Evaluating the Robustness of Natural Language Reward Shaping Models to Spatial Relations

Antony Yun

The University of Texas at Austin
Computer Science
Successes of Reinforcement Learning

https://bair.berkeley.edu/blog/2020/05/05/fabrics/
https://deepmind.com/blog/article/AlphaFold-Using-AI-for-scientific-discovery
My Work

- Construct a challenge dataset in the Meta-World reward shaping domain that contains spatially relational language
- Evaluate robustness of existing natural language reward shaping models
Outline

- Background on Deep Learning, Reinforcement Learning
- Natural language reward shaping
- Our Dataset
- Results
Background: Neural Networks

- Function approximators
- Trained with gradient descent

\[ f(\quad ) = [0.12, 0.05, ...] \]

https://github.com/caoscott/SReC
Background: Neural Networks

https://www.researchgate.net/figure/Illustration-of-LSTM-block-s-is-the-sigmoid-function-which-play-the-role-of-gates-during-fig2_322477802
Background: Reinforcement Learning

- Learn a policy by interacting with the environment
- Optimize cumulative discounted reward

https://deepmind.com/blog/article/producing-flexible-behaviours-simulated-environments
http://web.stanford.edu/class/cs234/index.html
Background: Markov Decision Process (MDP)

\[ M = \langle S, A, T, R, \gamma \rangle \]

- \( S \) = states
- \( A \) = actions
- \( T \) = transition function
- \( R \) = reward
- \( \gamma \) = discount factor
Background: Policy Based RL

- Parameterized policy
  \[ \pi_\theta(s, a) = P[a \mid s, \theta] \]
- Want optimal policy that maximizes expected reward
- Learned by gradient descent on final reward
- We use Proximal Policy Optimization (PPO)

[Schulman et al., 2017]
Challenges with RL

- Sample inefficient

https://www.alexirpan.com/2018/02/14/rl-hard.html
Challenges with RL

- Sample inefficient
- Good reward functions are hard to find
  - Sparse: easy to design

https://www.alexirpan.com/2018/02/14/rl-hard.html
Challenges with RL

- Sample inefficient
- Good reward functions are hard to find
  - Sparse: easy to design
  - Dense: easy to learn

https://www.alexirpan.com/2018/02/14/rl-hard.html
Background: Reward Shaping

- Provide additional *potential* reward
- Does not change the optimal policy

\[
R' = R + F
\]
\[
F(s, a, s') = \gamma \phi(s') - \phi(s)
\]

[Ng et al, 1999]
Prior Work: LEARN

- Language-based shaping rewards for Montezuma's Revenge
- Non-experts can express intent
- 60% improvement over baseline

"Jump over the skull while going to the left"

[Goyal et al, 2019]
Prior Work: LEARN

[Image of diagram showing the LEARN model with language and action components]

Sequence of past actions ($a_1, \ldots, a_{t-1}$)

Observation + Reward

Agent

Environment

Action-frequency vector

Jump over the skull while going to the left.

Encoded action-frequency vector ($D_1$-dimensional)

Linear Linear

Language encoder

($D_3$-dimensional)

Concat

Encoded command ($D_2$-dimensional)

Jump over the skull while going to the left

Linear + Softmax

Probabilities (RELATED / UNRELATED)

Probabilities (RELATED / UNRELATED)

[Goyal et al, 2019]
Meta-World

- Object manipulation domain involving grasping, placing, and pushing
- Continuous action space, multimodal data, complex goal states

Figure 1. A simulated robot completing a task in the Meta-World domain

[Yu et al, 2019]
Dense Rewards in Meta-World

\[ R = R_{\text{reach}} + R_{\text{grasp}} + R_{\text{place}} \]
\[ = -\|h - o\|_2 + \mathbb{I}_{\|h - o\|_2 < \epsilon} \cdot c_1 \cdot \min\{o_z, z_{\text{target}}\} + \mathbb{I}_{|o_z - z_{\text{target}}| < \epsilon} \cdot c_2 \cdot \exp\left\{\frac{\|o - g\|_2^2}{c_3}\right\} \]

\[ R = R_{\text{reach}} + R_{\text{push}} \]
\[ = -\|h - o\|_2 + \mathbb{I}_{\|h - o\|_2 < \epsilon} \cdot c_2 \cdot \exp\left\{\frac{\|o - g\|_2^2}{c_3}\right\} \]

[Yu et al, 2019]
Dense Rewards in Meta-World

Table 3: A list of reward functions used for each of the Meta-World tasks.
Pix2R Dataset

- 13 Meta-World tasks, 9 objects
- 100 scenarios per task
- Videos generated using PPO on dense rewards
- 520 human-annotated descriptions from Amazon Mechanical Turk
- Use video trajectories + descriptions to approximate dense reward

[Goyal et al, 2020]
Pix2R Architecture

[Goyal et al, 2020]
Pix2R Results

- Adding shaping reward speeds up policy learning sparse rewards
- Sparse + Shaping rewards perform comparably to Dense rewards

[Goyal et al, 2020]
Extending Pix2R Dataset

● Each scenario has only one instance of each object
● Descriptions use simplistic language

● **Goal:** construct a dataset containing relational language
● Probe whether model is learning multimodal semantic relationships or just identification
● Motivate development of more robust models
Relational Data

- "Turn on the coffee machine on the left"
- "Press the coffee maker furthest from the button"
Video Generation

- Target object + duplicate object + distractors
- Train PPO with dense reward until success
- 6 tasks (button\_top, button\_side, coffee\_button, handle\_press\_top, door\_lock, door\_unlock)
- 5 scenarios per task
- 30 total scenarios
Collecting Natural Language Descriptions

- Amazon Mechanical Turk
- ‘Please ensure that the instruction you provide uniquely identifies the correct object, for example, by describing it with respect to other objects around it.’
- At least 3 descriptions per scenario (131 total)
- Manually create negative examples
Evaluation

- Can Pix2R encode relations between objects?
- Evaluate on test split of new data
- 6 scenarios, 3 descriptions, 5 runs each → 90 runs
Baselines and Models

- Sparse: PPO with binary reward
- Dense: PPO with expert Meta-World reward
Baselines and Models

- Sparse: PPO with binary reward
- Dense: PPO with expert Meta-World reward
- Original: PPO shaped by Pix2R trained on original dataset
Baselines and Models

- Sparse: PPO with binary reward
- Dense: PPO with expert Meta-World reward
- Original: PPO shaped by Pix2R trained on original dataset
- Augmented: PPO shaped by Pix2R trained on combined dataset
Baselines and Models

- Sparse: PPO with binary reward
- Dense: PPO with expert Meta-World reward
- Original: PPO shaped by Pix2R trained on original dataset
- Augmented: PPO shaped by Pix2R trained on combined dataset
- Reduced: PPO shaped by Pix2R trained on original dataset, excluding relational descriptions
Results

- All agents perform comparably, except sparse
- Reduced even performs slightly better
- Scenarios could be too simple
- Inconclusive, further experimentation needed
Conclusion

- Pix2R is robust to our specific challenge dataset
- No immediately obvious shortcomings
- Room for further probing through challenge datasets
Future Work

● Improving our existing challenge dataset
  ○ Refine environment generation to create more challenging scenarios
  ○ Multi-stage AMT pipeline for higher quality annotations
● Other challenge datasets
  ○ Can construct targeted, "adversarial" examples for any ML task
Acknowledgements

Dr. Ray Mooney

Prasoon Goyal