Skill-based Model-based Reinforcement Learning

Lucy Xiaoyang Shi, Joseph J. Lim, Youngwoon Lee.

2023.02.23

김나영

Skill-based Model-based Reinforcement Learning

Lucy Xiaoyang Shi, Joseph J. Lim, Youngwoon Lee.

Motivation

- Planning every action for long-horizon tasks is not practical
- Humans efficiently plan with high-level skills to solve complex tasks

Approach

- Skill-based Model-based RL framework (SkiMo), which directly predicts the skill outcomes, rather than predicting all small details in the intermediate states, step by step

Results & New finding

- SkiMo extends the temporal horizon of model-based approaches and improves the sample efficiency for model-based RL and skill-based RL

Discussion & Comments

- SkiMo is a general framework that can be extended to RGB, depth, and tactile observations

Motivation

Idea from Human Intelligence

- The ability to plan abstractly for solving complex tasks
- It can be used to scale the model to long-horizon tasks by reducing the search space of behaviors

Suggestion

- A novel <u>skill-based</u> and <u>model-based</u> reinforcement learning (RL) method,
  which learns a model and a policy in a high-level skill space,
  enabling accurate long-term prediction and efficient long-term planning

Model-Based RL Model

Mechanism
- Learning a flat single-step dynamics model, which predicts the next state from the current state and action

Pros
- Can be used to simulate imaginary trajectories, which improves sample efficiency over model-free alternatives

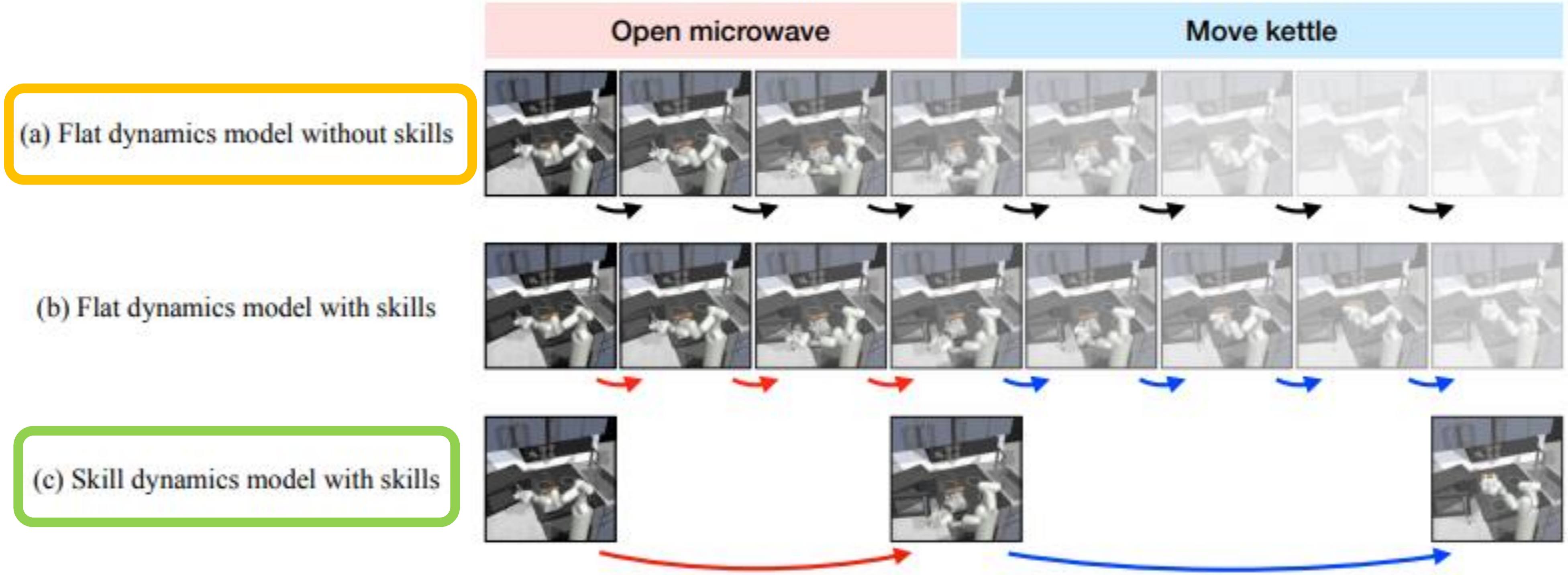
Cons
- Only limited success in long-horizon tasks due to
  - (1) inaccurate long-term prediction
  - (2) computationally expensive search

Skill-Based RL Model

Mechanism
- To solve long-horizon tasks by acting with multi-action subroutines (skills)

Pros
- (1) Enables systematic long-range exploration to plan farther into the future
- (2) Requires a shorter horizon for policy optimization, which makes long-horizon downstream tasks more tractable

Cons
- Requires a few million to billion environment interactions to learn



Skill-Based RL Model

Directly predicts the resultant state after skill execution, without neeing to model every intermediate step and low-level action

Model-Based RL Model

Predicts the immediate next state after one action execution

Wu et al.

- A temporally-extended dynamics model

Limit
- Conditions on low-level actions rather than skills
- Only used for low-level planning

Shaha et al.

- Learns a skill dynamics model

Limit
- A limited set of discrete, manually-defined skills

SkiMo

Mechanism
- Extract the skill space from data
- Devise a skill-level dynamics model

Meaning
- SkiMo is the first work that jointly learns skills and a skill dynamics model from data for model-based RL

Preliminaries

- Formulate a problem as a Markov decision process

Unlabeled Offline Data
- Assume reward-free task-agnostic dataset, which is a set of N state-action trajectories
- Do not assume this dataset contains solutions for the downstream task,
  tackling the downstream task requires recomposition of skills learned from diverse trajectories

Skill-based RL
- Skills = A sequence of actions with a fixed horizon H
- Parameterize skills as a skill latent z and skill policy π, that maps a skill latent and state
- (Step 1) The skill latent and skill policy can be trained using variational auto-encoder (VAE)
  - A skill encoder embeds a sequence of transitions into a skill latent z
  - A skill policy decodes it back to the original actions sequence
- (Step 2) Learn a skill prior p(z|s)* to guide the downstream task policy to explore promising skills

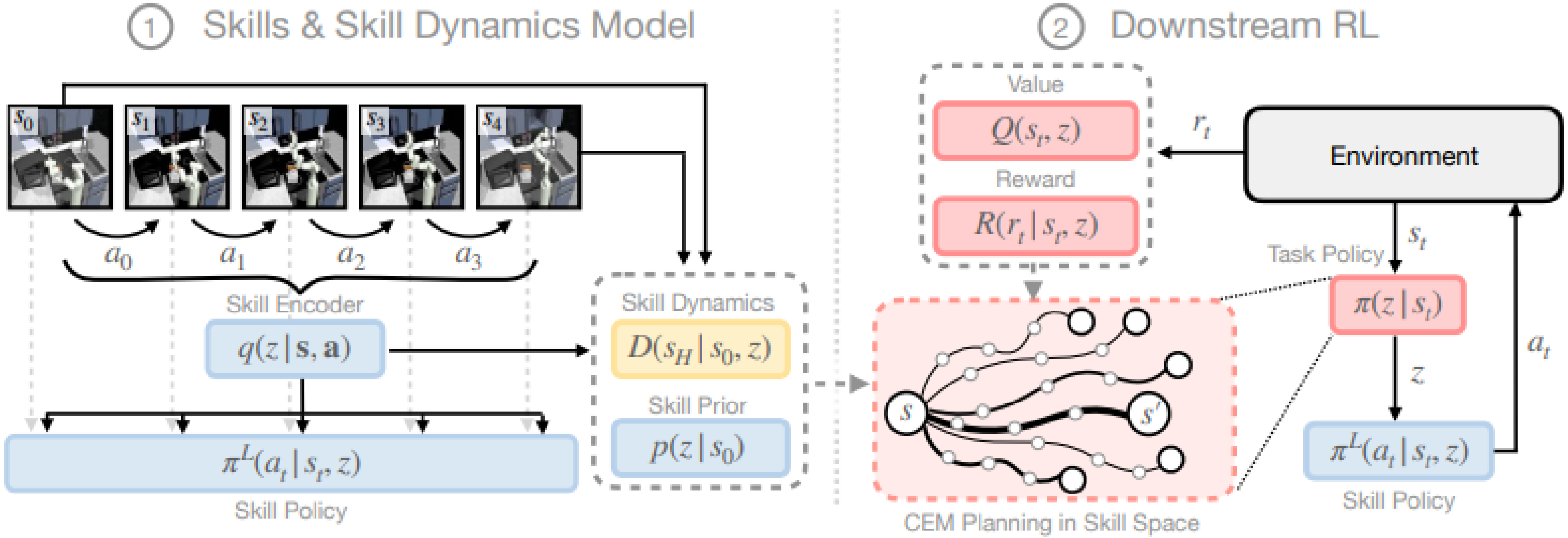
* A skill prior : the skill distribution in the offline data

SkiMo Model Components

- (1) Skill policy
- (2) Skill dynamics model
- (3) Task policy

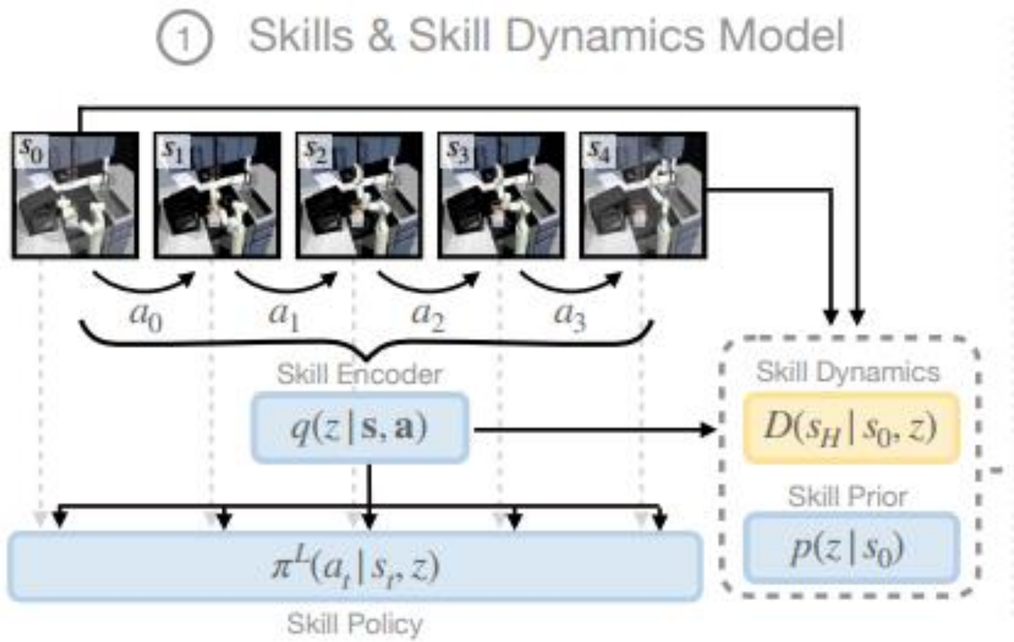
SkiMo Mechanism

- (Step 1) A state encoder encodes an observation s into the latent state h
- (Step 2) Then, given a skill z, the skill dynamics model predicts the skill effect in the latent space
- (Step 3) The task policy, reward function, and value function predict a skill, reward, and value on the (imagined) latent state, respectively



- SkiMo consists of two phases
  - (1) Learning the skill dynamics model and skills from an offline dataset
  - (2) Downstream task learning with the skill dynamics mode

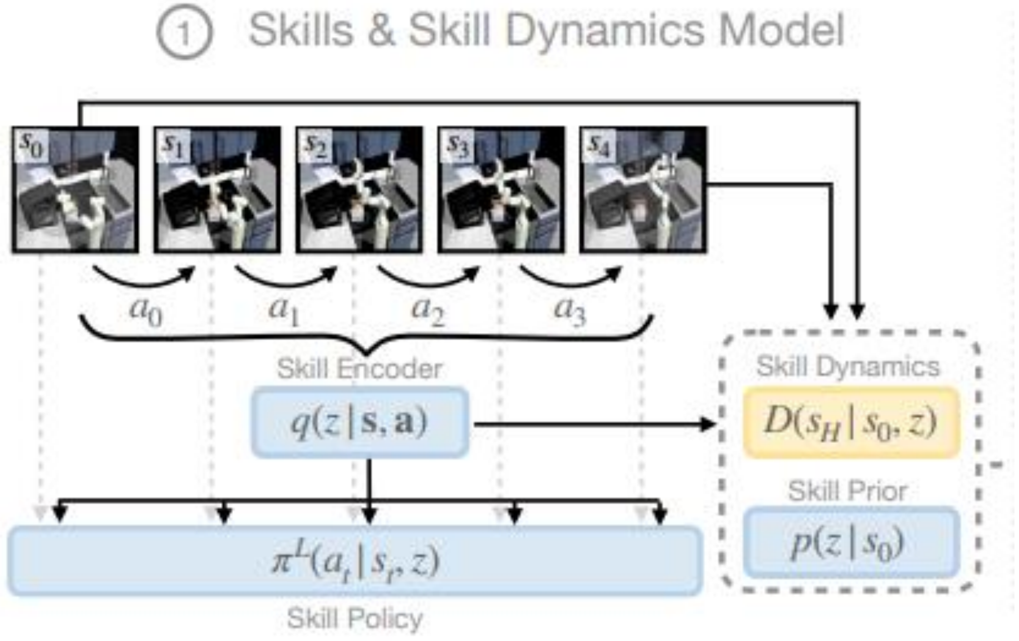
Pre-Training Skill Dynamics Model and Skills from Task-agnostic Data



- SkiMo leverages offline data to extract
  - (1) skills for temporal abstraction of actions
  - (2) skill dynamics for skill-level planning
  - on a latent state space
  - (3) a skill prior to guide exploration

- Jointly learn a skill policy and skill dynamics model, in a self-supervised manner
  - Shape the latent skill space Z and state embedding

Pre-Training Skill Dynamics Model and Skills from Task-agnostic Data



(1) Skill policy

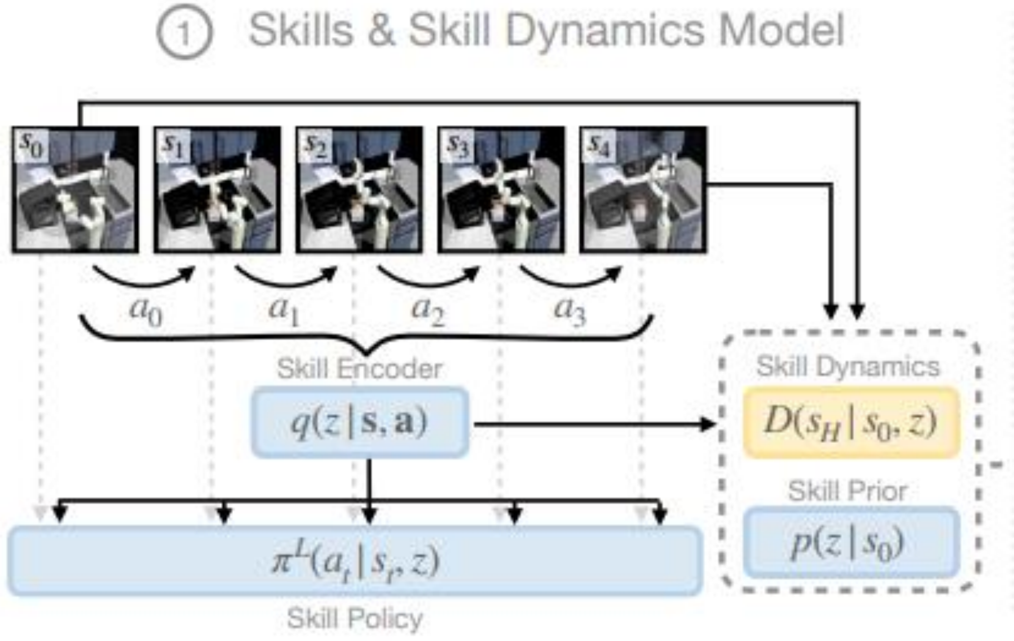
- To learn a low-dimensional <u>skill latent space Z*</u>, we train a conditional VAE on the offline dataset that reconstructs the action sequence through a skill embedding

- Given H consecutive states and actions,
  - (1) <u>a skill encoder</u> predicts a skill embedding z
  - (2) <u>a skill decoder</u> recontructs the original action sequence from z

$$\mathcal{L}_{\text{VAE}} = \mathbb{E}_{(\mathbf{s},\mathbf{a})_{0:H-1}\sim\mathcal{D}}\left[\frac{\lambda_{\text{BC}}}{H}\sum_{i=0}^{H-1}\underbrace{(\pi_\theta^L(\mathbf{s}_i,\mathbf{z})-\mathbf{a}_i)^2}_{\text{Behavioral cloning}}+\beta\cdot\underbrace{KL\big(q_\theta(\mathbf{z}|(\mathbf{s},\mathbf{a})_{0:H-1})\parallel p(\mathbf{z})\big)}_{\text{Embedding regularization}}\right], \quad (2)$$

* A skill latent space : it encodes action sequences

Pre-Training Skill Dynamics Model and Skills from Task-agnostic Data



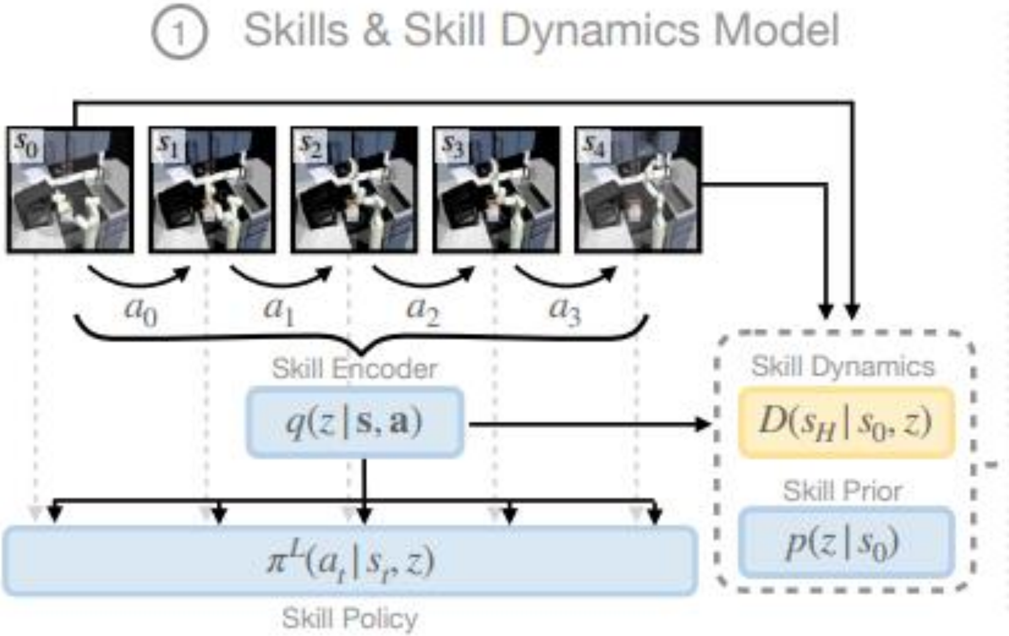
(2) Skill dynamics model

- (1) Learns to predict the latent state H-steps ahead conditioned on a skill, for N sequential skill transitions using the latent state consistency loss

- (2) To prevent a trivial solution and encode rich information from observations, we additionally train an observation decoder using the observation reconstruction loss

$$\mathcal{L}_{\text{REC}} = \mathbb{E}_{(\mathbf{s},\mathbf{a})_{0:NH}\sim\mathcal{D}}\left[\sum_{i=0}^{N-1}\left[\underbrace{\lambda_{\text{O}}\|\mathbf{s}_{iH} - O_\theta(E_\psi(\mathbf{s}_{iH}))\|_2^2}_{\text{Observation reconstruction}} + \underbrace{\lambda_{\text{L}}\|D_\psi(\hat{\mathbf{h}}_{iH},\mathbf{z}_{iH}) - E_{\psi^-}(\mathbf{s}_{(i+1)H})\|_2^2}_{\text{Latent state consistency}}\right]\right] \quad (3)$$

Pre-Training Skill Dynamics Model and Skills from Task-agnostic Data



(3) Skill prior

- is trained by minimizing the KL divergence between output distributions of the skill encoder and the skill prior

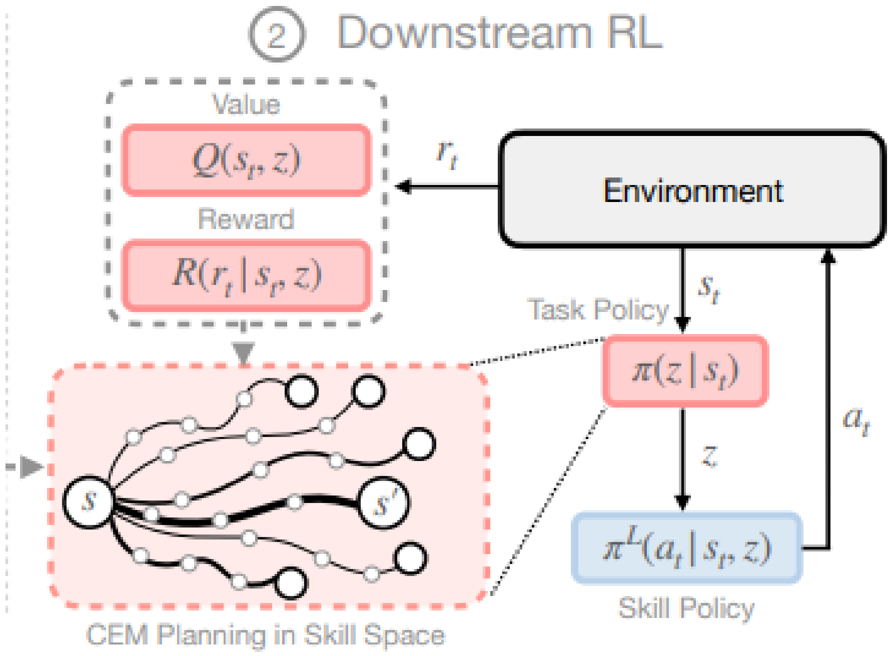
$$\mathcal{L}_{\mathrm{SP}} = \mathbb{E}_{(\mathbf{s},\mathbf{a})_{0:H-1}\sim\mathcal{D}}\left[\lambda_{\mathrm{SP}} \cdot KL\Big(\mathbf{sg}(q_\theta(\mathbf{z}|\mathbf{s}_{0:H-1},\mathbf{a}_{0:H-1})) \parallel p_\theta(\mathbf{z}|\mathbf{s}_0)\Big)\right], \quad (4)$$

Pre-Training Skill Dynamics Model and Skills from Task-agnostic Data

$$\boxed{\mathcal{L} = \mathcal{L}_{\text{VAE}} + \mathcal{L}_{\text{REC}} + \mathcal{L}_{\text{SP}}}$$

- Jointly train the policy, model, and prior, which leads to a well-shaped skill latent space that is optimized for both skill reconstruction and long-term prediction

Downstream Task Learning with Learned Skill Dynamics Model



- (1) Learns a high-level <u>task policy</u> that outputs a latent skill embedding z
- (2) skill embedding z is then translated into a sequence of H actions using the pre-trained <u>skill policy</u>

Downstream Task Learning with Learned Skill Dynamics Model

STEP 1

- The skill dynamics model and task policy can generate imaginary rollouts in the skill space by repeating
  - (1) sampling a skill
  - (2) predicting H-step future after executing the skill

STEP 2

- To evaluate imaginary rollouts, we train a reward function and Q-value function

$$\mathcal{L}'_{REC} = \mathbb{E}_{\mathbf{s}_t, \mathbf{z}_t, \mathbf{s}_{t+H}, r_t \sim \mathcal{D}} \Big[\underbrace{\lambda_{\mathrm{L}} \| D_\psi(\hat{\mathbf{h}}_t, \mathbf{z}_t) - E_{\psi^-}(\mathbf{s}_{t+H}) \|_2^2}_{\text{Latent state consistency}} + \underbrace{\lambda_{\mathrm{R}} \| r_t - R_\phi(\hat{\mathbf{h}}_t, \mathbf{z}_t) \|_2^2}_{\text{Reward prediction}} + \underbrace{\lambda_{\mathrm{V}} \| r_t + \gamma Q_{\phi^-}(\hat{\mathbf{h}}_{t+H}, \pi_\phi(\hat{\mathbf{h}}_{t+H})) - Q_\phi(\hat{\mathbf{h}}_t, \mathbf{z}_t) \|_2^2}_{\text{Value prediction}} \Big]. \quad (6)$$

STEP 3

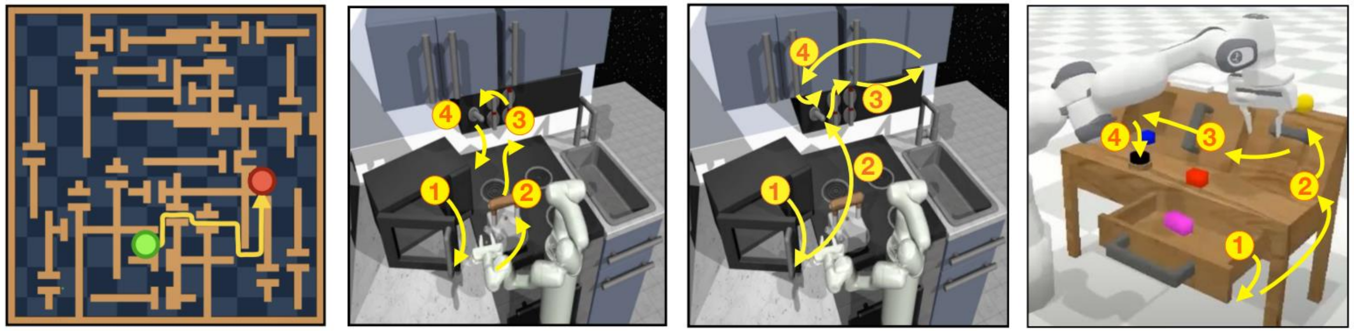
- Finetune the skill dynamics model and state encoder on the downstream task

STEP 4

- Train a high-level task policy to maximize the estimated Q-value while regularizing it to the pre-trained skill prior

$$\mathcal{L}_{RL} = \mathbb{E}_{\mathbf{s}_t \sim \mathcal{D}} \big[-Q_\phi(\hat{\mathbf{h}}_t, \pi_\phi(\mathbf{sg}(\hat{\mathbf{h}}_t))) + \alpha \cdot KL(\pi_\phi(\mathbf{z}_t | \mathbf{sg}(\hat{\mathbf{h}}_t)) \parallel p_\theta(\mathbf{z}_t | \mathbf{s}_t)) \big]. \quad (7)$$

Tasks



(a) Maze (b) Kitchen (c) Mis-aligned Kitchen (d) CALVIN

- Compare SkiMo with prior model-based RL and skill-based RL methods on four long-horizon tasks with sparse rewards

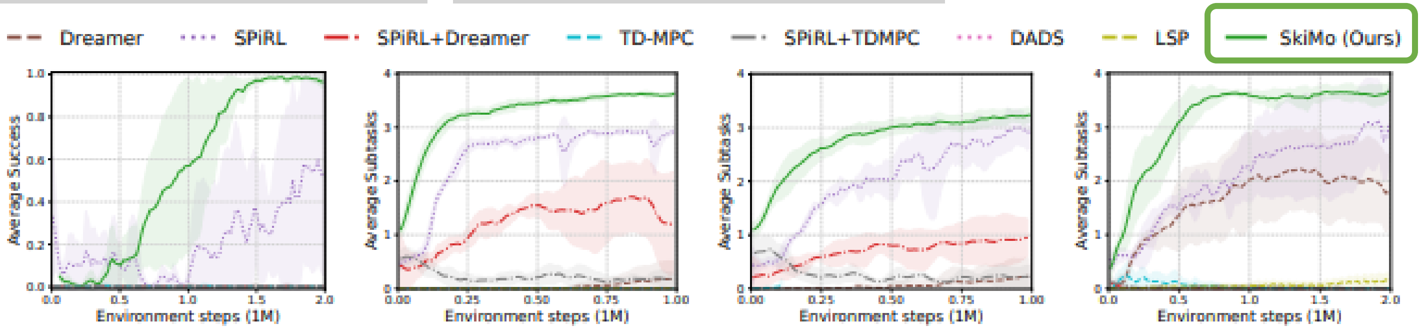
Maze
- (goal) to reach the fixed goal region in red
- (reward) a sparse reward of 100 only when it reaches the goal

Kitchen
- (goal) to perform four sequential subtasks
- (reward) a reward of 1 for every sub-task completion in order

Mis-aligned Kitchen
- (goal) to perform four sequential subtasks, has a low sub-task transition probability
- (reward) a reward of 1 for every sub-task completion in order

CALVIN
- (goal) to perform four sequential subtasks
- (reward) a reward of 1 for every sub-task completion in order

Results



(a) Maze (b) Kitchen (c) Mis-aligned Kitchen (d) CALVIN

Figure 4: Learning curves of our method and baselines. All averaged over 5 random seeds.

Maze

- A hard exploration problem due to the sparsity of the reward: the agent only receives reward after taking 1,000+ steps to reach the goal

Kitchen

- SkiMo reaches the same performance with 5x less environment interactions than SPiRL

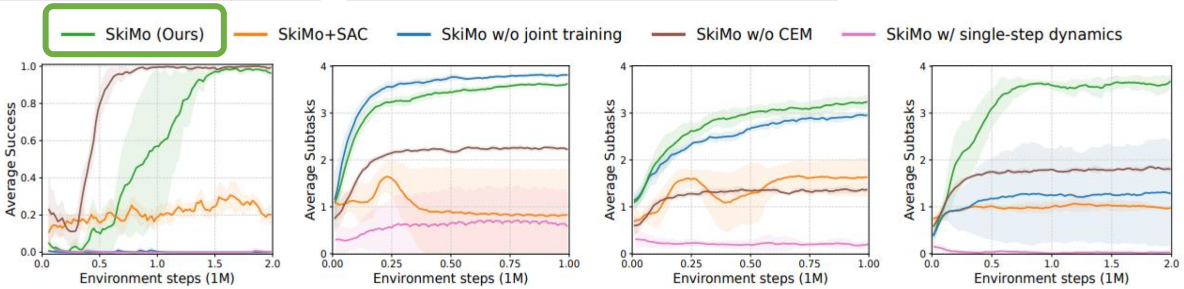
Mis-aligned Kitchen

- makes the downstream learning harder because the skill prior offers less meaningful regularization to the policy

CALVIN

- Is very task-agnostic: any particular sub-task transition has probability lower than 0.1% on average

Ablation Studies



(a) Maze (b) Kitchen (c) Mis-aligned Kitchen (d) CALVIN

Figure 5: Learning curves of our method and ablated methods. All averaged over 5 random seeds.

Model-based VS. Model-free

- SkiMo achieves better asymptotic performance and higher sample efficiency across all tasks than SkiMo+SAC, which uses model-free RL

Joint training of skills and skill dynamic model

CEM planning

- SkiMo learns significantly better and faster in Kitchen, Misaligned Kitchen, and CALVIN than SkiMo w/o CEM

Skill dynamics model

- The skill dynamic model can make accurate long-horizon predictions for planning due to significantly less compounding errors

- (1) A skill dynamics model reduces the long-term future prediction error via temporal abstraction
- (2) Without needing to plan step-by-step, downstream RL over the skill space allows for efficient and accurate temporally-extended reasoning
- (3) Joint training of the skill dynamics and skills further improves the sample efficiency by learning skills conducive to predict their consequences

Limitation and future work

- (1) While this method extracts fixed-length skills from offline data, the lengths of semantic skills may vary based on the contexts and goals
- (2) Further, although this experiment only focused on state-based inputs, SkiMo is a general framework that can be extended to RGB, depth, and tactile observations

Skill-based Model-based Reinforcement Learning

Lucy Xiaoyang Shi, Joseph J. Lim, Youngwoon Lee.

2023.02.23

김나영