Foundation Models for Semantic Novelty in Reinforcement Learning

Anonymous Author(s)
Affiliation
Address
email

Abstract

Effectively exploring the environment is a key challenge in reinforcement learning (RL). We address this challenge by defining a novel intrinsic reward based on a foundation model, such as contrastive language image pretraining (CLIP), which can encode a wealth of domain-independent semantic visual-language knowledge about the world. Specifically, our intrinsic reward is defined based on pre-trained CLIP embeddings without any fine-tuning or learning on the target RL task. We demonstrate that CLIP-based intrinsic rewards can drive exploration towards semantically meaningful states and outperform state-of-the-art methods in challenging sparse-reward procedurally-generated environments.

1 Introduction

Exploration is a key challenge in reinforcement learning (RL), especially when extrinsic feedback is sparsely provided by the environment. Intrinsic motivation provides an effective way for agents to explore the environment by rewarding the visitation of novel states [Oudeyer and Kaplan, 2009, Aubret et al., 2019]. An important question is then the definition of state novelty. Most existing methods compute novelty scores based on some form of distance function in a learned state representation, which is typically learned simultaneously with the policy [Bellemare et al., 2016a, Pathak et al., 2017]. For example, a popular recent approach RIDE [Raileanu and Rocktäschel, 2020] defines novelty as the distance of consecutive states in the embedding space of a dynamics model learned alongside the policy.

In this paper we propose a *foundation model* based intrinsic motivation scheme for RL, FoMoRL. Instead of computing novelty scores based on a domain-specific learned state representations, FoMoRL uses the pre-trained state representation of a foundation model, i.e., a generic model trained on large domain-independent visual-language data, such as contrastive language image pretraining (CLIP) [Radford et al., 2021]. In this manner FoMoRL may incorporate semantic knowledge about the state space without requiring any domain-specific pre-training, and thus help guide exploration towards semantically meaningful states such as picking up keys, opening the doors, and interacting with other useful objects relevant to the task.

The idea of FoMoRL is general to the choice of RL algorithm, motivation scheme, and the choice of foundation model. We apply FoMoRL on top of RIDE [Raileanu and Rocktäschel, 2020] and the IMPALA distributed off-policy RL algorithm [Espeholt et al., 2018]; and use CLIP as our foundation model [Radford et al., 2021]. Formally, FoMoRL can be applied to any visual RL tasks, with or without language conditioning; however, FoMoRL is expected to be effective in domains where human-interpretable semantic knowledge is useful to succeed.

We evaluate FoMoRL on the challenging sparse-reward procedurally-generated MiniGrid domain [Chevalier-Boisvert et al., 2018]. Procedurally generated environments present a significant learning

Submitted to 36th Conference on Neural Information Processing Systems (NeurIPS 2022). Do not distribute.

challenge as an agent is unlikely to visit a state more than once, and therefore the learning method needs to generalize to unseen scenarios. Somewhat surprisingly, our results suggest that CLIP representations can be very effective even in grid-world environments, despite that CLIP was trained on general human language-vision data that contains little (if any) grid-world like data points. Our analysis shows that CLIP representations encode fine-grained discrimination between different objects, colors, counts, and other visual attributes, which makes them effective for novelty based exploration even in grid-worlds. FoMoRL significantly outperforms state-of-the-art RIDE [Raileanu and Rocktäschel, 2020] method in 6 out of 9 challenging MiniGrid exploration tasks, and performs similarly in the rest of the tasks. FoMoRL learns to solve the tasks 20% faster, and achieve over 86% higher average return per episode across all tasks.

2 Related Work

Exploration in RL: There are many techniques for efficient exploration in model-free single-agent RL, including but not limited to intrinsic novelty reward [Bellemare et al., 2016b, Tang et al., 2017], predictability [Pathak et al., 2017, Oudeyer et al., 2007, Stadie et al., 2015], pure curiosity [Burda et al., 2019], Bayesian posteriors [Osband et al., 2016, Gal et al., 2017, Fortunato et al., 2018, O'Donoghue et al., 2018], information gain [Houthooft et al., 2016] or empowerment [Klyubin et al., 2005, Choshen et al., 2018].

For intrinsic motivation-based exploration, a key problem is defining the novelty score. Raileanu and Rocktäschel [2020] proposed RIDE which computes novelty scores based on differences between state representations of consecutive states. The state representations are learned from learning a transition function via forward and inverse dynamics model. Another work on random network distillation (RND) [Burda et al., 2018] uses a fixed randomly initialized neural network to represent the states and reward the exploration based on the error of a neural network predicting the fixed features. Never Give Up (NGU; [Badia et al., 2020]) combines an episodic novelty module based on inverse dynamics features with RND as a lifelong novelty module. DIAYN [Eysenbach et al., 2018] performs unsupervised exploration independent of the task and models the policies conditioned on a latent skill variable. In contrast to these works we propose to use pre-trained foundation models to define intrinsic rewards.

Concurrent to our efforts, Tam et al. [2022] proposed the idea of using pretrained representations from foundation models for exploration in RL. The high-level idea in the concurrent work is the same as ours; however, there are important differences in its application. The concurrent work builds on top of RND [Burda et al., 2018] and NGU [Badia et al., 2020] whereas we use RIDE [Raileanu and Rocktäschel, 2020]; and the concurrent work mostly focuses on Unity based 3D environments with language oracles, whereas we experiment with procedurally-generated long-horizon grid-world tasks that have no language oracle, and whose visual appearance differs significantly from the visual data used to pretrain the CLIP model.

Foundation Models in RL: Khandelwal et al. [2022] investigate the effectiveness of CLIP visual representations directly for control on Embodied AI tasks [Batra et al., 2020] by bypassing the learning of policy representations with CLIP embeddings. Their results demonstrated the effectiveness of CLIP representations for control on navigation-heavy Embodied AI tasks. Shridhar et al. [2022] proposed CLIPort which uses CLIP representations with imitation learning for robotic manipulation tasks, adding spatial understanding of the scenes as well. SayCan [Ahn et al., 2022] uses a Large Language Model (LLM) to supply high-level instructions to solve the tasks and pair it with an affordance based value function to see if the robot has required skills to perform those instructions. Similarly another paper [Patel and Pavlick, 2021] shows that LLMs such as GPT-2 [Radford et al., 2019] or GPT-3 [Brown et al., 2020] can learn to ground the concepts such as direction or colour that it is explicitly taught and also generalise to several instances of unseen concepts as well. In contrast to the above works we explore using intrinsic rewards based on foundational models for efficient semantic exploration in visual RL tasks.

3 Problem Formulation

A partially-observable Markov decision process (POMDP) is defined as a tuple $G = \langle \mathcal{S}, \mathcal{A}, P, R, \Omega, O, \gamma \rangle$. The true state of the environment is denoted by $s \in \mathcal{S}$. At each time step,

an agent chooses an action $a \in \mathcal{A}$, which causes a transition in the environment according to the state transition kernel $P(s'|s,a) : \mathcal{S} \times \mathcal{A} \times \mathcal{S} \to [0,1]$, and the agent receives an extrinsic reward $R_{\text{ext}} : \mathcal{S} \times \mathcal{A} \to \mathbb{R}$. $\gamma \in [0,1)$ is a discount factor.

Due to *partial observability*, the agent cannot observe the true state s, but receives a partially observable visual image $o \in \Omega$ drawn from observation kernel $o \sim O(s,a)$. At time t, the agent has access to its action-observation history $\tau_t \in \mathcal{T}_t \equiv (\Omega \times \mathcal{U})^t \times \Omega$, on which it conditions a stochastic policy $\pi(a_t|\tau_t)$. The stochastic policy induces an action-value function : $Q^\pi(s_t, \tau_t, a_t) = \mathbb{E}\left[G_t | s_t, \tau_t, a_t\right]$, where $G_t = \sum_{i=0}^{\infty} \gamma^i r_{t+i}$ is the *discounted return*.

4 FoMoRL

FoMoRL assigns an intrinsic reward based on CLIP's [Radford et al., 2021] visual encoder embeddings. CLIP embeddings enable semantic understanding of the images as it has been contrastively trained with image captioned large scale human data, thereby incorporating language abstractions. These language abstractions can summarise the images with important fine grained details and yet being very concise, much like our natural languages [Borghi and Binkofski, 2014].

4.1 Semantic Intrinsic Motivation

We define the representation of the current observation $\phi(o_t)$ as the pretrained representation from the CLIP encoder i.e. $\phi(o_t) = \text{clip}(o_t)$. The intrinsic reward R_{int} at time step t is computed as the L2-norm $||\phi(o_{t+1}) - \phi(o_t)||_2$ of the difference in the clip representation between consecutive states discounted by episodic state visitation counts similar to Raileanu and Rocktäschel [2020]:

$$r_{\text{int}}(s_t, a_t, o_t) = \frac{||\text{clip}(o_{t+1}) - \text{clip}(o_t)||_2}{\sqrt{N_{ep}(s_{t+1})}},$$

where $N_{ep}(s_{t+1})$ is the visitation count of state s_{t+1} during the current episode, which is initialized to 1 in the beginning of the episode. The state visitation count can be directly computed for MDPs with small number of discrete states, but can also be approximated for MDPs with large discrete or continuous state spaces [Martin et al., 2017, Machado et al., 2020]. The above intrinsic reward encourages the visitation of states which have significantly different CLIP representations from the current state. CLIP representations can perform fine-grained classification of abstract concepts in the visual observations and can therefore guide exploration towards semantically meaningful states which are useful for the downstream task. The division by state counts ensures that the agent does not exploit the intrinsic reward function by swinging between two states with large difference in CLIP embeddings. For this paper, we use the officially released ResNet50 $\times$ 16 model for the CLIP encoder which outputs state representations of dimension 1024.

4.2 Training Setup

We train our policy using IMPALA [Espeholt et al., 2018] in a distributed off-policy setup with multiple actors. The policy is trained to maximize the combined reward i.e. $r(s_t, a_t) = \alpha r_i(s_t, a_t) + r_e(s_t, a_t)$, where α is a hyperparameter. The policy learns its own state representations from scratch, while the pretrained representations are just used for guiding exploration through intrinsic rewards. The total loss L_{tot} function is composed of four components: L_{pg}: policy gradient loss, L_{base}: value function loss, and L_{ent}: entropy loss, with respective hyper parameter coefficients i.e.

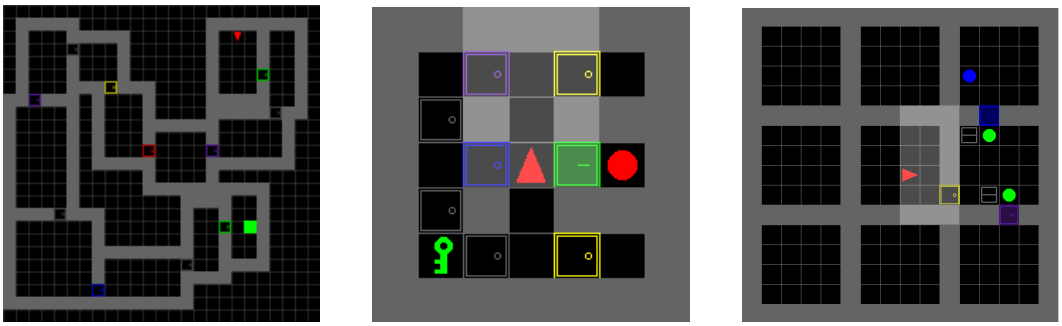
$$L_{\text{tot}} = L_{\text{pg}} + c_{\text{base}} L_{\text{base}} + c_{\text{ent}} L_{\text{ent}}$$

Since our policy is conditioned on partial observations, we use an LSTM [Hochreiter and Schmidhuber, 1997] to summarize the action-observation history τ_t.

5 Experiments

5.1 MiniGrid Domain

We evaluate our proposed method on three types of hard exploration tasks from the MiniGrid domain [Chevalier-Boisvert et al., 2018]: MultiRoomNXSY, KeyCorridorS3R3, and ObstructedMaze2Dlh.



(a) MultiRoomN12S10 (b) KeyCorridorS3R3 (c) ObstructedMaze2Dlh

Figure 1: Representative examples of different MiniGrid environments. The highlighted area showcases the partial observations the agent receives.

The environment consists of an $N \times M$ grid world with each cell is either empty or consist one of the objects such as wall, floor, lava, door, key, ball, box and goal. The agent can pick up and carry exactly one object, and each object has a discrete color. In order to open up a locked door, the agent has to pick up and carry the key matching the color of the door. The action space is discrete and allows agent to move forward, turn left/right, pick up an object, drop the object being carried, toggle (open doors or interact with objects), and a done action to mark the task complete. At each time step, the agent receives a partial and egocentric observation and the agent cannot see through walls or closed doors.

The MultiRoomNXSY task (Figure 1a) consists of X rooms, with size at most Y, connected in random orientations in every episode. This task has a series of connected rooms with doors that must be opened in order to get to the next room with the green goal square as the final destination the agent must get to. In the KeyCorridorS3R3 task (Figure 1b), the agent has to explore the environment to find a key hidden in another room, and then use the key to open the corresponding door and pickup the ball. In the ObstructedMaze2Dlh task (Figure 1c), the agent has to pick up a box which is placed in a corner of a maze. The doors are locked, the keys are hidden in boxes and doors are obstructed by balls. The NoisyTV [Burda et al., 2018] version of MultiRoomNXSY adds stochasticity to the environment by adding a random ball which changes color everytime the agent takes a particular action. The agents need to learn to avoid this distraction as it is irrelevant to the task.

The environment provides different input representations of the state: (1) compact grid encoding using 3 integer values: describing the type and color of the object in the cell, and a flag indicating whether doors are open or closed; (2) raw RGB visual observations from the procedurally generated environment.

5.2 Baseline

For all our experiments, we use IMPALA [Espeholt et al., 2018] as the base RL algorithm, and compare against the popular RIDE method by Raileanu and Rocktäschel [2020]. RIDE also uses IMPALA to train the policy with the intrinsic reward based on learning state representations which minimizes the forward and inverse dynamics loss. In the original paper, the authors use the compact grid encoded representation (see Section 5.1 for definition), which already contains all key required information about the visual observation, thereby making it simpler to learn a transition function and the state representations. In this paper, we instead use the raw visual RGB observations, which makes the task significantly harder as agents need to extract the features from raw images. Both the policy and intrinsic reward modules for RIDE and FoMoRL method receives a partial egocentric observation as input.

5.3 Results

Table 1 summarizes the results for various MiniGrid tasks. As shown in the table, FoMoRL consistently outperforms the original baseline method RIDE both with respect to better final average return and time to reach convergence. RIDE fails to learn efficient state representations in the partially

Table 1: Performance of RIDE and FoMoRL on a variety of hard exploration problems in MiniGrid. Both methods receive partially observable RGB images as input to the policy and intrinsic reward module. The values indicate the final mean episodic return averaged over multiple seeds after training on [xM] frames. The bracketed values (xM) indicate the average number of training frames (in millions) required to achieve the convergence value. The (-) indicates no convergence in [xM] frames.

	RIDE	FoMoRL
DoorKey-5x5 [15M]	**0.96 (5M)**	**0.96 (2.5M)**
MultiRoom-N7-S4 [30M]	0.71 (15M)	**0.76 (7M)**
MultiRoomNoisyTV-N7-S4 [30M]	0.7 (20M)	**0.74 (19M)**
MultiRoom-N7-S8 [30M]	0.0 (-)	**0.59 (29M)**
MultiRoom-N10-S4 [30M]	0.66 (10M)	**0.75 (6M)**
MultiRoom-N10-S10 [80M]	0.0 (-)	0.0 (-)
MultiRoom-N12-S10 [80M]	0.0 (-)	0.0 (-)
ObstructedMaze-2Dlh [80M]	0.0 (-)	**0.96 (45M)**
KeyCorridorS3R3 [30M]	0.0 (-)	**0.9 (18M)**

Table 2: Performance of FoMoRL on very-hard exploration tasks N10-S10 and N12-S10 in MiniGrid with partially observable RGB images as input to the policy and fully observable RGB images as input to the intrinsic reward module. The values indicate the final mean episodic return averaged over multiple seeds after training on [xM] frames. The bracketed values (xM) indicate the average number of training frames (in millions) required to achieve the convergence value.

	FoMoRL
MultiRoom-N10-S10 [80M]	**0.7 (22M)**
MultiRoom-N12-S10 [80M]	**0.72 (24M)**

observable RGB space and therefore completely fails to learn in a number of scenarios. On the other hand, FoMoRL directly uses CLIP representations as state embeddings for intrinsic motivation and thereby completely bypass the representation learning altogether.

As shown in the table, both RIDE and FoMoRL failed to learn anything on larger versions of MultiRoom environment i.e. MultiRoom-N10-S10 and MultiRoom-N12-S10 with 10 and 12 rooms respectively having a maximum room size of 10. We analyzed the training logs of these failed tasks and found that the failure is attributed to the partial observability of the RGB input image. Since intrinsic reward module only sees a partial observable egocentric observation, it tries to generate diversity within that restricted view and gather intrinsic reward. Therefore, the agent learns to move around from one cell to another and open doors only when it is within the observable area. The moving around from cell to cell ends up taking most of the horizon time before the agent can reach the final destination. We therefore tried to resolve this by allowing only the intrinsic module to access the full RGB image of the environment, and keeping the policy input partially observable. Table 2 clearly shows that providing the intrinsic reward module access to fully observable RGB images resolves the issue for FoMoRL.

For some tasks, we noticed that higher value of intrinsic reward coefficient led to the agent getting distracted from solving the original task and rather focus on generating more and more diversity, thereby making both RIDE and FoMoRL fail to learn. However, most environments were solvable by α={0.1, 0.5} for RIDE and α={0.02, 0.005} for FoMoRL.

5.4 CLIP Analysis on MiniGrid

In the previous section, we showed that CLIP representations are remarkably effective across multiple MiniGrid exploration tasks. We perform a small experiment to check what exactly CLIP understands from these RGB grid images. We input the RGB images to the CLIP's visual encoder and then input proposed language descriptions through the CLIP's language encoder and compute a similarity score. The higher score of similarity implies that CLIP representations prefer that description over others.

Figure 2 shows the test image which is inputted to the CLIP's visual encoder. The proposed language descriptions and the similarity scores are shown in 3.

Figure 3 shows the results for testing the CLIP representations to encode the color and count of objects, their spatial information, and more fine-grained information. We test the color understanding of the CLIP representations by comparing a correct description of the color of squares with an incorrect one, as there are no green squares present in the test image (Figure 2). We then inspect if CLIP representations prefers a description with spatial information of the key shaped object or not. To check the understanding of CLIP representations to represent the count of objects, we compare the number of blue squares with an incorrect description of three blue squares. Lastly, we test for more fine grained information within the yellow squares. One of the yellow squares has a circle within it and other one has a dash within (semantically meaning that the door is unlocked/locked).

Figure 3 indicates that CLIP representations can clearly discriminate between different colors and counts of objects, their spatial understanding, and even more fine grained information like whether a square has a mini circle or minus within. CLIP representations does not directly understand the significance of the objects within the RGB image, for example, it does not directly understand that a square indicates a door, or a square with a mini circle indicates an unlocked door. However, the representations can understand and distinguish between the geometrical shapes very well which is enough for diversity based semantic exploration. On the contrary, CLIP representations have already shown to be very effective for large range of tasks from text based video retrieval [Fang et al., 2021, Luo et al., 2021], text driven image manipulation [Patashnik et al., 2021] to embodied AI tasks [Khandelwal et al., 2022] based on realistic visual observations, for example, a kitchen scene containing a microwave.

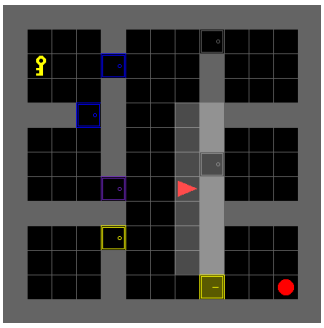


Figure 2: RGB Input Image for CLIP Embedding Analysis

6 Conclusion

We showcase that language abstractions incorporated within foundation models such as CLIP can be very useful for semantic exploration in RL, even on grid based environments. Our proposed method FoMoRL can be applied independent of the choice of RL algorithm, intrinsic motivation scheme, or the type of foundation model. FoMoRL achieves state-of-the-art performance with significantly higher average return and solving the task much faster than the baseline RIDE method. This work shows a proof of concept for leveraging foundation models for semantic novelty in RL. Future work may investigate realistic 3D environments like Habitat ObjectNav Challenge [Batra et al., 2020] and language conditioned tasks such as CALVIN [Mees et al., 2022]. It would be also interesting to leverage the language encoder from CLIP alongside the visual encoder to enable semantic exploration in language conditioned tasks.

References

Michael Ahn, Anthony Brohan, Noah Brown, Yevgen Chebotar, Omar Cortes, Byron David, Chelsea Finn, Keerthana Gopalakrishnan, Karol Hausman, Alex Herzog, et al. Do as i can, not as i say: Grounding language in robotic affordances. *arXiv preprint arXiv:2204.01691*, 2022.

Arthur Aubret, Laetitia Matignon, and Salima Hassas. A survey on intrinsic motivation in reinforcement learning. *arXiv preprint arXiv:1908.06976*, 2019.

Adrià Puigdomènech Badia, Pablo Sprechmann, Alex Vitvitskyi, Daniel Guo, Bilal Piot, Steven Kapturowski, Olivier Tieleman, Martín Arjovsky, Alexander Pritzel, Andew Bolt, et al. Never give up: Learning directed exploration strategies. *arXiv preprint arXiv:2002.06038*, 2020.

Dhruv Batra, Aaron Gokaslan, Aniruddha Kembhavi, Oleksandr Maksymets, Roozbeh Mottaghi, Manolis Savva, Alexander Toshev, and Erik Wijmans. ObjectNav Revisited: On Evaluation of Embodied Agents Navigating to Objects. In *arXiv:2006.13171*, 2020.

Top Predictions for testing colour:

Red triangle, yellow key, grey squares, **blue** squares, yellow squares, red circle, purple square: **79.25%**
Red triangle, yellow key, grey squares, **green** squares, yellow squares, red circle, purple square: 19.73%

Top Predictions for spatial understanding of key shaped object:

Red triangle, **top left yellow key**, grey squares, blue squares, yellow squares, red circle, purple square: **56.74%**
Red triangle, **yellow key**, grey squares, blue squares, yellow squares, red circle, purple square: 42.82%

Top Predictions for testing count of blue squares:

Red triangle, top left yellow key, grey squares, **two** blue squares, yellow squares, red circle, purple square: **62.11%**
Red triangle, top left yellow key, grey squares, **three** blue squares, yellow squares, red circle, purple square: 37.67%

Top Predictions for testing fine-grained information within yellow squares:

Red triangle, top left yellow key, grey squares, two blue squares, **two yellow squares with a circle and a dash**, red circle, purple square: **87.21%**
Red triangle, top left yellow key, grey squares, two blue squares, **two yellow squares both with a circle**, red circle, purple square: 12.76%

Figure 3: Similarity scores for proposed language descriptions of the input test image. (a) Test to check whether CLIP representations prefer a description with the presence of a green square (incorrect) or a grey square (correct) in the test image. (b) Test to check the count of blue squares in the test image. (c) Test to check if CLIP representations prefer the description with spatial information of the key in the the test image or not. (d) Test to check more fine-grained understanding of the test image by comparing against the shapes present within the yellow square in the test image.

Marc Bellemare, Sriram Srinivasan, Georg Ostrovski, Tom Schaul, David Saxton, and Remi Munos. Unifying count-based exploration and intrinsic motivation. *Advances in neural information processing systems*, 29, 2016a.

Marc G. Bellemare, Sriram Srinivasan, Georg Ostrovski, Tom Schaul, David Saxton, and Rémi Munos. Unifying count-based exploration and intrinsic motivation. In *Advances in Neural Information Processing Systems (NIPS) 29*, pages 1471–1479, 2016b.

Anna M Borghi and Ferdinand Binkofski. *Words as social tools: An embodied view on abstract concepts*, volume 2. Springer, 2014.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901, 2020.

Yuri Burda, Harrison Edwards, Amos Storkey, and Oleg Klimov. Exploration by random network distillation. *arXiv preprint arXiv:1810.12894*, 2018.

Yuri Burda, Harri Edwards, Deepak Pathak, Amos Storkey, Trevor Darrell, and Alexei A. Efros. Large-scale study of curiosity-driven learning. In *International Conference on Learning Representations (ICLR)*, 2019.

Maxime Chevalier-Boisvert, Lucas Willems, and Suman Pal. Minimalistic gridworld environment for gymnasium. `https://github.com/Farama-Foundation/MiniGrid`, 2018.

Leshem Choshen, Lior Fox, and Yonatan Loewenstein. Dora the explorer: Directed outreaching reinforcement action-selection. *arXiv preprint arXiv:1804.04012*, 2018.

Lasse Espeholt, Hubert Soyer, Remi Munos, Karen Simonyan, Vlad Mnih, Tom Ward, Yotam Doron, Vlad Firoiu, Tim Harley, Iain Dunning, et al. Impala: Scalable distributed deep-rl with importance weighted actor-learner architectures. In *International conference on machine learning*, pages 1407–1416. PMLR, 2018.

Benjamin Eysenbach, Abhishek Gupta, Julian Ibarz, and Sergey Levine. Diversity is all you need: Learning skills without a reward function. *arXiv preprint arXiv:1802.06070*, 2018.

Han Fang, Pengfei Xiong, Luhui Xu, and Yu Chen. Clip2video: Mastering video-text retrieval via image clip. *arXiv preprint arXiv:2106.11097*, 2021.

Meire Fortunato, Mohammad Gheshlaghi Azar, Bilal Piot, Jacob Menick, Matteo Hessel, Ian Osband, Alex Graves, Volodymyr Mnih, Remi Munos, Demis Hassabis, Olivier Pietquin, Charles Blundell, and Shane Legg. Noisy networks for exploration. In *International Conference on Learning Representations (ICLR)*, 2018. URL `https://openreview.net/forum?id=rywHCPkAW`.

Yarin Gal, Jiri Hron, and Alex Kendall. Concrete dropout. In *Advances in Neural Information Processing Systems (NIPS)*, pages 3584–3593, 2017.

Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural computation*, 9(8): 1735–1780, 1997.

Rein Houthooft, Xi Chen, Yan Duan, John Schulman, Filip De Turck, and Pieter Abbeel. Vime: Variational information maximizing exploration. *Advances in neural information processing systems*, 29, 2016.

Apoorv Khandelwal, Luca Weihs, Roozbeh Mottaghi, and Aniruddha Kembhavi. Simple but effective: Clip embeddings for embodied ai. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14829–14838, 2022.

Alexander S Klyubin, Daniel Polani, and Chrystopher L Nehaniv. Empowerment: A universal agent-centric measure of control. In *2005 ieee congress on evolutionary computation*, volume 1, pages 128–135. IEEE, 2005.

Huaishao Luo, Lei Ji, Ming Zhong, Yang Chen, Wen Lei, Nan Duan, and Tianrui Li. Clip4clip: An empirical study of clip for end to end video clip retrieval. *arXiv preprint arXiv:2104.08860*, 2021.

Marlos C Machado, Marc G Bellemare, and Michael Bowling. Count-based exploration with the successor representation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 5125–5133, 2020.

Jarryd Martin, Suraj Narayanan Sasikumar, Tom Everitt, and Marcus Hutter. Count-based exploration in feature space for reinforcement learning. *arXiv preprint arXiv:1706.08090*, 2017.

Oier Mees, Lukas Hermann, Erick Rosete-Beas, and Wolfram Burgard. Calvin: A benchmark for language-conditioned policy learning for long-horizon robot manipulation tasks. *IEEE Robotics and Automation Letters (RA-L)*, 7(3):7327–7334, 2022.

Brendan O'Donoghue, Ian Osband, Rémi Munos, and Volodymyr Mnih. The uncertainty Bellman equation and exploration. In *Proceedings of the 35th International Conference on Machine Learning (ICML)*, pages 3836–3845, 2018. URL `http://proceedings.mlr.press/v80/o-donoghue18a.html`.

Ian Osband, Benjamin Van Roy, and Zheng Wen. Generalization and exploration via randomized value functions. In *Proceedings of the 33rd International Conference on International Conference on Machine Learning (ICML)*, pages 2377–2386, 2016.

Pierre-Yves Oudeyer and Frederic Kaplan. What is intrinsic motivation? a typology of computational approaches. *Frontiers in neurorobotics*, page 6, 2009.

Pierre-Yves Oudeyer, Frdric Kaplan, and Verena V Hafner. Intrinsic motivation systems for autonomous mental development. *IEEE transactions on evolutionary computation*, 11(2):265–286, 2007.

Or Patashnik, Zongze Wu, Eli Shechtman, Daniel Cohen-Or, and Dani Lischinski. Styleclip: Text-driven manipulation of stylegan imagery. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 2085–2094, 2021.

Roma Patel and Ellie Pavlick. Mapping language models to grounded conceptual spaces. In *International Conference on Learning Representations*, 2021.

Deepak Pathak, Pulkit Agrawal, Alexei A Efros, and Trevor Darrell. Curiosity-driven exploration by self-supervised prediction. In *International conference on machine learning*, pages 2778–2787. PMLR, 2017.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9, 2019.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International Conference on Machine Learning*, pages 8748–8763. PMLR, 2021.

Roberta Raileanu and Tim Rocktäschel. Ride: Rewarding impact-driven exploration for procedurally-generated environments. *arXiv preprint arXiv:2002.12292*, 2020.

Mohit Shridhar, Lucas Manuelli, and Dieter Fox. Cliport: What and where pathways for robotic manipulation. In *Conference on Robot Learning*, pages 894–906. PMLR, 2022.

Bradly C Stadie, Sergey Levine, and Pieter Abbeel. Incentivizing exploration in reinforcement learning with deep predictive models. *arXiv preprint arXiv:1507.00814*, 2015.

Allison C Tam, Neil C Rabinowitz, Andrew K Lampinen, Nicholas A Roy, Stephanie CY Chan, DJ Strouse, Jane X Wang, Andrea Banino, and Felix Hill. Semantic exploration from language abstractions and pretrained representations. *arXiv preprint arXiv:2204.05080*, 2022.

Haoran Tang, Rein Houthooft, Davis Foote, Adam Stooke, OpenAI Xi Chen, Yan Duan, John Schulman, Filip DeTurck, and Pieter Abbeel. #Exploration: A study of count-based exploration for deep reinforcement learning. In *Advances in Neural Information Processing Systems (NIPS) 30*, pages 2753–2762. 2017.