Actionable Models: Unsupervised Offline Reinforcement Learning of Robotic Skills

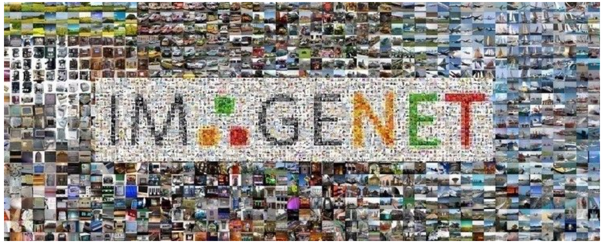
Yevgen Chebotar, Karol Hausman, Yao Lu, Ted Xiao, Dmitry Kalashnikov,
Jake Varley, Alex Irpan, Benjamin Eysenbach, Ryan Julian, Chelsea Finn, Sergey Levine

Google Research

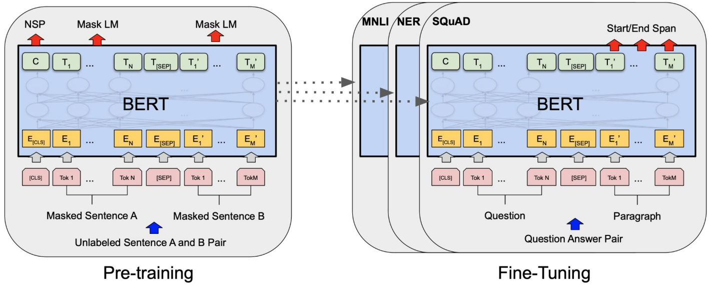
General-purpose pre-training

NLP, Computer Vision: **general-purpose** training objectives

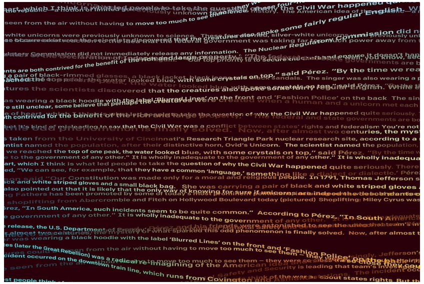
Pre-training on large datasets



ImageNet (Deng et al. 2009)



BERT (Devlin et al. 2018)



GPT-3 (Brown et al. 2020)

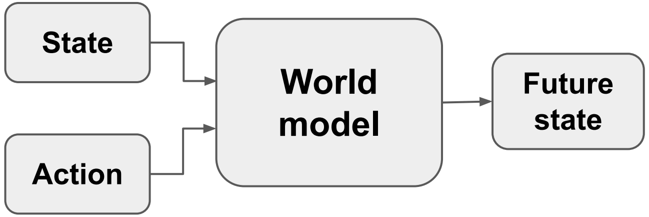
Are there **general-purpose** training objectives in Robotics?

How can we pre-train on **large robotic datasets**?

General-purpose training in Robotics

Classic view: train a **world model**

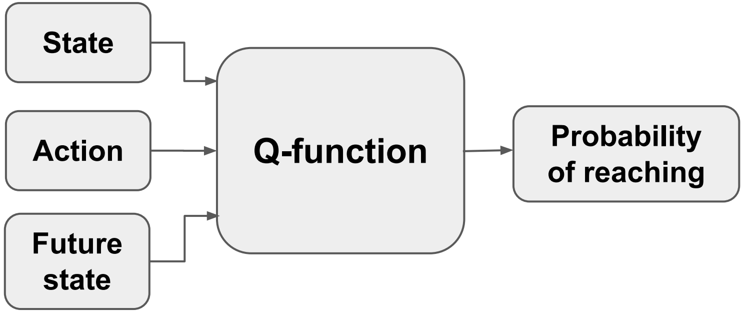
- Requires **generating possibly high-dimensional** states (e.g. robot camera images)
- Requires **additional steps** to extract policy (e.g. model-predictive control, additional policy optimization etc.)



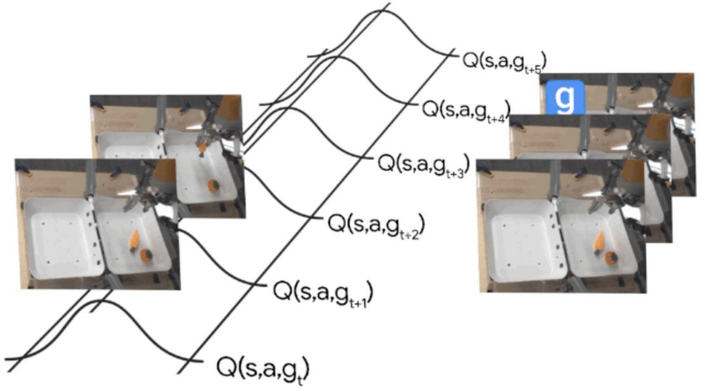
Alternative: train a **goal-conditioned Q-function**

- **Probability of reaching** a state in the future
- **Functional** understanding of the world
- Directly **actionable** representation

  Policy through $\arg\max_a Q(s, a, g)$



Actionable Models: Unsupervised Offline Learning



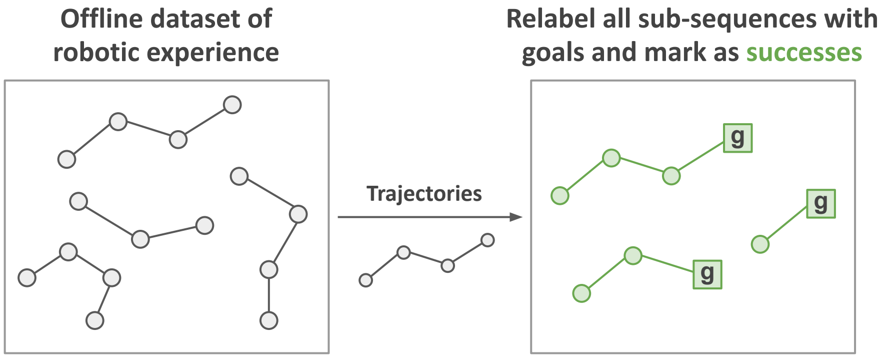
Training on all sub-sequences

Actionable Models Training

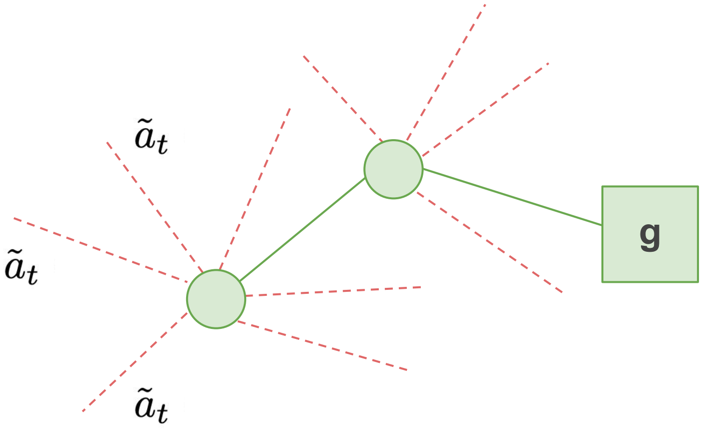
- Reach **all possible goal states / goal images** in a dataset

- **Unsupervised** objective for Robotics:
  - **Zero-shot generalization** for goal images
  - Downstream task **fine-tuning**

Actionable Models: Hindsight Relabeling


Offline dataset of robotic experience
Trajectories
Relabel all sub-sequences with goals and mark as successes
g
g
g

Actionable Models: Artificial Negatives

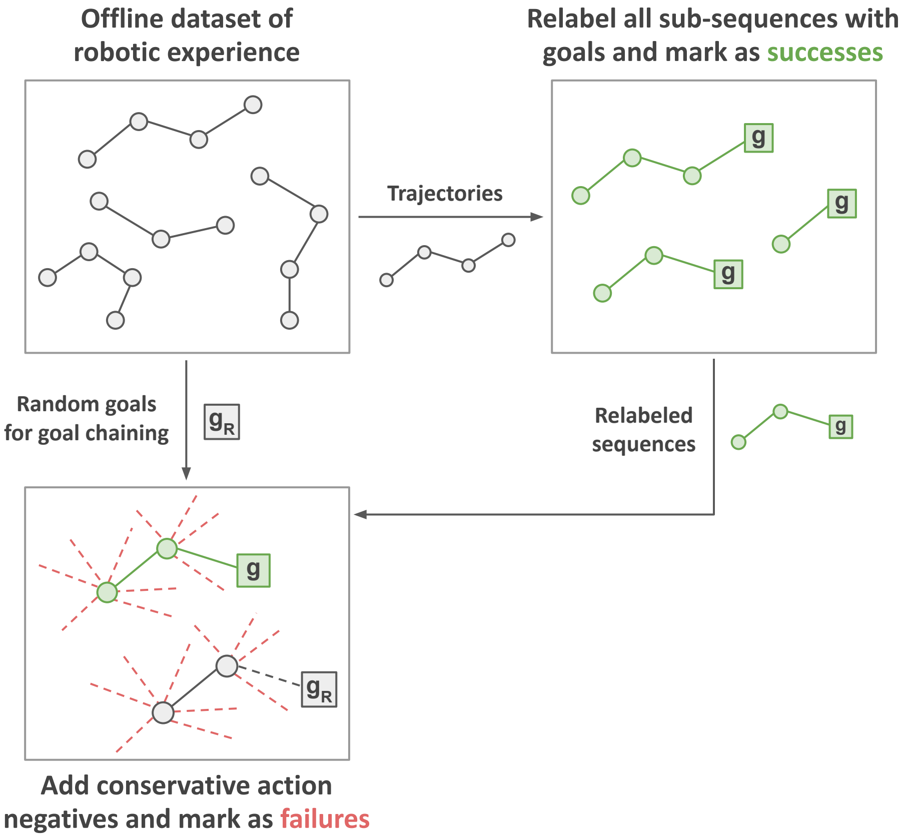


- **Offline hindsight relabeling:**
  only positive examples → need negatives

- **Conservative** strategy:
  minimize Q-values of unseen actions

- Sample **contrastive** artificial
  negative actions: $\tilde{a}_t \sim \exp(Q^\pi(s_t, \tilde{a}_t, g))$

Actionable Models: Goal Chaining



Offline dataset of robotic experience

Relabel all sub-sequences with goals and mark as successes

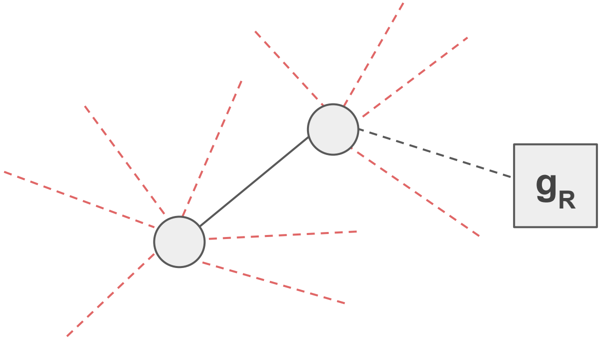
Trajectories

Random goals for goal chaining

Relabeled sequences

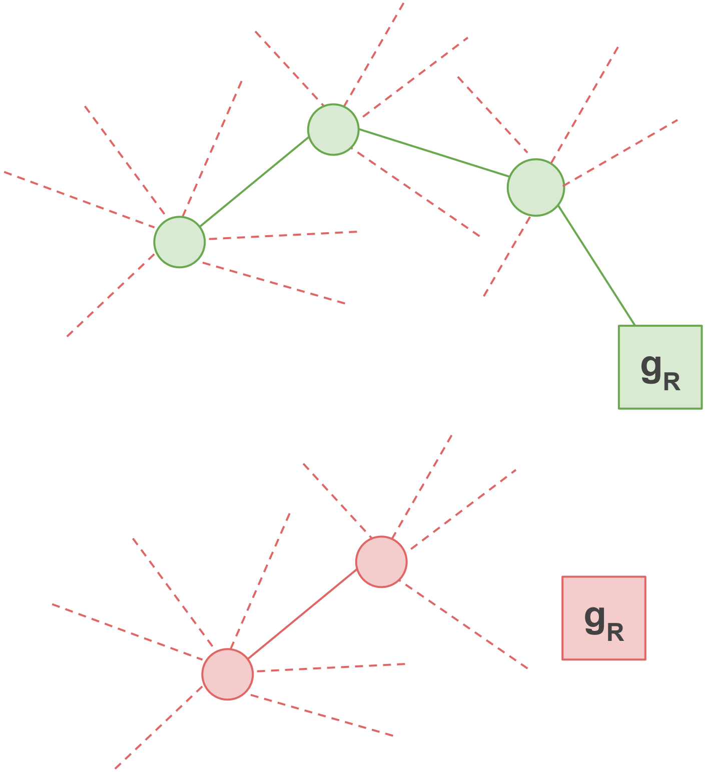
Add conservative action negatives and mark as failures

Actionable Models: Goal Chaining



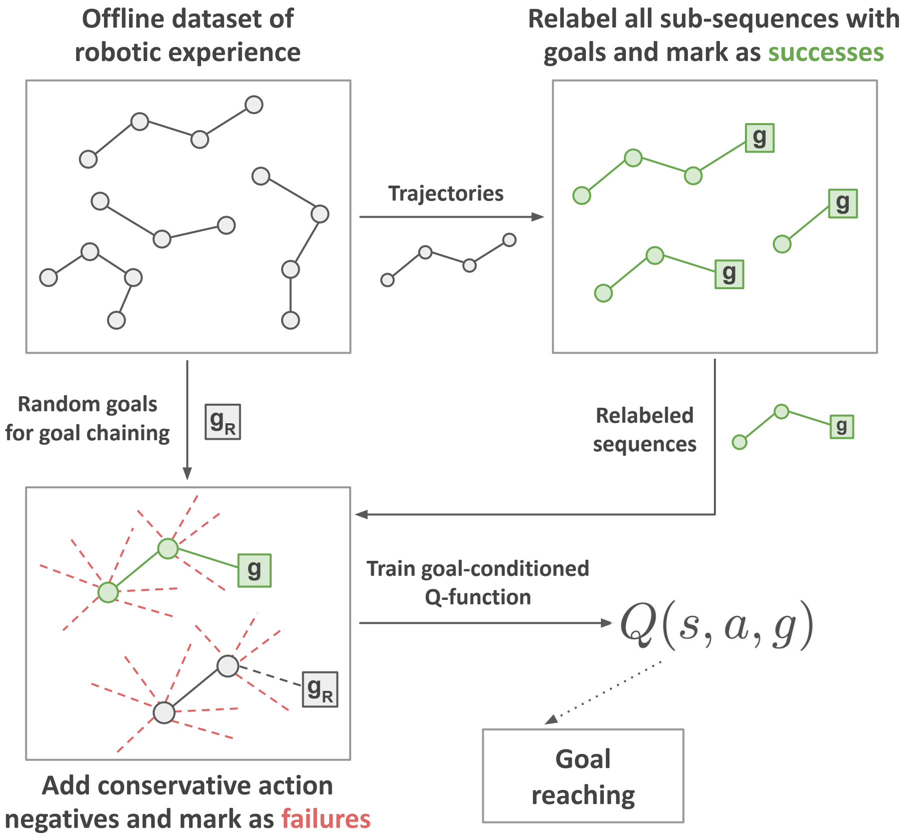
- Recondition on random goals to enable **chaining** goals **across episodes**

Actionable Models: Goal Chaining

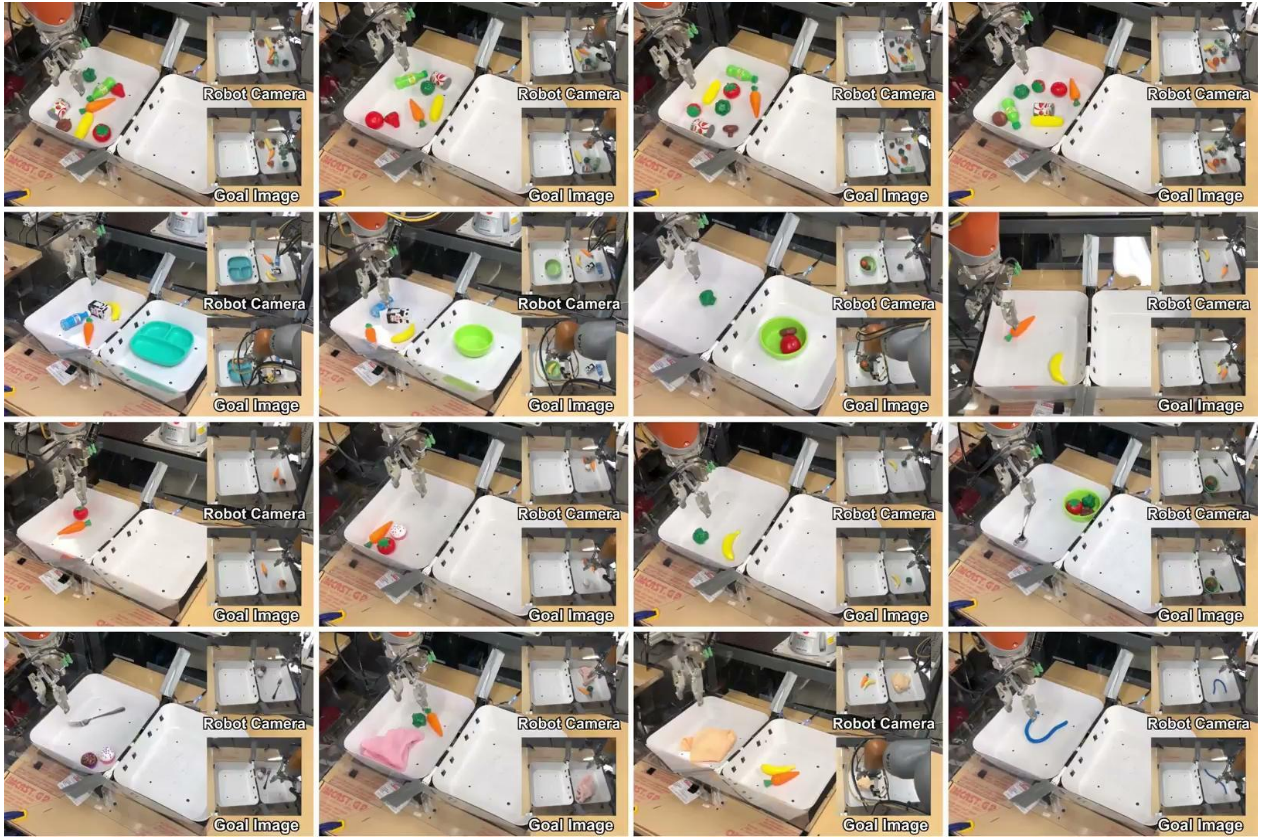


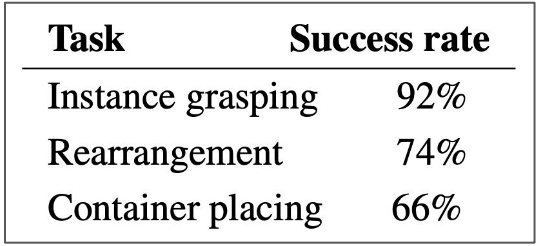
- Recondition on random goals to enable **chaining** goals **across episodes**
- If **pathway to a goal** exists:
  dynamic programming will propagate reward
- **No pathway to the goal**:
  conservative strategy will minimize Q-values

Actionable Models: Real world visual goal reaching


Offline dataset of robotic experience
Relabel all sub-sequences with goals and mark as successes
Trajectories
g
g
g
Random goals for goal chaining
g_R
Relabeled sequences
g
g
g_R
Train goal-conditioned Q-function
$Q(s, a, g)$
Goal reaching
Add conservative action negatives and mark as failures

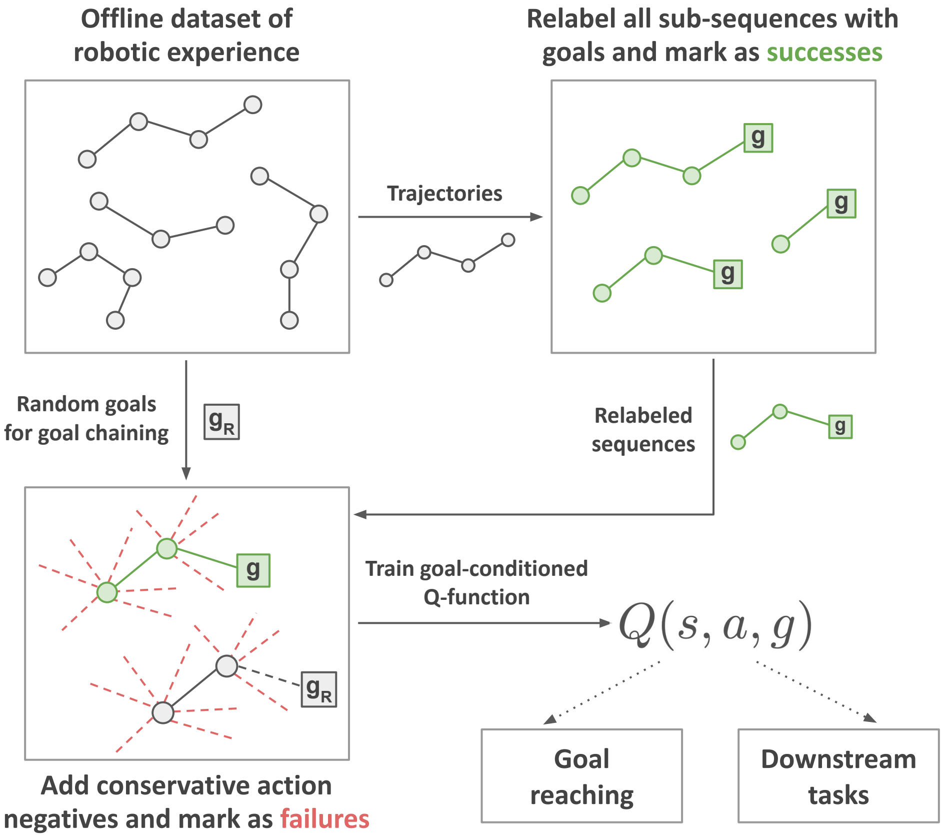
Actionable Models: Real world visual goal reaching





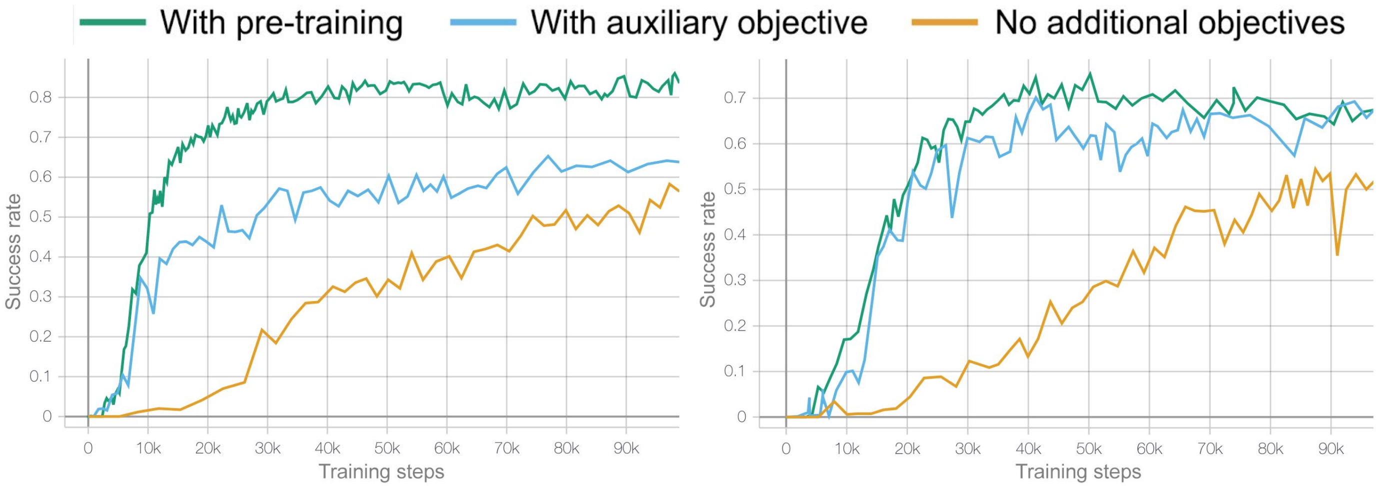
Task	**Success rate**
Instance grasping	92%
Rearrangement	74%
Container placing	66%

Actionable Models: Downstream tasks

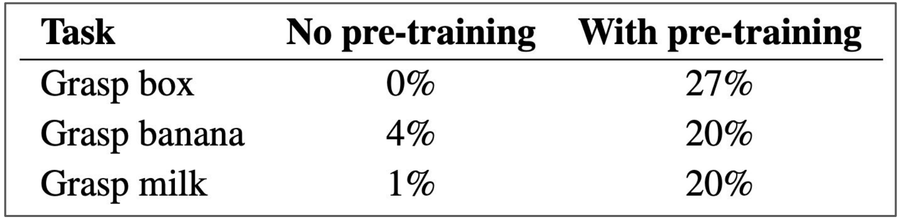

Offline dataset of robotic experience
Relabel all sub-sequences with goals and mark as successes
Trajectories
g
Random goals for goal chaining
g_R
Relabeled sequences
Train goal-conditioned Q-function
$Q(s, a, g)$
Add conservative action negatives and mark as failures
Goal reaching
Downstream tasks

Actionable Models: Downstream tasks

Simulated ablations



Real-world fine-tuning with a small amount of data



Task	No pre-training	With pre-training
Grasp box	0%	27%
Grasp banana	4%	20%
Grasp milk	1%	20%

Actionable Models:
Unsupervised Offline Reinforcement Learning of Robotic Skills

Thank you!