CONCRETE-TO-ABSTRACT GOAL EMBEDDINGS FOR SELF-SUPERVISED REINFORCEMENT LEARNING

Anonymous authors
Paper under double-blind review

ABSTRACT

Self-supervised reinforcement learning (RL) aims to train agents without pre-specified external reward functions, enabling them to autonomously acquire the ability to generalize across tasks. A common substitute for external rewards is the use of observational goals sampled from experience, especially in goal-conditioned RL. However, such goals often constrain the goal space: they may be too concrete (requiring exact pixel-level matches) or too abstract (involving ambiguous observations), depending on the observation structure. Here we propose a unified hierarchical goal space that integrates both concrete and abstract goals. Observation sequences are encoded into this partially ordered space, in which a subset relation naturally induces a hierarchy from concrete to abstract goals. This encoding enables agents to disambiguate specific states while also generalizing to shared concepts. We implement this approach using a recurrent neural network to encode sequences and an energy function to learn the partial order, trained end-to-end with contrastive learning. The energy function then allows to traverse the induced hierarchy to vary the degree of abstraction. In experiments on navigation and robotic manipulation, agents trained with our hierarchical goal space achieve higher task success and greater generalization to novel tasks compared to agents limited to purely observational goals.

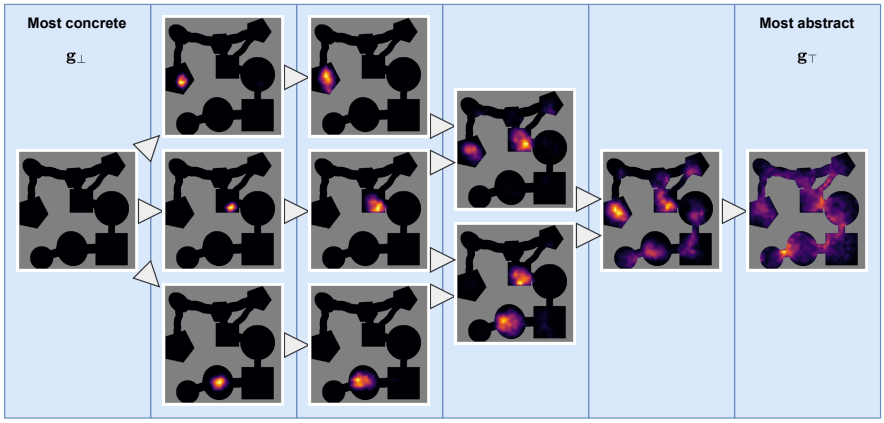


Figure 1: Making goals more abstract in the GridWorld environment.

1 INTRODUCTION

Over the last decade, reinforcement learning (RL) has achieved remarkable successes, both in mastering highly complex games and in domains where environments provide clearly defined reward functions (Mnih et al., 2015; Silver et al., 2016; Vinyals et al., 2019; Hafner et al., 2025). In real-world applications, however, such reward functions are rarely available or are highly non-trivial to specify (Amodei et al., 2016; Russell, 2016; Christiano et al., 2017). In contrast, humans learn about the world through exploration, play, and observation, largely without external reward signals telling them what to do (Begus et al., 2014; Gopnik et al., 1999; Goupil et al., 2016). Another problem

for real-world applications is task-specificity: an agent trained to solve one task may not be able to apply its learned skills to a different, even slightly modified task (Zhang et al., 2018; Delfosse et al., 2025).

One approach to address both the lack of external rewards and the need for generalization is unsupervised RL, in which agents are pretrained without relying on designed reward signals and later adapted to downstream tasks. This pretraining can enable faster inference of policies that generalize across a broad range of tasks. Unsupervised RL is characterized by a spectrum of diverse techniques, including intrinsic motivation approaches based on novelty, learning progress, or empowerment (Salge et al., 2014; Zhang et al., 2021), latent skill learning (Eysenbach et al., 2019; Sharma et al., 2020), goal-conditioned RL with self-selected goals (Nair et al., 2018; Bae et al., 2025), approaches that approximate long-term dynamics with successor measures (Agarwal et al., 2025b;a), and contrastive RL methods (Laskin et al., 2020; Schneider et al., 2021; Eysenbach et al., 2022).

Here, we focus on goal-conditioned RL within a self-supervised learning setting where tasks are specified by the goals the agent attempts to achieve. In the literature goals are often expressed directly in observation space (Ghosh et al., 2021; Eysenbach et al., 2022), which ties the agent's capabilities to the abstraction level as well as the structure of the underlying observation space. Typically, environments either provide local observations like egocentric images, or global observations such as precise states and positions. These modalities inherently bias how goals are interpreted: local observations may be ambiguous, leading to abstract goals unless additional context is provided, while global observations specify concrete states but make it harder to capture higher-level abstractions through composition. Thus, neither local nor global observations alone provide a sufficient basis for representing goals, as each is biased towards a single level of abstraction.

While the most concrete goals can be seen as elements of a sample space composed of observations or sequences of observations, more abstract goals can instead be seen as (soft) partitions on this space. In general, such goals, that consist of multiple observations, can be induced by constraints (Colas et al., 2019), by utility functions (Christiano et al., 2017; Vamplew et al., 2024) or by desired distributions over observational states (Pong et al., 2020; Ziebart et al., 2008). However, if different levels of abstraction are defined by separate functions or constraints (Sutton et al., 1999; Ho et al., 2019), then it might be difficult to capture their relationship to lower-level goals. An alternative is to represent both concrete and abstract goals in a shared latent space, analogous to word embeddings, where both tokens and longer texts are encoded in the same vector space (Mikolov et al., 2013). In the following, we thus introduce a hierarchical latent goal space that represents goals as vectors that encode observation sequences, organized according to varying levels of abstraction. This is achieved through contrastive training of an asymmetric energy function that indicates whether one observation sequence is contained within another. Interestingly, this construction supports join and meet operations on goals, allowing to make them more or less abstract within the hierarchy.

The paper is organized as follows. In Section 2 we introduce our methods, including the latent space encoding, the energy function training and the evaluation methods. For evaluation after pretraining, we confront agents with multiple novel reward functions and we search for the best fitting goal in the trained latent space to represent these reward functions. In Section 3, we illustrate our method in three different RL environments (GridWorld with LiDAR, MemoryMaze, FetchPush). We first illustrate the best fitting goals to represent abstract reward functions, and then show the performance of the corresponding pretrained goal-conditioned policies when searching for the best-fitting goal. In Section 4 we discuss our approach in the context of the wider literature and conclude in Section 5.

2 METHODS

Hierarchical goal space. The key idea is to arrange goals according to their levels of abstraction, forming a spectrum that runs from the most specific to the most general goal. We formalize this idea through a shared latent goal space, $\mathcal{G}$, which supports both representing goals at varying degrees of abstraction and traversing between abstraction levels. In particular, we encode observation sequences instead of single observations, which allows the representation to extract meaningful concrete and abstract patterns by exploiting spatio-temporal similarity as well as integrating shared information. Formally, we capture different levels of abstraction by imposing a partial order $\preceq$ over the latent goal space: If $\mathbf{g}_a \preceq \mathbf{g}_b$, then $\mathbf{g}_b$ represents a more abstract goal than $\mathbf{g}_a$. The most concrete

goals are denoted by $\mathbf{g}_\perp$, and the most abstract by $\mathbf{g}_\top$, such that

$$\mathbf{g}_\perp \preceq \mathbf{g} \preceq \mathbf{g}_\top \quad \forall \mathbf{g} \in \mathcal{G}.$$

Using the lattice structure of this partial order allows to traverse the latent space through join, $\mathbf{g}_a \vee \mathbf{g}_b = \inf\{\mathbf{g} \mid \mathbf{g}_a \preceq \mathbf{g}, \mathbf{g}_b \preceq \mathbf{g}\}$ and meet operations, $\mathbf{g}_a \wedge \mathbf{g}_b = \sup\{\mathbf{g} \mid \mathbf{g} \preceq \mathbf{g}_a, \mathbf{g} \preceq \mathbf{g}_b\}$. Intuitively, the join identifies the least abstract goal that encompasses both $\mathbf{g}_a$ and $\mathbf{g}_b$, while the meet identifies the most concrete goal they share.

Energy function and subset relation. In a self-supervised setting, the data available to the agent only consists of action–observation sequences without any external rewards. To recover useful relations between observations, the agent must therefore rely on the temporal regularities inherent in its interactions with the environment. We therefore propose to treat relations between sequences of observations as proxy for the partial order in goal space. A natural choice of such a relation is the subset relation in sequence space: if one sequence is contained in another, it corresponds to a more concrete goal within the hierarchy. We use a single contrastive learning framework to learn the encoding ϕ_θ into the latent goal space, as well as the characteristic function $\chi_\preceq$ of the partial order. Specifically, we model the latter as an energy $E_\theta(\mathbf{x}, \mathbf{y})$ that estimates whether $\mathbf{x} \preceq \mathbf{y}$, i.e.

$$E_\theta(\mathbf{x}, \mathbf{y}) \approx \chi_\preceq(\mathbf{x}, \mathbf{y}) = \begin{cases} 1 & \mathbf{x} \preceq \mathbf{y} \\ 0 & \text{else.} \end{cases}$$

Making sure that $E_\theta(\phi_\theta(\tau), \phi_\theta(\tau'))$ is close to 1 if τ is a subsequence of $\tau' = (\mathbf{o}_0, \ldots, \mathbf{o}_T)$ and close to 0 otherwise, enables to learn both the latent encoding and the ordering relation end-to-end, guided only by temporal consistency in the data.

Traversing the hierarchy. The induced hierarchy has a one-to-many structure: each concrete goal corresponds to multiple abstract goals that satisfy the partial order, and conversely, an abstract goal generally is more abstract than numerous concrete goals. The differentiability of the energy function allows to traverse this hierarchy by optimization, holding one input fixed while updating the other. In this way, we can move upward to find a more abstract goal ($\uparrow$) or downward to obtain a more concrete goal ($\downarrow$). Given a set of goals $\{\mathbf{g}_i\}_{i=1}^n$, our aim is to find a new goal that is either more abstract or more concrete than the entire set. To generate diverse solutions, we initialize the optimization from a randomly sampled goal, which produces different trajectories over the energy landscape and thus leads to different optimized goals:

$$\mathbf{g}_\uparrow^{(t+1)} = \mathbf{g}_\uparrow^t + \eta \nabla_{\mathbf{g}_\uparrow} \sum_{i=1}^n E(\mathbf{g}_i, \mathbf{g}_\uparrow^t) \qquad \approx \mathbf{g}_1 \vee \cdots \vee \mathbf{g}_n \qquad \textit{(more abstract)}$$

$$\mathbf{g}_\downarrow^{(t+1)} = \mathbf{g}_\downarrow^t + \eta \nabla_{\mathbf{g}_\downarrow} \sum_{i=1}^n E(\mathbf{g}_\downarrow^t, \mathbf{g}_i) \qquad \approx \mathbf{g}_1 \wedge \cdots \wedge \mathbf{g}_n \qquad \textit{(more concrete)}$$

where $\mathbf{g}^0 \sim \mathcal{G}$. Swapping the order of inputs in the energy function determines whether the optimization yields a more abstract or more concrete goal.

Learning the energy function. For practical implementation, we jointly learn the sequence encoding as well as the energy function that induces the partial order, using neural networks trained end-to-end. The overall architecture is shown in Figure 3. To handle sequences of arbitrary length, we use an recurrent neural network (RNN) based encoder (Cho et al., 2014), which maps the sequence τ into a fixed-dimensional context vector $\mathbf{c}_\tau$. Similar to Hafner et al. (2022), we further encode this context vector into a discrete latent representation $\mathbf{g}$ using a categorical encoder, where the discrete bottleneck encourages the formation of higher-level abstractions over goals. Differentiability is maintained by using a straight-through estimation of the gradients with respect to the discrete samples. This setup is closely related to the discrete variational autoencoder (VAE) used by Hafner et al. (2025), but instead of optimizing a reconstruction loss, we train the model with a reconstruction-free objective based on our proposed subset relation, using contrastive learning.

Given recorded data $\mathcal{D}$ consisting of trajectories $\tau = (\mathbf{o}_0, \ldots, \mathbf{o}_T)$, we construct datasets $\mathcal{D}_+$ and $\mathcal{D}_-$ of positive and negative pairs of trajectories, respectively, based on their relation in sequence

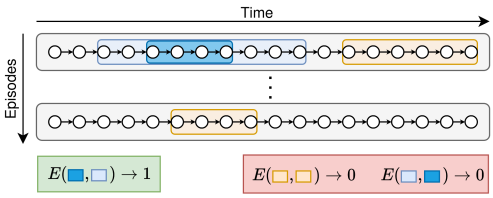


Figure 2: Subset selection for contrastive learning. Positive (green) and negative (red) examples are selected from observation trajectories.

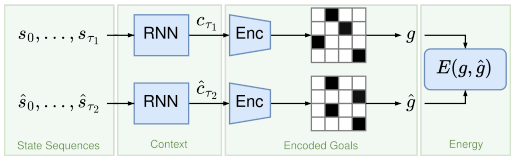


Figure 3: Proposed sequence abstraction model combining a learnable goal representation and similarity measure.

space. The energy function E_θ and sequence encoder ϕ_θ are learned jointly by optimizing

$$\begin{aligned}\mathcal{L}(\theta) = &- \mathbb{E}_{(\tau_1,\tau_2)\sim\mathcal{D}_+}\left[\log E_\theta(\phi_\theta(\tau_1), \phi_\theta(\tau_2))\right] \\ &- \mathbb{E}_{(\tau_1,\tau_2)\sim\mathcal{D}_-}\left[\log(1 - E_\theta(\phi_\theta(\tau_1), \phi_\theta(\tau_2)))\right].\end{aligned}$$

Here, $\mathcal{D}_+$ contains pairs (τ_1, τ_2) such that $\tau_1 \subset \tau_2$, including cases where both come from the same trajectory as well as composite trajectories formed from unrelated segments. In contrast, $\mathcal{D}_-$ contains pairs where $\tau_1 \not\subset \tau_2$, for example pairs (τ_1, τ_2) with $\tau_2 \subset \tau_1$, as well as non-overlapping trajectories, either from the same or from distinct base trajectories τ. Examples are shown in Figure 2.

Abstract hindsight relabeling. To learn policies in our approach, we chose contrastive reinforcement learning (CRL) introduced in Eysenbach et al. (2022), which is based on hindsight relabeling of achieved goals. In standard CRL, relabeling is restricted to single observations from past experience. This limits the diversity of goals available for training, since the agent repeatedly encounters only a narrow subset of the goal space, an effect that becomes more pronounced when training on a fixed, concrete task. Our goal encoding addresses this limitation by allowing entire sequences of observations to be relabeled as a single, more abstract goal. This provides the agent with richer supervision: it can learn from both fine-grained, single-observation goals and higher-level, temporally extended goals. We hypothesize that this broader and more expressive goal set improves generalization and task performance. In practice, we randomly sample the sequence length during training, allowing the agent to experience both concrete and abstract goals. It is important to note, however, that this strategy does not by itself yield a policy that can achieve fully abstract, compositional concepts that are possible through our learned goal abstraction. This higher-order hindsight relabeling merely increases the diversity of goals experienced by the policy (see Conclusion).

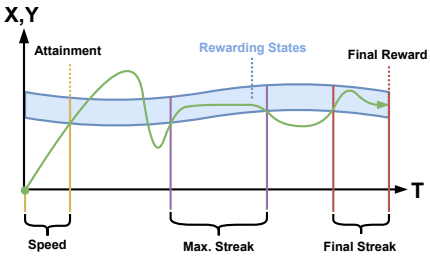
Encoding reward functions. To learn goal representations from reward signals, we employ a hindsight learning procedure that translates rewarding observations into parametrized goals. This procedure uses contrastive learning to combine high-rewarding goals through a similarity measure in goal space:

$$\mathcal{L}(\vartheta) = \mathop{\mathbb{E}}_{(r_t,\mathbf{g}_t)\sim\mathcal{D},\mathbf{g}^*\sim p_\vartheta}\Bigg[\underbrace{-r_t \log E(\mathbf{g}_t, \mathbf{g}^*)}_{\text{hindsight}} \quad - \quad \underbrace{(1 - r_t)\log(1 - E(\mathbf{g}_t, \mathbf{g}^*))}_{\text{negative feedback}}\Bigg],$$

where $\mathcal{D}$ are reward-goal tuples obtained from either environment interactions or from curated data. For our approach, observations are encoded into goals using the sequence encoder, and similarity is computed using the energy function. By optimizing for a more abstract goal than all rewarding goals, we learn a representation of the reward function at hand. In contrast, observational goals use cosine similarity to combine high-reward observations directly.

Generalization to novel rewards. We borrow ideas from zero-shot RL, where we evaluate the agent's behavior on novel, unseen rewards to evaluate the *persuadability* of our agents (Levin, 2022). To compare the performance of observational goals against our proposed hierarchical goal space, we pre-train two agents on the corresponding goal space with the same original, single-observation goal reaching task in mind.

The agent is tasked with exploring the state space autonomously to encode reward functions. For this, the agent collects rewards during fixed length episodes under the policy induced by the current, best fitting goal. As a result, the data is goal-oriented towards reaching high rewarding states which otherwise may be too unlikely to get sufficient data for. However, this may introduce a bias where the agent focuses more on a frequent rewarding states instead of capturing the full reward function. We evaluate the agents behavior under the performance criteria given in Figure 4 to capture different aspects of the induced policy behavior.



Criterion	Description
Attainment	Rewarding state reached
Total Reward	Time fraction at reward
Final Reward	Reward in the final state
Final Streak	End consecutive rewards
Max. Streak	Max consecutive rewards
Avg. Streak	Avg. consecutive rewards
Speed	Steps to first reward

Figure 4: Agent performance criteria.

Goal Visualization. In order to get an intuition of what the goal space actually encodes, we use the learned energy function to visualize goals as follows. First, we sample 10000 observations alongside their position from the environment. Next, we encode them as single-observation goals and compare them in the learned energy function, effectively checking if they are more concrete than some target goal. For observation goals we use cosine similarity instead of the energy function. Finally, an energy heatmap can be plotted by using the position information and computed energies.

3 Experiments

We evaluate our approach on different environments with local and global observations. The agents are pretrained on the original, single observation goal reaching task with observational goals provided by the environment. For our approach, we encode the observations into latent goals. First, we show how our goal representation can be used to traverse between concrete and abstract goals. Next, we analyze the learned goal representation by trying to encode novel reward functions. Finally, we conduct experiments to analyze the induced agent behavior under novel goals.

3.1 Environments

We use two navigation tasks with ego-centric, local observations as well as one robot manipulation task with a global, image based observation space. Further, we define various new reward functions for each environment. Rewards are binary and state-based. A full list of rewards is given in A.2.

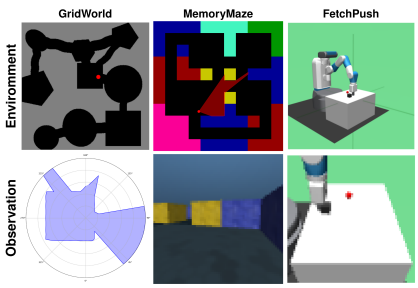


Figure 5: Environments.

GridWorld is a 2D navigation task where agents navigate through a grid-based environment using discrete actions. A new episode starts when the agent reaches the goal. Contrary to normal position-based GridWorld environments, our agent receives egocentric LiDAR observations. For analysis we define observation-based rewards (such as circular room) as well as spatial rewards (specific room, disjoint rooms).

MemoryMaze (Pasukonis et al., 2022) is originally a long-term memory task based on discrete 2D navigation with image observations. We adapt the task to use the environment for goal-based navigation where goals are provided as image observations. Episodes end when the goal is reached. We again define additional observational rewards (such as yellow corners) and spatial rewards (such as a specific room).

FetchPush (Plappert et al., 2018) is a robotic manipulation task where a gripper is tasked to move an object to a specific position and keep it there until the episode ends. We use the adapted task from Eysenbach et al. to obtain image-based observations, making the environment challenging. Goal images show the object at some random position with the gripper close to the object. As novel reward functions we define gripper rewards and object rewards such as moving the gripper to the table border or pushing the object off the table.

3.2 RESULTS

Traversing the hierarchy. Optimizing the energy function to traverse the induced hierarchy proves effective for generating goals at varying levels of abstraction. Figure 1 shows how we can traverse the hierarchy in the GridWorld environment. Starting from the most concrete goal, we can optimize for single-position goals, abstract them to broader regions and even combine individual regions to more abstract, disjoint regions. Finally, the hierarchy is bounded by the most abstract goal. Note that only one possible abstraction is shown at a time, while the partial order allows for different abstractions fulfilling the subset relation. With this observation in mind, we show how we can learn abstract representations of rewards by combining individual goals, corresponding to high rewarding states, into a single abstract goal.

Encoding reward functions. A fundamental challenge in self-supervised reinforcement learning is achieving sufficient exploration to capture the complete state space. Since data collection occurs within the environment, an agent's policy inherently constrains which regions of the state space can be observed. To properly evaluate our pretrained goal representation's ability to encode diverse reward functions, we need comprehensive state coverage that is independent of any particular policy. We construct a dataset by systematically sampling observations and their corresponding rewards across the environment to ensure uniform coverage of the entire state space. Using this unbiased dataset, we then optimize goals to encode high-reward states, effectively capturing diverse reward functions through the methodology detailed in Section 2. This optimization process is applied to both observational goals and our abstract goals, enabling direct comparison of their representational capabilities. Figure 6 demonstrates our goal representation's ability to encode diverse reward

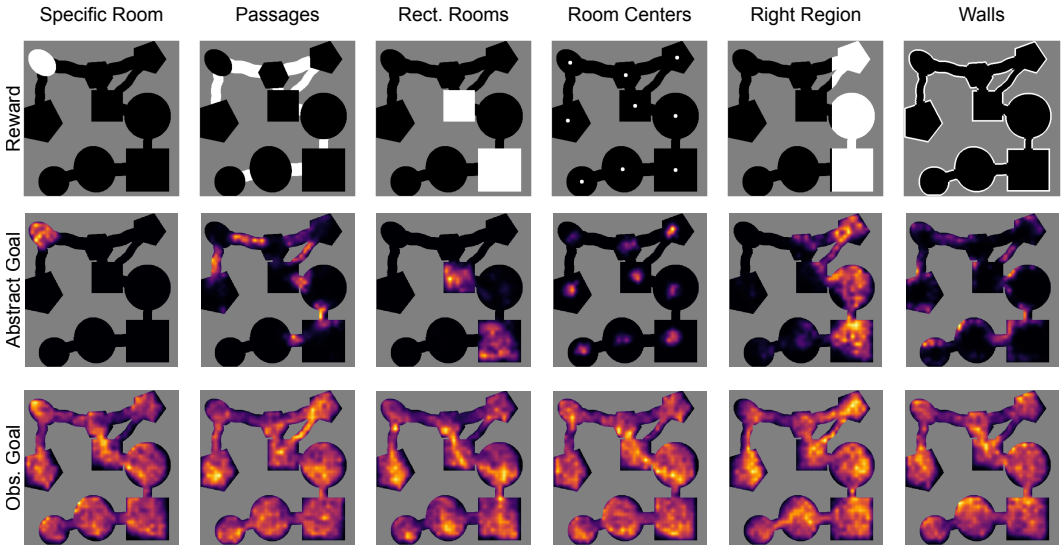


Figure 6: Learned goal spaces from diverse data in the GridWorld environment. White regions in the top row indicate high rewarding states. There is a single abstract goal to represent each considered reward but no single observation.

functions in the GridWorld environment, ranging from concrete spatial targets (specific rooms) to increasingly abstract concepts (room centers, disjoint rooms). This spectrum reveals critical differences in representational requirements: while some rewards correspond directly to observable features (e.g., passages that can be identified from single observations), others demand spatial and

compositional knowledge of the environment. Our analysis reveals limitations with respect to observational goals. While raw observations can partially encode simple, visually-identifiable objectives like corridor passages, they fail to capture spatial relationships and compositional properties effectively. Furthermore, the optimized observations show overall high and noisy activity in cosine similarity, indicating fundamental limitations in their capacity to serve as robust abstract goal representations. More results supporting our interpretaion in the MemoryMaze and FetchPush environments are given in subsection A.1.

Adaptation to novel rewards. We extend our analysis to a more realistic task by considering data obtained by environment interactions instead of relying on full state coverage. This is accomplished by using the optimized goal as a target for the agent while refining the goal. This methodology allows the agent to bias the observed state space down to meaningful observations in order to solve the task more efficiently. Consequently, the goal is not necessarily required to encode the full reward function. Figure 7 presents a comparative analysis of agents trained with abstract goal represen-

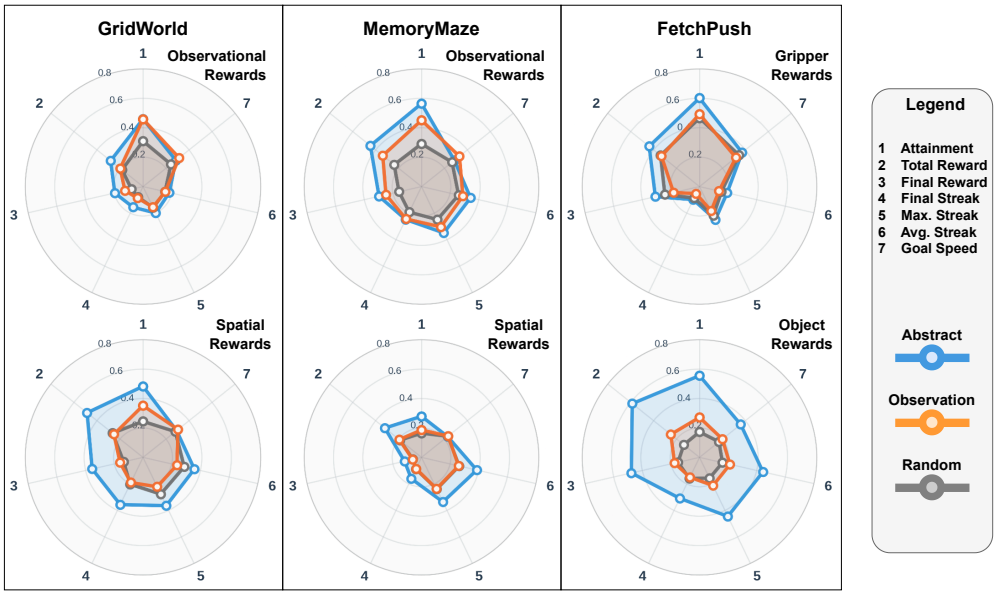


Figure 7: Agent performance on novel rewards for all considered environments.

tations versus observational goal representations, evaluated according to the performance criteria outlined in Figure 4. Our findings reveal comparable performance between both agent types on observational tasks, where the available observations are sufficient to encode the reward function. However, a difference emerges on spatial tasks that demand both spatial and compositional knowledge about the environment. Our goal representation demonstrates better performance across most evaluation metrics, while agents using observational goals exhibit near-random performance. This performance gap indicates that agents trained with our goal representation have better generalization capabilities for abstract goals that cannot be adequately represented within the observation space. In the FetchPush environment, where all observations are global, we analyze the agents' behavior on abstract gripper and object-based rewards. We deliberately excluded concrete spatial objectives (such as positioning objects or grippers at specific coordinates) due to their extremely low success probability without sufficient exploration. Our results demonstrate that only our goal representation successfully handles both abstract object manipulation and gripper control tasks. In contrast, observational goals fail on both tasks.

4 Related Work

Self-supervised representation learning. Self-supervised representation learning tries to learn meaningful representations without explicit supervision signals like labels (Ericsson et al., 2021).

Contrastive learning has emerged as a powerful paradigm for self-supervised representation learning by augmenting data with labels, often obtained by defining a suitable similarity measure between data points (Jaiswal et al., 2021). The core principle involves bringing semantically similar samples closer together while pushing dissimilar samples apart in the learned embedding space. This approach has shown remarkable success across domains, including computer vision with methods like SimCLR (Chen et al., 2020) as well as natural language processing through approaches like contrastive sentence representation learning (Kim et al., 2021) and supervised contrastive learning for pre-trained language model fine-tuning (Gunel et al., 2021). Temporal contrastive learning extends these principles to sequential data by leveraging temporal relationships as augmentations. Contrastive learning through time (Schneider et al., 2021) takes inspiration from biology to learn object representations by forming augmentations from successive views in temporal sequences. Contrastive learning can be understood through the lens of energy-based models, where the similarity measures used to bring positive pairs together and push negative pairs apart implicitly form an energy landscape over the representation space (LeCun et al., 2006). Such energy functions can be used to learn compositional concepts as shown in Du et al. (2021). Our approach combines the idea of using temporal contrastive learning in combination with an energy function to guide representation learning. Moreover, we extend temporal contrastive learning to not only using temporal relations between individual images but to temporal relations between whole trajectories through the introduced subset similarity in sequence space.

Abstraction in RL. The complexity of most RL tasks makes abstraction indispensable. The literature mainly distinguishes state abstraction and temporal abstraction (Abel, 2020). State abstraction reduces the size of the state space by engineering or learning low-dimensional features from raw sensory inputs (Mnih et al., 2015). Often this is achieved with reconstruction-based compression methods like VAEs that do not always focus on relevant features (Ha & Schmidhuber, 2018; Hafner et al., 2025; 2019). Hence, there has been a flurry of reconstruction-free compression methods based on contrastive learning (InfoNCE and contrastive predictive coding (Oord et al., 2018; Ma & Collins, 2018), or DeepInfoMax (Hjelm et al., 2019)). Our proposed method for goal embeddings follows this line of research, but adds the pre-order structure on the latent space that is absent in previous methods. As we encode observation trajectories there is also some level of temporal abstraction that we did not explore in the current study. In the future, it will be interesting to pursue this avenue in the context of hierarchical RL (Vezhnevets et al., 2017; Nachum et al., 2018) where concrete and abstract goal vectors could be used to communicate between different agents in the hierarchy. Unsupervised pretraining goal-conditioned policies with such goals could also form the basis to discover a diverse set of skills (Eysenbach et al., 2019).

Representation learning in self-supervised RL. The concepts of goals, skills, and intentions share fundamental similarities as they all represent desired outcomes or behaviors that guide agent decision-making. Ghosh et al. (2023) learns intention-conditioned value functions by encoding how outcome likelihoods change when the policy acts with a particular intention in mind. Skill based methods aim to find latent representations of reproducible behavior (Eysenbach et al., 2019; Sharma et al., 2020). To find latent skills, information theoretic ideas are employed to maximize mutual information between states and skills while making skills distinguishable (Eysenbach et al., 2019), between skills and future outcomes (Sharma et al., 2020) or between skills and state transitions (Laskin et al., 2022). Another way to encode diverse behavior is to exploit the linear dependence of the Q value function on the reward Touati & Ollivier (2021); Agarwal et al. (2025b). Agarwal et al. (2025b) show that any agent behavior which can be represented by visitation distributions can be described as a affine combinations of policy independent basis functions. While our current embedding space is based on observational sequences and thus more related to goals, extending this approach to sequences of actions or action and observation sequences may support a representation that forms a bridge between goals and skills.

Goal-conditioned RL. Goal-conditioned reinforcement learning enables agents to learn diverse policies by conditioning behavior on desired goals. As the overall objective is attaining a goal, rewards are in general sparse which poses a fundamental problem. Hindsight experience replay (Andrychowicz et al., 2017) and in general hindsight relabeling proved to be an effective method to tackle sparsity by relabeling failed trajectories as success under a different goal Andrychowicz et al. (2017); Ghosh et al. (2021). Different extensions to this idea where proposed to deal with

dynamics goals (Ren et al., 2019) or prioritize the experience for better relabeling (Zhao & Tresp, 2018). Ghosh et al. (2021) rephrase the goal-conditioned learning problem as supervised learning without rewards, by relabeling experiences trajectories as success and using self-imitation learning to directly optimize the policy. Eysenbach et al. (2022) demonstrated that contrastive learning can be reinterpreted as goal-conditioned reinforcement learning, showing that contrastive objectives naturally lead to goal-reaching behavior. In our work we simply used existing goal-conditioned RL approaches, in particular contrastive methods, to learn policies. The innovation of our work focuses on the representation of the goal space that can be used by such methods.

Goal representation learning While observations are commonly used as goals, various approaches for learning latent goal representations have been proposed. Nair et al. (2018) employ a VAE with reconstruction loss to embed observations into a latent space, enabling the sampling of novel goals and computation of distances in latent space. Building on this foundation, Nair et al. (2020) extend the approach using a conditional VAE that incorporates future goals to encode goal feasibility. Co-Reyes et al. (2018) take a different approach by encoding entire trajectories into latent representations using a VAE trained with reconstruction loss, subsequently training a policy conditioned on these latent trajectories to replicate the encoded sequences. Hafner et al. (2022) utilizes discrete latent sub-goals derived from a discrete VAE applied to world model states to guide reward optimization. We built on both ideas, encoding trajectories into latent representations and using a discrete VAE for encoding goals but in contrast to other approaches, we use reconstruction only to facilitate initial convergence before fully transitioning to contrastive learning.

Evaluation. Self-supervised RL methods like goal-conditioned RL and successor feature methods are often used to study zero-shot RL (Schaul et al., 2015; Barreto et al., 2018), i.e. instant generalization to unseen tasks. While we also evaluated our method in terms of adaptation to novel reward functions, our policies where not optimized during pretraining for abstract goals. Instead, we focused on representation learning of the goal space in the energy function and showed that there is some emergent capability of our policies to deal with abstract goals despite being trained exclusively on concrete goals. In the future it will also be interesting to compare our method to other zero-shot RL methods, but for this comparison to be meaningful our policies should be also pretrained with abstract goals, which requires a kind of curriculum, which was not the purpose of the current study.

5 Conclusion

In this paper, we introduce a novel approach for representing goals as embedding vectors in a latent space with varying levels of abstraction for self-supervised reinforcement learning. Existing methods typically define goals either through hand-engineered, goal-dependent reward functions or directly in terms of observations, thereby constraining the level of abstraction to the properties of the observation space. Our proposed method addresses this limitation by encoding sequences of observations into a latent goal space, learned in an unsupervised manner with contrastive learning, where a partial order naturally induces a hierarchy of abstraction. Traversing this hierarchy in our goal space leads to more abstract, unseen goals which can be exploited to encode novel reward functions as goals. Through experiments in navigation and robotic manipulation, we have demonstrated that agents trained with our hierarchical goal space achieve higher task success and significantly greater generalization to novel, unseen tasks compared to agents reliant on purely observational goals.

While our goal representation is shown to be effective at encoding a variety of reward functions, there are several remaining challenges. Currently, the agent's policy is only trained with concrete goals during pretraining. Accordingly, it is not surprising that results show a discrepancy between what the goal representation can encode and what a pretrained agent can actually achieve with these abstract goals. Therefore, future work should focus on developing a more effective mechanism for training agents to fully utilize these rich, abstract representations. The concepts of compositionality and multi-level abstraction could also be explored further, particularly in the context of hierarchical RL where goal vectors with multiple levels of abstraction could be easily communicated between different modules. Ultimately, our goal representation approach can be used in conjunction with other RL methods in an unsupervised fashion to foster learning and generalization in the absence of explicit reward functions, a critical step toward applying reinforcement learning to open-ended environments.

REPRODUCIBILITY STATEMENT

We commit to release code upon acceptance. Code will be made available to the reviewers via an anonymous repository. Our implementation includes network architectures and code to learn the proposed goal representation as well as helper scripts for visualization. Hyper-parameters and architectural details are specified in the appendix A.3.

REFERENCES

David Abel. *A theory of abstraction in reinforcement learning*. Phd thesis, Brown University, May 2020.

Siddhant Agarwal, Caleb Chuck, Harshit Sikchi, Jiaheng Hu, Max Rudolph, Scott Niekum, Peter Stone, and Amy Zhang. A unified framework for unsupervised reinforcement learning algorithms. In *Workshop on Reinforcement Learning Beyond Rewards@ Reinforcement Learning Conference 2025*, 2025a.

Siddhant Agarwal, Harshit Sikchi, Peter Stone, and Amy Zhang. Proto successor measure: Representing the behavior space of an RL agent. In *Forty-second International Conference on Machine Learning*, 2025b. URL https://openreview.net/forum?id=mUDnPzopZF.

Dario Amodei, Chris Olah, Jacob Steinhardt, Paul Christiano, John Schulman, and Dan Mané. Concrete problems in ai safety. *arXiv preprint arXiv:1606.06565*, 2016.

Marcin Andrychowicz, Filip Wolski, Alex Ray, Jonas Schneider, Rachel Fong, Peter Welinder, Bob McGrew, Josh Tobin, OpenAI Pieter Abbeel, and Wojciech Zaremba. Hindsight experience replay. In I. Guyon, U. Von Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett (eds.), *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017. URL https://proceedings.neurips.cc/paper_files/paper/2017/file/453fadbd8a1a3af50a9df4df899537b5-Paper.pdf.

Junik Bae, Kwanyoung Park, and Youngwoon Lee. Tldr: Unsupervised goal-conditioned rl via temporal distance-aware representations. In *Conference on Robot Learning*, pp. 2183–2204. PMLR, 2025.

Andre Barreto, Diana Borsa, John Quan, Tom Schaul, David Silver, Matteo Hessel, Daniel Mankowitz, Augustin Zidek, and Remi Munos. Transfer in deep reinforcement learning using successor features and generalised policy improvement. In *International Conference on Machine Learning*, pp. 501–510. PMLR, 2018.

Katarina Begus, Teodora Gliga, and Victoria Southgate. Infants learn what they want to learn: Responding to infant pointing leads to superior learning. *PloS one*, 9(10):e108817, 2014.

Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *Proceedings of the 37th International Conference on Machine Learning*, pp. 1597–1607. PMLR, 2020.

Kyunghyun Cho, Bart van Merrienboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. Learning phrase representations using rnn encoder-decoder for statistical machine translation, 2014. URL https://arxiv.org/abs/1406.1078.

Paul F Christiano, Jan Leike, Tom Brown, Miljan Martic, Shane Legg, and Dario Amodei. Deep reinforcement learning from human preferences. *Advances in neural information processing systems*, 30, 2017.

John Co-Reyes, YuXuan Liu, Abhishek Gupta, Benjamin Eysenbach, Pieter Abbeel, and Sergey Levine. Self-consistent trajectory autoencoder: Hierarchical reinforcement learning with trajectory embeddings. In *International conference on machine learning*, pp. 1009–1018. PMLR, 2018.

Cédric Colas, Pierre Fournier, Mohamed Chetouani, Olivier Sigaud, and Pierre-Yves Oudeyer. Curious: intrinsically motivated modular multi-goal reinforcement learning. In *International conference on machine learning*, pp. 1331–1340. PMLR, 2019.

Quentin Delfosse, Jannis Blüml, Fabian Tatai, Théo Vincent, Bjarne Gregori, Elisabeth Dillies, Jan Peters, Constantin A. Rothkopf, and Kristian Kersting. Deep reinforcement learning agents are not even close to human intelligence. In *Eighteenth European Workshop on Reinforcement Learning*, 2025. URL https://openreview.net/forum?id=TqGnZcXGGJ.

Yilun Du, Shuang Li, Yash Sharma, Joshua B. Tenenbaum, and Igor Mordatch. Unsupervised learning of compositional energy concepts. *CoRR*, abs/2111.03042, 2021. URL https://arxiv.org/abs/2111.03042.

Linus Ericsson, Henry Gouk, Chen Change Loy, and Timothy M. Hospedales. Self-supervised representation learning: Introduction, advances and challenges. *CoRR*, abs/2110.09327, 2021. URL https://arxiv.org/abs/2110.09327.

Benjamin Eysenbach, Abhishek Gupta, Julian Ibarz, and Sergey Levine. Diversity is all you need: Learning skills without a reward function. In *International Conference on Learning Representations*, 2019. URL https://openreview.net/forum?id=SJx63jRqFm.

Benjamin Eysenbach, Tianjun Zhang, Sergey Levine, and Russ R Salakhutdinov. Contrastive learning as goal-conditioned reinforcement learning. In S. Koyejo, S. Mohamed, A. Agarwal, D. Belgrave, K. Cho, and A. Oh (eds.), *Advances in Neural Information Processing Systems*, volume 35, pp. 35603–35620. Curran Associates, Inc., 2022. URL https://proceedings.neurips.cc/paper_files/paper/2022/file/e7663e974c4ee7a2b475a4775201ce1f-Paper-Conference.pdf.

Dibya Ghosh, Abhishek Gupta, Ashwin Reddy, Justin Fu, Coline Manon Devin, Benjamin Eysenbach, and Sergey Levine. Learning to reach goals via iterated supervised learning. In *International Conference on Learning Representations*, 2021. URL https://openreview.net/forum?id=rALA0Xo6yNJ.

Dibya Ghosh, Chethan Anand Bhateja, and Sergey Levine. Reinforcement learning from passive data via latent intentions. In *International Conference on Machine Learning*, pp. 11321–11339. PMLR, 2023.

Alison Gopnik, Andrew N Meltzoff, and Patricia K Kuhl. *The scientist in the crib: Minds, brains, and how children learn.* William Morrow & Co, 1999.

Louise Goupil, Margaux Romand-Monnier, and Sid Kouider. Infants ask for help when they know they don't know. *Proceedings of the National Academy of Sciences*, 113(13):3492–3496, 2016.

Beliz Gunel, Jingfei Du, Alexis Conneau, and Ves Stoyanov. Supervised contrastive learning for pre-trained language model fine-tuning. In *International Conference on Learning Representations*, 2021.

David Ha and Jürgen Schmidhuber. Recurrent world models facilitate policy evolution. *Advances in neural information processing systems*, 31, 2018.

Tuomas Haarnoja, Aurick Zhou, Pieter Abbeel, and Sergey Levine. Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor. In Jennifer G. Dy and Andreas Krause (eds.), *ICML*, volume 80 of *Proceedings of Machine Learning Research*, pp. 1856–1865. PMLR, 2018. URL http://dblp.uni-trier.de/db/conf/icml/icml2018.html#HaarnojaZAL18.

Danijar Hafner, Timothy Lillicrap, Jimmy Ba, and Mohammad Norouzi. Dream to control: Learning behaviors by latent imagination. *arXiv preprint arXiv:1912.01603*, 2019.

Danijar Hafner, Kuang-Huei Lee, Ian Fischer, and Pieter Abbeel. Deep hierarchical planning from pixels. *Advances in Neural Information Processing Systems*, 2022.

Danijar Hafner, Jurgis Pasukonis, Jimmy Ba, and Timothy Lillicrap. Mastering diverse control tasks through world models. *Nature*, pp. 1–7, 2025.

R Devon Hjelm, Alex Fedorov, Samuel Lavoie-Marchildon, Karan Grewal, Phil Bachman, Adam Trischler, and Yoshua Bengio. Learning deep representations by mutual information estimation and maximization. In *International Conference on Learning Representations*, 2019. URL https://openreview.net/forum?id=Bklr3j0cKX.

Mark K Ho, David Abel, Thomas L Griffiths, and Michael L Littman. The value of abstraction. *Current opinion in behavioral sciences*, 29:111–116, 2019.

Ashish Jaiswal, Ashwin Ramesh Babu, Mohammad Zaki Zadeh, Debapriya Banerjee, and Fillia Makedon. A survey on contrastive self-supervised learning. *Technologies*, 9(1), 2021. ISSN 2227-7080. doi: 10.3390/technologies9010002. URL https://www.mdpi.com/2227-7080/9/1/2.

Taeuk Kim, Kang Min Yoo, and Sang-goo Lee. Self-guided contrastive learning for bert sentence representations. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics*, pp. 2528–2540, 2021.

Durk P Kingma, Tim Salimans, Rafal Jozefowicz, Xi Chen, Ilya Sutskever, and Max Welling. Improved variational inference with inverse autoregressive flow. *Advances in neural information processing systems*, 29, 2016.

Michael Laskin, Aravind Srinivas, and Pieter Abbeel. Curl: Contrastive unsupervised representations for reinforcement learning. In *International conference on machine learning*, pp. 5639–5650. PMLR, 2020.

Michael Laskin, Hao Liu, Xue Bin Peng, Denis Yarats, Aravind Rajeswaran, and Pieter Abbeel. Cic: Contrastive intrinsic control for unsupervised skill discovery. In *Advances in Neural Information Processing Systems*, 2022.

Yann LeCun, Sumit Chopra, Raia Hadsell, M Ranzato, and F Huang. A tutorial on energy-based learning. 2006.

Michael Levin. Technological approach to mind everywhere: An experimentally-grounded framework for understanding diverse bodies and minds. *Frontiers in Systems Neuroscience*, Volume 16 - 2022, 2022. ISSN 1662-5137. doi: 10.3389/fnsys.2022.768201.

Zhuang Ma and Michael Collins. Noise contrastive estimation and negative sampling for conditional models: Consistency and statistical efficiency. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pp. 3698–3707, 2018.

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*, 2013.

Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Alex Graves, Ioannis Antonoglou, Daan Wierstra, and Martin Riedmiller. Playing atari with deep reinforcement learning, 2013. URL https://arxiv.org/abs/1312.5602.

Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Andrei A Rusu, Joel Veness, Marc G Bellemare, Alex Graves, Martin Riedmiller, Andreas K Fidjeland, Georg Ostrovski, et al. Human-level control through deep reinforcement learning. *nature*, 518(7540):529–533, 2015.

Ofir Nachum, Shixiang Shane Gu, Honglak Lee, and Sergey Levine. Data-efficient hierarchical reinforcement learning. *Advances in neural information processing systems*, 31, 2018.

Ashvin Nair, Shikhar Bahl, Alexander Khazatsky, Vitchyr Pong, Glen Berseth, and Sergey Levine. Contextual imagined goals for self-supervised robotic learning. In *Conference on Robot Learning*, pp. 530–539. PMLR, 2020.

Ashvin V Nair, Vitchyr Pong, Murtaza Dalal, Shikhar Bahl, Steven Lin, and Sergey Levine. Visual reinforcement learning with imagined goals. *Advances in neural information processing systems*, 31, 2018.

Aaron van den Oord, Yazhe Li, and Oriol Vinyals. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748*, 2018.

Jurgis Pasukonis, Timothy Lillicrap, and Danijar Hafner. Evaluating long-term memory in 3d mazes. *arXiv preprint arXiv:2210.13383*, 2022.

Matthias Plappert, Marcin Andrychowicz, Alex Ray, Bob McGrew, Bowen Baker, Glenn Powell, Jonas Schneider, Josh Tobin, Maciek Chociej, Peter Welinder, Vikash Kumar, and Wojciech Zaremba. Multi-goal reinforcement learning: Challenging robotics environments and request for research, 2018.

Vitchyr Pong, Murtaza Dalal, Steven Lin, Ashvin Nair, Shikhar Bahl, and Sergey Levine. Skew-fit: State-covering self-supervised reinforcement learning. In *International Conference on Machine Learning*, pp. 7783–7792. PMLR, 2020.

Zhizhou Ren, Kefan Dong, Yuan Zhou, Qiang Liu, and Jian Peng. Dher: Hindsight experience replay for dynamic goals. In *International Conference on Learning Representations*, 2019.

Stuart Russell. Should we fear supersmart robots. *Scientific American*, 314(6):58–59, 2016.

Christoph Salge, Cornelius Glackin, and Daniel Polani. Empowerment–an introduction. In *Guided Self-Organization: Inception*, pp. 67–114. Springer, 2014.

Tom Schaul, Daniel Horgan, Karol Gregor, and David Silver. Universal value function approximators. In Francis Bach and David Blei (eds.), *Proceedings of the 32nd International Conference on Machine Learning*, volume 37 of *Proceedings of Machine Learning Research*, pp. 1312–1320, Lille, France, 07–09 Jul 2015. PMLR. URL https://proceedings.mlr.press/v37/schaul15.html.

Felix Schneider, Xia Xu, Markus Roland Ernst, Zhengyang Yu, and Jochen Triesch. Contrastive learning through time. In *SVRHM 2021 Workshop @ NeurIPS*, 2021.

Archit Sharma, Shixiang Gu, Sergey Levine, Vikash Kumar, and Karol Hausman. Dynamics-aware unsupervised discovery of skills. In *International Conference on Learning Representations*, 2020.

David Silver, Aja Huang, Chris J Maddison, Arthur Guez, Laurent Sifre, George Van Den Driessche, Julian Schrittwieser, Ioannis Antonoglou, Veda Panneershelvam, Marc Lanctot, et al. Mastering the game of go with deep neural networks and tree search. *nature*, 529(7587):484–489, 2016.

Richard S Sutton, Doina Precup, and Satinder Singh. Between mdps and semi-mdps: A framework for temporal abstraction in reinforcement learning. *Artificial intelligence*, 112(1-2):181–211, 1999.

Ahmed Touati and Yann Ollivier. Learning one representation to optimize all rewards. *CoRR*, abs/2103.07945, 2021. URL https://arxiv.org/abs/2103.07945.

Peter Vamplew, Cameron Foale, Conor F Hayes, Patrick Mannion, Enda Howley, Richard Dazeley, Scott Johnson, Johan Källström, Gabriel Ramos, Roxana Radulescu, et al. Utility-based reinforcement learning: Unifying single-objective and multi-objective reinforcement learning. In *Proceedings of the 23rd International Conference on Autonomous Agents and Multiagent Systems*, pp. 2717–2721, 2024.

Alexander Sasha Vezhnevets, Simon Osindero, Tom Schaul, Nicolas Heess, Max Jaderberg, David Silver, and Koray Kavukcuoglu. FeUdal networks for hierarchical reinforcement learning. In Doina Precup and Yee Whye Teh (eds.), *Proceedings of the 34th International Conference on Machine Learning*, volume 70 of *Proceedings of Machine Learning Research*, pp. 3540–3549. PMLR, 06–11 Aug 2017. URL https://proceedings.mlr.press/v70/vezhnevets17a.html.

Oriol Vinyals, Igor Babuschkin, Wojciech M. Czarnecki, Michaël Mathieu, Andrew Dudzik, Junyoung Chung, David H. Choi, Richard Powell, Timo Ewalds, Petko Georgiev, Junhyuk Oh, Dan Horgan, Manuel Kroiss, Ivo Danihelka, Aja Huang, Laurent Sifre, Trevor Cai, John P. Agapiou, Max Jaderberg, Alexander S. Vezhnevets, Rémi Leblond, Tobias Pohlen, Valentin Dalibard, David Budden, Yury Sulsky, James Molloy, Tom L. Paine, Caglar Gulcehre, Ziyu Wang, Tobias Pfaff, Yuhuai Wu, Roman Ring, Dani Yogatama, Dario Wünsch, Katrina McKinney, Oliver Smith, Tom Schaul, Timothy Lillicrap, Koray Kavukcuoglu, Demis Hassabis, Chris Apps, and David Silver. Grandmaster level in StarCraft II using multi-agent reinforcement learning. *Nature*, 575(7782): 350–354, November 2019. ISSN 1476-4687. doi: 10.1038/s41586-019-1724-z.

Chiyuan Zhang, Oriol Vinyals, Remi Munos, and Samy Bengio. A study on overfitting in deep reinforcement learning. *arXiv preprint arXiv:1804.06893*, 2018.

Jin Zhang, Jianhao Wang, Hao Hu, Tong Chen, Yingfeng Chen, Changjie Fan, and Chongjie Zhang. Metacure: Meta reinforcement learning with empowerment-driven exploration. In *International Conference on Machine Learning*, pp. 12600–12610. PMLR, 2021.

Rui Zhao and Volker Tresp. Energy-based hindsight experience prioritization. *arXiv preprint arXiv:1810.01363*, 2018.

Chongyi Zheng, Benjamin Eysenbach, Homer Rich Walke, Patrick Yin, Kuan Fang, Ruslan Salakhutdinov, and Sergey Levine. Stabilizing contrastive RL: Techniques for robotic goal reaching from offline data. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=Xkf2EBj4w3.

Brian D. Ziebart, Andrew Maas, J. Andrew Bagnell, and Anind K. Dey. Maximum entropy inverse reinforcement learning. In *Proceedings of the 23rd National Conference on Artificial Intelligence - Volume 3*, AAAI'08, pp. 1433–1438. AAAI Press, 2008. ISBN 9781577353683.

A Appendix

A.1 Additional Results

Here we present additional results on the representative power of our goal representation. First, we consider the MemoryMaze environment where we learn abstract goals from image observations by using comprehensive data sampled uniformly from the environment. Our goal representation is able to encode all considered goals, ranging from observational goals like colored corners or walls to more abstract spatial goals combining different rooms as shown in Figure 8. In contrast, observational goals struggle with encoding spatial concepts like a specific room but are able to encode observational properties like colored walls or corners. For the FetchPush environment, only

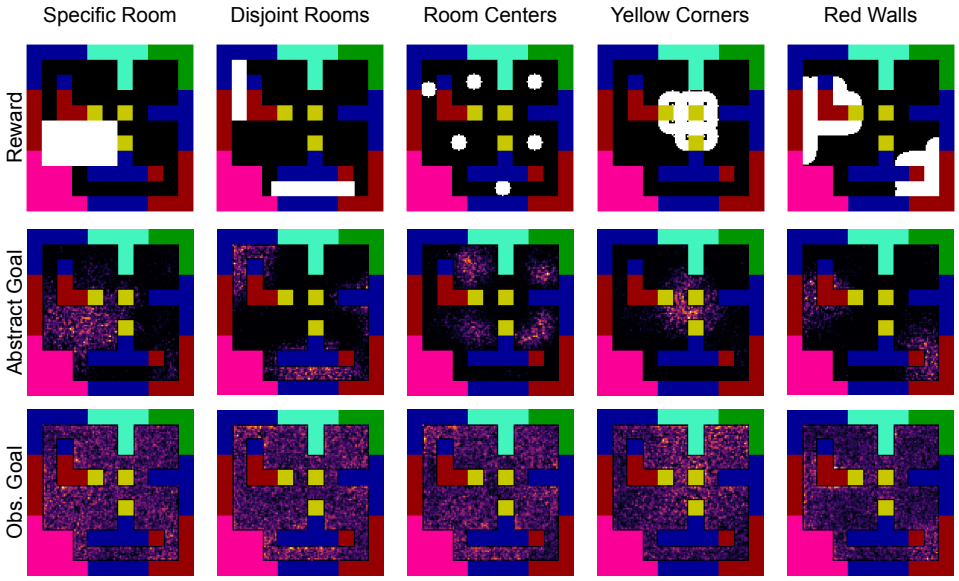


Figure 8: Learned goal spaces from diverse data in the MemoryMaze environment. White regions in the top row indicate high rewarding states.

spatial goals regarding the gripper and object are considered (see Figure 9. While our method is able to encode both, gripper based (specific regions, off-table) and object based (specific regions, off-table) rewards, the observational goals are not able to capture any meaningful structure.

A.2 Environment specific rewards

We add novel reward function to the GridWorld, MemoryMaze and FetchPush environments. Table 1 shows an overview of all rewards used for evaluation. For the GridWorld and MemoryMaze environments, rewards are categorized as spatial (requiring positional and compositional knowledge) or observational (encodable from single observations). For the FetchPush environment, reward functions focus on spatial properties of the gripper and object as the observations contain solely global information.

A.3 Model Architectures & Hyperparameters

Contrastive RL. We base our implementation of CRL (Eysenbach et al., 2022) on the code provided by Zheng et al. (2024). For all experiments we use a two layer multi-layer perceptron (MLP) based policy with 256 hidden units each and ReLU activation. For image based environments, all images are scaled to $64 \times 64 \times 3$. We encode images to latent representation size of 256 using a convolutional neural network (CNN) based encoder as proposed in Mnih et al. (2013). The encoders in the parametrized Q-value function consist of MLPs of two hidden layers with 256 units and ReLU activation, projecting down to a representation dimension of 32. Note that the encoder consists of two separate networks for state-action and goal encoding. For experiments using continuous actions

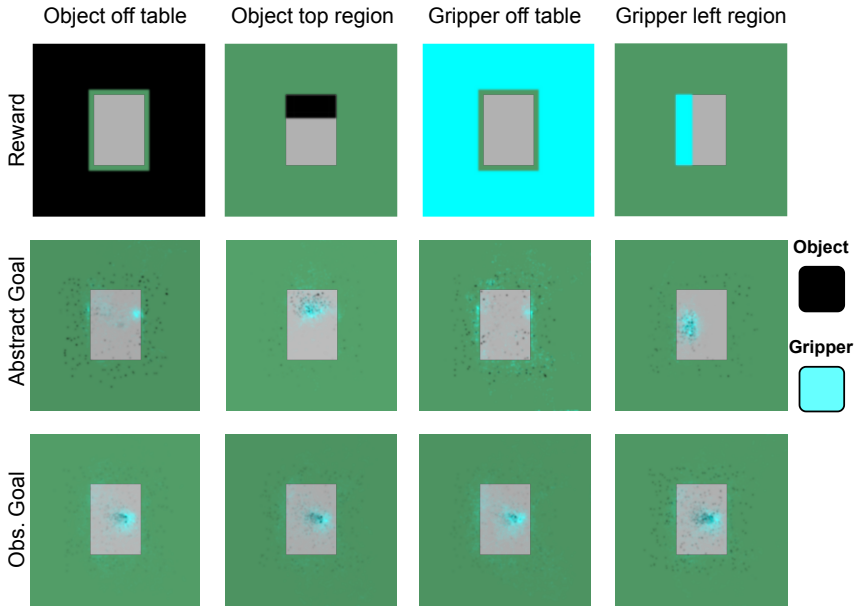


Figure 9: Learned goal spaces from diverse data in the FetchPush environment (top-down view). Black regions indicate high rewarding states for the object while cyan regions are used for gripper rewards.

Environment	Category	Rewards	Description
GridWorld	Spatial	Individual rooms Disjoint rooms (two rooms) Regions (top, down, left, right)	Requires spatial knowledge
	Observational	Room centers Shaped rooms (elliptical, polygonal, . . .) Close to wall Passages (between rooms)	Possible to encode by a single observation
MemoryMaze	Spatial	Individual rooms Disjoint rooms (two rooms)	Requires spatial knowledge
	Observational	Looking at walls (any wall, red wall) Looking at colored corners (blue, yellow)	Possible to encode by a single observation
FetchPush	Gripper-based	Off table On table border In table region (top, left, bottom, right)	Focus on gripper representation encoding
	Object-based	Off table On table border In table region (top, left, bottom, right)	Focus on object representation encoding

Table 1: Environment-specific reward functions.

we use adaptive entropy regularization as proposed in Haarnoja et al. (2018) with a target entropy of 0.0. For discrete actions we bound the maximum D_{KL} between policy and a uniform prior policy by 1 via minimization ("free-bits" trick proposed in Kingma et al. (2016) and used by Hafner et al. (2025)). Batch-sizes vary depending on the problem and are mostly limited by memory: 128 for image based tasks and 2048 for all other experiments. We try to maximize batch sizes as Zheng et al. (2024) showed that in general larger batch sizes are desired for CRL.

Goal Abstraction. We encode sequences with a gated recurrent unit (Cho et al., 2014) with a hidden state size of 256. After encoding the whole sequence we use the last hidden state and use a discrete VAE (Hafner et al., 2022; 2025) using two hidden layers to project the RNN hidden state to

a multi-categorical distribution. We use 16 categories with 16 possible values, resulting in a one-hot encoding of shape 16×16. For image based observations we first encode the images using the CNN described in Mnih et al. (2013) to obtain a 256 dimensional encoding and apply our architecture to the encoded images. The energy function is parametrized as a simple 3 layer neural network with ReLU activation and a hidden dimension of 256. We use binary cross entropy as loss to train the energy function and a fixed batch size of 256 for all experiments.

Reward encoding. While rewards were translated into goals $\mathbf{g}^*$ by optimizing

$$\mathcal{L}(\vartheta) = \mathop{\mathbb{E}}_{(r_t,\mathbf{g}_t)\sim\mathcal{D},\mathbf{g}^*\sim p_\vartheta}\Big[\underbrace{-r_t \log E(\mathbf{g}_t,\mathbf{g}^*)}_{\text{hindsight}} \quad - \quad \underbrace{(1-r_t)\log(1-E(\mathbf{g}_t,\mathbf{g}^*))}_{\text{negative feedback}}\Big],$$

we dropped the the negative feedback term when encoding observational goals in the FetchPush environment, because constrastive learning would eliminate static background information in this case and thereby destroy almost all observational information.

To practically encode rewards into goals we parametrized the goal by a simple neural network, receiving a zero-vector of shape 64 as input and outputting a goal in the corresponding goal space. For image based goals, we use three transposed convolution layers, reversing the architecture of the CNN encoder. For the LiDAR observations and our approach we use a simple three layer MLP. Discretization for our goals is achieved by a final categorical distribution.

Network sizes. We ensured that network sizes are comparable between different goal representations by increasing/decreasing the width of hidden layers accordingly to match the number of trainable parameters.

A.4 Training procedure & evaluation

We split training into a pretraining phase, where the policy and goal representation is pretrained, and a fine tuning phase, where we optimize over goals to encode reward functions. In the pretraining phase, we collect interactions in the environments for 16 steps while performing one training step as a trade-off between sample efficiency and speed. The agents are pretrained on the original, single observation goal reaching task where observations are provided by the environment. For GridWorld and FetchPush, pre-training lasts for 1000000 environment steps. For MemoryMaze, we use 500000 environment steps. During the first 50000 environment steps we use additional losses to stabilize training and improve performance. First, we use a reconstruction objective, reconstructing single observations from the encoded goals. This helps with early learning, especially in image based environment where otherwise the subset relation takes a long time to find good representations. Note that we fully transition to contrastive learning later in training, as reconstruction hinders convergence at some point. Furthermore, we regularize the discrete latent space by imposing a D_{KL} constraint on the categorical distribution effectively regularizing it towards a uniform distribution. This constraint loosens over the course of the first 50000 environment steps. This is required as during early training the encoder may collapse.

To see what the goal representation is capable of, diverse data is provided directly by the environment which guarantees good coverage of the state space. Figure 6, Figure 8 and Figure 9 are generated from goals learned with this data. Note that the policy is not required for these experiments. For each reward function, 20000 observations are sampled sequentially with training conducted every 16 steps. Finally, the agent is tasked to learn the reward function from environment interactions to evaluate how good the agent performs given the goal representation. For each reward function, the agent interacts with the environment for a total of 20000 steps. The environment steps are split up into episodes of 50 steps. The resulting behavior is analyzed using the performance criteria given Figure 4 and depicted in Figure 7. After learning the reward function, the agent is tasked to reach the optimized goal 100 times starting from different initial states and with a budget of 50 environment steps. For each reward function, three runs are performed and averaged. Then reward groups are averaged to obtain the final plot. The star plot in Figure 7 min-max normalizes the performance metrics such that 0 is the worst observed performance and 1 is the best observed performance over all rewards in an environment. Attainment dependent metrics are scaled by the attainment value such that non-attaining episodes result in worst performance (0).

A.5 USE OF LLMS

This work utilized large language models (LLMs) for two specific purposes: (1) manuscript preparation, especially grammar checking and LaTeX formatting assistance, and (2) visualization enhancement, where LLMs helped to improve plotting code aesthetics and to generate visualizations for the environments. Importantly, LLMs were used solely for presentation and formatting purposes. All underlying research data and experimental results remain unmodified to preserve scientific integrity.