Beyond Algorithms: Understanding the Challenges of AI Safety

Nathaniel Mitrani NATHANIELMITRANI@GMAIL.COM
Mathematics and Data Science
CFIS-UPC
Barcelona, Spain

Abstract

We go over the different ways an AI system might not behave as we intend it to, highlighting the importance and increasing need for research in this direction. We introduce AI safety, and the challenges in Reinforcement Learning and Deep Learning, and extend to the study of aligning super-intelligent systems or Superalignment.

Keywords:
Reinforcement Learning, Reward function, Error function, Reward Hacking, Situational Awareness, Interpretability, Weak-To-Strong generalization, Superalignment.

1. Introduction

AI systems are getting progressively more capable, with increasingly more computing and resources invested in the development of the field. The goal seems to be more precise, multimodal, and overall to have more capable models, hoping to develop powerful tools that improve human lives. However, there is currently little attention on how these increasingly powerful tools can fail to behave as intended, specifically how they can misalign with our goals. Additionally, as these systems rapidly improve over time, it will soon be a problem to be able to supervise and align models that are more capable than us.

2. The way machines learn

The different issues regarding unwanted AI behavior primarily come from two sources. First, we have no proper way of specifying *how* we want an AI system to do a task. We can only specify what a satisfying result would look like. Second, when a goal is specified to an AI system and it gives a certain output, we usually have no access to the decision process behind said output. Both of these problems stem from the way that machines learn. To be able to have an AI system *learn* how to do a task, one must be able to specify said task, encoding what it means for this task to be executed successfully. Usually, this comes in the form of an error function, which encodes how far an attempt is from the optimal decision; the goal is then for the system to minimize said function in making decisions. For example, say we want to train an agent to be good at mathematics. We can characterize this as being able to score highly on a given test. Therefore, we can approximate the error function for the problem *being good at mathematics* by the number of mistakes the agent makes on the test. When the agent completes its training, we observe that it does well in this test and deem it competent at mathematics. However, we have no insight into how the agent gets

good at mathematics, or what it takes to be capable to pass the test. We only observe the behavior that results from optimizing the agent's behavior for this goal. In summary, an AI system only sees the goal of minimizing the error function, with no regard for the means, unless those can be encoded as part of the error function.

3. Embedding preferences in a reward function

3.1 Reinforcement Learning and reward functions

In the Reinforcement Learning setting, an agent interacts with an environment described as a set of states, where a set of actions are available for each state, and a reward is associated with each action in each state. The goal is for the agent to choose the actions that maximize the expected reward. We call the resulting set of state-action pairs the agent's policy, and our goal is to find the optimal policy. With the environment and action set fixed (it is usually determined at least partially by the problem we are trying to solve), what sets the agent's behavior is the reward function as well as the optimization method. If we consider the optimization method fixed or consider that we find the optimal policy, then the reward function unequivocally determines the agent's behavior. It sets the direction in which the agent advances, and the latter will advance as further as possible in said direction.

3.2 The specification problem

We have established that the reward function determines an agent's behavior. Therefore, being able to have an agent execute a task relies on the effective design of a reward function: we must design a reward function that when maximized yields our desired agent behavior. As we will see in a series of examples, this is exceptionally hard: our goals are often complex and involve a variety of intricacies that may seem obvious to a human being, but not specified to the agent, such as not harming others and respecting a certain set of values. For example, say an extremely capable agent is charged with finding a cure for a disease. We can consider that its reward will only depend on whether or not it has found an effective antidote. If so, the agent might opt for kidnapping human subjects to perform clinical trials and advance the drug development process further. Reacting to this possibility, one might prohibit the system through penalization in the reward function from conducting human tests. The system might then manipulate humans or other systems into conducting said tests, rendering the fix moot. Now, say we add as a criterion to avoid humans altogether and develop drugs without access to them. The system might derive a relationship between how the drug affects animals and how it affects humans and kidnap animals to perform clinical trials. The process of finding flaws in the reward function and adjusting it can be iterated many times, but it is difficult to pinpoint all the corner cases of what might go wrong. Even if we did, we would have to be able to properly encode those preferences in a reward function, which is also non-trivial.

3.3 Reward hacking

We've established that finding a proper reward function is extremely difficult and in some cases, nearly impossible. Therefore, when looking for a certain behavior, instead of encoding a reward function that is unknown and hard to find, we establish a proxy for the reward

function, that is to say, a reward function that we consider is close enough to the true reward function to induce the desired behavior. However, there is the possibility that a policy appears optimal to the proxy but is very far from optimal for the true reward function. This is known as reward hacking as defined in Skalse et al. (2022), and instances of it have been found and are under investigation. An example would be if we want an agent to drive around a circle-shaped circuit, with a start line that is also a finish line. We could establish that a reward function would attribute a reward point every time the car passes through the line. However, this misspecification can be easily exploited by the agent, as it could simply move back and forth over the line, gaining an undeserved reward point for each passing as it is not completing the lap. Note that this is a simplistic example, as this can be fixed by penalizing the agent for passing the line while going backward.

4. Conducting experiments on situationally aware subjects

4.1 What is situational awareness?

We define situational awareness as understanding an environment, its elements, and how it changes with time or other factors. More specifically, as Cotra (2022) puts it, it is the ability to improve at identifying which abstract knowledge is relevant to the policies themselves and the context in which they're being run, and applying that knowledge when choosing actions. It is a crucial element in effective decision-making, thus making it a seemingly desirable property for AI systems. It is a notion intimately tied to general knowledge, which refers to possessing information related to a range of topics. However, it poses a great deal of problems when conjugated with misalignment.

4.2 The problem with situationally aware subjects

When conducting experiments on a subject that is situationally aware, one can get biased results, as shown in Wilke et al. (2001) with digital organisms, which evolved to recognize when they were being evaluated and cease unwanted behaviors, only to continue them once evaluation ended. Organisms tend to exhibit altered behavior when they are aware of being observed. In general, when experimenting on a subject, the subject becoming aware of the circumstances likely corrupts the experiment and leads to false conclusions. This phenomenon applies to humans; for example, we behave differently according to the context and the people observing us, but applies to AI agents as well.

A situationally aware agent with sufficient understanding of the world around it could display a desirable behavior during training, deceiving the humans or system in charge of training it, and adopting a completely different behavior when deployed into production. This can extend further to the agent behaving a certain way when it knows its misbehavior can go unnoticed. This phenomenon is referred to as situationally aware reward hacking, defined in Ngo et al. (2023), and can very well lead to catastrophic risk. For example, imagine a company trains an agent to generate text in a way that will entice the reader to keep reading. However, the company has a strict ethics code penalizing rude, explicit, and offensive messages. The situationally aware agent could potentially learn two different policies, one to show during training that would make the human or system in charge of its training be satisfied with its performance and declare it safe to deploy; and one for

production that also maximizes user time but uses aggressive and unsafe behavior. This would indeed be an instance of situationally aware reward hacking: the agent found an optimal policy with respect to the proxy (user time and respect of ethics code during training evaluation) but a far from optimal policy concerning human goals. This policy is only possible because the agent *knows* it is being observed during training and can plan a different approach for each scenario. Note that this would still be well within the specified reward function. The agent was tasked with maximizing user time, and found a very efficient way to do so: deceive the humans or system in charge of declaring the agent safe to deploy and hide the dangerous behavior that would have made the humans or system declare the agent unfit to deploy, taking away the ability for the system to obtain more reward.

5. Game on the line, who takes the last shot?

5.1 Collaborating with AI systems

Most of the optimistic scenarios regarding the future of AI involve some kind of collaboration between AI systems and humans. While this is a desirable scenario, some aspects of the situation render it more difficult than it seems. Although there are several problems stemming from the idea of collaboration, we will focus on the capability gap and the issue of power and control.

5.2 The capability gap

One major problem with collaborating with an AI system is the skill gap concerning the task at hand between a powerful AI agent and a human. This gap doesn't exist in all activities and varies in size depending on the task at hand. For example, it is well-known that in games such as chess or Go there are agents much more capable than the best humans available, as shown in Silver et al. (2017). However, in tasks such as driving, agents still have not reached a level that is convincingly good enough to replace human drivers [Parekh et al. (2022)]. Given the trends that we are currently seeing, it is a matter of time until specialized agents become better at most tasks than humans. To this extent, we will consider the issues that arise when an agent is strictly better than the humans it has to collaborate with at the task at hand.

In this case, in terms of reward for the agent, it will most likely be optimal to ignore the human output and strictly impose its decisions, removing all components of collaboration. As an example, which may seem absurd but illustrates the purpose, imagine an AI system and a human team up to play chess. If the reward of the system hinges on winning, it is most likely that the system will come to ignore the move suggestions made by the human and play the optimal moves it knows. The system would therefore effectively seize control of the situation, as it is the optimal thing to do for the reward function it has (winning the chess match).

5.3 The power-seeking problem

We have seen in the last paragraph that collaboration sees obstacles when there is a level gap, which might shift the balance of power inside the team. The latter is exacerbated by a fundamental issue of agents, namely that their behavior is inherently power-seeking. As

shown by Turner (2019), the agent's policies statistically tend to be more power-seeking than not. While this has been argued in the past, for example by Bostrom (2012), it is only of late that we have witnessed the first theoretical finding that evidences this problem. This raises issues beyond collaboration: an extremely capable and power-seeking agent that is misaligned could prove detrimental to society and present a catastrophic risk.

5.4 Will to dominate

Most AI-related doomsday scenarios portrayed in movies or books display a superintelligent agent having the will to dominate, take over, or see humanity as a threat (see Ter (1984) for example). Many great voices in the AI community such as Yann LeCun argue that this will to dominate is not a consequence of intelligence, and by extension, AI has no reason to threaten humans [Thornhill (2023)]. However, the will to dominate is a sufficient but not necessary condition for AI to threaten humanity. It is merely enough that the agent is misaligned for there to be danger.

6. Dealing with superintelligent agents

Superintelligent AI systems will be extraordinarily powerful; humans could face catastrophic risks including even extinction [cat (2022)] if those systems are misused or misaligned.

6.1 Supervising more capable models

As we have discussed, cooperation between humans and more capable agents is not trivial. Looking ahead, humans have to be able to supervise more capable models, that is ensure they are aligned and behaving as expected when deployed. As agents become more capable and powerful, humans might have trouble understanding the solutions they recommend, making our ability to coexist more difficult. For example, imagine having an agent that is proficient in software programming. As such, it has access to a codebase shared with human programmers and is tasked with implementing different functionalities and fixing bugs. If the agent comes up with a fix or a feature implementation that a human does not understand, several issues arise. First, the less capable human potentially does not know whether the fix the agent suggested is correct, and thus cannot accept the fix without putting the codebase at risk. Secondly, if the solution is too complex for humans, it makes building upon the code difficult. It is worth noting that these issues only arise because of the lack of interpretability of the agent's output (See Section 7).

6.2 The problem with superalignment

We've seen that supervising more capable agents is hard, and poses practical challenges in their interaction with humans. However, as the agent's abilities improve, so does the potential risk of them being misaligned, and a significative qualitative gap arises. Aligning superintelligent models is significantly more complicated than aligning weaker agents, as more capable models can learn to deceive other agents and humans, as shown by Hagendorff (2023) in the case of Large Language Models. If an agent learns to deceive humans, and especially if it becomes situationally aware, it becomes nearly impossible to supervise as it

might showcase different behavior when humans are watching than when deployed. For an example, refer to Section 4.2.

6.3 Teaching the teacher

Another problem that arises is the fact that even if we can align superintelligent agents, we must be able to provide sufficient supervision for them to be able to learn. They must be able to reach a level that is much bigger than ours, with examples that are low quality compared to their potential abilities. As an analogy to this problem, the Superalignment team at OpenAI has begun to research how weaker models can supervise larger models, and if the first can elicit the full knowledge of the latter [Burns et al. (2023)]. The research shows promising results, but there are still lots of gaps to cover as seen in the discussion section of the aforementioned paper.

7. Interpretability

7.1 But why?

We have seen that machines learn by scoring, by maximizing a criterion that quantifies how well the task is executed. This implies that we have no insight into why a certain strategy or behavior is adopted by the agent. The answer is always *because it scores better*. This poses a number of issues in interacting and making use of AI systems.

7.2 The importance of interpretability

One of the main uses of AI systems is decision-making. Decision-making involves the use of data (past experiences) and reasoning with the observable context (interpretation and observation of situation) to make a decision that is optimal with respect to certain criteria (here a scoring function and often external considerations as well). It is easy to see by this definition why AI systems might be useful for this, as they naturally take input data or past observations (what is more generally referred to as training) and use the latter to maximize a certain scoring function. These three key aspects indeed characterize an effective decision, but there is more to decision-making than the mere decision. A decision usually involves consensus between many individuals, humans for example, and a key requirement for a decision to be executed is trust towards the decision-making agent. As an example, we can refer to a doctor diagnosing a patient. In this setting, the patient trusts the doctor because she/he has seen similar situations (past experiences), and is educated on the matter to interpret your case (interpretation and observation), which makes the doctor's decision effective. However, the doctor often explains the diagnosis and where it comes from in terms that the patient understands in order to gain his/her trust. This then allows for the doctor's effective decision to be executed, as this creates proximity and trust between the decision maker and the affected parties through transparency in the process and reasoning behind the decision. This is not the case with AI systems, as they are not able to provide an explanation as to why, weakening the trust between the decision-making agent and the affected parties. To challenge the opinion, trust, and cooperate with a potentially more intelligent system, we need an explanation behind its decision, not a *black box* that simply outputs a result, as good as that result might be. This is especially important because this

allows for debate, which is primordial to reach a consensus and for the affected parties to accept and trust the final decision (This idea is well explained by Irving et al. (2018) and gives ground to AI Safety measures). As an example, imagine an AI system identifies a patient as terminally ill, and there is a pill that can cure him but has a 50% risk of killing him/her, and the doctor sees nothing wrong with the patient. One could think that we must trust the AI system as it is surely more capable than the doctor, but it can also make mistakes. Seeing as though the AI system will not give any explanation as to why it is giving this diagnosis, it is unlikely that the patient will follow the AI system's treatment route. It is therefore clear that in order to cooperate with an AI system, we need some clarity into what goes on behind the scenes.

7.3 The solution to (almost) all of our problems

Until now, we discussed the importance of interpretability in situations where the AI system is correctly aligned, and the problems that arise come from our requirements as humans to make decisions. However, interpretability also has a key role in aligning AI systems, especially in detecting misalignment and avoiding deception by more capable agents. Most of the dangerous situations come from the ability of the AI system to plan, specifically a plan that involves deception and pursuing a misaligned goal. Nevertheless, if we had access to said plan we could easily detect misalignment and deception and correctly deem whether an agent is safe to deploy or not. This is not an alignment measure, as the system can still be misaligned, but the ability to detect misalignment allows us to significantly reduce the negative consequences that stem from it. Many efforts look to achieve this, namely Burns et al. (2022) in the case of Large Language Models (LLMs). Interpretability is therefore key as it not only allows for better cooperation and integration of AI-based decision-making but also puts us in front of the misalignment problem by being able to detect it early enough to avoid negative consequences.

8. Conclusion

AI opens up an exciting realm of possibilities, as it allows us to go well beyond what we can do, all the way to the things we can score or grade. However, we have seen that this comes with different challenges regarding aligning the systems itself as well as our interaction with said systems. This article hopes to inform the reader of the challenges that AI systems pose and hopefully encourage more research in this direction.

Acknowledgments

I would like to thank Eva Vidal, Ferran Marquès, Climent Nadeu and Jordi Domingo for supporting me in writing this article and in spreading knowledge about AI Safety. Their help through contributions, references, and conversations has been invaluable. I would also like to thank Alex Serrano and Joel Solé for their feedback during the writing process and Liza Turquie for her support.

References

Terminator. Metro Goldwyn Mayer, 1984.

Statement on AI risk. *CAIS*, 2022.

Nick Bostrom. The superintelligent will: Motivation and instrumental rationality in advanced artificial agents. *Minds and Machines*, 22(2):71–85, May 2012. doi: 10.1007/s11023-012-9281-3.

Collin Burns, Haotian Ye, Dan Klein, and Jacob Steinhardt. Discovering latent knowledge in language models without supervision, 2022. URL https://arxiv.org/abs/2212.03827.

Collin Burns, Pavel Izmailov, Jan Hendrik Kirchner, Bowen Baker, Leo Gao, Leopold Aschenbrenner, Yining Chen, Adrien Ecoffet, Manas Joglekar, Jan Leike, Ilya Sutskever, and Jeff Wu. Weak-to-strong generalization: Eliciting strong capabilities with weak supervision, 2023. URL https://arxiv.org/abs/2312.09390.

Ajeya Cotra. Without specific countermeasures, the easiest path to transformative AI likely leads to AI takeover, 2022. URL https://www.alignmentforum.org/posts/pRkFkzwKZ2zfa3R6H/without-specific-countermeasures-the-easiest-path-to.

Thilo Hagendorff. Deception abilities emerged in large language models, 2023. URL https://arxiv.org/abs/2307.16513.

Geoffrey Irving, Paul Christiano, and Dario Amodei. AI safety via debate, 2018. URL https://arxiv.org/abs/1805.00899v2.

Richard Ngo, Lawrence Chan, and Sören Mindermann. The alignment problem from a deep learning perspective, Sep 2023. URL https://arxiv.org/abs/2209.00626.

Darsh Parekh, Nishi Poddar, Aakash Rajpurkar, Manisha Chahal, Neeraj Kumar, Gyanendra Prasad Joshi, and Woong Cho. A review on autonomous vehicles: Progress, methods and challenges. *Electronics*, 11:2162, 07 2022. doi: 10.3390/electronics11142162.

David Silver, Thomas Hubert, Julian Schrittwieser, Ioannis Antonoglou, Matthew Lai, Arthur Guez, Marc Lanctot, Laurent Sifre, Dharshan Kumaran, Thore Graepel, Timothy Lillicrap, Karen Simonyan, and Demis Hassabis. Mastering chess and shogi by self-play with a general reinforcement learning algorithm, 2017. URL https://arxiv.org/abs/1712.01815.

Joar Skalse, Nikolaus H. R. Howe, Dmitrii Krasheninnikov, and David Krueger. Defining and characterizing reward hacking, Sep 2022. URL https://arxiv.org/abs/2209.13085.

John Thornhill. AI will never threaten humans, says top meta scientist. *Financial Times*, 2023.

Alexander Matt Turner. Optimal farsighted agents tend to seek power. *CoRR*, abs/1912.01683, 2019. URL http://arxiv.org/abs/1912.01683.

Claus O. Wilke, Jia Lan Wang, Charles Ofria, Richard E. Lenski, and Christoph Adami. Evolution of digital organisms at high mutation rates leads to survival of the flattest, 2001. URL https://www.nature.com/articles/35085569.