Scalable agent alignment

Jan Leike · BAGI 2019
What we want from ML

move 37

circling boat
What we want from ML

move 37

circling boat

@janleike
The agent alignment problem

How can we create agents that *behave* in accordance with the user's intentions?
“Preference payload” questions

- Whose preferences should the agent be aligned to?
- How should preferences of different users be aggregated?
- How should they be traded off against each other?
- When should the agent be disobedient?
“Preference payload” questions

- Whose preferences should the agent be aligned to?
- How should preferences of different users be aggregated?
- How should they be traded off against each other?
- When should the agent be disobedient?

These questions are **important**.

We’re **not discussing** these questions here.

We’re only considering the **technical problem** of aligning one agent to one user.
Desiderata

Economical

Scalable

Image sources:
https://www.porttechnology.org/
https://realanimetraining.com/
Assumption 1

Rather than formally specifying user intentions, we can instead learn these intentions to a sufficiently high accuracy.
For many tasks, evaluation of outcomes is easier than producing the correct behavior.
Evaluation is easier than behavior
Reward modeling
Reward modeling

What?

How?

Agent

User

Environment

DeepMind

@janleike
Some tasks are hard to evaluate

Human-level control through deep reinforcement learning

 Reward model

Agent

User

Environment

reward

observation

feedback

action

trajectory
Evaluation assistance tasks

- Well-written
- Novel
- Experiments correct
- Proofs correct

...
Evaluation assistance tasks

- Well-written
- Novel
- Experiments correct
- Proofs correct

...
Evaluation assistance tasks

- Well-written
- Novel
- Experiments correct
- Proofs correct

...
Recursive reward modeling
Recursive reward modeling
Challenges

- Amount of feedback
- Feedback distribution
- Reward hacking
- Unacceptable outcomes
- Reward-result gap
Challenges

- Amount of feedback
  - Online feedback
  - Off-policy feedback

- Feedback distribution
  - Leveraging existing data
  - Hierarchical feedback

- Reward hacking
  - Natural language
  - Model-based RL

- Unacceptable outcomes
  - Side-constraints
  - Adversarial training

- Reward-result gap
  - Uncertainty estimates
  - Inductive bias
Establishing trust

- Design choices
- Testing
- Interpretability
- Formal verification
- Theoretical guarantees

Safety certificates
Thanks! :)  

Blog post: https://goo.gl/azGMtA


---

Scalable agent alignment via reward modeling: a research direction

Jan Leike  
DeepMind

David Krueger  
Mila

Tom Everitt  
DeepMind

Miljan Martic  
DeepMind

Shahid Meini  
DeepMind

Shane Legg  
DeepMind

Abstract

One obstacle to applying reinforcement learning algorithms to real-world problems is the lack of suitable reward functions. Designing such reward functions is difficult in part because the agent only has an implicit understanding of the task objective. This gives rise to the agent alignment problem: how do we create agents that behave in accordance with the user’s intentions? We outline a high-level research direction to solve the agent alignment problem centered around reward modeling: learning a reward function from interaction with the user and optimizing the learned reward function with reinforcement learning. We discuss the key challenges we expect to face when scaling reward modeling to complex and general domains, concrete approaches to mitigate these challenges, and ways to establish trust in the resulting agents.

1 Introduction

Games are a useful benchmark for research because progress is easily measurable. Atari games come with a score function that captures how well the agent is playing the game; board games or competitive multiplayer games such as Dota 2 and StarCraft II have a clear winner or loser at the end of the game. This helps us determine empirically which algorithms and architectural improvements work best.

However, the ultimate goal of machine learning (ML) research is to go beyond games and improve human lives. To achieve this, we need ML to assist us in real-world domains, ranging from simple tasks like ordering food or answering emails to complex tasks like software engineering or running a business. Yet, performance on these and other real-world tasks is not easily measurable, since they do not come readily equipped with a reward function. Instead, the objective of the task is only indirectly available through the intentions of the human user.

This requires scaling to a fine limit. On the one hand, we want ML to generate creative and brilliant solutions like AlphaGo’s Move 37 (March, 2016)—a move that no human would have recommended, yet is completely inside the game in AlphaGo’s Grain. On the other hand, we want to avoid degenerate solutions that lead to undesired behavior like exploiting bugs in the environment simulator (Clark & Amodei, 2016, Leibman et al., 2018). In order to differentiate between these two outcomes, our agent needs to understand its user’s intentions, and robustly achieve these intentions with its behaviour. We frame this as the agent alignment problem:

How can we create agents that behave in accordance with the user’s intentions?