Adversarial vs. active learning

Paul Christiano Follow
Oct 3, 2015 · 3 min read

My most recent proposal for AI control involves an adversarial process: one AI proposes a decision, while another decides whether to flag that decision as problematic. Similarly, one AI describes a justification for a decision, while another highlight steps of the justification that a human would be likely to object to.

This process may sound somewhat exotic and removed from practice. I want to argue that it is actually an intuitive approach to active learning. (Continuing today's theme.)

What training data to label?

This proposal addressees a simple problem: if we only label a small subset of of the data, then we are liable not to correct errors for a very long time.

For example, suppose that 99.999% of the decisions that our learning system makes are straightforward. Then if we label training data at random, 99.999% of our labels only help with the straightforward part of the problem, and we will have to label 1,000,000 data points before we have provided even 10 correctly labelled non-straightforward examples. Even if our learner learns very quickly, this will be a serious problem if getting the non-straightforward examples right is important.

Once our learner can handle the simple cases, we'd really like to redirect our attention to the 0.001% of remaining interesting cases.

Sometimes these will be easy to identify. For example, perhaps they are the cases where our learner is least confident. Or perhaps the straightforward cases are very similar to each other and anything that looks unusual is not straightforward.

Other times there will be more subtle clues. Perhaps there is a simple classifier that agrees with our learner on the easy cases but not the hard cases. Or perhaps there are features of the environment that predict instance difficulty. Or so on.

In general, it would be nice to apply all available machinery to try to identify the important cases to label—including training another similarly sophisticated learning system to identify these cases. In context, "useful to label" means roughly: "a label would change the original learner's behavior." This in turn means roughly: "a data point that would trip up the original learner."

And this is precisely the adversarial game that I described: one learner tries to make decisions that will do well *if* they are reviewed, and the other learner tries to call out decisions for review if they are likely to be reviewed badly. There are a few subtleties (for example: if the original learner does badly, we also want to know if they could really have done better), but the adversarial framework mostly resolves these issues in an intuitive way.

Arguments

The discussion above applies to a narrow kind of adversarial interaction: one player is picking individual decisions that are likely to be bad and challenging them.

An argument is a more general kind of adversarial interaction. However, it is often desirable for similar reasons.

For example, consider a learner tasked with producing a vacation plan. After this learner outputs a plan, an adversarial learner can try to identify the aspect of the plan most likely to be problematic (potentially with the collaboration of a human mediator)—e.g. that a connection between flights is very short given the history of delays for the first flight, and the vacation is not robust to missing the connection. This information can then be given to a human, who can evaluate whether it is problematic, arbitrating the argument. This data can then be used to train both learners.

Further machine learning systems might be used to help the human settle the conflicting claims, e.g. to assess how likely that particular flight really is to be delayed, or to predict how unhappy the user will be if they miss their connection. If we want each step of the discussion to be structured in a way that facilitates subsequent steps, it makes sense to frame the entire process as a longer argument with richer recursive structure. The participants in this argument are solving a reinforcement learning problem, planning for the entire process of the argument rather than a single step.

These elaborations are further from practice than the simple kind of active learning described in the last section. And probably today it wouldn't be worth using any techniques distinctive to reinforcement learning in this setting. But the basic idea is not especially exotic, and could be easily implemented today.