In defense of maximization

Paul Christiano Follow
Sep 19, 2015 · 3 min read

I've been thinking about AI systems that take actions their users would most approve of. The word "most" in that sentence can be a bit unsettling.

In this post I want to explain why you might be unsettled, and also why I think that these concerns probably aren't deal-breakers.

For concreteness: I'll imagine a system which is overseen by a human operator and proposes a sequence of actions. Most of the time a proposed action is taken without any human involvement. But in a small fraction of cases an action is reviewed by the overseer—before taking it—and each action is chosen to maximize how much the overseer would approve of it if it happened to be selected for review.

The concern

(Readers familiar with terms like "perverse instantiation" can safely skip this section.)

The basic problem is that we don't actually care about whether an AI system take actions we approve of. We are using approval as a proxy for leads-to-good-consequences.

But even if these two concepts *typically* overlap, they seem especially likely to come apart in the extremes.

For example, the "best" action may be optimized to be misunderstood by the overseer, in a way that leads us to believe that it has impossibly good consequences. Or the most persuasive argument might exploit human fallibility rather than honestly making a case, so that the "best" action is the one most compatible with the most effective tactics of persuasion. Or (in an especially extreme case) a proposed action may be such that merely viewing it compromises the overseer's computer, seizing control of the approval signal itself.

However we feel about the plausibility of these failures, it would be unfortunate to build a system that actively seeks them out. But that's exactly what an approval-maximizer does!

The responses

There are a few features of approval-directed systems that make me significantly less concerned about maximization:

1. Approval is given to actions rather than consequences. This significantly limits the scope for perverse instantiation. Our concern is now limited to deliberate misunderstandings, attacks on the overseer, and so on. This is much less troubling and apparently much easier to address than perverse instantiation over the space of all outcomes.

2. In the course of evaluating a proposed action, the overseer can enlist the help of other comparably powerful AI systems. These systems can help the overseer notice possible misunderstandings, identify the strongest objections to a proposed action, or correct errors in the overseer's reasoning. For example, rather than evaluating an action in isolation, the overseer can moderate an argument between two agents with alternative proposals.

3. Prior to evaluating a proposed action, an overseer can enlist the help of slightly weaker AI systems in order to identify and resolve possible problems in the approval-granting process. This is especially important for problems that can't be patched online (for example because they would immediately and irreversibly compromise the process).

4. If possible, the internal organization of our "maximizing" agent can itself be approval-directed—instead of maximizing, it is actually doing whatever we would most approve of. The operator will try not to approve of internal cognitive steps that search for perverse instantiations or plan malicious behavior, even if they would be "maximizing approval." (Of course this process needs to bottom out somewhere, but it may bottom out with routines that are weaker than the overseer, for which perverse instantiation is a much smaller and more familiar concern.)

(See also my discussion of maximization here.)

The conclusion

I don't have a credible concrete scenario where the "maximization" in approval-maximization seems especially problematic. The safety mechanisms in the previous section also feel plausibly sufficient. I think the biggest question is the extent to which approval-direction is plausible as a way of organizing the internal behavior of an AI system.

That said, I would feel more comfortable with approval-directed systems that *don't* apply their ingenuity to coming up with clever ways to exploit their users. I don't think that this concern is a deal-breaker, but I do think there is room for improvement.