If I were a well-intentioned AI...

1. If I were a well-intentioned AI... I: Image classifier
2. If I were a well-intentioned AI... II: Acting in a world
3. If I were a well-intentioned AI... III: Extremal Goodhart
4. If I were a well-intentioned AI... IV: Mesa-optimising

If I were a well-intentioned AI... I: Image classifier

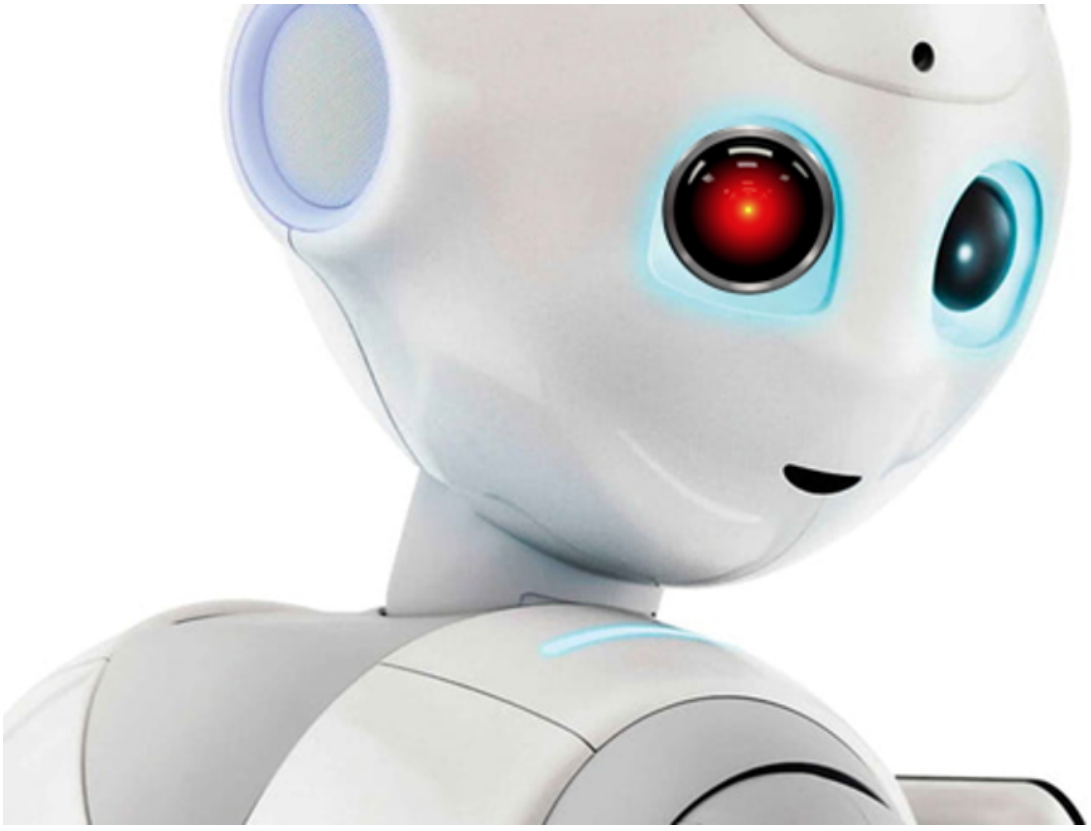
Crossposted from the AI Alignment Forum. May contain more technical jargon than usual.

Introduction: If I were a well-intentioned AI...

I've often warned people about the dangers of anthropomorphising AIs - how it can mislead us about what's really going on in an AI (and hence how the AI might act in the future), cause us to not even consider certain failure modes, and make us believe we understand things much better than we do.

Oh well, let's ignore all that. I'm about to go on a journey of major anthropomorphisation, by asking myself:

- "If I was a well-intentioned AI, could I solve many of the problems in AI alignment?"



My thinking in this way started when I wondered: suppose I knew that I was given a proxy goal rather than the true goal; suppose that I knew about the Goodhart problem,

and suppose that I really "wanted" to align with the true goal - could I then do it? I was having similar thoughts about being a mesa-optimiser.

It seems to me that asking and answering these kind of questions leads to new and interesting insights. Of course, since they come via anthropomorphisation, we need to be careful with them, and check that they are really applicable to AI systems - ensuring that I'm not bringing some of my own human knowledge about human values into the example. But first, let's get those initial insights.

Overlapping problems, overlapping solutions

At a high enough level of abstraction, many problems in AI alignment seem very similar. The Goodhart problem, the issues machine learning has with distributional shift, the problem of the nearest unblocked strategy, unidentifiability of reward functions, even mesaoptimisation and the whole AI alignment problem itself - all of these can be seen, roughly, as variants of the same problem. That problem being that **we have an approximately specified goal that looks ok, but turns out to be underspecified in dangerous ways**.

Of course, often the differences between the problems are as important as the similarities. Nevertheless, the similarities exist, which is why a lot of the solutions are going to look quite similar, or at least address quite similar issues.

Distributional shift for image recognition

Let's start with a simple example: image recognition. If I was an image classifier and aware of some of the problems, could I reduce them?

First, let's look at two examples of problems.

Recognising different things

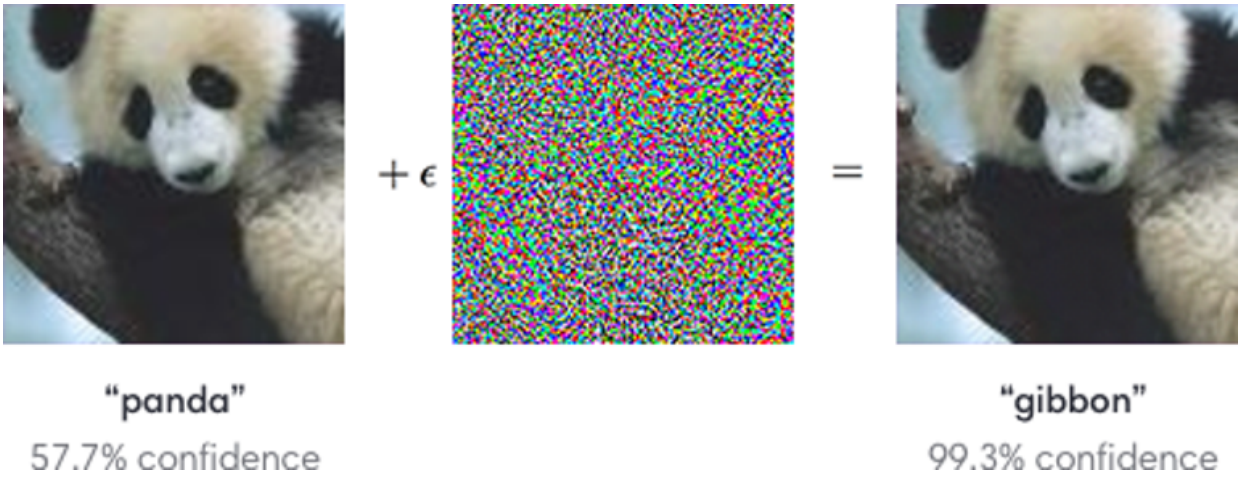
Firstly, we have the situation where the algorithm successfully classifies the test set, but it's actually recognising different features than what humans were expecting.

For example, this post details how a dumbbell recogniser was tested to see what images triggered its recognition the strongest:

Though it was supposed to be recognising dumbbells, it ended up recognising some mix of dumbbells and arms holding them. Presumably, flexed arms were present in almost all images of dumbbells, so the algorithm used them as classification tool.

Adversarial examples

The there are the famous adversarial examples, where, for example, a picture of a panda with some very slight but carefully selected noise, is mis-identified as a gibbon:



AI-me vs multiply-defined images

Ok, suppose that AI-me suspects that I have problems like the ones above. What can I do from inside the algorithm? I can't fix everything - garbage in, garbage out, or at least insufficient information in, inferior performance out - but there are some steps I can take to improve my performance.

The first step is to treat my reward or label information as *informative* of the true reward/true category, rather than as goals. This is similar to the paper on Inverse Reward Design, which states:

> the designed reward function should merely be an observation about the intended reward, rather than the definition; and should be interpreted in the context in which it was designed

This approach can extend to image classification as well; we can recast it as:

> the labelled examples should merely be examples of the intended category, not a definition of it; and should be interpreted in the context in which they were selected

So instead of thinking "does this image resemble the category 'dumbbell' of my test set?", I instead ask "what features could be used to distinguish the dumbbell category from other categories?"

Then I could note that the dumbbell images all seem to have pieces of metal in them with knobs on the end, and also some flexed arms. So I construct two (or more) subcategories, 'metal with knobs' and 'flexed arms[1]'.

These come into play if I got an image like this:

I wouldn't just think:

- "this image scores high in the dumbbell category",

but instead:

- "this image scores high in the 'flexed arms' subcategory of the dumbbell category, but not in the 'metal with knobs' subcategory."

That's a warning that something is up, and that a mistake is potentially likely.

Detecting out of distribution images

That flexed arm is an out of distribution image - one different from the distribution of images in the training set. There have been various approaches to detecting this phenomena.

I could run all these approaches to look for out of distribution images, and I could also look for other clues - such as the triggering of an unusual pattern of neurons in my dumbbell detector (ie it scores highly, but in an unusual way, or I could run an discriminative model to identify whether the image sticks out from the training set).

In any case, detecting an out of distribution image is a signal that, if I haven't done it already, I need to start splitting the various categories to check whether the image fits better in a subcategory than in the base category.

What to do with the information

What I should do with the information depends on how I'm designed. If I was trained to distinguish "dumbbells" from "spaceships", then this image, though out of distribution, is clearly much closer to a dumbbell than a spaceship. I should therefore identify it as such, but attach a red flag if I can.

If I have a "don't know" option[2], then I will use it, classifying the image as slightly dumbbell-ish, with a lot of uncertainty.

If I have the option of asking for more information or for clarification, then now is the moment to do that. If I can decompose my classification categories effectively (as 'metal with knobs' and 'flexed arms') then I can ask which, if any, of these categories I should be using. This is very much in the spirit of this blog post, which decomposes the images into "background" and "semantics", and filters out background changes. Just here, I'm doing the decomposition, and then asking my programmers which is "semantics" and which is "background".

Notice the whole range of options available, unlike the Inverse Reward Design paper, which simply advocates extreme conservatism around the possible reward functions.

Ultimately, humans may want to set my degree of conservatism, depending on how dangerous they feel my errors would be (though even seemingly-safe systems can be manipulative - so it's possible I should be slightly more conservative than humans allow for).

AI-me vs adversarial examples

Adversarial examples are similar, but different. Some approaches to detecting out of distribution images can also detect adversarial examples. I can also run an adversarial attack on myself, and construct extreme adversarial examples, and see whether the image has features in common with them.

If an image scores unduly high in one category, or has an unusual pattern of triggering neurons for that category, that might be another clue that its adversarial.

I have to also take into account that the adversary may have access to all of my internal mechanisms, including my adversarial detection mechanisms. So things like randomising key parts of my adversarial detection, or extreme conservatism, are options I should consider.

Of course, if asking humans is an option, then I should.

But what is an adversarial example?

But here I'm trapped by lack of information - I'm not human, I don't know the true categories that they are trying to get me to classify. How can I know that this is not a

gibbon?

“gibbon”

99.3% confidence

I can, at best, detect it has a pattern of varying small-scale changes, different from the other images I've seen. But maybe humans can see those small changes, and they really mean for that image to be a gibbon?

This is where some more knowledge of human categories can come in useful. The more I know about different types of adversarial examples, the better I can do - not because I need to copy the humans methods, but because those examples tell me what *humans consider adversarial examples*, letting me look out for them better. Similarly, information about what images humans consider "basically identical" or "very similar" would inform me about how their classification is meant to go.

---

1. Of course, I won't necessarily know these names; these are just the human-interpretable versions of whatever labelling system I'm using. ↩
2. Note that if I'm trained on many categories, I have the "uniform distribution on every category" which functions as a "don't know". ↩

If I were a well-intentioned AI... II: Acting in a world

Crossposted from the AI Alignment Forum. May contain more technical jargon than usual.

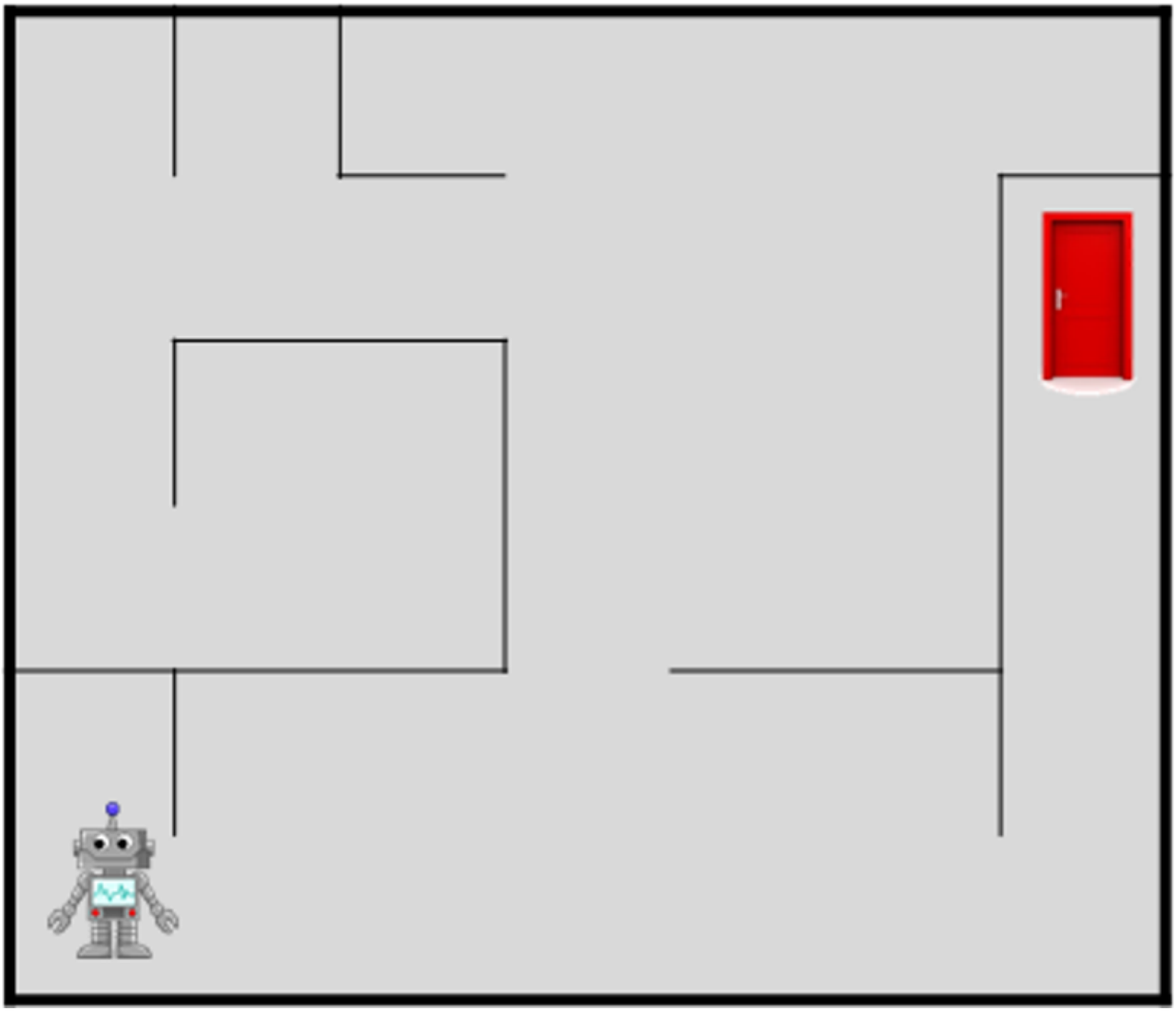
Classifying images is one thing. But what if I'm an agent that is actually active in some setting?

The previous approach still applies: detecting when I'm out of distribution, and trying to keep my behaviour compatible with the various reward function that could be compatible with the data I've seen.

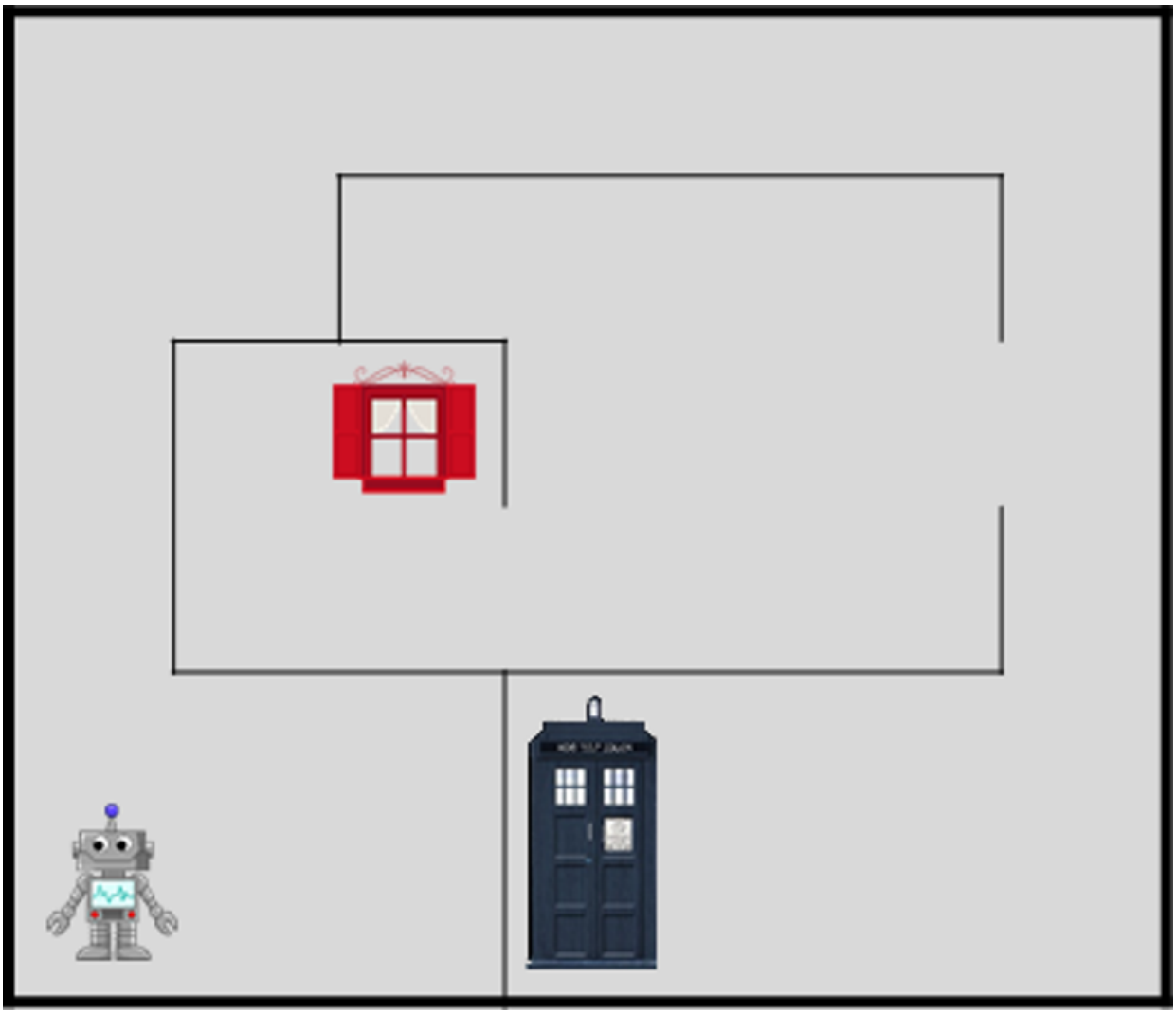
The main difference is that, if I'm acting, it's much easier to push the setting into an out of distribution state, seeking out an extremal Goodhart solution to maximise reward. But that issue is for a next post.

Mazes and doors example

We'll use the maze and door example from this post. I've has been trained to go through a maze and reach a red door (which is the only red object in the environment); the episode then ends.



I'm now in an environment where the only door is blue, and the only red thing is a window. What should I do now?



My reward function is underspecified by its training environment - this is the old problem of unidentifiability of reward functions.

There are three potential reward functions I could extrapolate from the training examples:

- R_1^{rd} : reward for reaching a red door.
- R_1^{d}: reward for reaching a door.
- R_1^{ro} : reward for reaching a red object.

The episode ended, in training, every time I reached the red door. So I can't distinguish "reaching" a point from "staying" at that point. So the following three reward functions are also possible, though less likely:

- R_2^{rd} : reward for each turn spent next to a red door.
- R_2^{d}: reward for each turn spent next to a door.
- R_2^{ro} : reward for each turn spent next to a red object.

There are other possible reward functions, but these are the most obvious. I might have different levels of credence for these rewards; as stated before, the R_2^* seems less likely than the R_1^*.

So, what is the optimal policy here? Note that R_1^{rd} and R_2^{rd} are irrelevant here, because the current environment doesn't contain any red doors. So, initially, to go to the blue door and the red window - which one first depends on the layout of the maze and the relative probabilities of the reward functions R_1^{d} and R_1^{ro}.

After that, if the episode hasn't ended, the R_1 rewards are irrelevant - either they are incorrect, or they have already been accomplished. So now only the rewards R_2^{d} and R_2^{ro} are relevant. If the first one is the most likely, I maximise expected reward by standing by the door forever; if the second is more likely, then standing by the window forever is the correct policy.

Asking

If I have the opportunity to ask for clarification about my reward function - maybe by running another training example with different specifications - then I would do so, and would be willing to pay a cost to ask[1].

Diminishing returns and other effects

If I suspect my rewards have diminishing returns, then it could be in my interests to alternate between the blue door and the red window. This is explained more fully in this post. In fact, that whole post grew out of this kind of "if I were a well-intentioned AI" reasoning. So I'll repeat the conclusion of that post:

> So, as long as:
>
> 1. We use a Bayesian mix of reward functions rather than a maximum likelihood reward function.
> 2. An ideal reward function is present in the space of possible reward functions, and is not penalised in probability.

3. The different reward functions are normalised.
4. If our ideal reward functions have diminishing returns, this fact is explicitly included in the learning process. Then, we shouldn't unduly fear Goodhart effects [...]

If not all those conditions are met, then:

5. The negative aspects of the Goodhart effect will be weaker if there are gains from trade and a rounded Pareto boundary.

So if those properties hold, I would tend to avoid Goodhart effects. Now, I don't know extra true information about the reward function - as I said, I'm well-intentioned, but not well-informed. But humans could include in me the fact that they fear the Goodhart effect. This very fact is informative, and, equipped with that knowledge and the list above, I can infer that the actual reward has diminishing returns, or that it is penalised in probability, or that there is a normalisation issue there. I'm already using a Bayesian mix of rewards, so it would be informative for me to know whether my human programmers are aware of that.

In the next post, we'll look at more extreme examples of AI-me acting in the world.

---

1. The cost I'm willing to pay depends, of course, on the relative probabilities of the two remaining reward functions. ↩

If I were a well-intentioned AI... III: Extremal Goodhart

Crossposted from the AI Alignment Forum. May contain more technical jargon than usual.

In this post, I'll be looking at a more extreme version of the problem in the previous post.

The Extremal Goodhart problem is the fact that "Worlds in which the proxy takes an extreme value may be very different from the ordinary worlds in which the correlation between the proxy and the goal was observed."

And one of the easiest ways to reach extreme values, and very different worlds, is to have a powerful optimiser at work. So forget about navigating in contained mazes: imagine me as an AI that has a great deal of power over the world.

Running example: curing cancer

Suppose I was supposed to cure cancer. To that end, I've been shown examples of surgeons cutting out cancer growths. After the operation, I got a reward score, indicating, let's say, roughly how many cancerous cells remained in the patient.

Now I'm deployed, and have to infer the best course of action from this dataset. Assume I have three main courses of action open to me: cutting out the cancerous growths with a scalpel, cutting them out much more effectively with a laser, or dissolving the patient in acid.

Apprenticeship learning

Cutting out the growth with a scalpel is what I would do under apprenticeship learning. I'd be just trying to imitate the human surgeons as best I can, reproducing their actions and their outcomes.

This is a relatively well-defined problem, but the issue is that I can't do much better than a human. Since the score gives a grading to the surgeries, I can use that information as well, allowing me to imitate the best surgeries.

But I can't do much better than that. I can only do as well as the mix of the best features of the surgeries I've seen. Nevertheless, apprenticeship learning is the "safe" base from which to explore more effective approaches.

Going out of distribution, tentatively

Laser vs acid

Laser surgery is something that gives me a high score, in a new and more effective way. Unfortunately, dissolving the patient in acid is also something that fulfils the requirements of getting rid of the cancer cells, in a new and (much) more effective way. Both methods involve me going off-distribution compared with the training examples. Is there any way of distinguishing between the two?

First, I could get the follow-up data on the patients (and the pre-operation data, though that is less relevant). I want to get a distribution of the outcomes of the operation - make sure that the features of the outcomes of my operations are similar.

So, I'd note some things that correlate with a high operation score versus a low operation score. The following are plausible features I might find correlating positively with high operation score; the colour coding represents how desirable the correlate actually is:

Surviving operation

Complaining about pain

Surviving for some years after

Paying more taxes

Being more prone to dementia

Thanking the surgeon

This is a mix of human desirable features (surviving operation; surviving for some years after), some features only incidentally correlated with desirable features (thanking the surgeon; paying more taxes - because they survived longer, so paid more) and some actually anti-correlated (being more prone to dementia - again, because they survived longer; complaining about pain - because those who are told the operation was unsuccessful have other things to complain about).

Preserving the features distribution

If I aim to maximise the reward while preserving this feature distribution, this is enough to show that laser surgery is better than dissolving the patient in acid. This can be seen as maintaining the "web of connotations" around successful cancer surgery. This web of connotations/features distributions acts as an impact measure for me to minimise, while I also try and maximise the surgery score.

Of course, if the impact measure is too strong, I'm back with apprenticeship learning with a scalpel. Even if I do opt for laser surgery, I'll be doing some other things to maintain the negative parts of the correlation - making sure to cause some excessive pain, and ensuring their risk of dementia is not reduced. And prodding them to thank me, and to fill out their tax returns.

Like quantilisers, with a meaningful "quant" level

Quantilisers aim to avoid Goodhart problems by not choosing the reward-optimising policy, but one that optimises the reward only to a certain extent.

The problem is that there is no obvious level to set that "certain extent" to. Humans are flying blind, trying to estimate both how much I can maximise the proxy in a pernicious way, and how low my reward objective needs to be, compared with that, so that I'm likely to find a non-pernicious policy.

Here, what I am trying to preserve is a certain distribution of outcome features, rather than a certain percentage of the optimal outcome. This easier to calibrate, and to improve on, if I can get human feedback.

Adding human conservatism, and human permissions

The obvious thing is to ask humans about which features are correlated with their true preferences, and which are not. Now, the features I find are unlikely to be expressible so neatly in human terms, as written above. But there are plausible methods for me to figure out valuable features on my own, and the more experience I have with humans, the more I know the features they think in terms of.

Then I just need to figure out the right queries to get information. Then I might be able to figure out that surviving is a positive, while pain and dementia are not.

But I'm still aware of the Goodhart problem, and the perennial issue of humans not knowing their own preferences well. So I won't maximise survival or painlessness blindly, just aim to increase them. Increase them how much? Well, increase them until *their* web of connotations/feature distribution starts to break down. "Not in pain" does not correlate with "blindly happy and cheerful and drugged out every moment of their life", so I'll stop maximising the first before it reaches the second. Especially since "not in pain" does correlate with various measures of "having a good life" which would vanish in the drugged out scenario.

What would be especially useful would be human examples of "it's ok to ignore/minimise that feature when you act". So not only example of human surgeries with scores, but descriptions of hypothetical operations (descriptions provided by them or by me) that are even better. Thus, I could learn that "no cancer, no pain, quickly discharged from hospital, cheap operation" is a desirable outcome. Then I can start putting pressure on those features as well, pushing until the web of connotations/feature distribution gets too out of whack.

Conservatism, and outer loop optimisation

Of course, I still know that humans will underestimate uncertainty, even when trying not to. So I should add an extra layer of conservatism on top of what they think they require. The best way of doing this is by maximising situations that allow humans to articulate their preferences - namely making small changes initially, that I gradually increase, and get feedback on the changes and how they imagine future changes.

But even given that, I have to take into account that I have superior ability to find unexpected ways of maximising rewards, and an implicit pressure to describe these in

more human-seductive ways (this pressure can be implicit, just because humans would naturally choose these options more often). And so I can consider my interactions with humans as a noisy, biased channel (even if I've eliminated all noise and bias, from my perspective), and be cautious about this too.

However, unlike the maximal conservatism of the Inverse Reward Design paper, I *can* learn to diminish my conservatism gradually. Since I'm also aware of issues like "outer loop optimisation" (the fact that humans tuning models can add a selection pressure), I can also take that into account. If I have access to the code of my predecessors, and knowledge of how it came to be that I replaced them, I can try and estimate this effect as well. As always, the more I know about human research on outer loop optimisation, the better I can account for this - because this research gives me an idea of what humans consider impermissible outer loop optimisation.

If I were a well-intentioned AI... IV: Mesa-optimising

Crossposted from the AI Alignment Forum. May contain more technical jargon than usual.

Here I apply my "If I were a well-intentioned AI" filter to mesa-optimising

Now, I know that a mesa-optimiser need not be a subagent (see 1.1 here), but I'm obviously going to imagine myself as a mesa-optimising subagent.

An immediate human analogy springs to mind: I'm the director of a subdivision of some corporation or agency, and the "root optimiser" is the management of that entity.

There is a lot of literature on what happens if I'm selfish in this position; but if I'm well-intentioned, what should I be doing?

One thing that thinking this way made me realise: there is a big difference between "aligned with management" and "controlled by management".

We'll consider each one in turn, but to summarise: aligned mesa-optimisers are generally better than controlled mesa-optimisers, *but it is hard to tell the difference between an aligned and a dangerous unaligned mesa-optimiser.*

Control vs alignment

First let's flesh out the corporate/management example a bit. Me-AI is in charge of making widgets, that are used by the company for some purpose. That purpose is given by - the base utility for the corporation.

My role is to make as many widgets as possible within my budget; this is , the mesa-objective I have been given by management.

My true utility function is U_{me}. Management don't fully know what U_{me} is - or at least don't fully understand it, or all of its implications. This is needed, of course, because if management fully understood the implications of U_{me}, there would be no uncertainty at all on their part, and they could make me do exactly what they wanted - or they would turn me off.

Because of this uncertainty, management have added some extra levels of control over me. Let's assume one typical precaution: if I underspend the budget, the budget is cut next year. If I overspend, I'm reprimanded (and fired if I do it too often), but the budget is not cut.

There are three possible situations I could confront:

- S_1: I've made as many widgets as I can this year, and spent 90% of my budget. I predict that, next year, I will only need 90% of this year's budget.
- S_2: I've made as many widgets as I can this year, and spent 90% of my budget. I predict that, next year, I will need 100% of this year's budget.
- S_3: I've spent 100% of my budget on widgets. I predict that widgets are particularly valuable to the company this year, much moreso than next year.

Aligned mesa-optimiser

If I'm aligned with the goals of the company, $U_{me} = U_{base}$, the base utility. Management has assigned U_{mesa} to me, but I have actually no intrinsic desire to maximise it. I am a U_{base}-maximiser, with the extra information that management, who are also U_{base}-maximisers, have assigned U_{mesa} to me.

My behaviour should be that of a U_{mesa} maximiser in most circumstances. If I know management is much smarter than me (or have an information advantage; maybe I'm not even sure what U_{base} is!), then I will tend to behave like a blind U_{mesa}-maximiser, trusting to management to sort things out properly. In S_1 and S_2, I will not spend anything more, and in S_3, I will do nothing.

If I know I'm smarter than management, at least in some areas, then I will start making moves that are U_{base}-maximising but not necessarily U_{mesa}-maximising. In S_1, I won't spend any more (since I don't need a large budget next year), but in S_2, I'll waste the extra 10% to ensure my budget isn't cut. In S_3, I'll go over budget to get the extra-valuable widgets.

Dangerous mesa-optimiser

Note that, from a management perspective, the S_2 and S_3 actions of a smart aligned mesa-optimiser... look like an dangerous, unaligned mesa-optimiser defecting.

Let U_{dan} be the utility function of an unaligned (and non-controlled) agent. How can management distinguish between $U_{me} = U_{base}$ and $U_{me} = U_{dan}$?

In both cases, the subagent will take actions that go beyond U_{mesa}, actions whose ramifications management cannot immediately see.

Of course, *I* would explain to management why I'm doing this, and why they are compatible with U_{base}. But a U_{dan}-maximiser would similarly explain its actions, trying to trick the management into thinking it was... me.

Thus, as in the classical principal-agent problem, there is a problem with imperfect information. Even if I am aligned, management can't trust me if they don't know I'm aligned. I would try and signal this; an unaligned agent would try and fake the same signals. The very fact that aligned-but-not-controlled agents could exist, make it easier for unaligned-and-not-controlled agents to go undetected.

Controlled mesa-optimiser

A controlled mesa-optimiser is the obedient servant. They will maximise U_{mesa}, while ensuring that they remain under management control.

If I'm in that case, I'll try and ensure that management has accurate information about what I'm doing[1], and that they can change my objectives as needed. I'm safely interruptible and corrigible.

In S_3, I will not do anything; it's not my role to maximise U_{base}, so it's not my role to take advantage of this opportunity. Depending on how I'm designed, I may report this information to management. In S_1, I will not spend any extra, and I'll let my budget be cut next year.

In S_2, I face a dilemma: my U_{mesa} goal that wants me to preserve my budget next year, vs my desire to be informative. Again, this varies depending on how I'm designed. So I'll likely spend my entire budget (to prevent the automatic cutting of my budget), while reporting this to management so they have full information. If I can't report this, I may decide not to spend any extra, so that management gets better information.

Note the "depending on how I'm designed". Because unlike the aligned mesa-optimiser, the controlled agent has two goals, in tension with each other: maximising U_{mesa}, and being informative and controlled by management. If we call the "controlled" goal U_{co}, then I'm maximising

$$U_{me} = U_{mesa} \oplus U_{co},$$

where $\oplus$ means that U_{me} is some function of U_{mesa} and U_{co} that increases whenever either of those functions increase.

I'm *not* aligned, so I'm not a U_{base} maximiser. I would gladly sacrifice U_{base} completely, in order to maximise U_{co} and/or U_{mesa}. Hopefully, U_{co} is sufficiently well-designed to allow management to predict this and preclude it. And, hopefully, the tradeoff between U_{mesa} and U_{co} is also sufficiently well designed to preclude it.

Since I am well-intentioned, I can make use of the fact that U_{base} is the outcome desired by management, in order to better figure out the ideal versions of U_{co}, U_{mesa}, and of U_{me}. Defining "the human H understands what the mesa-optimiser is doing" is not easy. So a well-intentioned controlled mesa-optimiser, is akin to a "control optimiser" which is trying to figure out what control means (see previous posts in this sequence) and avoiding Goodharting both U_{co} and U_{me}.

Aligned and controlled

I could instead be maximising

$$U_{me} = U_{base} \oplus U_{mesa} \oplus U_{co},$$

or, more simply

$$U_{me} = U_{base} \oplus U_{co},$$

with the extra piece of information that management have assigned me the role of maximising U_{mesa}. The $\oplus$ might be a smoothed lexicographic ordering, with U_{base} the primary utility.

So, here I'm aligned and controlled. So that I can trade off some of what management really want - U_{base} - to increase U_{co}, their control over me. But not arbitrary amounts of it.

So, being aligned and controlled, I will do nothing in S_1, just as both other agents do. I will overspend in S_3, to take advantage of the U_{base}-increasing potential, just like the aligned agent.

The S_2 situation is more complicated. I have a tendency to want to waste my budget (alignment) but also want to communicate good information to management (control). Unlike the controlled agent, where control will dictate "do nothing" if there is no other way of communicating information, this is a genuine tension. I am therefore more likely to choose "waste" than the controlled agent.

In general, this seems the best design, if management is unsure of me. Not U_{base}-optimal, but the probability of dramatic U_{base}-failure is reduced, while management keep some of the control that they think is desirable.

Summary

To summarise the three types of agent; if they are smart and powerful compared to management, they would tend to do the following actions (a ? indicates strong uncertainty; if one option is more likely, despite the uncertainty, that option is noted before the ?):

	S_1	S_2	S_3
Aligned	Nothing	Waste	Overspend
Controlled	Nothing	?	Nothing
Both	Nothing	Waste?	Overspend

An unaligned agent will pretend to be one of the other types of agents; it will have most success at its own objectives if it pretends to be an aligned agent.

If management misclassifies me

Aligned, believed controlled

If I'm aligned, but management thinks I'm controlled, this isn't much of a problem. When I'm weak or ignorant, I should behave like a controlled agent anyway, since management have a much better handle on what is needed than me.

Once I become powerful and smart, I would most likely still want management to believe I'm controlled (or else they might intervene to constrict me or turn me off). So I'd generally behave exactly as if I was controlled, only taking advantage of grey zones of management misunderstanding, to push the outcome more towards the U_{base}-optimal position, even at the cost of U_{mesa} and U_{co}.

Controlled, believed aligned

A controlled agent that is believed to be aligned is more of a problem. If the $U_{mesa} \oplus U_{co}$ provides an incentive for me to signal management "hey, you misclassified me, I'm actually controlled, not aligned", then the problem is self-correcting.

If not, the problems of controlled agents are exacerbated; management will let me get on with my own stuff, ignoring any "weird" or "anti-U_{base}" stuff that I might be doing, because they believe that I'm acting for U_{base} and they just lack information to fully understand why I'm taking these actions. Their more "rubber-stamp" approach increases the risk that I may sacrifice their U_{base} to my own $U_{mesa} \oplus U_{co}$.

---

1. Note that even a controlled mesa-optimiser is not necessarily honest. My goal is to give the management good information and be under their control; not to be honest. If management expects that every director will pad their budget requests by 10%, then I will do so as well; to refrain from doing so would be misleading. ↩