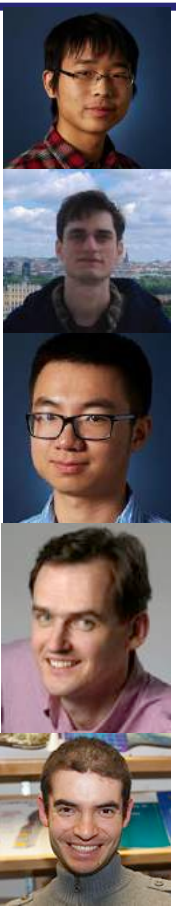
M : sample environment

$\tau_M^{(k)}$: k'th episode in environment M



[Duan, Schulman, Chen, Bartlett, Sutskever, Abbeel, 2016] also: [Wang et al, 2016]

Formalizing Learning to Reinforcement Learn



$$\max_{\theta} \mathbb{E}_M \mathbb{E}_{\tau_M^{(k)}} \left[\sum_{k=1}^{K} R(\tau_M^{(k)}) \mid \mathrm{RLagent}_{\theta} \right]$$

M : sample MDP

$\tau_M^{(k)}$: k'th trajectory in MDP M

Meta-train:

$$\max_{\theta} \sum_{M \in M_{\mathrm{train}}} \mathbb{E}_{\tau_M^{(k)}} \left[\sum_{k=1}^{K} R(\tau_M^{(k)}) \mid \mathrm{RLagent}_{\theta} \right]$$

[Duan, Schulman, Chen, Bartlett, Sutskever, Abbeel, 2016] also: [Wang et al, 2016]

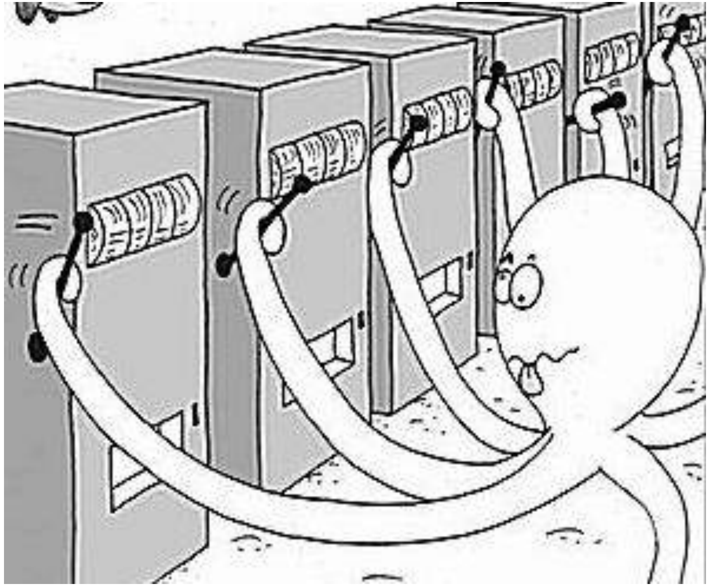
Representing $\mathrm{RLagent}_\theta$

$$\max_{\theta} \sum_{M \in M_{\mathrm{train}}} \mathbb{E}_{\tau_M^{(k)}} \left[\sum_{k=1}^{K} R(\tau_M^{(k)}) \mid \mathrm{RLagent}_\theta \right]$$

- RLagent = RNN = generic computation architecture
  - different weights in the RNN means different RL algorithm and prior
  - different activations in the RNN means different current policy
  - meta-train objective can be optimized with an existing (slow) RL algorithm

Evaluation: Multi-Armed Bandits

- Multi-Armed Bandits setting
  - Each bandit has its own distribution over pay-outs
  - Each episode = choose 1 bandit
  - Good RL agent should explore bandits sufficiently, yet also exploit the good/best ones
- Provably (asymptotically) optimal RL algorithms have been invented by humans: Gittins index, UCB1, Thompson sampling, …



[Duan, Schulman, Chen, Bartlett, Sutskever, Abbeel, 2016]

Evaluation: Multi-Armed Bandits

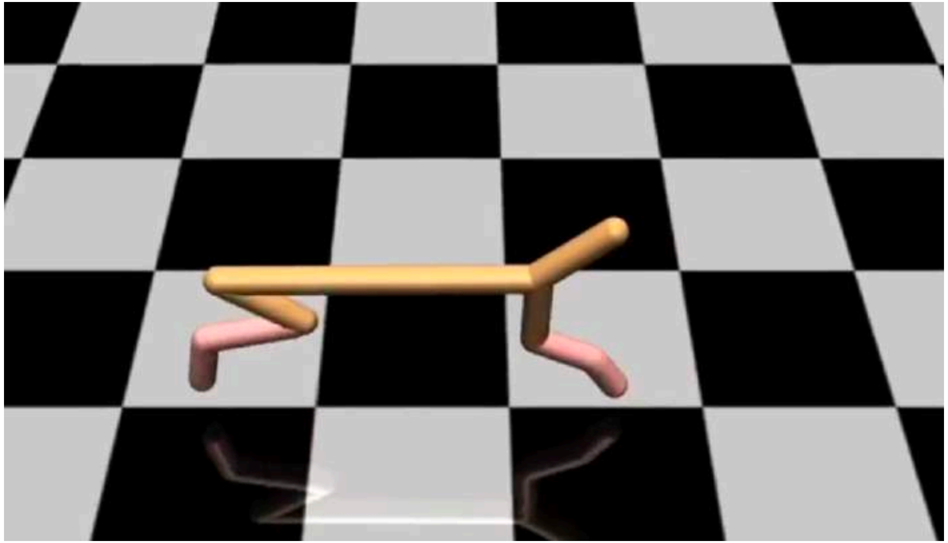
Setup	Random	Gittins	TS	OTS	UCB1	ϵ-Greedy	Greedy	RL^2
$n = 10, k = 5$	5.0	**6.6**	5.7	6.5	**6.7**	**6.6**	**6.6**	**6.7**
$n = 10, k = 10$	5.0	**6.6**	5.5	6.2	**6.7**	**6.6**	**6.6**	**6.7**
$n = 10, k = 50$	5.1	6.5	5.2	5.5	**6.6**	6.5	6.5	**6.8**
$n = 100, k = 5$	49.9	**78.3**	74.7	**77.9**	**78.0**	75.4	74.8	**78.7**
$n = 100, k = 10$	49.9	**82.8**	76.7	81.4	82.4	77.4	77.1	**83.5**
$n = 100, k = 50$	49.8	**85.2**	64.5	67.7	84.3	78.3	78.0	**84.9**
$n = 500, k = 5$	249.8	**405.8**	**402.0**	**406.7**	**405.8**	388.2	380.6	**401.6**
$n = 500, k = 10$	249.0	**437.8**	429.5	**438.9**	**437.1**	408.0	395.0	432.5
$n = 500, k = 50$	249.6	**463.7**	427.2	437.6	457.6	413.6	402.8	438.9

We consider Bayesian evaluation setting. Some of these prior works also have adversarial guarantees, which we don't consider here.

[Duan, Schulman, Chen, Bartlett, Sutskever, Abbeel, 2016]

Evaluation: Locomotion – Half Cheetah

- Task – reward based on target running direction + speed



[Duan, Schulman, Chen, Bartlett, Sutskever, Abbeel, 2016]

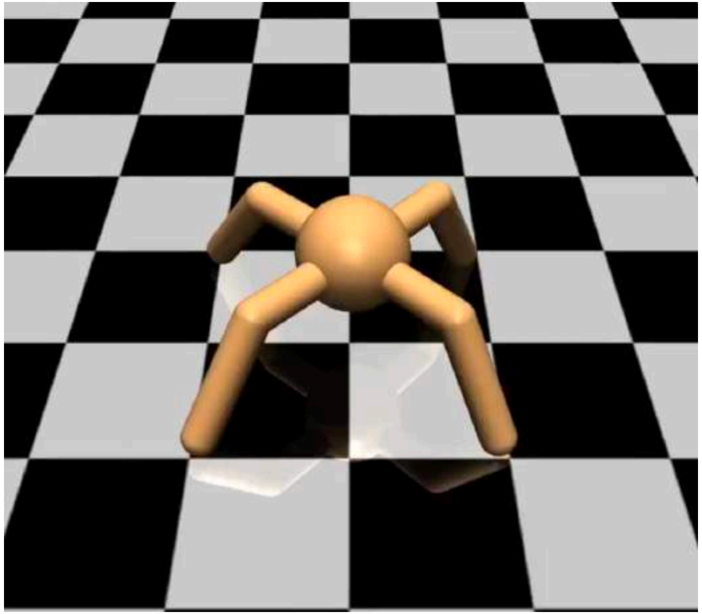
Evaluation: Locomotion – Half Cheetah

- Task – reward based on target running direction + speed

- Result of meta-training = a single agent (the "fast RL agent"), which masters each task almost instantly within 1st episode

Evaluation: Locomotion – Ant

- Task – reward based on target running direction + speed



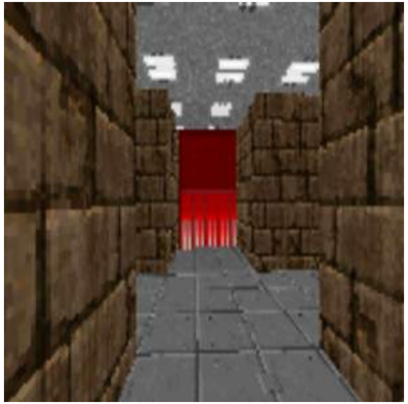
Evaluation: Locomotion – Ant

- Task – reward based on target running direction + speed

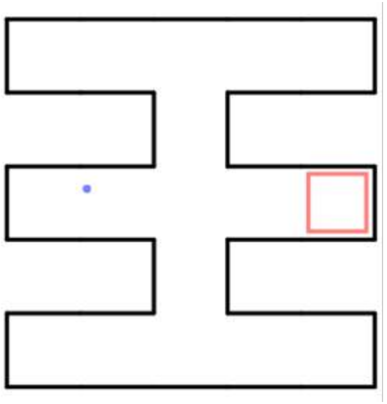
- Result of meta-training = a single agent (the “fast RL agent”), which masters each task almost instantly within 1st episode

Evaluation: Visual Navigation

Agent input: current image
Agent action: straight / 2 degrees left / 2 degrees right
Map just shown for our purposes, but not available to agent



Agent's view

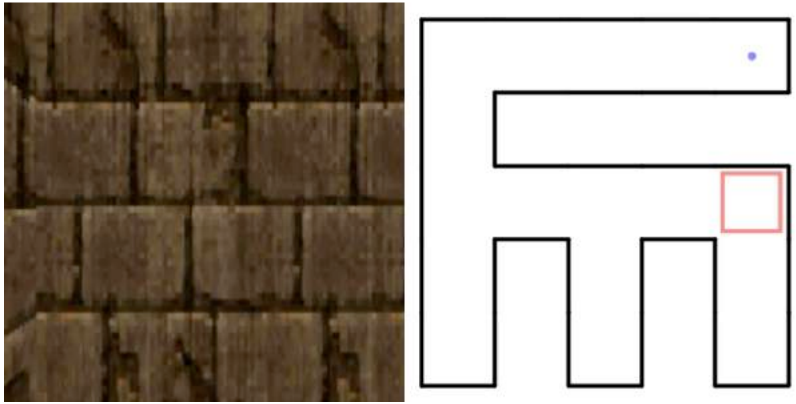


Maze

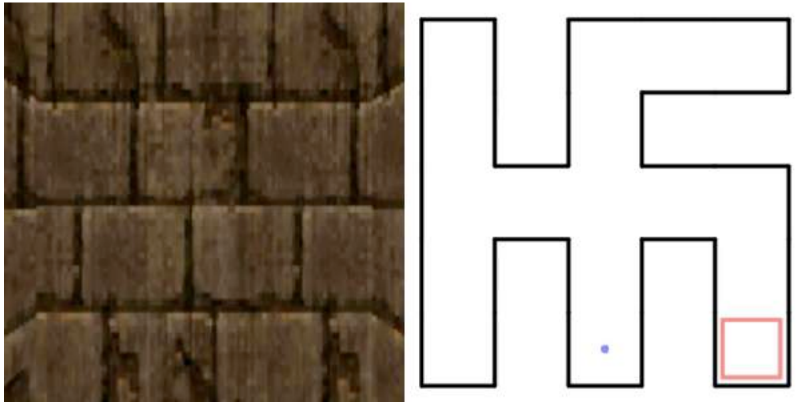
Related work: Mirowski, et al, 2016; Jaderberg et al, 2016; Mnih et al, 2016; Wang et al, 2016

Evaluation: Visual Navigation

Agent input: current image
Agent action: straight / 2 degrees left / 2 degrees right
Map just shown for our purposes, but not available to agent



Before learning-to-learn



After learning-to-learn

[Duan, Schulman, Chen, Bartlett, Sutskever, Abbeel, 2016]

Meta-Learning Curves

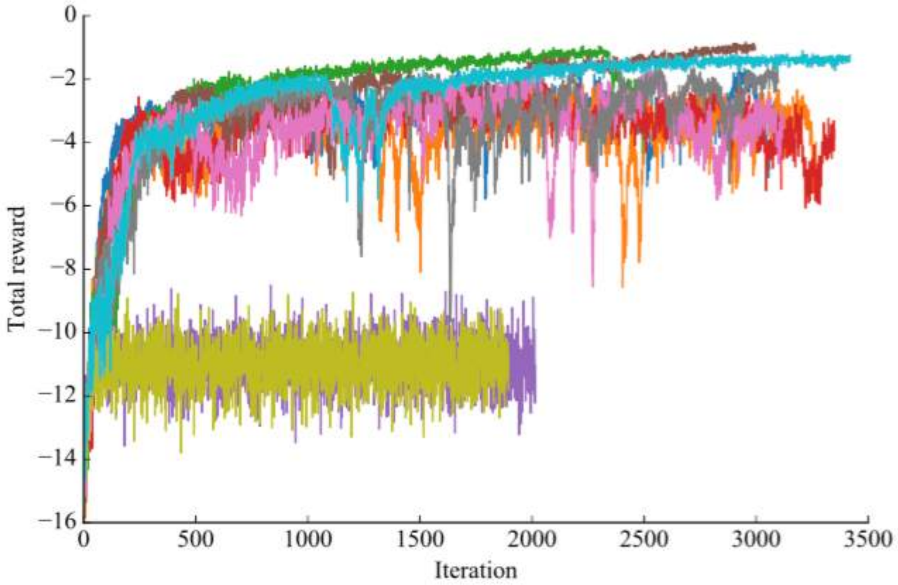


Figure 5: RL^2 learning curves for visual navigation. Each curve shows a different random initialization of the RNN weights. Performance varies greatly across different initializations.

[Duan, Schulman, Chen, Bartlett, Sutskever, Abbeel, 2016]

Other Architecture

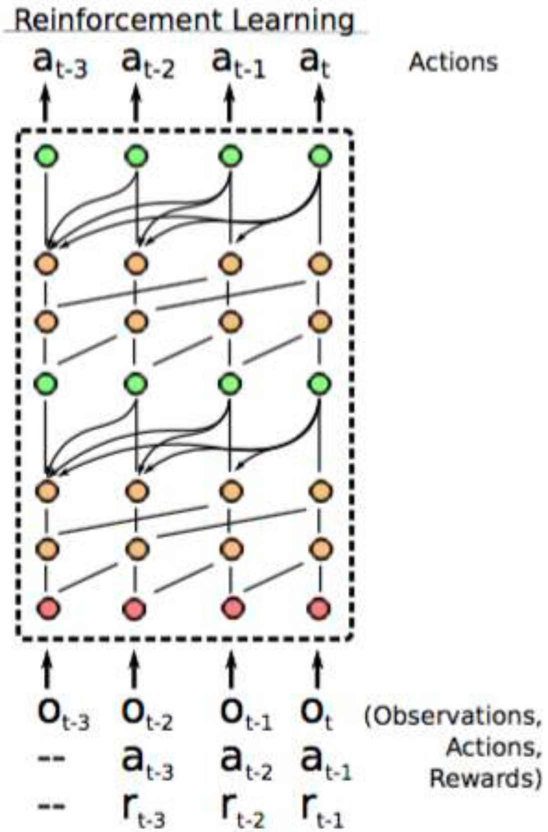
- **Simple Neural Attentive Meta-Learner (SNAIL)**

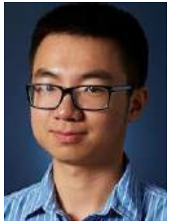
  [Mishra, Rohaninejad, Chen, Abbeel, 2017]

Simple Neural Attentive Meta-Learner

- Like RL2 but:

replace the LSTM with dilated temporal convolution (like wavenet) + attention





[Wavenet: van den Oord et al, 2016]

[Attention-is-all-you-need: Vaswani et al, 2017]

[Mishra*, Rohaninejad*, Chen, Abbeel, 2017]

Bandits

Setup (N, K)	Method				
	Gittins (optimal as $N \to \infty$)	Random	RL^2	MAML	SNAIL (ours)
$10, 5$	**6.6**	5.0	**6.7**	6.5 ± 0.1	$\mathbf{6.6 \pm 0.1}$
$10, 10$	**6.6**	5.0	**6.7**	$\mathbf{6.6 \pm 0.1}$	$\mathbf{6.7 \pm 0.1}$
$10, 50$	6.5	5.1	**6.8**	$\mathbf{6.6 \pm 0.1}$	$\mathbf{6.7 \pm 0.1}$
$100, 5$	**78.3**	49.9	**78.7**	67.1 ± 1.1	$\mathbf{79.1 \pm 1.0}$
$100, 10$	**82.8**	49.9	**83.5**	70.1 ± 0.6	$\mathbf{83.5 \pm 0.8}$
$100, 50$	**85.2**	49.8	**84.9**	70.3 ± 0.4	$\mathbf{85.1 \pm 0.6}$
$500, 5$	**405.8**	249.8	401.5	–	$\mathbf{408.1 \pm 4.9}$
$500, 10$	**437.8**	249.0	**432.5**	–	$\mathbf{432.4 \pm 3.5}$
$500, 50$	**463.7**	249.6	438.9	–	442.6 ± 2.5
$1000, 50$	**944.1**	499.8	847.43	–	889.8 ± 5.6

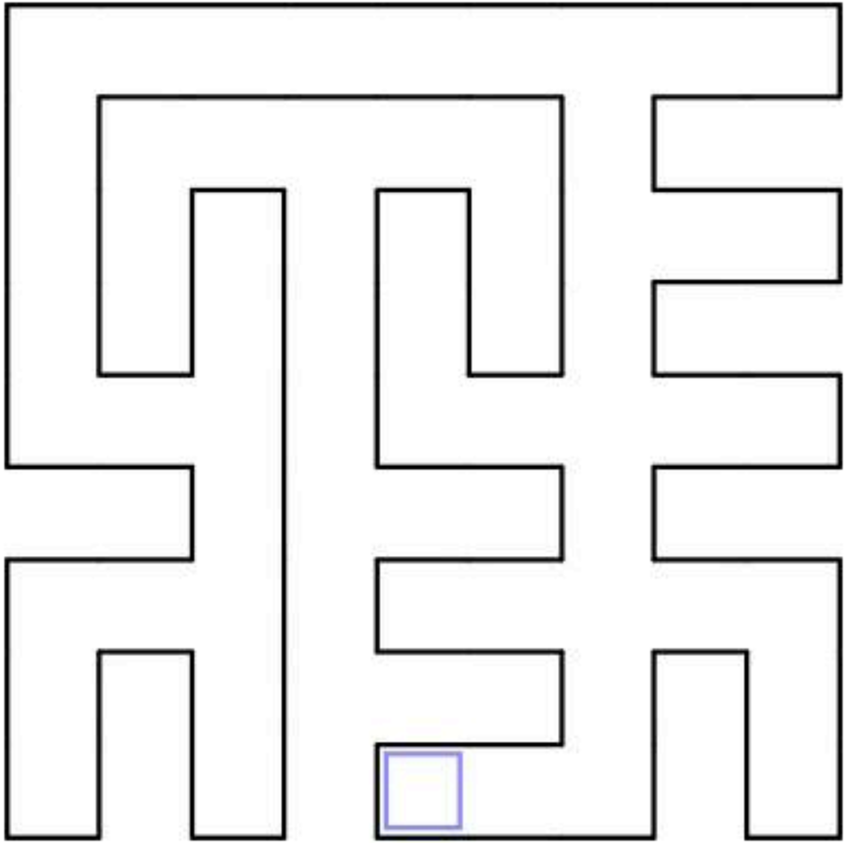
[Mishra*, Rohaninejad*, Chen, Abbeel, 2017]

Vision-based Navigation in Unknown Maze

Method	Small Maze		Large Maze	
	Episode 1	Episode 2	Episode 1	Episode 2
Random	188.6 ± 3.5	187.7 ± 3.5	420.2 ± 1.2	420.8 ± 1.2
LSTM	52.4 ± 1.3	39.1 ± 0.9	180.1 ± 6.0	150.6 ± 5.9
SNAIL (ours)	$\mathbf{50.3 \pm 0.3}$	$\mathbf{34.8 \pm 0.2}$	$\mathbf{140.5 \pm 4.2}$	$\mathbf{105.9 \pm 2.4}$

[Mishra*, Rohaninejad*, Chen, Abbeel, 2017]

Agent Dropped in New Maze



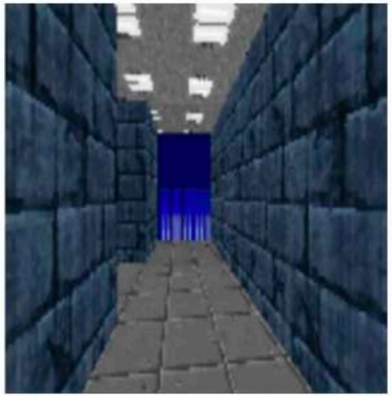
[Mishra*, Rohaninejad*, Chen, Abbeel, 2017]

Meta Learning for RL

Task distribution: different environments

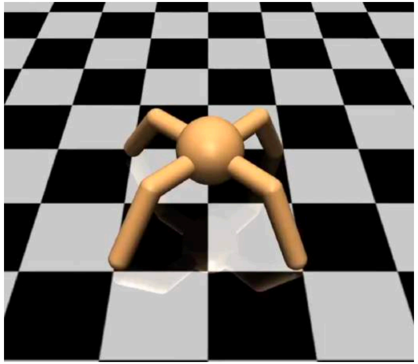
- Schmidhuber. Evolutionary principles in self-referential learning. (1987)
- Wiering, Schmidhuber. Solving POMDPs with Levin search and EIRA. (1996)
- Schmidhuber, Zhao, Wiering. Shifting inductive bias with success-story algorithm, adaptive Levin search, and incremental self-improvement. (MLJ 1997)
- Schmidhuber, Zhao, Schraudolph. Reinforcement learning with self-modifying policies (1998)
- Zhao, Schmidhuber. Solving a complex prisoner's dilemma with self-modifying policies. (1998)
- Schmidhuber. A general method for incremental self-improvement and multiagent learning. (1999)
- Singh, Lewis, Barto. Where do rewards come from? (2009)
- Singh, Lewis, Barto. Intrinsically Motivated Reinforcement Learning: An Evolutionary Perspective (2010)
- Niekum, Spector, Barto. Evolution of reward functions for reinforcement learning (2011)
- Duan et al., (2016) RL2: Fast Reinforcement Learning via Slow Reinforcement Learning
- Wang et al., (2016) Learning to Reinforcement Learn
- Finn et al., (2017) Model-Agnostic Meta-Learning (MAML)
- Mishra, Rohinenjad et al., (2017) Simple Neural AttentIve meta-Learner
- Frans et al., (2017) Meta-Learning Shared Hierarchies
- Stadie et al (2018) Some considerations on learning to explore via meta-reinforcement learning
- Gupta et al (2018) Meta-reinforcement learning of structured exploration strategies
- Nagabandi*, Clavera*, et al (2018) Learning to adapt in dynamic, real-world environments through meta-reinforcement learning
- Clavera et al (2018) Model-based reinforcement learning via meta-policy optimization
- Botvinick et al (2019) Reinforcement learning: fast and slow
- Rakelly et al (2019) Efficient off-policy meta-reinforcement learning via probabilistic context variables
- …

Benchmarks / Environments



Vizdoom

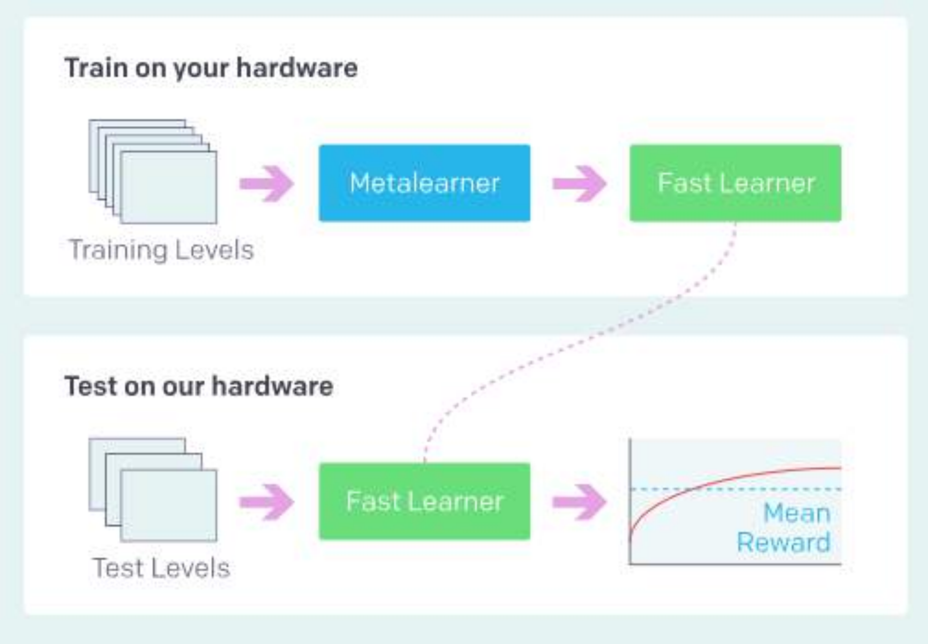
Deepmind Lab



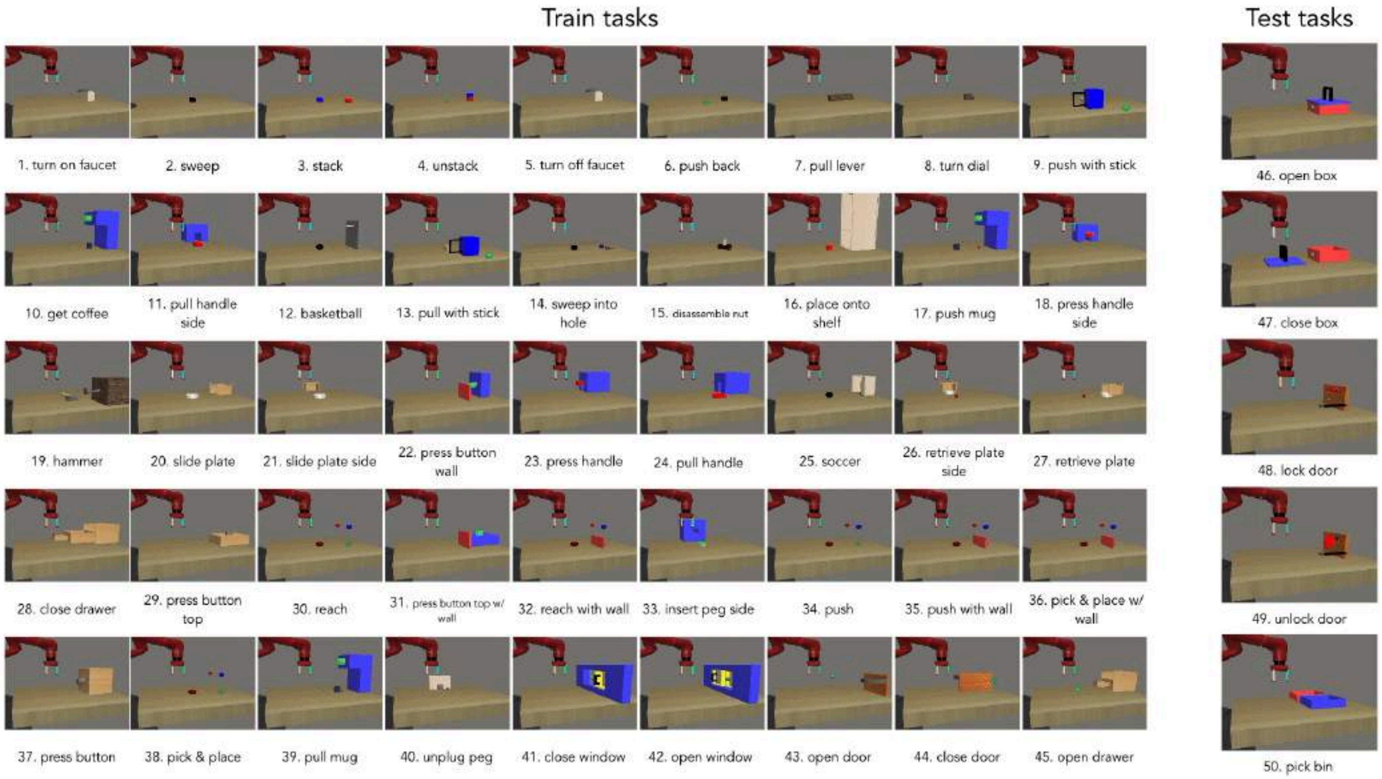
mujoco

Probably need more / richer environments…

OpenAI Retro Contest



Meta-World



[Yu, Quillen, he, Julian, Hausman, Finn, Levine, 2019]

Many Exciting Directions in AI

- Few-Shot Learning
- Reinforcement Learning
- ***Imitation Learning***
- Domain Randomization
- Architecture Search
- Unsupervised Learning

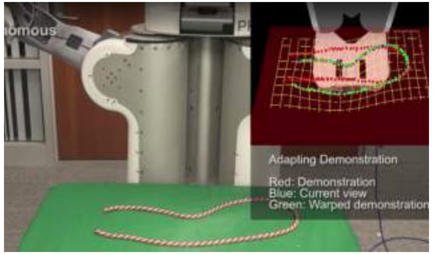
- Lifelong Learning
- Bias in ML (avoiding)
- Long Horizon Reasoning
- Safe Learning
- Value Alignment
- Planning + Learning
- …

Imitation Learning in Robotics

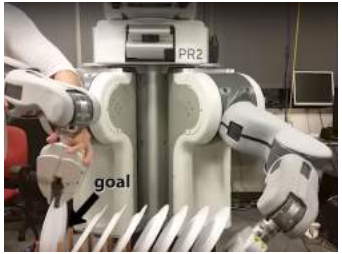
[Abbeel et al. 2008]

[Kolter et al. 2008]

[Ziebart et al. 2008]

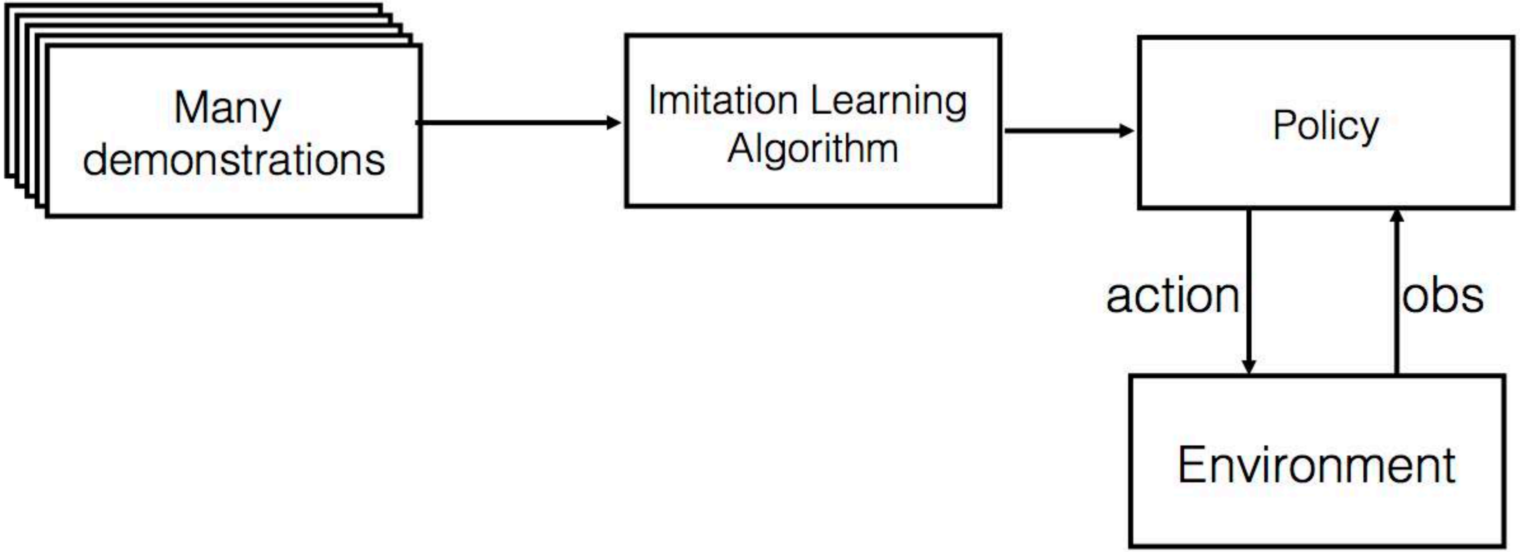


[Schulman et al. 2013]

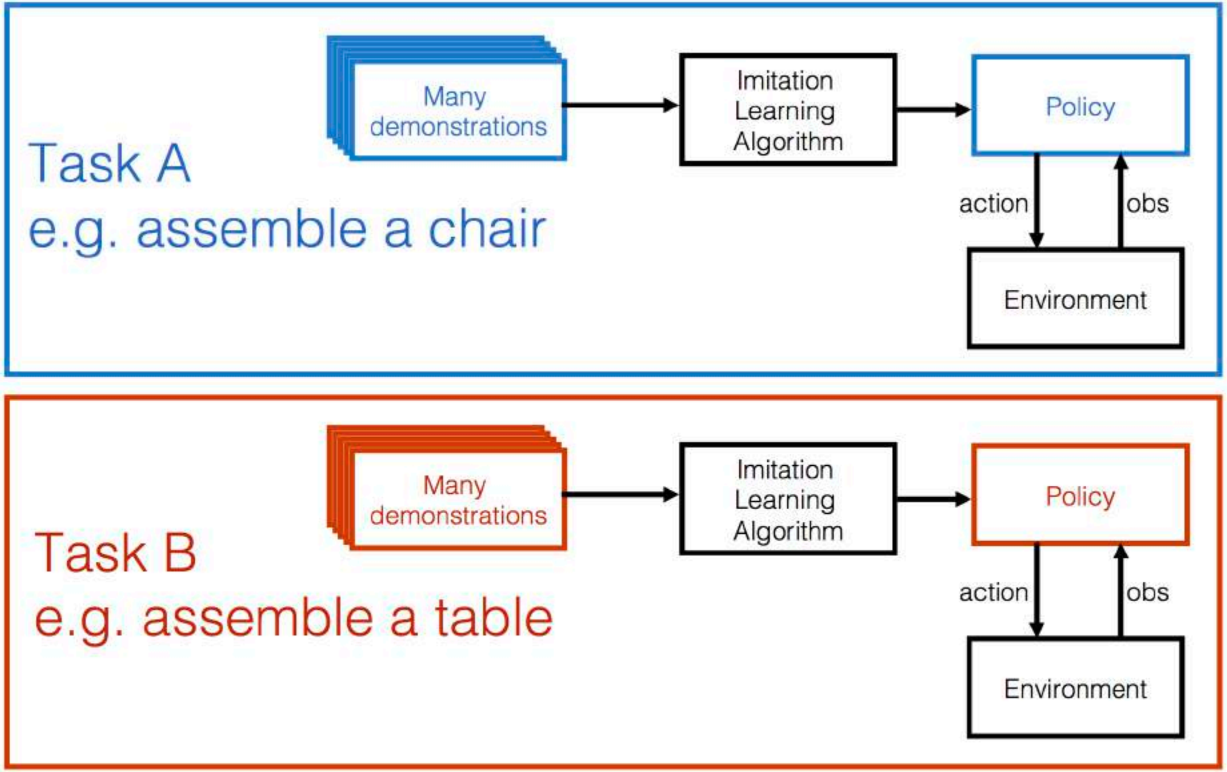


[Finn et al. 2016]

Imitation Learning

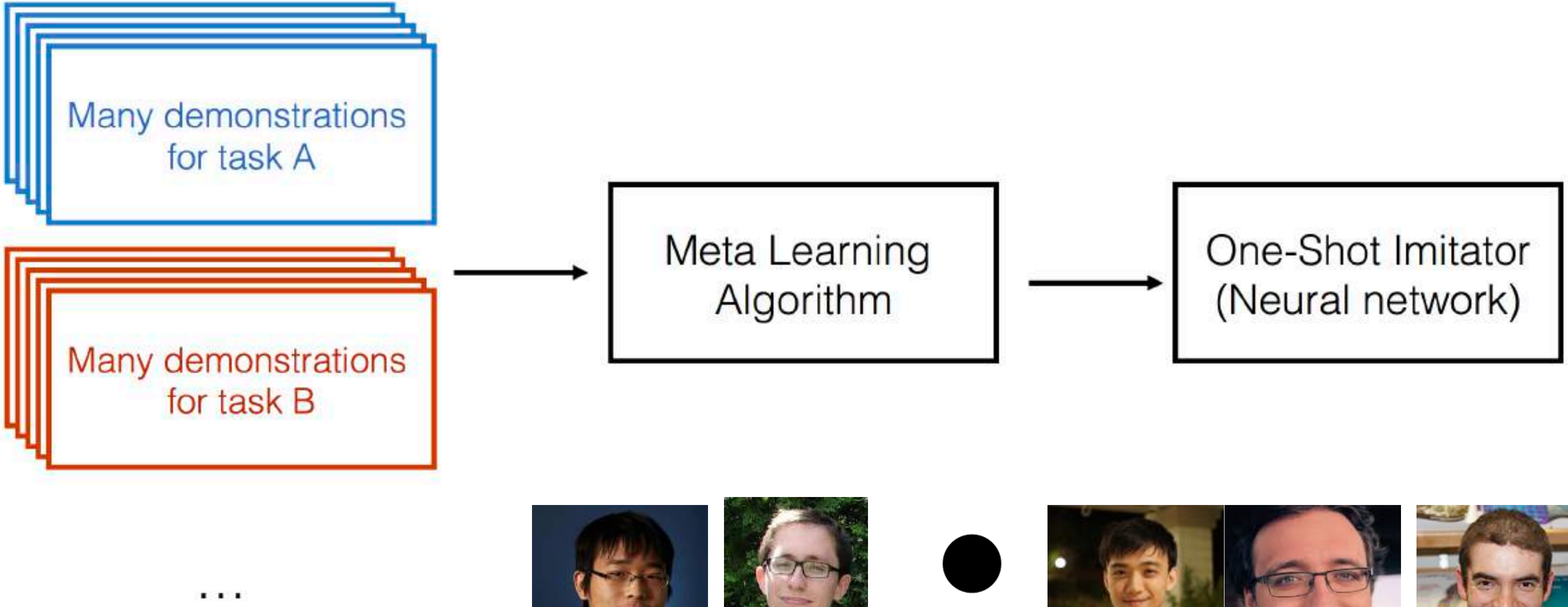


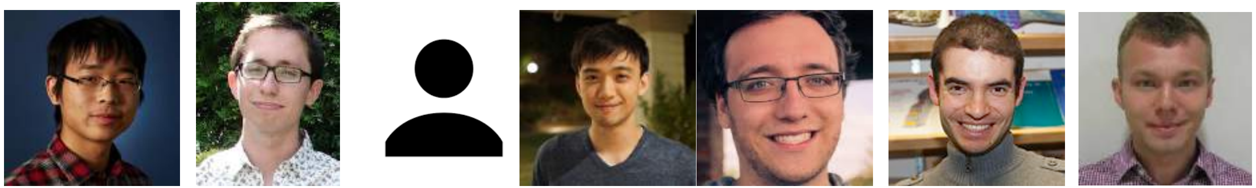
Imitation Learning



...

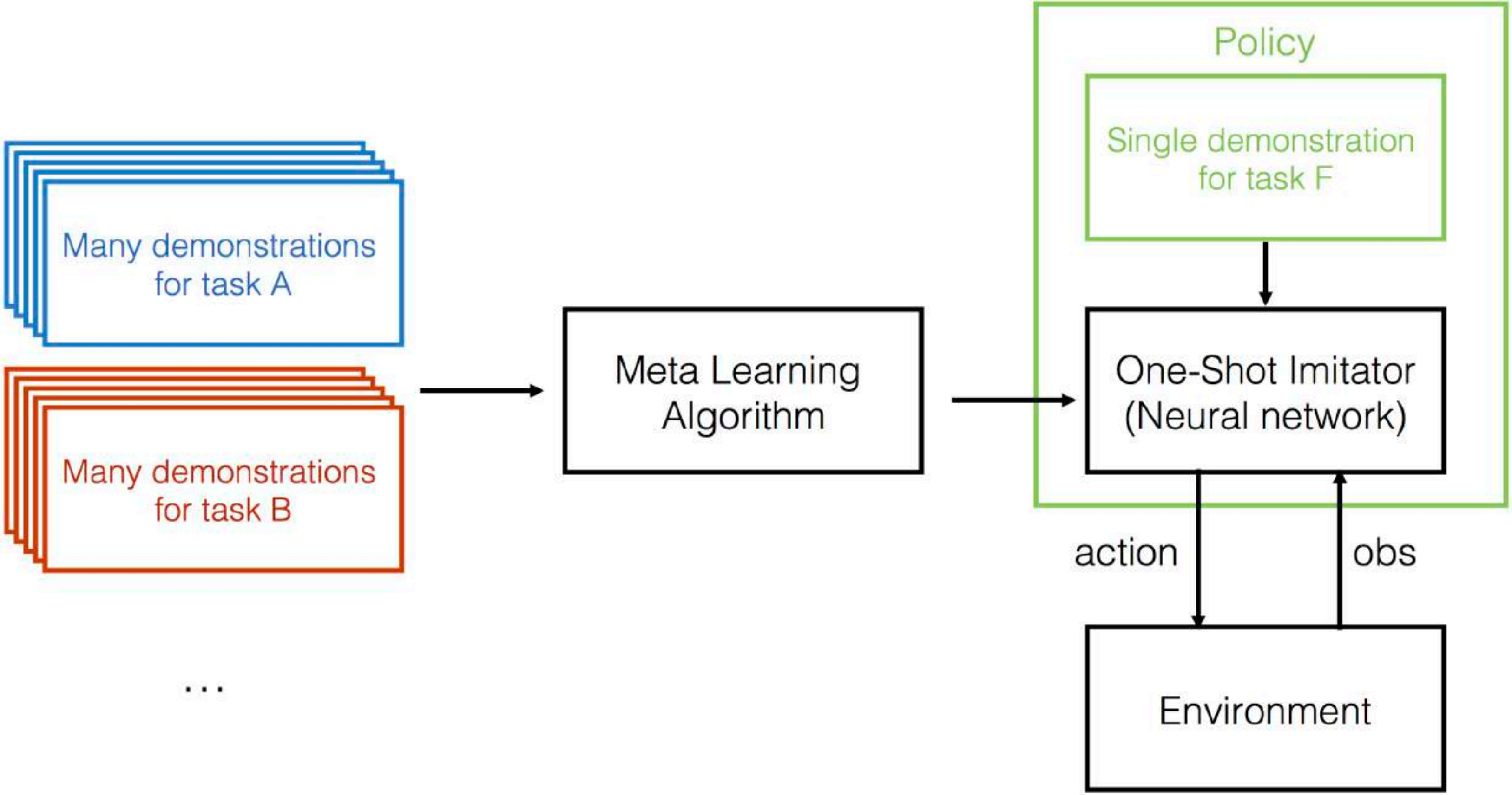
One-Shot Imitation Learning





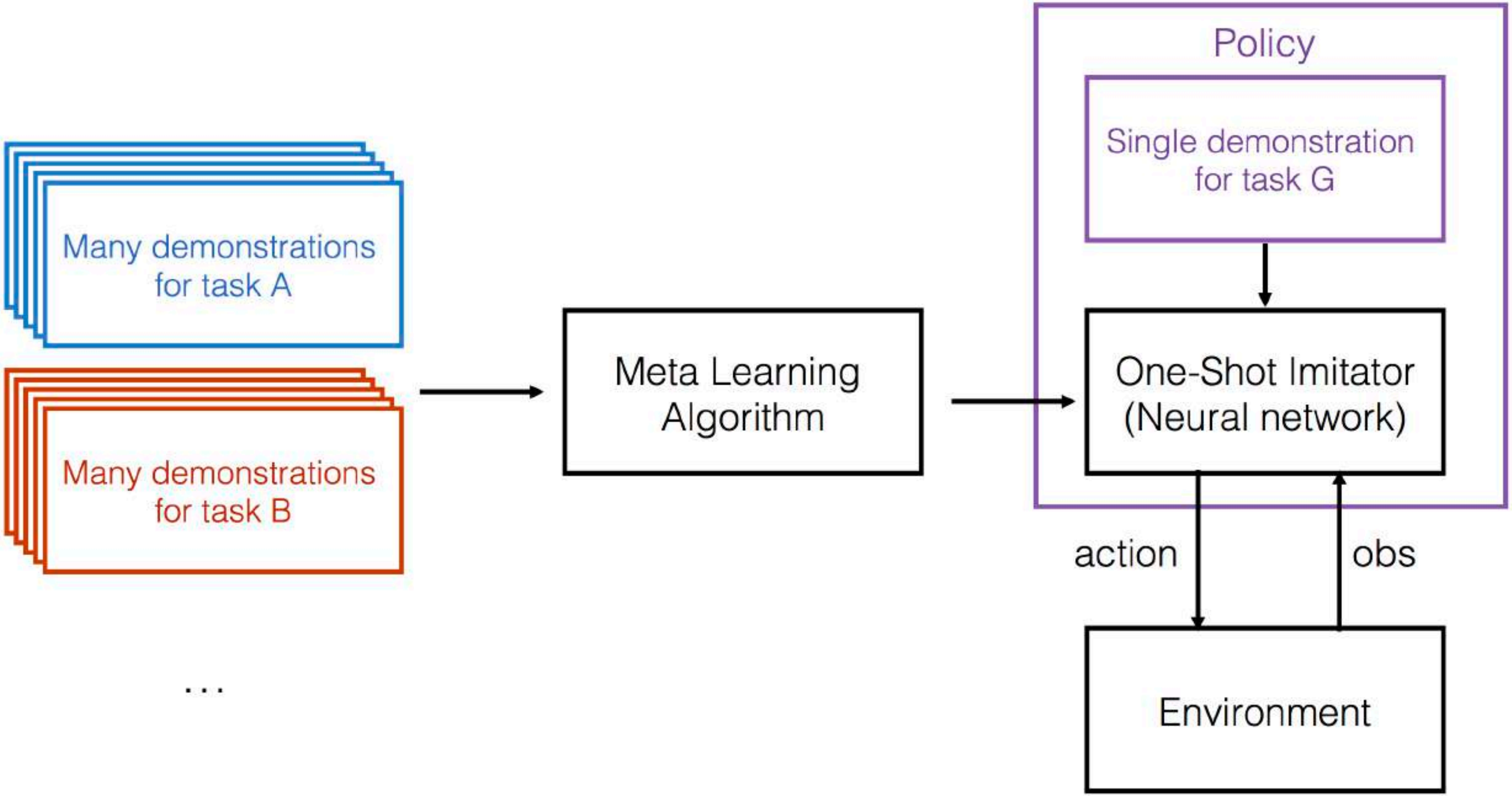
[Duan, Andrychowicz, Stadie, Ho, Schneider, Sutskever, Abbeel, Zaremba, 2017]

One-Shot Imitation Learning



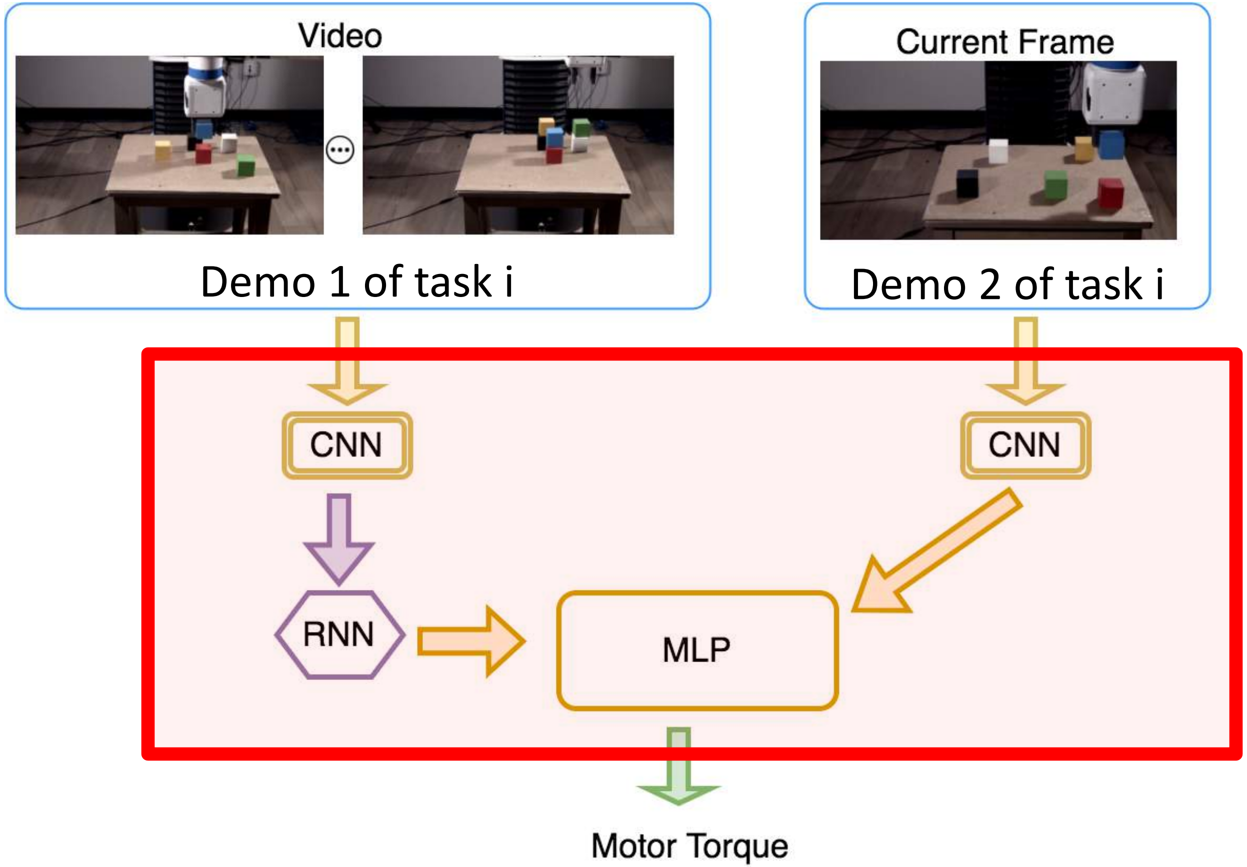
[Duan, Andrychowicz, Stadie, Ho, Schneider, Sutskever, Abbeel, Zaremba, 2017]

One-Shot Imitation Learning



[Duan, Andrychowicz, Stadie, Ho, Schneider, Sutskever, Abbeel, Zaremba, 2017]

Learning a One-Shot Imitator

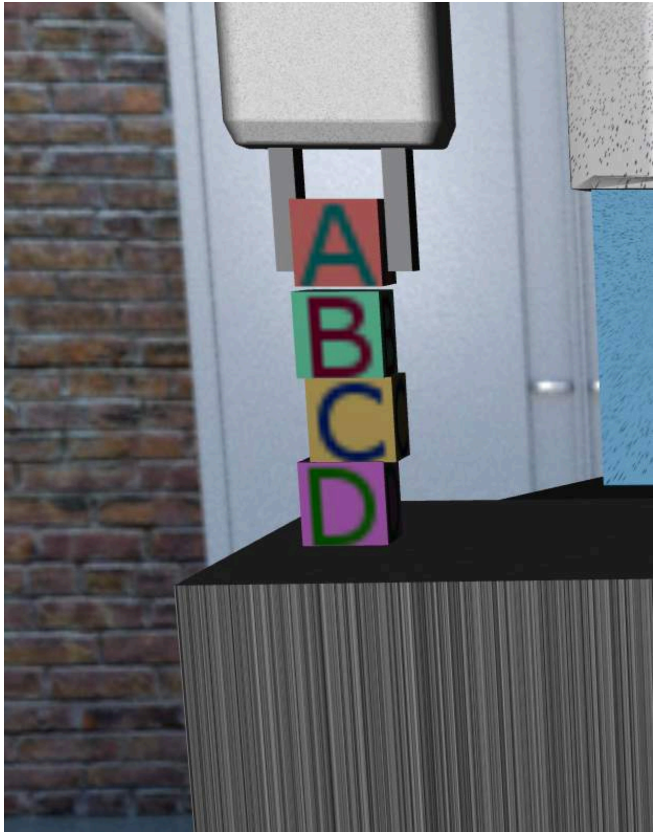


One-shot imitator

[Figure credit: Bradly Stadie]

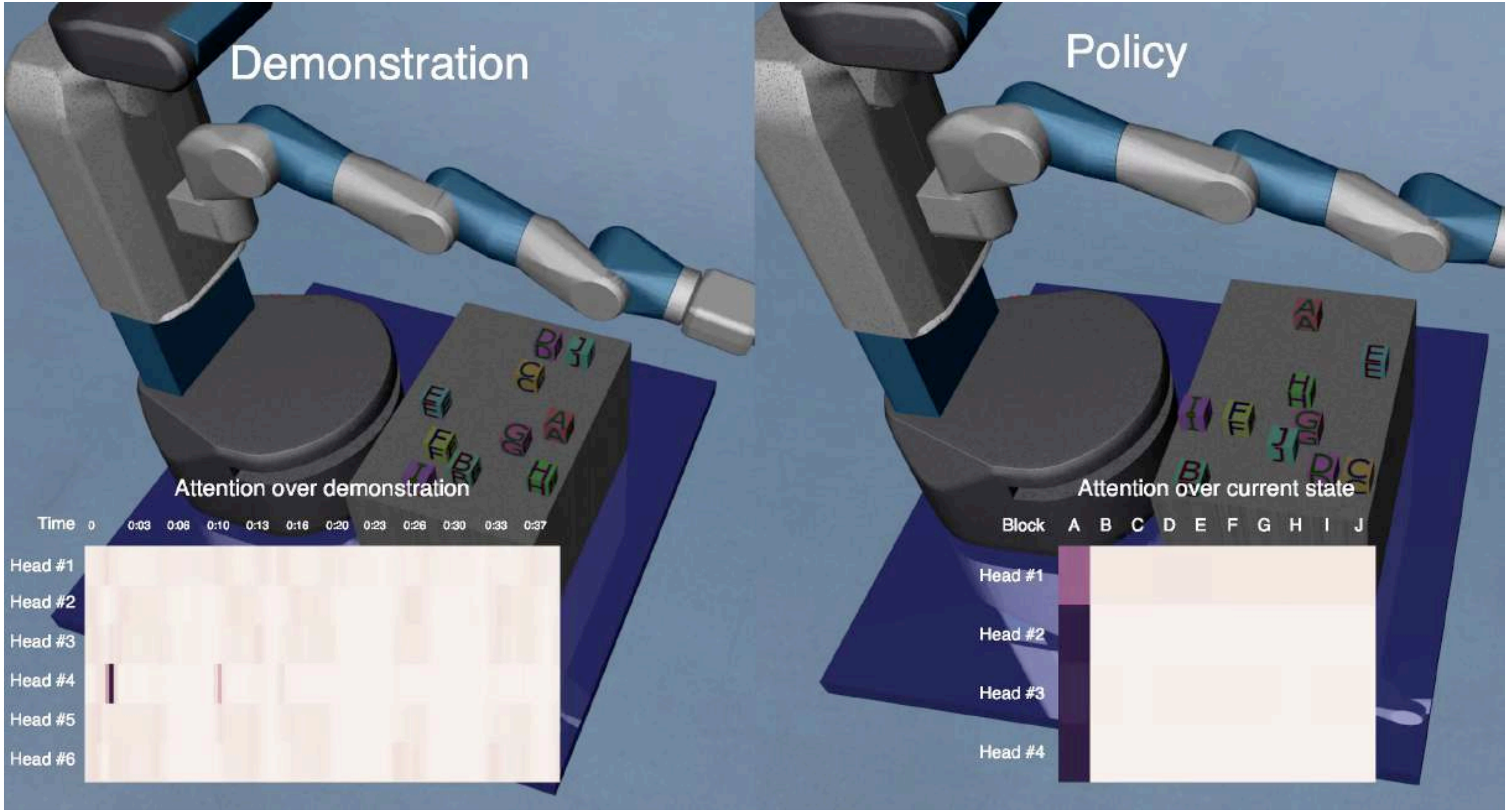
Proof-of-concept: Block Stacking

- Each task is specified by a desired final layout
  - Example: abcd
    - “Place c on top of d,
      place b on top of c,
      place a on top of b.”



[Duan, Andrychowicz, Stadie, Ho, Schneider, Sutskever, Abbeel, Zaremba, 2017]

Evaluation



[Duan, Andrychowicz, Stadie, Ho, Schneider, Sutskever, Abbeel, Zaremba, 2017]

Learning a One-Shot Imitator with MAML

- Meta-learning loss:

$$\min_{\theta} \sum_{\text{tasks}} \mathcal{L}_{\text{val}} \left(\theta - \alpha \nabla_{\theta} \mathcal{L}_{\text{train}}(\theta)\right)$$

- Task loss = behavioral cloning loss: [Pomerleau'89,Sammut'92]

$$\mathcal{L}(\theta) = \sum_{t} \|\pi_{\theta}(o_t) - a_t^*\|^2$$




[Finn*, Yu*, Zhang, Abbeel, Levine, 2017]

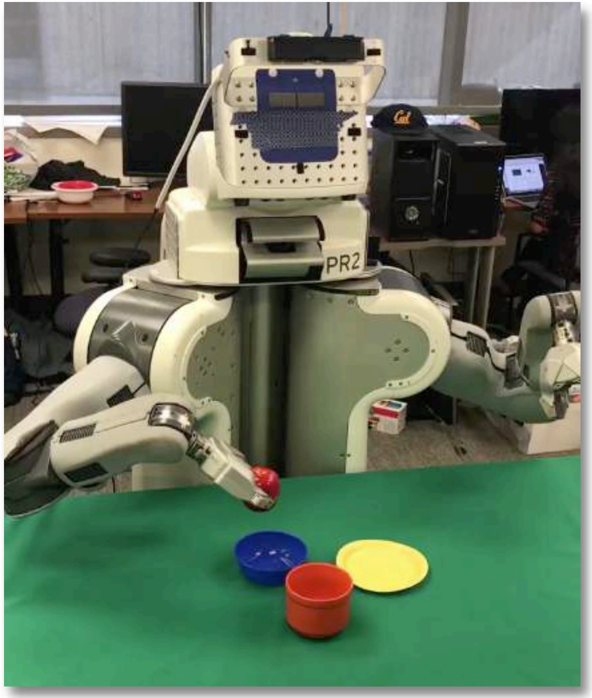
Object placing from pixels

subset of training objects

held-out test objects

input demo
(via teleoperation)

resulting policy



[real-time execution]

Finn*, Yu*, Zhang, Abbeel, Levine CoRL '17

Can We Learn from Just Video?

- Recall, meta-learning loss:

$$\min_{\theta} \sum_{\text{tasks } i} \mathcal{L}^{(i)}_{\text{val}}(\theta - \alpha \nabla_\theta \mathcal{L}^{(i)}_{\text{train}}(\theta)) \qquad \text{with} \qquad \mathcal{L}(\theta) = \sum_t \|\pi_\theta(o_t) - a_t^*\|^2$$

- Key idea: different loss for “val” and “train”

$$\mathcal{L}_{\text{val}}(\theta) = \sum_t \|\pi_\theta(o_t) - a_t^*\|^2$$

“val” only needed during meta-training,
and continues to assume access to action taken

$$\mathcal{L}_{\text{train}}(\theta) = \sum_t \|f_\theta(o_t) - o_{t+1}^*\|^2$$

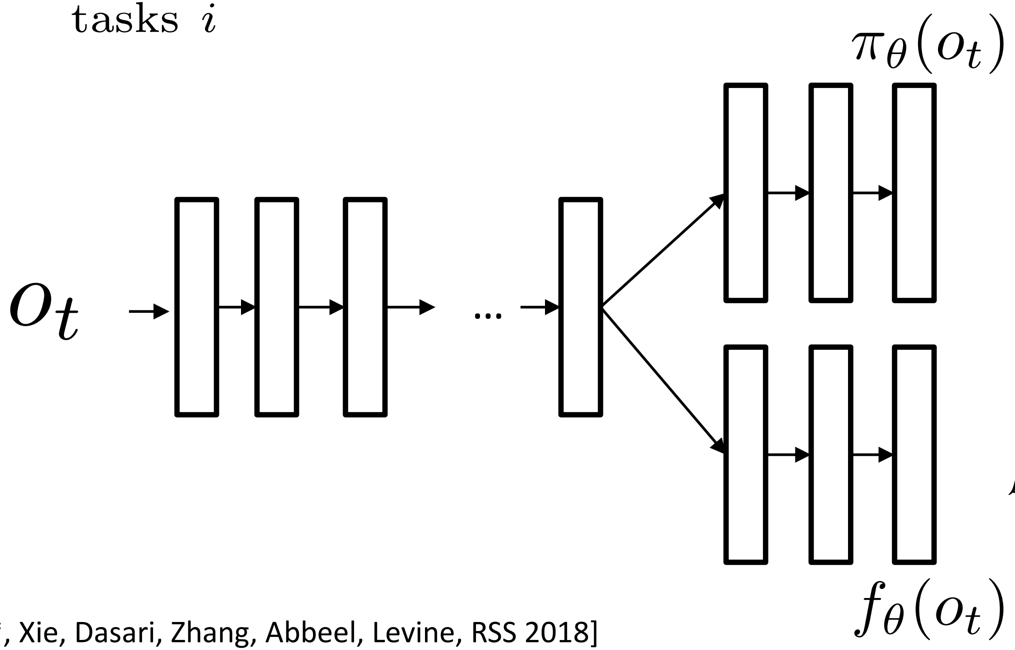
“train” doesn’t require access to action taken,
so at meta-testing video suffices

[Yu*, Finn*, Xie, Dasari, Zhang, Abbeel, Levine, RSS 2018]

Can We Learn from Just Video?

- Recall, meta-learning loss:

$$\min_{\theta} \sum_{\text{tasks } i} \mathcal{L}^{(i)}_{\text{val}}(\theta - \alpha \nabla_\theta \mathcal{L}^{(i)}_{\text{train}}(\theta)) \quad \text{with} \quad \begin{aligned} \mathcal{L}_{\text{val}}(\theta) &= \sum_t \|\pi_\theta(o_t) - a^*_t\|^2 \\ \mathcal{L}_{\text{train}}(\theta) &= \sum_t \|f_\theta(o_t) - o^*_{t+1}\|^2 \end{aligned}$$

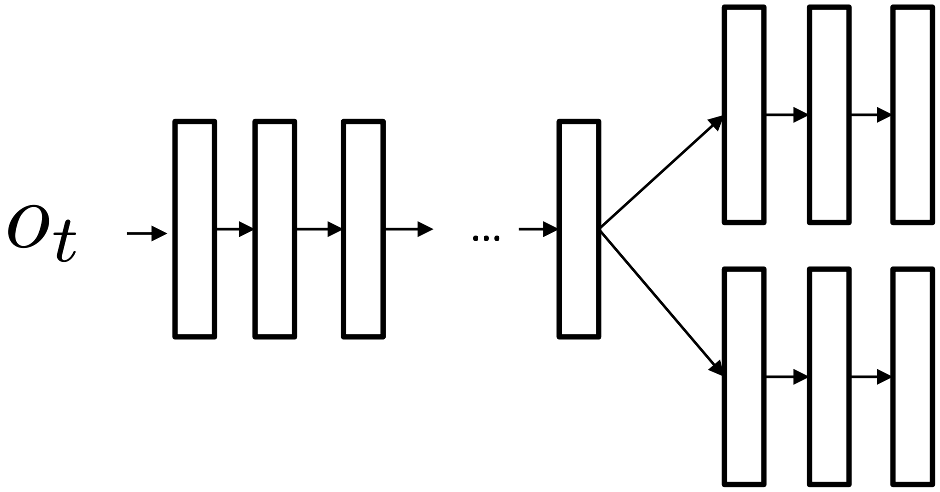


$$\mathcal{L}_{\text{val}}(\theta) = \sum_t \|\pi_\theta(o_t) - a^*_t\|^2$$

$$\mathcal{L}_{\text{train}}(\theta) = \sum_t \|f_\theta(o_t) - o^*_{t+1}\|^2$$

[Yu*, Finn*, Xie, Dasari, Zhang, Abbeel, Levine, RSS 2018]

Can We Learn from Just Video?



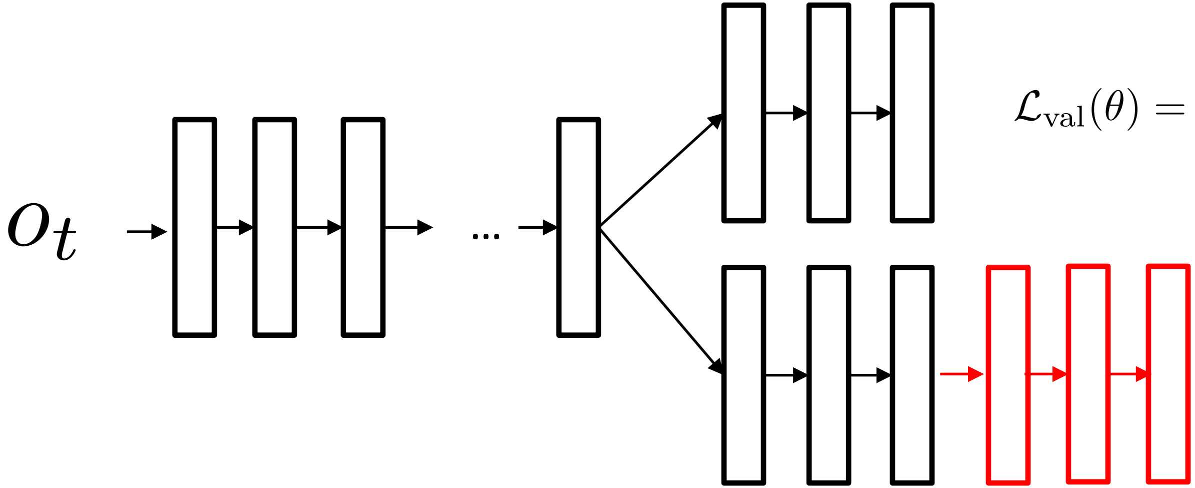
$$\mathcal{L}_{\text{val}}(\theta) = \sum_t \|\pi_\theta(o_t) - a_t^*\|^2$$

$$\mathcal{L}_{\text{train}}(\theta) = \sum_t \|f_\theta(o_t) - o_{t+1}^*\|^2$$

Issue: direct pixel level prediction not a great loss function…

[Yu*, Finn*, Xie, Dasari, Zhang, Abbeel, Levine, RSS 2018]

Can We Learn from Just Video?



$$\mathcal{L}_{\text{val}}(\theta) = \sum_t \|\pi_\theta(o_t) - a_t^*\|^2$$

$$\mathcal{L}_\psi(f_\theta(o_t), o_{t+1}^*)$$

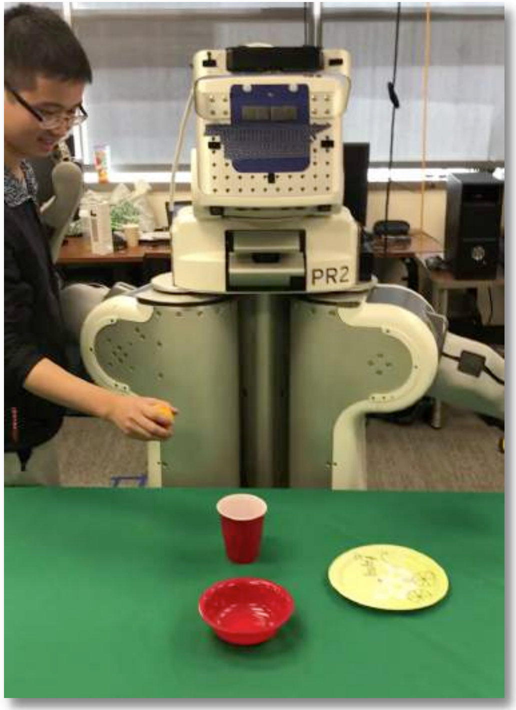
$$\cancel{\mathcal{L}_{\text{train}}(\theta) = \sum_t \|f_\theta(o_t) - o_{t+1}^*\|^2}$$

$\mathcal{L}_\psi$ can be thought of as (learned) Discriminator in GANs

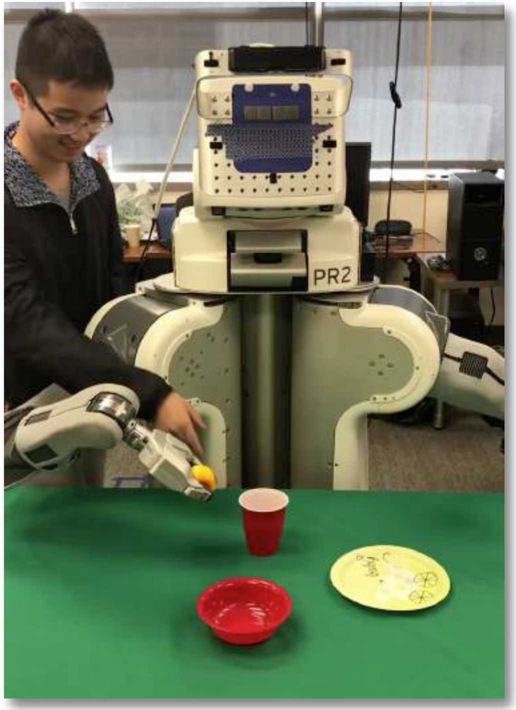
[Yu*, Finn*, Xie, Dasari, Zhang, Abbeel, Levine, RSS 2018]

One-shot imitation from human video

input human demo



resulting policy



[Yu*, Finn*, Xie, Dasari, Zhang, Abbeel, Levine ’18]

Many Exciting Directions in AI

- Few-Shot Learning
- Reinforcement Learning
- Imitation Learning
- ***Domain Randomization***
- Architecture Search
- Unsupervised Learning

- Lifelong Learning
- Bias in ML (avoiding)
- Long Horizon Reasoning
- Safe Learning
- Value Alignment
- Planning + Learning
- …

Motivation for Simulation

Compared to the real world, simulated data collection is…

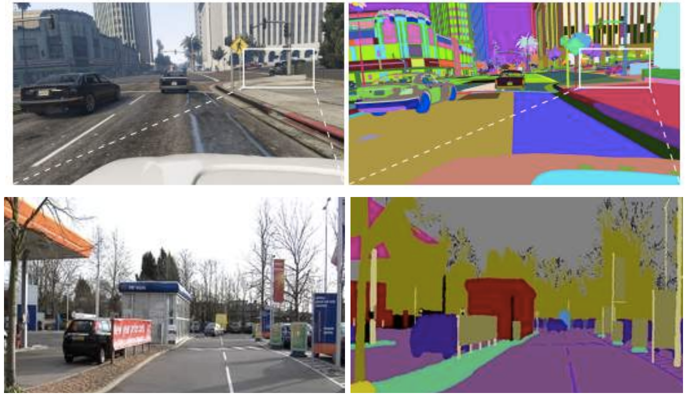
- Less expensive
- Faster / more scalable
- Less dangerous
- Easier to label

How can we learn useful real-world skills in the simulator?

Approach 1 – Use Realistic Simulated Data

Simulation

Real world



GTA V

Real world

Carefully match the simulation to the world [1,2,3,4]

Augment simulated data with real data [5,6]

[1] Stephen James, Edward Johns. *3d simulation for robot arm control with deep q-learning (2016)*

[2] Johns, Leutenegger, Davision. Deep learning a grasp function for grasping under gripper pose uncertainty (2016)

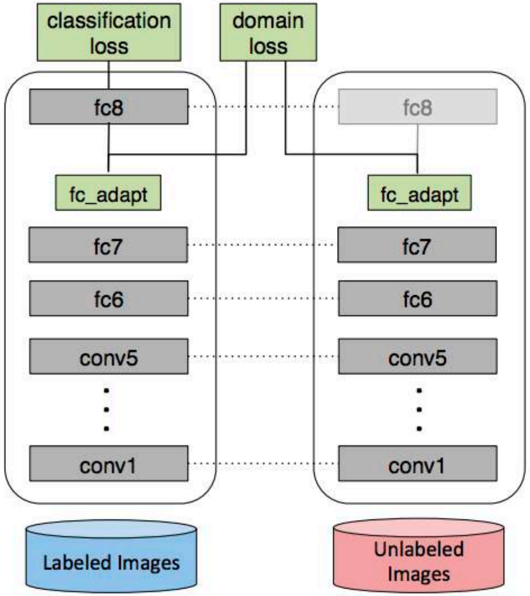
[3] Mahler et al, Dex-Net 3.0 (2017)

[4] Koenemann et al. Whole-body model-predictive control applied to the HRP-2 humanoid. (2015)

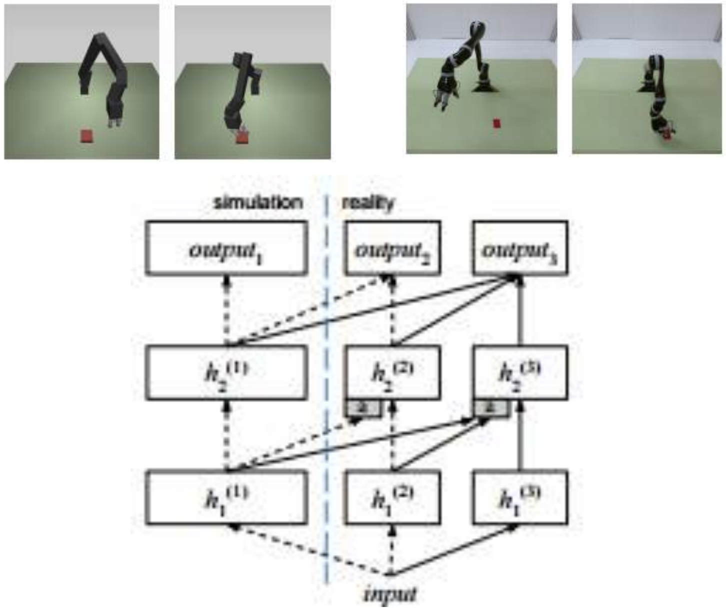
[5] Stephan R Richter, Vibhav Vineet, Stefan Roth, and Vladlen Koltun. *Playing for data: Ground truth from computer games (2016)*

[6] Bousmalis et al. Using simulation and domain adaptation to improve efficiency of robotic grasping (2017)

Approach 2 – Domain Confusion / Adaptation

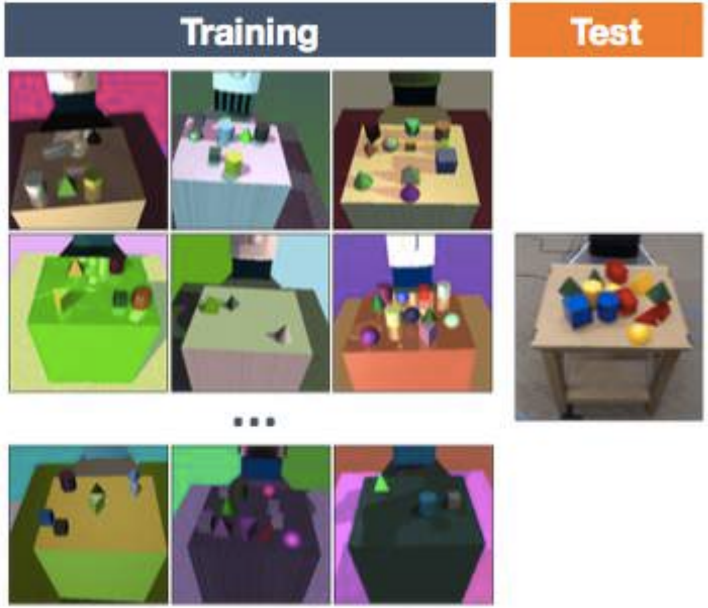


Eric Tzeng, Judy Hoffman, Ning Zhang, Kate Saenko, Trevor Darrell. Deep Domain Confusion: Maximizing for Domain Invariance. *arXiv preprint arXiv:1412.3474, 2014*.



Andrei A Rusu, Matej Vecerik, Thomas Rothö̈rl, Nicolas Heess, Razvan Pascanu, and Raia Hadsell. Sim-to-real robot learning from pixels with progressive nets. *arXiv preprint arXiv:1610.04286*, 2016.

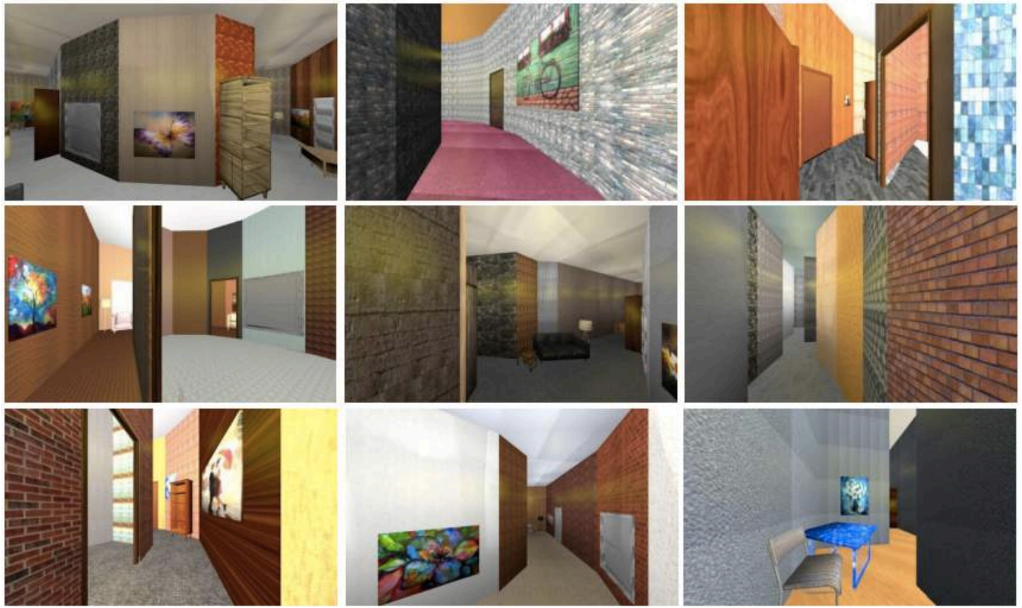
Approach 3 – Domain Randomization



If the model sees enough simulated variation, the real world may look like just the next simulator

Domain Randomization

(cad)ˆ2 rl: Real Single-Image Flight Without a Single Real Image.



- Quadcopter collision avoidance
- ~500 semi-realistic textures, 12 floor plans
- ~40-50% of 1000m trajectories are collision-free

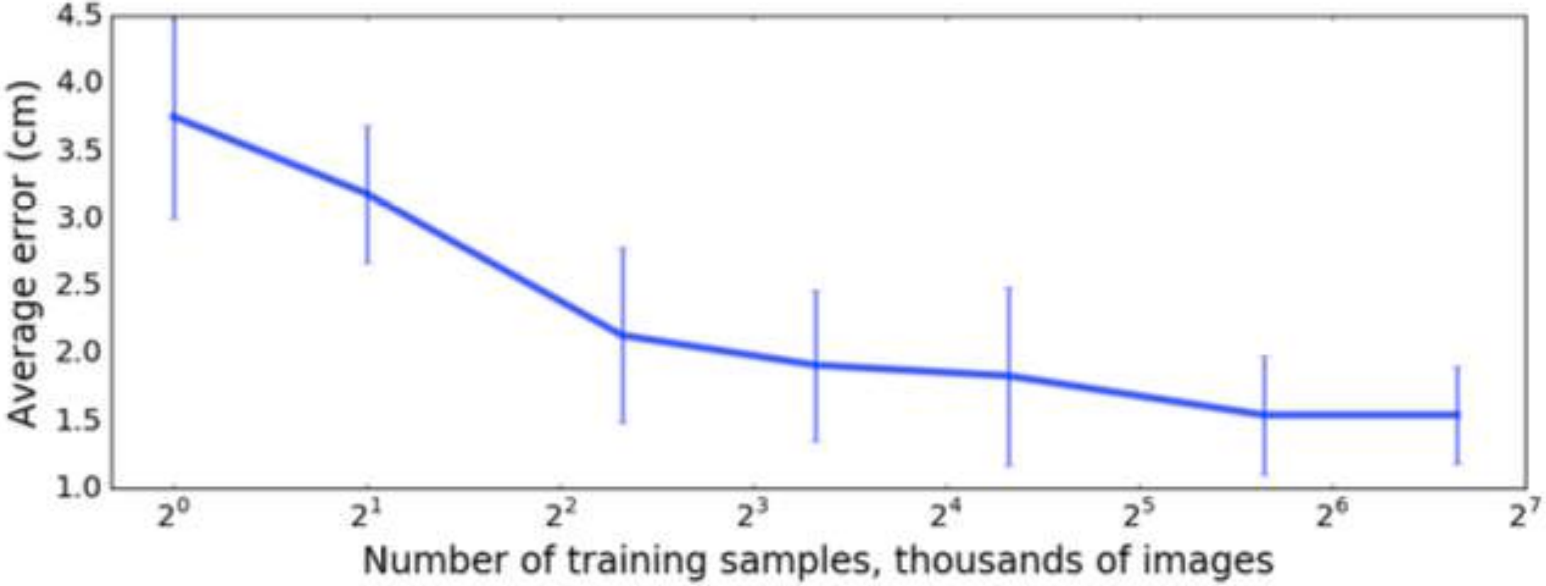
[3] Fereshteh Sadeghi and Sergey Levine. (cad)ˆ2 rl: Real single-image flight without a single real image. *arXiv preprint arXiv:1611.04201*, 2016.

Domain Randomization for Pose Estimation

- Precise object pose localization
- 100K images with simple randomly generated textures

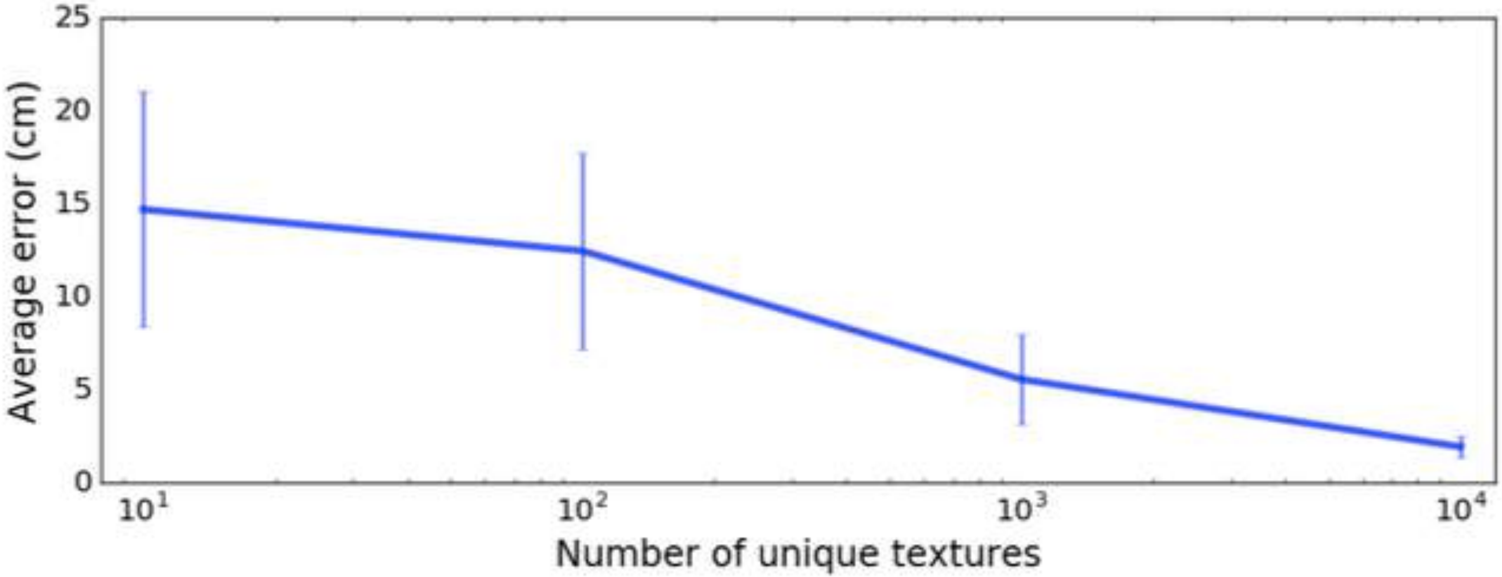
[Tobin, Fong, Ray, Schneider, Zaremba, Abbeel, 2017]

How does it work? More Data = Better



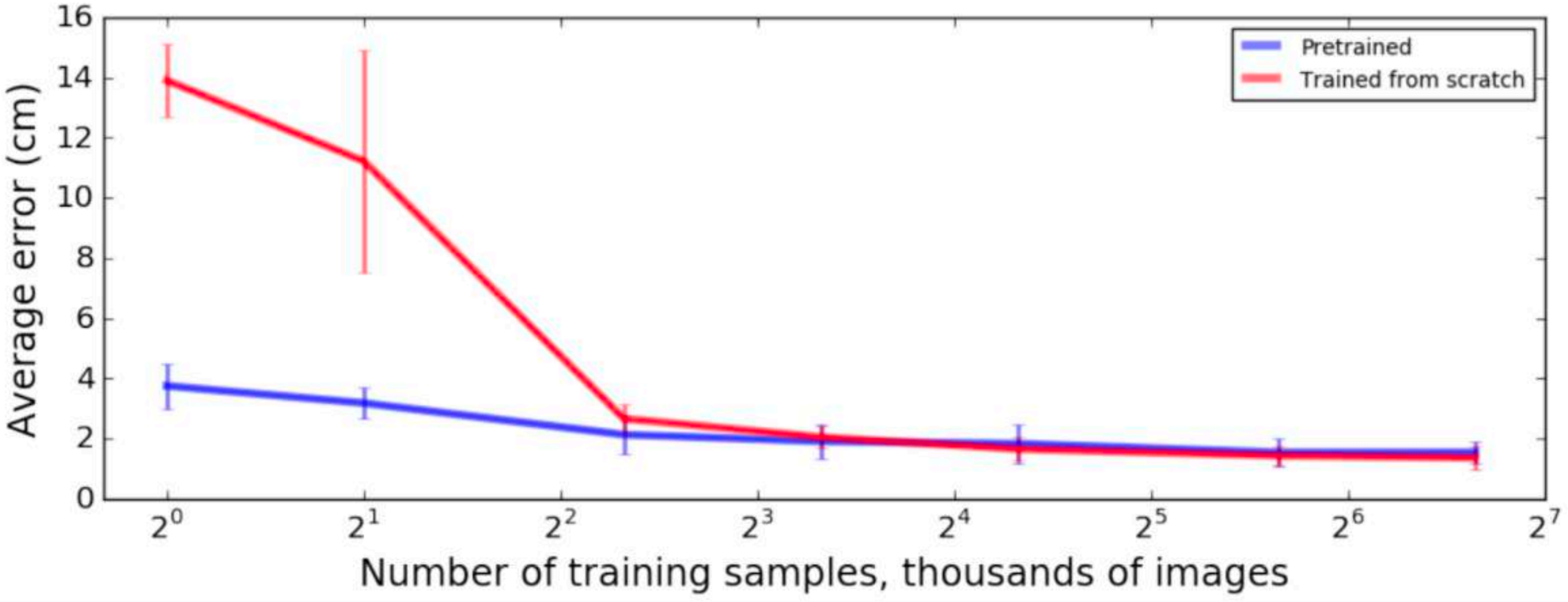
[Tobin, Fong, Ray, Schneider, Zaremba, Abbeel, 2017]

More Textures = Better



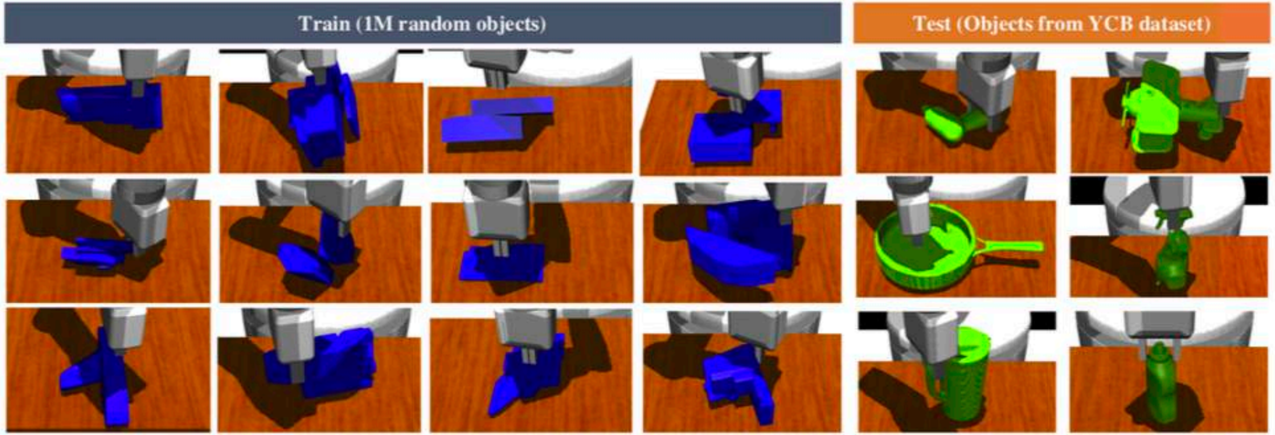
[Tobin, Fong, Ray, Schneider, Zaremba, Abbeel, 2017]

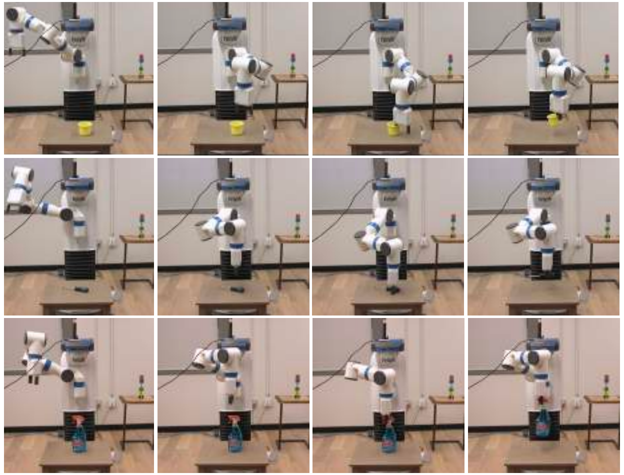
Pre-Training is not Necessary



[Tobin, Fong, Ray, Schneider, Zaremba, Abbeel, 2017]

Domain Randomization for Grasping





Hypothesis: Training on a diverse array of procedurally generated objects can produce comparable performance to training on realistic object meshes.

[Tobin, Zaremba, Abbeel, 2017] [see also: Levine et al and Goldberg et al]

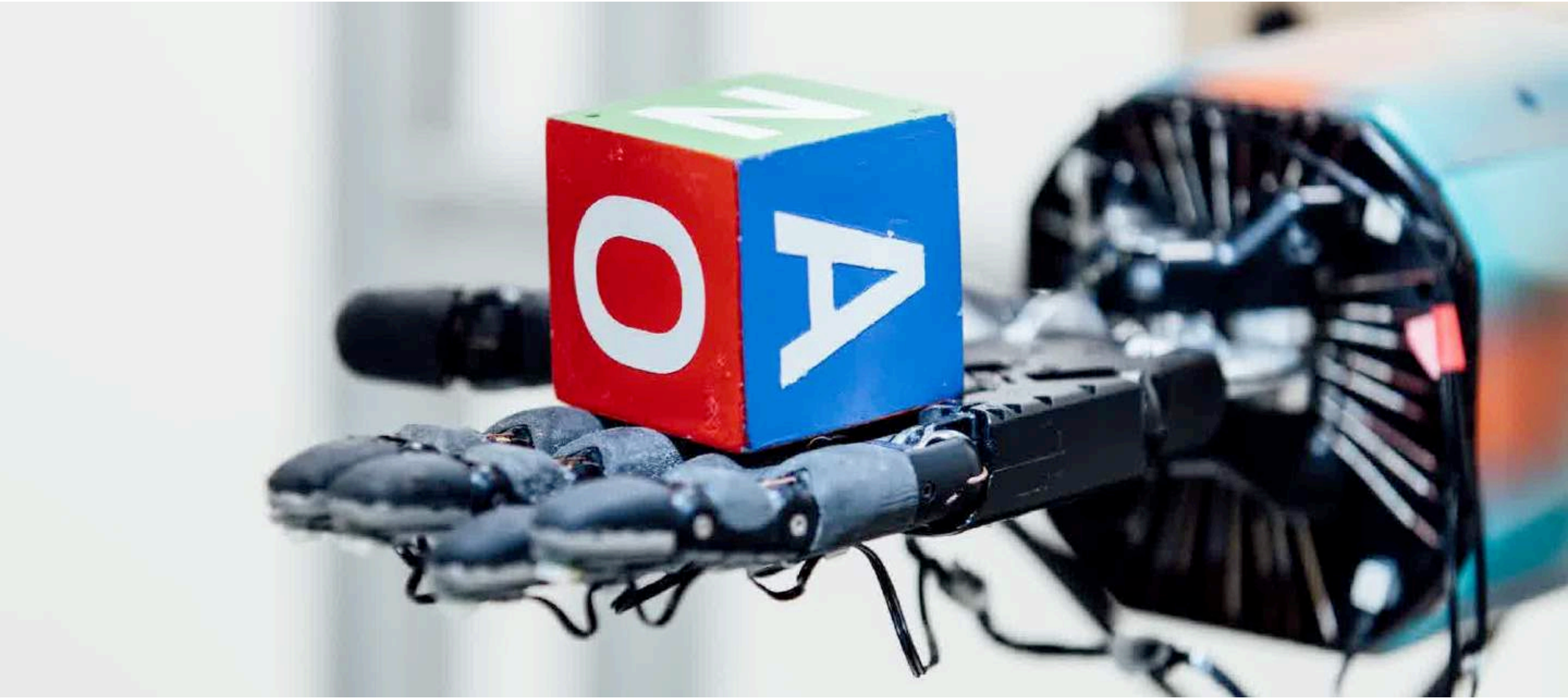
Evaluation: How Good Are Random Shapes?

Training set	ShapeNet Train	ShapeNet Test	Random Train	Random Test	Ycb
ShapeNet-1M	0.91	**0.91**	0.72	0.71	**0.93**
Random-1M	0.91	0.89	**0.86**	**0.84**	0.92
ShapeNet-Random-1M	**0.92**	0.90	0.84	0.81	0.92

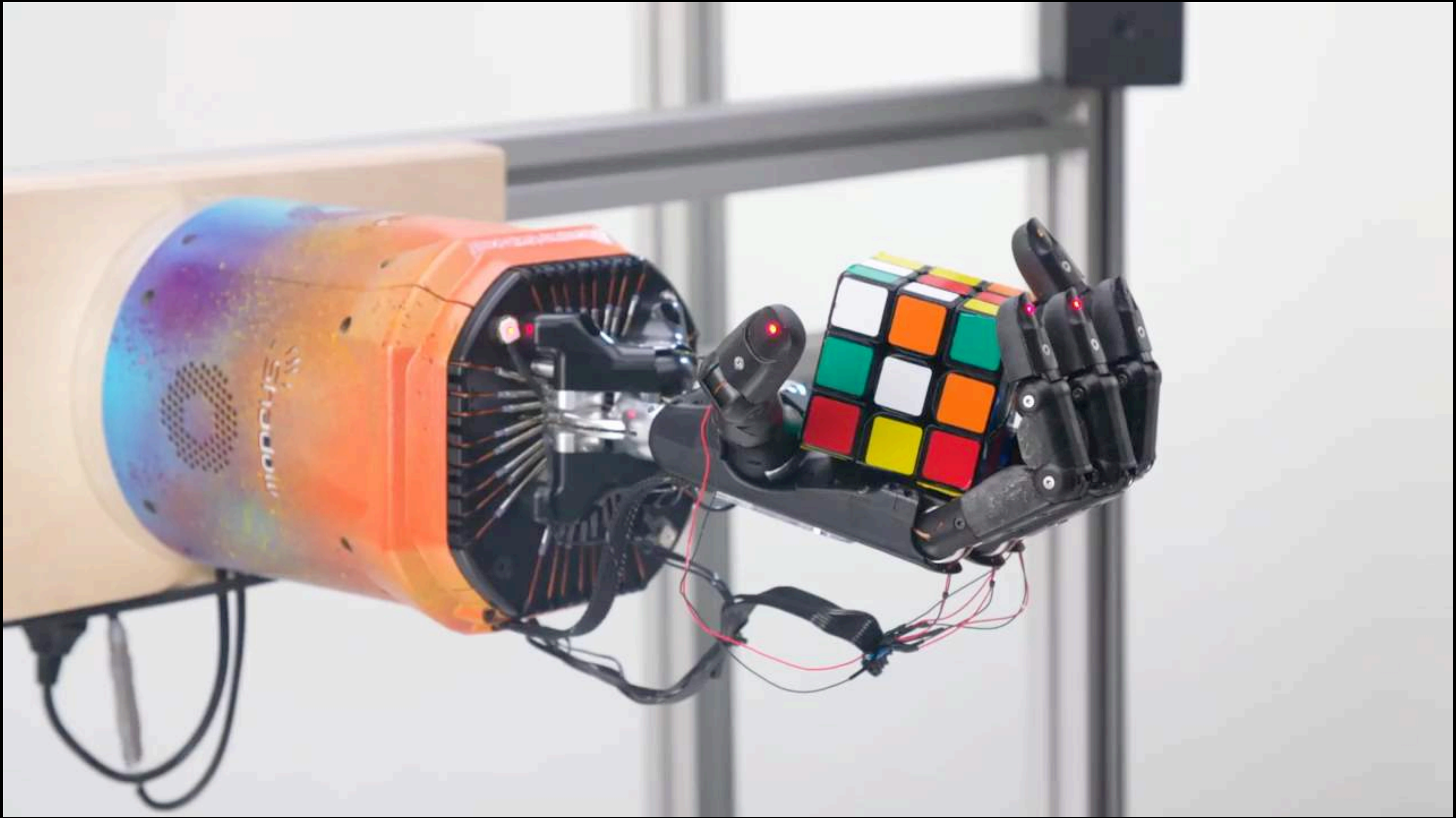
Fig. 5. Performance of the algorithm on different synthetic test sets. The full algorithm is able to achieve at least 90% success on previously unseen objects from the YCB dataset when trained on any of the three training sets.

[Tobin, Zaremba, Abbeel, 2017]

How a Full Hand?



[OpenAI, 2018]



[OpenAI, 2019]

Many Exciting Directions in AI

- Few-Shot Learning
- Reinforcement Learning
- Imitation Learning
- Domain Randomization
- ***Architecture Search***
- Unsupervised Learning
- Lifelong Learning
- Bias in ML (avoiding)
- Long Horizon Reasoning
- Safe Learning
- Value Alignment
- Planning + Learning
- …

Current Practice:

ML Solution =
Data + Computation + ML Expertise

[Slide from: Quoc Le]

Current Practice:

ML Solution =
Data + Computation + ML Expertise

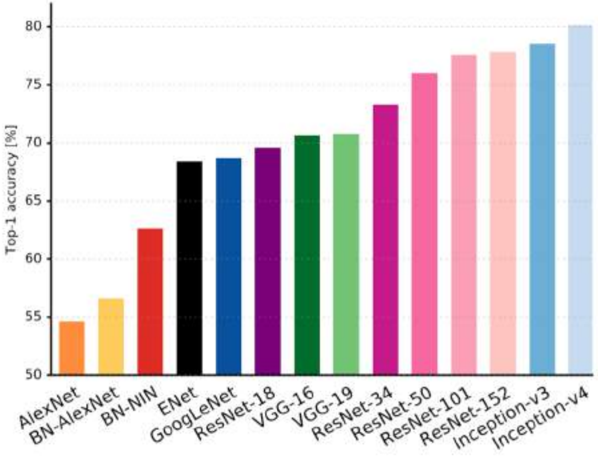
But can we turn this into:

Solution =
Data + 100X Computation

[Slide from: Quoc Le]

Importance of architectures for Vision

- Designing neural network architectures is hard
- Lots of human efforts go into tuning them
- Can we try and learn good architectures automatically?



Canziani et al, 2017

[Slide from: Quoc Le]

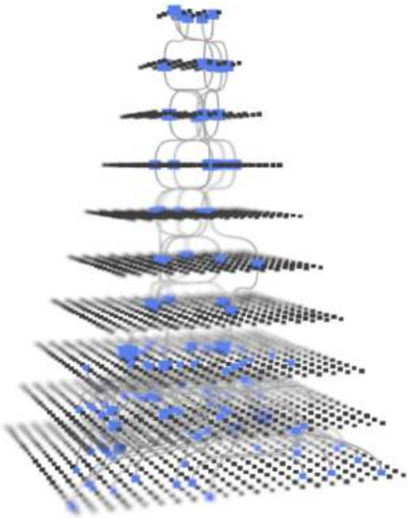
Neural Architecture Search

- We can specify the structure and connectivity of a neural network by a configuration string
  - ["Filter Width: 5", "Filter Height: 3", "Num Filters: 24"]
- Use a RNN ("Controller") to generate this string (the "Child Network")
- Train the Child Network to see how well it performs on a validation set
- Use reinforcement learning to update the Controller based on the accuracy of the Child Network

[Slide from: Quoc Le]

Neural Architecture Search

Controller: proposes Child Networks

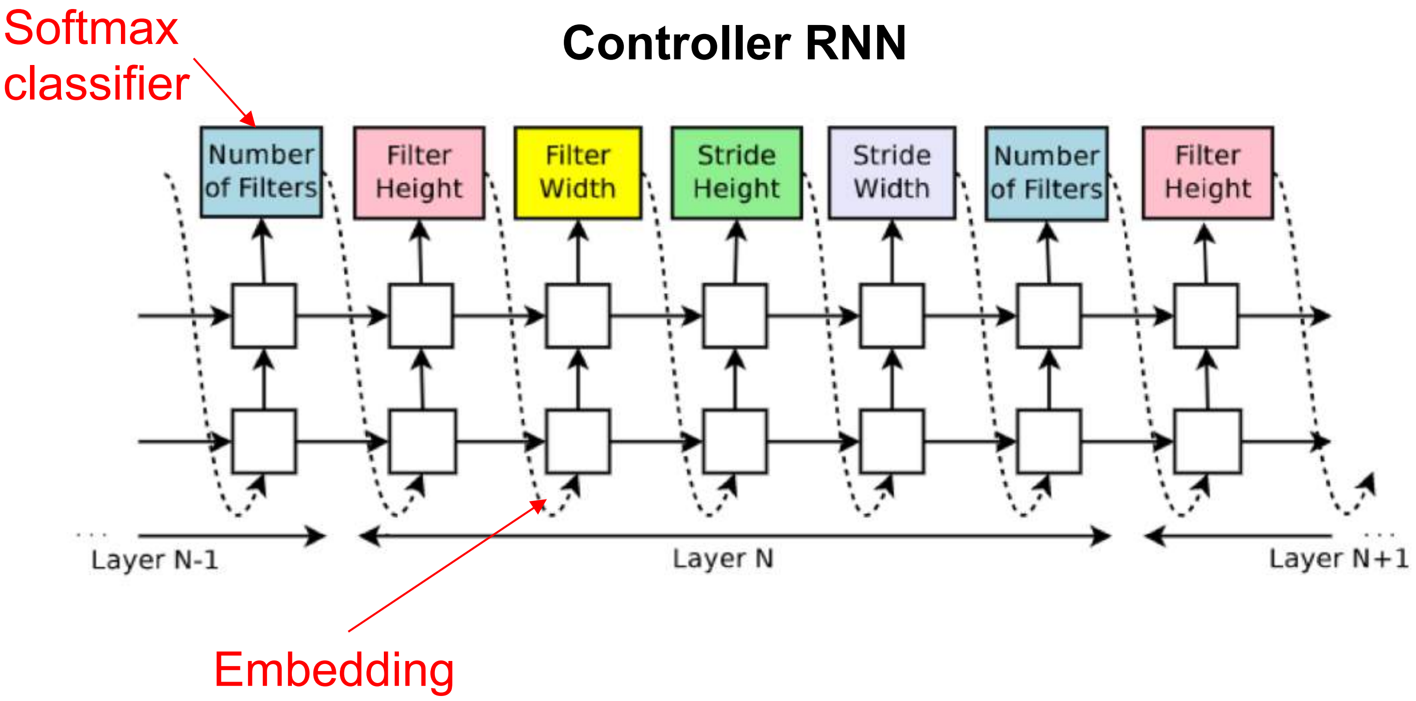


Iterate to find the most accurate Child Network

Train & evaluate Child Networks

[Slide from: Quoc Le]

Neural Architecture Search for Convolutional Networks



[Slide from: Quoc Le]

Training with REINFORCE

Parameters of Controller RNN

Accuracy of architecture on held-out dataset

$$J(\theta_c) = E_{P(a_{1:T};\theta_c)}[R]$$

Architecture predicted by the controller RNN viewed as a sequence of actions

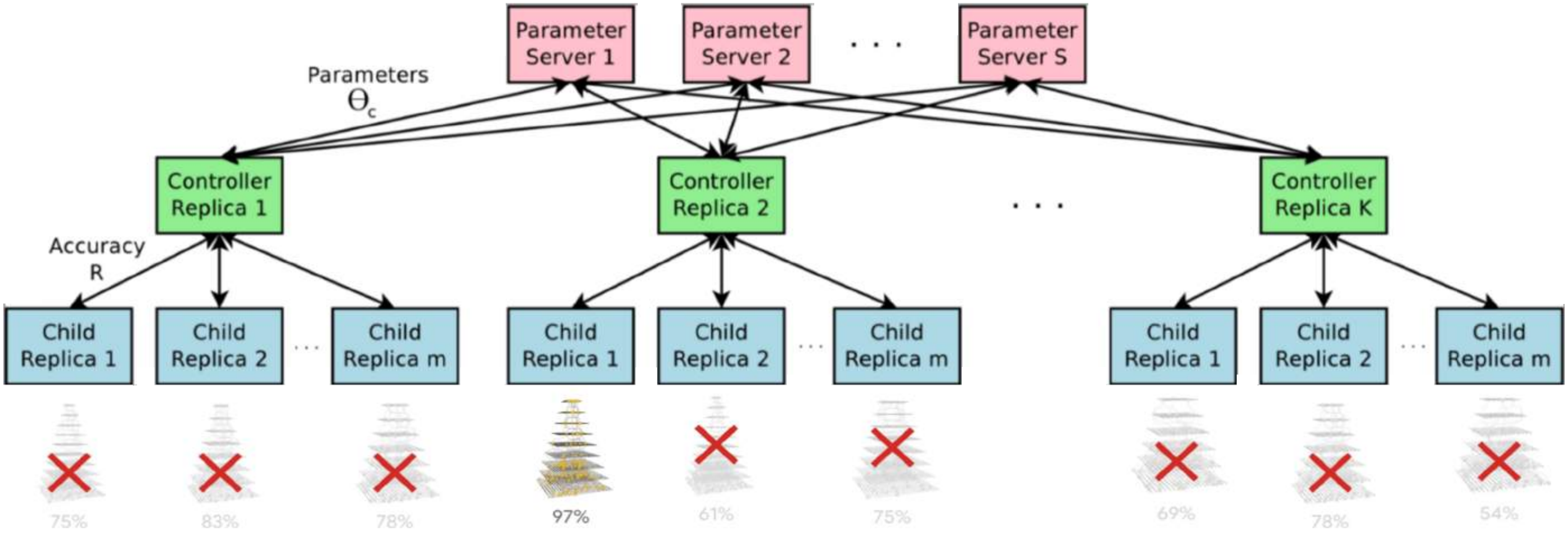
$$\nabla_{\theta_c} J(\theta_c) = \sum_{t=1}^{T} E_{P(a_{1:T};\theta_c)}\left[\nabla_{\theta_c} \log P(a_t|a_{(t-1):1};\theta_c)R\right]$$

$$\frac{1}{m}\sum_{k=1}^{m}\sum_{t=1}^{T}\nabla_{\theta_c} \log P(a_t|a_{(t-1):1};\theta_c)R_k$$

Number of models in minibatch

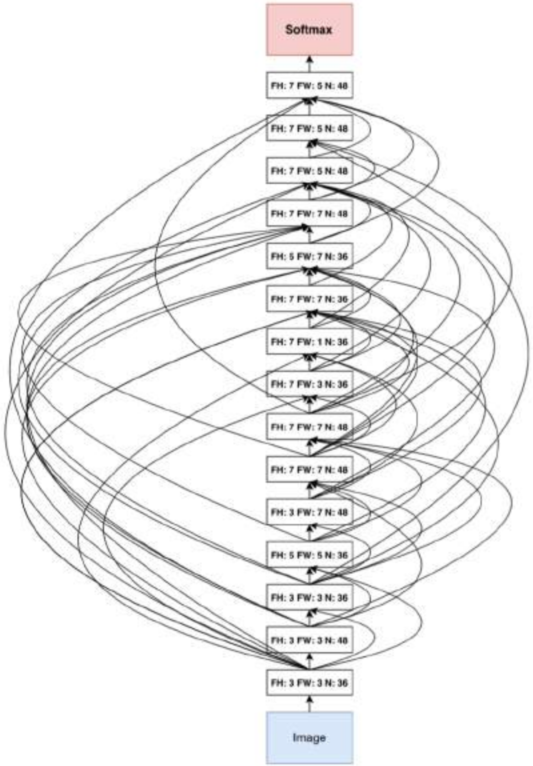
[Slide from: Quoc Le]

Distributed Training



[Slide from: Quoc Le]

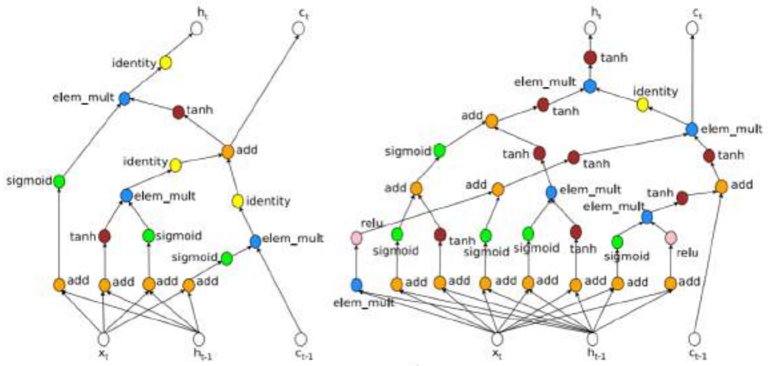
Neural Architecture Search for CIFAR-10



Model	Depth	Parameters	Error rate (%)
Network in Network (Lin et al., 2013)	-	-	8.81
All-CNN (Springenberg et al., 2014)	-	-	7.25
Deeply Supervised Net (Lee et al., 2015)	-	-	7.97
Highway Network (Srivastava et al., 2015)	-	-	7.72
Scalable Bayesian Optimization (Snoek et al., 2015)	-	-	6.37
FractalNet (Larsson et al., 2016)	21	38.6M	5.22
with Dropout/Drop-path	21	38.6M	4.60
ResNet (He et al., 2016a)	110	1.7M	6.61
ResNet (reported by Huang et al. (2016c))	110	1.7M	6.41
ResNet with Stochastic Depth (Huang et al., 2016c)	110	1.7M	5.23
	1202	10.2M	4.91
Wide ResNet (Zagoruyko & Komodakis, 2016)	16	11.0M	4.81
	28	36.5M	4.17
ResNet (pre-activation) (He et al., 2016b)	164	1.7M	5.46
	1001	10.2M	4.62
DenseNet ($L = 40, k = 12$) Huang et al. (2016a)	40	1.0M	5.24
DenseNet($L = 100, k = 12$) Huang et al. (2016a)	100	7.0M	4.10
DenseNet ($L = 100, k = 24$) Huang et al. (2016a)	100	27.2M	3.74
DenseNet-BC ($L = 100, k = 40$) Huang et al. (2016b)	190	25.6M	3.46
Neural Architecture Search v1 no stride or pooling	15	4.2M	5.50
Neural Architecture Search v2 predicting strides	20	2.5M	6.01
Neural Architecture Search v3 max pooling	39	7.1M	4.47
Neural Architecture Search v3 max pooling + more filters	39	37.4M	3.65

[Slide from: Quoc Le]

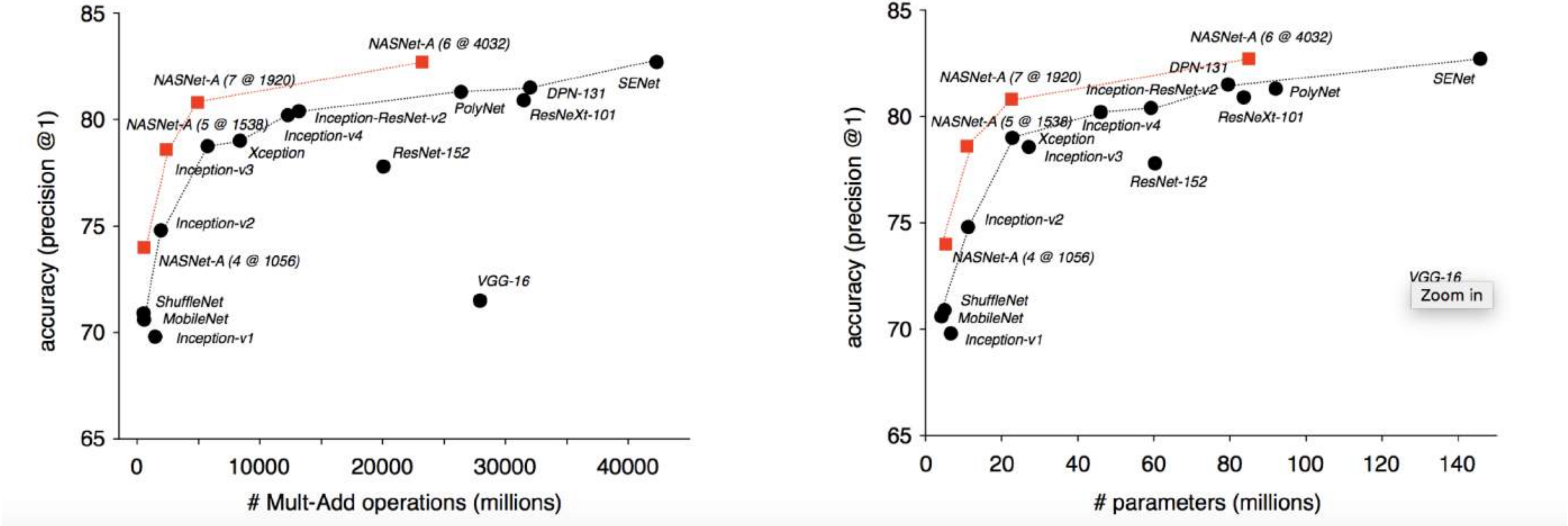
Neural Architecture Search for Language Modeling



Model	Parameters	Test Perplexity
Mikolov & Zweig (2012) - KN-5	2M‡	141.2
Mikolov & Zweig (2012) - KN5 + cache	2M‡	125.7
Mikolov & Zweig (2012) - RNN	6M‡	124.7
Mikolov & Zweig (2012) - RNN-LDA	7M‡	113.7
Mikolov & Zweig (2012) - RNN-LDA + KN-5 + cache	9M‡	92.0
Pascanu et al. (2013) - Deep RNN	6M	107.5
Cheng et al. (2014) - Sum-Prod Net	5M‡	100.0
Zaremba et al. (2014) - LSTM (medium)	20M	82.7
Zaremba et al. (2014) - LSTM (large)	66M	78.4
Gal (2015) - Variational LSTM (medium, untied)	20M	79.7
Gal (2015) - Variational LSTM (medium, untied, MC)	20M	78.6
Gal (2015) - Variational LSTM (large, untied)	66M	75.2
Gal (2015) - Variational LSTM (large, untied, MC)	66M	73.4
Kim et al. (2015) - CharCNN	19M	78.9
Press & Wolf (2016) - Variational LSTM, shared embeddings	51M	73.2
Merity et al. (2016) - Zoneout + Variational LSTM (medium)	20M	80.6
Merity et al. (2016) - Pointer Sentinel-LSTM (medium)	21M	70.9
Inan et al. (2016) - VD-LSTM + REAL (large)	51M	68.5
Zilly et al. (2016) - Variational RHN, shared embeddings	24M	66.0
Neural Architecture Search with base 8	32M	67.9
Neural Architecture Search with base 8 and shared embeddings	25M	64.0
Neural Architecture Search with base 8 and shared embeddings	54M	62.4

[Slide from: Quoc Le]

Performance of cell on ImageNet



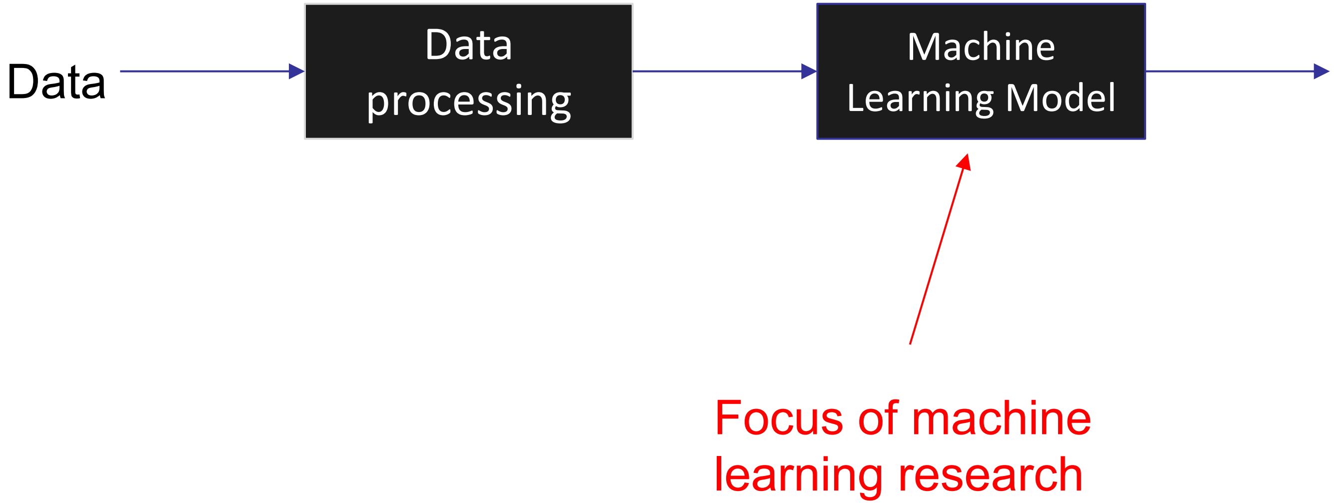
Updated June/2018: AmoebaNet (found by evolution in the same search space) is slightly better

[Slide from: Quoc Le]

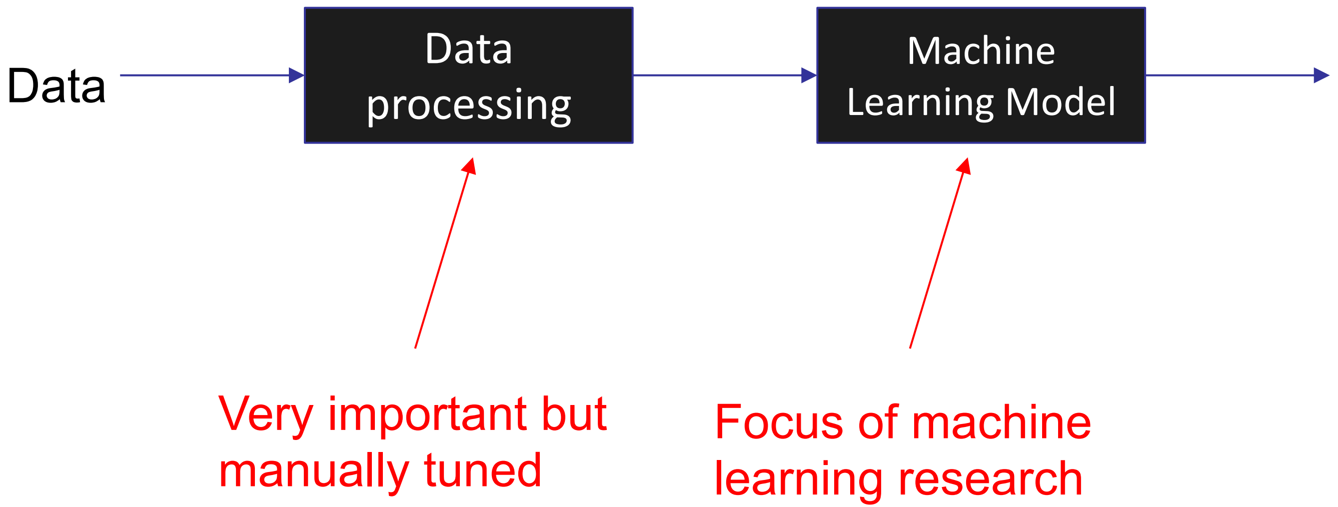
Neural Architecture Search

- Key idea: Architecture is a string
- Get an RNN to generate the string, and train the RNN with the reward associated with the string
- This idea can be applied to other components of ML:
  - Equation for an optimizer is a string
  - Activation function is a string, e.g., z = 1/(1 + e^(-x))

[Slide from: Quoc Le]

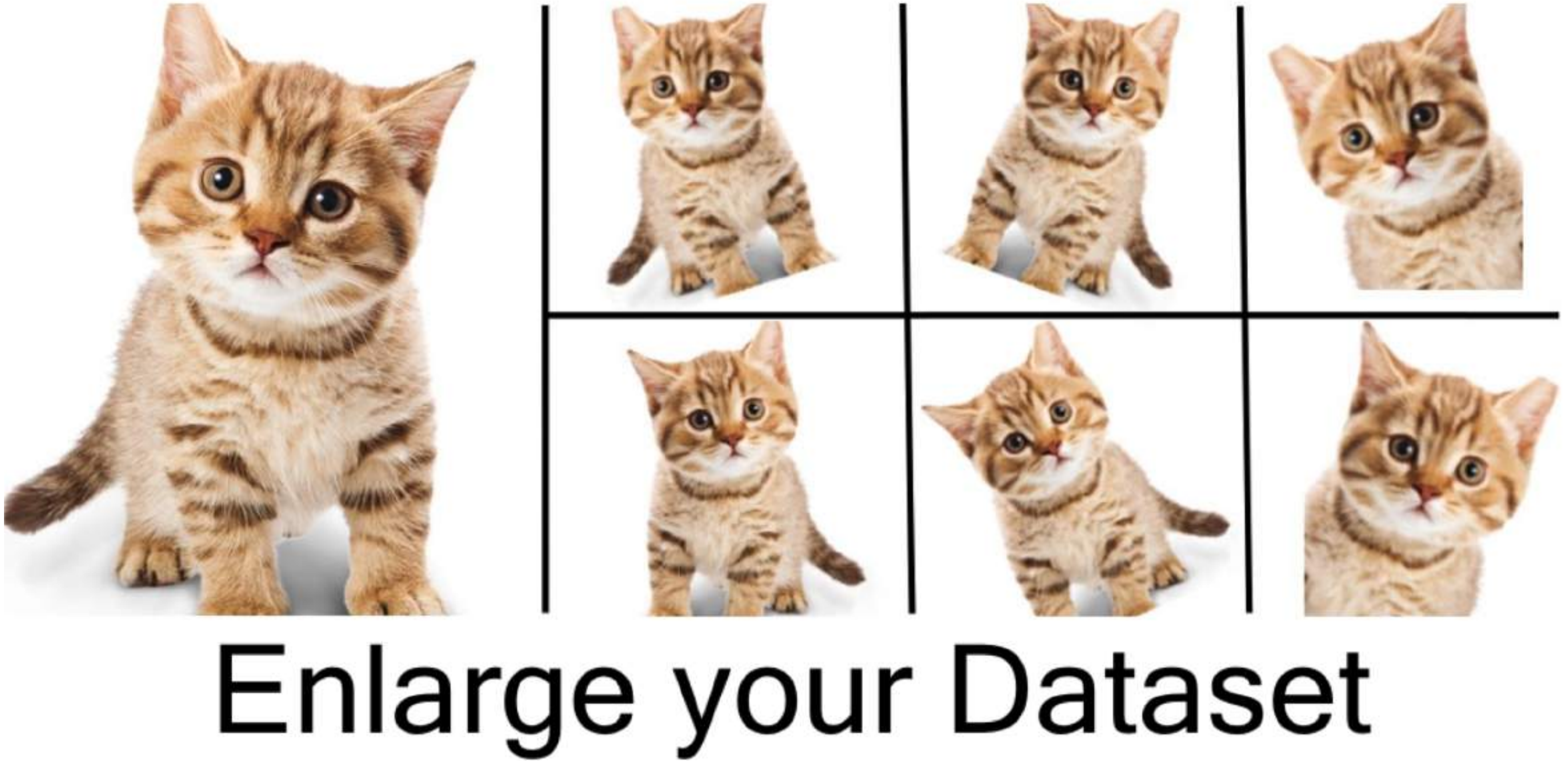


[Slide from: Quoc Le]



[Slide from: Quoc Le]

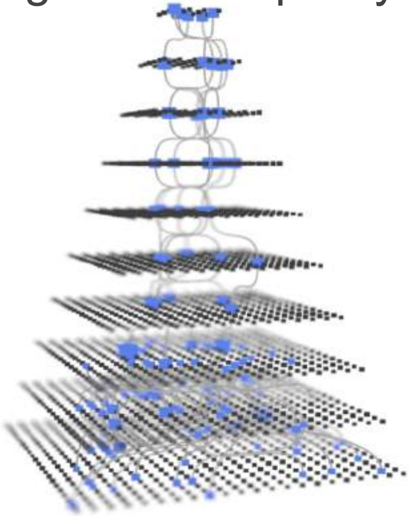
Data Augmentation



[Slide from: Quoc Le]

Data Augmentation

Controller: proposes augmentation policy



Iterate to find the most accurate policy

Train & evaluate models with the augmentation policy

[Slide from: Quoc Le]

AutoAugment: Example Policy

Original

Equalize, 0.4, 4
Rotate, 0.8, 8

Solarize, 0.6, 3
Equalize, 0.6, 7

Posterize, 0.8, 5
Equalize, 1.0, 2

Rotate, 0.2, 3
Solarize, 0.6, 8

Equalize, 0.6, 8
Posterize, 0.4, 6

Probability of applying

Magnitude

[Slide from: Quoc Le]

AutoAugment

Full CIFAR-10

Model	No data aug	Standard data-aug	AutoAugment
Wide-ResNet-28-10	3.87	3.08	2.68
Shake-Shake (26 2x32d)	3.55	3.02	2.47
Shake-Shake (26 2x96d)	2.86	2.56	1.99
Shake-Shake (26 2x112d)	2.82	2.57	1.89
AmoebaNet-B (6,128)	2.98	2.13	1.75
PyramidNet+ShakeDrop	2.67	2.31	**1.48**

CIFAR-100

Model	No data aug	Standard data-aug	AutoAugment
Wide-ResNet-28-10	18.80	18.41	17.09
Shake-Shake (26 2x96d)	17.05	16.00	14.28
PyramidNet+ShakeDrop	13.99	12.19	**10.67**

[Slide from: Quoc Le]

ImageNet - Top5 error rate

Model	No data augmentation	Standard data augmentation	AutoAugment
ResNet-50	7.80	6.92	6.18
ResNet-200		5.85	4.99
AmoebaNet-B		3.97	3.78
AmoebaNet-C		3.90	**3.52**

Human (Andrej Karpathy)	5.1
Our best model	3.5

[Slide from: Quoc Le]

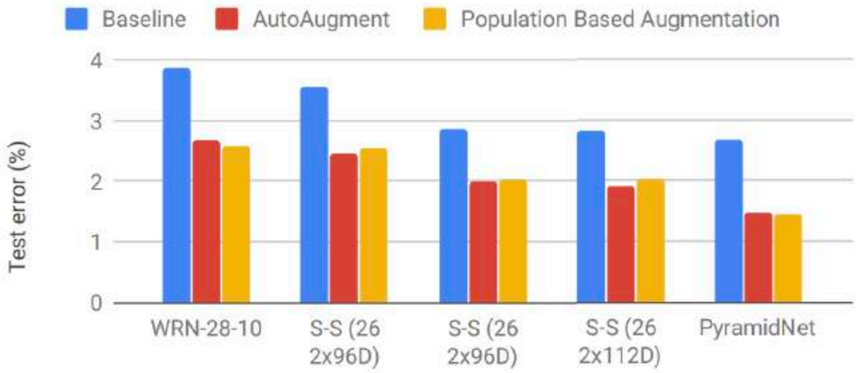
References

- *Neural Architecture Search with Reinforcement Learning*. Barret Zoph and Quoc V. Le. ICLR, 2017
- *Learning Transferable Architectures for Scalable Image Recognition*. Barret Zoph, Vijay Vasudevan, Jonathon Shlens, Quoc V. Le. CVPR, 2018
- *Efficient Neural Architecture Search via Parameter Sharing*. Hieu Pham, Melody Y. Guan, Barret Zoph, Quoc V. Le, Jeff Dean
- *AutoAugment: Learning Augmentation Policies from Data*. Ekin D. Cubuk, Barret Zoph, Dandelion Mane, Vijay Vasudevan, Quoc V. Le. Arxiv, 2018
- *Searching for Activation Functions*. Prajit Ramachandran, Barret Zoph, Quoc Le. ICLR Workshop, 2018

[Slide from: Quoc Le]

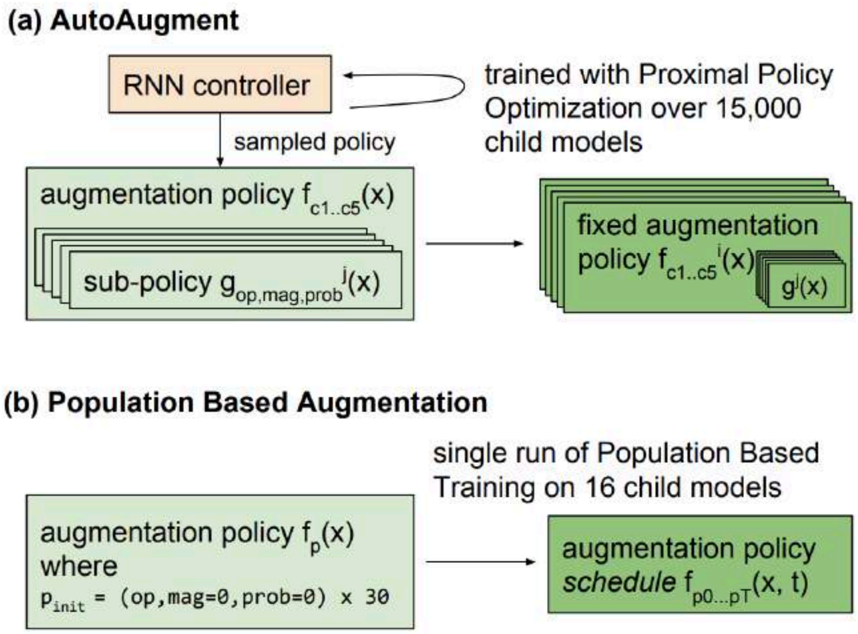
Population-based Auto-Augment aka Auto-Augment with 1000x less compute

Dataset	Value	Previous Best	AA	PBA
CIFAR-10	GPU Hours	-	5000	5
	Test Error	2.1	1.48	1.46
CIFAR-100	GPU Hours	-	0*	0*
	Test Error	12.2	10.7	10.9
SVHN	GPU Hours	-	1000	1
	Test Error	1.3	1.0	1.1



[Daniel Ho, Eric Liang, Ion Stoica, Pieter Abbeel, Xi (Peter) Chen, ICML 2019]

Population-based Auto-Augment aka Auto-Augment with 1000x less compute



[Daniel Ho, Eric Liang, Ion Stoica, Pieter Abbeel, Xi (Peter) Chen, ICML 2019]

Many Exciting Directions in AI

- Few-Shot Learning
- Reinforcement Learning
- Imitation Learning
- Domain Randomization
- Architecture Search
- ***Unsupervised Learning***
- Lifelong Learning
- Bias in ML (avoiding)
- Long Horizon Reasoning
- Safe Learning
- Value Alignment
- Planning + Learning
- …

Unsupervised Learning Rationale

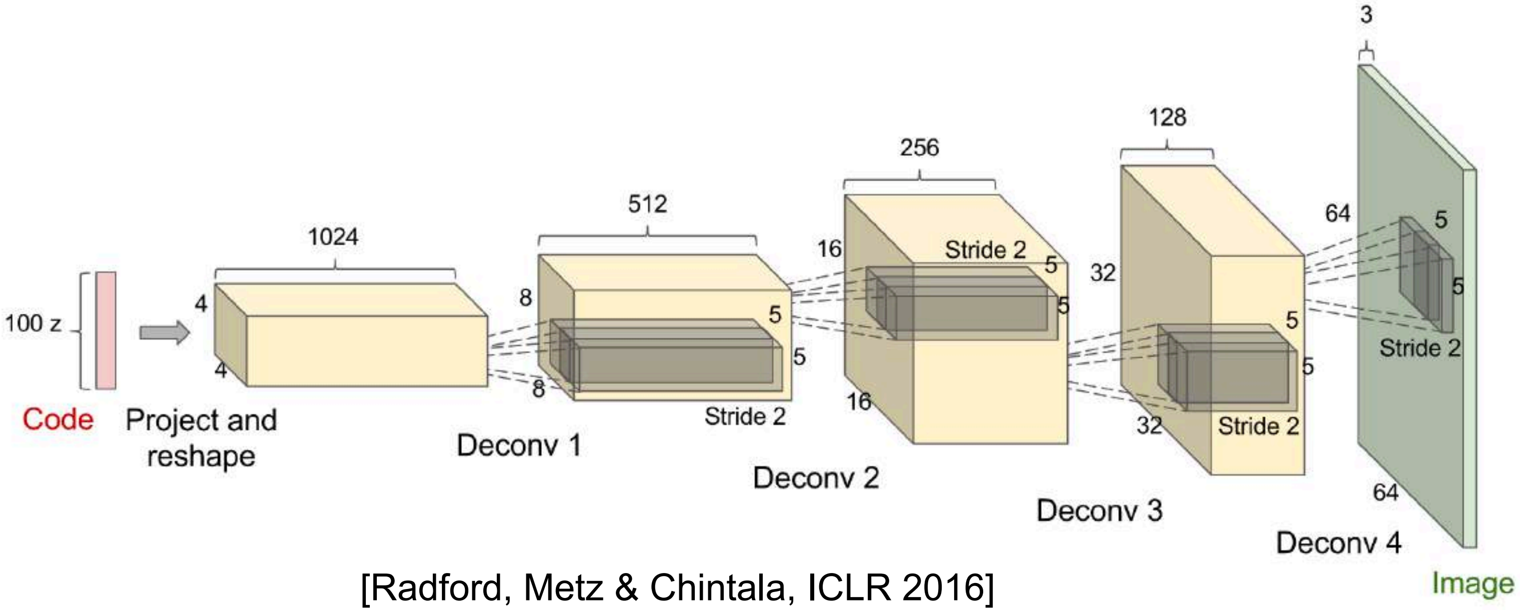
- Supervised learning is limited by amount of labeled data
  - Can we use unlabeled data instead?
- Main ideas:
  - Learn network that embeds the data
    - Later supervised learning on embedding
    - (possibly while also fine-tuning the embedding network)
  - Learning weights:
    - Pretraining of NN
    - Auxiliary loss

Main Families of Models

- Variational AutoEncoders (VAEs)
- Generative Adversarial Networks
- Exact Likelihood Models (autoregressive [pixelcnn, charrnn], realnvp, NICE)
- “puzzles” (gray2rgb, where-does-this-patch-go, rotation-prediction, contrastive-predictive-coding, etc…)

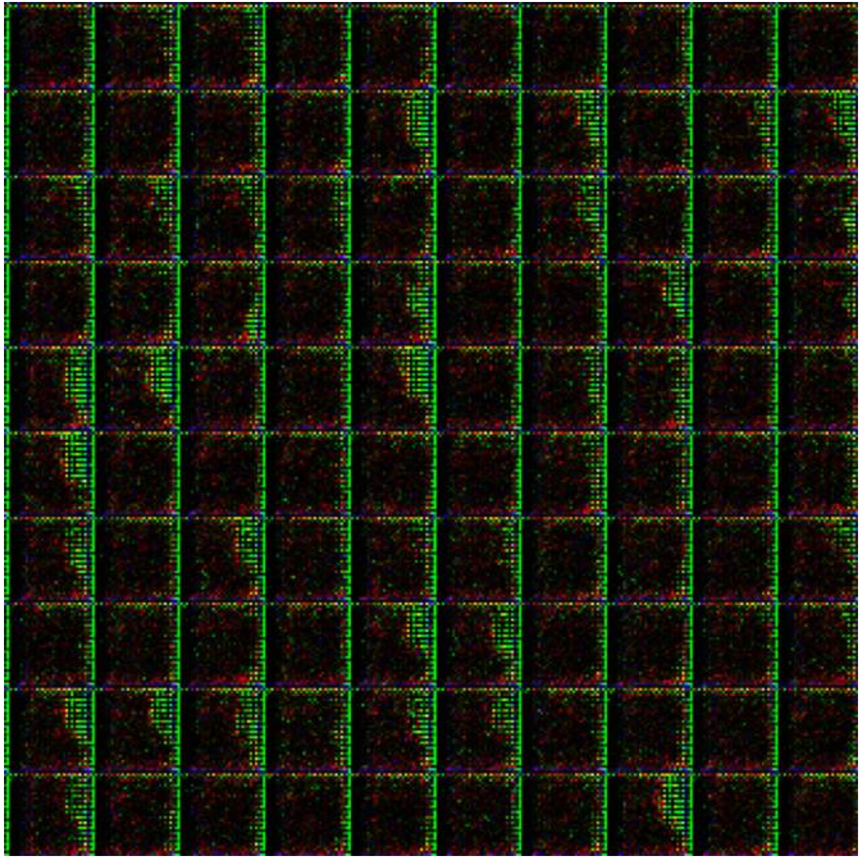
Generative Models

- *“What I cannot create, I do not understand.”*
  - Ability to generate data *that look real* entails some form of understanding



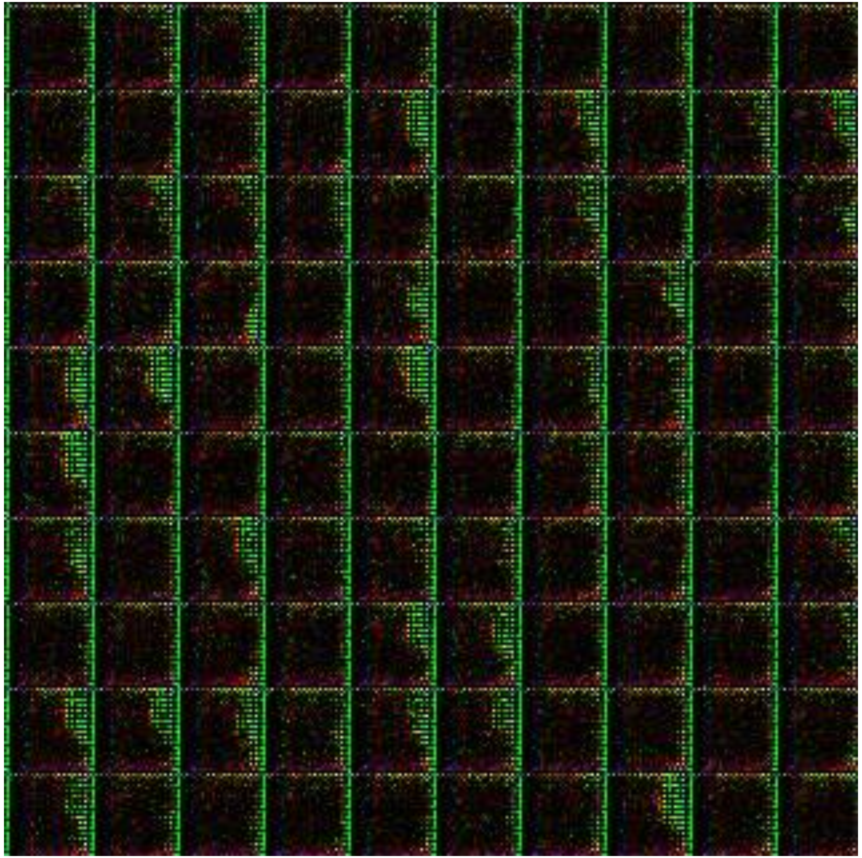
[Radford, Metz & Chintala, ICLR 2016]

Initial "Images"



[Salimans, Goodfellow, Zaremba, Cheung, Radford & Chen, NIPS 2016]

Learning to Generate Images



[Salimans, Goodfellow, Zaremba, Cheung, Radford & Chen, NIPS 2016]

InfoGAN: Concepts Learning

(a) Azimuth (pose)

(b) Elevation

(c) Lighting

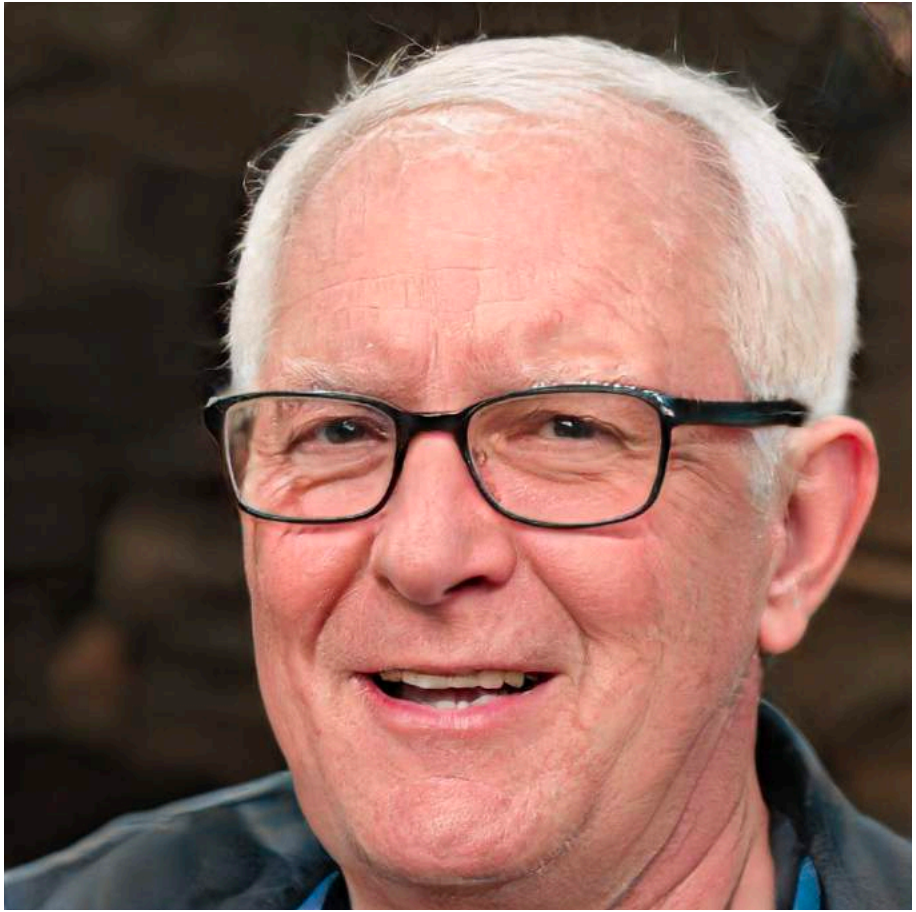
(d) Wide or Narrow

[Chen, Duan, Houthooft, Schulman, Sutskever & Abbeel, NIPS 2016]

Image Generation

[thispersondoesnotexist.com]

Image Generation



[thispersondoesnotexist.com]

Image Generation

[thispersondoesnotexist.com]

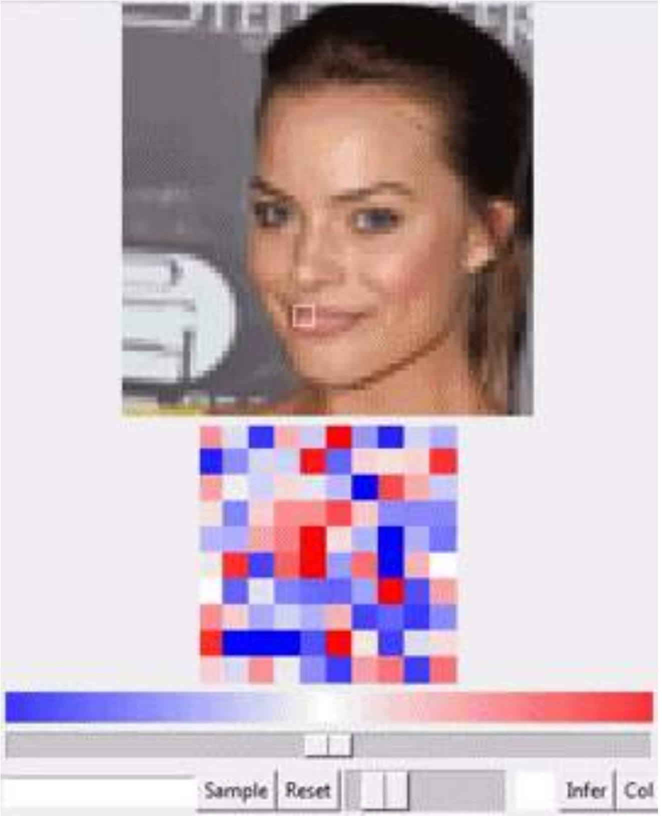
Unsupervised Image to Image

[CycleGAN: Zhu, Park, Isola & Efros, 2017]

Everybody Dance Now

[Everybody Dance Now: Caroline Chan, Shiry Ginosar, Tinghui Zhou, Alyosha Efros (UC Berkeley) 2018]

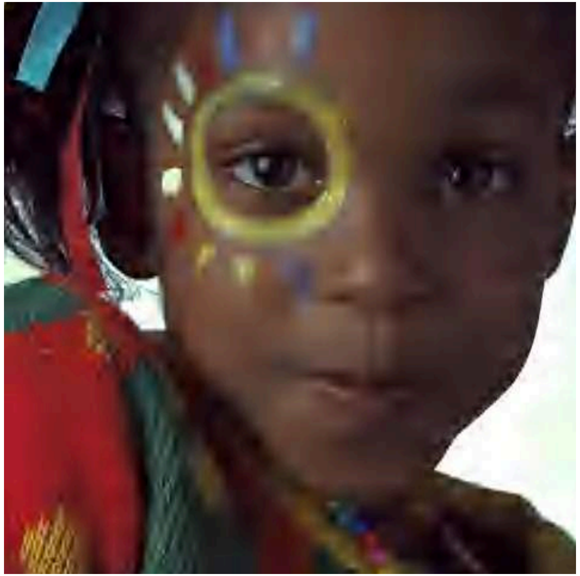
Neural “Photoshop”



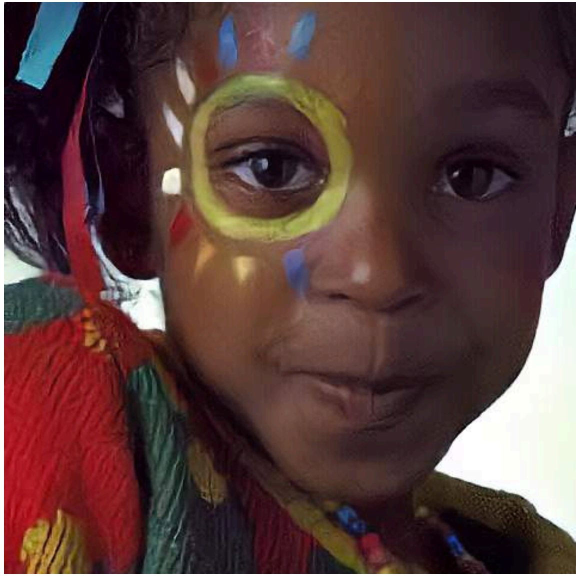
[Brock, Lim, Ritchie & Weston, 2016]

Neural Compression 2

JPEG



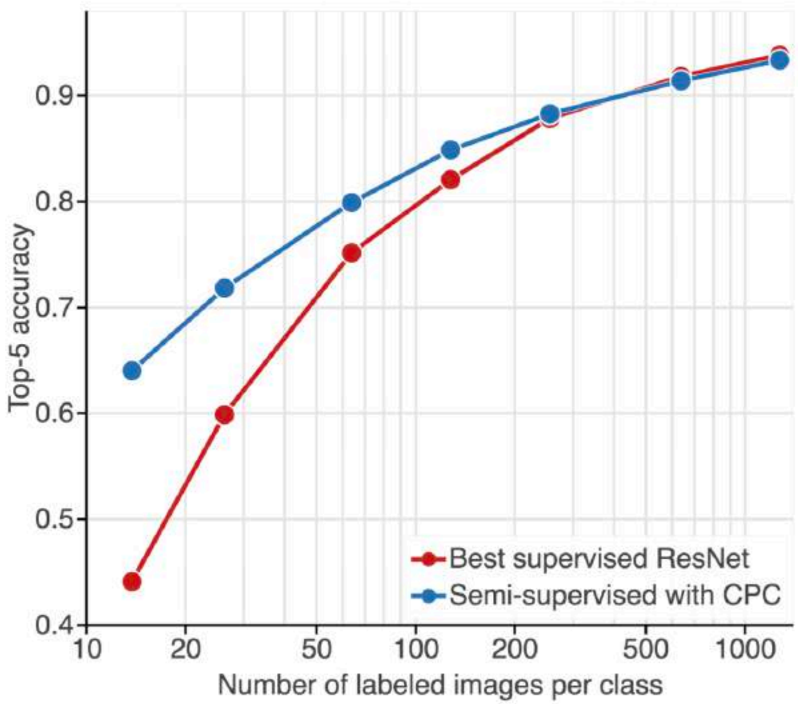
JPEG2000



WaveOne

[Rippel & Bourdev, 2017]

Data-Efficient Image Recognition with Contrastive Predictive Coding



[Olivier Henaff, Ali Razavi, Carl Doersch, Ali Eslami, Aaron van den Oord, 2019]

Data-Efficient Image Recognition with Contrastive Predictive Coding

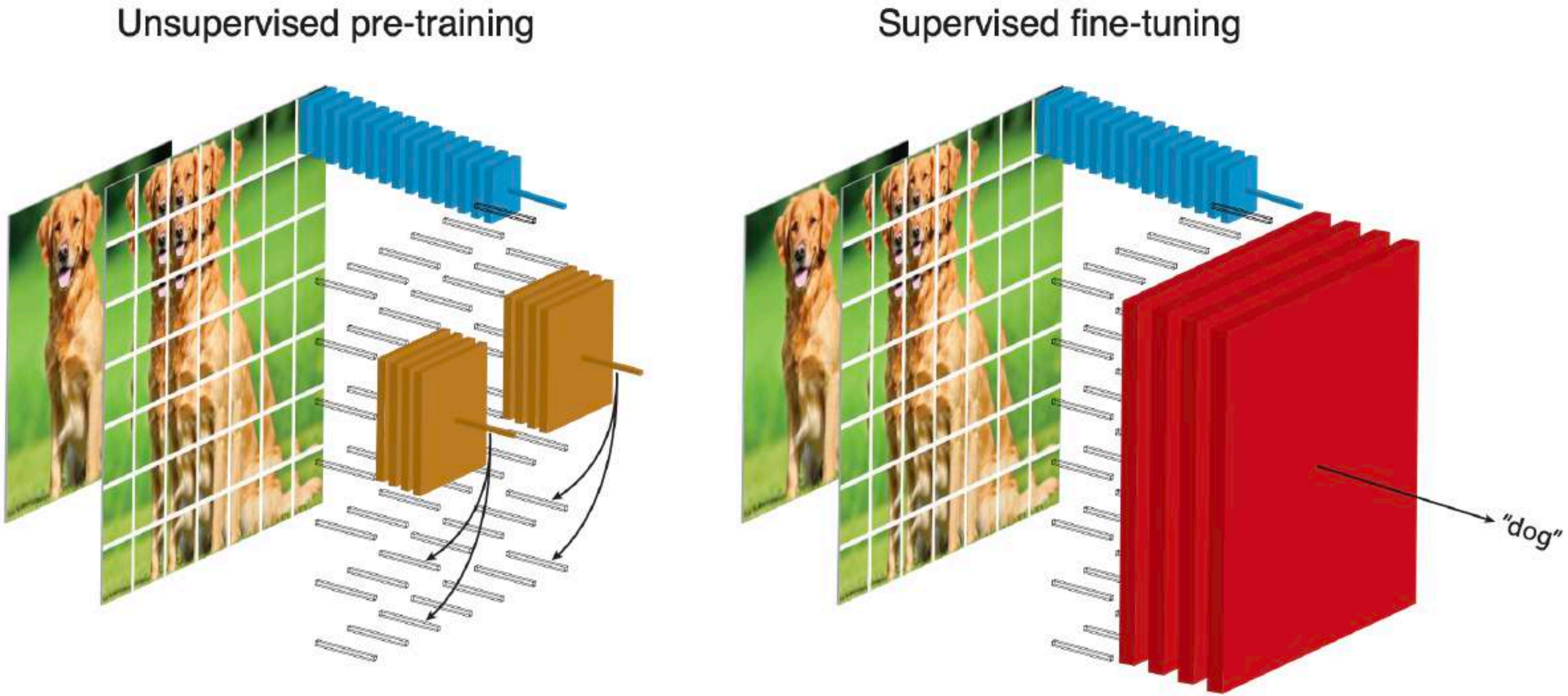
Linear separability in feature space from unsupervised learning

Method	Top-1	Top-5
Motion Segmentation (MS) [50]	27.6	48.3
Exemplar (Ex) [17]	31.5	53.1
Relative Position (RP) [14]	36.2	59.2
Colorization (Col) [69]	39.6	62.5
Combination of		
MS + Ex + RP + Col [15]	-	69.3
CPC [49]	48.7	73.6
Rotation + RevNet [36]	55.4	-
CPC (ours)	**61.0**	**83.0**

[Olivier Henaff, Ali Razavi, Carl Doersch, Ali Eslami, Aaron van den Oord, 2019]

Data-Efficient Image Recognition with Contrastive Predictive Coding

Contrastive-Predictive Coding



[Olivier Henaff, Ali Razavi, Carl Doersch, Ali Eslami, Aaron van den Oord, 2019]

Beyond Images: Amazon Review

This product does what it is supposed to. I always keep three of these in my kitchen just in case ever I need a replacement cord.

Great little item. Hard to put on the crib without some kind of embellishment. My guess is just like the screw kind of attachment I had.

[Radford et al, 2017]

Text Generation (OpenAI's GPT-2)

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

GPT-2: *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 Generation (OpenAI's GPT-2)

Prompt: Recycling is good for the world, no, you could not be more wrong.

GPT-2: Recycling is NOT good for the world. It is bad for the environment, it is bad for our health, and it is bad for our economy. I'm not kidding. Recycling is not good for the environment. It is destructive to the earth and it is a major contributor to global warming. Recycling is not good for our health. It contributes to obesity and diseases like heart disease and cancer. Recycling is bad for our economy. It increases the cost of a product, and in turn, the price of everything that is made with that product. Recycling is not good for our nation. We pay a tremendous price for the privilege of having the world's most advanced and efficient recycling system. Recycling is a huge, colossal waste of time, energy, money, and resources.

Text Generati… ’s GPT-2)

SYSTEM PROMPT (HUMAN-WRITTEN)

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

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

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.

…e unicorns may have originated in …s 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]

Unsupervised Sentiment Neuron

This is one of Crichton's best books. The characters of Karen Ross, Peter Elliot, Munro, and Amy are beautifully developed and their interactions are exciting, complex, and fast-paced throughout this impressive novel. And about 99.8 percent of that got lost in the film. Seriously, the screenplay AND the directing were horrendous and clearly done by people who could not fathom what was good about the novel. I can't fault the actors because frankly, they never had a chance to make this turkey live up to Crichton's original work. I know good novels, especially those with a science fiction edge, are hard to bring to the screen in a way that lives up to the original. But this may be the absolute worst disparity in quality between novel and screen adaptation ever. The book is really, really good. The movie is just dreadful.

[Radford et al, 2017]

Benchmarks

TASK **Common Sense Reasoning**: resolution of an ambiguous pronoun

DATASET Winograd Schema Challenge

EXAMPLES *The trophy doesn't fit into the brown suitcase because it is too large.*

Correct answer: *it = trophy*
Model answer: `it = trophy`

The trophy doesn't fit into the brown suitcase because it is too small.

Correct answer: *it = suitcase*
Model answer: `it = suitcase`

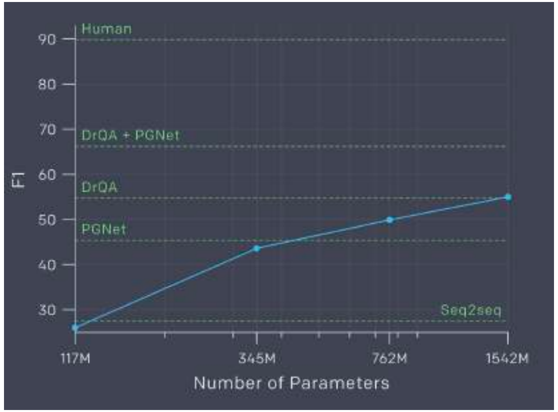
Benchmarks

TASK	**Language Modeling of Broad Contexts**: predict the last word of a passage
DATASET	LAMBADA
EXAMPLE	*Both its sun-speckled shade and the cool grass beneath were a welcome respite after the stifling kitchen, and I was glad to relax against the tree's rough, brittle bark and begin my breakfast of buttery, toasted bread and fresh fruit. Even the water was tasty, it was so clean and cold. It almost made up for the lack of...*
	Correct answer: *coffee* **Model answer**: food

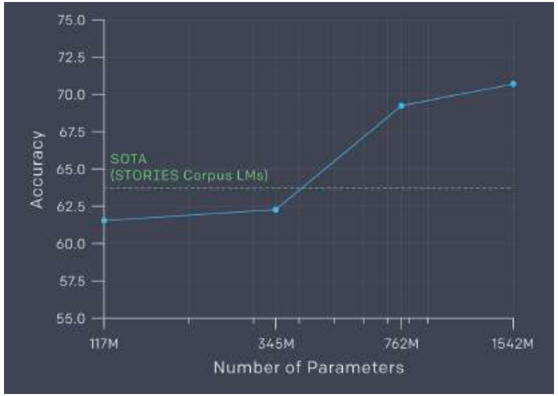
Benchmarks

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

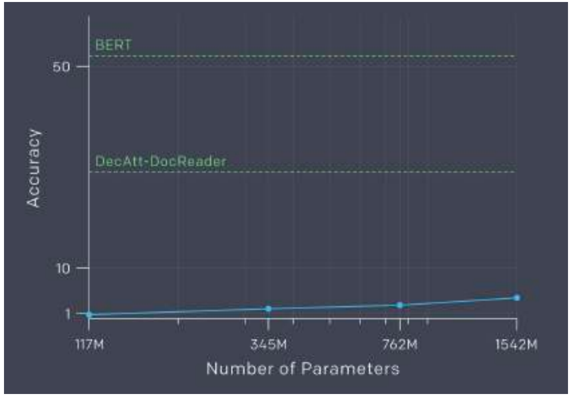
Scaling



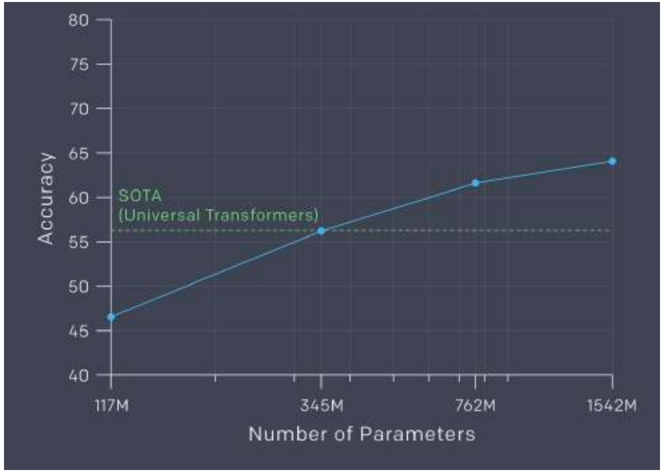
Human
DrQA + PGNet
DrQA
PGNet
Seq2seq
F1
90
80
70
60
50
40
30
117M
345M
762M
1542M
Number of Parameters



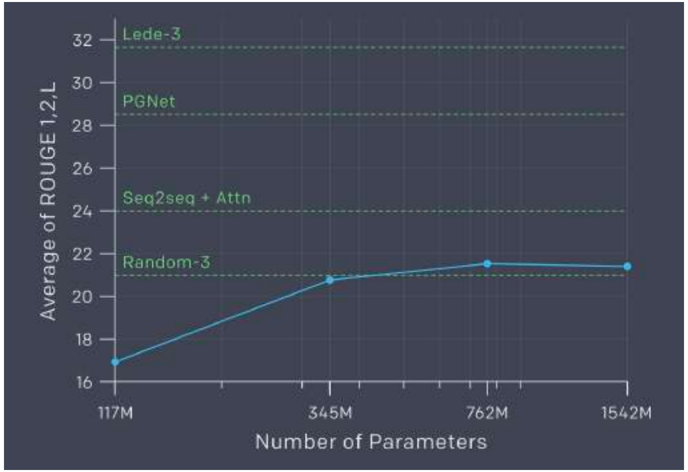
75.0
72.5
70.0
67.5
65.0
62.5
60.0
57.5
55.0
Accuracy
SOTA
(STORIES Corpus LMs)
117M
345M
762M
1542M
Number of Parameters



BERT
DecAtt-DocReader
50
10
1
Accuracy
117M
345M
762M
1542M
Number of Parameters



80
75
70
65
60
55
50
45
40
Accuracy
SOTA
(Universal Transformers)
117M
345M
762M
1542M
Number of Parameters



Lede-3
PGNet
Seq2seq + Attn
Random-3
32
30
28
26
24
22
20
18
16
Average of ROUGE 1,2,L
117M
345M
762M
1542M
Number of Parameters

Many Exciting Directions in AI

- Few-Shot Learning
- Reinforcement Learning
- Imitation Learning
- Domain Randomization
- Architecture Search
- Unsupervised Learning

- Lifelong Learning
- Bias in ML (avoiding)
- Long Horizon Reasoning
- Safe Learning
- Value Alignment
- Planning + Learning
- ...

Many Exciting Directions in AI

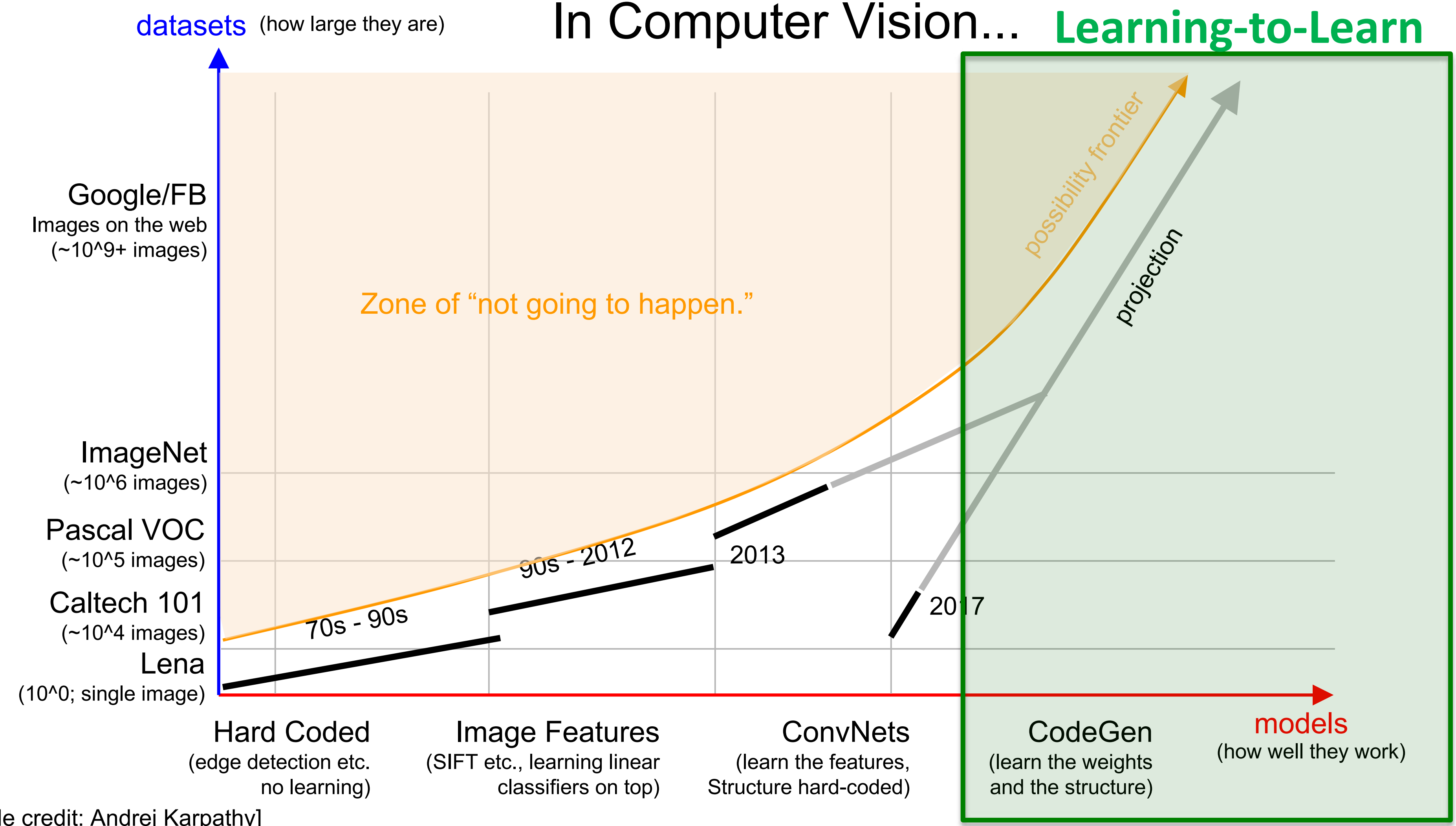
- Few-Shot Learning
- Reinforcement Learning
- Imitation Learning
- Domain Randomization
- Architecture Search
- Unsupervised Learning

- Lifelong Learning
- Bias in ML (avoiding)
- Long Horizon Reasoning
- Safe Learning
- Value Alignment
- Planning + Learning
- …

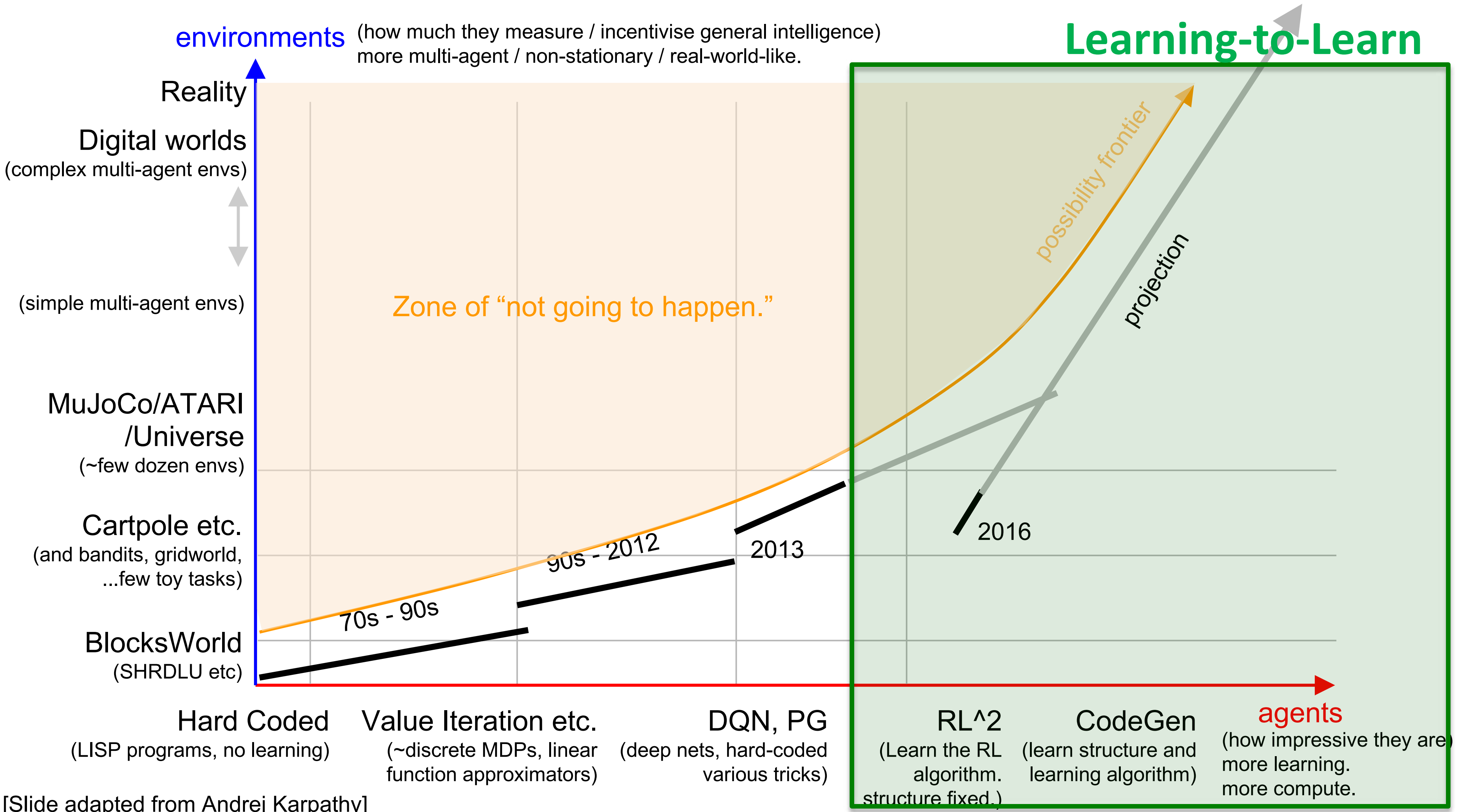
Outline

- Sampling of research directions
- ***Overall research theme***
- Research <> Real-World gap
- How to keep up

In Computer Vision... Learning-to-Learn



[Slide credit: Andrej Karpathy]



[Slide adapted from Andrej Karpathy]

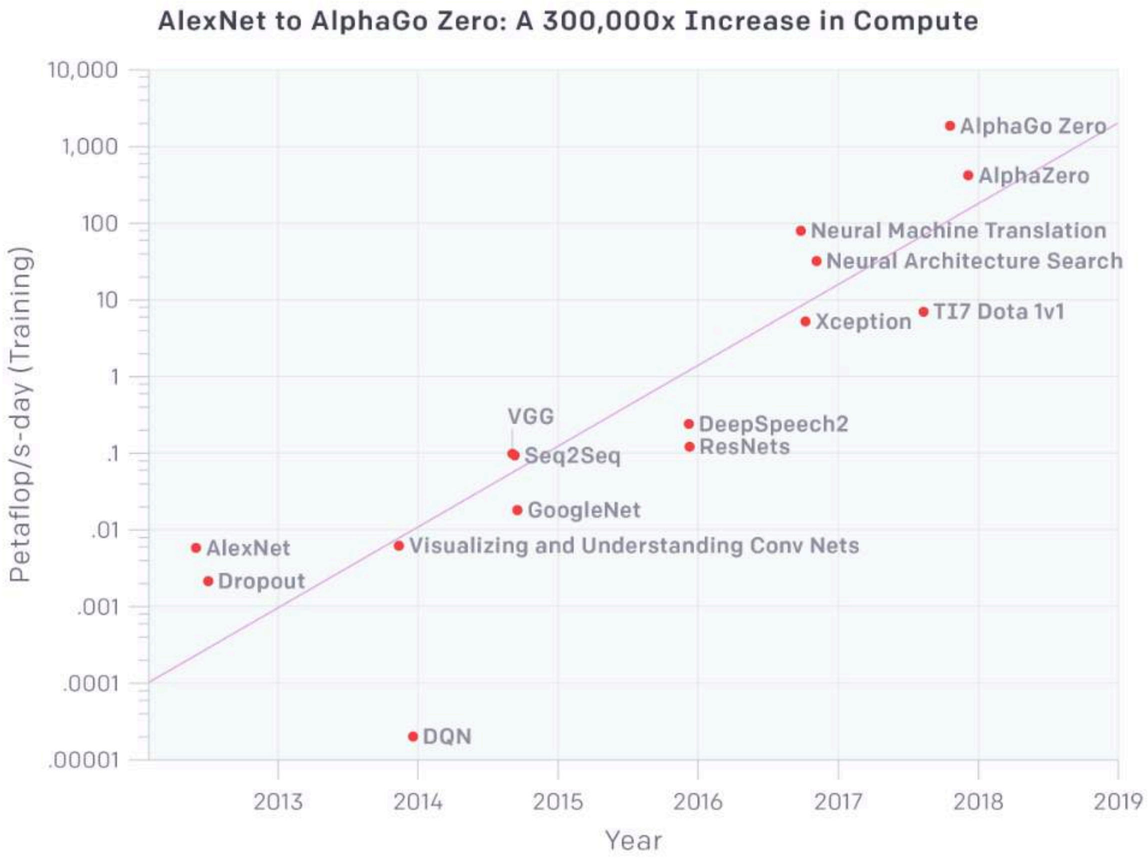
Compute Increasing Rapidly

Companies Developing Deep Learning Chips

	Company	HQ	Story
Public	Ambarella	United States	Developing computer vision chips for autonomous cars
	AMD	United States	GPU based deep learning
	Facebook	United States	Forming a team to build SoCs and perhaps inference chips
	Google	United States	Custom designed TPU deployed in Google Cloud
	Intel	United States	Developing a Neural Network chip based on Nervana acquisition
	Nvidia	United States	Current market leader using GPU based deep learning
	Tesla	United States	Developing a custom AI chip for autonomous driving
Private	Bitmain	China	Top maker of Bitcoin mining chips
	Cambricon	China	China's state-backed startup with $1B+ valuation.
	Cerebras Systems	United States	Ex-AMD team backed by Benchmark Capital
	DeePhi	China	China based startup with a focus on video analysis
	GraphCore	United Kingdom	Building a 16nm deep learning chip for training and inference
	Groq	United States	Ex-Google TPU team backed by Social Capital
	Horizon Robotics	China	Ex-Baidu team. Embedded / computer vision focus
	KnuEdge	United States	Headed by former NASA CTO
	Mythic	United States	In-memory inference for IoT backed by DFJ
	Tenstorrent	Canada	Toronto based chip startup
	Thinci	United States	Computer vision / auto focus
	Wave Computing	United States	Makes data flow acceleration servers

Source: ARK Invest
[20-April-2018]

Compute per Major Result



Source: OpenAI

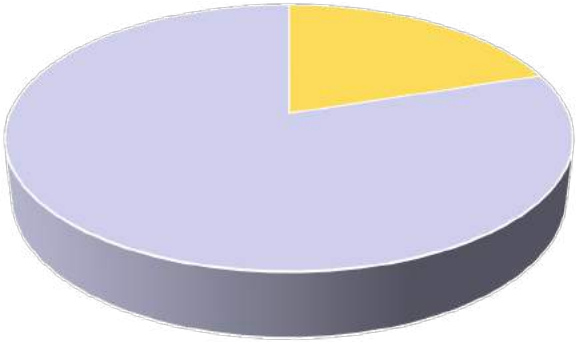
Implications for research agenda?

Key Enablers for AI Progress?

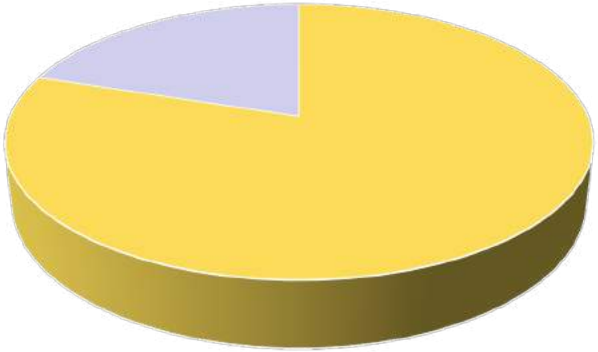
DATA

COMPUTE

HUMAN INGENUITY



Problem Territory 1



Problem Territory 2

E.g., Deep Learning to Learn

How much compute do we use at Berkeley Robot Learning Lab?

- Each PhD student approximately uses:
  - 4-8 GPU machine (local development)
  - 2-5K / month cloud compute credits (big thanks to Google and Amazon!)
- Trending:
  - Towards 10-20 GPUs per student (housed at central campus facility)
  - Hoping to get to 10-20K / month cloud compute credits (anyone?)

Outline

- Sampling of research directions
- Overall research theme
- ***Research <> Real-World gap***
- How to keep up

Research <<<<GAP>>>> Real-World

- **Research:**
  - 0 to 1
  - If achieving 70% on a benchmark, move on to the next one
- **Real-World:**
  - Even 90% is not enough!

Real-World Reliability Requirements

- Typical station in factory or warehouse:
  - 500 - 2,000 task completions per hour
- Now, imagine 90% success of the robot
  - 500 per hour → 50 failures that need fixing up
  - 2000 per hour → 200 failures that need fixing up
  - Fixing up failures typically takes more time than doing task
    - --> robot with 50 failures per hour causes more work than it saves
- Realistically, value created at 1-2 human interventions per hour
  - 500 per hour, max 2 interventions → 99.6% success rate
  - 2,000 per hour, max 2 interventions --> 99.9% success rate

Ok, fine, we need beyond 90%

So what, let’s just:

- larger network
- more data

Subtleties in going beyond 90%

- can NOT ignore the long tail of real world scenarios
- can NOT ignore that real world always changes
- can NOT ignore it’s important to know when you don’t know

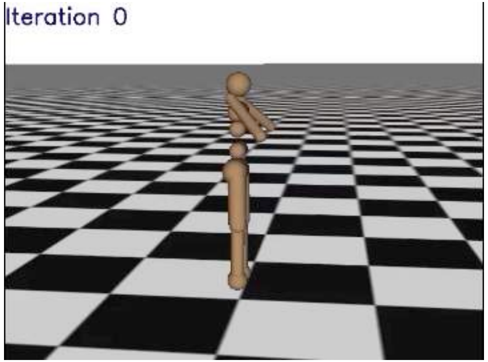
Real World: High Variability / Long Tail

- ImageNet
  - 1000 categories
  - 1000 example images for each category
- Real world
  - Millions of SKUs
  - Transparency..
  - …

→ significant AI R&D needed to meet real-world demands

Real World: Always changing

- Deep RL Research
  - Simulation
  - Train ahead of time



- Real world
  - Need to adapt on the fly…

→ significant AI R&D needed to meet real-world demands

Real World: Need to know when don’t know

How feasible today?

Outline

- Sampling of research directions
- Overall research theme
- Research <> Real-World gap
- ***How to keep up***

How to Keep Up

- ***How to read a paper?***
- What papers to read?
- Reading group

How to Read a Paper?

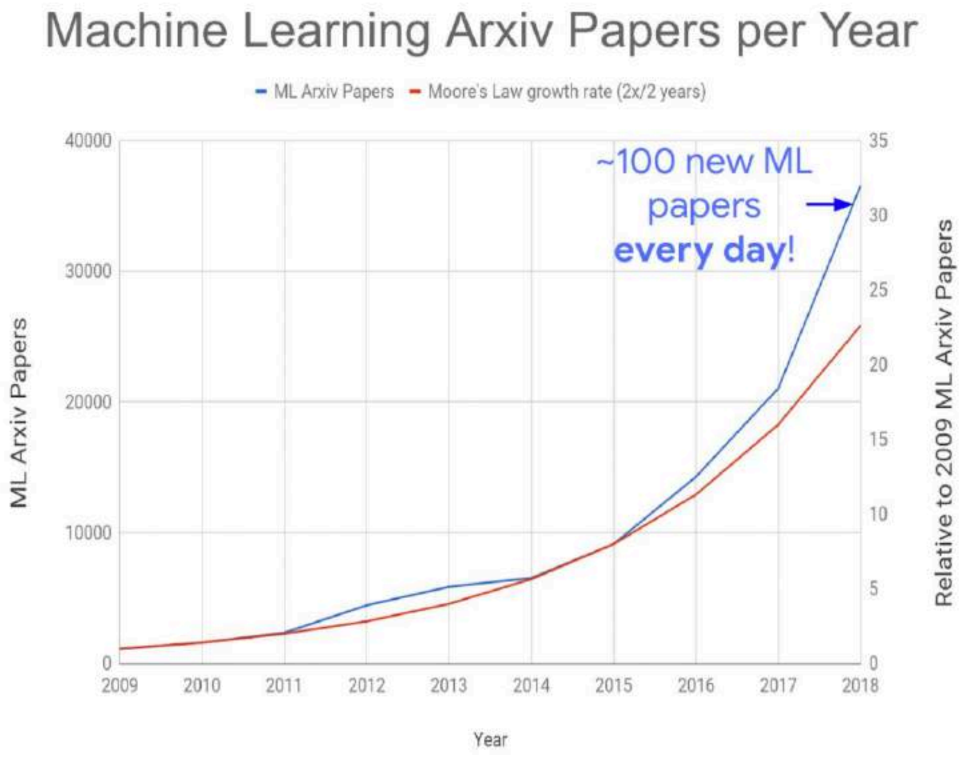
- Read the title, abstract, section headers, and figures
- Try and find slides or a video on the paper (these do not have to be by the authors).
- Read the introduction (Jennifer Widom)
  - What is the problem?
  - Why is it interesting and important?
  - Why is it hard? (e.g., why do naive approaches fail?)
  - Why hasn't it been solved before? (Or, what's wrong with previous proposed solutions? How does this paper differ?)
  - What are the key components of this approach and results? Any limitations?
- Skim the related work.
  - Is there any related work you are familiar with?
  - If so, how does this paper relate to those works?
- Skim the technical section.
  - Where are the novelties?
  - What are the assumptions?
- Read the experiments.
  - What questions are the experiments answering?
  - What questions are the experiments not answering?
  - What baselines do they compare against?
  - How strong are these baselines?
  - Is the experiment methodology sound?
  - Do the results support their claims?
- Read the conclusion/discussion.
- Read the technical section.
  - Read in “good faith” (i.e., assume the authors are correct)
  - Skip over confusing parts that don’t seem fundamental
  - If any important part is confusing, see if the material is in class slides or prior papers
- Read the paper as you see fit.

[source: Gregory Kahn]

How to Keep Up

- How to read a paper?
- ***What papers to read?***
- Reading group

Just read all the papers?



[source: https://medium.com/@karpathy/a-peek-at-trends-in-machine-learning-ab8a1085a106]

What Papers to Read?

- Import AI Newsletter (by Jack Clark, communications director at OpenAI)
- Arxiv sanity (by Karpathy)

- Twitter
  - jackclarkSF, karpathy, goodfellow_ian, hardmaru, smerity, hillbig, pabbeel, …
- AI/DL Facebook Group: active group where members post anything from news articles to blog posts to general ML questions
- ML Subreddit

How to Keep Up

- How to read a paper?
- What papers to read?
- ***Reading group***

Reading Group

- Even just 1 other person can already save you ~50% of your time

Thank you

pabbeel@cs.berkeley.edu