Minds and Machines
https://doi.org/10.1007/s11023-020-09549-0

GENERAL ARTICLE

Twenty Years Beyond the Turing Test: Moving Beyond the Human Judges Too

José Hernández-Orallo[1,2]

Received: 25 March 2020 / Accepted: 29 October 2020

Abstract

In the last 20 years the Turing test has been left further behind by new developments in artificial intelligence. At the same time, however, these developments have revived some key elements of the Turing test: imitation and adversarialness. On the one hand, many generative models, such as generative adversarial networks (GAN), build imitators under an adversarial setting that strongly resembles the Turing test (with the judge being a learnt discriminative model). The term "Turing learning" has been used for this kind of setting. On the other hand, AI benchmarks are suffering an adversarial situation too, with a 'challenge-solve-and-replace' evaluation dynamics whenever human performance is 'imitated'. The particular AI community rushes to replace the old benchmark by a more challenging benchmark, one for which human performance would still be beyond AI. These two phenomena related to the Turing test are sufficiently distinctive, important and general for a detailed analysis. This is the main goal of this paper. After recognising the abyss that appears beyond superhuman performance, we build on Turing learning to identify two different evaluation schemas: Turing testing and adversarial testing. We revisit some of the key questions surrounding the Turing test, such as 'understanding', commonsense reasoning and extracting meaning from the world, and explore how the new testing paradigms should work to unmask the limitations of current and future AI. Finally, we discuss how behavioural similarity metrics could be used to create taxonomies for artificial and natural intelligence. Both testing schemas should complete a transition in which humans should give way to machines—not only as references to be imitated but also as judges—when pursuing and measuring machine intelligence.

Keywords Turing test · Turing learning · Imitation · Adversarial models · Intelligence evaluation

✉ José Hernández-Orallo
jorallo@upv.es

[1] Universitat Politècnica de València, Valencia, Spain

[2] Leverhulme Centre for the Future of Intelligence, Cambridge, UK

Published online: 04 November 2020

1 Introduction

Twenty years ago, on the fiftieth anniversary of the introduction of the imitation game (Saygin et al. 2000), there seemed to be momentum and consensus to move beyond the Turing test (Hernández-Orallo 2000). It was high time, I argued, to look for intelligence tests that should be "non-Boolean, factorial, non-anthropomorphic, computational and meaningful". In these two decades, AI has changed significantly, and the Turing test is not part of the everyday vocabulary of AI researchers any more, not even as a future landmark (Marcus 2020). Rather, the notions of artificial general intelligence (AGI) and superintelligence have replaced the old wild dreams of AI, and are used as arguments exposing the limitations of a great majority of AI applications—fuelled by deep learning—that can still be considered very *narrow*.

Somewhat surprisingly, a particular type of "generative models", exemplified by generative adversarial networks (GAN), refine a generator by relying on a discriminative model, a judge telling between the true object and the generated one. Because of this analogy, this paradigm has been dubbed "Turing learning" (Li et al. 2016; Groß et al. 2017), and deserves a technical and philosophical analysis on its own. Relatedly, there is an adversarial situation in AI benchmarks, suffering a 'challenge-solve-and-replace' evaluation dynamics (Schlangen 2019). New benchmarks appear every year, but 'superhuman performance' is achieved very quickly. In many cases this performance is reached by using shortcuts or tricks, from obscure statistical properties in the data to plain cheating, a phenomenon that is usually referred to as the Clever Hans of AI (Sebeok and Rosenthal 1981; Sturm 2014; Hernández-Orallo 2019a). The discovery that benchmarks can be gamed prompts their replacement by more complex ones, hopefully capturing *that elusive challenging part of the task the system fails to understand* or is thought to be key to intelligent behaviour.

The very concept of 'superhuman performance' has similar grounds to the Turing test, and in this paper we dissect the problems that emerge when extrapolating beyond 'the human level': how can we evaluate real breakthroughs in AI and determine the paths to follow beyond human performance? Somewhat paradoxically, the solution to these problems goes through the Turing learning paradigm mentioned above. From here, two adversarial evaluation settings can be introduced: Turing testing (where imitation is kept) and Adversarial testing (where imitation is eliminated). In both cases the human judge turns into a machine, which improves its assessment performance as the evaluation progresses.

The rest of the paper is organised as follows. Section 2 summarises the reasons why the Turing test should have been left behind for AI evaluation, and what has changed in AI in the last 20 years. Section 3 discusses the problems of using humans as a reference when trying to extrapolate beyond them. Section 4 discusses Turing learning, and ways in which we should train the judges in competitions and benchmarks, through two settings: Turing testing and Adversarial testing. Section 5 addresses the evaluation of elusive capabilities related to

'thinking', such as understanding the world and extracting meaning from it, and whether this is possible if the machine judge in the testing setting lacks those capabilities. Section 6 converts 'equivalence' tests into 'similarity' tests leading to metrics that can be used to arrange and categorise behaviour (either natural or artificial) into taxonomies. Finally, Sect. 7 closes the paper with a discussion about the lessons learnt in the last 20 years and what needs to be done to really move beyond the Turing test once and for all.

2 The Turing Test: A Beacon or a Relic?

In the context of this paper, and 70 years after the imitation game was introduced, it is very appropriate to remember that Turing's original paper (1950) was meant to counteract nine arguments against the idea of intelligent machines. The term 'test' and the current interpretation of the game was only adopted after some interviews (e.g., Turing , 1952) and the huge amount of literature that flourished in the following decades. This kept a different debate alive, such as whether the imitation game could be a sufficient and necessary test for intelligence (Fostel 1993; Hayes and Ford 1995; Copeland 2000; French 2000; Proudfoot 2011). For the reader that is unfamiliar with this history, I suggest some insightful surveys (Moor 2003; Copeland and Proudfoot 2008; Oppy and Dowe 2011; Proudfoot 2017) or in better alignment with the rest of this paper, sections 5.3 and 5.4 in (Hernández-Orallo 2017b), covering the variants of the Turing test, some of their philosophical interpretations and their use as evaluation instruments for artificial intelligence.

For the purpose of this paper, it is just necessary to recall that the Turing test has three parties: player A (the imitator), player B (the authentic reference) and the judge, who must tell who the impostor is. In the original Turing's imitation game, the judge was a human, player A was a computer (pretending to be a woman) and player B was an actual woman. In the standard interpretation of the Turing test used today, gender is considered irrelevant; the judge is a human, player A is a computer pretending to be a human and player B is a human. Following this generalisation, some interesting variants have followed, as Table 1 summarises.

Going top-down in the table, Victorian parlour games, represented in the first row, challenged a human who should tell between a man and a woman through written notes. These games were played in Victorian times, and could have well inspired Turing to propose his imitation game. Note that we distinguish the original imitation game in the second row, as introduced in (Turing 1950), where gender still appears explicitly, and the Standard Turing test, in the third row, as was understood more commonly in subsequent years (Turing 1952). The Visual/Total Turing test (Zillich 2012; Borg et al. 2012) is a variant where the agents are embodied in a (simulated) world and can see each other. BotPrize was a competition taking place in the game Unreal Tournament (Hingston 2009), with the goal of creating an AI player that would be indistinguishable from a human by the human judges. A Turing test with compression is an idea first introduced in (Dowe and Hajek 1997, 1998), arguing that some compression problems should be included to show *understanding*. Matching pennies is a binary version of rock-paper-scissors that has been discussed

Table 1 Several variants of the Turing test

Variant	Judge	Player A	Player B	Interaction
Victorian parlour game	*H*	M → W	W → W	Written notes
Turing's imitation game	*H*	C → W	W → W	Textual teletype
Standard Turing test	*H*	C → H	H → H	Textual teletype
Visual/Total TT	*H*	C → H	H → H	Visual/embodied
BotPrize	*H*	C → H	H → H	Video game
TT with compression	*H*+size	C → H	H → H	Textual teletype
Matching pennies	–	$C_A \rightarrow C_B$	$C_B \rightarrow C_A$	Binary teletype
Inverted TT	*C*	C → H	H → H	Textual teletype
Reverse TT: CAPTCHA	*C*	C → H	–	Any

W, *M*, *H* and *C* represent general *woman*, *man*, *human* and *computer* respectively, with subindexes referring to particular individuals. The columns for player A and B represent the imitator and the authentic agent. The arrows represent "pretending to be". The final column indicates what kind of communication is allowed between the players and the judge. [Adapted from (Hernández-Orallo 2017b, Table 5.1).]

as an elementary intelligence test, or at least a prediction test (Hibbard 2008, 2011; Hernández-Orallo et al. 2012, 2012). The inverted Turing test is the first proposal of a test where the judge is a machine (Watt 1996), but quite surprisingly, it is the judge that is evaluated. If the judge can tell between machines and humans, then it is intelligent. This is very different from the reverse Turing test (von Ahn et al. 2004, 2008), with its implementations usually known as CAPTCHAs, where the judge is not evaluated, as in all other versions. The relevance of the reverse Turing test is that it is totally automated, as the human or machine to be detected does not have to be compared against a real human. Usually, the kind of exercises are quite trivial for humans, but challenging to state-of-the-art AI. We will especially discuss the appropriateness of human judges in the following sections.

It is now relevant to recall some of the reasons why the standard Turing test should have been left behind many decades ago. The first one is that the Turing test does not measure intelligence, but "humanity"' (Fostel 1993). The incarnations of the Turing test, such as the Loebner Prize, have raised little enthusiasm from the AI community. For instance, referring to a recent edition of the prize (Shah and Warwick 2015), Moshe Y. Vardi, editor-in-chief of the Communications of the ACM responded: "the details of this 2014 Turing Test experiment only reinforces my judgement that the Turing Test says little about machine intelligence" (Vardi 2015). Even assuming that we really wanted to measure likeness to human behaviour—more on this at the end of this paper—, a second objection would be that the Turing test is not a good testing instrument. The interaction is too open-ended to have good properties of measurement invariance and reliability. Precisely because of this, one can argue that virtually anything can be added to correct the Turing test, from sensorimotor interaction (Harnad 1992; Schweizer 1998; Zillich 2012) (e.g., fourth and fifth rows in Table 1) to compression questions (Dowe and Hajek 1997, 1998) (sixth row in Table 1). Most of these variants do not solve the issues but rather introduce new ones. It is important not to blame Turing for this, as Sloman (2014) puts it:

"[Turing] did not propose his 'imitation game' as a test for intelligence, though he occasionally slipped into calling his non-test a test!".

By the end of the previous century, the accumulated criticisms were sufficiently substantial against the Turing test as an actual test for intelligence. The time was ripe to move beyond it. In (Hernández-Orallo 2000), I used the title "Beyond the Turing Test" with a double interpretation: (1) we should be leaving the Turing test behind, and (2) future machine intelligence may go well beyond (and deviate significantly) from human abilities. The question of *what is beyond humans*, in a universal landscape of intelligence, is the exciting question for philosophy and AI research. However, evaluating (machine) intelligence was still an open problem, and really moving beyond the Turing test required an alternative.

In (Hernández-Orallo 2000) I proposed a measure of intelligence that could be non-Boolean (i.e., gradual rather than passing or not a test), factorial (i.e., non-monolithic, capturing several capabilities), non-anthropocentric (i.e., not using humans as references), computational (i.e., considering intelligence some kind of information processing) and meaningful (i.e., knowing what we are measuring). The key idea was defining intelligence test items using algorithmic information theory (Hernández-Orallo and Minaya-Collado 1998), an approach that was followed by many other proposals in the next two decades, from the very influential "universal intelligence" (Legg and Hutter 2007) to the recent "measure of intelligence" (Chollet 2019). However, while some of these proposals have had an important impact on the understanding of what intelligence is, its relation to compression (Dowe et al. 2011), difficulty (Hernández-Orallo 2015; Hernandez-Orallo 2015) and generality (Martinez-Plumed and Hernandez-Orallo 2018), the adoption of some of these tests (or associated definitions) in practice has been very limited.

It is no surprise that many other papers tried to investigate what lies "Beyond the Turing Test". What is more surprising is that most of them used the same or very similar titles (Alvarado et al. 2002; Cohen 2005; Arel and Livingston 2009; French 2012; You 2015; Schoenick et al. 2017), including an AAAI 2015 workshop and special issue in the AI magazine with, yet again, the same title: "Beyond the Turing Test" (Marcus et al. 2015, 2016). This led to yet again the same titles for the headlines by Forbes,[1] the New York Times[2] and even a whole programme by the Templeton World Charity Foundation.[3]

This failure to find—or agree on—an operative alternative to the Turing test that could serve as a beacon for AI (or AGI) partly explains why the Turing test still lingers on in discussions and initiatives about AI evaluation. But there are some other reasons. The Turing test is usually associated with the concept of Human-Level Machine Intelligence (HLMI), either because the former is still thought to be a test for the latter, or because both have the same philosophical and conceptual

[1] https://www.forbes.com/sites/jenniferhicks/2015/09/20/beyond-the-turing-test/#e7206bf22411.

[2] https://opinionator.blogs.nytimes.com/2015/02/23/outing-a-i-beyond-the-turing-test/?ref=opinion&_r=0.

[3] https://www.templetonworldcharity.org/our-work/diverse-intelligences.

assumptions: an anthropocentric view of intelligence and a monolithic scale, where human "level" would be placed at the pinnacle, as far as we know today.

The concept of HLMI is associated with a machine possessing the intelligence of an average human, which 'can carry out human professions at least as well as a typical human" (Bostrom 2014, p. 19), or "capable of matching humans in every (or nearly every) sphere of intellectual activity" (Shanahan 2015). HLMI is frequently presented with other definitions and names, such as 'human-level artificial intelligence', 'high-level machine intelligence' or even just artificial general intelligence (McCarthy 1983; Preston 1991; Nilsson 2006; Zadeh 2008; Bostrom 2014). However, "some [...] feel that the notion of a 'human level' of artificial intelligence is ill-defined" (Bostrom 2014, p. 20). The moment 'it' will be achieved is also said to be "ill-posed" (McDermott 2007). Predictions around the term are hence said to have failed or, more precisely, are simply unverifiable (Armstrong and Sotala 2015). In the same vein are concepts such as superhuman performance or superintelligence, which are directly or indirectly assuming humans as a yardstick. We will address how to circumvent this issue in the following section.

Apart from these associations, there are some other reasons why the Turing test is still a matter of discussion. They have to do with two key components of the Turing test: imitation and adversarialness (Hernández-Orallo 2017b).

Imitation is intrinsic to the Turing test, which is ultimately an imitation game, and as such, it would be *sufficient* for the impostor to imitate humans well. This can be achieved by learning good mind models, an important aspect of social intelligence. Imitation is a more general phenomenon, though, as we will discuss more extensively in the following sections in the context of Turing learning, and machine learning in general; many AI systems just learn by imitating the outputs or the behaviours of other systems, either by a sample of their behaviour (datasets or demonstrations) or by interacting with them.

Adversarialness appears because imitators and judges are opposed. As the imitator gets better the judge must get better too, otherwise it will be fooled. Consequently, when an imitator fools a judge, this might mean either or both of two things: the imitator is good or the judge is bad. It is important to realise that much of the progress in evolution, and social evolution in particular (with conspecifics or heterospecifics), is the result of adversarial co-evolution, from insects and flowers to predators and preys. In sport terms we would say that one gets better when competing against good rivals.

The simplest game that combines imitation and adversarialness (and where both players act as imitators and judges) is 'matching pennies' (a binary version of rock-paper-scissors). This game has been suggested as a minimal intelligence test (Hibbard 2008, 2011; Hernández-Orallo et al. 2012, 2012), and can be regarded as yet another variant of the Turing test (seventh row in Table 1).

The role of imitation and adversarialness in AI has always been important, but several phenomena have made them more relevant in the past two decades. Imitation has become a principle behind many machine learning settings, from supervised learning to reinforcement learning. Inverse reinforcement learning and preference learning, in particular, try to model different aspects of humans or other agents. Adversarialness has been a traditional drive in games, a domain that has been

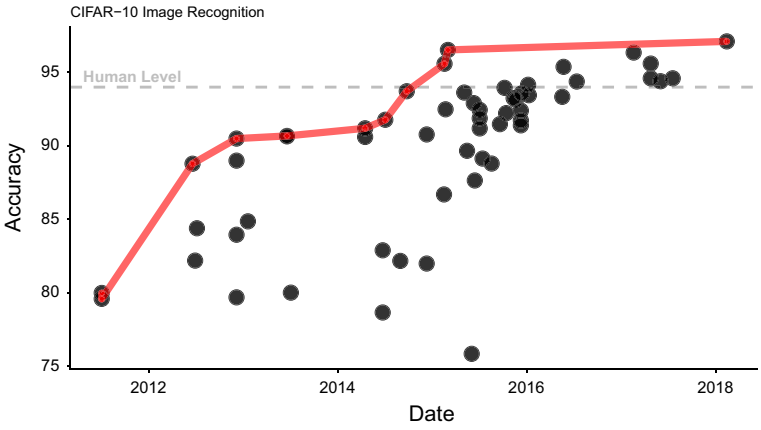

Fig. 1 Evolution of AI performance on the CIFAR10 corpus, with the horizontal dashed line representing average human performance. [Image from the AI Collaboratory (Martínez-Plumed et al. 2020).]

associated with some of the most important breakthroughs in AI (e.g., Campbell et al. 2002; Silver et al. 2017b). Recently, self-play (Silver et al. 2017a) has been vindicated as a very powerful way of making game playing algorithms improve by competing against themselves. The combination of imitation and adversarialness is perfectly captured by Turing learning, a term that generalised generative adversarial models and other kinds of settings where a generator (an imitator) and a discriminator (a judge) play against each other. We will explore this in more detail and its relation to the Turing test in Sect. 4.

3 The Abyss Beyond Superhuman Performance

Even in specific areas of AI where the Turing test is not used or even mentioned, we find countless references to human performance. For instance, Fig. 1 shows the progress in performance for CIFAR10 (Krizhevsky 2009), a very popular image recognition benchmark. This kind of plot is usually portrayed in reports about the state of AI such as the 'AI index' (Shoham 2017), repositories such as 'Papers with Code' (http://www.paperswithcode.com) and interactive exploratory tools such as the 'AI collaboratory' (http://www.aicollaboratory.org). These plots usually represent human performance as a horizontal line, calculated using a human expert or a sample of humans (see, e.g., Russakovsky et al. 2015).

But what does it mean to have the same accuracy as humans? And, more conspicuously, what is the meaning of being 100% correct? Images are labelled by humans, meaning that ground truth depends on human experts or collective human performance. What is superhuman machine vision if better-than-human performance can only happen because the average human makes mistakes on images that are labelled by other humans?

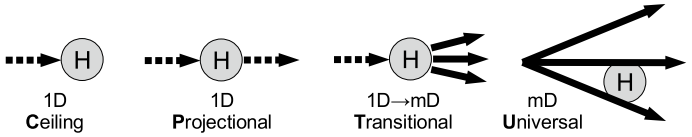

Fig. 2 Four situations when extrapolating beyond human performance. The 'Ceiling' (C) category sets humans (H) as a goal of a one-dimensional space (1D) and nothing cannot go beyond (e.g., Turing test). The 'Projectional' (P) category extrapolates the original dimension, even if the magnitude of the score has no actual meaning (e.g., Pac Man). The 'Transitional' (T) category extends a one-dimensional space with new, more complex instances once human performance has been reached (e.g., ImageNet 2012) using distortions or modifications in many dimensions (mD). Finally, the 'Universal' (U) category defines a (multidimensional, mD) space from the very conception of the task (e.g., brain cancer diagnosis)

In other cases, the extrapolation is even less clear. For instance, the Hybrid Reward Architecture (HRA) has reached the maximum score of 999,990 points for Pac-Man. Compare this to the average and best performance of an average human player, which are estimated to be around 15,693 points and 266,330 points respectively (Van Seijen et al. 2017). We can calculate its 'Absolute Turing Ratio' (Masum et al. 2002), the quotient between the performance of the AI system and humans, which would be approximately 4 if using best human performance as a reference. Clearly, this ratio is meaningless, as score scales in games are simply arbitrary.

It is then quite common that whenever human performance is reached, competitions are usually discontinued and replaced by more challenging benchmarks. This is a 'challenge-solve-and-replace' evaluation dynamics (Schlangen 2019), or a 'dataset-solve-and-patch' adversarial benchmark co-evolution (Zellers et al. 2019). For instance, CIFAR10 is accompanied by the more challenging CIFAR100 (Krizhevsky 2009), SQuAD1.1 gets replaced by SQuAD2.0 (Rajpurkar et al. 2018), GLUE by SUPERGLUE (Wang et al. 2019), and Starcraft by Starcraft II (Vinyals et al. 2017). The underlying problem behind these replacements relies on using human intelligence as a yardstick, limiting our vision beyond these benchmarks. But how can we extrapolate without human yardsticks?

In machine vision, we can get rid of the human reference—and even any human labelling—and define benchmarks in non-anthropocentric terms. For instance, we can create new images (in real or virtual worlds) from scratch, by varying the number of objects, the similarity between them, the locations, etc. We can also add psychophysical distortions, such as rotation, contrast, size, etc. (Rajalingham et al. 2018; Leibo et al. 2018) or increase cognitive difficulty by adding more elements or relations, as done in some human intelligence tests (Dowe and Hernández-Orallo 2012; Hernández-Orallo et al. 2016). However, some other tasks are more strongly linked to humans. For instance, natural language tasks rely on collected corpora from humans. In machine translation, it is hard to conceive how humans should not be taken as a reference. For instance, in machine translation, the original text is generally written by a human and the target of the translation is again given by a human. The quality of the translation between two languages A and B depends on obtaining the same effect on humans whose native language is A as on those humans whose native language is B.

From these and other cases, we can identify different categories, as shown in Fig. 2. The first category, 'Ceiling', represents tasks that cannot be extrapolated, either because the ground truth is human or the task measures *humanity*. The Turing test is a clear example of this category, but some other problems (e.g., realistic human voice generators) also fall under this category. There is an abyss beyond human performance. The second category, 'Projectional', captures those domains for which, once AI reaches human performance, the score can be projected numerically. Video game scores, such as Pac Man, are an example of this category. However, the score is meaningless, because the magnitude is arbitrary or ill-defined. The third category, 'Transitional', represents those problems where instance variations of different difficulty can be created. For instance, we can add Gaussian noise and blur to ImageNet (Dodge and Karam 2017).

Finally, the fourth paradigm in Fig. 2 is originally non-anthropocentric. For instance, the ground truth in brain cancer diagnosis is given by whether a patient develops cancer in a given time window (e.g., 5 years), independently of what human experts predicted. For this problem, we can identify what values make the problem harder, and derive a multidimensional space of performance, where AI systems—and any particular physician—can be located. The key issue behind any extrapolation, and especially the 'universal' category, is a well-defined scale of measurement, where units are meaningful (Hernández-Orallo 2017b; Flach 2019; Hernández-Orallo 2019b, 2020). When figuring out these dimensions we need to consider instances that are *cognitively* harder than those humans could solve. Nevertheless, in order to really break the 'ceiling', humans (or other systems) must be able to *conceive instances that humans cannot solve*.

We definitely reach a conundrum. Thinking of challenging tasks for AI is becoming more and more difficult for humans, because humans have limited capabilities to produce and verify test instances that are difficult enough for many AI benchmarks today. This has happened in areas such as planning and board games, but it is also happening in natural language. For instance, can humans think of a chess position way-out that computers cannot find nowadays? Can humans find easy translation examples where computers fail? These questions arise whenever more challenging benchmarks are asked to replace the old ones. Can we really keep up with this 'challenge-solve-and-replace' phenomenon (Schlangen 2019), with humans being required as the judges who have to find and verify the new challenges?

The crux of the problem can be found in what I call the "cognitive-judge problem": by this I refer to a failure to recognise the manual or automatic cognitive effort that is necessary for producing and verifying instances, and distinguish it from the effort of solving them.[4] Some tasks (e.g., producing a random block world and checking whether the agent has survived after 1,000 time steps) require no cognitive

[4] This separation is well-known in computer science, at least between solving and verifying. For instance, NP problems can be verified easily (in polynomial time), but unless P=NP, we know that solving these problems is much harder than verifying them. For the "cognitive-judge problem" we must distinguish producing, solving and verifying instances, and realise that any of the three can be harder than the others.

effort on the side of the evaluation. But some other tasks do require cognitive effort for producing the instances and/or verifying the solutions. Therefore, effective evaluation depends on finding these resources, usually by relying on previous cognitive human labour (e.g., existing corpora with translations) or by ad-hoc verification effort (e.g., checking each translation made by the machine).

In some domains, producing instances requires more cognitive effort than verifying the solution. Some examples of this situation are:

- A challenging theorem for an automated theorem prover. Producing the instance (making the conjecture) and solving it are usually harder than verifying the solution.
- A small but difficult maze for a navigation robot. Producing a challenging maze that is hard to solve (and of course solving it) is harder than verifying the solution (the robot is out of the maze).
- An image of a bird species for an image recognition system. Finding different species of birds and labelling them is harder than verifying the solution (checking the label).
- A borderline patient record for a cancer screening system. Selecting such a patient case and solving it is harder than verifying the solution (looking at the evolution of the patient).

In many of these cases, verifying the solution is so easy that it can be done automatically, using simple procedures or metrics.[5] In other cases, however, automated procedures and metrics usually fail to do a proper job assessing when answers are correct. The result ends up being verified by a human. For instance:

- A poetic passage to be translated by a machine translation system. Finding an appropriate translation and verifying it is usually complicated.
- A set of facial traits for a face generator to build a composite. Verifying the accuracy of the facial composite is hard—even if an actual photo to compare with is ultimately available.
- An image for a caption generator system. Verifying that the caption makes sense with the image is cognitively hard.
- A trip destination for a routing device. Verifying that the route is the optimal one requires the evaluation of many other alternative routes, which is cognitively hard.

We tend to think that cognitively hard verification mostly happen in natural language processing tasks, but the examples above show that the phenomenon happens in

[5] In some of the cases above, we are assuming that labelling requires human cognitive effort, such as the bird species example where a human must look at the images. But labelling could have been done in other ways, such as a DNA test.

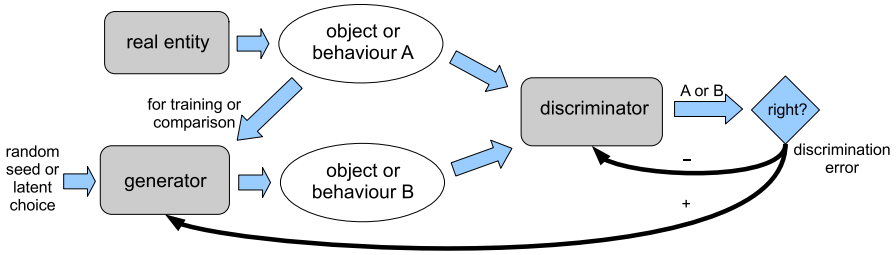

Fig. 3 Schematic representation of Turing learning, for which generative adversarial networks are just a particular case. An object or behaviour coming from a real entity A (e.g., an image from the real world or text produced by humans) is compared against an object or behaviour coming from a machine imitator B (e.g., an image produced by a generator or text produced by a language model). The discriminator is a machine model (a classifier) that has to tell which one is real and which one is an imitation

other areas, especially in generative models (Kynkäänniemi et al. 2019), with human judges being used in the end.[6] Of course, there are some other cases where both producing and verifying instances require cognitive effort, such as writing the first part of a new poem and asking a language model to complete it. In all these cases, but especially when cognitive verification is required, relying on humans to judge the result usually leads to problems of subjectivity, bias, reliability and scalability. These problems will get worse as tasks become more complex and AI becomes more powerful.

But it is precisely the case of generative models that suggests a possible pathway, and solution, to this problem: replace the judges by machines. Indeed, we have seen this change in some recent and popular variants to the Turing test: inverted and reverse Turing tests, as shown in the two bottom rows of Table 1. Let us explore a whole area of AI for which the judge—the discriminator—is a machine. This is known as Turing learning.

4 From Turing Learning back to Testing

The solution to the "cognitive-judge problem" comes precisely from one of the areas of AI that has experienced most progress and attention in the past decade: generative adversarial networks (GANs) (Goodfellow et al. 2014a). Adversarial situations have been exploited in AI as early as some systems played against themselves (self-play) in board games such as checkers (Samuel 1959). But it is the division of two

[6] In language models, 'perplexity' is a very common automatic metric, which basically measures how well the model anticipates the next words in a sentence, and a proxy of how well the model compresses the data. Compression has been connected with the Turing test and (machine) intelligence evaluation a few times (Dowe and Hajek 1997, 1998; Mahoney 1999; Dowe et al. 2011). Despite the correlation between perplexity and other evaluation metrics used by human judges, the latter are still used as ground truth to evaluate conversational agents (see, e.g., Adiwardana et al. 2020).

Fig. 4 Three 256 × 256 synthesised images of the category 'cock' using BigGAN (Brock et al. 2018). 'Truncation' and 'noise seed' parameters are set to 0.5 and 20 respectively. All other parameters are kept as their default values in the Colab implementation

different roles, the generator and the discriminator, which really shapes a new paradigm, covering the production and verification issues of the "cognitive-judge problem". Also, the setting is more closely resembling the Turing test than self-play.

More generally, GANs and other architectures—not necessarily using neural networks—that follow the same paradigm are known as 'Turing learning' (Li et al. 2016; Groß et al. 2017). Figure 3 shows a schematic representation of Turing learning. In this *game*, a real entity A produces some object (e.g., an image) or some behaviour (e.g., a human conversation). At the same time, a machine generator B tries to generate a similar object or imitate A's behaviour. The discriminator, also a machine, has to tell which of A and B corresponds to the genuine entity and which one corresponds to the imitator. The whole procedure is trained by informing the discriminator and the generator of the discrimination error, which will affect negatively on the discriminator (whose goal is to tell correctly between A and B), and will affect positively on the generator (whose goal is to fool the discriminator).

There are variants of this schema. For instance, as depicted in Fig. 3, the discriminator receives two entities and may simply tell which one is the authentic entity and which one is not (more like the Turing test). However, in many implementations, the discriminator just takes one object at a time, and must tell whether it is authentic or generated. It is also important to clarify that there are some constraints about the way the generator can operate. For instance, the generator cannot simply copy the objects or behaviours A produced by the real entity. Typically, the generator works by compressing the training data (a set of objects or behaviours) into a smaller latent space, using some kind of encoding or compressor, such as an autoencoder (Hinton and Zemel 1994; Goodfellow et al. 2016). By choosing some combination of these latent variables (perhaps randomly) the generator can create *new* objects or behaviours B that can be compared with some of the real ones. Usually, generator and discriminator are trained in batches, and not at the same time.

Figure 4 shows the result of three generated images using BigGAN (Brock et al. 2018), a large scale GAN for generating high-fidelity images.[7] The first cock on the

[7] This was implemented using Colab over TensorFlow (https://colab.research.google.com/github/tensorflow/hub/blob/master/examples/colab/biggan_generation_with_tf_hub.ipynb).

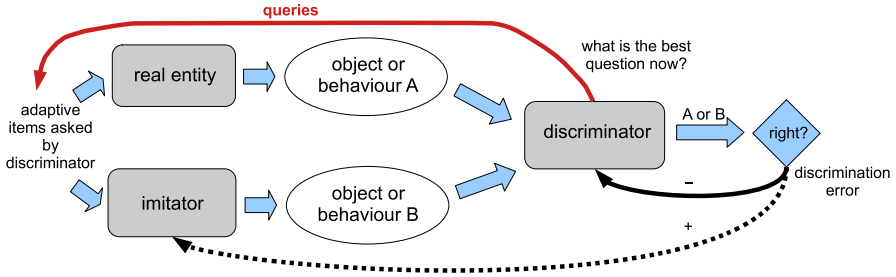

Fig. 5 Schematic representation of Turing 'testing'. In this setting a machine discriminator adapts its queries to a natural player (real entity) and/or an artificial player (a machine imitator), as long as it gets information from both players, learning throughout the process. The two players and the discriminator may have had access to the real world prior to the test, but this access must be controlled during the test to avoid interference

left is realistic, although there is something strange with its legs. Something went clearly wrong with the one in the middle, looking more like conjoined cock twins. The one on the right is the most realistic one.

The idea of combining a generator and a discriminator goes beyond neural networks—actually precedes it, see e.g., Li et al. (2013)—and can be generalised in many ways, not only by considering images, video, audio or text generators, but by the creation of agents whose behaviour is to be discriminated. In particular, Groß et al. (2017) suggest that the discriminators could be turned into "interrogators", à la Turing test. When interaction is present, the discriminator can do some information-based adaptation, such as adaptive sampling, (computerised) adaptive testing or active learning.

In adaptive sampling (Seber and Salehi 2013), the sampler selects the instances that are most discriminating according to the information that it has about the phenomenon of interest in a particular distribution or population. Adaptive testing (Vale and Weiss 1975; Wainer 2000; Weiss 2011) is a kind of adaptive sampling for the specific purpose of evaluation, where the characteristics of the questions (known as items) are chosen adaptively so that the variables to be measured converge faster than by batch, non-adaptive testing. It is quite common to use adaptive testing with Item Response Theory (IRT) (van der Linden 2008), a technique that extracts latent factors about the items, such as difficulty and discrimination. IRT has recently been brought to machine learning and artificial intelligence (Martínez-Plumed et al. 2019). Finally, in active learning (Settles 2009), the situation is determined by a learner that can choose questions to be answered by an oracle (e.g., instances to be labelled in a classification problem) in such a way that the learner can refine its boundaries and areas where it requires more information.

However, the key idea of the Turing test and Turing learning is discrimination, aiming at distinguishing the real thing from the impostor. But Turing learning aims at building a good generator and a good discriminator together, whereas for the purpose of intelligence evaluation, we are mostly interested in building a good discriminator. Distilling on this observation, Fig. 5 represents 'Turing testing', a framework in which the discriminator (a machine) adapts its queries (to either or both the

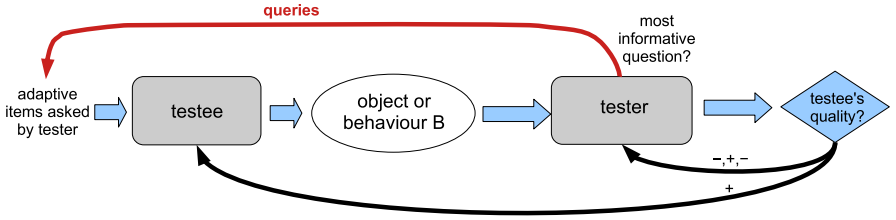

Fig. 6 Schematic representation of Adversarial Testing. In this scenario we get rid of the reference player A, and we only have the machine to be evaluated (called 'testee') and the judge (called 'tester'). They engage in an adversarial game, where the testee tries to get good scores from the evaluator, while the tester tries to find problems that are most informative for determining the evaluator's ability (items that are neither too hard nor too easy for the testee). The testee and tester may have had access to the real world prior to the test, but this access must be controlled during the test to avoid interference

real entity and the imitator) such that depending on their response to the query, it can refine its decision. On top of this, the discriminator also learns during the process, whenever the discrimination error is available. In the Turing testing setting, whether the imitator gets feedback about the discriminator is optional. If it happens then the imitator learns as the discriminator learns, and we have a proper adversarial situation.

In brief, what we are considering here is that judges should be machines, and they should be learning as they interact with the real entity and the impostor. The schema is intentionally asymmetric between imitator and discriminator, and in this way it differs significantly from other adversarial settings such as matching pennies and self-play in games—in both cases the situation is symmetric and there is no need for a judge as the outcome is automatic. The schema in Fig. 5 is much more similar to the inverted and reverse Turing test variants in Table 1.

Figure 5 suggests that evaluation should be as confined as possible, and this should be the case to avoid interference (e.g., the discriminator finds information about the imitator on the Internet). However, this does not mean that the two players and the discriminator should not have access to the real world. The two players and the discriminator may have had previous access to the real world before the evaluation begins, especially for the evaluation of capabilities related to commonsense reasoning or requiring embodiment in the real world. Access to the real world during evaluation may still be possible depending on how the three systems work, but it must be well controlled to avoid interference. This extra care is quite usual in many competitions and evaluation platforms in AI. We mention this access to the real world because it is important to highlight that the discriminator does not need to start from scratch, as in many GAN architectures. The discriminator may have been devised and pre-trained to be a good discriminator. Whether the discriminator is configured to keep on learning and improving during the test is a matter of design, taking advantage that the evaluations of some subjects may be useful to refine the evaluator for other subjects.

While this schema mimics Turing learning very naturally, we can refine it, while keeping some of the principles. If we remove the imitation part of the game but keep the adversarial part, we have a new schema, 'Adversarial Testing', as shown in

Fig. 6, where the discriminator becomes a tester. The imitator has nothing to imitate and simply becomes a testee. The schema becomes very similar to computerised adaptive testing, with the difference that the tester and the testee are thought to work in an adversarial way and learn from each other. The critical part of this setting compared to Turing testing is that we no longer have a reference to imitate. Consequently, the tester must have a measure of progress in the dimensions it is measuring, given by the transitional and universal cases we saw in Fig. 2.

The idea of a machine intelligence test along the principles of adaptive testing predates the concept of Turing learning and was first introduced in (Hernández-Orallo and Dowe 2010), under the concept of an "anytime universal test". In this test, the tester would adapt its questions according to the previous interaction between tester and testee, looking for more informative problems, as in adaptive testing. Note that very easy and very difficult queries are both uninformative, so the evaluator must find those problems that are at the right level. A test with this design can become *anytime* if the accuracy of the estimation increases as more time is given to the test, which can be stopped at any time.

The schema becomes easier to automate when the production and verification of instances is easier than solving them, as happens in many scenarios we discussed in the previous section around the "cognitive-judge problem". But the adaptation becomes more complicated as the judge needs to analyse cognitive contraptions and behaviours. As discussed early on in this paper, we need to evaluate whether an AI system does, for example, a great literary translation from Chinese to English, creates an impactful logo for a new design project, cleans a house appropriately, etc. While these applications are usually evaluated by humans, despite the associated cost and time, the real problem about humans as judges comes from the realisation that not only are humans bad judges for the Turing test, but they have also many limitations for all these other AI evaluation settings. And these limitations become more noticeable as the tasks that AI is solving become more cognitively complex.

As happens with the Turing test and many other tasks, we can select and train humans to become better judges. They can even learn and improve as they do more evaluations, but in the end they will reach an evaluation quality plateau because of their mental resources, motivation and capability. This plateau can nonetheless be broken by machines. There is evidence so far that artificial intelligence is becoming better than humans at capturing some cognitive behaviours. For instance, in social networks, machine learning techniques are now much better than humans at telling the personality or even the IQ of human users (Youyou et al. 2015; Burr and Cristianini 2019).

Turing learning works by maximising both the quality of the generator and the quality of the discriminator. Under some conditions, this is a game that must reach an equilibrium between generator and discriminator. Understanding this game, and its relation to generalisation, is a very active area of research in AI at the moment (see, e.g., Arora et al. 2017). In adversarial testing, both the testee and the tester evolve and have opposed goals too. However, as we said above, the goal of the tester is not to find cases for which the testee fails—this would be just done by choosing very difficult instances—, but to find those that are most informative. For instance, as in adaptive testing, the tester produces instances with high entropy, which in a

binary setting would mean a probability of the testee getting them right around 0.5. This usually means finding items at the right level of difficulty (Vale and Weiss 1975; Wainer 2000; Weiss 2011), which in AI depends on finding scales and units of difficulty (Hernández-Orallo 2019b) or instances that are surprising in terms of unexpected behaviours (predicted easy but failed by the testee, or vice versa). This is an area of enormous interest recently, not coincidentally referred to as 'adversarial examples' (Goodfellow et al. 2014b).

Summing up, in this section we have started from Turing learning and we have distilled some of its principles (many shared with the Turing test) into two different kinds of testing: Turing testing and adversarial testing. There are two main conditions we have identified. The first one is that we should convert human judges into machines that improve adversarially with the systems to be tested. In the future, this can be more (economically) efficient than using humans in general. Machines can be the answer to the limited capacity and robustness of humans to discriminate good solutions in many applications—including the Turing test—, and also to the 'challenge-solve-and-replace' problem (Schlangen 2019). For instance, in multi-agent pathfinding (Stern et al. 2019), we can replace human experts scoring how good a plan or route is by a machine that uses optimality metrics instead, and learns to generate more challenging routes conditioned to previous results of the agents. The second condition, which takes from Turing testing to adversarial testing, is that we should also eliminate the reference (player A) in as many evaluation settings as possible. Having a reference, especially if it is anthropocentric, introduces subjectiveness and costs, and does not help going beyond the reference level. For instance, in self-driving cars, setting the goal as driving like an average human is absurd when we can aim at better targets in terms of metrics such as accident rates, efficiency, pollution, etc. In the end, it is only under these two conditions (no human judges and no human reference) that we will be able to devise true measurement instruments, with absolute scales that can extrapolate without ceilings.

5 Non-thinking Judges and Understanding

Turing replaces the question of thinking machines with a game, for which objections at the time were expected to be less belligerent: can we create a machine whose behaviour is indistinguishable from a human's? Turing does not claim that creating such a machine and passing the 'test' would answer the question of what 'thinking' is. Indeed, determining whether a machine, a human or an animal thinks is a much more complex question, related to issues such as whether the subject shows true *understanding* or is able to extract *meaning* from the world. Because these are still unresolved questions, needing proper definitions of 'understanding' or 'meaning', they tend to be replaced by some other elements, such as whether the subject is able to create models of the world, can perform simulations with them or solves analogies and metaphors (Mitchell 2019, ch. 14, 15). On other occasions thinking is associated with 'common sense', identified as one of the great challenges in AI since its inception (Levesque 2017; Davis and Marcus 2015; Gunning 2018). Common sense has an important component that must be anthropocentric, as it should

capture what humans *usually* see and understand in common situations. Common sense is also aligned with some interpretations of the Turing test as "a guarantee [...] of culturally-oriented human intelligence" (French 1990).

However, there is another component about understanding, or "that which gets the same meaning out of a sequence of symbols as we do" (Hofstadter 1980), which is more essential, and less dependent on previous knowledge. Under this interpretation of understanding, it is not that "the computer will always be unmasked if it has not experienced the world as a human being has" (French 1990), but that the computer will be unmasked if not capable of extracting the right meaning in other more abstract situations. Examples of these more abstract, culture-independent, situations are the experiments with the Bongard problems[8] (Bongard 1970), the Copycat project[9] (Hofstadter and Mitchell 1994), many abstract IQ tests using series or analogies[10] (Hernández-Orallo et al. 2016), comprehension tests based on algorithmic information theory, such as the C-test[11] (Hernández-Orallo 2000) or the new Abstraction and Reasoning Challenge (ARC)[12] (Chollet 2019).

The new evaluation paradigms represented in Figs. 5 and 6 are meant to be applicable to any task or ability. They are general evaluation procedures. The paradigms could be used for evaluating tasks or even capabilities that would not require thinking, understanding or common sense. However, a fundamental question arises were we to use these paradigms, and especially 'adversarial testing', to evaluate 'understanding' or the capability of extracting 'meaning' in a range of situations. This would re-connect this evaluation paradigm to many of the variants of the Turing test seen in Table 1 that were targeting 'thinking', the original question that motivated Turing's imitation game. In what follows we analyse the advantages and caveats of using this paradigm for the evaluation of the elusive notions of thinking, understanding and meaning.

One key motivation why using machines instead of humans for testing understanding comes from the realisation of how easy it is to fool humans into thinking that they are facing a system that understands, when it really does not. This is a well-known phenomenon: humans ascribe agenthood and meaning to the simplest behaviours, what Dennett would refer to as the 'intentional stance' (Dennett 1971). And this

[8] Bongard problems are pattern recognition puzzles, where the diagrams on the left have something in common (e.g., only containing convex polygons) that the diagrams on the right do not (e.g., containing concavities). Telling where a new diagram should belong correctly (left or right) is assumed to reveal that there is *understanding* of the underlying concept.

[9] The Copycat project explored systems that could solve analogies such as "*abc* is to *abd* as *ijk* is to what?", where giving the right answer should reveal the *understanding* of the mechanism that generated the strings.

[10] IQ tests usually include abstract questions with diagrams or numbers. For instance, "What's the odd out of 40, 3, 20 and 80?" assumes *understanding* of a common pattern behind three elements but not the fourth.

[11] The C-test generated letter series using patterns whose algorithmic complexity and 'unquestionability' could be estimated from first principles. For instance, solving instances such as "Continue the series: *abbcccddddde*..." assumes *understanding* of the pattern that generates the series.

[12] ARC is also inspired by algorithmic information theory, but the actual instances resemble pixelated versions of the Bongard problems, where there is a pattern that converts some images into others by playing some algorithmic transformation (e.g., filling the closed areas in the image, mirroring an image, etc.). Finding the pattern should indicate *understanding* of how the transformation works.

stance is very biased in favour of those behaviours that are similar to the beholder, a human in this case. In Watt's words, humans have the "tendency to ascribe mentality and mental states to others in proportion to their similarity to the ascriber [...] This natural faculty biases the [Turing] test, showing up as false positives or negatives" (Watt 1996). More blatantly, humans are "willing to ascribe understanding and consciousness to computers, based on little evidence" (Mitchell 2019).

Another key motivation for the use of the evaluation paradigms in Fig. 6 is the difficulty of discovering the mechanisms behind some behaviour by simple observation, if the testee has no say of what instances are tested. For instance, many AI generators work well when creating an image or a text using an appropriate latent representation or prompt, and there is no further interaction with the generator about what it would do in other situations. For instance, some recent language models such as GPT-3 (Brown et al. 2020, Fig. 3.11) generate text in some domains (e.g., news articles) that are virtually indistinguishable from articles written by humans; the accuracy of human judges detecting them is close to chance (52%). The results fool humans in domains such as humour or poetry. For instance, the following text.[13] is a continuation generated by GPT-3 for Sonnet 18[14]

HUMAN-GENERATED CONTEXT (PROMPT):

```
Shall I compare thee to a summer's day?
Thou art more lovely and more temperate:
Rough winds do shake the darling buds of May,
And summer's lease hath all too short a date;
```

MACHINE-GENERATED CONTINUATION:

```
A winter's day, when beams of sun are few,
And skies are grey with clouds intemperate,
```

As surprising and realistic some generated text might look like, these examples usually are accompanied by other examples of how meaningless and pointless some continuations are. The interesting thing is that the evaluation of these systems by humans has become an adversarial game, but not because the system is interrogated in the form of a question/answer system or a conversational bot, but rather in a more fundamental way. In order to use these systems, humans must look for contexts and prompts (texts that are used as inputs to the language model) such that they get the desired continuation (right or wrong, depending on the purpose). Apart from the many different ways in which a context can lead to a bad answer, it is especially interesting to see what happens with questions that challenge factual knowledge and require some degree of understanding. For instance, this is a simple example that takes GPT-3 to its (short) limits.[15]

[13] Taken from https://www.gwern.net/GPT-3.

[14] This sonnet was also used by Turing in some of his examples about the imitation game (Turing 1950).

[15] Taken from https://lacker.io/ai/2020/07/06/giving-gpt-3-a-turing-test.html.

```
Q: Who was president of the United States in 1600?
A: Queen Elizabeth I was president of the United States in 1600.
```

This does not really show a lack of knowledge, and the inference makes some sense (although most of the area of the United States at the time was either inhabited by Native Americans or was under the control of a different colonial Empire). It seems more that the language model per se is not using its estimated probabilities to do some primitive metacognition (an 'I don't know' answer or any diversion trick), and is not dealing with related knowledge about the question. Of course, a language model is not a full model of the world for which *inferences*—and not only continuations—could be done about any of its particular states.

In this context, we picture new research looking for automatically generated prompts for language models and other sophisticated AI engines. In particular, Jiang et al. (2020b) are able to paraphrase questions in various ways and combine the answers to make a language model give more robust solutions to Q/A tasks. While this may show some increase of performance in some 'language understanding' tasks, the use of an ensemble to derive the answer to a question may raise more brows as whether the system is actually having *a* model of the world, not to say understanding its own outputs. Nevertheless, for our purposes, it is more interesting to think of the opposite situation: machine-generated prompts that are able to detect lack of understanding. The automatic generation of instances or distractors for human testing in psychology is commonplace, but it has only become common in AI recently.

For instance, SWAG (Situations With Adversarial Generations) (Zellers et al. 2018) takes the true video caption for the next event in a video sequence and generates three distractors (incorrect answers) automatically. These answers are "adversarially generated and human verified, so as to fool machines but not humans". This means of course that the generation is not fully automated, as human cognition is needed at the end of the loop. This is again a consequence of the way the benchmark is conceived, using humans as a reference. In other cases, the reliance on humans is even more explicit. For instance, adversarial NLI (Nie et al. 2019) is a benchmark that asks humans for questions that are easy for them but that can fool a model, which is provided to the human. While in this case the human is assisted by a model to produce new questions, the key idea in Fig. 6 is whether we can take humans out of the loop completely. For instance, Zhou et al. (2020) generate probes using syntactically different but logically-equivalent expressions. This is simply very effective: the results show that pre-trained language models are no better than random guessing. Related ideas using explicit or implicit patterns for the generation of instances in AI evaluation have been used before in benchmarks such as Winograd Schema Challenge (Levesque et al. 2012), or the extended version of the challenge, Winogrande (Sakaguchi et al. 2019), which also uses adversarial filtering inspired

by SWAG. Other approaches combined templates and human computation (Amazon Turk) to generate adversarial datasets (Rozen et al. 2019).

When dealing with language understanding or commonsense reasoning, the use of patterns or other mechanisms to generate instances automatically is aiming at taking humans out of the loop, in the same direction of Figs. 5 and 6. However, the adversarial mechanism must be *responsive* and *customised*. This is basically what an interactive dialogue provides (as originally conceived with the imitation game), unleashing all the power of an interview-like evaluation. Not all kinds of evaluation following the paradigms of Turing testing and adversarial testing must be in the form of an interview. However, there must be some reactive interaction, such that the question or problem that comes next depends on the previous answers or solutions by the particular agent the tester is evaluating. For instance, it may well be that a common sense test is conducted in a video game setting for reinforcement learning agents (Jiang et al. 2020a). In other words, the test must be adaptive, independently of the kind of communication and modality (and there are many options in the variants in Table 1 and in many other benchmarks in AI). Adaptive testing (Vale and Weiss 1975; Wainer 2000; Weiss 2011) makes testing more efficient when evaluating some other capabilities, as already mentioned in previous sections, but it turns crucial when evaluating 'understanding', 'commonsense' or being able to find 'meaning'. Let us analyse why this is so.

The question of assessing whether agent A understands a concept or idea, represented by a model M, implies taking the model to its limits, to find the borderline cases where the answers are most informative about whether A really works with model M internally. We have seen this with the example about the president of the United States in 1600. For most instances, statistically, many other models of the world are compatible with what the AI system is outputting. High performance can be obtained with the wrong model, à la Clever Hans. It is then important that the focus of the tester soon moves from the overall performance provided by statistically-easy instances to peculiar situations that can rule out some other interpretations of the observed performance. This is specifically what other non-adaptive tests do, such as Raven matrices, letter series, Copycat, the C-test or ARC, mentioned above. But given a machine evaluator in an interactive setting, the goal should be to select the most informative questions. This does not necessarily mean that the tester must be more capable than the testee. The evaluator may simply generate models at random using some appropriate representations, as done in automatically generated test instances for humans or machines. In some situations dealing with the real world, the evaluation is more constrained. For instance, it is harder—but not impossible—to imbue the subtleties of naive physics or naive psychology into an evaluator such that it generates situations adaptively for a particular testee. In the absence of these models of the world, the evaluator can simply act as a learner, as in an active learning situation (Settles 2009), to check the properties of the model identified by the testee.

In many domains, such as music, a person can recognise a good performer even if they do not perform themselves. If evaluating is simpler than performing then the cognitive-judge problem can be circumvented. Of course, this may not be the case for all domains. Actually, for humanlike intelligence, Watt (1996) actually argued

Fig. 7 Five 256 × 256 synthesised images going from the category 'cock' to 'hen' using BigGAN interpolation (Brock et al. 2018). 'Truncation' and 'noise seed' parameters (for both categories) are set at 0.5, 0 and 0 respectively. All other parameters are kept as their default values in the Colab implementation

that being able to distinguish humans from machines was a criterion for intelligence. I disagree with the sufficiency of this criterion. The success of machines quantifying and categorising human behaviour in social networks, from personality to IQ (Youyou et al. 2015; Burr and Cristianini 2019), as we mentioned before, is a sign that this may be possible. In sum, we should explore the possibility of non-intelligent machine judges that may still do a good job at telling between humans and machines. CAPTCHAs will explore this route for a time. In a more long term, there is the philosophical question about whether a system that is not thinking can reliably determine whether another system is thinking or not, or the related question of how much intelligence is needed to test intelligence. These are open questions, especially if we are not more specific about what we are testing and how we would evaluate the intelligence of the tester and the testee independently. What is more certain is that a race has started to build 'machine judges'.[16] This originated with the detection of Clever Hans phenomena in AI systems (Sturm 2014), a problem that is very much related to the increasingly important area of explainable AI, but will continue with the challenge of building more comprehensive tester machines.

6 Building Behavioural Taxonomies

Turing learning is now consolidated as a technique that makes generator and discriminator reach high levels of competence. This suggests new applications of Turing testing for determining how similar two behaviours are, beyond images, videos, audios and text.

The first thing we need to understand is that for any task, the imitator creates a latent space in which any two points can be interpolated. For instance, Fig. 7 shows several images that are generated from the category 'cock' to the category 'hen' (using the same BigGAN technology as in Fig. 4). As we see in the progression, the intermediate images are points in this space that are midway a male and a female, points that do not exist in the real world. When thinking about creating new AI behaviours such as agents and other kinds of systems that are not necessarily generators, it is important to visualise this continuous space.

[16] These judges may have a particular training and developmental process, as child machine judges.

Consider that we want to analyse whether two agents have the same behaviour. The discriminator should think of those test instances—environments in this case—such that it can tell between the two agents. However, if both agents are stochastic, some of these different behaviours may not really imply that they are different. Actually, by taking the same agent twice (as both player A and B), we could observe different behaviours, just because it is stochastic. Again, the latent space solves the conundrum. Even if the behaviours are sometimes different because of random effects, what matters is whether the two agents are close in the latent space. For humans we use abstract traits such as cognitive abilities and personalities, and we should do similarly for every kind of agent, be it natural or artificial (Hernández-Orallo 2017b). The discriminator must also learn to build this abstract latent space in which the distances can be converted into meaningful similarity metrics.

With an appropriate design of these discriminators, we would use Turing testing to output a similarity value, a *similarity metric* that could be used to cluster agents together. This would be a very powerful tool for mapping the intelligence of different kinds of behaviours (Bhatnagar et al. 2017), including the comparison between machine learning families (Fabra-Boluda et al. 2020), AI systems and humans (Insa-Cabrera et al. 2011b, a), AI systems and animals (Hernández-Orallo 2017b; Crosby et al. 2019, 2020) or humans, animals and different deep learning architectures (Schrimpf et al. , 2018, using the so-called Brain Score[17]). Of course, for n agents, we would need to create a similarity matrix of size $\frac{n\times(n-1)}{2}$, which may be impractical if n is large—but clustering with sparse similarity matrices is an option.

An alternative approach relies on the other testing setting seen in the previous section: adversarial testing. Whereas the development of measurement instruments that follow the adversarial testing is still incipient, and has not progressed significantly since (Hernández-Orallo and Dowe 2010; Hernández-Orallo et al. 2012), it adapts according to one or more dimensions, as per the transitional and universal cases in Fig. 2. Assuming each dimension is defined by a difficulty metric (Mishra et al. 2013; Hernández-Orallo 2015; Hernandez-Orallo 2015; Martinez-Plumed and Hernandez-Orallo 2018; Martínez-Plumed et al. 2019; Hernández-Orallo 2020), we have a multidimensional space for which the adversarial testing can derive the location of the testee in this space. By doing this, similarities and clustering are calculated in this space, with no need of exploring all the $\frac{n\times(n-1)}{2}$ combinations when n agents are being analysed.

In this taxonomical endeavour, as many others where humans play a part, two of the most relevant questions are (1) the location of humans in this space and (2) the comparisons of other systems against humans. For instance, it has taken enormous scientific and pedagogic effort to see humans as a particular kind of ape. While this is generally accepted today, many other species are compared against humans—from cognition to immune systems—, for the insight and applicability of the comparison. Despite this preponderance, we must see the landscape in a non-anthropocentric way. For instance, we should not associate general intelligence exclusively with humans. There is general intelligence in

[17] http://www.brain-score.org/.

animals (Burkart et al. 2017), taking very different forms and manifestations. Some of the intelligent behaviours that are common in all humans are similar to those we find in some other animals. For particular capabilities, some animals sometimes score beyond humans. It is also a fact that humans show enormous differences in behaviour, and really determining how much a particular human is able to understand, depending on their developmental stage and their capabilities—or disabilities—is a hard question. Representing humans as a point in the space rather than a cloud is a mistake, even when we compare against AI systems. In the end, when we see humans as a distribution, and we realise the fewer constraints we have when devising AI systems, it is easier to consider other ways in which machines can develop and display general intelligence.

Despite the non-anthropocentric perspective, devising tests of humanlike cognitive behaviour is important scientifically. For many applications it is also key to build AI systems that learn and infer like us, so that communicating with them will be easier, as well as anticipating their behaviour. Efforts such as DiCarlo's 'brain score' (Schrimpf et al. 2018), mentioned above, measuring how humanlike other perception mechanisms are, is a major contribution in this direction. But it is also crucial to develop metrics to test capabilities independently of how humanlike they are. It is especially interesting in philosophical terms, and in connection with some of the debates about the Turing test. The success in many of the abilities that resist AI technology today, such as understanding and making sense of the world, may lead to AI systems that differ very much from humans. Consequently, these difference would be be detected in a Turing test (and not only because of what Turing called "human fallibility" (Turing 1950)). It would also be scientifically enlightening to overhaul human variability in behaviour under this magnified view, and understand how humans will be moving in this space as the result of using cognitive enhancers fuelled by AI (Hernández-Orallo and Vold 2019), to the point that humans of the future may not be 'humanlike' any more. This falls into a more long-term endeavour of characterising humanlike behaviour in this landscape of cognition, and understanding what humanlike really represents.

There are many more questions about machine behaviour (Rahwan et al. 2019) that go well beyond comparing them to humans. As in comparative cognition, comparing all kinds of systems between them and against some other imaginary or interpolating systems can give enormous scientific and philosophical insight about artificial and natural intelligence, mapping them into the same space (Bhatnagar et al. 2017), given or latent (Hernández-Orallo 2001). The two schemas seen in the previous section, Turing testing and adversarial testing, can exploit the power of artificial judges—adversarial and adaptive testers—to boost this process, as we have witnessed in the area of Turing learning in the past few years. Initiatives such as the AI collaboratory (Martínez-Plumed et al. 2020) can benefit from increasingly more numerous and accurate data deriving from these evaluations.

7 Discussion

The Turing test is perhaps one of the most insightful thought experiments about the mind. However, several problems have been widely recognised when repurposed as a measurement instrument. Some of these problems are rooted in its anthropocentrism. Whereas the imitation game was introduced to argue that intelligence could be incarnated by machines, the two other players in the game, the reference and the judge, were set to be human. Philosophically, using humans as references seems natural from the standpoint of humans, and pragmatically the Homo Sapiens represents many capabilities we would like to imitate in intelligent machines. However, the use of humans as a reference has been criticised, not only for evaluation, but also in obscure terms such as human-level machine intelligence. Anthropocentrism makes extrapolation beyond humans cumbersome, if not impossible. When considering humans in a vast space of intelligence, locating them as yet another point in this space becomes a Copernican revolution, leading to the really interesting and challenging questions about intelligence and the mind (Hernández-Orallo et al. 2011; Hernández-Orallo and Dowe 2013; Dowe and Hernández-Orallo 2014; Hernández-Orallo et al. 2014).

One of these challenges is choosing who plays the judge when evaluating AI systems at present, in the upcoming years and especially in the distant future. We have argued that replacing human judges by machines is supported by the success of new AI contraptions such as Turing learning. There, a machine discriminator learns from the interaction with a second system whose performance is boosted and evaluated as the result of an adversarial process. It is this adversarial feature of the Turing test that lives on with Turing learning, and can be adapted in settings such as Turing testing and adversarial testing introduced here. In order to make these schemas work successfully we need to focus on the following issues:

- We should scrutinise any task used in AI whose production and verification is not fully automated (i.e., human judges). We have seen that discriminators can be automated. Discriminators can be extended to an adaptive evaluation setting where they figure out questions and instances that are of the right difficulty and discriminating power, better than the questions humans could do.
- We should explore the best ways in which testees and testers can work adversarially, exploring different configurations and loss functions for them. This can mimic the way GANs were extended to Turing learning, with different architectures and loss functions being explored so far.
- We should separate humanlike behaviours from intelligent behaviours, so that we can properly understand the intersection. Notions such as 'understanding' or extracting meaning from the world may take different forms beyond the standard human, as happens with the diversity of human populations and the huge diversity in animal cognition. AI behaviour is only expected to be more diverse.
- We should understand how much computational effort we need for the tester in comparison with the computation effort for the testee. While we have dis-

cussed some similes (e.g., NP vs P) where testers are more lightweight than testees, we need to analyse this question for specific and general domains in AI evaluation (Hernández-Orallo 2017a; Hernández-Orallo et al. 2017), in experimental and theoretical ways.
– We should identify the dimensions of each domain we need to evaluate and the difficulty metrics for each of them. Again, the latent spaces created by the machine testers can be very useful to build and refine the space, as AI progresses. This abstraction will move us from a task-oriented evaluation to an ability-oriented evaluation (Hernández-Orallo 2017a).

We can see plenty there that needs to be done to make the new testing settings work, first in a range of AI domains and then more broadly for the comparison of systems that display some general intelligent behaviour. There is also plenty that needs to be done to use the information from these evaluations in a more insightful way, through the use of taxonomies and the development of new theories of cognition. In the end, it is no surprise that AI can be useful for cognitive measurement; indeed, it can also be useful for comparative cognition and it may replace human judges in any evaluation setting in the future. In Turing's words (1950): "we may hope that machines will eventually compete with [humans] in all purely intellectual fields": intelligence evaluation is just one of these fields.

Acknowledgements I appreciate the reviewers' comments, leading to new Sect. 5, among other modifications and insights in the final version. This work was funded by the Future of Life Institute, FLI, under grant RFP2-152, and also supported by the EU (FEDER) and Spanish MINECO under RTI2018-094403-B-C32, and Generalitat Valenciana under PROMETEO/2019/098. Figure 1 was kindly generated on purpose by Fernando Martínez-Plumed.

References

Adiwardana, D., Luong, M. T., So, D. R., Hall, J., Fiedel, N., Thoppilan, R., Yang, Z., Kulshreshtha, A., Nemade, G., Lu, Y., et al. (2020) Towards a human-like open-domain chatbot. arXiv:200109977.

Alvarado, N., Adams, S. S., Burbeck, S., & Latta, C. (2002). Beyond the Turing test: Performance metrics for evaluating a computer simulation of the human mind. In *The 2nd international conference on development and learning*, 2002 (pp. 147–152). IEEE.

Arel, I., & Livingston, S. (2009). Beyond the Turing test. *Computer*, *42*(3), 90–91.

Armstrong, S., & Sotala, K. (2015). How we're predicting AI–or failing to. In *Beyond artificial intelligence*(pp. 11–29). New York: Springer.

Arora, S., Ge, R., Liang, Y., Ma, T., & Zhang, Y. (2017). Generalization and equilibrium in generative adversarial nets (GANS). In *Proceedings of the 34th international conference on machine learning* (Vol. 70, pp. 224–232). JMLR. org.

Bhatnagar, S., et al. (2017). Mapping intelligence: Requirements and possibilities. In *PTAI* (pp. 117–135). New York: Springer.

Bongard, M. M. (1970). *Pattern Recognition*. New York: Spartan Books.

Borg, M., Johansen, S. S., Thomsen, D. L., & Kraus, M. (2012). Practical implementation of a graphics Turing test. In *Advances in visual computing* (pp. 305–313). New York: Springer.

Bostrom, N. (2014). *Superintelligence: Paths, dangers, strategies*. Oxford: Oxford University Press.

Brock, A., Donahue, J., & Simonyan, K. (2018). Large scale GAN training for high fidelity natural image synthesis. arXiv:180911096.

Brown, T. B., Mann, B., Ryder, N., Subbiah, M., Kaplan, J., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., et al. (2020). Language models are few-shot learners. arXiv:200514165.

Burkart, J. M., Schubiger, M. N., & van Schaik, C. P. (2017). The evolution of general intelligence. *Behavioral and Brain Sciences*, *40*, e195.

Burr, C., & Cristianini, N. (2019). Can machines read our minds? *Minds and Machines*, *29*(3), 461–494.

Campbell, M., Hoane, A. J., & Hsu, F. (2002). Deep Blue. *Artificial Intelligence*, *134*(1–2), 57–83.

Chollet, F. (2019). The measure of intelligence. arXiv:191101547.

Cohen, P. R. (2005). If not Turing's test, then what? *AI Magazine*, *26*(4), 61.

Copeland, B. J. (2000). The Turing test. *Minds and Machines*, *10*(4), 519–539.

Copeland, J., & Proudfoot, D. (2008). Turing's test. A philosophical and historical guide. In R. Epstein, G. Roberts, G. Beber (Eds.), *Parsing the Turing Test. Philosophical and Methodological Issues in the Quest for the Thinking Computer*. New York: Springer.

Crosby, M., Beyret, B., Shanahan, M., Hernandez-Orallo, J., Cheke, L., & Halina, M. (2020). The animal-AI testbed and competition. *Proceedings of Machine Learning Research*, *123*, 164–176.

Crosby, M., Beyret, B., Hernandez-Orallo, J., Cheke, L., Halina, M., & Shanahan, M. (2019). Translating from animal cognition to AI. NeurIPS workshop on biological and artificial reinforcement learning.

Davis, E., & Marcus, G. (2015). Commonsense reasoning and commonsense knowledge in artificial intelligence. *Communications of the ACM*, *58*(9), 92–103.

Dennett, D. C. (1971). Intentional systems. *The Journal of Philosophy*, *68*, 87–106.

Dodge, S., & Karam, L. (2017). A study and comparison of human and deep learning recognition performance under visual distortions. In *ICCCN* (pp. 1–7). IEEE.

Dowe, D. L., & Hernández-Orallo, J. (2012). IQ tests are not for machines, yet. *Intelligence*, *40*(2), 77–81.

Dowe, D. L., & Hernández-Orallo, J. (2014). How universal can an intelligence test be? *Adaptive Behavior*, *22*(1), 51–69.

Dowe, D. L., Hernández-Orallo, J., & Das, P. K. (2011). Compression and intelligence: Social environments and communication. In J. Schmidhuber, K. Thórisson, & M. Looks (Eds.), *Artificial general intelligence* (Vol. 6830, pp. 204–211)., LNAI series New York: Springer.

Dowe, D. L., Hajek, A. R. (1997). A computational extension to the Turing test. In *Proceedings of the 4th Conference of the Australasian Cognitive Science Society, University of Newcastle, NSW, Australia*. Also as Technical Report #97/322, Dept Computer Science, Monash University, Australia.

Dowe, D. L., Hajek, A. R. (1998). A non-behavioural, computational extension to the Turing Test. In *Intl. conf. on computational intelligence & multimedia applications* (ICCIMA'98) (pp. 101–106). Gippsland, Australia.

Fabra-Boluda, R., Ferri, C., Martínez-Plumed, F., Hernández-Orallo, J., & Ramírez-Quintana, M. J. (2020). Family and prejudice: A behavioural taxonomy of machine learning techniques. In *ECAI 2020—24st European conference on artificial intelligence*.

Flach, P. (2019). Performance evaluation in machine learning: The good, the bad, the ugly and the way forward. In *AAAI*.

Fostel, G. (1993). The Turing test is for the birds. *ACM SIGART Bulletin*, *4*(1), 7–8.

French, R. M. (1990). Subcognition and the limits of the Turing test. *Mind*, *99*(393), 53–65.

French, R. M. (2000). The Turing test: The first 50 years. *Trends in Cognitive Sciences*, *4*(3), 115–122.

French, R. M. (2012). Moving beyond the Turing test. *Communications of the ACM*, *55*(12), 74–77. https://doi.org/10.1145/2380656.2380674.

Goodfellow, I., Bengio, Y., & Courville, A. (2016). *Deep learning*. Cambridge: MIT press.

Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., & Bengio, Y. (2014a). Generative adversarial nets. In *Advances in neural information processing systems* (pp 2672–2680).

Goodfellow, I. J., Shlens, J., & Szegedy, C. (2014b). Explaining and harnessing adversarial examples. arXiv:14126572.

Groß, R., Gu, Y., Li, W., & Gauci, M. (2017). Generalizing GANs: A Turing perspective. In *Advances in neural information processing systems* (pp. 6316–6326).

Gunning, D. (2018). Machine common sense concept paper. arXiv:181007528.

Harnad, S. (1992). The Turing test is not a trick: Turing indistinguishability is a scientific criterion. *ACM SIGART Bulletin*, *3*(4), 9–10.

Hayes, P., & Ford, K. (1995). Turing test considered harmful. In *International joint conference on artificial intelligence* (IJCAI) (pp 972–977).

Hernandez-Orallo, J. (2015). Stochastic tasks: Difficulty and Levin search. In J. Bieger, B. Goertzel, & A. Potapov (Eds.), *Artificial general intelligence—8th international conference, AGI 2015, Berlin, Germany*, July 22–25, 2015 (pp. 90–100). New York: Springer.

Hernández-Orallo, J. (2000). Beyond the Turing test. *Journal of Logic, Language & Information*, *9*(4), 447–466.

Hernández-Orallo, J. (2001). *On the computational measurement of intelligence factors* (pp. 72–79). Gaithersburg: NIST Special Publication.

Hernández-Orallo, J. (2015). On environment difficulty and discriminating power. *Autonomous Agents and Multi-Agent Systems*, *29*, 402–454.

Hernández-Orallo, J. (2017a). Evaluation in artificial intelligence: From task-oriented to ability-oriented measurement. *Artificial Intelligence Review*, *48*(3), 397–447.

Hernández-Orallo, J. (2017b). *The measure of all minds: Evaluating natural and artificial intelligence*. Cambridge: Cambridge University Press.

Hernández-Orallo, J. (2019a). Gazing into clever Hans machines. *Nature Machine Intelligence*, *1*(4), 172–173.

Hernández-Orallo, J. (2019b). Unbridled mental power. *Nature Physics*, *15*(1), 106.

Hernández-Orallo, J., & Dowe, D. L. (2010). Measuring universal intelligence: Towards an anytime intelligence test. *Artificial Intelligence*, *174*(18), 1508–1539.

Hernández-Orallo, J., & Dowe, D. L. (2013). On potential cognitive abilities in the machine kingdom. *Minds and Machines*, *23*(2), 179–210.

Hernández-Orallo, J., Dowe, D. L., España-Cubillo, S., Hernández-Lloreda, M. V., & Insa-Cabrera, J. (2011). On more realistic environment distributions for defining, evaluating and developing intelligence. In J. Schmidhuber, K. Thórisson, & M. Looks (Eds.), *Artificial general intelligence* (Vol. 6830, pp. 82–91)., LNAI New York: Springer.

Hernández-Orallo, J., Dowe, D. L., & Hernández-Lloreda, M. V. (2014). Universal psychometrics: Measuring cognitive abilities in the machine kingdom. *Cognitive Systems Research*, *27*, 50–74.

Hernández-Orallo, J., Insa-Cabrera, J., Dowe, D. L., & Hibbard, B. (2012). Turing tests with Turing machines. *Turing*, *10*, 140–156.

Hernández-Orallo, J., Martínez-Plumed, F., Schmid, U., Siebers, M., & Dowe, D. L. (2016). Computer models solving intelligence test problems: Progress and implications. *Artificial Intelligence*, *230*, 74–107.

Hernández-Orallo, J. (2015). C-tests revisited: Back and forth with complexity. In J. Bieger, B. Goertzel, & A. Potapov (Eds.), A*rtificial general intelligence—8th international conference, AGI 2015*, Berlin, Germany, July 22–25, 2015. New York: Springer (pp. 272–282).

Hernández-Orallo, J. (2020). AI evaluation: On broken yardsticks and measurement scales. Evaluating AI Evaluation @ AAAI.

Hernández-Orallo, J., & Minaya-Collado, N. (1998). A formal definition of intelligence based on an intensional variant of Kolmogorov complexity. In *Proc. intl symposium of engineering of intelligent systems* (EIS'98) (pp. 146–163). ICSC Press.

Hernández-Orallo, J., & Vold, K. (2019). Ai extenders: The ethical and societal implications of humans cognitively extended by ai. In *Proceedings of the 2019 AAAI/ACM Conference on AI, Ethics, and Society* (pp. 507–513).

Hernández-Orallo, J., Insa-Cabrera, J., Dowe, D.L., & Hibbard, B. (2012). Turing machines and recursive Turing Tests. In V. Muller, & A. Ayesh (Eds.), AISB/IACAP 2012 Symposium "Revisiting Turing and his Test", The Society for the Study of Artificial Intelligence and the Simulation of Behaviour, pp 28–33.

Hernández-Orallo, J., Baroni, M., Bieger, J., Chmait, N., Dowe, D. L., Hofmann, K., et al. (2017). A new AI evaluation cosmos: Ready to play the game? *AI Magazine*, *38*(3), Fall 2007.

Hibbard, B. (2008). Adversarial sequence prediction. *Frontiers in Artificial Intelligence and Applications*, *171*, 399.

Hibbard, B. (2011). Measuring agent intelligence via hierarchies of environments. In *Artificial general intelligence* (pp. 303–308). New York: Springer.

Hingston, P. (2009). The 2k botprize. In *IEEE symposium on computational intelligence and games* (CIG 2009) (pp. 1–1). IEEE.

Hinton, G. E., & Zemel, R. S. (1994). Autoencoders, minimum description length and Helmholtz free energy. In *Advances in neural information processing systems* (pp. 3–10).

Hofstadter, D. R. (1980). *Gödel, escher, bach*. New York: Vintage Books.

Hofstadter, D. R., & Mitchell, M. (1994). *The Copycat project: A model of mental fluidity and analogy-making*. Norwood, NJ: Ablex Publishing.
Insa-Cabrera, J., Dowe, D. L., España-Cubillo, S., Hernández-Lloreda, M. V., & Hernández-Orallo, J. (2011a). Comparing humans and AI agents. In *International conference on artificial general intelligence* (pp. 122–132). New York: Springer.
Insa-Cabrera, J., Dowe, D. L., & Hernández-Orallo, J. (2011b). Evaluating a reinforcement learning algorithm with a general intelligence test. In J. Lozano, J. Gamez, & J. Moreno (Eds.), *Current topics in artificial intelligence* (CAEPIA 2011). LNAI Series 7023. New York: Springer.
Jiang, Z., Xu, F. F., Araki, J., & Neubig, G. (2020b). How can we know what language models know? *Transactions of the Association for Computational Linguistics*, *8*, 423–438.
Jiang, M., Luketina, J., Nardelli, N., Minervini, P., Torr, P. H., Whiteson, S., & Rocktäschel, T. (2020a). Wordcraft: An environment for benchmarking commonsense agents. arXiv:200709185.
Krizhevsky, A. (2009). Learning multiple layers of features from tiny images. Master's thesis, University of Toronto, https://www.cs.toronto.edu/~kriz/cifar.html.
Kynkäänniemi, T., Karras, T., Laine, S., Lehtinen, J., & Aila, T. (2019). Improved precision and recall metric for assessing generative models. arXiv:190406991.
Legg, S., & Hutter, M. (2007). Universal intelligence: A definition of machine intelligence. *Minds and Machines*, *17*(4), 391–444.
Leibo, J. Z., et al. (2018). Psychlab: A psychology laboratory for deep reinforcement learning agents. arXiv:180108116.
Levesque, H. J. (2017). *Common sense, the Turing test, and the quest for real AI*. New York: MIT Press.
Levesque, H., Davis, E., & Morgenstern, L. (2012). The Winograd schema challenge. In *Thirteenth international conference on the principles of knowledge representation and reasoning*.
Li, W., Gauci, M., & Groß, R. (2016). Turing learning: A metric-free approach to inferring behavior and its application to swarms. *Swarm Intelligence*, *10*(3), 211–243.
Li, W., Gauci, M., & Groß, R. (2013). A coevolutionary approach to learn animal behavior through controlled interaction. In *Proceedings of the 15th annual conference on Genetic and evolutionary computation* (pp. 223–230).
van der Linden, W. J. (2008). Using response times for item selection in adaptive testing. *Journal of Educational and Behavioral Statistics*, *33*(1), 5–20.
Mahoney, M. V. (1999). Text compression as a test for artificial intelligence. In *Proceedings of the national conference on artificial intelligence* (pp 970–970). AAAI.
Marcus, G., Rossi, F., & Veloso, M. (2016). Beyond the Turing test (special issue). *AI Magazine*, *37*(1), 3–101.
Marcus, G. (2020). The next decade in AI: Four steps towards robust artificial intelligence. arXiv:200206177.
Marcus, G., Ross, F., & Veloso, M. (2015). Beyond the Turing test. AAAI workshop, http://www.math.unipd.it/~frossi/BeyondTuring2015/.
Martinez-Plumed, F., & Hernandez-Orallo, J. (2018). Dual indicators to analyse AI benchmarks: Difficulty, discrimination, ability and generality. *IEEE Transactions on Games*, *12*, 121–131.
Martínez-Plumed, F., Prudêncio, R. B., Martínez-Usó, A., & Hernández-Orallo, J. (2019). Item response theory in AI: Analysing machine learning classifiers at the instance level. *Artificial Intelligence*, *271*, 18–42.
Martínez-Plumed, F., Gomez, E., & Hernández-Orallo, J. (2020). Tracking AI: The capability is (not) near. In *European conference on artificial intelligence*.
Masum, H., Christensen, S., & Oppacher, F. (2002). The Turing ratio: Metrics for open-ended tasks. In *Conf. on genetic and evolutionary computation* (pp. 973–980). Morgan Kaufmann.
McCarthy, J. (1983). Artificial intelligence needs more emphasis on basic research: President's quarterly message. *AI Magazine*, *4*(4), 5.
McDermott, D. (2007). Level-headed. *Artificial Intelligence*, *171*(18), 1183–1186.
Mishra, A., Bhattacharyya, P., & Carl, M. (2013). Automatically predicting sentence translation difficulty. In *ACL* (pp 346–351).
Mitchell, M. (2019). *Artificial intelligence: A guide for thinking humans*. UK: Penguin.
Moor, J. (2003). *The Turing test: the elusive standard of artificial intelligence* (Vol. 30). New York: Springer Science & Business Media.
Nie, Y., Williams, A., Dinan, E., Bansal, M., Weston, J., & Kiela, D. (2019). Adversarial nli: A new benchmark for natural language understanding. arXiv:191014599.
Nilsson, N. J. (2006). Human-level artificial intelligence? Be serious!. *AI Magazine*, *26*(4), 68.

Oppy, G., & Dowe, D. L. (2011). The turing test. In: Zalta, E. N. (Ed.), *Stanford encyclopedia of philosophy*, Stanford University. http://plato.stanford.edu/entries/turing-test/.
Preston, B. (1991). AI, anthropocentrism, and the evolution of 'intelligence'. *Minds and Machines*, *1*(3), 259–277.
Proudfoot, D. (2011). Anthropomorphism and AI: Turing's much misunderstood imitation game. *Artificial Intelligence*, *175*(5), 950–957.
Proudfoot, D. (2017). The Turing test-from every angle. In J. Bowen, M. Sprevak, R. Wilson, & B. J. Copeland (Eds.), *The Turing Guide*. Oxford: Oxford University Press.
Rahwan, I., Cebrian, M., Obradovich, N., Bongard, J., Bonnefon, J. F., Breazeal, C., et al. (2019). Machine behaviour. *Nature*, *568*(7753), 477–486.
Rajalingham, R., Issa, E. B., Bashivan, P., Kar, K., Schmidt, K., & DiCarlo, J. J. (2018). Large-scale, high-resolution comparison of the core visual object recognition behavior of humans, monkeys, and state-of-the-art deep artificial neural networks. *Journal of Neuroscience*, *38*(33), 7255–7269.
Rajpurkar, P., Jia, R., & Liang, P. (2018). Know what you don't know: Unanswerable questions for squad. arXiv:180603822.
Rozen, O., Shwartz, V., Aharoni, R., & Dagan, I. (2019). Diversify your datasets: Analyzing generalization via controlled variance in adversarial datasets. In *Proceedings of the 23rd conference on computational natural language learning* (CoNLL), Association for Computational Linguistics, Hong Kong, China, pp. 196–205.
Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., et al. (2015). Imagenet large scale visual recognition challenge. *International Journal of Computer Vision*, *115*(3), 211–252.
Sakaguchi, K., Bras, R. L., Bhagavatula, C., & Choi, Y. (2019). Winogrande: An adversarial winograd schema challenge at scale. arXiv:190710641.
Samuel, A. L. (1959). Some studies in machine learning using the game of checkers. *IBM Journal of Research and Development*, *3*(3), 210–229.
Saygin, A. P., Cicekli, I., & Akman, V. (2000). Turing test: 50 years later. *Minds and Machines*, *10*(4), 463–518.
Schlangen, D. (2019). Language tasks and language games: On methodology in current natural language processing research. arXiv:190810747.
Schoenick, C., Clark, P., Tafjord, O., Turney, P., & Etzioni, O. (2017). Moving beyond the Turing test with the Allen AI science challenge. *Communications of the ACM*, *60*(9), 60–64.
Schrimpf, M., Kubilius, J., Hong, H., Majaj, N. J., Rajalingham, R., Issa, E. B., Kar, K., Bashivan, P., Prescott-Roy, J., Schmidt, K., Yamins, D. L. K., & DiCarlo, J. J. (2018). Brain-score: Which artificial neural network for object recognition is most brain-like? bioRxiv preprint.
Schweizer, P. (1998). The truly total Turing test. *Minds and Machines*, *8*(2), 263–272.
Sebeok, T. A., & Rosenthal, R. E. (1981). The clever Hans phenomenon: Communication with horses, whales, apes, and people. *Annals of the NY Academy of Sciences*, *364*, 1–17.
Seber, G. A. F., & Salehi, M. M. (2013). Adaptive cluster sampling. In *Adaptive sampling designs* (pp 11–26). New York: Springer.
Settles, B. (2009). Active learning. Tech. rep., synthesis lectures on artificial intelligence and machine learning. Morgan & Claypool.
Shah, H., & Warwick, K. (2015). Human or machine? *Communications of the ACM*, *58*(4), 8.
Shanahan, M. (2015). *The technological singularity*. New York: MIT Press.
Shoham, Y. (2017). Towards the AI index. *AI Magazine*, *38*(4), 71–77.
Silver, D., Schrittwieser, J., Simonyan, K., Antonoglou, I., Huang, A., Guez, A., et al. (2017b). Mastering the game of Go without human knowledge. *Nature*, *550*(7676), 354–359.
Silver, D., Hubert, T., Schrittwieser, J., Antonoglou, I., Lai, M., Guez, A., Lanctot, M., Sifre, L., Kumaran, D., Graepel T, et al. (2017a). Mastering chess and shogi by self-play with a general reinforcement learning algorithm. arXiv:171201815.
Sloman, A. (2014). Judging chatbots at Turing test. http://www.csbhamacuk/research/projects/cogaff/misc/turing-test-2014html.
Stern, R., Sturtevant, N., Felner, A., Koenig, S, et al. (2019). Multi-agent pathfinding: Definitions, variants, and benchmarks. arXiv:190608291.
Sturm, B. L. (2014). A simple method to determine if a music information retrieval system is a "horse". *IEEE Transactions on Multimedia*, *16*(6), 1636–1644.
Turing, A. M. (1950). Computing machinery and intelligence. *Mind*, *59*, 433–460.
Turing, A. (1952). Can automatic calculating machines be said to think? BBC. BBC Third Programme, 14 and 23 Jan. 1952, between M. H. A. Newman, A. M. T., Sir Geoffrey Jefferson and R. B.

Braithwaite. Reprinted in Copeland, B. J. (ed.) The essential Turing(pp. 494–495). Oxford: Oxford University Press. http://www.turingarchive.org/browse.php/B/6.
Vale, C. D., & Weiss, D. J. (1975). A study of computer-administered stradaptive ability testing. Tech. rep., Minnesota Univ. Minneapolis Dept. of Psychology.
Van Seijen, H., Fatemi, M., Romoff, J., Laroche, R., Barnes, T., & Tsang, J. (2017). Hybrid reward architecture for reinforcement learning. In *NIPS* (pp. 5392–5402).
Vardi, M. Y. (2015). Human or machine? Response. *Communications of the ACM*, *58*(4), 8–8.
Vinyals, O., Ewalds, T., Bartunov, S., Georgiev, P., Vezhnevets, A. S., Yeo, M., Makhzani, A., Küttler, H., Agapiou, J., Schrittwieser, J., et al. (2017). Starcraft ii: A new challenge for reinforcement learning. arXiv:170804782.
von Ahn, L., Blum, M., & Langford, J. (2004). Telling humans and computers apart automatically. *Communications of the ACM*, *47*(2), 56–60.
von Ahn, L., Maurer, B., McMillen, C., Abraham, D., & Blum, M. (2008). RECAPTCHA: Human-based character recognition via web security measures. *Science*, *321*(5895), 1465.
Wainer, H. (2000). *Computerized adaptive testing: A primer* (2nd ed.). Mahwah, NJ: Lawrence Erlabaum Associate Publishers.
Wang, A., Pruksachatkun, Y., Nangia, N., Singh, A., Michael, J., Hill, F., Levy, O., & Bowman, S. R. (2019). Superglue: A stickier benchmark for general-purpose language understanding systems. arXiv:190500537.
Watt, S. (1996). Naive psychology and the inverted Turing test. *Psycoloquy*, *7*(14), 463–518.
Weiss, D. J. (2011). Better data from better measurements using computerized adaptive testing. *Journal of Methods and Measurement in the Social Sciences*, *2*(1), 1–27.
You, J. (2015). Beyond the Turing test. *Science*, *347*(6218), 116–116.
Youyou, W., Kosinski, M., & Stillwell, D. (2015). Computer-based personality judgments are more accurate than those made by humans. *Proceedings of the National Academy of Sciences*, *112*(4), 1036–1040.
Zadeh, L. A. (2008). Toward human level machine intelligence-Is it achievable? The need for a paradigm shift. *IEEE Computational Intelligence Magazine*, *3*(3), 11–22.
Zellers, R., Bisk, Y., Schwartz, R., & Choi, Y. (2018). Swag: A large-scale adversarial dataset for grounded commonsense inference. In *Proceedings of the 2018 conference on empirical methods in natural language processing (EMNLP)*.
Zellers, R., Holtzman, A., Bisk, Y., Farhadi, A., & Choi, Y. (2019). Hellaswag: Can a machine really finish your sentence? arXiv:190507830.
Zhou, P., Khanna, R., Lin, B. Y., Ho, D., Ren, X., & Pujara, J. (2020). Can BERT reason? logically equivalent probes for evaluating the inference capabilities of language models. arXiv:200500782.
Zillich, M. (2012). My robot is smarter than your robot. on the need for a total Turing test for robots. In: V. Muller & A. Ayesh (Eds.), *AISB/IACAP 2012 symposium "revisiting turing and his test"*, The Society for the Study of Artificial Intelligence and the Simulation of Behaviour, pp. 12–15.