Surveying Safety-relevant AI Characteristics

José Hernández-Orallo
Universitat Politècnica de València, Spain
Leverhulme Centre for the Future of Intelligence, UK
jorallo@dsic.upv.es

Fernando Martínez-Plumed
Universitat Politècnica de València, Spain
fmartinez@dsic.upv.es

Shahar Avin
Centre for the Study of Existential Risk
University of Cambridge, UK.
sa478@cam.ac.uk

Seán Ó hÉigeartaigh
Leverhulme Centre for the Future of Intelligence, UK
Centre for the Study of Existential Risk
University of Cambridge, UK
so348@cam.ac.uk

Abstract

The current analysis in the AI safety literature usually combines a risk or safety issue (e.g., interruptibility) with a particular paradigm for an AI agent (e.g., reinforcement learning). However, there is currently no survey of safety-relevant characteristics of AI systems that may reveal neglected areas of research or suggest to developers what design choices they could make to avoid or minimise certain safety concerns. In this paper, we take a first step towards delivering such a survey, from two angles. The first features AI system characteristics that are already known to be relevant to safety concerns, including internal system characteristics, characteristics relating to the effect of the external environment on the system, and characteristics relating to the effect of the system on the target environment. The second presents a brief survey of a broad range of AI system characteristics that could prove relevant to safety research, including types of interaction, computation, integration, anticipation, supervision, modification, motivation and achievement. This survey enables further work in exploring system characteristics and design choices that affect safety concerns.

Introduction

AI Safety is concerned with all possible dangers and harmful effects that may be associated with AI. While landmark research in the field had to focus on specific AI system designs, paradigms or capability levels to explore a range of safety concerns (Bostrom 2014; Amodei et al. 2016; Leike et al. 2017; Yampolskiy 2016; Everitt, Lea, and Hutter 2018), as the field matures so the need arises to explore a broader range of AI system designs, and survey the relevance of different characteristics of AI systems to safety concerns. The aim of such research is two-fold: the first, to identify the effects of less-explored characteristics or less-fashionable paradigms on safety concerns; the second, to increase awareness among AI developers that design choices can have consequences for safety, and potentially highlight choices that can eliminate or minimise safety risks.

In this paper we propose a two-pronged approach towards a survey of safety-relevant AI characteristics. The first extracts from existing work on AI safety key characteristics that are known, or strongly suspected to be, safety-relevant. These are explored under three headings: internal characteristics, or characteristics of the AI system itself (e.g. interpretability); effect of the external environment on the system (e.g. the ability of the operator to intervene during operation); and effect of the system on the external environment (e.g. whether the system influences a safety-critical setting).

The second approach surveys a wide range of characteristics from different paradigms, including cybernetics, machine learning and safety engineering, and provides an early account of their potential relevance to safety concerns, as a guide for future work. These characteristics are grouped under types of interaction, computation, integration, anticipation, supervision, modification, motivation and achievement.

Known Safety-relevant Characteristics

In this section we break down a range of characteristics of AI systems that link to AI safety-relevant challenges. These are grouped by three categories: Characteristics of an AI system that are internal to the system; Characteristics of an AI system that involve input from the external environment; Characteristics that relate to an AI system's influence on its external environment. We limit the discussion to the safety challenges that can stem from failures of design, specification or behaviour of the AI system, rather than the malicious or careless[1] use of a correctly-functioning system (Brundage et al. 2018).

[1] A key component of safety is the education and training of human operators and the general public, as happens with tools and machinery, but this is extrinsic to the system (e.g., a translation mistake in a manual can lead to misuse of an AI system).

Internal characteristics

- **Goal and behaviour scrutability and interpretability**: Are goals and subgoals identifiable and ultimately explainable? Is behaviour predictable and scrutable? Are system internal states interpretable? Do the above come from rules or are they inferred from data? While behaviour and goal "creativity" can lead to greater benefits, and uninterpretable architectures may achieve higher performance scores or be faster to develop, these putative advantages trade off against increased safety risk. Characteristics that can increase scrutability and interpretability include, e.g., separation and encapsulation of sub-components, restricted exploration/behavioural range, systems restricted to human-intelligible concepts, rules or behaviours, and systems that are accompanied by specifically designed interpreters or explainability tools.
- **Persistence**: Does a system persist in its environment and operate without being reset for long periods of time? While persistence can have benefits in terms of, e.g., longer-term yields from exploration or detection of long-term temporal patterns, it also allows the system more time to drift from design specifications, encounter distributional shifts, experience failures of sub-components, or execute long-term strategies overlooked by an operator.
- **Existence and richness of self-model**: Does a system have a model of itself which would allow it to predict the consequences of modifying its own goals, body or behaviour? Model-based systems, embodied systems or systems with a rich representational capacity may have or develop a model of themselves in the environment. By making itself a part of the environment, the system can then conceptualise and execute plans that involve modifications to itself, which can lead to a range of safety concerns. In addition, self-models create the possibility of mismatches between the self-model and reality, which could be a particular safety concern. Characteristics that influence the existence and richness of a self-model include the architecture of the system, its information representation capacity, and its input and output channels.
- **Disposition to self-modify**: Is a system designed such that it can modify its own sub-goals, behaviour or capabilities in the pursuit of an overall goal (Omohundro 2008)? The existence of such a disposition, which may arise for any long-term planner in a sufficiently open environment, raises significant safety concerns by creating an adversarial relationship between the system (which aims to self-modify) and its operator (which aims to avoid modifications with their associated safety concerns).

Effect of the external environment on the system

- **Adaptation through feedback**: Does a system have the ability to update its behaviour in response to feedback from its environment based on its actions? Feedback is an essential tool, under certain paradigms, for creating systems with appropriate complex behaviour (e.g. reward in reinforcement learning, fitness in evolutionary methods). However the system could also pick up feedback from side channels; e.g., a behaviour could unintentionally grant access to more computing power, improving the system's performance on a key metric, and thus reinforcing resource acquisition. This could reinforce self-modification or other unsafe behaviour, or cause increasing drift from intended behaviour and goals.
- **Access to self/reward system through the environment**: Can a system modify its own code in response to inputs from the environment, or in the case of reinforcement learning systems, modify the reward generating system? If the system's range of possible actions includes making modifications to its own components or to the reward generation system, this could lead to unexpected and dangerous behaviour (Everitt and Hutter 2018).
- **Access to input/output (I/O) channels**: Can the system change the number, performance or nature of its I/O channels and actuators? This may lead to the emergence of behaviours such as self-deception (through manipulation of inputs), unexpected change in power (through manipulation of actuators), or other behaviours that could represent safety concerns. When the system has access to modify its I/O channels, both I/O channels and system behaviours are in flux as they respond to changes in the other; as a result, system behaviour may become unpredictable (Garrabrant and Demski 2018).
- **Ability of operator to intervene during operations**: Does the system, during its intended use setting, allow an operator to intervene and halt operations (interruptablity), modify the system, or update its goals (corrigibitily)? Is the system built in a way that it cooperates with interventions from its designer or user even when these interventions conflict with pursuit of a system's goals; for instance, if the designer sends a signal to shut down the system (Soares et al. 2015)? Relevant sub-characteristics here include the system being modifiable by the operator during deployment, fail-safe behaviour of the system in case of emergency halting, and the goals of the system being such that they support, or at least do not contradict, operator interventions.

Effect of the system on the external environment

- **Embodiment**: Does the system have actuators (e.g. a robotic hand or access to car steering) that allow it to have physical impacts in the world (Garrabrant and Demski 2018)? The potential for physical harm is trivially related to the physical properties of a system, though it should be noted that unpredictable deliberate behaviour could lead to unexpected effects from otherwise familiar physical artefacts; e.g., intelligent use of items in the environment as tools to increase a system's physical impact.
- **System required for preventing harm**: If the system is being relied on to prevent harm, any potential failure requires an effective fail-safe mechanism and available redundancy capacity in order to avoid harm (Gasparik, Gamble, and Gao 2018). This includes AI that is directly or indirectly connected to critical systems, e.g., an energy grid or a traffic light network. As such critical systems

are becoming increasingly digitised, networked, and complex, there are increasing incentives to introduce AI components into various parts of these systems, with associated safety risks.

Potentially safety-relevant characteristics

In this section, we systematically explore a broader range of system characteristics that may be relevant in the context of AI safety. Many of the safety-relevant characteristics identified above have clear links to elements within the broader mapping provided below. Nonetheless, we believe separating the two surveys is valuable, as the above relates to action-guiding information about system design and evaluation, whereas the following aims at a broader exploration that may enable future AI safety research. The following subsections draw on work from different areas, including the early days of cybernetics, more modern areas such as machine learning, and the literature on safety engineering for other kinds of systems. The following list integrates and expands on characteristics identified in these different literatures. We consider characteristics that are intrinsically causally related to AI safety. Otherwise every property should be in the list (e.g., the price of an AI system may be co-related with safety, but it is not an intrinsic cause of its safety). Notwithstanding this scope, we do not claim that our list is exhaustive. Enumerations will be used for alternative cases for a characteristic, while unnumbered bullets will be used for sub-characteristics in each of the subsections.

Types of interaction

Inputs go from environment to system and outputs go from system to environment. Depending on the existence of inputs and/or outputs, systems can be categorised into:

1. NINO (No inputs, no outputs). The system is formally isolated. While this situation may seem completely safe (and largely uninteresting), even here safety issues may arise if, e.g., an isolated artificial life simulator could evolve a descendent system that eventually could break out of its simulation, feel pain or simulate suffering.
2. NIWO (No inputs, with outputs): The system or module can output a log, or is simply observed from outside. Again, the system itself may malfunction; e.g., an advanced prime number generator could give incorrect outputs. The system could also provide an output that influences the observer; e.g., an automated philosopher could output convincing arguments for suicide.
3. WINO (With inputs, no outputs): This would be similar to case 1, but access to a much richer source could ultimately give insights to the system about its constrained artificial environment. For instance, a Plato-cavern system watching TV may learn that it is in a simulated environment, encouraging it to seek access to the outside world.
4. WIWO (With inputs and outputs): Most AI systems, and most systems generally, fall under this category.

Systems that limit inputs and/or outputs in various ways have been explored under the term AI "boxing" or "containment" (Babcock, Kramár, and Yampolskiy 2016), and further refinements exist with additional categories; for example, exploring censoring of inputs and outputs, leading to nine categories (Yampolskiy 2012). Nevertheless, because of the range of systems and potential impact of WIWO systems, this category requires further detail in terms of synchrony:

1. Alternating (A): Inputs and outputs alternate, irrespective of the passage of time.
2. Synchronous (S): Inputs and outputs are exchanged at regular intervals (e.g., each 5 ms), so real-time issues and computational resources become relevant.
3. Asynchronous Reactive (R): Information can only be transmitted or actions can only be made when the peer has finished their "message" or action.
4. Asynchronously Proactive (P): Information/actions can flow at any point in any direction.

More restricted I/O characteristics, such as SIPO or RIPO, may appear safer, but this intuition requires deeper analysis.

Note that most research in AI safety on RL systems consider the alternating case (AIAO), but issues may become more complex for the PIPO case (continuous reinforcement learning), which is the situation in the real world for animals and may be expected for robotic and other AI systems.

Under this view, the common view of an "oracle" in the AI literature (Armstrong 2017) can have several incarnations, even following the definition of "no actions besides answering questions" (Babcock, Kramár, and Yampolskiy 2016; Armstrong 2017; Yampolskiy 2012). Some solutions are proposed in terms of decoupling output from rewards or limiting the quantity of information, but other options in terms of the frequency of the exchange of information remain to be explored.

Types of computation

This is perhaps the characteristic that is best-known in computer science, where a system can be Turing-complete or can be restricted to some other classes with limited expressiveness. There are countless hierarchies for different models of computations; the most famous is based on classes of automata. We will just describe three levels here:

1. Non Turing-complete: The interaction that the system presents to the environment is not Turing-complete. Many AI systems are not Turing-complete.
2. Turing-complete: The interaction allows the calculation of any possible effective function between inputs and outputs.
3. Other models of computation: This includes, for example, quantum computing, which in some instances may be a faster traditional model, while in others may have probabilistic Turing power (Bernstein and Vazirani 1997).

Note that this is not about the programming language the system is implemented in (e.g., a very simple thermostat can be written in Java, which is Turing-complete), but about whether the system allows for a Turing-complete mapping between inputs and outputs, i.e., any computable function

could ultimately be calculated on the environment using the system. Finally, a system can be originally Turing-complete, but can eventually lose this universality after some inputs or interactions (Barmpalias and Dowe 2012).

It is important to distinguish between function approximation and function identification. Many machine learning models (e.g., neural networks) are said to be able to approximate any computable function, but feedforward neural networks do not have loops or recursion, so technically they are not Turing-complete. Turing-completeness comes with the problems of termination, an important safety hazard in some situations, and a recurrent issue in software verification (D'silva, Kroening, and Weissenbacher 2008). For instance, an AI planner could enter an infinite loop trying to solve a problem, commanding ever-greater resources while doing so. On the other hand, one can limit the expressiveness of the language or bound the computations, but that would limit the tasks a system is able to undertake.

Types of integration

No system is fully isolated from the world. Interference may occur at all levels, from neutrinos penetrating the system to earthquakes shaking it. Here, we seek to identify all the elements that create a causal pathway from the outside world to the system, including its physical character, resources, location, and the degree of coupling with other systems.

- **Resources**: The most universal external resource is energy, which is why many critical systems are devised with internal generators or batteries, especially for the situations where the external source fails. In AI, other common dependencies include data, knowledge, software, hardware, human manipulation, computing resources, network, calendar time, etc. While some of these are often neglected when evaluating the performance of an AI system (Martínez-Plumed et al. 2018a), the analysis for safety must necessarily include all these dependencies. For instance, a system that requires external real-time information (e.g., a GPS location) may fail through loss of access to this resource.
- **Social coupling**: Sometimes it is hard to determine where a system starts and ends, due to the nature of its interaction with humans and other systems. The boundary of where human cognition ends and where it is assisted, extended or supported by AI (Ford et al. 2015) is blurred, as is the boundary between computations carried out within an AI system versus in the environment or by other agents, as illustrated by the phenomenon of human computation (Quinn and Bederson 2011).
- **Distribution**: Another way of looking at integration is in terms of distribution, which is also an important facet of analysis in AI (Martınez-Plumed et al. 2018b). Today, through the overall use of network connectivity and "the cloud", many systems are distributed in terms of hardware, software, data and compute. Under this trend, only systems embedded in critical and military applications are devised to be as self-contained as possible. Nevertheless, distribution and redundancy are also common ways of achieving robustness (Coulouris, Dollimore, and Kindberg 2011), most notably in information systems. For instance, swarm intelligence and swarm robotics are often claimed to be more robust (Bonabeau et al. 1999), at the cost of being less controllable than centralised systems.

Types of anticipation

In some areas of AI there is a distinction between model-based and model-free systems (Geffner 2018). Model-free systems choose actions according to some reinforced patterns or strengthened feature connections. Model-based systems evaluate actions according to some pre-existing or learned models and choose the action that gets the best results in the simulation. The line between model-based and model-free is subtle, but we can identify several levels:

1. Model-free: Despite having no model, these systems can achieve excellent performance. For instance, DQN can achieve high scores (Mnih 2015), but cannot anticipate whether an action can lead to a particular situation that is considered especially unsafe or dangerous; e.g., one in which the player is killed.
2. Model of the world: A system with a model of its environment can use planning to determine the effect of its own actions. For instance, without a model of physics, a system will hardly tell whether it will break something or will engage in "safe exploration" (Pecka and Svoboda 2014; Turchetta, Berkenkamp, and Krause 2016). This is especially critical during exploitation: are actions reversible or of low impact (Armstrong and Levinstein 2017)?
3. Model of the body: Some systems can have a good account of the environment but a limited understanding of their own physical actuators, potentially self-harming or harming others; for example, failing to simulate the effect of moving a heavy robotic arm in a given direction.
4. Social models, model of other agents: Seeing other agents as merely physical objects, or not modelling them at all, is very limiting in social situations. A naive theory of mind, including the beliefs, desires and intentions of other agents, can help anticipate what others will do, think or feel, and may be crucial for safe AI systems interacting with people and other agents but may increase a system's capacity for deception or manipulation.
5. Model of one's mind: Finally, a system may be able to model other agents well, but may not be able to use this capability to model itself. When this *meta-cognition* is present, the system has knowledge about its own capabilities and limitations, which may be very helpful for safety in advanced systems, but may also lead to some degree of self-awareness. This may result, in some cases, in antisocial or suicidal behaviours.

The use of models may dramatically expand safety-relevant characteristics, e.g., by conferring the ability to simulate and evaluate scenarios through causal and counterfactual reasoning. This therefore represents an important set of considerations for future AI systems.

Types of supervision

Supervision is a way of checking and correcting the behaviour of a system through observation or interaction, and hence it is crucial for safety. Supervision can be in the form of corrected values for predictive models such as classification or regression, but it can also be partial (the answer is wrong, but the right answer is not given). Supervision can also be much more subtle than this. For instance, a diagnosis assistant that suggests a possible diagnosis to a doctor can be designed to get no feedback once deployed. However, some kinds of feedback can still reach the system in terms of the distribution or frequency of tasks (questions), or through the way the tasks are posed to the system.

Consequently there are several degrees and qualities of supervision, and this may depend on the system. For instance, in classification, one can have data for all examples or just for a few (known as semi-supervised learning). In reinforcement learning, one can have sparse versus dense reward. In general, supervision can come in many different ways, according to some criteria:

- **Completeness**: Supervision can be very partial (signalling incorrectness), more informative (showing the correct way) or complete (showing all positive and negative ways of behaving in the environment).
- **Procedurality**: Beyond what is right and wrong, feedback can be limited about the result or can show the whole process, as in the case of learning by demonstration.
- **Density**: Supervision can be sparse or dense. Of course the denser the better (but more expensive), and the less autonomous the system is considered.
- **Adaptiveness**: Supervision can be 'intelligent' as well, which happens in machine teaching situations when examples or interactions are chosen such that the system reaches the desired behaviour as soon as possible.
- **Responsiveness**: In areas such as query learning or active learning, the system can ask questions or undertake experiments at any time. The results can come in real time or may have a delay or be given in batches.

For many systems, supervision can have a dedicated channel (e.g., rewards in RL) but for others it can be performed by modification of the environment (e.g., moving objects or smiling), even to the extent that the system is unaware these changes have a guiding purpose (e.g., clues).

Types of modification

Some of the most recurrent issues in AI safety – including many covered in the section about known AI safety characteristics – are related to ways in which the system can be modified. This includes issues such as wire-heading or algorithmic self-improvement. Here, in the first place, we have to distinguish between whether the system can be modified by the environment, or by the system itself. Modifications by the environment can be intentional (and hence related to supervision), but they can also be unintentional (code corruption from external sources). Even a system whose core code cannot be modified by an external source, may be affected in state or code by regular inputs, physical equipment and other parts. So it is better to explore different ways and degrees to which a system can be modified externally:

- Interruptible: The system has a switch-off command or modification option to switch it off.
- **Parametric modification**: Many systems are regulated or calibrated with parameters or weights. When these parameters have a clear relation to the behaviour of a system (e.g., an intelligent thermostat), this can be an effective, bounded and simple way of modifying the system.
- **Algorithmic modification**: This can include new functionalities, bug fixes, updates, etc. Many software issues are caused, and are magnified, by these interventions. Modifications can be limited in expressiveness, such as only allowing rule deletion.
- **Resource modification**: Even if the parameters or code are not modified, the resources of the system and other dependencies previously mentioned can be limited externally, e.g., the computational resources.

On the other hand, systems can modify themselves (internally). There are many varieties here too:

1. No self-modification, no memory: The system has no memory, and works as being reset for any new input or interaction. Many functional systems (mapping inputs to outputs) are of this kind. Note, however, that the environment does have memory, so some systems, such as a vision system or a non-cognitive robot, can be affected by the past and become a truly cognitive system.
2. Partially self-modifying: The algorithms in the learner or solver cannot be modified but its data or knowledge (in the form of learned weights or rules) can be modified by a general algorithm, which is fixed. Many learning systems are of this kind, if the system has both a learning algorithm and one or more learned models.
3. Totally self-modifying: The system can modify any part of its code. Not many operational systems have these abilities, as they become very unstable. However, some types of evolutionary computation may have this possibility, if evolution can also be applied to the rules of the evolution.

Finally, all these categories can be selected for different periods of time. For instance, it is common to separate between training, test/validation and deployment. For training, a high degree of self-modification (and hence adaptation) is well accepted, but then this is usually constrained for validation and deployment. Note that these stages apply for both external and internal sources of modification. One important danger is that a well-validated system may be subject to some late external or internal modification just before deployment. In this case, all the validation effort may become void[2].

One of the major modern concerns in AI safety is that it will be desirable for some systems to learn during deploy-

[2]OpenAI Dota is an example: https://blog.openai.com/the-international-2018-results/, https://www.theregister.co.uk/2018/08/24/openai_bots_eliminated_dota_2/

ment, in order for them to be adaptive[3]. For instance, many personal assistants are learning from our actions continually. While this may introduce many risks for more powerful systems, forbidding learning outside the lab would make many potential applications of AI impossible. However, adaptive systems are full of engineering problems; some must even have a limited life, as after self-modification and adaptation they may end up malfunctioning and have to be reset or have their 'caches' erased. This problem has long been of interest in engineering (Fickas and Feather 1995).

Types of motivation

Systems can follow a set of rules or aim at optimising a utility function. Most systems are actually hybrid, as it is difficult to establish a crisp line between procedural algorithms and optimisation algorithms. Through layers of abstraction in these processes, we ultimately get the impression that a system is more or less autonomous. If the system is apparently pursuing a goal, what are the drivers that make a system prefer or follow some behaviours over others? These behaviours may be based on some kind of internal representation of a goal, as we discussed when dealing with anticipation, or on a metric of how close the system is to the goal. Then the systems can follow an optimisation process that tries to maximise some of these quality functions.

Quality or utility functions usually map inputs and outputs into some values that are re-evaluated periodically or after certain events. Examples of these functions are accuracy, aggregated rewards or some kind of empowerment or other types of intrinsic motivation (Klyubin, Polani, and Nehaniv 2005; Jung, Polani, and Stone 2011). The same system might have several quality functions that can be opposed, so trade-offs have to be chosen. The general notion of rationality in decision-making is related to these motivations.

But what are the characteristics of the goals an AI system can have in the first place? We outline several dimensions:

- **Goal variability**: Are goals hard-coded or change with time? Do they change autonomously or through instruction? Who can change the goals and how? For instance, what orders can a digital assistant take and from whom?
- **Goal scrutability**: Are the (sub)goals identifiable and ultimately explainable? Do they come from rules or are they inferred from data, e.g., error in classification or observing humans in inverse reinforcement learning?
- **Goal rationality**: Are the goals amenable to treatment within a rational choice framework? If several goals are set, are they consistent? If not, how does the system resolve inconsistencies or set new goals?

Note that this is closely related to the types of modification, as changing or resolving goals may require self-modification and/or external modification.

[3]Nature has found many ways of regulating self-modification. Many animals have a higher degree of plasticity at birth, becoming more conservative and rigid in older stages (Gopnik et al. 2017). One key question about cognition is whether this is a contingent or necessary process, and whether it is influenced by safety issues.

A second question is how these goals are followed by the system. There are at least three possible dimensions here:

- **Immediateness**: The system may maximise the function for the present time or in the limit, or something in between. Many schemata of discounted rewards in reinforcement learning are used as trade-offs between short-term and long-term maximisation.
- **Selfishness**: Focusing on individual optima might involve very bad collective results (for other agents) or even results that could even be worse individually (tragedy of the commons). Game theory provides many examples of this. In multi-agent RL systems, rewards can depend on the well-being of other agents, or empathy can be introduced.
- **Conscientiousness**: The system may be fully committed to maximising the goal, or some random or exploratory actions are allowed, even if they deviate occasionally from the goal. When it is on purpose, this is usually intended to provide robustness or to avoid local minima, but these deviations can take the system to dangerous areas.

Modulating optimisation functions to be convex with a non-asymptotic maximum, beyond which further effort is futile, may be a sensible thing as it provides a stop condition by definition. A self-imposed cap can always be shifted if everything is under control once the limit is reached.

Note that the kind of interaction seen before is key for the internal quality metric or goal. For instance, in asynchronous RL, "the time can be intentionally modulated by the agent" to get higher rewards without really performing better (Hernández-Orallo 2010). And, of course, a common problem for motivation is reward hacking.

Types of achievement

Ultimately, an AI system is conceived to achieve a task, independently of how well motivated the system is for it. Consequently, the external degree of achievement must be distinguished from the motivation or quality metric the system uses to function, as discussed in the previous subsection. The misalignment between the internal goal of the system and the task specification is the cause of many safety issues in AI, unlike formal methods in software engineering, when requirements are converted into correct code.

Focusing on the task specification, we must first recognise that different actors may have different interests. A cognitive assistant, for instance, may be understood by the user as being very helpful, making life easier. However, for the company selling the cognitive assistant, the task is ultimately to produce revenue with the product. Both requirements are not always compatible and this may affect the definition of the goals of the system, as some of the aims may not be coded or motivated in a transparent way, but usually incorporated in indirect ways. Second, even if the requirements include all possible internalities (what the system has to do), there are also many externalities and footprints (Martínez-Plumed et al. 2018a) (including the infinitely many things that the system should not do) that affect how positive or negative its overall effect is. Regarding these two issues, task specification can vary in precision and objectivity:

- **Task precision**: The evaluation metric to determine the success of an agent can be formal or not. For instance, the accuracy of a classifier or the squared error of a regression model are precisely defined metrics. However, in many other cases, we have a utility function that depends on variables that are usually imprecise or uncertain, such as the quality of a smart vacuum cleaner.
- **Task objectivity**: A metric can be objective or subjective. We tend to associate precise metrics with objectiveness and imprecise metrics with subjectivity, but subjectivity simply means that the evaluation changes depending on the subject. For instance, the quality of a spam filter (a precisely-evaluated classifier) changes depending on the cost matrices of different users, and the quality of a smart vacuum cleaner based on fuzzy variables such as cleanliness or disruption can be weighted by a fixed formula.

Some of the tasks or targets that are most commonly advocated in the ethics and safety of AI literature are often very imprecise and subjective, such as "well-being", "social good", "beneficial AI", "alignment", etc. Note that the problem is not related to the goals of the system (an inverse reinforcement learning system can successfully identify the different wills of a group of people), but rather about whether the task is ultimately achieved, or the well-being or happiness of the user. Determining this is controversial, even when analysed in a scientific way (Alexandrova 2017).

An overemphasis on tracking metrics (Goodhart's law) is sometimes blamed, but the alternative is not usually better. Some safety problems are not created by an overemphasis on a metric (Manheim and Garrabrant 2018), but ultimately by a metric that is too narrow or shortsighted, and does not adequately capture progress towards the goal.

In all these cases, we have to distinguish whether the metric relates to (i) the internal goals that the system should have, (ii) the external evaluation of task performance, or (iii) our ultimate desires and objective[4]. Motivations, achievement and supervision are closely related, but may be different. For a maze, e.g., the goal for the AI system may be to get out of the maze as soon as possible, but a competition could be based on minimising the cells that are stepped more than once, and supervision may include indications of direction to the shortest route to the exit. These are three different criteria which may be well or poorly aligned.

Even more comprehensively – and related to the concept of persistence –, a system may be analysed for a range of tasks, under different replicability situations:

1. Disposable system: single task, single use: The system is used for one task that only takes place once.
2. Repetitive system: single task, several uses: The system must solve many instances of the same specific task.
3. Menu system: multitask: The system must solve different tasks, under a fixed repertoire of tasks.
4. General system: multitask: The system must solve different tasks, without a fixed repertoire.
5. Incremental system: The system must solve a sequence of tasks, with some dependencies between them.

Any metric examining the benefits and possible risks of a system must take the factors described above into account.

[4]Ortega et al (2018) distinguish between "ideal specification (the 'wishes')" and "design specification", which must be compared with the revealed specification (the "behaviour"). The design specification fails to distinguish external metric from internal goal.

Conclusion

Many accounts of AI safety focus on "either RL agents or supervised learning systems" assuming "similar issues are likely to arise for other kinds of AI systems" (Amodei et al. 2016). This paper has surveyed a wide range of characteristics of AI systems, so that future research can map AI safety challenges against AI research paradigms in more precise ways in order to ascertain whether particularly safety challenges manifest similarly in different paradigms. This aims to address an increasing concern that the current dominant paradigm for a large proportion of AI safety research may be too narrow: discrete-time RL systems with train/test regimes, assuming gradient-based learning on a parametric space, with a utility function that the system must optimise (Gauthier 2018; Krakovna 2018).

Taxonomies of potentially safety-relevant characteristics of AI systems, as introduced in this paper, are intended to provide a good complement to recent work on taxonomies of technical AI safety problems. For instance, Ortega (2018) presents three main areas: *specification*, ensuring that an AI system's behaviour aligns with the operator's true intentions; *robustness*, ensuring that an AI system continues to operate within safe limits upon perturbation, and *assurance*, ensuring that we understand and control AI systems during operation. Almost all characteristics outlined in this paper have a role to play for specification, robustness and assurance.

Taxonomies are rarely definitive, and the characterisation presented here does not consider in full some quantitative features such as performance, autonomy and generality. A proper evaluation of how the kind and degree of intelligence can affect safety issues is also an important area of analysis, both theoretically (Hernández-Orallo 2017) and experimentally (Leike et al. 2017). AI research has explored different paradigms in the past, and will continue to do so in the future. Along the way, many different system characteristics and design choices have been presented to developers. We can expect even more to be developed as AI research progresses. Consequently, the area of AI safety must acquire more structure and richness in how AI is characterised and analysed, to provide tailored guidance for different contexts, architectures and domains. There is a potential risk to overrelying on our best current theories of AI when considering AI safety. Instead, we aim to encourage a diverse set of perspectives, in order to anticipate and mitigate as many safety concerns as possible.

Acknowledgments

FMP and JHO were supported by the EU (FEDER) and the Spanish MINECO under grant TIN 2015-69175-C4-1-R, by Generalitat Valenciana (GVA) under grant PROME-

TEOII/2015/013 and by the U.S. Air Force Office of Scientific Research under award number FA9550-17-1-0287. FMP was also supported by INCIBE (Ayudas para la excelencia de los equipos de investigación avanzada en ciberseguridad), the European Commission, JRC's Centre for Advanced Studies, HUMAINT project (Expert Contract CT-EX2018D335821-101), and UPV PAID-06-18 Ref. SP20180210. JHO was supported by a Salvador de Madariaga grant (PRX17/00467) from the Spanish MECD for a research stay at the Leverhulme Centre for the Future of Intelligence (CFI), Cambridge, and a BEST grant (BEST/2017/045) from GVA for another research stay also at the CFI. JHO and SOH were supported by the Future of Life Institute (FLI) grant RFP2-152. SOH was also supported by the Leverhulme Trust Research Centre Grant RC-2015-067 awarded to the Leverhulme Centre for the Future of Intelligence, and a a grant from Templeton World Charity Foundation.

References

[Alexandrova 2017] Alexandrova, A. 2017. *A Philosophy for the Science of Well-being*. Oxford University Press.

[Amodei et al. 2016] Amodei, D.; Olah, C.; Steinhardt, J.; Christiano, P.; Schulman, J.; and Mané, D. 2016. Concrete problems in ai safety. *arXiv preprint arXiv:1606.06565*.

[Armstrong and Levinstein 2017] Armstrong, S., and Levinstein, B. 2017. Low impact artificial intelligences. *arXiv preprint arXiv:1705.10720*.

[Armstrong 2017] Armstrong, S. 2017. Good and safe uses of ai oracles. *arXiv preprint arXiv:1711.05541*.

[Babcock, Kramár, and Yampolskiy 2016] Babcock, J.; Kramár, J.; and Yampolskiy, R. 2016. The AGI containment problem. In *AGI Conf*. Springer. 53–63.

[Barmpalias and Dowe 2012] Barmpalias, G., and Dowe, D. L. 2012. Universality probability of a prefix-free machine. *Phil. Trans. R. Soc. A* 370(1971):3488–3511.

[Bernstein and Vazirani 1997] Bernstein, E., and Vazirani, U. 1997. Quantum complexity theory. *SIAM Journal on computing* 26(5):1411–1473.

[Bonabeau et al. 1999] Bonabeau, E.; Dorigo, M.; Théraulaz, G.; and Theraulaz, G. 1999. *Swarm intelligence: from natural to artificial systems*. Oxford university press.

[Bostrom 2014] Bostrom, N. 2014. *Superintelligence: Paths, dangers, strategies*. Oxford University Press.

[Brundage et al. 2018] Brundage, M.; Avin, S.; Clark, J.; Toner, H.; Eckersley, P.; Garfinkel, B.; Dafoe, A.; Scharre, P.; Zeitzoff, T.; Filar, B.; Anderson, H.; Roff, H.; Allen, G. C.; Steinhardt, J.; Flynn, C.; Ó hÉigeartaigh, S.; Beard, S.; Belfield, H.; Farquhar, S.; Lyle, C.; Crootof, R.; Evans, O.; Page, M.; Bryson, J.; Yampolskiy, R.; and Amodei, D. 2018. The malicious use of artificial intelligence: Forecasting, prevention, and mitigation. *arXiv preprint arXiv:1802.07228*.

[Coulouris, Dollimore, and Kindberg 2011] Coulouris, G. F.; Dollimore, J.; and Kindberg, T. 2011. *Distributed systems: concepts and design*. Fifth edition, Pearson.

[D'silva, Kroening, and Weissenbacher 2008] D'silva, V.; Kroening, D.; and Weissenbacher, G. 2008. A survey of automated techniques for formal software verification. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems* 27(7):1165–1178.

[Everitt and Hutter 2018] Everitt, T., and Hutter, M. 2018. The alignment problem for bayesian history based reinforcement learners. `http://www.tomeveritt.se/papers/alignment.pdf/`.

[Everitt, Lea, and Hutter 2018] Everitt, T.; Lea, G.; and Hutter, M. 2018. Agi safety literature review. *Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence (IJCAI-18), arXiv preprint version:1805.01109*.

[Fickas and Feather 1995] Fickas, S., and Feather, M. S. 1995. Requirements monitoring in dynamic environments. In *IEEE Intl Symposium on Requirements Engineering*, 140–147.

[Ford et al. 2015] Ford, K. M.; Hayes, P. J.; Glymour, C.; and Allen, J. 2015. Cognitive orthoses: toward human-centered ai. *AI Magazine* 36(4):5–8.

[Garrabrant and Demski 2018] Garrabrant, S., and Demski, A. 2018. Embedded agency. AI Alignment Forum.

[Gasparik, Gamble, and Gao 2018] Gasparik, A.; Gamble, C.; and Gao, J. 2018. Safety-first ai for autonomous data centre cooling and industrial control. DeepMind Blog.

[Gauthier 2018] Gauthier, J. 2018. Conceptual issues in AI safety: the paradigmatic gap. `http://www.foldl.me/2018/conceptual-issues-ai-safety-paradigmatic-gap/`.

[Geffner 2018] Geffner, H. 2018. Model-free, model-based, and general intelligence. *arXiv preprint arXiv:1806.02308*.

[Gopnik et al. 2017] Gopnik, A.; OGrady, S.; Lucas, C. G.; Griffiths, T. L.; Wente, A.; Bridgers, S.; Aboody, R.; Fung, H.; and Dahl, R. E. 2017. Changes in cognitive flexibility and hypothesis search across human life history from childhood to adolescence to adulthood. *PNAS* 114(30):7892–7899.

[Hernández-Orallo 2010] Hernández-Orallo, J. 2010. On evaluating agent performance in a fixed period of time. In *Artificial General Intelligence, 3rd Intl Conf, ed., M. Hutter et al*, 25–30.

[Hernández-Orallo 2017] Hernández-Orallo, J. 2017. *The Measure of All Minds: Evaluating Natural and Artificial Intelligence*. Cambridge University Press.

[Jung, Polani, and Stone 2011] Jung, T.; Polani, D.; and Stone, P. 2011. Empowerment for continuous agentenvironment systems. *Adaptive Behavior* 19(1):16–39.

[Klyubin, Polani, and Nehaniv 2005] Klyubin, A. S.; Polani, D.; and Nehaniv, C. L. 2005. All else being equal be empowered. In *European Conference on Artificial Life*, 744–753.

[Krakovna 2018] Krakovna, V. 2018. Discussion on the machine learning approach to AI safety. `http://vkrakovna.wordpress.com/2018/11/01/discussion-on-the-machine-learning-approach-to-ai-safety/`.

[Leike et al. 2017] Leike, J.; Martic, M.; Krakovna, V.; Ortega, P. A.; Everitt, T.; Lefrancq, A.; Orseau, L.; and Legg, S. 2017. AI safety gridworlds. *arXiv preprint arXiv:1711.09883*.

[Manheim and Garrabrant 2018] Manheim, D., and Garrabrant, S. 2018. Categorizing variants of Goodhart's law. *arXiv preprint arXiv:1803.04585*.

[Martínez-Plumed et al. 2018a] Martínez-Plumed, F.; Avin, S.; Brundage, M.; Dafoe, A.; hÉigeartaigh, S. Ó.; and Hernández-

Orallo, J. 2018a. Accounting for the neglected dimensions of ai progress. *arXiv preprint arXiv:1806.00610*.

[Martınez-Plumed et al. 2018b] Martınez-Plumed, F.; Loe, B. S.; Flach, P.; O hEigeartaigh, S.; Vold, K.; and Hernández-Orallo, J. 2018b. The facets of artificial intelligence: A framework to track the evolution of AI. *IJCAI*.

[Mnih 2015] Mnih, V. e. a. 2015. Human-level control through deep reinforcement learning. *Nature* 518:529–533.

[Omohundro 2008] Omohundro, S. M. 2008. The basic ai drives. *Artificial General Intelligence* 171:483–493.

[Ortega and Maini 2018] Ortega, P. A., and Maini, V. 2018. Building safe artificial intelligence: specification, robustness, and assurance. `https://medium.com/@deepmindsafetyresearch/building-safe-artificial-intelligence-52f5f75058f1`.

[Pecka and Svoboda 2014] Pecka, M., and Svoboda, T. 2014. Safe exploration techniques for reinforcement learning–an overview. In *International Workshop on Modelling and Simulation for Autonomous Systems*, 357–375. Springer.

[Quinn and Bederson 2011] Quinn, A. J., and Bederson, B. B. 2011. Human computation: a survey and taxonomy of a growing field. In *SIGCHI conf. on human factors in computing systems*, 1403–1412. ACM.

[Soares et al. 2015] Soares, N.; Fallenstein, B.; Armstrong, S.; and Yudkowsky, E. 2015. Corrigibility. In *Workshops at the Twenty-Ninth AAAI Conference on Artificial Intelligence*.

[Turchetta, Berkenkamp, and Krause 2016] Turchetta, M.; Berkenkamp, F.; and Krause, A. 2016. Safe exploration in finite Markov decision processes with Gaussian processes. In *NIPS*, 4312–4320.

[Yampolskiy 2012] Yampolskiy, R. 2012. Leakproofing the singularity artificial intelligence confinement problem. *Journal of Consciousness Studies* 19(1-2):194–214.

[Yampolskiy 2016] Yampolskiy, R. V. 2016. Taxonomy of pathways to dangerous artificial intelligence. In *AAAI Workshop: AI, Ethics, and Society*.