Efficient Hierarchical Reinforcement Learning via Mutual Information Constrained Subgoal Discovery

Kaishen Wang[1,2], Jingqing Ruan[2], Qingyang Zhang[2], and Dengpeng Xing[1,2(✉)]

[1] University of Chinese Academy of Sciences, Beijing 100049, China
[2] Institute of Automation, Chinese Academy of Sciences, Beijing 100190, China
{wangkaishen2021,ruanjingqing2019,zhangqingyang2019,
dengpeng.xing}@ia.ac.cn

Abstract. Goal-conditioned hierarchical reinforcement learning has demonstrated impressive capabilities in addressing complex and long-horizon tasks. However, the extensive subgoal space often results in low sample efficiency and challenging exploration. To address this issue, we extract informative subgoals by constraining their generation range in mutual information distance space. Specifically, we impose two constraints on the high-level policy during off-policy training: the generated subgoals should be reached with less effort by the low-level policy, and the realization of these subgoals can facilitate achieving the desired goals. These two constraints enable subgoals to act as critical links between the current states and the desired goals, providing more effective guidance to the low-level policy. The empirical results on continuous control tasks demonstrate that our proposed method significantly enhances the training efficiency, regardless of the dimensions of the state and action spaces, while ensuring comparable performance to state-of-the-art methods.

Keywords: Hierarchical reinforcement learning · Subgoal discovery · Mutual information

1 Introduction

Goal-conditioned hierarchical reinforcement learning (HRL) [9,11,12,19] has received much attention due to its significant performance in solving complex and long-term tasks. Among HRL frameworks, goal-conditioned HRL typically consists of a high-level policy and a low-level policy. The high-level policy decomposes the desired goal into simpler subgoals, allowing the low-level policy to learn and explore more effectively. However, identifying informative and reachable subgoals in the extensive subgoal space to enhance sample and training efficiency still remains a major challenge.

This work is supported by the Program for National Nature Science Foundation of China (62073324).

B. Luo et al. (Eds.): ICONIP 2023, CCIS 1961, pp. 76–87, 2024.
https://doi.org/10.1007/978-981-99-8126-7_6

Over the past few years, several works [8,9,12,20] have been proposed to improve sample efficiency in HRL. Kulkarni et al. [9] predefine a set of key states as the subgoal space, which is efficient but requires task-relevant knowledge. Nachum et al. [12] propose the off-policy correction method that enables the high-level policy to be trained in an off-policy manner. However, its training cost increases considerably when dealing with larger state and action spaces. Zhang et al. [20] leverage the concept of adjacency distance to confine subgoals within a reachable range of k steps, which reduces the subgoal space and enhances training efficiency. Nonetheless, maintaining an adjacency matrix to calculate the distances between subgoals can incur additional training expense and storage requirement. Kim et al. [8] construct a landmark graph and select subgoals by planning on the graph based on prior work [20]. Although these approaches reduce the subgoal space, they also entail an increase in training time.

In this paper, we propose a novel method called **M**utual **I**nformation-based **S**ubgoal **D**iscovery (MISD) to improve training efficiency while maintaining sample efficiency. Concretely, by maximizing the mutual information between the subgoal and the actually achieved goal, this subgoal can be reached easily by the low-level policy, as illustrated in Fig. 1. Analogously, by maximizing the mutual information between the subgoal and the desired goal, the probability of achieving the desired goal will increase after accomplishing this subgoal. Our main contributions are outlined below: 1) We introduce mutual information as a metric of distance between subgoals and identify the most informative subgoals for the agent to explore in HRL. 2) Our method can take advantage of the correlation between subgoals to avoid costly goal-relabeling and reduce the need for extensive exploration, resulting in improved sample efficiency. 3) The experimental results demonstrate that our method is dimension-agnostic with respect to the state and action spaces and significantly improves training efficiency in diverse continuous control tasks.

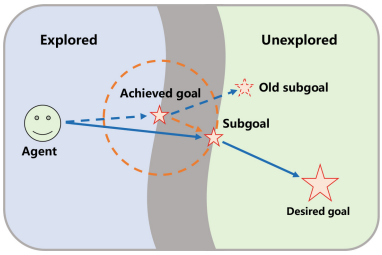


Fig. 1. The dashed blue line denotes the historical transition, the solid blue line denotes the new transition, and the dashed orange line denotes the subgoal constraint range. The old subgoal may be unachievable in reality due to being too far away. (Color figure online)

The remaining of the paper is structured as follows: Sect. 2 discusses related works in HRL. Section 3 briefly introduces goal-conditioned HRL and mutual information estimator. Section 4 presents the framework of our method. Section 5 includes experimental settings, empirical results, ablation studies and visualizations. Last, Sect. 6 concludes this paper with a summary.

2 Related Works

Subgoal Discovery. Goal-conditioned HRL[9,13], which incorporates high-level and low-level policies, has demonstrated immense potential in solving

diverse complex tasks. By combining the hindsight technique [1], Levy et al. [11] can train multiple levels of policies concurrently. However, the vast subgoal space limits the identification of effective subgoals. In order to tackle this issue, some methods [5,13] utilize online planning to select feasible subgoals. Hafner et al. [5] apply a world model to generate an imagined trajectory and select subgoals. However, this approach needs additional training of the world model for planning purpose. Some graph-based methods [7,19] have also been proposed to address this challenge. Nevertheless, these approaches require creating a graph and performing online planning over it, which can be computationally expensive and time-consuming.

Mutual Information. In recent years, mutual information is applied in skill-based HRL [3,18] to generate diverse skills, increase the exploration ability of the agent, and enhance the transferability of the skills. Eysenbach et al. [3] can train various skills using mutual information without relying on environmental rewards, but these skills need to be pre-trained before they can be transferred to downstream tasks. In practice, due to unknown data distributions, calculating the mutual information accurately is often difficult [15]. Oord et al. [14] propose the InfoNCE loss, a method of contrastive learning [4,10], to estimate the lower bound on mutual information. In our paper, we utilize mutual information as a distance metric [6,16] between subgoals by learning a representation function which maps these subgoals to the mutual information distance space.

3 Preliminaries

Goal-Conditioned HRL can be expressed as a finite horizon Markov Decision Process (MDP) with tuple $(\mathcal{S}, \mathcal{A}, \mathcal{G}, P, R, \gamma)$, where $\mathcal{S}$ is the state space, $\mathcal{A}$ is the action space, $\mathcal{G}$ is the subgoal space which is mapped from $\mathcal{S}$ by the function $\varphi : \mathcal{S} \to \mathcal{G}$, $P : \mathcal{S} \times \mathcal{A} \times \mathcal{S} \to [0, 1]$ is the transition function, $R : \mathcal{S} \times \mathcal{A} \to \mathbb{R}$ is the reward function, and $\gamma \in [0, 1)$ is the discount factor. Following prior works [12], we formulate the framework composed of two hierarchies: a high-level policy $\pi^h_{\theta_h}$ and a low-level policy $\pi^l_{\theta_l}$ parameterized by θ_h and θ_l, respectively. The high-level policy generates subgoal $g_t \sim \pi^h_{\theta_h}(s_t, g_d)$ every k steps until the episode terminates at step T, where $g_d \in \mathcal{G}$ is the desired goal that the agent needs to achieve. When $t \equiv 0 \pmod{k}$, the low-level policy receives subgoal $g_t \in \mathcal{G}$ from the high-level policy, otherwise, it resorts to using a fixed subgoal transition function:

$$g_{t+1} = h(s_t, g_t, s_{t+1}) = \varphi(s_t) + g_t - \varphi(s_{t+1}). \tag{1}$$

Then, the low-level policy performs a primitive action $a_t \sim \pi^l_{\theta_l}(s_t, g_t)$, which results in the environment transferring to the next state according to the transition function $s_{t+1} \sim P(s_{t+1}|s_t, a_t)$ and giving reward $r_t \sim R(s_t, a_t)$. Without involving environmental rewards, the low-level policy is motivated by the high-level policy with intrinsic reward $r^l_t = -||\varphi(s_t) + g_t - \varphi(s_{t+1})||_2$.

Based on the above setups, the transitions of high-level and low-level policies can be denoted as $(s_t, g_d, g_t, r^h_t, s_{t+k})$ and $(s_t, g_t, a_t, r^l_t, s_{t+1})$, respectively, where

$r_t^h = \sum_{t:t+k-1} r_t$, and s_{t+k} is the achieved state by the low-level policy. The objective of the high-level policy is to maximize the expected cumulative reward provided by the environment:

$$\mathcal{L}_{rew}(\theta_h) = -\mathbb{E}_{\pi^h_{\theta_h}}\Big[\sum_{t=0}^{T-1} \gamma^t r_t^h\Big]. \tag{2}$$

Mutual Information Estimator [14,17] is a technique used to estimate the mutual information between two random variables. In our method, we employ the InfoNCE loss [14] to estimate the lower bound on mutual information, which learns representations by maximizing the similarity between positive samples and minimizing the similarity between negative samples:

$$\begin{aligned}\mathcal{L}_{InfoNCE} &= -\mathbb{E}_{s\in\mathcal{S}}\left[\log\frac{\exp(\psi_\phi(s_i)^T \cdot \psi_\phi(s_j)/\tau)}{\sum_{n=0}^{N}\exp(\psi_\phi(s_i)^T \cdot \psi_\phi(s_n)/\tau)}\right] \\ &\geq \log(N) - I(s_i; s_j),\end{aligned} \tag{3}$$

where ψ_ϕ is a encoder network parameterized by ϕ, s_i and s_j are different states, τ is a temperature scale factor, s_n is the state sample, N is the number of samples, and I is the mutual information function.

4 Methodology

In this section, we present MISD: **M**utual **I**nfomation-based **S**ubgoal **D**iscovery, a simple and effective method for training the high-level policy with mutual information distance constraints, as shown in Fig. 2.

4.1 Mutual Information Distance Space

Previous works [8,20] have extensively studied the measurement of the distance between different subgoals, utilizing the shortest transition steps. However, these approaches overlook the correlation between subgoals which can be used to gauge the difficulty of achieving them. In contrast, we propose the concept of the mutual information distance space to estimate the distance between subgoals without considering the transition steps. In the mutual information distance space, a smaller distance corresponds to a higher mutual information, indicating a stronger correlation and a higher likelihood of achieving the subgoals jointly, even with a random policy. Therefore, we define the distance between the subgoals g_i and g_j as follows:

$$d_{st}(g_i, g_j) := -\mathbb{E}_{\pi\in\prod}[I(g_i; g_j|\pi)], \tag{4}$$

where $\prod$ is the set of policy π used by the agent, $g_i = \varphi(s_i)$, and $g_j = \varphi(s_j)$.

Minimizing the distance between subgoals can facilitate their successful realization with less effort. However, accurately and directly calculating mutual

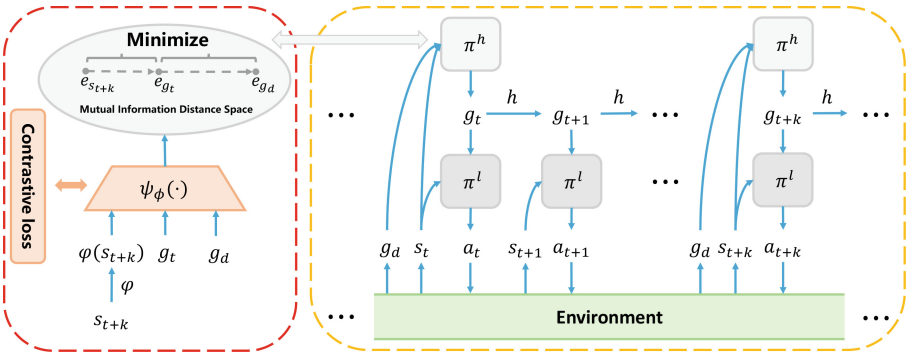


Fig. 2. The MISD framework with mutual information distance constraints implemented by the subgoal encoder ψ_ϕ (dashed red box), where $e_{s_{t+k}} = \psi_\phi(\varphi(s_{t+k}))$, $e_{g_t} = \psi_\phi(g_t)$, and $e_{g_d} = \psi_\phi(g_d)$. The encoder, trained using a contrastive loss, maps the subgoals to the mutual information distance space which is then utilized to train the high-level policy to constrain subgoals generation. (Color figure online)

information is often impractical due to intractable data distributions [15]. Instead, we estimate a lower bound on mutual information by using the InfoNCE [14] along with a limited set of policies that the agent has employed in recent C episodes. Based on the fact that the adjacent states have relatively higher mutual information in MDP, we select the next achieved goal reached by the low-level policy as the positive sample, and randomly sampled achieved goals from the current episode as the negative samples. Consequently, we derive the following optimization objective function:

$$\mathcal{L}_{dis}(\phi) = -\mathbb{E}_{s_i \in \mathcal{S}}\left[\log \frac{\exp(\psi_\phi(\varphi(s_i))^T \cdot \psi_\phi(\varphi(s_{i+1}))/\tau)}{\sum_{n=0}^{N} \exp(\psi_\phi(\varphi(s_i))^T \cdot \psi_\phi(\varphi(s_n))/\tau)}\right], \tag{5}$$

where ψ_ϕ is the subgoal encoder parameterized by ϕ, s_{i+1} is the next state following state s_i, τ is a temperature scale, s_n is the state sample, and N is the number of samples.

By optimizing the objective function, we can obtain a subgoal encoder $\psi_\phi(\cdot)$, which allows for the mapping of subgoals to the mutual information distance space and simplifies the calculation of distances between them. For subgoals g_i and g_j, the minimization of distance is equivalent to the maximization of the numerator of Eq. 5, as they demonstrate a negative correlation:

$$\begin{aligned} I(g_i; g_j) &\propto \frac{\exp(sim(\psi_\phi(g_i), \psi_\phi(g_j)))}{\sum_{n=0}^{N} \exp(sim(\psi_\phi(g_i), \psi_\phi(g_n)))} \\ &\propto sim(\psi_\phi(g_i), \psi_\phi(g_j)) \\ &\propto -d_{st}(g_i, g_j), \end{aligned} \tag{6}$$

where g_i and g_j are different subgoals, g_n is the subgoal sample, N is the number of samples, and sim denotes the similarity scoring function. Inspired by [14], we

choose the cosine similarity function defined as follows:

$$f_{cs}(\psi_\phi(g_i), \psi_\phi(g_j)) = \frac{\psi_\phi(g_i)^T \cdot \psi_\phi(g_j)}{||\psi_\phi(g_i)||_2 \cdot ||\psi_\phi(g_j)||_2}. \tag{7}$$

Alternatively, the minimization of mutual information distance is equivalent to the maximization of Eq. 7, which we then employ to enforce the aforementioned distance constraints.

4.2 Efficient Subgoal Discovery with Distance Constraint

To identify informative subgoals that efficiently guide the low-level policy to accomplish the desired goal, we introduce two mutual information distance constraints on the high-level policy.

Constrain with the Achieved Goal. In HRL, goal-relabeling [1,12] is an effective technique to help the high-level policy more quickly learn to choose achievable subgoals. However, this approach has a limitation in that the high-level policy may only be aware of specific subgoals that the low-level policy can achieve while remaining unaware of other subgoals that may be more relevant. We address this limitation by constraining the distance between subgoals and achieved goals to make these subgoals representation more informative and relevant, allowing for more effective guidance of the low-level policy. Specifically, the subgoals proposed by the high-level policy are highly correlated with the achieved goals that have already been reached by the low-level policy, making them easier to achieve. Moreover, the ability of the low-level policy can be fed back to the high-level policy to generate more effective subgoals, maintaining the consistency between hierarchical policies and enhancing the stability of the learning process. Therefore, transition samples can be used directly for training to improve data efficiency without goal-relabeling. In summary, the objective function can be written as follows:

$$\mathcal{L}_{ag}(\theta_h) = -\mathbb{E}_{\pi^h_{\theta_h}}[f_{cs}(\psi_\phi(g_t), \psi_\phi(\varphi(s_{t+k})))], \tag{8}$$

where $g_t \sim \pi^h_{\theta_h}(s_t, g_d)$, and $\varphi(s_{t+k})$ is the achieved goal reached by the low-level policy after k steps.

Constrain with the Desired Goal. After imposing the distance constraint mentioned above, the subgoals generated by the high-level policy may be too simple for the low-level policy, resulting in limited exploration capability. To address this issue, we introduce another constraint with the desired goal, aiming to ensure that the desired goal can be easily achieved once the subgoal is accomplished by the low-level policy. For this purpose, we minimize the distance between the subgoal and the desired goal, thereby expanding the exploration area around the achieved goal by the low-level policy, formulated as follows:

$$\mathcal{L}_{dg}(\theta_h) = -\mathbb{E}_{\pi^h_{\theta_h}}[f_{cs}(\psi_\phi(g_t), \psi_\phi(g_d))], \tag{9}$$

where $g_t \sim \pi^h_{\theta_h}(s_t, g_d)$, and g_d is the desired goal.

Algorithm 1. MISD algorithm

Initialize: the trajectory buffer $\mathcal{B} \leftarrow \emptyset$;
Initialize: θ_h, θ_l, and ϕ for $\pi^h_{\theta_h}$, $\pi^l_{\theta_l}$, and ψ_ϕ;
for $n = 1$ to $num\_episodes$ **do**
  $t = 0$;
  Reset the environment, sample the initial state s_0 and the desired goal g_d;
  repeat
    if $t \equiv 0 \pmod{k}$ **then**
      Generate subgoal $g_t \sim \pi^h_{\theta_h}(s_t, g_d)$;
    else
      Perform subgoal transition $g_t = h(s_{t-1}, g_{t-1}, s_t)$;
    end if
    Execute low-level action $a_t \sim \pi^l_{\theta_l}(s_t, g_t)$;
    Sample next state $s_{t+1} \sim \mathcal{P}(s_{t+1}|s_t, a_t)$ and reward $r_t \sim \mathcal{R}(s_t, a_t)$;
    $t = t + 1$;
  until episode terminates;
  Store the trajectory in buffer $\mathcal{B}$;
  Train high-level policy $\pi^h_{\theta_h}$ according to Equation 10;
  Train low-level policy $\pi^l_{\theta_l}$;
  if $n \equiv 0 \pmod{C}$ **then**
    Train the subgoal encoder ψ_ϕ with Equation 5 using buffer $\mathcal{B}$;
    Clear $\mathcal{B}$;
  end if
end for

4.3 The Policy Optimization

By enforcing mutual information distance constraints on both parts as previously stated, we can obtain the final objective function of the high-level policy:

$$\mathcal{L}_{high}(\theta_h) = \mathcal{L}_{rew}(\theta_h) + \alpha \cdot [(1-\beta) \cdot \mathcal{L}_{ag}(\theta_h) + \beta \cdot \mathcal{L}_{dg}(\theta_h)], \tag{10}$$

where $\alpha \in [0, +\infty)$ is a scale factor, and $\beta \in [0, 1]$ is the distance coefficient used to determine the exploration range. In practice, we incorporate $\mathcal{L}_{ag}$ and $\mathcal{L}_{dg}$ as additional terms into the original loss function $\mathcal{L}_{rew}$ of the high-level policy. For the low-level policy, we utilize a common reinforcement learning algorithm, e.g., temporal-difference learning methods, to train it as usual without modification. The main process of our method is presented in Algorithm 1.

5 Experiments

We design experiments to answer the following questions: 1) How does MISD perform compared to state-of-the-art methods on various continuous control tasks? 2) Can MISD enhance sample and training efficiency in HRL? 3) Does the dimensionality of the state and action spaces affect the training efficiency of MISD? 4) What is the impact of hyperparameters on the performance of MISD?

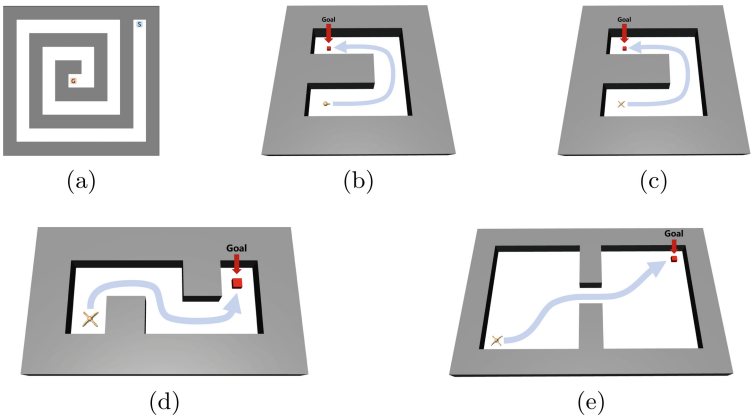


Fig. 3. Environment descriptions. (a) Spiral: navigate from the edge (denoted as 'S') to the center (denoted as 'G') with sparse rewards. (b) Point Maze (U-shape) and (c) Ant Maze (U-shape): navigate from a fixed position to a fixed target location with dense rewards. (d) Ant Maze (S-shape): navigate from a random position to a fixed target location with sparse rewards. (e) Ant TwoRooms: navigate from a fixed position in one room to a fixed target position in another room with dense rewards. (Color figure online)

5.1 Environment Setup

We evaluate our method on diverse control tasks with continuous state and action spaces based on the Mujoco simulator [2], as illustrated in Fig. 3. In all tasks, the agent needs to achieve goals with either sparse or dense rewards, where the subgoal space consists of two dimensions that correspond to the position (x, y) of the agent. The code of our method is available at https://github.com/RandyButters/MISD.

5.2 Comparative Experiments

We compare MISD with the following baselines: 1) HIRO [12]: a baseline that proposes the off-policy correction method to improve data efficiency. 2) HRAC [20]: a baseline that employs k-step reachability to constrain the range of subgoal and reduce the subgoal space. 3) HIGL [8]: a baseline that utilizes a graph of landmarks for online planning and to guide the generation of subgoals.

We present the learning curves of success rate plotted against both training time and steps, as shown in Figs. 4 and 5, respectively. The results indicate that our proposed MISD algorithm demonstrates a significant improvement in training efficiency by reducing the overall training time while achieving a comparable performance and sample efficiency to other baselines.

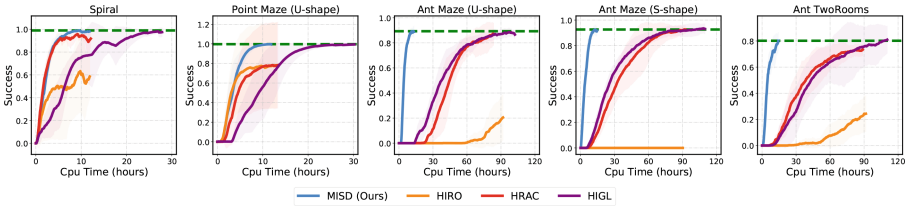


Fig. 4. The learning curves over training time, averaged over 5 trials and smoothed equally for visual clarity. The dashed line represents the best performance achieved by our method. (Color figure online)

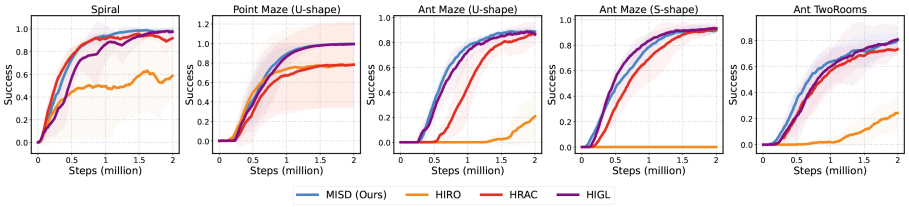


Fig. 5. The learning curves over training steps, averaged over 5 trials and smoothed equally for visual clarity. (Color figure online)

Additionally, we generate a bar chart of the training time across the Mujoco environments utilized in our experiments, as illustrated in Fig. 6. The results demonstrate that the training efficiency of MISD remains stable, while other baselines experience a significant increase in training time as the state and action spaces become larger. For example, the Ant environment has larger state and action spaces compared to the simpler Point environment, additional training time is required due to its increased complexity, resulting in lower training efficiency for the baseline methods.

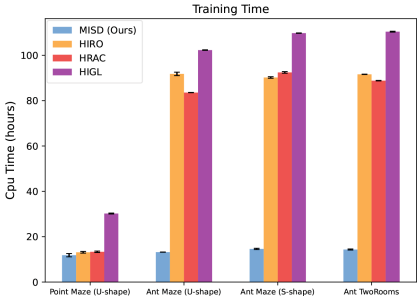


Fig. 6. Training time on different environments within the same benchmark, averaged over 5 trials. (Color figure online)

5.3 Visualizations

We visualize both achieved goals and subgoals at different training stages in the Ant Maze (U-shape) environment in Fig. 7. These subgoals generated by the high-level policy are consistently located near the achieved goals by the low-level policy, effectively guiding the low-level policy to reach the desired goal and improving training efficiency. Figure 8 displays the features of subgoals extracted by the subgoal encoder ψ_ϕ after dimension reduction by t-SNE. Initially, the

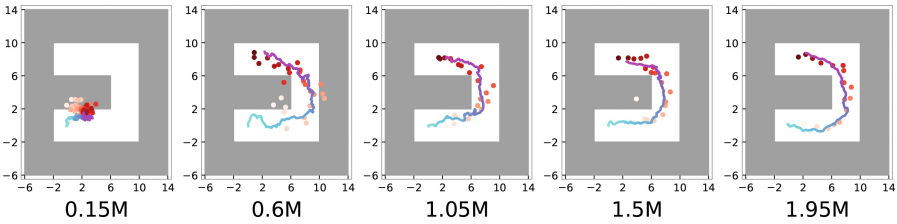


Fig. 7. Visualization of subgoals generated by the high-level policy and achieved goals accomplished by the low-level policy within one episode at a series of training steps in the Ant Maze (U-shape) environment (red for the subgoals, blue for the achieved goals, light for the start, dark for the end). (Color figure online)

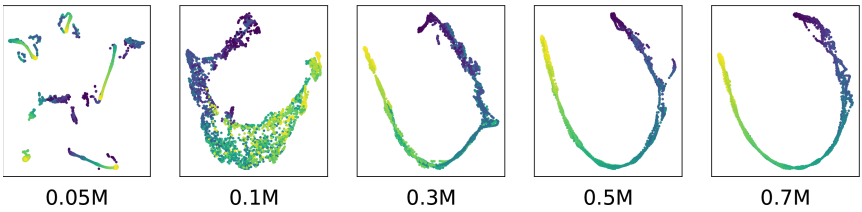


Fig. 8. Subgoal representation learning process in the Ant Maze (U-shape) environment at different training steps. Each subfigure contains 5 trajectories with the start positions highlighted in blue and the end positions in yellow. The visualization only shows the learning phase and excludes the stable phase that occurs afterward. (Color figure online)

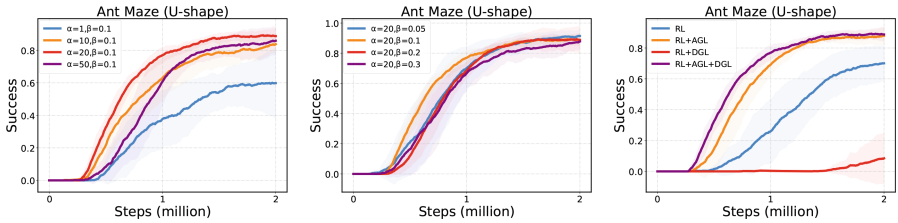
feature map is disordered at 0.05M steps but becomes gradually more ordered over time, reaching stability at 0.3M steps and remaining stable thereafter. This indicates that the features can be learned quickly and become stable as the environment is explored extensively, facilitating the generation of reliable and consistent subgoals by the high-level policy.

5.4 Ablation Studies

We perform ablation studies on the Ant Maze (U-shape) environment to analyze the impact of parameter α, parameter β, and component loss in Fig. 9.

Parameters α and β: The result of Fig. 9a indicates that $\alpha = 20$ is better for promotion in the learning process. Our algorithm MISD is robust against β due to the comparable performance with various values, as shown in Fig. 9b. Notably, we set $\alpha = 20$ and $\beta = 0.1$ as default values based on our ablation study results.

Component loss: The high-level policy employs three loss functions: $\mathcal{L}_{rew}$, $\mathcal{L}_{ag}$ and $\mathcal{L}_{dg}$, represented as RL, AGL, and DGL in the Fig. 9c, respectively. Compared to the algorithm with only loss $\mathcal{L}_{rew}$, the addition of loss $\mathcal{L}_{ag}$ can significantly improve performance by generating efficient samples that can be directly



(a) Parameter α (b) Parameter β (c) Component loss

Fig. 9. Ablation studies on different parameters and components: (a) parameter α, (b) parameter β, and (c) component loss. All curves are averaged over 5 trials and smoothed equally for visual clarity. (Color figure online)

used for training without goal-relabeling. By incorporating loss $\mathcal{L}_{dg}$ under the constraint of loss $\mathcal{L}_{ag}$, the algorithm can enhance exploration capability and speed up learning process.

6 Conclusion

We propose a novel method MISD, Mutual Information-based Subgoal Discovery, which utilizes mutual information constraint to identify informative subgoals in hierarchical reinforcement learning. By minimizing the mutual information distance between subgoals and both the achieved and desired goals, MISD effectively enhances sample and training efficiency. The experimental results demonstrate that our method achieves comparable performance to state-of-the-art methods while significantly reducing the training time in various continuous control tasks. Moreover, MISD is dimension-agnostic with respect to the state and action spaces, highlighting its potential applicability in real-world scenarios.

References

1. Andrychowicz, M., et al.: Hindsight experience replay. In: Advances in Neural Information Processing Systems, vol. 30, pp. 5048–5058. Curran Associates, Inc. (2017)
2. Duan, Y., Chen, X., Houthooft, R., Schulman, J., Abbeel, P.: Benchmarking deep reinforcement learning for continuous control. In: Proceedings of The 33rd International Conference on Machine Learning, vol. 48, pp. 1329–1338. PMLR (2016)
3. Eysenbach, B., Gupta, A., Ibarz, J., Levine, S.: Diversity is all you need: learning skills without a reward function. In: International Conference on Learning Representations (2018)
4. Eysenbach, B., Zhang, T., Levine, S., Salakhutdinov, R.R.: Contrastive learning as goal-conditioned reinforcement learning. In: Advances in Neural Information Processing Systems, vol. 35, pp. 35603–35620. Curran Associates, Inc. (2022)
5. Hafner, D., Lee, K.H., Fischer, I., Abbeel, P.: Deep hierarchical planning from pixels. In: Advances in Neural Information Processing Systems. vol. 35, pp. 26091–26104. Curran Associates, Inc. (2022)

6. Hartikainen, K., Geng, X., Haarnoja, T., Levine, S.: Dynamical distance learning for semi-supervised and unsupervised skill discovery. In: International Conference on Learning Representations (2020)
7. Huang, Z., Liu, F., Su, H.: Mapping state space using landmarks for universal goal reaching. In: Advances in Neural Information Processing Systems, vol. 32, p. 1942–1952. Curran Associates, Inc. (2019)
8. Kim, J., Seo, Y., Shin, J.: Landmark-guided subgoal generation in hierarchical reinforcement learning. In: Advances in Neural Information Processing Systems, vol. 34, pp. 28336–28349. Curran Associates, Inc. (2021)
9. Kulkarni, T.D., Narasimhan, K., Saeedi, A., Tenenbaum, J.: Hierarchical deep reinforcement learning: integrating temporal abstraction and intrinsic motivation. In: Advances in Neural Information Processing Systems, vol. 29, pp. 3675–3683. Curran Associates, Inc. (2016)
10. Laskin, M., Srinivas, A., Abbeel, P.: CURL: Contrastive unsupervised representations for reinforcement learning. In: Proceedings of the 37th International Conference on Machine Learning, vol. 119, pp. 5639–5650. PMLR (2020)
11. Levy, A., Konidaris, G., Platt, R., Saenko, K.: Learning multi-level hierarchies with hindsight. In: International Conference on Learning Representations (2019)
12. Nachum, O., Gu, S.S., Lee, H., Levine, S.: Data-efficient hierarchical reinforcement learning. In: Advances in Neural Information Processing Systems, vol. 31, p. 3303–3313. Curran Associates, Inc. (2018)
13. Nasiriany, S., Pong, V., Lin, S., Levine, S.: Planning with goal-conditioned policies. In: Advances in Neural Information Processing Systems, vol. 32, p. 14843–14854. Curran Associates, Inc. (2019)
14. Van Den Oord, A., Li, Y., Vinyals, O.: Representation learning with contrastive predictive coding. arXiv preprint arXiv:1807.03748 (2018)
15. Paninski, L.: Estimation of entropy and mutual information. Neural Comput. **15**(6), 1191–1253 (2003)
16. Pong, V., Gu, S., Dalal, M., Levine, S.: Temporal difference models: model-free deep RL for model-based control. In: International Conference on Learning Representations (2018)
17. Poole, B., Ozair, S., Van Den Oord, A., Alemi, A., Tucker, G.: On variational bounds of mutual information. In: Proceedings of the 36th International Conference on Machine Learning, vol. 97, pp. 5171–5180. PMLR (2019)
18. Sharma, A., Gu, S., Levine, S., Kumar, V., Hausman, K.: Dynamics-aware unsupervised discovery of skills. In: International Conference on Learning Representations (2020)
19. Zhang, Q., Yang, Y., Ruan, J., Xiong, X., Xing, D., Xu, B.: Balancing exploration and exploitation in hierarchical reinforcement learning via latent landmark graphs. In: 2023 International Joint Conference on Neural Networks (IJCNN), pp. 1–8. IEEE (2023)
20. Zhang, T., Guo, S., Tan, T., Hu, X., Chen, F.: Generating adjacency-constrained subgoals in hierarchical reinforcement learning. In: Advances in Neural Information Processing Systems, vol. 33, pp. 21579–21590. Curran Associates, Inc. (2020)