Mimicry and meeting halfway

Paul Christiano Follow
Sep 19, 2015 · 7 min read

Imitation learning can potentially avoid some of the pathologies associated with extremely powerful reinforcement learners. However, imitation learning has a different problem: a weak learner will not generally be able to imitate a powerful expert, even if it would be able to learn to do the underlying task in an acceptable way.

This post describes a response to this problem. The basic idea is that a human operator can carry out a task in a way that is designed to be easy to imitate. This approach can simultaneously (1) robustly avoid doing anything that is unacceptable to the user, (2) search for a way to achieve a task within the abilities available to an AI system.

Fix some task X, such as producing an explanatory document of a given level of quality and clarity. Note that we need not have any formal way of deciding whether the task has been accomplished. I'll describe a procedure for learning to accomplish X in a human-like way.

We will assume we have a learning algorithm which is capable of distinguishing successful implementations of X from failures, and another algorithm which is capable of learning to achieve X.

Our proposal is essentially an elaboration of generative adversarial modeling, or of the apprenticeship learning scheme proposed by Abbeel and Ng (the relationship is fleshed out here). In our elaboration, the process-to-be-modeled collaborates with the modeler (while making sure that it still achieves the task). Our elaboration is relevant when the generative task is intrinsically much more difficult than the discriminative task, such that this collaboration is necessary to make the task feasible.

The actors

We'll consider three actors:

- Hugh, the human overseer. (Hugh plays the role of the data distribution in the adversarial generative models framework).

- Arthur, the AI system. (Arthur plays the role of the generative model G.)
- Eva, another AI system acting as an evaluator. (Eva plays the role of the discriminative model D.)

We'll be able to teach Arthur to achieve the task X if it can be achieved by the "intersection" of Arthur and Hugh—we'll define this more precisely later, but note that it may be significantly weaker than *either* Arthur or Hugh.

The game

In each round of the game, Hugh and Arthur will attempt to perform the task.

Eva will look at their outputs and attempt to classify them as coming from Hugh or Arthur. We'll assume that Eva is a powerful enough learner that she can learn to distinguish between a player who is successfully achieving the task and one who is not.

Hugh will always perform the task, and subject to that constraint he will try to fool Eva into thinking that he is Arthur.

Arthur does not "know" anything about the task, he is simply trying to fool Eva into thinking that he Hugh.

We'll assume the game reaches an equilibrium, where one of the players is stuck: either Eva can't distinguish Arthur and Hugh, or Hugh can't succeed at the task while looking like Arthur, or Arthur can't look like Hugh.

Critically, at no point is anyone ever optimizing for performance or other real-world consequences—they are only ever optimizing for the outputs of this simple prediction game. All of the optimization power is in Hugh's decisions about how to achieve X.

We can imagine playing this game many times, giving each participant plenty of time to adjust to the behavior of the others. If the AI systems are able to generalize across domains, then each time we play the game it may involve a different domain. (And if the systems aren't able to generalize across domains, then it is inevitable that we will have to provide separate training data for each new domains.)

Performance

I mentioned before that this system matched the performance of the "intersection" of Hugh and Arthur. Intuitively by "intersection" I mean the system whose capabilities are the intersection of Arthur's capabilities and Hugh's, i.e. which can do something iff both Hugh and Arthur can do it. M

This definition is subtle, because it operates at every level simultaneously: for example, if Hugh can move only by walking, and Arthur can move only by crawling, then the intersection cannot move at all, even though both can move. More formally, we'll say that the intersection of Hugh and Arthur can accomplish a task, relative to Eva, if each of them can accomplish that task in such a way that Eva can't distinguish them.

To see why this is relevant (and to make it more precise), suppose that Hugh would like to train Arthur to achieve X, but cannot. What can we infer from this failure?

By hypothesis, Arthur is failing to achieve X. But by construction, Hugh always achieves X. And by assumption, Eva can distinguish success at X from failure at X. So Eva can distinguish between Arthur and Hugh.

Eva's distinguisher induces a partition on the set of ways to achieve X, into those classified as "Hugh" and those classified as "Arthur." Arthur is not able to find any action in the space marked Hugh, and Hugh is not able to find any action in the space marked Arthur.

This precisely describes what tasks can be Arthur can be trained to accomplished by this scheme.

We say: **the intersection of Hugh and Arthur (with respect to some learning algorithm Eva) can achieve X if Eva** *cannot* **learn a partition of ways of achieving X into two sets H and A such that:**

- **Hugh cannot learn to achieve X by a means in A.**
- **Arthur cannot learn to achieve X by a means in H.**

Note that in particular, if one of Hugh and Arthur can't achieve X, then the intersection of the two can't achieve X.

Conversely, if the intersection of Hugh and Arthur *can't* learn to do a task, then it seems intuitive to say that Arthur cannot accomplish that

task in a human-like way. So in one sense this scheme is the best we should hope for.

As with approval-direction in general, we will then need to us a bootstrapping scheme to scale up to Arthur's capabilities from the intersection of Hugh and Arthur. This bootstrapping process is conceptually similar to, but somewhat more subtle than, the process for approval-directed agents more broadly. (And I am correspondingly more uncertain about how often it would work.)

Safety

At no point in this scheme is any process explicitly or implicitly optimizing for any external consequence. All of the optimization occurs in Hugh's decision-making. This makes it seem much less likely that there will be any carefully engineered catastrophe.

We can get some additional reassurance from Eva's judgments. That is, it is very unlikely that Hugh would take any action with catastrophic consequences. If Eva is intelligent enough to recognize catastrophic consequences then she could use them as a signal that a decision was made by Arthur. As a result, Arthur's only incentive is to avoid catastrophic consequences (and we can also reject any action that Eva thinks is probably produced by Arthur).

Transparency

In this proposal the evaluator Eva plays two rolls:

1. It defines the mimicry task and provides feedback to Arthur. This is the same role played by the discriminative model in the adversarial generative models framework.

2. It defines the "be easy to imitate task" and provides feedback to Hugh.

Role [2] is necessary for the proposal to work. But in order for it to be effective, the models learned by Eva must be sufficiently comprehensible that Hugh can understand the constraints they impose on his behavior. This might be achieved either by providing fine-grained feedback so that Hugh can learn by trial and error, or by training a transparent model, or by training Eva to produce explanations of her judgments.

In general, Hugh may need to make extensive use of AI assistants in order to successfully model an appropriate behavior. There is no

bright line between better understanding Eva's model and the other aspects of the bootstrapping process.

With counterfactuals

We've described a procedure for training Arthur to perform a task. Actually performing this procedure is quite labor-intensive. As usual, the hope would be to train a model once, and then use it with only minimal human involvement. Counterfactual oversight provides a clean understanding of what exactly the model is being trained to do. But thinking about the resulting system is a little bit counterintuitive, so it it's worth thinking about what the final product looks like.

In production, Arthur is given enough data to understand his current situation, including the desires of the users (which are reflected in the counterfactual behavior of the overseer). He then outputs an action x.

Most of the time action x is simply taken, and we move on to the next decision. But sometimes (either randomly or when an adversarial system believe that Arthur may have made an error) a human overseer is contacted and involved with the decision.

At this point the execution of the entire system pauses, with the action x not yet taken. The overseer Hugh is given time to understand the situation (but is not shown the action x).

The evaluator Eva is woken up and also given enough data to understand the situation (but is not shown the action x). Hugh is given the opportunity to interact with Eva, to understand her model of the situation and how she will behave in the next step. He can also work with other AI assistants to help model an appropriate behavior.

Hugh ultimately produces an action x' which both (1) reflects Hugh's desires about what Arthur should do, and (2) is likely to fool Eva into thinking that x' was output by Arthur.

Eva is then given x' and x, and her goal is to guess which is which. If Eva succeeds, then she receives a payoff of 1 and Arthur receives a payoff of -1. If Eva fails, then she receives a payoff of -1 and Arthur receives a payoff of 1.

Arthur's output x is chosen in order to maximize the expected payoff in this round, assuming that the round actually happened.

Conclusion

This post suggests a simple scheme that may capture some of the benefits of directly imitating behaviors, without running into the same fundamental obstruction. There are many clear challenges for implementing anything long these lines; some can already be explored in the context of the generative adversarial models framework (for example, empirical results highlight the necessity of maintaining rough parity between Eva and Arthur), while others are distinctive to the role of the human overseer.

I expect there is room for many more sophisticated schemes that capture the best aspects of mimicry and maximization, but I think that this proposal should make us more optimistic about being able to find a working solution, and less concerned that there is a fundamental tradeoff between flexibility and safety.

(*Funding for this research was provided by a grant from the Future of Life Institute.)*