Language as an Abstraction for Hierarchical Deep Reinforcement Learning

Paper Authors: Yiding Jiang, Shixiang Gu, Kevin Murphy, Chelsea Finn
Problem Overview

- Learning a variety of **compositional, long horizon** skills while being able to **generalize** to novel concepts remains an open challenge.

- Can we leverage the compositional and generalizable structure of **language** as an abstraction for goals to help decompose problems?
Learning Sub-Goals

Hierarchical Reinforcement Learning:
- High-level policy: $\pi_h(g \mid s)$
- Low-level policy: $\pi_l(a \mid s, g)$
Language as an abstraction for goals

Hierarchical Reinforcement Learning:
- High-level policy: \( \pi_h(g \mid s) \)
- Low-level policy: \( \pi_l(a \mid s, g) \)

What if \( g \) is an sentence in human language? Some motivations in paper:
1) High-level policies would generate interpretable goals
2) An instruction can represent a region of states that satisfy some abstract criteria
3) Sentences have a compositional and generalizable structure
4) Humans use language as an abstraction for reasoning, planning, and knowledge acquisition
Concrete Examples Studied

High Level:
(a) Object arrangement
(b) Object ordering
(c) Object sorting
(d) Color ordering
(e) Shape ordering
(f) Color & shape ordering

Low level:
(a) Goal is $g_0$: “There is a red ball; are there any matte cyan sphere right of it?”
Environment

- New environment using MuJoCo physics engine and CLEVR language engine.
- Binary reward function, only if all the constraints are met
- State-based observation:
  \[ s \in \mathbb{R}^{10} \quad |A| = 40 \]
- Image-based observation:
  \[ s \in \mathbb{R}^{64 \times 64 \times 3} \quad |A| = 800 \]
Methods
Low-Level Policy

Language to state mapping
\[ \omega(g|s) \rightarrow \Omega(s) = \{g \mid \omega(g|s) > 0\} \]

Checking if a state satisfies an instruction
\[ \Psi : S \times G \rightarrow \{0, 1\} \]

Trained on sampled language instructions
\[ g \sim \mathcal{U}(\{g \in \Omega(s_{t+1}) \mid \Psi(s_{t+1}, g) = 0\}) \]
Low-Level Policy

\[ \pi_l(a \mid g, s_t) \]

Reward Function

\[
R(s_t, a_t, s_{t+1}, g) = \begin{cases} 
0 & \text{if } \Psi(s_{t+1}, g) = 0 \\
\Psi(s_{t+1}, g) \oplus \Psi(s_t, g) & \text{if } \Psi(s_{t+1}, g) = 1
\end{cases}
\]
Low-Level Policy

\[ \pi_l(a | g, s_t) \]

Reward Function

\[
R(s_t, a_t, s_{t+1}, g) = \begin{cases} 
0 & \text{if } \Psi(s_{t+1}, g) = 0 \\
\Psi(s_{t+1}, g) \oplus \Psi(s_t, g) & \text{if } \Psi(s_{t+1}, g) = 1 
\end{cases}
\]

Can be very sparse

Hindsight Instruction Relabeling (HIR)

- Similar to Hindsight Experience Replay (HER)
- HIR is used to relabel the goal with an instruction that was satisfied.
- Enable the agent to learn from many different language goals at once
High-Level Policy

- Double Q-Learning Network [1]
- Reward given only if all constraints were satisfied from the environment
- Instructions (goals) are pick, not generated.
- Uses extracted visual features from the low-level policy and then extract salient spatial points with spatial softmax. [2]


Experiments
Experimentation Goals

- **Compositionality**: How does language compare with alternative representations?
- **Scalability**: How well does this framework scale?
  - With **instruction diversity**
  - With **state dimensionality**
- **Policy Generalization**: Can the policy systematically generalize by leveraging the structure of language?
- **Overall**, how does this approach compare to state-of-the-art hierarchical RL approaches?
Experimentation Goals

- **Compositionality**: How does language compare to alternative representations?
- **Scalability**: How well does this framework scale?
  - With instruction diversity
  - With state dimensionality
- **Policy Generalization**: Can the policy systematically generalize by leveraging the structure of language?
- **Overall**, how does this approach compare to state-of-the-art hierarchical RL approaches?
Compositionality: How does language compare to alternative representations?

- One-hot instruction encoding
- Non-compositional Representation: loss-less autoencoder for instructions.
Experimentation Goals

● **Compositionality**: How does language compare with alternative representations?

● **Scalability**: How well does this framework scale?
  ○ With *instruction diversity*
  ○ With *state dimensionality*

● **Policy Generalization**: Can the policy systematically generalize by leveraging the structure of language?

● **Overall**, how does this approach compare to state-of-the-art hierarchical RL approaches?
Scalability: How well does this framework scale?

- With **instruction diversity**
- With **state dimensionality**
Experimentation Goals

- **Compositionality:** How does language compare with alternative representations?
- **Scalability:** How well does this framework scale?
  - With instruction diversity
  - With state dimensionality
- **Policy Generalization:** Can the policy systematically generalize by leveraging the structure of language?
- **Overall,** how does this approach compare to state-of-the-art hierarchical RL approaches?
Policy Generalization: Can the policy systematically generalize by leveraging the structure of language?

<table>
<thead>
<tr>
<th></th>
<th>Standard train</th>
<th>Standard test</th>
<th>Standard gap</th>
<th>Systematic train</th>
<th>Systematic test</th>
<th>Systematic gap</th>
</tr>
</thead>
<tbody>
<tr>
<td>Language</td>
<td>21.50 ± 2.28</td>
<td>21.49 ± 2.53</td>
<td>0.001</td>
<td>20.09 ± 2.46</td>
<td>8.13 ± 2.34</td>
<td>0.596</td>
</tr>
<tr>
<td>Non-Compos</td>
<td>6.26 ± 1.18</td>
<td>5.78 ± 1.44</td>
<td>0.077</td>
<td>7.54 ± 1.14</td>
<td>0.76 ± 0.69</td>
<td>0.899</td>
</tr>
<tr>
<td>Random</td>
<td>0.17 ± 0.20</td>
<td>0.21 ± 0.17</td>
<td>-</td>
<td>0.11 ± 0.19</td>
<td>0.18 ± 0.22</td>
<td>-</td>
</tr>
</tbody>
</table>

Random: 70/30 random split of the instruction set.

Systematic: Training set doesn’t include “red” in the first half of instructions, and Test set is the complement. => Zero-shot Adaptation
Experimentation Goals

- **Compositionality**: How does language compare with alternative representations?

- **Scalability**: How well does this framework scale?
  - With *instruction diversity*
  - With *state dimensionality*

- **Policy Generalization**: Can the policy systematically generalize by leveraging the structure of language?

- **Overall**, how does this approach compare to state-of-the-art hierarchical RL approaches?
High-Level Policy Experiments

DDQN: non-hierarchical

HIRO and OC: hierarchical, non-language based
High Level Policy Experiments (Visual)
Takeaways

● **Strengths:**
  - High-level policies are human-interpretable
  - Low-level policy can be re-used for different high-level objectives
  - Language abstractions generalized over a region of goal states, instead just an individual goal state
  - Generalization to high dimensional instruction sets and action spaces

● **Weakness:**
  - Low-level policy depends on the performance of another system for its reward
  - HIR is dependent on the performance of another system for its new goal label
  - The instruction set is domain-specific
  - The number of subtasks are fixed
Future Work

- Instead of picking instructions, generate them
- Dynamic or/and learned number of substeps
  - Curriculum learning by decreasing the number of substeps as the policies are training
  - Study how does the parameter effects the overall performance of the model
- Finetune policies to each other, instead just training them separately
- Concern about practicality: for any problem need both a set of sub-level instructions and a language oracle which can validate their fulfilment
- Other ways to validate low-level reward
Potential Discussion Questions

● Is it prideful to try to use language to try to impose language structure on these subgoals instead of looking for less human-motivated solutions?

● In two equally performing models, one with language interpretability seems inherently better due to interpretability. Does this make these types of abstractions likely for the future?

● Can you think of any other situations in which this hierarchical model could be implemented? Would language always be appropriate?
Appendix
Overall Approach: Object Ordering

Arrange the objects so that their colors range from red to blue in the horizontal direction, and keep the object to close vertically.

Answer: Cyan, purple, green, blue, red (from left to right)
Overall Approach: Object Ordering

There is a cyan rubber ball, are there purple matte spheres right of it?
Overall Approach: Object Ordering

There is a green rubber sphere, are purple matte sphere behind it?
Overall Approach: Object Ordering

There is a blue rubber sphere; are the cyan matte spheres on the left side?
Overall Approach: Object Ordering

Answer: Cyan, purple, green, blue, red (from left to right)

There is a green rubber sphere; are the purple matte balls left of it?
State-based Low-Level Policy
Vision-based Low-Level Policy