Artif Intell Rev
DOI 10.1007/s10462-016-9505-7

Evaluation in artificial intelligence: from task-oriented to ability-oriented measurement

José Hernández-Orallo[1]

Abstract The evaluation of artificial intelligence systems and components is crucial for the progress of the discipline. In this paper we describe and critically assess the different ways AI systems are evaluated, and the role of components and techniques in these systems. We first focus on the traditional *task*-oriented evaluation approach. We identify three kinds of evaluation: human discrimination, problem benchmarks and peer confrontation. We describe some of the limitations of the many evaluation schemes and competitions in these three categories, and follow the progression of some of these tests. We then focus on a less customary (and challenging) *ability*-oriented evaluation approach, where a system is characterised by its (cognitive) abilities, rather than by the tasks it is designed to solve. We discuss several possibilities: the adaptation of cognitive tests used for humans and animals, the development of tests derived from algorithmic information theory or more integrated approaches under the perspective of universal psychometrics. We analyse some evaluation tests from AI that are better positioned for an ability-oriented evaluation and discuss how their problems and limitations can possibly be addressed with some of the tools and ideas that appear within the paper. Finally, we enumerate a series of lessons learnt and generic guidelines to be used when an AI evaluation scheme is under consideration.

Keywords AI evaluation · AI competitions · Machine intelligence · Cognitive abilities · Universal psychometrics · Turing test

1 Introduction

The evaluation of any discipline must necessarily be linked to the purpose of the discipline. What is the purpose of artificial intelligence (AI)? McCarthy's pristine definition of AI sets this unambiguously: "[AI is] the science and engineering of making intelligent machines"

✉ José Hernández-Orallo
jorallo@dsic.upv.es

[1] DSIC, Universitat Politècnica de València, Valencia, Spain

Published online: 19 August 2016

(McCarthy 2007). As a consequence, AI evaluation should focus on evaluating the intelligence of the artefacts it builds. However, as we will further discuss below, 'intelligence tests' (of whatever kind) are not the everyday evaluation approach for AI. The explanation for this is that most AI research is better identified by Minsky's more pragmatic definition: "[AI is] the science of making machines capable of performing tasks that would require intelligence if done by [humans]" (Minsky (1968), p. v). As a result, AI evaluation focuses on checking whether machines do these *tasks* well.

This has led to an important anomaly of AI. AI artefacts solve these tasks *without featuring intelligence*. Paradoxically, this is one of the reasons of AI success. Systems are designed for a particular functionality and perform their task more *predictably* than humans, from driving cars to supply chain planning. Frequently, some tasks are not considered AI problems any more, once they are solved without full-fledged intelligence. This phenomenon is known as the "AI effect" (McCorduck 2004). It would be unfair, however, to deny that some current AI systems, especially those that incorporate some learning potential, exhibit some intelligent behaviour.

Anyway, it is not the purpose of this paper to dig further into the time-worn debate between weak (or soft) AI and strong (or hard) AI, and whether AI should solve problems as humans would do or whether AI has to achieve what is usually referred to as human-level intelligence. We will just focus on the distinction between narrow AI versus general AI, but always recognising that both approaches are valid and genuine parts of AI research. It is useful to have specialised AI systems that solve specific tasks, as well as systems that have abilities so that they can solve new problems they have never faced before. The intention of stressing this duality is that this should necessarily pervade the evaluation procedures in AI. Specialised AI systems should require a task-oriented evaluation, while general-purpose AI systems (also known as AGI systems, from the term 'artificial general intelligence') should require an ability-oriented evaluation. In practice, however, we see that some general-purpose AI systems are evaluated with a narrow set of tasks. Also, some general-purpose *components*, such as planning or learning techniques, are integrated—or re-programmed—into systems with specific goals, or become specialised after many training or interaction trials, losing their plasticity as a result.

This paper focuses on the way evaluation is done in AI. As any science and engineering discipline, measuring is crucial for AI. Disciplines progress when they have objective evaluation tools to measure the elements and objects of study, assess the prototypes and artefacts that are being built and examine the discipline as a whole. As we will discuss in subsequent sections, despite the significant progress in the past couple of decades (with the generalisation of several AI benchmarks and competitions) there still remains a large margin for improvement in the way AI systems are evaluated. This is partially because we do not see AI evaluation as a *measurement process* (Hand 2004). Also, it is probably a crucial moment to overhaul the way AI evaluation is performed, after the recent progress in areas of AI that are distinct from the narrow AI approach, such as developmental robotics (Asada et al. 2009), deep learning (Arel et al. 2010), inductive programming (Gulwani et al. 2015), artificial general intelligence (Goertzel and Pennachin 2007), universal artificial intelligence (Hutter 2007), etc.

By overhauling AI evaluation, we aim to fill a gap, because, to our knowledge, there is no overarching analysis about how evaluation is performed in AI and how it can be improved and adapted to the challenges of the future. Some previous works discussing AI evaluation (Newell and Simon 1976; Gaschnig et al. 1983; Rothenberg et al. 1987; Geissman and Schultz 1988; Decker et al. 1989; Langley 1987, 2011; Buchanan 1988; Simon 1995; Baldwin and Yadav 1995; Falkenauer 1998; Langford 2005; Legg and Hutter 2007a; Whiteson et al. 2011;

Drummond and Japkowicz 2010; Anderson et al. 2011; Madhavan et al. 2009; Schlenoff et al. 2011) are relatively old, non-comprehensive, restrictive to a specific area of AI, limited to one particular approach and/or focused on the experimental methodology rather than what is being measured and how. Nonetheless, we will refer to many of these works in this text. In fact, whereas this paper aims to give a broad coverage of AI evaluation, some of the works above can still be very useful as in-depth analysis for particular domains or approaches.

Some ideas from the old analysis still hold today. For instance, Cohen and Howe (1988) introduced several criteria for evaluating research problems, methods, implementations, experiments' design, and evaluation of the experiments. In the criteria for experiments' design, we see several of the topics we will address in the paper: "1. How many examples can be demonstrated?" (are they sufficient and qualitatively different and illustrative?), "2. Should the program's performance be compared to a standard?", "3. What are the criteria for good performance?", "4. Does the program purport to be general (domain-independent)?" (do the domains being tested constitute a representative class?), and "5. Is a series of related programs being evaluated?". Other statements in (Cohen and Howe 1988) are not so up-to-date, and show that there has been an improvement in AI evaluation. For instance, we found the recommendation "that editors, program committees, and reviewers should begin to insist on evaluation". Today this recommendation has been generalised. For instance, Conrad and Zeleznikow (2013) report that more than 60 % of the published papers in ICAIL (the International Conference on Artificial Intelligence and Law) in 1987 did not have any evaluation in front of 20 % in 2011. A similar trend is seen in journal papers, as the number of empirical papers without evaluation becomes marginal from 2005 and 2014 (Conrad and Zeleznikow 2015). The most widespread book on artificial intelligence also confirms that "AI has advanced more rapidly in the past decade because of greater use of the scientific method in experimenting with and comparing approaches" (Russell and Norvig 2009, p. 30). Hence, a lack of evaluation may no longer be the problem. However, there is still a great deal of disaggregation, many ad-hoc procedures, bad habits and loopholes about what is being measured and how it is being measured. This paper will focus on these issues.

Looking at what is being measured, it seems that we have to distinguish between AI *systems* and AI *components*. Systems (such as AI agents, cognitive architectures or robots) can be evaluated as they are, since they take some sort of problem (by a specification or by rewards) and can be evaluated in terms of a utility function. Components (such as particular techniques, algorithms, methods or tools) cannot be evaluated if there is no specification for the component. Hoever, even in this case, their quality will ultimately be assessed in terms of their functionality as part of one or more systems. For instance, a (self-driving) car can be evaluated as a system but its components (e.g., an engine or camera), even if they fulfil their own specifications perfectly, are ultimately evaluated in the way they serve the whole performance of the car. Actually, a Formula One engine would be inappropriate for a family car, and a monoscopic camera would be inappropriate for some self-driving cars (but not others). Nevertheless, we can still characterise and measure an engine according to several specifications, such as power, consumption and robustness. Similarly, in AI there are some techniques that can be evaluated independently using several dimensions (e.g., a SAT solver), but they really make sense when integrated into a system truly solving one or more problems.

In this paper we will mostly refer to the evaluation of AI systems, as we consider AI components can be evaluated as is usually done in (the rest of) computer science (e.g., analysing their compliance with the specification and their computational complexity). Having said this, in AI there are cases that are arguably half-way between a standalone system

and a component, such as planners, machine learning methods, natural language processing tools, etc. Our main criterion will be to consider those working artefacts that can solve one or more problems without further integration or programming. If there is human intervention once the evaluation starts then we are really evaluating the programmers, integrators or curators, and not only the AI system (or component). A component that is able to solve different problems by reprogramming will not be considered general-purpose. For instance, a machine with a Lisp interpreter where a programmer can code specific applications, such as a chess player or a theorem prover, is not a general-purpose *system*, even if the language is. Otherwise, any Turing-complete programming language or platform would ultimately be considered a general-purpose AI system, which would be absurd in terms of actual capabilities, autonomously.

We will start with a survey of task-oriented evaluation in AI, by far more common in AI research. The notion of performance is relatively easy to determine as it is directly linked to the set or class of problems we are interested in for the evaluation. Nonetheless, we will identify several problems, most of them derived from the confusion of a task definition with its evaluation. An appropriate sampling procedure from the class of problems defining the task is not always easy. We will give some hints to derive better evaluation protocols. With this perspective we will argue that white-box evaluation (by algorithm inspection) is becoming less predominant in AI, and we will devote the rest of the paper to black-box evaluation (by behaviour). We will distinguish three types of behavioural evaluation: by human discrimination (performing a comparison against or by humans), problem benchmarks (a repository or generator of problems) and by peer confrontation (1-vs-1 or multi-agent 'matches'). We will survey some of the competitions and repositories in these three categories, looking at the progression of evaluation tools and the development of performance metrics. We will highlight some problems in how these evaluation schemes are developed and used.

In a second part of the paper, we will pay attention to the more elusive and challenging problem of ability-oriented evaluation. The three types of evaluation seen for task-oriented evaluation are not directly applicable, as we now do not want to evaluate systems for what they do but for what they are able to (learn to) do. In other words, we are looking for signs or indications that show that the system has a certain ability. One idea that has been around since the inception of AI is to use human (or animal) intelligence tests, such as the IQ-tests used in psychometrics. During over a century, psychometrics and comparative psychology have developed rigorous experimental techniques and derived many theories of (the evaluation of) *human* intelligence, identifying several factors empirically, which can be arranged multidimensionally or hierarchically, and usually linked to abilities, such as verbal skills, short-term or long-term memory, spatial skills or general intelligence. Each particular test tries to identify a series of exercises that are representative (necessary and sufficient) for a given ability. We will briefly discuss their use and possible adaptation for the evaluation of AI systems. A quite different, and less anthropocentric, approach is based on algorithmic information theory (AIT), a theory developed in the past decades bridging computation and information theory, where the length and steps taken by an algorithmic solution to a problem are key to analyse its complexity and the patterns it contains. For instance, Kolmogorov complexity, the length of the shortest algorithm describing a sequence, is one of key notions in AIT. Using AIT, problem classes and their difficulty are derived from computational principles. In this way, we are sure about what we are actually evaluating. Also, exercise generators can be derived from first principles. A more unified view integrating different evaluation paradigms and procedures found in many disciplines is also described, known as 'universal psychometrics', linked to the notion of 'universal test', a test that can be applied to any kind of system. We

also see some examples from areas in AI (transfer learning, inductive programming, cognitive architectures and developmental robotics) where general-purpose systems have been evaluated.

Finally, in a third part of the paper, we look at a more gradual view bridging the task-oriented and ability-oriented types of evaluation in terms of task classes that go from specific to general. The discussion also includes some guidelines about how competitions and problem generators can be improved, integrated or overhauled for a more robust and efficient AI system evaluation. This is followed by the conclusions.

2 Task-oriented evaluation

AI is a successful discipline. The range of applications has been greatly enlarged over the years. We have successful applications in computer vision, speech recognition, music analysis, machine translation, text summarisation, information retrieval, robotic navigation and interaction, automated vehicles, game playing, prediction, estimation, planning, automated deduction, expert systems, etc. (see, e.g., Russell and Norvig 2009). Most of these application problems are specific. This implies that the goals are clear and that researchers can focus on the problem. This does not mean that we are not allowed to use more general principles, components and techniques to solve many of these problems, but that the task is sufficiently specific so that systems can be specialised for these tasks. For instance, robotic navigation of a Mars rover can share some of the techniques with a driverless car on Earth, but the final application is extremely specialised in both cases.

This specialisation leads to an application-specific (task-oriented) evaluation. In fact, going from an abstract problem to a specific task is encouraged: "refine the topic to a task", provided it is "representative" (Cohen and Howe 1988). Given a precise definition of the task, we only need to define a notion of performance from it. Clearly, we measure performance, and not intelligence. In fact, many of the most successful AI systems solve each problem in a way that is different from the way humans solve the same problem. Also, AI systems usually include a great amount of built-in programming and knowledge for the task. It is not unfair to say that we evaluate the researchers that have designed the system rather than the system itself. For instance, we can say that it was the research team after Deep Blue (Campbell et al. 2002) (with the help of a powerful computer) who actually defeated Kasparov. Things have changed significantly in the way AlphaGo (Silver et al. 2016) defeated Lee Sedol, the world's top Go player at the time, but the system is still strongly specialised for the game, although the components used (deep neural networks and reinforcement learning) are general-purpose.

Disregarding who is praiseworthy for each new successful application, AI systems that address specialised problems with a clear performance should be easy to evaluate. The reality is not that straightforward, mostly because there are many different (and usually ad-hoc) evaluation approaches. Let us examine them.

2.1 Types of performance measurement in AI

An application domain, as described above, can be characterised by a set of problems, tasks or exercises M. In order to evaluate each exercise $\mu \in M$ we can get a measurement $R(\pi, \mu)$ of the performance of system π. Measurements can be imperfect. Also, the system, the problem or the measurement may be non-deterministic. As a result, it is usual to work with the expected value of the performance of π as $\mathbb{E}[R(\pi, \mu)]$.

The definition of M and R does not specify how we want to aggregate the results when M has more than one problem. The most common approaches are[1]

– Worst-case performance[2]:

$$\Phi_{min}(\pi, M) = \min_{\mu \in M} \mathbb{E}[R(\pi, \mu)] \tag{1}$$

– Best-case performance:

$$\Phi_{max}(\pi, M) = \max_{\mu \in M} \mathbb{E}[R(\pi, \mu)] \tag{2}$$

– Average-case performance:

$$\Phi(\pi, M, p) = \sum_{\mu \in M} p(\mu) \cdot \mathbb{E}[R(\pi, \mu)] \tag{3}$$

where p is a probability distribution on M.

It is assumed that the magnitudes of R for different $\pi \in M$ are commensurate. For instance, if R can range between 0 and 1 for problem μ_1 but ranges between 0 and 10,000 for problem μ_2, the latter will have a much higher weight and will dominate the aggregation. This is not necessarily wrong, e.g., if they are measured with the same unit (e.g., euros). In general, however, R is a construct that needs to be normalised. The choice of a performance metric that is sufficiently normalised such that the results are commensurate is not always easy, but possible to some extent (see, e.g., Whiteson et al. 2011).

At this point, it is pertinent to make a comment about the well-known no-free-lunch (NFL) theorems (Wolpert and Macready 1995; Wolpert 1996, 2012), as these theorems are usually misunderstood. These theorems state that given all possible problems, *under some particular distributions*, no method can work better than any other on average. The argument to support this interpretation is that, considering all problems, if method π_A is better than method π_B for one problem then π_B will be better than π_A for another problem. Some people have even interpreted that research in AI (including search and optimisation problems in computer science) is futile. However, the NFL theorems can only be applied when the assumptions hold. The conditions state that M must be infinite and include all possible problems. Also, the problems can be shuffled without affecting the probability, which can be expressed as "block uniformity" (Igel and Toussaint 2005), for which the uniform distribution would be a special case. Nonetheless, these conditions are not plausible if the problems are taken from the real world. It is unrealistic to assume that the problems we face are taken from a series of random bits, or that a problem, and its opposite problem (whatever it is) are equally probable. Many other distributions are much more plausible. A universal distribution (Solomonoff 1964; Li and Vitányi 2008), e.g., which is consistent with the idea that problems are generated by physical laws, processes, living creatures, etc., states that random (incompressible) problems are less likely. So, for many distributions p, the conditions of the NFL do not hold and we find that there can be methods π_A and π_B such that: $\Phi(\pi_A, M, p) > \Phi(\pi_B, M, p)$. In fact, there can be optimal methods for inductive inference (Lattimore and Hutter 2013), some free

[1] Worst-case performance and best-case performance are special cases of a rank-based aggregation (using the cumulative distribution of results), with other possibilities such as the median, the first decile, etc. Rank-based aggregation, especially worst-case performance, is more robust to systems getting good scores on many easy problems but doing poorly on the difficult problems.

[2] Note that this formula does not have the size of the instance as a parameter, and hence it is not comparable to the usual view of worst-case analysis of algorithms.

lunches for co-evolution (Wolpert and Macready 2005), and other areas, although it seems that for optimisation the free lunches are very small (Everitt et al. 2014).

After this clarification, it is relevant to determine how R is going to be obtained. For relatively simple solutions, we can analyse the code or the algorithm of the system π. If the code can be well understood then its computational properties and behaviour can be clearly determined. We use the term 'white-box' evaluation when R is inferred through program inspection or algorithm analysis. White-box evaluation is powerful because we can obtain R theoretically for a given agent π and a problem class M (provided both are defined theoretically). One common type of problem that is evaluated with a white-box approach takes place when the solution to the problem has to be correct or optimal (i.e., perfect). In this case, the performance metric R is defined in terms of time and/or space resources. This is the case of classical computational complexity theory. Worst-case analysis (Eq. 1) is more common than average-case analysis (Eq. 3), although the latter has also become popular recently (Knuth 1973; Levin 1986; Goldreich and Vadhan 2007). Nonetheless, many AI problems are so challenging nowadays that perfect solutions are no longer considered as a constraint. Instead, approximate solvers are designed to optimise a performance metric that is defined in terms of the level of *error* of the solution and the time and/or space resources. In this case, the use of an average-case analysis is more common, although worst-case analysis (Eq. 1) can also be studied under some paradigms (e.g., Probably Approximately Correct learning, Valiant 1984). In agent theory, the behaviour of the agent (and its properties) can be analysed under some paradigms such as Belief-Desire-Intention (BDI) agents (see, e.g., a testability approach, Winikoff and Cranefield 2014). The theoretical analysis of 'white-box' evaluation has also been applied to games. For instance, in board games, algorithms can be derived and analysed whether they are optimal, such as noughts and crosses (tic-tac-toe) and English draughts (checkers), the latter solved by Jonathan Schaeffer (Schaeffer et al. 2007). Finally, in game theory, the expected pay-off plays the role of R and optimal strategies can be determined for some simple games, as well as equilibria and other properties. In games, some results can be obtained independently of the opponent, but others are only true if we also know the algorithm that the other players are using (so it becomes a double 'white-box' approach to evaluation).

As AI systems become more sophisticated, white-box assessment becomes more difficult, if not impossible, because the unpredictability of complex systems. Many AI systems incorporate many different techniques and have stochastic behaviours. This is also in agreement with a view of AI as an experimental science (Buchanan 1988; Simon 1995). As a result, a black-box approach is taken.[3] This means that R is obtained exclusively from the behaviour of the system in an empirical way. In this case, average-case evaluation is usual.[4]

There are many kinds of black-box assessment in AI, but we can group them into three main categories:

– Human discrimination: the assessment is made by and/or against humans through observation, scrutiny and/or interview. Although it can be based on a questionnaire or a procedure, the assessment is usually informal and subjective. In AI, this kind of evaluation is not very usual, except for the Turing test and variants, as we will discuss later

[3] The distinction between white and black box can be enriched to consider those problems where the solution must be accompanied by a verification, proof or explanation (Hernández-Orallo 2000b; Alpcan et al. 2014).

[4] Although it is not uncommon, as we will see, that the set of problems from M are chosen by the research team that is evaluating its own method, so the probability to choose from M can be biased in such a way that it is actually a best-case evaluation.

on, despite being more common in other disciplines dealing with behaviour, such as psychology, ethology and comparative psychology.
- Problem benchmarks: the assessment is performed against a collection or repository of problems (M). This approach is very frequent in AI, where we have problem libraries, repositories, corpora, etc. Unlike other areas where this approach is also common (such as psychology and comparative psychology), most tests and repositories in AI are *public*. However, as the public access to the benchmark before the evaluation can lead to "evaluation overfitting", there have also been some occasional evaluations in AI following a "secret generalized methodology" (Whiteson et al. 2011). For instance, M can be generated in real time using a problem generator, which actually defines M and p.
- Peer confrontation: the assessment is performed through a series of (1-vs-1 or n-vs-n) matches. The result is relative to the other participants. Given this relative value, in order to allow for a numerical comparison, sophisticated performance metrics can be derived (e.g., the Elo system in chess, Elo 1978). Whereas this is the common approach in several domains such as games and multi-agent systems, other AI domains can use this format, especially if systems are evaluated according to the best one in terms of resources or accuracy, in a competitive way, or when the evaluation is set up as a challenge, because it is difficult to give a score but the best of two systems can still be objectively determined.

The combination of some of the above is also common for evaluation. In addition, there are domains in AI that can be evaluated with the three types above. For instance, common-sense reasoning can be analysed by human discrimination (interviewing), benchmarks (comprehension tests) and confrontation (competitions such as Jeopardy!, a TV quiz, Ferrucci et al. 2010). In what follows, we analyse each of the three categories in more detail.

2.2 Evaluation by human discrimination

In this first category we include the evaluation approaches that are performed by a comparison with or by humans. The Turing test (Turing 1950; Oppy and Dowe 2011) is a case in which there is both comparison against humans and evaluation by human judges. While the 'imitation game' was introduced by Turing as a philosophical instrument in his response to nine objections against machine intelligence, the game has been (mis-)understood as an actual test, the Turing test, ever since, with the standard interpretation of one human, one machine pretending to be a human, and a human interrogator through a teletype acting as a judge. The latter must tell which one is the machine and the human.

Not only has the game been taken as an actual test, but it has had several implementations, such as the Loebner Prize,[5] held every year since 1991. Despite the criticisms of how this prize is conducted and its interpretation through the years, there have been more implementations. In 2014, Kevin Warwick organised a similar competition that took place at the Royal Society in London. Even if the results were not significantly different to previous results of the Loebner Prize (or even what Weizenbaum's ELIZA was able to do 50 years ago, Weizenbaum 1966), the over-reaction and publicity of this outcome were preposterous. The reputation of the implementations of the Turing test was (further) stained with statements such as this: "If a computer is mistaken for a human more than 30 % of the time during a series of 5 min keyboard conversations it passes the test. No computer has ever achieved this, until now. Eugene managed to convince 33 % of the human judges (30 judges took part […]) that it was human." (Warwick 2014). And Warwick went on: "We are therefore proud to declare that

[5] http://www.loebner.net/Prizef/loebner-prize.html.

Alan Turing's Test was passed for the first time. […] This milestone will go down in history as one of the most exciting".

Is the imitation game a valid test? Even assuming that the times and thresholds are stricter than the previous incarnations, the Turing test has many problems as an intelligence test. First, it is a test of humanity, relative to human characteristics (i.e., anthropocentric). It is neither gradual nor factorial and needs human intervention (it cannot be automated). If done properly, it may take too much time. Even so, as we have seen, it can be gamed by non-intelligent chatterbots. As a result, the Turing test is neither a sufficient nor a necessary condition for intelligence. Despite the criticism, the Turing test still has many advocates (Proudfoot 2011). It is also an inspiration for countless philosophical debates and has led to connections with other concepts in AI or computation (Hernández-Orallo et al. 2012b).

In any case, Turing is not to be blamed by a failure of the Turing test as a useful test to evaluate AI systems. Turing did not conceive the test as a practical test to measure intelligence up to and beyond human intelligence. He is not to blame for a philosophical construct that has had a great impact in the philosophy and understanding of machine intelligence, but a negative impact on its measurement.

Does this mean that we should discard the idea of evaluating AI systems by human judges or by comparing with humans? Not at all. Recently, there have been many variants of the Turing test: Total Turing tests (Schweizer 1998), Visual Turing tests including sensory information, Toddler Turing tests (Alvarado et al. 2002), robotic interfaces, virtual worlds, etc. (Mueller and Minnery 2008; Hingston 2010). These may be useful for chatterbot evaluation, personal assistants, robots and videogames. For instance, it is within the area of videogames where the notion of 'believability' has appeared, which is understood as the property of a bot of looking 'believable' as a human (Livingstone 2006; Hingston 2012). This term is interesting, as it clearly detaches these tests from the evaluation of intelligence. In videogames, there are applications where we want bots that can fool opponents into thinking that they are just another human player. Other highly subjective properties may also be of interest: enjoyability, resilience, aggressiveness, fun, etc.

Finally, there is a kind of test that is related to the Turing test, the so-called CAPTCHA (Completely Automated Public Turing test to tell Computers and Humans Apart) (von Ahn et al. 2004; Ahn et al. 2008). It is said to be a 'reverse Turing test' because the goal is to tell computers and humans apart in order to ensure that an action or access is only performed by a human (e.g., making a post, registering in a service, etc.). CAPTCHAs are quick and practical, omnipresent nowadays. However, they are designed according to the tasks that are solved by the current state of AI technology. At present, for instance, a common CAPTCHA is a series of distorted letters, which are usually easy to recognise by humans but not by machines (e.g., current OCR systems struggle). Logically, when character recognition systems and other techniques improve, current CAPTCHAs are broken (see, e.g., Bursztein et al. 2014), and CAPTCHAs need to be updated to more distorted words or to other tasks that are beyond AI technology. Similarly, the detection of bots in social networks (sybils) and crowdsourcing platforms rely on tests that are variants of CAPTCHAs, the Turing test, or the observation and analysis of user profiles and behaviour (Chu et al. 2010; Wang et al. 2012).

Table 1 includes a selection of evaluation schemes under the human-discrimination category. As it is not possible to go into the details of all of them because of brevity, let us choose some that are most representative. Of particular interest is the BotPrize competition, which has been held since 2008. This contest awards the bot that is deemed more believable (playing like a human) by the other (human) players. The competition uses a first-person shooter videogame, the DeathMatch game type, as used in Unreal Tournament 2004. It is important to clarify that the bots do not process the image but receive a description of it through textual

Table 1 List of some evaluation schemes in the human-discrimination category

Evaluation scheme	Description
Loebner prize[a]	General Turing test implementation
University of reading TT 2014[b]	Occasional Turing test implementation (Kevin Warwick)
BotPrize[c]	Contest about bot believability in videogames (Livingstone 2006; Hingston 2012)
Robo chat challenge[d]	Chattering bots competition
CAPTCHAs[e]	Spotting bots in applications requiring humans (von Ahn et al. 2004; Ahn et al. 2008)
Humies awards[f]	Human-competitive results using genetic and evolutionary computation (Koza 2010)
Graphics turing test	Computer-generated virtual world versus a real camera (McGuigan 2006; Borg et al. 2012)

[a] http://www.loebner.net/Prizef/loebner-prize.html
[b] http://www.reading.ac.uk/news-and-events/releases/PR583836.aspx
[c] http://botprize.org/
[d] http://www.robochatchallenge.com/
[e] http://www.captcha.net/
[f] http://www.human-competitive.org

messages in a specific language through the GameBots2004 interface (Pogamut). For the competition, chatting is disabled (as it is not a chatbot competition). There is a "judging gun" and the human judges also play, trying to play normally (a prize for the judges exists for those that are considered more "human" by other judges).

Some questions have been raised about how well the competition evaluates the believability of the participants. For instance, believability is said to be better assessed from a third-person perspective (judging recorded video of other players without playing) than with a first-person perspective (Togelius et al. 2012). The reason is that third-person human judges can concentrate on judging instead on not being killed or aiming at high scores. Actually, this third-person perspective was included in the 2014 competition using a crowdsourcing platform (Llargues-Asensio et al. 2014) so that the two judging systems were incorporated: the First-Person Assessment (FPA), using the BotPrize in-game judging system, and the Third-Person Assessment (TPA), using a crowdsourcing platform. Another issue that could be considered in the future is a richer (and more challenging) representation of the environment, closer to the way humans perceive the images of the game (such as the graphical processing required for the Arcade Learning Environment Bellemare et al. 2013) or the General Video Game Competition (Schaul 2014) we will mention later on. Like the Turing test, the more time the system is evaluated the more accurate the evaluation can be. Nevertheless, in BotPrize, repetitions are used, because each game can lead to different situations according to some random components of the game that are not controlled by the player or the judges. Because of this, in the 2014 competition, each player was judged around 25 times. Finally, about the results, we see how brittle the notion of humannesss or believability can be. The most human bot got a humanness score of 52.2 % while the most human human just got slightly better with a score of 53.3 %.

There are other competitions and awards to evaluate the progress of a domain or area in AI that are evaluated by a human committee, with a loose set of criteria about how a system is to be evaluated. For instance, the Humies awards (Koza 2010), also shown in Table 1,

grant a prize to those methods showing replicable results that are able to solve a problem in a "human-competitive" way, "a long-standing problem for which there has been a succession of increasingly better human-created solutions". The evaluation is performed by a judging committee, who shortlists a few finalists that have to present their results. Accordingly, it is not exactly AI systems what are evaluated but AI researchers. Nonetheless, this allows for the evaluation of components and new tools if these are shown to solve several "human-competitive" problems with them.

Finally, as a summary of the limitations and potentials of the human-discrimination category, we first acknowledge that some variants are being useful, based on the advantage that the intelligence and expertise of the evaluator can be used in a less strict way than other kinds of evaluation. However, the format differs significantly from a standard Turing test. For instance, the human-discrimination approach to evaluation can be just solved by a more traditional interview format with a procedure or storyline (as in psychology interviews), or by an evaluation through observation (using a committee of dedicated judges). This casts doubts about whether evaluation by imitation using the standard interpretation of the Turing test is practical for task-oriented evaluation in AI. It is the concept that is useful, and deserves being adapted to several applications, where the proper setting for observation, interaction and interrogation has to be analysed in order to have an accurate and practical assessment.

2.3 Evaluation through problem benchmarks

In this very common approach to evaluation, M is defined as a set of problems. This fits Eq. 3 perfectly. Necessarily, the quality of the evaluations depends on M and how exhaustively this set is explored. There are other issues that could compromise the quality of the measurement. For instance, when M is a *public* problem repository and is not very large, we find that the systems can specialise for M. Also, the solutions may also be available beforehand, or can be inferred by humans, so the systems can embed part of the solutions. In fact, a system can succeed in a benchmark with a small size of M by using a technique known as the "big switch", i.e., the system recognises which problem is facing and uses the hardwired solution for that specific exercise. Things can become worse if the selection of examples from M is made by the researchers themselves (e.g., the usual procedure in machine learning of selecting 10 or 20 datasets from the UCI repository, the University of California Irvine Machine Learning Repository, Bache and Lichman 2013, as we will discuss below). In general, the size of M and a bona fide attitude to research somewhat limit these concerns. Nonetheless, it is generally acknowledged that most systems actually embed what the researchers have learnt from M. In a way, again, these benchmarks actually evaluate the researchers, not their systems.

The above-mentioned problem is known as 'evaluation overfitting' (Whiteson et al. 2011), 'method overfitting' (Falkenauer 1998) or "clever methods of overfitting" (Langford 2005). For instance, we can evaluate a self-driving cars in a small parking lot or a restricted part of a city, and we may get one car with excellent performance in this area in particular. To avoid or reduce this problem, it is much better if M is very large or infinite, or at least the problems are not disclosed until evaluation time (the part of the city, or even the city, is not known in advance). Problem generators are an alternative. However, it is not always easy to generate a large M of realistic problems (e.g., in a car driving domain). Generators can be based on the use of some prototypes with parameter variations or distortions. These prototypes can be "based on reality", so that the generator "takes as input a real domain, analyses it automatically and generates deformations […] that follow certain high-level characteristics" (Drummond and Japkowicz 2010). More powerful and diverse generators can be defined by the use of problem representation languages. A general and elegant approach is to determine

a probabilistic or stochastic generator (e.g., a grammar) of problems, which directly defines the probability p for the average-case performance Eq. 3. Nonetheless, it is not easy to make a generator that can rule out unusable or Frankenstein-like problems. As an alternative, when a generative model is inappropriate, virtual simulators inspired in real life can be used (Vázquez et al. 2014).

When the set of problems is large or generated, we clearly cannot evaluate AI systems efficiently with the whole set M. So we need to do some sampling of M. It is at this point when we need to distinguish the benchmark or problem definition from an effective evaluation. Assume we have a limited number of exercises n that we can administer. The goal will be to reduce the variance of the measurement given n. One naive approach is to sort M by decreasing p and evaluate the system with the first n exercises. This maximises the accumulated mass for p for a given n. One problem about this procedure is that it is highly predictable. Systems will surely specialise on the first n exercises. For instance, in the self-driving car domain, all systems would specialise for the most important routes, which are probably a few motorways and city avenues. Also, this approach is not very meaningful when R is non-deterministic and/or not completely reliable. Repeated testing may be necessary, which raises the question of whether to explore a higher n or to perform more repetitions.

Random sampling using p seems to be a more reasonable alternative. As said above, if R is non-deterministic and/or subject to measurement error, then random sampling can be with replacement. If M and p define the benchmark, is probability-proportional sampling on p the best way to evaluate systems? The answer is no, in general. Again, in the self-driving car domain, we would probably have the cars evaluated on important routes only. There are better ways of approximating Eq. 3. The idea is to sample in such a way that the diversity of the selection is increased. For instance, for cars, all kinds of roads and streets should be considered. This 'diversity-driven sampling" is related to several kinds of sampling, such as importance sampling (Srinivasan 2002), stratified sampling (Cochran 2007) and other forced Monte Carlo procedures. The key issue is that we use a *different* probability distribution for sampling. Although there are many ways of obtaining a 'diverse' sample, we just highlight two main approaches that can be useful for AI evaluation:

- Information-driven sampling: assume that we have a similarity function $sim(\mu_1, \mu_2)$, which indicates how similar (or correlated) exercises μ_1 and μ_2 in M are. In this case, we need to sample on M such that the accumulated mass on p is high and that diversity is also high. The rationale is that if μ_1 and μ_2 are very similar, using one of them can 'fill the gap' of the other, and we can assume as if both μ_1 and μ_2 had been explored, actually accumulating $p(\mu_1) + p(\mu_2)$. One possible way of doing this is by *stratified sampling* using clustering (not to be confused with cluster sampling). Information-driven sampling suffers from the need of defining the similarity function sim. An alternative is to derive m features that describe the exercises, so creating an m-dimensional space where distances and other topological information can be used to support the notion of diversity (and performing clustering). For instance, if we want to evaluate routes for self-driving cars, we might cluster a database of routes by their distance and their traffic density. If five clusters of a minimum cardinality are found, we can just sample a few routes from each cluster. An example of this procedure is shown in Fig. 1 (left).
- Difficulty-driven sampling. A set M can contain very easy and very challenging problems. Using easy problems for good systems or difficult problems for bad systems is not very optimal. The idea to optimise the evaluation is to choose a range of difficulties for which the evaluation results may be informative (or to give higher probability to exercises inside this range), as in Fig. 1 (right). This procedure is done to a greater or lesser degree in

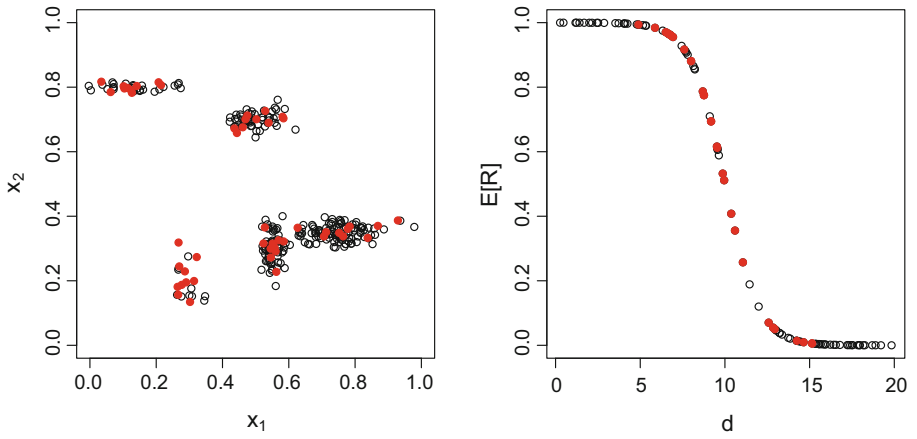

Fig. 1 *Left* a figurative repository M with $|M| = 300$ exercises shown with *empty black circles*. Two features x_1 and x_2 are used to describe the most relevant characteristics of the exercises (according to diversity). These features are used to cluster them into five groups. Next, stratified sampling using these clusters is performed with a sample size of $n = 50$. Clusters are of different size (60, 20, 70, 110, 30) but 10 samples (shown in *solid red circles*) are taken from each cluster. Because of the constant number of examples per cluster, in order to estimate Φ, measurements for under-represented clusters are multiplied by their size. *Right* a repository of $|M| = 100$ exercises. A measure of difficulty d has been derived that is monotonically decreasing with (estimated) expected performance (for a group of agents or for the problem overall). Only $n = 30$ exercises are sampled in the area where the results may be most informative. (Color figure online)

many evaluations and benchmarks in AI. In fact, more challenging problems are usually added over the years, as the systems are able to solve the easy problems (which soon become 'toy problems'). One of the crucial points of difficulty-driven sampling is the definition of a difficulty function $d : M \rightarrow \mathbb{R}^+$. Ideally, we would like that for every $\pi, \Phi(\pi, \mu_1, p) > \Phi(\pi, \mu_2, p)$ iff $d(\mu_1) < d(\mu_2)$. In practice, this condition is too strong, and more flexible characterisations are expected, such as that for every π, and two difficulties a and b such that $a \leq b$ we have that $\Phi(\pi, M_a, p) \geq \Phi(\pi, M_b, p)$ (where M_a denotes all the exercises in M of difficulty a). This could still be too strong and we may use a relaxed version such that for every π, there is a t such that for all a and $b \geq a + t$: $\Phi(\pi, M_a, p) \geq \Phi(\pi, M_b, p)$. In experimental sciences, we have a population-based view of difficulty such that $d(\mu)$ is monotonically decreasing on $\mathbb{E}_{\pi \in \Omega}[\Phi(\pi, \mu, p)]$, where Ω is a population of subjects, agents or systems that are evaluated for the same problem class. In fact, Item Response Theory (Embretson and Reise 2000) in psychometrics follows this approach. Finally, we can derive the difficulty of a problem as a function of the complexity of the problem itself. The complexity metric can be specific to the application, such as the complexity for mazes in (Bagnall and Zatuchna 2005; Zatuchna and Bagnall 2009) or grid-world domains in (Sturtevant 2012), or it can be a more general approach (e.g., Kolmogorov complexity). Note that some of the definitions of difficulty above would not be possible for a set M and distribution p if the conditions of the NFL theorem held.

Both the information-driven sampling and the difficulty-driven sampling can be made adaptive (Seber and Salehi 2013), common in population surveys and many experimental sciences. However, when evaluating performance, it is difficulty-driven sampling that has been used more systematically in the past, especially in psychometrics. In psychometrics, difficulty is inferred from a population of subjects (in the case of AI, this could be a set of

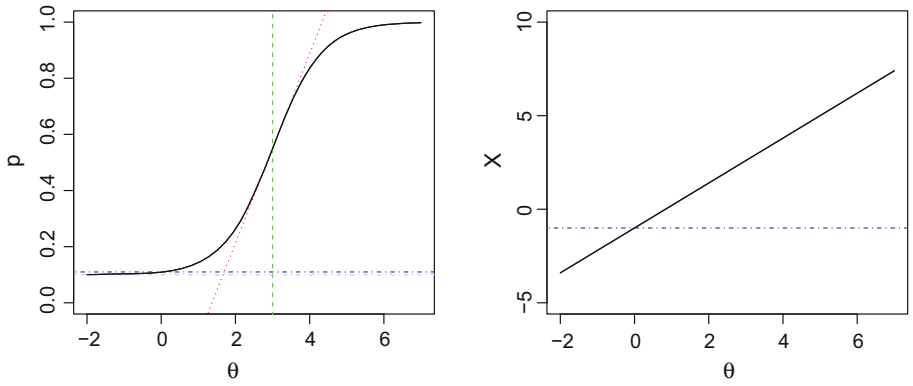

Fig. 2 *Left* item response function (or curve) for a binary score item with the following parameters for the logistic model: discrimination $a = 1.5$, item location $b = 3$, and chance $c = 0.1$. The discrimination is shown by the slope of the curve at the midpoint: $a(1 - c)/4$ (in *dotted red*), the location is given by b (in *dashed green*) and the chance is given by the *horizontal line* at c (in *dashed-dotted grey*, at 0.1), which is very close to the zero-proficiency expected result $p(\theta) = z$ (here shown in *dashed-dotted blue*, at 0.11). *Right* a linear model for a continuous score item with parameter $z = -1$ and $\lambda = 1.2$. The *dashed-dotted line* shows the zero-ability expected result. (Color figure online)

solvers or algorithms). Instead of difficulty, items are analysed by proficiency, represented by θ, a corresponding concept to difficulty from the point of view of the solver (higher problem difficulty requires higher agent proficiency).

Item response theory (IRT) (Embretson and Reise 2000) estimates mathematical models to infer the associated probability and informativeness estimations for each item. When R is discrete or bounded, one very common model is the three-parameter logistic model, where the item response function (or curve) corresponds to the probability that an agent with proficiency θ gives a correct response to an item. This model is characterised as follows:

$$p(\theta) \triangleq c + \frac{1 - c}{1 + e^{-a(\theta - b)}}$$

where a is the *discrimination* (the maximum slope of the curve), b is the *difficulty* or item location (the value of θ leading to a probability half-way between c and 1, i.e., $(1 + c)/2$), and c is the chance or asymptotic minimum (the value that is obtained by *random* guess, as in multiple choice items). The zero-ability expected result is given when $\theta = 0$, which is exactly $z = c + \frac{1-c}{1+e^{ab}}$. Figure 2 (left) shows an example of a logistic item response curve.

For continuous R, if they are bounded, the logistic model above may be appropriate. On other occasions, especially if R is unbounded, a linear model may be preferred (Mellenbergh 1994; Ferrando 2009):

$$X(\theta) \triangleq z + \lambda\theta + \epsilon$$

where z is the intercept (zero-ability expected result), λ is the loading or slope, and ϵ is the measurement error. Again, the slope λ is positively related to most measures of discriminating power (Ferrando 2012). Figure 2 (right) shows an example of a linear item response curve.

Working with item response models is very useful for the design of tests, because if we have a collection of items, we can choose the most suited one for the subject (or population) we want to evaluate. According to the results that the subject has obtained on previous items, we may choose more difficult items if the subject has succeeded on the easy ones, we may

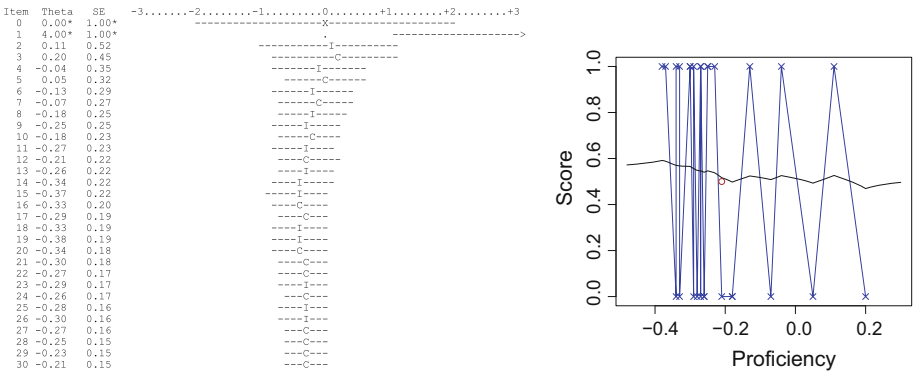

Fig. 3 An example of an IRT-based adaptive test (freely adapted from (Weiss 2011, Fig. 8). *Left* the process and proficiencies (*thetas*) used until convergence. The final proficiency calculated by the test was −0.21 with a standard error of 0.15. *Right* the results shown on a plot. The *black curve* shows a Euclidean kernel smoothing with a constant of 0.1

look for those items that are most discriminating (i.e., most informative) in the area we have doubts, etc. Note that discrimination is not a global issue: a curve may have a very high slope at a given point, so it is highly discriminating in this area, but the curve will almost be flat when we are far from this point. Conversely, if we have a low slope, then the item covers a wide range of difficulties but the result of the item will not be so informative as for a higher slope.

Figure 3 shows an example of an adaptive test using IRT. The sequence of exercise difficulties is shown on the left. The plot on the right shows that averaging the results (especially here, as the outcome of R is discrete, either 0 or 1) makes the estimation of Φ more difficult with a non-adaptive test. These tools, although generally unknown in AI research, could be very useful for its evaluation in the future.

Table 2 includes a selection of evaluation schemes in the problem benchmarks category. We see the variety of repositories, challenges and competitions. As it is impossible to survey all of them in detail, we will focus on some of them, such as the ICAPS (International Conference on Automated Planning and Scheduling) competitions, the evaluation in the area of machine learning (including data mining, data science and KDD challenges), reinforcement learning and videogames.

The ICAPS competitions around the areas of planning and scheduling have taken place since they were started by Drew McDermott in 1998. The recent planning competitions (as for the 2014 edition, Vallati et al. 2015) feature several tracks, for deterministic, learning, continuous probabilistic and discrete probabilistic domains. The original deterministic track now contains many domains (both sequential and temporal). The actual domains used for the evaluation are only disclosed during the evaluation, which in some recent editions has been done with the organisers running the planners themselves instead of the participants. Since all participatns use the same machine, the sole criterion is set that the problem must be (maybe optimally, depending on the subtracks) solved within a given amount of time. In the learning version the planners can take advantage of previous problems of the same domain, using random variants generated with the use of a distribution for each domain. Repetitions are used to account for variations in the actual computation resources (e.g., cloud platforms). The probabilistic versions have different performance metrics and a more reduced set of domains, but the general principles of the competition are similar.

In machine learning and related areas (data mining and data science), because of its recent relevance, the number of repositories and competitions is growing incessantly. Let us start our

analysis with one of the most widespread repositories in computer science, the UCI machine learning repository (Bache and Lichman 2013). Most of the discussion below is applicable to other repositories and, to some extent, to competitions and challenges in machine learning.

The UCI repository includes many supervised (classification and regression) and some unsupervised datasets. The repository is publicly available and is regularly used in machine learning research. The repository is a multi-domain collection of datasets that has been very useful in the development of several machine learning techniques in the past two decades, such as ensemble methods (Zhou 2012) or meta-learning (Brazdil et al. 2008), where the evaluation over multiple datasets was crucial to compare the improvements of new algorithms, parameters and tools.

The usage procedure, which is referred to as "The UCI test" (Macià and Bernadó-Mansilla 2014) or the "de facto approach" (Drummond and Japkowicz 2010; Japkowicz and Shah

Table 2 List of some evaluation schemes in the problem-benchmarks category

Evaluation scheme	Description
CADE ATP system competition[a]	Theorem proving (Sutcliffe and Suttner 2006; Sutcliffe 2009)
Termination competition[b]	Termination of term rewriting programs (Marché and Zantema 2007)
The reinforcement learning competition[c]	Reinforcement learning (Whiteson et al. 2010; Dimitrakakis et al. 2014)
Syntax-guided synthesis competition[d]	Program synthesis (Alur et al. 2013)
International Aerial Robotics Competition[e]	Pilotless aircraft competition
DARPA grand challenge[f]	Autonomous ground vehicles
DARPA urban challenge[g]	Driverless vehicles
DARPA cyber grand challenge[h]	Computer security
DARPA save the day[i]	Rescue Robotic challenge (Jacoff et al. 2003)
The planning competition[j]	Planning (Long and Fox 2003; Vallati et al. 2015)
UCI[k] and KEEL[l] repositories	Machine learning (Bache and Lichman 2013; Alcalá et al. 2010)
PRTools[m]	Pattern recognition problem repository
KDD-cup challenges[n] and kaggle[o]	Machine learning and data mining competitions.
Challenges in machine learning[p]	Miscellaneous machine learning challenges
Plagiarism detection[q]	Plagiarism, authorship and social software misuse (Potthast et al. 2013)
The General Video Game Competition[r]	General video game players (Schaul 2014; Perez et al. 2015)
Hutter prize[s] and related benchmarks[t]	Text compression
Pedestrian benchmarks	Pedestrian detection (Gerónimo and López 2014)
ImageClef, language image retrieval[u]	Cross-language annotation and retrieval of images (Caputo et al. 2014)
Europarl[v], SE times[w], the euromatrix[x]	Machine translation corpora (Starkie et al. 2006)
NIST OpenMT and DARPA TIDES MT[y]	Automatic translation between human languages
Linguistic data consortium corpora[z]	NLP corpora
Angry birds AI Competition[aa]	Angry birds video game

Table 2 continued

Evaluation scheme	Description
The Arcade Learning Environment[ab]	Atari 2600 video games (reinforcement learning) (Bellemare et al. 2013)
GP benchmarks[ac]	Genetic programming (McDermott et al. 2012; White et al. 2013)
Pathfinding benchmarks[ad]	Gridworld domains (mazes) (Sturtevant 2012)
FIRA HuroCup[ae]	Humanoid robot competitions (Anderson et al. 2011)

[a] http://www.cs.miami.edu/~tptp/CASC/
[b] http://termination-portal.org/wiki/Termination_Competition_2014
[c] http://www.rl-competition.org/
[d] http://www.sygus.org/
[e] http://www.aerialroboticscompetition.org/
[f] http://archive.darpa.mil/grandchallenge04/index.htm
[g] http://archive.darpa.mil/grandchallenge/
[h] http://www.cybergrandchallenge.com/
[i] http://www.theroboticschallenge.org/
[j] http://ipc.icaps-conference.org/
[k] http://archive.ics.uci.edu/ml/
[l] http://sci2s.ugr.es/keel/datasets.php
[m] http://prtools.org/
[n] http://www.sigkdd.org/kddcup/index.php
[o] http://www.kaggle.com/
[p] http://www.chalearn.org/
[q] http://pan.webis.de/
[r] http://www.gvgai.net/
[s] http://prize.hutter1.net/
[t] http://mattmahoney.net/dc/text.html
[u] http://www.imageclef.org/
[v] http://www.statmt.org/europarl/
[w] http://www.statmt.org/setimes/
[x] http://matrix.statmt.org/matrix/info
[y] http://www.nist.gov/itl/iad/mig/openmt.cfm
[z] https://www.ldc.upenn.edu/new-corpora
[aa] https://aibirds.org/
[ab] http://www.arcadelearningenvironment.org/
[ac] http://gpbenchmarks.org/
[ad] http://www.movingai.com/benchmarks/
[ae] http://www.fira.net/contents/sub03/sub03_1.asp

2011), follows the general form of Eq. 3 where M is the repository, p is the choice of datasets and R is one particular performance metric (accuracy, AUC, Brier score, F-measure, MSE, etc., Ferri et al. 2009; Hernández-Orallo et al. 2012a). With the chosen datasets, several algorithms (where one or more are usually introduced by the authors of the research work) can be evaluated by their performance on the datasets. The aggregation over several datasets according Eq. 3, however, is not very common in machine learning, as there is the general belief that averaging the results for several datasets is wrong, as results are not commensurate (see, e.g., Demšar 2006). We already discussed this issue in Sect. 2.1 and saw that there are ways to normalise the performance metric or use some utility measure instead (e.g., what are the costs, in euros, of false positives and false negatives for each dataset) such that they can be aggregated. Nonetheless, statistical tests are the predominant and encouraged approach to evaluation validation by the machine learning research community.

"The UCI test" can be seen as a bona-fide mix of the problem benchmark approach and the peer confrontation approach. Even if there is a repository (M), only a few problems are chosen, and can be cherry-picked (p is changing and arbitrary). Also, as the researchers' algorithm must be compared to other algorithms to show that the new one is better, a few competing algorithms are chosen, which can also be cherry-picked, without much effort on fine-tuning their best parameters. Finally, as the results are analysed by statistical tests, cross-validation or other repetition approaches are used to reduce the variance of $R(\pi, \mu, p)$ so that we have fewer "ties". This procedure frequently leads to claims about new methods being better than the rest. Many of these claims are based on attempting many techniques and variants until some of them are better than state of the art. As a result, many results are usually affected by parameter overfitting, especially when using cross-validation (Rao et al. 2008) or are reproducible, but not replicable (Drummond 2009), i.e., whenever a few things are changed (about the kind of data, the application domain or the parameter tuning) the improvement completely vanishes. Nonetheless, the UCI repository is not to blame for this situation, but rather the methodology where statistical significance for a few datasets is given more value than a commensurate average aggregate performance on a large collection of datasets.

As a result, there have been suggestions of a better use of the UCI repository. These suggestions imply an improvement of the procedure but also of the repository itself. For instance, UCI+, "a mindful UCI" (Macià and Bernadó-Mansilla 2014), proposes the characterisation of the problems in the UCI repository by a set of complexity measures from (Ho and Basu 2002). This characterisation can be used to make samples that are more diverse and representative. Also, they discuss the notion of a problem being 'challenging', trying to infer a notion of 'difficulty'. In the end, an artificial dataset generator is proposed to complement the original UCI dataset. It is a distortion-based generator (similar to Soares's UCI++, Soares 2009). Finally, Macià and Bernadó-Mansilla (2014) suggest ideas about sharing and arranging the results of previous evaluations so that each new algorithm can be compared immediately with many other algorithms using the same experimental setting. This idea of 'experiment database' (Vanschoren et al. 2012) has already been set up. Openml[6] (van Rijn et al. 2013; Vanschoren et al. 2014) is an open science platform that integrates machine learning data, software and results.

Although some of these improvements are in the line of better sampling approaches (more representative and more effective), there are still many issues about the way these repositories are constructed and used. The complexity measures could be used to derive how representative a problem is with respect to the whole distribution in order to make a more adequate sampling procedure. Also, a pattern-based generator instead of a distortion-based generator could give more control of what is generated and its difficulty. This could be done with a stochastic generative grammar for different kinds of patterns, as is usually done with artificial datasets, using Gaussians or geometrical constructs. Finally, if results are aggregated according to Eq. 3, the experimental setting and the use of repetitions should be overhauled. For instance, by using 20 different problems with 10 repetitions using cross-validation (a very common setting in machine learning experiments) we have less information than by using 200 different problems with 1 repetition. Choosing the least informative procedure only makes sense because of the way results are fitted into the statistical tests and also because repetitions usually involve less effort than preparing a large number of datasets.

In other words, a benchmark can be well used or not, depending, e.g., on how datasets are chosen, and the evaluation procedures used for comparing algorithms. As an alternative,

[6] http://openml.org/.

independently-organised competitions or challenges are a fairer way of comparing progress in machine learning and related areas. For instance, several KDD challenges are organised at several conferences (such as the KDD Cup by ACM SIGKDD[7] and the Discovery Challenges at ECML/PKDD).[8] In these competitions, a single dataset or domain is used and the prize is given to the participant that is able to integrate better statistical and machine learning tools to get the best results using the chosen metric. This can work with automated submission procedures and leaderboards, such as those displayed by Kaggle[9] and other platforms. These competitions, if performed for a wide range of problems at a time, could be a way of controlling some of the methodological problems of how the repositories, such as the UCI, are used. Nevertheless, these will be still evaluating AI teams instead of systems. This is perhaps motivated by the view that machine learning algorithms are usually seen as components rather than systems. One way or another, there are no current competitions of non-interactive machine learning where the systems are really compared.

Reinforcement learning can be considered a part of machine learning, which is usually very different in terms of techniques and evaluation procedures, compared to what happens in other non-interactive machine learning areas, used in data mining and data science applications. In reinforcement learning, one of the main features is that the quality of a system is evaluated according to an aggregated metric of the received rewards during a session, episode or trial. The reinforcement learning competition[10] has been running intermittently since 2004. In 2014, three domains were included (helicopter, polyathlon and invasive species) (Dimitrakakis et al. 2014), although in the past there have been many other domains. Teams must be registered in a server and must provide a system that works under the RL-glue standard interface, by modifying a basic RL agent that is given with the training pack for each domain. Participants can take part in each of the three domains independently. Trials are repeated for robustness and leaderboards are built according to the aggregated reward metric.

One relevant insight extracted from the recent editions is that some "seemingly 'easy' domains have old approaches which remain quite hard to beat. It is consequently a difficult task to find new, sufficiently challenging domains; in the last competition [the invasive species problem was tried], which is an apparently very complex problem, but for which a very simple approach seemed to perform the best in the competition. In the end, the metrics [used] to test the algorithms are quite important, as different metrics may put different algorithms on top. To give an example, if we measure the total reward over a very large number of time steps, we may favor algorithms that are asymptotically optimal but which perform badly in the short term. In the end, a single number can't say very much" (Dimitrakakis 2016).

One domain that is very well-suited for RL-like agents is videogames. Since very realistic videogames in 3D with complex textures are generally beyond the state of the art for artificial vision systems, several proposals have been undertaken where simple arcade games are used instead. For instance, the Arcade Learning Environment[11] integrates many Atari 2600 videogames (Bellemare et al. 2013). The screen consists of 160×210 pixels, with a 128-colour palette and 18 actions. About 55 different games can be used as challenges for "reinforcement learning, model learning, model-based planning, imitation learning, transfer learning, and intrinsic motivation" (Bellemare et al. 2013). For instance, the score can be processed and mapped to the reward input of a generic RL interface (as done by Mnih

[7] http://www.kdd.org/kdd-cup.

[8] http://www.ecmlpkdd2015.org/discovery-challenges.

[9] http://www.kaggle.com.

[10] http://www.rl-competition.org/.

[11] http://www.arcadelearningenvironment.org/.

et al. (2015), showing fantastic performance), although using a very large number of training sessions.

One of the key issues of the Arcade Learning Environment is how scores are integrated and compared. Bellemare et al. (2013) discuss three ways in which scores can be normalised: *comparing to a reference score*, e.g., a random agent, *normalising with a baseline set*, e.g., using several agents to get some kind of average baseline score and *inter-algorithm normalisation*, i.e., setting the best algorithm for each game to 1 and then normalising the rest. Also, there is an interesting discussion on how scores for different games are aggregated. First, it is very important that they are first normalised, otherwise they will not be commensurate. Second, several options exist, such as an *average score*, *median score* or a *score distribution* (a quantile plot).

The General Video Game Competition[12] (Perez et al. 2015) is based on the Video Game Description Language (Schaul 2014), a language that allows arcade video games to be defined at an abstract level, describing objects and dynamics in a two-dimensional space. The analysis of the screen can be done at a more abstract level, without necessarily using an artificial vision approach at regular screenshots, as in the Arcade Learning Environment. Some games are simplified versions of popular games (e.g., Pacman) whereas others have been created on purpose. Games are defined in a reinforcement learning setting, but there are several reward schemas: binary (there is an all-or-nothing reward, depending on whether the agent achieves a final goal), incremental (a more traditional cumulative reward system) and discontinuous (somewhat in between). The competition is organised in three stages: training, validation and test. The games for the last stage are not known by the participants to prevent evaluation overfitting. Overall, this is a very significant effort towards more general artificial intelligence. Consequently it will be discussed again in Sect. 3. Nevertheless, it is not, for the moment, an ability-oriented approach, since it just aggregates results for several games without identifying the relevance of several abilities in each of them.

In a very different setting, several DARPA challenges for autonomous vehicles, security and rescue robots have been held in the past years. Some of them strongly rely on hardware and the quality of sensors according to the particular application, such as the DARPA Urban Challenge for autonomous vehicles[13] and the DARPA Save the Day (http://www.theroboticschallenge.org/) for Rescue Robots. A more software-oriented challenge is DARPA Cyber Grand Challenge[14] where a purpose-built computer competes against the circuit's greatest experts in CTF (Capture the Flag), a tournament circuit where experts reverse engineer software, probe its weaknesses, search for deeply hidden flaws, and create securely patched replacements. In general, DARPA integrates very specific and challenging domains that require a strong commitment for participation in terms of resources and the integration of techniques.

In the area of natural language processing (NLP), machine translation area is one where there is an abundance of corpora[15] as well as several evaluation efforts coordinated by the National Institute of Standards and Technology (NIST OpenMT) and DARPA (TIDES MT)[16]. The particular focus of each evaluation series has changed over the years but they are aimed at the general problem of automatic translation between human languages. One of the issues in these evaluations is how to rank alternative translations, using volunteer human assessments,

[12] http://www.gvgai.net/.

[13] http://archive.darpa.mil/grandchallenge/.

[14] http://www.cybergrandchallenge.com/.

[15] http://www.statmt.org/europarl/, http://www.statmt.org/setimes/, http://matrix.statmt.org/matrix/info.

[16] http://www.nist.gov/itl/iad/mig/openmt.cfm.

and the derivation of metrics from them. In a way, because of the human factor, some of these tasks can also be considered a case of the previous subsection (human discrimination), but in a much more controlled scenario, given the corpora.

Finally, some benchmarks integrate many domains, but at the same time are very specific (like the DARPA challenges). ImageClef, the CLEF Cross Language Image Retrieval Track[17], is a benchmark for the evaluation of cross-language annotation and retrieval of images (Caputo et al. 2014). The goal is the annotation and retrieval of images in various domains. The 2014 edition consisted of four tasks: domain adaptation, scalable concept image annotation, liver CT image annotation and robot vision. These changed into other five tasks in 2015: image annotation, medical classification, medical clustering and liver annotation.

Overall, even if the repositories and competitions in machine learning and other domains seen above may have particular issues, many of the benchmarks and competitions in Table 2 suffer from the same problems about how representative the problems are (if M is small), how representative the sample is (if M is large) or whether there are some kinds of problems that can be solved with specific approaches that dominate the sample. Other issues are the estimation of task difficulty and whether M is able to really discriminate between a set of AI systems. Also, none of the benchmarks in AI are adaptive.

2.4 Evaluation by peer confrontation

In the evaluation by peer confrontation, we evaluate a system by letting it compete against another system. This usually means that a match is played between peers. This is usual for games (including game theory) and also common in multi-agent research. The results of each match (possibly repeated with the same peer) may serve as an estimation of which of the two systems is best (and how much). Nonetheless, the main problem about this approach is that the results are relative to the opponents. This is natural in games, as people are said to be good or bad at chess, for instance, depending on whom they are compared to.

Despite this relative character of the evaluation, we can still see the average performance according to Eq. 3. In order to do this, we must first identify the set of opponents Ω. Then, the set of problems M is enriched (or even substituted) by the parametrisation of each single game (e.g., chess) with different competitors from Ω. In 1-vs-1 matches we have that $|M| = |\Omega| - 1$ (if we do not consider a match between a system and itself). In other multi-agent situations where many agents play at the same time, $|M|$ can grow combinatorially on $|\Omega|$.

Nonetheless, for AI research, our main concern is about robustness and standardisation of results. For instance, how can we compare results between two different competitions if opponents are different? If these competitions are performed year after year, how can we compare progress? If there are common players, we can use rankings, such as the Elo ranking (Elo 1978), or more sophisticated rating systems (Smith 2002; Masum and Christensen 2003), to see whether there is progress. In fact, it would be very informative for AI competitions based on peer confrontation to keep all participants from previous editions in subsequent editions. However, this comes with a drawback, as systems could specialise to the kind of opponents that are expected in a competition. If a high percentage of competitors are inherited from previous editions, specialisation to those old (and bad) systems could be common.

It is insightful to think how many of these issues are addressed in sport competitions. For instance, some tournaments adapt their matches according to previous information (by round, by ranking, etc.). In fact, a league may be redundant (for the same reasons why

[17] http://www.imageclef.org/.

the information-driven or difficulty-driven sampling are introduced) and other tournament arrangements are more effective with almost the same robustness and far fewer matches.

As an alternative, games and multi-agent environments could be evaluated against standardised opponents. However, how can we choose a set of standardised opponents? If the opponents are known, the systems can be specialised to the opponents. For instance, in an English draughts (checkers) competition, we could have players being specialised to play against Chinook, the proven optimal player (Schaeffer et al. 2007). Again, this ends up again in the design of an opponent generator. This of course does not mean a random player (which is usually very bad), but players that can play well. One option is to use old systems where some parameters are changed. Alternatively, a more far-reaching approach is to define an agent language and generate players (programs) with that language. As it is expected that this generation will not achieve very good players (otherwise we would be facing a very simple problem), a possible solution is to give more information and resources to these standardised opponents to make them more competitive (e.g., in some applications these opponents could have more sophisticated sensor mechanisms or some extra information about the match that regular players do not have).

Given the set Ω composed of old opponents or generated opponents, we need to assess whether Ω is sufficiently challenging and whether it is able to discriminate the participants. For instance, some competitions in AI finally award a champion, but there is the feeling that the result is mostly arbitrary and caused by luck, as happens with many sport competitions[18]. How can we assess whether the set Ω has sufficient difficulty and discriminating power? This is of course a hard problem, which has recently been analysed in (Hernández-Orallo 2014), which is not only applicable to multi-agent systems but for the assessment of any kind of task, by calculating the size of the simplest policy that solves the problem.

Table 3 shows a sample of evaluation schemes based on peer confrontation. Once again, because of obvious space constraints, we will just choose some representative and interesting cases from the table. First, we will see the Computer Olympiad[19], an event that congregates many board games, which has been held intermittently since 1989. After so many years, systems are very sophisticated and completely specialised to one game, which usually requires a very good integration of knowledge about the game and heuristics. One relevant issue of the evolution of the olympiad is that some games soon disappeared from the competitions, such as checkers, because an optimal solution was found for the game (and hence the competition lost interest for computers).

The General Game Competition, which has been run yearly since 2005, can be seen as a reaction to the computer olympiad and the classical approach to solve specific games, as represented by the superhuman results in particular games such as chess, draughts, poker and others. According to the webpage[20], "general game players are systems able to accept descriptions of arbitrary games at runtime and able to use such descriptions to play those games effectively without human intervention. In other words, they do not know the rules until the games start". Games are described in the language GDL (Game Description Language). The description of the game is given to the players. Different kinds of games are allowed, such as noughts and crosses (tic tac toe), chess, in static or dynamic worlds, with complete or partial information, with varying number of players, with simultaneous or alternating plays, two-player or single-player, etc. For the competition, games are chosen —non-randomly,

[18] Statistical tests are not used to determine when a contestant can be said to be significantly better than another.

[19] http://www.icga.org/.

[20] http://games.stanford.edu/.

Table 3 List of some evaluation schemes in the peer-confrontation category

Evaluation scheme	Description
Robocup[a] and FIRA[b]	Robotics (robot football/soccer) (Kitano et al. 1997; Kim 2004)
General game playing AAAI competition[c]	General game playing using GDL (Genesereth et al. 2005)
World Computer Chess Championship[d]	Chess
Computer Olympiad[e]	Board games
Annual Computer Poker Competition[f]	Poker
Trading Agents Competition[g]	Trading agents (Wellman et al. 2004; Ketter and Symeonidis 2012)
Warlight AI Challenge[h]	Strategy games (Warlight)

[a] http://www.robocup.org/
[b] http://www.fira.net
[c] http://games.stanford.edu/
[d] http://www.icga.org/
[e] http://www.icga.org/
[f] http://www.computerpokercompetition.org/
[g] http://tradingagents.eecs.umich.edu/
[h] http://theaigames.com/competitions/warlight-ai-challenge/rules

i.e., manually by the organisers— from the pool of games already described in GDL and new games are also newly introduced for the competition. As a result, game specialisation is difficult. The competition is run like many sports tournaments, with players participating in qualifying rounds (where they may be tested in single player games, e.g., Sudoku) or in two-player games against a sample player. Those qualified participate in preliminary rounds, where scores from single-player games and results against the other competitors are aggregated. The top four players from them qualify for the semifinal and final rounds. Results are basically win/loss/tie. This loses the information about partial situations during the game or the tactics that have been used during the game.

Despite being one of the most interesting AI competitions, there is still some margin for improvement. For instance, a more sophisticated analysis of how difficult and representative each problem is would be useful. For instance, several properties about the adequacy of an environment or game for peer-confrontation evaluation could be identified and analysed depending on the population of opponents that is being considered. Also, rankings (e.g., using the Elo system mentioned above) could be calculated, and former participants could be kept for the following competitions, so there are more participants (and more overlap between competitions). A more radical change would be to learn without the description of the game, as a reinforcement learning problem (where the system learns the rules from many matches).

The RoboCup Soccer competition[21] is clearly a competition where team confrontation takes place, and both intrateam cooperation and interteam competition are required. Even if it is a robotic competiton, the modalities are played with the same hardware (or with some strict hardware categories or specifications), so that the competition can focus on AI techniques, and not on sensorimotor hardware optimisation. Obviously, in many categories, the competition becomes more challenging not because new domains are introduced or because the rules are changed, but rather because the other opponents improve. The competition is held as usual sports tournaments, either as a league or with rounds.

[21] http://www.robocup.org/.

Summing up our observations on peer confrontation problems, we see that the dependency on the set Ω makes this kind of evaluation more problematic. Nonetheless, as AI research is becoming more socially oriented, with significantly more presence of multi-agent systems and game theory, an effort has to be undertaken to make this kind of evaluation more systematic, instead of the plethora of arrangements that we see in sports, for instance. Basically, the issue is that the organisers want to limit the number of confrontations when the number of participants is high, but may be reluctant to use a schema based on rounds because it can be highly unreliable. Apart from the Elo ranking (Elo 1978) and other rating systems mentioned above (Smith 2002; Masum and Christensen 2003), some recent studies have analysed partially completed sports competitions, and how possible and necessary winners can be derived from partial tournaments, and arrange pending pairwise comparisons accordingly (Aziz et al. 2015).

2.5 Highlights and directions of the evaluation of specialised AI systems

Given the three kinds of evaluation in the previous subsections (for which there may be some overlap, as we have seen, or cases that are difficult to classify, such as the AI cooperation game competition[22]), we now give a more comprehensive view of the key issues about all these initiatives. The first thing that we realise is that the evaluation efforts are extremely scattered across AI disciplines, and it is quite common to find duplicated efforts, for which solutions have to be found again and again. Apparently, there is limited exchange of experiences between them, just because the domains are different. It is then very useful to look at some organisations that can serve to centralise and exchange insights and lessons-learnt across several domains, apart from identifying new needs for benchmarks and competitions.

Examples of these organisations are government institutions such as NIST and DARPA, scientific organisations such as AAAI, or on-purpose associations, such as *chalearn*[23], focused on machine learning and with an associated book series. In particular, NIST has performed a continued effort towards the evaluation of several domains in AI. In fact, the series of workshops on Performance Metrics for Intelligent Systems, held from 2000 to 2012 at the National Institute of Standards & Technology (Meystel 2000; Messina et al. 2001; Evans and Messina 2001; Meystel et al. 2003b, a; Gordon 2007; Madhavan et al. 2009; Schlenoff et al. 2011) is the most relevant continuous effort in the analysis of the state of the art, the methodology and the progress of application-oriented evaluation in AI.

While the first two workshops discussed a possible analysis of the "Space of Intelligence" (Meystel 2000), their participants were "not looking for and [were] not interested in a nouveau Turing test" (Messina et al. 2001). The preference for task-oriented evaluation soon prevailed: "the more that we can make it clear that we are interested in *performance*, rather than intelligence, per se, the better off we will be" (Simmons 2000). From 2003 onwards the workshops focused almost exclusively on "performance measures" for "practical problems in commercial, industrial, and military applications" (Meystel et al. 2003b), covering, e.g., self-driving cars, robotic rescue systems, distributed control, human-robot interaction, soldier-worn sensor systems, Mars rovers, mining robots, smart grid systems, manufacturing robots, healthcare systems, etc.

Several interesting conclusions were drawn from the workshop reports through the years. For instance, there was the perception that the focus on applications have been created a schism with artificial intelligence. A session of the 2007 workshop focused on

[22] http://gaips.inesc-id.pt/geometryfriends/.

[23] http://www.chalearn.org/.

"(re-)establishing or increasing collaborative links between artificial intelligence and intelligent systems" (Gordon 2007), as the latter were supposed to be concerned about control and robotics. Actually, there was a perception of progress but, because the great amount of systems fell in the category of "cyber physical systems", it is difficult to tell in many domains whether this comes from better hardware, better sensors or better AI methods. Also, the specialisation of many metrics to the domain and the lack of continuity in some evaluation procedures were recognised as one of the limitations.

Apart from the PerMIS workshop, NIST has been organising several workshops where different AI domains are investigated and evaluated, and the evaluation procedures have more continuity and standardisation. For instance, NIST holds the Text Analysis Conference (TAC),[24] to encourage research in NLP by providing tests and common evaluation procedures. The tracks may change each year, but the conference has usually covered question answering, recognising textual entailment, summarisation and knowledge base population. A similar series is the Text Retrieval Conference (TREC)[25] encourages research in information retrieval from large text collections. The tracks take participants that submit their results and are evaluated, not as a real competition but more like a certification, where each participant is given a report about the shortcomings of their results. The tracks are adjusted year after year according to the results and the discussion of committees during the conferences and on the mailing list. This is not different from the way other competitions evolve in other organisations. However, NIST is an organisation that is specialised in evaluation and some methodological issues are ensured and shared across different domains.

From the more academic associations in artificial intelligence, there has been a renewed interest in establishing a series of tasks that could serve as a real evaluation of the progress of AI (You 2015; Marcus et al. 2016). The suggestion is to include several diverse tasks. For instance, the Winograd Schema Challenge is a commonsense reasoning task[26] (Levesque et al. 2012; Levesque 2014; Morgenstern et al. 2016) introduced by Hector Levesque in his 2013 IJCAI Research Excellence Address. The task features questions such as. "The trophy would not fit in the brown suitcase because it was too big (small). What was too big (small)? Answer 0: the trophy, Answer 1: the suitcase". This task and others (some resembling physically embodied versions of the Turing tests) could be chosen by IJCAI and AAAI for possible inclusion into a regular "Turing Championship" or "Turing Olympics" (You 2015). It is unclear why and how this is very different from what has been regularly done by NIST, apart from less emphasis on hardware and robotics.

3 Towards ability-oriented evaluation

AI is successful in many ways nowadays but it took a long time to flourish in applications (e.g., driverless cars, machine translators, game bots, etc.). Most of them correspond to specific tasks and require task-oriented evaluation. Other tasks that are still not solved by AI technology are already evaluated in this way and will be successful one day. However, if instead of AI applications we think about AI systems, we see that there are some kinds of AI systems for which task-oriented evaluation is not appropriate. For instance, cognitive robots, artificial pets, assistants, avatars, smartbots, etc., are not designed to cover one particular application but are expected to be customised by the user for a variety of tasks. In order to cover this wide range of (previously unseen) tasks, these systems must have some abilities

[24] http://www.nist.gov/tac/.

[25] http://trec.nist.gov/.

[26] http://commonsensereasoning.org.

such as reasoning skills, inductive learning abilities, verbal abilities, motion abilities, etc. Hence, this means that apart from task-oriented evaluation methods we may also need ability-oriented evaluation techniques.

Things are more conspicuous when we look at the evaluation of the *progress* of AI as a discipline. If we look at AI with Minsky's 1968 definition seen in the introduction, i.e., by achievement of tasks that would require intelligence, AI has progressed very significantly. For instance, one way of evaluating AI progress is to look at a task and check in which category an AI system is placed: *optimal* if no other system can perform better, *strong super-human* if it performs better than all humans, *super-human* if it performs better than most humans, *par-human* if it performs similarly to most humans, and *sub-human* if it performs worse than most humans (Rajani 2011). Note that this approach does not imply that the task is necessarily evaluated with a human-discriminative approach. Having these categories in mind, we can see how AI has scaled up for many tasks, even before AI had a name. For instance, calculation became super-human in the 19th century, cryptography in the 1940s, simple games such as noughts and crosses became optimal in 1960s, more complex games (draughts, bridge) a couple of decades later, printed (non-distorted) character recognition in the 1970s, statistical inference in the 1990s, chess in the 1990s, speech recognition in the 2000s, and TV quizzes, driving a car, technical translation, Texas hold 'em poker in the 2010s. According to this evolution, the progress of AI has been impressive (Bostrom 2014). The use of human intelligence as a baseline has been used in competitions (such as the humies awards)[27] or to define ratios, where median human performance is set at a zero scale, such as the so-called Turing-ratio (Masum et al. 2002; Masum and Christensen 2003), with values greater than 0 for super-human performance and values lower than 0 for sub-human performance.

However, let us first realise that no system can do (or can learn to do) all of these things together. The big-switch approach may be useful for a few of them (e.g., a robot with an advanced computer vision system that detects whether it is facing a chess board or a bridge table and then switch to the appropriate program to play the game that it has just recognised). Second, if we look at AI with McCarthy's definition seen in the introduction, i.e., by making intelligent machines, things are less encouraging. Not only has the progress been more limited, but also there is a huge controversy for quantifying this progress (in fact, some argue that machines are more intelligent today than 50 years ago while others say that there has been no progress at all other than computational power). Hence, worse than having a poor progress or no progress at all, we regard with contempt that we do not have effective evaluation mechanisms to evaluate this progress. It seems that none of the evaluation schemes seen in the previous section are able to evaluate whether the AI systems of today are more intelligent than the AI systems of yore. Also, for developmental robotics and other areas of AI where systems are supposed to improve their performance with time, we want to know if a 6-month-old robot has progressed over its initial state, in the same way that we see how abilities increase and crystallise with humans, from toddlers to adults. Ability-oriented evaluation, and not task-oriented evaluation, seems to have a better chance of answering this question.

To make the point unequivocal, we could even go beyond McCarthy's definition of AI without the use of 'intelligence' and define this view of AI as the *science and engineering of making machines do tasks they have never seen and have not been prepared for beforehand*. Clearly, this view puts more emphasis on learning, but it also makes it crystal clear that task-oriented evaluation, as have been performed for years, would not fit the above definition.

It would be unfair to forget to acknowledge that some attempts seen in the previous section have made an effort for a more general AI evaluation. The general game competition seen

[27] www.human-competitive.org.

in the previous section is one example of how some things are changing in evaluation. Users and researchers are becoming tired of a big-switch approach. They yearn for and conceive systems that are able to cover more and more general task classes. Nonetheless, it is still a limited generalisation, which is too based on a very specific range of tasks. Many good players at the General Game Competition would be helpless at any game of the Arcade Learning Environments, and vice versa. Actually, only some reinforcement learning (and perhaps genetic programming) systems can at least participate in (adaptations to) of both games—excelling in them would not be possible though without an important degree of specialisation.

In the rest of this section we will introduce what an ability is and how they can be evaluated in AI. The title of this section (starting with 'Towards') suggests that what follows is more interdisciplinary and contains proposals that are not well consolidated yet, or that may even go in the wrong direction. Nonetheless, let us be more lenient and have in mind that ability-based evaluation is much more challenging than task-specific evaluation.

3.1 Cognitive abilities

We must first clarify that we are talking about *cognitive* abilities, in the same way that in the previous section we referred to *cognitive* tasks. Some AI applications require physical abilities, most especially in robotics, but AI deals with how the sensors and actuators are controlled, not about their strength, consumption, etc. After this clarification, we can define a cognitive ability as *a property of individuals that allows them to perform well in a range of information-processing tasks*. At first sight this definition may just look like a change of perspective (from problems to systems). However, what we see now is that the ability is required, and performance is worse without featuring the ability. In other words, the ability is necessary but it does not have to be sufficient (e.g., spatial abilities are necessary but not sufficient for driving a car). Also, the ability is assumed to be general, to cover a range of tasks. Actually, general intelligence would be one of these cognitive abilities, one that covers *all* cognitive tasks: "general intelligence is a very broad trait that encompasses quickness and quality of response to all cognitive tasks" (Strickler 1973).

The major issue about abilities is that they are 'properties', and as such they have to be conceptualised and identified. While tasks can be seen as measuring *instruments*, abilities are *constructs*. In psychology, many different cognitive abilities have been identified and have been arranged in different ways (Schaie 2010). For instance, one well-known comprehensive theory of human cognitive abilities is the Cattell–Horn–Carroll theory (Keith and Reynolds 2010). Figure 4 shows a graphical representation of these abilities. The top level represents the *g* factor or general intelligence, the middle level identifies a set of broad abilities and the bottom level may include many narrow abilities. Again, this top level seems to saturate all tasks: "g is common to all cognitive tasks including learning tasks" (Alexander and Smales 1997).

Interestingly, this is not surprising from an AI standpoint. The broad abilities seem to correspond to subfields in AI. For instance, looking at any AI textbook (e.g., Russell and Norvig 2009), we can enumerate areas such as problem solving, use of knowledge, reasoning, learning, perception, natural language processing, etc., that would roughly correspond to some of the cognitive abilities in Fig. 4.

Can we evaluate broad abilities as we did for specific tasks? Application-specific (task-oriented) approaches will not do. But is ability-oriented evaluation ready for this? The answer, as we will see below, is that this type of evaluation is still in a very incipient stage in AI. There are several reasons for this. First, general (ability-oriented) evaluation is more challenging.

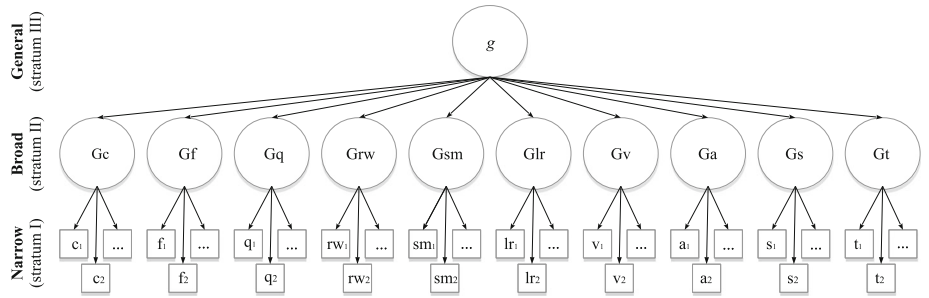

Fig. 4 Cattell–Horn–Carroll's three stratum model. The broad abilities are Crystallised Intelligence (Gc), Fluid Intelligence (Gf), Quantitative Reasoning (Gq), Reading and Writing Ability (Grw), Short-Term Memory (Gsm), Long-Term Storage and Retrieval (Glr), Visual Processing (Gv), Auditory Processing (Ga), Processing Speed (Gs) and Decision/Reaction Time/Speed (Gt)

Second, we no longer have a clear definition of the task(s). In fact, defining the ability depends on a conceptualisation, and from there we need to find a set of representative exercises that require the ability. And third, there have not been too many general AI systems to date, so task-oriented evaluation has seemed sufficient for the evaluation of AI systems so far. However, things are changing as new kinds of AI systems (e.g., developmental robotics) are becoming more general.

Before starting with some approaches in the direction of ability-oriented evaluation, it can be argued that some existing evaluation schemes in AI are already ability-oriented. For example, even if the planning competition features a set of tasks, it goes around the *ability* of planning, which is more general than any particular task. However, the systems are not able to determine when planning is required for a range of problems. In other words, the ability is not a resource of the system, but the very goal of the system. In the end, it is the researchers who incorporate planning modules in several application-specific systems, and not the systems that independently enable their planning abilities to solve a new problem.

3.2 The anthropocentric approach: psychometrics

Psychometrics was developed by Galton, Binet, Spearman and many others at the end of the 19th century and first half of the 20th century. An early concept that arose was the need of distinguishing tasks requiring very specific knowledge or skills from general abilities. For instance, an "idiot savant" could have a lot of knowledge or could have developed a sophisticated skill during the years for some specific domain, but could be obtuse for other problems. On the contrary, a very able person with no previous knowledge could perform well in a range of tasks, provided they are culture-fair. This distinction took several decades to consolidate. In a way, this bears resemblance with the narrow versus general dilemma in AI.

Psychometrics is concerned about measuring cognitive abilities, personality traits and other psychological properties (Sternberg 2000). Factors differ from abilities, in principle, in that they are obtained through testing and further analysed through systematic approaches based on factor analysis. Some factors have been equated and named after existing abilities while others are 'discovered' and receive new technical names. Several indices can be derived from a battery of tests by aggregating abilities and factors. One joint index that is usually determined from some of these tests is known as IQ (Intelligence Quotient). Although IQ was originally normalised by the subject's age (hence its name), its value for adults today is normalised relative to an adult population, assuming a normal distribution with mean $\mu = 100$

and standard deviation $\sigma = 15$. This corresponds to a more sophisticated (normalised) aggregation of results for several items, which again resembles our Eq. 3.

IQ tests incorporate items of variable difficulty. Item difficulty is determined by the percentage of subjects that are able to solve the item, or using functional models in Item Response Theory (Lord 1980; Embretson and Reise 2000), as seen in the previous section. Note that this difficulty assessment is relative to the population and not derived from the nature of the item itself.

IQ tests are easy to administer, fast and accurate, and they are used by companies and governments, essential in education and pedagogy. IQ tests are generally culture-fair through the use of abstract exercises (except for the verbal comprehension abilities).

As they work reasonably well for humans, their use for evaluating machines has been suggested many times, even since the early days of AI, with the goal of constructing "a single program that would take a standard intelligence test" (Newell A 1973). More recently, their use has been vindicated by Bringsjord and Schmimanski (Bringsjord and Schimanski 2003; Bringsjord 2011), under the so-called 'Psychometric AI' (PAI), as "the *field* devoted to building information-processing entities capable of at least solid performance on all established, validated tests of intelligence and mental ability, a class of tests that includes not just the rather restrictive IQ tests, but also tests of artistic and literary creativity, mechanical ability, and so on". It is important to clarify that PAI is a redefinition or new roadmap for AI—not an evaluation methodology—and does not further develop or adapt IQ tests for AI systems. In fact, PAI does not explicitly claim that IQ tests (or other psychometric tests) are the best way to evaluate AI systems, but it is said that an "agent is intelligent if and only if it excels at all established, validated tests of intelligence" (later broadened to any other psychometric test) (Bringsjord and Schimanski 2003; Bringsjord 2011). The question of whether these tests are a necessary and sufficient condition for machines and the limitations of PAI as a guide for AI research have been recently discussed in (Besold 2014).

Not surprisingly, this claim that IQ tests are the best way to evaluate AI systems has recently come from human intelligence research. Detterman, editor of the Intelligence Journal, wrote an editorial (Detterman 2011) where he suggested that Watson (the then recent winner of the *Jeopardy!* TV quiz (Ferrucci et al. 2010)) should be evaluated with IQ tests. The challenge is very explicit: "I, the editorial board of *Intelligence*, and members of the International Society for Intelligence Research will develop a unique battery of intelligence tests that would be administered to that computer and would result in an actual IQ score" (Detterman 2011). Detterman established two levels for the challenge, a first level, where the type of IQ tests can be seen beforehand by the AI system programmer, and a second level, where the types of tests would have not seen beforehand. Only computers passing the second level "could be said to be truly intelligent" (Detterman 2011). The need for two levels seems related to the big-switch approach and the problem overfitting issue, which we have already mentioned in previous sections for AI evaluation schemes. It is apposite at this point to recall that academic and professional IQ tests and many other standardised psychological tests are never made public, because otherwise people could practise on them and game the evaluation. Note that the non-disclosure of the tests until evaluation time is something that we only find in very few evaluation schemes in the previous section.

Detterman was unaware that almost a decade before, in 2003, Sanghi and Dowe (Sanghi and Dowe 2003) implemented a small program (less than 1000 lines of code) which could score relatively well on many IQ tests, as shown in Table 4. The program used a big-switch approach and was programmed to some specific kinds of IQ tests the authors had seen beforehand. The authors still made the point unequivocally: this program is not intelligent and can pass IQ tests.

Table 4 Results by a rudimentary program for passing IQ tests [from Sanghi and Dowe (2003)]

Test	IQ score	Human average
A.C.E. IQ test	108	100
Eysenck test 1	107.5	90–110
Eysenck test 2	107.5	90–110
Eysenck test 3	101	90–110
Eysenck test 4	103.25	90–110
Eysenck test 5	107.5	90-110
Eysenck test 6	95	90–110
Eysenck test 7	112.5	90–110
Eysenck test 8	110	90–110
IQ test labs	59	80–120
Testedich IQ test	84	100
IQ test from Norway	60	100
Average	96.27	92–108

While it must be conceded that the results only reach the first level of Detterman's challenge—so there is a test administration issue (i.e., an evaluation flaw)—there are some weaknesses about human IQ tests that would also arise if a system passed the second level as well. In particular, "the editorial board of *Intelligence*, and members of the International Society for Intelligence Research" could be tempted to devise or choose those IQ tests that are more 'machine-unfriendly'. If AI systems eventually passed some of them, the battery could be refined again and again, in a similar way as how CAPTCHAs are updated when they become obsolete. In other words, this selection (or battery) of IQ tests would need to be changed and made more elaborate year after year as AI technology advances. Also, the limitations of this approach if AI systems ever become more intelligent than humans are notorious.

The main problem about IQ tests is that they are anthropocentric, i.e., they have been devised for humans and take many things for granted. For instance, most assume that the subject can understand natural language to read the instructions of the exercise. On top of that, they are specialised to some human groups. For instance, tests are significantly different when evaluating small children, people with disabilities, etc. Also, the relation between items and abilities have been studied during the past century exclusively using humans, so it is not clear that a set of items would measure the same ability for a human or for a machine. For instance, is it reasonable to expect that well-established tests of choice reaction time be correlated with intelligence in machines as they are correlated in humans (Deary et al. 2001)? Or, what makes a set of psychometric tests different from a set of "human intelligence tasks" in Amazon Mechanical Turk (Buhrmester et al. 2011)? For a more complete discussion about why IQ tests are not ready for AI evaluation, the reader is referred to a response (Dowe and Hernández-Orallo 2012) to Detterman's editorial.

Having said all this and despite the limitations of IQ tests for AI evaluation, their use is becoming more popular in the past decade (including robotics, Schenck 2013) and systems whose results are like those of Table 4 are becoming common (for a survey, see Hernández-Orallo et al. 2016, for an open library of IQ tests, see PEBL).[28]

[28] http://pebl.sourceforge.net/battery.html.

As just said, one of the problems of IQ tests is that they are specialised for humans. In fact, standardised adult IQ tests do not work with people with disabilities or children of different ages. In a similar way, we do not expect animals to behave well on a standard human IQ test, starting from the fact that they will not be able to read the text. This leads us to the consideration of how cognitive abilities are evaluated in animals. Comparative psychology and comparative cognition (Shettleworth 2010; Shettleworth et al. 2013) are the main disciplines that perform this evaluation. For a time, much research about cognitive abilities in animals was performed on apes. The term 'chimpocentric' was introduced as a criticism about tests that had gone from being anthropocentric to being chimpocentric. Nonetheless, in the past decades, the perspective is much more general and any species may be a subject of study for comparative psychology: mammals (apes, cetaceans, dogs and mice), birds and some cephalopods. The evaluation focuses on "basic processes", such as perception, attention, memory, associative learning and the discrimination of concepts, and recently on more sophisticated instrumental or social abilities (Shettleworth et al. 2013).

One of the most distinctive features of animal evaluation is the use of rewards, as instructions cannot be used. This setting is very similar to the way reinforcement learning works. Animal evaluation has also brought attention to the relevance of the interface. Clearly, the same test may require very different interfaces for a dolphin and a bonobo.

Human evaluation and animal evaluation have become more integrated in the past years, and testing procedures half way between psychometrics and comparative cognition are becoming more usual. For instance, several kinds of skills are evaluated in human children and apes in (Herrmann et al. 2007). In recent years, many abilities that were considered exclusively human have been found to some extent in many animals (Wasserman and Zentall 2006; Shettleworth 2010).

Does the enlargement from humans to the whole animal kingdom suggest that these tests for animals can be used for machines? While the lower ranges of the studied abilities and the use of rewards can facilitate its application to AI systems significantly (at least for some autonomous systems in cognitive robotics, autonomous development, 'animats', robotic pets and other AI systems that are designed to resemble human or animal behaviours), we still have many issues about whether they can be applied in AI (at least directly). First, the selection of tasks and abilities is not systematic. Second, many of the tasks that are applied to animals would be too easy for machines (e.g., memory). And third, others would be too difficult (e.g., orientation, recognition and interaction in the real world). Nonetheless, there seems to be an increasing interest for the evaluation of the so-called 'animats' (AI systems that are inspired by or resemble an animal, Williams and Beer 2010) and the evaluation procedures for animals are the first candidates to try.

3.3 Evaluation using AIT

A radically different approach to AI evaluation started in the late 1990s. If intelligence was viewed as a "kind of information processing" (Chandrasekaran 1990) then it seemed reasonable to look at information theory for an "essential nature or formal basis of intelligence and the proper theoretical framework for it" (Chandrasekaran 1990). This was finally done with *algorithmic information theory* (AIT), and the related notions of Solomonoff universal probability (Solomonoff 1964), Kolmogorov complexity (Li and Vitányi 2008) and Wallace's Minimum Message Length (MML) (Wallace and Boulton 1968; Wallace and Dowe 1999), which we describe below.

There are several good properties about *algorithmic information theory* for evaluation. First, several definitions of information and complexity can be defined exclusively in com-

putational terms, actually relative to a Universal Turing Machine (UTM), a fundamental and universal model of effective computation. For instance, the Kolmogorov complexity of an object (expressed as a binary string) relative to a UTM is defined as the shortest program (for that machine) that describes/outputs the object. Even if these definitions depend on the UTM that is used, the invariance theorem states that their values will only differ with respect to other UTM up to a constant that only depends on the two different UTMs (because one can emulate the other) (Li and Vitányi 2008). The notion of algorithmic probability, introduced by Solomonoff, allows a universal distribution to be defined for each UTM, which is just the probability of objects as outputs of a UTM fed by a fair coin. While, in general, this means that compressible strings are more likely than incompressible ones, it can be shown that every computable probability distribution can be approximated by a universal distribution. In a way, Solomonoff, the father of algorithmic probability (Solomonoff 1964) gave a theoretical backing to Occam's razor. There are reasons to think that many phenomena and, as a result, many of the problems that we face every day, follow a universal distribution. This is directly linked to Eq. 3 again, and the discussion about the choice of the probability p. Also, we have the relevant fact, which is very significant for evaluation as well, that universal distributions are immune to the no-free-lunch theorems, where system performance can differ very significantly for induction (Lattimore and Hutter 2013; Hibbard 2009). And finally, Kolmogorov complexity and algorithmic probability are two sides of the same coin, which led to a formal connection of compression and inductive inference. One particular common (Bayesian) interpretation can be made under the Minimum Message Length, where the best hypothesis is the one that minimises the length of the theory and the length of coding the evidence using the theory. It has been acknowledged that Solomonoff "solved the problem of induction" (Solomonoff 1996; Dowe 2013).

Of course, not everything in AIT is straightforward. For instance, some of these concepts lead to incomputable functions, although approximations exist, such as Levin's Kt (Levin 1973). In Levin's Kt, it is not only the size of the program that is minimised but also the logarithm of the computational steps taken by the program to produce the string. In this way, finding the program that minimises the sum of these two terms becomes computable, and also has some important connections with heuristics (Levin 2013), and artificial intelligence (Solomonoff 1984).

Chaitin suggested the application of AIT to the "definition of intelligence and measures of its varios components" (Chaitin 1982), but the use of AIT for (artificial) intelligence evaluation started with a variant of the Turing test that featured compression problems (Dowe and Hajek 1997, 1998), to make the test more sufficient. While one of the goals of this work was to criticise Searle's Chinese room[29], this is one of the first intelligence test proposals using AIT. At roughly the same time, a formal definition of intelligence in the form of a so-called C-test was derived from AIT (Hernández-Orallo and Minaya-Collado 1998; Hernández-Orallo 2000a). Figure 5 shows examples of sequences that appear in this test. They clearly resemble some exercises found in IQ tests, such as Thurstone letter series (Thurstone 1938a). The major differences are that (1) sequences are obtained by a generator (a UTM with some post-conditions about the generated sequence, ensuring the unquestionability of the series continuation and less dependency on the reference machine) and (2) the fact that

[29] The Chinese room argument was introduced by (Searle 1980) to argue against the possibility of a machine having a mind, by comparing a computer processing inputs and outputs as symbols with a person knowing no Chinese in a room receiving messages in Chinese that have to be answered, also in Chinese, using a series of books to map inputs to outputs. Given the relevance of machine learning in AI nowadays, among other things, the argument has mostly faded today.

$k = 9$: a, d, g, j, … Answer: m
$k = 12$: a, a, z, c, y, e, x, … Answer: g
$k = 14$: c, a, b, d, b, c, c, e, c, d, … Answer: d

Fig. 5 Several series of different complexity 9, 12, and 14 used in the *C*-test (Hernández-Orallo 2000a)

each sequence is accompanied by a theoretical assessment of difficulty (a variant of Levin's Kt complexity). Note the implications for evaluation of such a test, as exercises are derived from first principles (instead of being contrived by psychometricians) and the difficulty of these exercises is intrinsic, and not based on how difficult humans find them. Finally, these sequences were used to define a test by aggregating results in a way that highly resembles our recurrent Eq. 3, where M is formally defined as including *all* possible sequences (following some conditions) and the probability is defined to cover a range of difficulties, leading to a difficulty-driven sampling as in Fig. 1 (right).

Some preliminary experimental results showed that human performance correlated with the absolute difficulty (k) of each exercise and also with IQ test results for the same subjects. This encourages the use of this approach for IQ-test re-engineering. With the aim of a more complete test, some extensions of the *C*-test were suggested, such as transforming it to work with interactive agents ("cognitive agents […] with input/output devices for a complex environment" (Hernández-Orallo and Minaya-Collado 1998) where "rewards and penalties could be used instead" (Hernández-Orallo 2000b)). Despite its explanatory power about IQ tests, this line of research was held back by some literal views of compression as intelligence, and even the proposal of tests of intelligence based on the compression of text (Mahoney 1999). The use of AIT for measuring intelligence was more sharply dashed in 2003 (at least towards general intelligence tests resembling IQ tests used for machines) by the evidence that very simple—non-intelligent—programs could pass IQ tests (Sanghi and Dowe 2003), as we have discussed in Sect. 3.2 (see Table 4).

Nonetheless, the extension to interactive agents was performed anyway. Interestingly, when agents and environments are considered in terms of Eq. 3, we just find a performance aggregation over a set of environments, exactly as had been formulated several times in the past: "intelligence is the ability of a decision-making entity to achieve success in a variety of goals when faced with a range of environments" (Fogel 1991). Note that this roughly corresponds to the psychometric view of general intelligence as key to performance in a range (or all) cognitive tasks. A crucial aspect was then to define this *range* of environments, i.e., the choice of the distribution in Eq. 3. One option was to include *all* environments. In order to do this in a meaningful, elegant way (and get rid of any no-free lunch theorem), AIT and reinforcement learning were combined (Legg and Hutter 2007b). Equation 3 was instantiated with all environments as tasks with a universal distribution for p, i.e., $p(\mu) = 2^{-K(\mu)}$, with $K(\mu)$ being the Kolmogorov complexity of each environment μ.

These proposals present several problems. First, some constructions are not computable, so approximations need to be used. Second, most environments are not really discriminative, and all agents will score the same, will just 'die' or be stuck after a few steps. Third, overweighting very small environments (by the use of a universal distribution or a complexity limit) makes the definition very dependent on the reference machine chosen as environment generator. Finally, time (or speed) is not considered for the environment or for the agent. For more details about these (and other) issues and some possible solutions, the reader is referred to (Hibbard 2009) and (Hernández-Orallo and Dowe 2010, Sects. 3.3 and 4). Taking into account these solutions, some actual tests have been developed (Insa-Cabrera et al. 2011b, a; Legg and Veness 2013). While the results may still be useful to rank some state-of-the-art machines, if

they are not compared to humans (or animals), as we discuss in the following section, the validation (or more precisely the refutation) of these tests as true intelligence tests cannot be done.

Summing up, the AIT approach is characterised by the definition of tests from formal information-based principles. This is in stark contrast to other approaches where tasks are collected, refined by trial-and-error or invented in a more arbitrary way. Most of the approaches to AI evaluation using AIT seen above have aimed at defining and measuring *general* intelligence, which is placed at the very top of the hierarchy of abilities (and hence at the opposite extreme from a specialised task-oriented evaluation). However, many interesting things can happen if AIT is applied at other layers of the hierarchy, for general cognitive abilities other than intelligence, as suggested in (Hernández-Orallo 2000c) for the passive case and hinted in (Hernández-Orallo and Dowe 2010, Sects. 6.5 and 7.2) for the dynamic cases, with the use of different kinds of videogames as environments (two of the most recently introduced benchmarks and competitions are in this direction, Bellemare et al. 2013; Schaul 2014; Perez et al. 2015). Finally, the information-theoretic approach is not isolated from some of the approaches seen so far in Sect. 2. Actually, some hybridisations and integrated approaches have been proposed (Hernández-Orallo et al. 2011; Insa-Cabrera et al. 2012), apart from the compression-enriched Turing tests (Dowe and Hajek 1997, 1998), already mentioned above.

There have been other approaches that are related to AIT with the aim of defining a measure (or theory) of intelligence in mathematical terms, such as Smith's "uniformly asymptotically competitive intelligence", relying on an enumeration algorithm looking for policies (Smith 2006), or Yampolskiy's 'efficiency theory' (Yampolskiy 2015, chap. 9). Also, AIT connected very well with early perspectives about the principle of simplicity under the so-called structural complexity (Krueger and Osherson 1980). Psychology and cognition began to include more explicit references to Kolmogorov complexity and related notions in AIT (Chater 1999; Chater and Vitányi 2003; Feldman 2003; Leeuwenberg and Van Der Helm 2012) and derived into the analyses or derivation of more tests using ideas from Kolmogorov complexity (Strannegård et al. 2013b, a; Schmid and Ragni 2015; Nizamani 2015).

The tasks and tests generated with AIT have been restricted to interactive (learning) systems in a sequential context, tasks resembling those found in IQ tests or in a reinforcement learning setting, as seen above. However, for AI practitioners in other areas such as more traditional machine learning, datasets could be generated using principles from AIT, quantifying the complexity of the best hypothesis for a dataset (including noise or not). Although not based on AIT, this has been applied with a statistical approach rather than an informational approach based on the distortion of actual datasets, as mentioned in the previous section (Soares 2009; Macià and Bernadó-Mansilla 2014). Of more interest can be the creation of datasets whose dependence on previous datasets is based on an algorithmic modification rather than statistical distortions. We will discuss this in Sect. 3.5 in the context of transfer learning (sequences of tasks) and inductive programming.

3.4 Universal psychometrics

The previous sections show a fragmentation of techniques and problems. This fragmentation originated by the kind of measurement we are interested in (task-oriented or ability-oriented, collected or AIT-derived tests) but most especially by the kind of subject that is being measured. In (Hernández-Orallo and Dowe 2010), the notion of 'universal test' is introduced, as a test that is applicable to "any biological or artificial system that exists at this time or in the future": human, non-human animal, enhanced human, machine, hybrid or collective. The stakes were set high, as the tests should work without knowledge about the subject, derive from computational principles, be unbiased (species, culture, language, …), require

no human intervention, be practical, produce a meaningful score, and be anytime (the more time we have for the test the higher the reliability of the score). Note that in order to apply the same test to several subjects we are allowed to customise the interface, provided the features and difficulty of the items are permitted to remain unaltered. Also, we need to think about the speed of the subject, and adapt to it accordingly. Also, the capabilities of the subject can be quite varied, so the ranges of difficulty need to adapt to the agent. That suggests that universal tests must necessarily be adaptive.

A first framework for universal, anytime intelligence tests is introduced in (Hernández-Orallo and Dowe 2010), where a class of environment is carefully chosen to be discriminative. The test starts with very simple environments and adapts to the subject's performance and speed. In this regard, this resembles a difficulty-driven sampling as described in Sect. 2.3. The set of tasks (environments) was developed upon some of the ideas about using AIT for intelligence evaluation, as seen in Sect. 3.3. Some experiments were performed (Insa-Cabrera et al. 2011b, a) using the environment class defined in (Hernández-Orallo 2010). Difficulty was estimated using a variant of Levin's Kt. As a way of checking whether the results were meaningful, the same test compared Q-learning (Watkins and Dayan 1992) with humans. Two different interfaces were designed on purpose. The test gave consistent results for Q-learning and humans when considered separately, but were less reasonable when put together. The experimental settings featured many limitations (simplifications, non-adaptiveness, absence of noise, low-complexity patterns, no incrementality, no social behaviour, etc.) and, probably because of this, the results did not show the actual difference between Q-learning and humans. Despite the limited results, the experiment had quite a repercussion (Kleiner 2011; Biever 2011; Yonck 2012). Nonetheless, the tests were a first effort towards a universal test and highlighted some of the challenges.

One possible explanation for all these limitations is that universal intelligence tests may simply be impossible (Smith 2006; Edmondson 2012) or the very notion of a general intelligent system unfeasible (Melkikh 2014), supported by the no-free-lunch theorems (Wolpert and Macready 1995; Wolpert 1996, 2012). Another less extreme explanation is based on the concern about a generator of environments lacking richness of interaction and social behaviours. In other words, an environment that is randomly generated will have an extremely low probability of showing some social behaviour, which is a distinctive trait of human intelligence. This has suggested other ways of generating the environments and ways of incorporating other agents into them (e.g., the Darwin-Wallace distribution, Hernández-Orallo et al. 2011), but it is still an open research question how to adapt these ideas to the measurement of social intelligence and multi-agent systems (Insa-Cabrera et al. 2012; Insa-Cabrera 2016).

The fragmentation of approaches and the need of solving many of the above issues has suggested the introduction of a new perspective, dubbed 'universal psychometrics' (Hernández-Orallo et al. 2014). Universal psychometrics focuses on the measurement of cognitive abilities for the 'machine kingdom', which comprises any (cognitive) system, individual or collective, either artificial, biological or hybrid. This comprehensive view is born with many hurdles ahead. Evaluation is always harder the less we know about the subject. The less we take for granted about the subjects the more difficult it is to construct a test for them. For instance, human intelligence evaluation (psychometrics) works because it is highly specialised for humans. Similarly, animal testing works (relatively well) because tests are designed in a very specific way to each species. And some of the AI evaluation schemes we have already seen work because they are specialised for some kind of AI systems that are designed for some specific applications. In the case of AI, who would try to tackle a more general problem (evaluating any system) instead of the actual problem (evaluating

Fig. 6 The realm of evaluable subjects for universal psychometrics

machines)? The answer to this question is that the *actual* problem for AI is the *universal* problem. Notions such as 'animat' (Williams and Beer 2010), machine-enhanced humans (Cohen 2013), human-enhanced machines (von Ahn 2009), other kind of hybrids and, most especially, collectives (Quinn and Bederson 2011) of any of the former, suggest that the distinction between animals, humans and machines is not only inappropriate, but no longer useful to advance in the evaluation of cognitive abilities. The notion of 'machine kingdom', as illustrated in Fig. 6, is not very surprising to the current scientific paradigm but clarifies which class of subjects is most comprehensive.

Universal psychometrics attempts to integrate and standardise a series of concepts. A subject is seen as a physically computable (resource-bounded) interactive system. Cognitive tasks are seen as physically computable interactive systems with a score function. Interfaces are defined between subjects and tasks (observations-outputs, actions-inputs). Cognitive abilities are seen as properties over a set of cognitive tasks (or task classes). As a result, the separation between task-specific and ability-specific becomes a progressive thing, depending on the generality of the task class. Distributions are defined over task classes and results as aggregated performance on a task class (again, a generalised version of Eq. 3). Difficulty functions are computationally defined from each task. Overall, some of these elements found in psychometrics, comparative cognition and AI evaluation are overhauled here with the theory of computation and AIT. As a result, cognitive abilities are no longer *what the cognitive tests measure*, as in human psychometrics, so adapting the (in)famous statement that "measurable intelligence is simply what the tests of intelligence test" (Boring 1923), but they are properties that emanate from (general) classes of tasks, perfectly defined in computational terms. As a consequence, the relation between abilities can be explored experimentally, but also theoretically, and measures are absolute and not relativised wrt. a population (except for social abilities). This could imply some revitalisation of the white-box approach, especially for those AI systems that can be formally described in a theoretical way (e.g., some results in Hutter (2007) and Hibbard (2009) take a white-box evaluation approach).

This view of a cognitive ability is consistent with its association with a "class of cognitive tasks" (Carroll 1993) that must be 'representative' for the ability. From the association between abilities and classes of tasks, we see that by merging two cognitive task classes we get a more general cognitive task class, and a more general ability. Typically, this is studied in a hierarchical way, starting with the so-called elementary cognitive tasks (Carroll 1993, p. 11) (closely related to the notion of primary mental abilities of Thurstone 1938a). This redraws our dilemma between task-oriented and ability-oriented into a gradual hierarchy from specific tasks to general abilities, with general intelligence at the very top (including *all* possible cognitive tasks, i.e., *all* interactive Turing machines with a score function). The

questions about how to sample from a task class for an effective evaluation can be generalised from our discussion in Sect. 2.

This sets a dual view of cognitive tasks on the one hand and cognitive systems on the other hand, where both spaces (the ability space and the machine kingdom) can be explored. Interestingly, both cognitive tasks and cognitive systems are defined as interactive systems, reflecting a duality world-agent. One singularity of cognitive systems (as well as the environments they are in) is that they can evolve with time, and their abilities can change. In other words, it seems that some abilities need to be constructed on top of other previously consolidated abilities, and this seems to be independent of the subject to some extent, in the same way that it seems difficult to be able to multiply without being able to add. A theoretical analysis of ability interdependency, how they can develop and the notion of potential intelligence, are still in a very incipient stage (Hernández-Orallo and Dowe 2013).

As we have discussed, human-based psychometrics is inadequate for most of the systems and techniques and AI. In particular, much of our notion of 'ability' is very anthropocentric, and does not reflect well the diversity of artificial systems. This is why universal psychometrics is not meant to be an extension of human psychometrics. Again, this does not mean that all techniques from psychometrics should be excluded, as there are some interesting principles and tools there (e.g., item response theory, adaptive tests, etc.), as already discussed in this paper. The key idea of universal psychometrics is to categorise all the possible kinds of systems, artificial or natural and see how they can be evaluated, without any anthropocentric assumption.

Of course, there can be objections and disagreements about the way many of the key concepts in universal psychometrics should be understood and defined. For instance, the definition of *subjects* as interactive cognitive systems may apparently exclude some AI systems, although these could still be embedded if an input-output or reward setting. There can also be objections about what a universal test should look like (Dowe and Hernández-Orallo 2014). But a more integrated view of cognitive abilities for humans, animals, robots, agents, 'animats', hybrids, swarms, etc., is not only possible but useful.

3.5 Highlights and directions of the evaluation of general-purpose AI systems

The previous subsections take ideas from areas outside artificial intelligence (e.g., psychology and information theory) and suggest their use for a more ability-oriented evaluation of AI systems. This may give the impression that there have not been efforts to evaluate more abstract abilities (and general-purpose competence) in AI. In this section we briefly discuss some of the most promising areas where this kind of evaluation has been attempted, its limitations and how it can be merged with the ideas seen in the previous subsections.

Machine learning was discussed in Sect. 2.3, as a kind of task-oriented evaluation. We already saw that some areas in machine learning, such as reinforcement learning, and some recent proposals and competitions such as the Arcade Learning Environment (Bellemare et al. 2013) and the General Video Game Competition (Perez et al. 2015) go in the direction of more general systems. The basic idea is that by combining an evaluation based on a utility function that depends on the task at hand (as usual in reinforcement learning) and the use of *many* tasks, by aggregation, some kind of more general ability is expected to be measured. Related to reinforcement learning, new platforms and projects are sprouting every year, with an extra focus on general-purpose evaluation. For instance, Project Malmo[30] is a platform developed by Microsoft over the well-known video game sandbox Minecraft, where different tasks can

[30] https://www.microsoft.com/en-us/research/project/project-malmo/.

be created and AI agents can be programmed for them (Johnson et al. 2016; Abel et al. 2016). Another platform that integrates the Atari games in the Arcade Learning Environment but also some continuous control tasks is the OpenAI Gym[31], which also includes interfaces with some deep reinforcement learning libraries (Duan et al. 2016).

Nevertheless, by changing a few minor things in one game or task, we observe that current systems have to learn everything from scratch. Basically, the concepts that are learnt are not sufficiently abstract, or general, to be brought from one situation or another (apart from the fact that the systems are usually rebooted between tasks). Transfer learning (Pan and Yang 2010), seen as "the improvement of learning in a new task through the transfer of knowledge from a related task that has already been learned" (Torrey and Shavlik 2009), is presented as a solution for this. Other related areas are domain adaptation (Jiang 2008), multi-task learning (Caruana 1997; Thrun 1996), lifelong learning (Thrun and Pratt 2012), incremental learning (Khreich et al. 2012) and active learning (Settles 2012). These areas have been very active in the past two decades, and there is already a number of benchmarks, competitions and challenges. However, the challenges (such as the 'unsupervised and transfer learning challenge' Mesnil et al. 2012) usually evaluate teams instead of systems, or the systems are restricted to some kind of specific technology.

The evaluation metrics in most of these challenges are defined in terms of performance, as there is no easy way to check how able the system is to build new concepts that can be generalised between tasks, or how previous information is used. In order to do this, other areas of machine learning, such as inductive programming (Gulwani et al. 2015) may be more advantageous, because several systems are general-purpose but also based on declarative languages, so that the invented abstract constructs can be inspected. Hence, it is easier to analyse what kind of concepts are required from some tasks for other tasks. Some evaluation benchmarks along these lines have been proposed around a repository[32], but they are still in an incipient stage. One interesting point of this approach is that the systems are not anthropocentric; they are just conceptual learners. Combining the previous notions of transfer learning, multi-task learning and incremental learning with constructive (Turing-complete) languages, there is the question of how to generate a set of tasks that depend on the creation of previous concepts, such as, for instance, learning multiplication after learning addition. This can be well modelled in inductive programming in such a way that the difficulty of the transfer or abstraction needed can be measured with a principled approach such as algorithmic information theory (instead of a distortion-based approach or a subjective evaluation of the conceptual jump according to the difficulty found by humans). Similarly, the way in which concepts are learnt from natural language in a continual way, as in NELL (Carlson et al. 2010), is closely related to the analysis of whether the system is able to capture more complex and abstract concepts incrementally.

Developmental robotics (Cangelosi et al. 2015) is an area that is particularly focused on the progress of an AI system from sensorimotor interaction towards more abstract concepts, in an incremental way. At the current stage of the area, there are not many objective tests, benchmarks and competitions to evaluate cognitive development in general, other than some qualitative assessments (e.g., Tables 6.4, 7.5 and 8.2 in Cangelosi et al. 2015), which are usually very antropocentric. Quantitative tests focus mostly on the sensorimotor part.[33] The RoCKIn competitions (Amigoni et al. 2015), and the associated challenges (RoCKIn@home and RoCKIn@work, even if they define specific benchmarks such as the assistance of aging

[31] https://gym.openai.com/.

[32] http://www.inductive-programming.org/repository.html.

[33] http://www.robot.uji.es/EURON/en/index.htm, http://rockinrobotchallenge.eu/Benchmarking_Robotics.

people in a house environment or the assembling of mechanical parts in a factory, distinguish between tasks and functionalities (such as object perception, navigation, speech understanding and object manipulation). However, none of these tasks or evaluation frameworks fully delve into the concept development part, in terms of a proper cumulative or lifelong learning. Going beyond "one cognitive faculty" (Cangelosi et al. 2015, p. 272) is one of the future challenges.

Finally, cognitive architectures is one of the areas in artificial intelligence (or more precisely, in cognitive science) that has been traditionally associated with the creation of general-purpose systems, with a clear orientation in modelling human performance (so they are clearly anthropocentric), although for some of them the goal has been to really display some intelligence behaviour in general, which is nowadays more generally referred to as (interactive) "cognitive sytems" (Langley 2012). Let us have a look at the way evaluation is performed in this area.

First, there are several ways of comparing—at the conceptual level— the existing cognitive architecutres (e.g., Sloman and Scheutz 2002). However, we are interested in integrated evaluation benchmarks, especially those that can identify abilities. For instance, the DARPA's Brain-Inspired—later Biologically-Inspired—Cognitive Architectures program (BICA) maintains a comparison table of architectures.[34] There, we can find several "common general paradigms" that can be used to compare them, such as problem solving, decision making, analogy, language processing, working memory, perceptual illusions, implicit memory, metacognition, social tasks, personality and motivational dynamics. A similar list also classifying the existing architectures can be found at the CogArch webpage.[35] Similarly, the 'Newell test' (Anderson and Lebiere 2003), which is not an actual test but a set of criteria for architectures or theories of cognition, distilled into 12 criteria for cognition: "flexible behavior, real-time performance, adaptive behavior, vast knowledge base, dynamic behavior, knowledge integration, natural language, learning, development, evolution, and brain realization", integrating two lists from (Newell 1980, 1990), hence the name. Adams et al. (2012) identify several 'competency areas' for an AGI system: perception, memory, attention, social interaction, planning, motivation, actuation, reasoning, communication, learning, emotion, modelling self/other, building/creation and use of quantities, many of which match with subdisciplines in AI.

Focusing on tests instead of taxonomies, the most remarkable proposal is the so-called 'cognitive decathlon', briefly suggested by Vere (1992), and also linked to the Newell test by Anderson and Lebiere (2003). The actual cognitive decathlon was completed by Mueller et al. (2007), with two other tests (Challenge Scenarios and Biovalidity Assessment) that jointly "cover a core set of cognitive, perceptual, and motor skills typical for a 2-year-old human child". The Cognitive Decathlon features 25 'levels' in six categories: vision, search, manual control and learning, knowledge learning, language and concept learning, and simple motor control. There is a partial implementation of the cognitive decathlon (Mueller 2010), but no systematic evaluation has been performed using it.

The cognitive decathlon is a very interesting case because there is a clear mapping between the abilities and the proposed tests. This is not the case for other approaches. For instance, the competencies mentioned above by Adams et al. (2012) are replaced by different test scenarios that are assumed to cover all the competencies: general video-game learning, preschool learning, reading comprehension, story or scene comprehension, school learning and the 'Wozniak Test' (walk into an unfamiliar house and make a cup of coffee). More recently, the

[34] http://bicasociety.org/cogarch/architectures.htm.

[35] http://cogarch.org/index.php/Capabilities.

so-called "I-athlon" (Adams et al. 2016) also groups a set of tasks, but taking more inspiration from the state of the art in AI than developmental psychology. However, an ability-oriented evaluation is lost after these integrations. Basically, this is the trivial way of preventing specialisation: including many tasks in a battery and increasing the *breadth* of the tasks (Goertzel et al. 2009; Wang 2010; Rohrer 2010). But test generalisation is not the same as an ability-oriented evaluation.

Overall, we have seen that ability-oriented and general-purpose evaluation approaches have both been pursued in artificial intelligence, but the actual proposals are still very incipient, and more research and discussion is needed.[36] They will require some of the ideas seen in the previous subsections for a more foundational approach in the way abilities are identified and their difficulty quantified, especially as many of these systems cannot be evaluated with anthropocentric tests. Also, we have seen that general-purpose evaluation is not only about autonomous robots, agents, cognitive architectures or other kinds of cognitive systems, but it can also be applied to non-interactive systems, such as inductive programming and other constructive learning approaches, which of course lack other abilities that are required in an interactive world.

4 Lessons learnt and guidelines for AI practitioners

There is a (hierarchical) continuum between task-oriented evaluation and ability-oriented evaluation. As a result, it would be misleading to consider that the progress in AI should be analysed by the progress of specific systems (as it is usually done and mentioned in the previous section) but it would also be a mistake to evaluate AI exclusively by the progress of general-purpose systems. Populating the space of benchmarks, tests and competitions is a kind of work that, to an extent, started many decades ago. However, the analysis of one particular test is of limited insight for AI researchers if there is a complete lack of understanding of the relationships of tasks and abilities in this space. For instance, in this continuum of benchmarks, we need to realise that if "one system demonstrates impressive natural language processing and another demonstrates impressive perception [this] does not imply that these capabilities can be integrated in order to automate the perceptual and linguistic aspects" (Brundage 2016). The way some tasks and abilities are related in humans is not expected to extrapolate to AI. Also, the way many subareas in AI have developed may reflect the pace in which the applications and technologies appeared historically in AI, and may not reflect a fundamental way in which AI system should be evaluated. In other words, both natural intelligence research and artificial intelligence research have been highly anthropocentric in the creation of benchmarks and the analysis of their relationship. This means that whenever a group of AI researchers think of creating a new competition, extreme care should be put on not reusing previous benchmarks or tests (from psychology or AI) if it is not clear what the tasks are really measuring and how they relate to other tasks.

Another important lesson learnt when one analyses the myriad of benchmarks and competitions in AI is that it is not always clear *the subject* we are evaluating, an AI system, some of its components or the way a team of AI practitioners have put everthing together to score well for the competition. As we have mentioned in this paper, many competitions, especially in machine learning (and related areas such as data mining and data science), evaluate teams who are able to integrate many AI tools, by thoroughly studying the problem. It is the human intelligence of the team what these competitions usually award, rather than the real value of

[36] http://users.dsic.upv.es/~flip/EGPAI2016/.

the tools used. This relates to our initial discussion that evaluating components instead of systems is more prone to misinterpretations since components must be integrated by an AI practitioner.

Even if the benchmark, test or competition aims at evaluating AI systems (and not AI teams), the most important issue is "evaluation overfitting", which is usually counteracted by changes in the evaluation year after year, or the addition of more tasks, instead of properly analysing why this overfitting happens. Even with these tactics, AI practitioners can still use a "big switch" approach and get good results on average.

Of more logistic character, a general comment shared by their organisers is that competitions are hard to organise and maintain. Technology changes quickly, and servers, platforms must be updated almost every year. It is also difficult to encourage participation, given the large number of competitions. As a result, the integration of many benchmarks and competitions and the support from prestigious organisations and governments is key so that the best AI researchers are involved in creating the future evaluation initiatives and the best AI researchers also participate in them.

Given the caveats mentioned above about AI evaluation, we now enumerate a series of generic guidelines for AI practitioners willing to create or improve a benchmark or competition:

- The definition of Ω, the set of possible systems that can be evaluated (or that can be opponents in peer confrontation evaluation), must be clarified from the beginning, as well as whether the AI systems are fully autonomous or require the integration and finetuning of AI researchers. If humans are considered, the way in which they are admitted and how they are instructed must be defined. The more general Ω is the less we can assume about the evaluation process. If Ω is heterogeneous (e.g., a universal test), different interfaces must be considered.
- The definition of M, the set of possible tasks, and its associated distribution p configure what we are measuring. This can be built from a set of problems or using a generator. This pair $\langle M, p \rangle$ has to be representative of a task (in task-oriented evaluation) or an ability (in ability-oriented evaluation). If it is a peer confrontation evaluation, M will be enlarged with as many combinations between game (environment) and agents in Ω are possible. The distribution p will be updated accordingly.
- The definition of R and its aggregation Φ must ensure that the values $R(\mu)$ for all $\mu \in M$ are going to be commensurate and that the aggregation is bounded. An analysis about expected measurement error is useful at this point. The robustness of R depending on the length or time left for each episode will indicate whether repetitions are needed to reduce the measurement error given by $R(\mu)$.
- As much as possible, the similarity between tasks or a set of features describing them should be identified. An intrinsic difficulty function (even if approximate) is always very useful. Showing the distribution of difficulty for M can be highly informative. If difficulty is available, item response curves could be prepared.
- The sampling method must be as efficient as possible, by using, e.g., an information-driven sampling or a range of difficulties if we have a non-adaptive evaluation. For the peer-confrontation evaluation, the arrangement of matches can be designed beforehand if the evaluation is not adaptive. Similarly, the procedure for an adaptive evaluation must also be carefully designed to ensure measurement robustness. Simulations can be useful to estimate this.
- Information about how the evaluation is performed (including R, Φ and some illustrative problems) can be disclosed to the systems that are being evaluated (or to their designers).

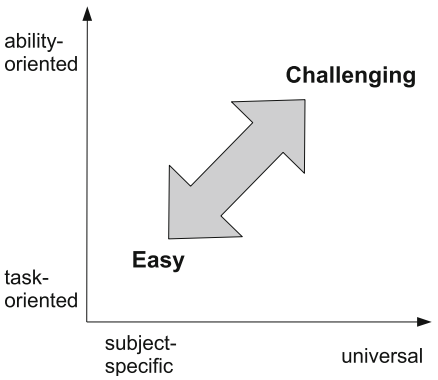

Fig. 7 Tests become more or less challenging depending on the generality of the class of subjects considered (from subject-specific to universal) and the class of abilities (from task-oriented evaluation to ability-oriented evaluation)

However, Ω, M and p should not be disclosed. If possible, the problems should not be disclosed after the evaluation either, as keeping them secret makes it possible to compare with the same problems for different subjects or at different times (e.g., we can evaluate progress of a system or a discipline during a period).

– After the evaluation, results must be analysed beyond the mere calculation of the aggregated results. Item response functions and agent response functions (Hernández-Orallo et al. 2014) can be constructed empirically from the results and compared with the theoretical functions or any other information about Ω and M. Discrepancies or anomalies may suggest that the evaluation scheme has to be revised. Results of the evaluation must become public at the highest possible detail, so they can be analysed and compared by other researchers and participants (following, e.g., the notion of 'experiment database' (Vanschoren et al. 2012), such as in the machine learning community).[37]

The above list is very different from a recent list of "principles for designing an AI competition" such that they induce real progress in AI (Shieber 2016). Shieber is more elementary and general in the desiderata, focusing on the rules of the competition ('occasionality', 'flexibility' and 'absoluteness') and the nature of the task ('reasonableness'). It is of course an open question to what extent any or both of the above recommendation lists will be followed on a regular basis for AI evaluation. It can be argued whether AI evaluation has been a priority for AI in the past, but it seems that it has not been recognised as an imperative problem or a mainstream area of research. Anyhow, the question of AI evaluation remains and there is space for significant improvement, even for the most specific sets Ω and M (bottom-left part of Fig. 7). At the other end, measuring intelligence and doing it universally is a key ingredient for understanding what intelligence is (and, of course, to devise intelligent artefacts). Many interesting questions and applications lay in the middle of Fig. 7, as AI evaluation is no longer limited to task-specific evaluation of AI systems or to evaluating progress in AI. Instead, AI is becoming able to evaluate systems that learn to solve instead of systems that are programmed to solve.

5 Conclusions

We started this paper looking at the way AI evaluation is commonly performed, through task-oriented evaluation, mostly with a black-box approach. We identified several problems

[37] http://openml.org/.

and limitations, and we noticed that there is still a huge margin for improvement available in the way AI systems are evaluated. The key issues are the set of tasks M and their distribution p, as well as distinguishing the definition of the problem class (aggregation) from an effective sampling procedure (testing procedure). Then we switched to ability-oriented evaluation, a much more immature approach, but that may have a more relevant role in the future. The notion and evaluation of abilities is more elusive than the notion and evaluation of tasks. We have argued that this requires the integration of several perspectives that are currently scattered efforts in AI, psychometrics, AIT and comparative cognition. The different areas, philosophies, tools, foundations, terminologies and the different kinds of subjects to be evaluated can be unified with an integrated perspective known as universal psychometrics. Here, the exploration of the machine kingdom mirrors the exploration of the set of possible cognitive abilities/tasks, from the duality environment-agent (task-subject). In both spaces we aim at becoming more general, which is where evaluation is more challenging (see Fig. 7). This resembles the duality in the theory of computation (e.g., problem classes and automata classes).

In any case, and with any of the approaches seen so far, a more scientific theory of AI evaluation is being required for many applications (CAPTCHAs, social networks, agent certification, etc.) and it will be more and more common in a future with a plethora of bots, robots, artificial agents, avatars, control systems, 'animats', hybrids, collectives, etc. It is also crucial for alternatives to the concept of 'human-level intelligence' and for the assessement of the plausibility of futuristic ideas such as 'the technological singularity' (Eden et al. 2013), especially because some of the prophecies and forecasts disregard that the first thing to consider about the future of intelligence is to have metrics to detect whether AI progresses and in which direction.

Summing up, AI requires an accurate, effective, non-anthropocentric, meaningful and computational way of evaluating its progress, by evaluating its artefacts. This paper has just portrayed the state of the art of AI evaluation, its challenges and the avenues for future work.

Acknowledgments I thank the organisers of the AEPIA Summer School On Artificial Intelligence, held in September 2014, for giving me the opportunity to give a lecture on 'AI Evaluation'. This paper was born out of and evolved through that lecture. The information about many benchmarks and competitions discussed in this paper have been contrasted with information from and discussions with many people: M. Bedia, A. Cangelosi, C. Dimitrakakis, I. GarcÍa-Varea, Katja Hofmann, W. Langdon, E. Messina, S. Mueller, M. Siebers and C. Soares. Figure 4 is courtesy of F. Martínez-Plumed. Finally, I thank the anonymous reviewers, whose comments have helped to significantly improve the balance and coverage of the paper. This work has been partially supported by the EU (FEDER) and the Spanish MINECO under Grants TIN 2013-45732-C4-1-P, TIN 2015-69175-C4-1-R and by Generalitat Valenciana PROMETEOII2015/013.

References

Abel D, Agarwal A, Diaz F, Krishnamurthy A, Schapire RE (2016) Exploratory gradient boosting for reinforcement learning in complex domains. arXiv preprint arXiv:1603.04119

Adams S, Arel I, Bach J, Coop R, Furlan R, Goertzel B, Hall JS, Samsonovich A, Scheutz M, Schlesinger M, Shapiro SC, Sowa J (2012) Mapping the landscape of human-level artificial general intelligence. AI Mag 33(1):25–42

Adams SS, Banavar G, Campbell M (2016) I-athlon: towards a multi-dimensional Turing test. AI Mag 37(1):78–84

Alcalá J, Fernández A, Luengo J, Derrac J, García S, Sánchez L, Herrera F (2010) Keel data-mining software tool: data set repository, integration of algorithms and experimental analysis framework. J Mult Valued Logic Soft Comput 17:255–287

Alexander JRM, Smales S (1997) Intelligence, learning and long-term memory. Personal Individ Differ 23(5):815–825

Alpcan T, Everitt T, Hutter M (2014) Can we measure the difficulty of an optimization problem? In: IEEE information theory workshop (ITW)
Alur R, Bodik R, Juniwal G, Martin MMK, Raghothaman M, Seshia SA, Singh R, Solar-Lezama A, Torlak E, Udupa A (2013) Syntax-guided synthesis. In: Formal methods in computer-aided design (FMCAD), 2013, IEEE, pp 1–17
Alvarado N, Adams SS, Burbeck S, Latta C (2002) Beyond the Turing test: performance metrics for evaluating a computer simulation of the human mind. In: Proceedings of the 2nd international conference on development and learning, IEEE, pp 147–152
Amigoni F, Bastianelli E, Berghofer J, Bonarini A, Fontana G, Hochgeschwender N, Iocchi L, Kraetzschmar G, Lima P, Matteucci M, Miraldo P, Nardi D, Schiaffonati V (2015) Competitions for benchmarking: task and functionality scoring complete performance assessment. IEEE Robot Autom Mag 22(3):53–61
Anderson J, Lebiere C (2003) The Newell test for a theory of cognition. Behav Brain Sci 26(5):587–601
Anderson J, Baltes J, Cheng CT (2011) Robotics competitions as benchmarks for AI research. Knowl Eng Rev 26(01):11–17
Arel I, Rose DC, Karnowski TP (2010) Deep machine learning—a new frontier in artificial intelligence research. IEEE Comput Intell Mag 5(4):13–18
Asada M, Hosoda K, Kuniyoshi Y, Ishiguro H, Inui T, Yoshikawa Y, Ogino M, Yoshida C (2009) Cognitive developmental robotics: a survey. IEEE Trans Auton Ment Dev 1(1):12–34
Aziz H, Brill M, Fischer F, Harrenstein P, Lang J, Seedig HG (2015) Possible and necessary winners of partial tournaments. J Artif Intell Res 54:493–534
Bache K, Lichman M (2013) UCI machine learning repository. http://archive.ics.uci.edu/ml
Bagnall AJ, Zatuchna ZV (2005) On the classification of maze problems. In: Bull L, Kovacs T (eds) Foundations of learning classifier system. Studies in fuzziness and soft computing, vol. 183, Springer, pp 305–316. http://rd.springer.com/chapter/10.1007/11319122_12
Baldwin D, Yadav SB (1995) The process of research investigations in artificial intelligence - a unified view. IEEE Trans Syst Man Cybern 25(5):852–861
Bellemare MG, Naddaf Y, Veness J, Bowling M (2013) The arcade learning environment: an evaluation platform for general agents. J Artif Intell Res 47:253–279
Besold TR (2014) A note on chances and limitations of psychometric ai. In: KI 2014: advances in artificial intelligence. Springer, pp 49–54
Biever C (2011) Ultimate IQ: one test to rule them all. New Sci 211(2829, 10 September 2011):42–45
Borg M, Johansen SS, Thomsen DL, Kraus M (2012) Practical implementation of a graphics Turing test. In: Advances in visual computing. Springer, pp 305–313
Boring EG (1923) Intelligence as the tests test it. New Repub 35–37
Bostrom N (2014) Superintelligence: paths, dangers, strategies. Oxford University Press, Oxford
Brazdil P, Carrier CG, Soares C, Vilalta R (2008) Metalearning: applications to data mining. Springer, New York
Bringsjord S (2011) Psychometric artificial intelligence. J Exp Theor Artif Intell 23(3):271–277
Bringsjord S, Schimanski B (2003) What is artificial intelligence? Psychometric AI as an answer. In: International joint conference on artificial intelligence, pp 887–893
Brundage M (2016) Modeling progress in ai. AAAI 2016 Workshop on AI, Ethics, and Society
Buchanan BG (1988) Artificial intelligence as an experimental science. Springer, New York
Buhrmester M, Kwang T, Gosling SD (2011) Amazon's mechanical turk a new source of inexpensive, yet high-quality, data? Perspect Psychol Sci 6(1):3–5
Bursztein E, Aigrain J, Moscicki A, Mitchell JC (2014) The end is nigh: generic solving of text-based captchas. In: Proceedings of the 8th USENIX conference on Offensive Technologies, USENIX Association, p 3
Campbell M, Hoane AJ, Hsu F (2002) Deep Blue. Artif Intell 134(1–2):57–83
Cangelosi A, Schlesinger M, Smith LB (2015) Developmental robotics: from babies to robots. MIT Press, Cambridge
Caputo B, Müller H, Martinez-Gomez J, Villegas M, Acar B, Patricia N, Marvasti N, Üsküdarlı S, Paredes R, Cazorla M et al (2014) Imageclef 2014: overview and analysis of the results. In: Information access evaluation. Multilinguality, multimodality, and interaction, Springer, pp 192–211
Carlson A, Betteridge J, Kisiel B, Settles B, Hruschka ER Jr, Mitchell TM (2010) Toward an architecture for never-ending language learning. In: AAAI, vol 5, p 3
Carroll JB (1993) Human cognitive abilities: a survey of factor-analytic studies. Cambridge University Press, Cambridge
Caruana R (1997) Multitask learning. Mach Learn 28(1):41–75
Chaitin GJ (1982) Gödel's theorem and information. Int J Theor Phys 21(12):941–954
Chandrasekaran B (1990) What kind of information processing is intelligence? In: The foundation of artificial intelligence—a sourcebook. Cambridge University Press, pp 14–46

Chater N (1999) The search for simplicity: a fundamental cognitive principle? Q J Exp Psychol Sect A 52(2):273–302
Chater N, Vitányi P (2003) Simplicity: a unifying principle in cognitive science? Trends Cogn Sci 7(1):19–22
Chu Z, Gianvecchio S, Wang H, Jajodia S (2010) Who is tweeting on twitter: human, bot, or cyborg? In: Proceedings of the 26th annual computer security applications conference, ACM, pp 21–30
Cochran WG (2007) Sampling techniques. Wiley, New York
Cohen PR, Howe AE (1988) How evaluation guides AI research: the message still counts more than the medium. AI Mag 9(4):35
Cohen Y (2013) Testing and cognitive enhancement. Technical repor, National Institute for Testing and Evaluation, Jerusalem, Israel
Conrad JG, Zeleznikow J (2013) The significance of evaluation in AI and law: a case study re-examining ICAIL proceedings. In: Proceedings of the 14th international conference on artificial intelligence and law, ACM, pp 186–191
Conrad JG, Zeleznikow J (2015) The role of evaluation in ai and law. In: Proceedings of the 15th international conference on artificial intelligence and law, pp 181–186
Deary IJ, Der G, Ford G (2001) Reaction times and intelligence differences: a population-based cohort study. Intelligence 29(5):389–399
Decker KS, Durfee EH, Lesser VR (1989) Evaluating research in cooperative distributed problem solving. Distrib Artif Intell 2:487–519
Demšar J (2006) Statistical comparisons of classifiers over multiple data sets. J Mach Learn Res 7:1–30
Detterman DK (2011) A challenge to Watson. Intelligence 39(2–3):77–78
Dimitrakakis C (2016) Personal communication
Dimitrakakis C, Li G, Tziortziotis N (2014) The reinforcement learning competition 2014. AI Mag 35(3):61–65
Dowe DL (2013) Introduction to Ray Solomonoff 85th memorial conference. In: Dowe DL (ed) Algorithmic probability and friends. Bayesian prediction and artificial intelligence, lecture notes in computer science, vol 7070. Springer, Berlin, pp 1–36
Dowe DL, Hajek AR (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
Dowe DL, Hajek AR (1998) A non-behavioural, computational extension to the Turing test. In: International conference on computational intelligence and multimedia applications (ICCIMA'98), Gippsland, Australia, pp 101–106
Dowe DL, Hernández-Orallo J (2012) IQ tests are not for machines, yet. Intelligence 40(2):77–81
Dowe DL, Hernández-Orallo J (2014) How universal can an intelligence test be? Adapt Behav 22(1):51–69
Drummond C (2009) Replicability is not reproducibility: nor is it good science. In: Proceedings of the evaluation methods for machine learning workshop at the 26th ICML, Montreal, Canada
Drummond C, Japkowicz N (2010) Warning: statistical benchmarking is addictive. Kicking the habit in machine learning. J Exp Theor Artif Intell 22(1):67–80
Duan Y, Chen X, Houthooft R, Schulman J, Abbeel P (2016) Benchmarking deep reinforcement learning for continuous control. arXiv preprint arXiv:1604.06778
Eden AH, Moor JH, Soraker JH, Steinhart E (2013) Singularity hypotheses: a scientific and philosophical assessment. Springer, New York
Edmondson W (2012) The intelligence in ETI—what can we know? Acta Astronaut 78:37–42
Elo AE (1978) The rating of chessplayers, past and present, vol 3. Batsford, London
Embretson SE, Reise SP (2000) Item response theory for psychologists. L. Erlbaum, Hillsdale
Evans JM, Messina ER (2001) Performance metrics for intelligent systems. NIST Special Publication SP, pp 101–104
Everitt T, Lattimore T, Hutter M (2014) Free lunch for optimisation under the universal distribution. In: 2014 IEEE Congress on evolutionary computation (CEC), IEEE, pp 167–174
Falkenauer E (1998) On method overfitting. J Heuristics 4(3):281–287
Feldman J (2003) Simplicity and complexity in human concept learning. Gen Psychol 38(1):9–15
Ferrando PJ (2009) Difficulty, discrimination, and information indices in the linear factor analysis model for continuous item responses. Appl Psychol Meas 33(1):9–24
Ferrando PJ (2012) Assessing the discriminating power of item and test scores in the linear factor-analysis model. Psicológica 33:111–139
Ferri C, Hernández-Orallo J, Modroiu R (2009) An experimental comparison of performance measures for classification. Pattern Recogn Lett 30(1):27–38
Ferrucci D, Brown E, Chu-Carroll J, Fan J, Gondek D, Kalyanpur AA, Lally A, Murdock J, Nyberg E, Prager J et al (2010) Building Watson: an overview of the DeepQA project. AI Mag 31(3):59–79
Fogel DB (1991) The evolution of intelligent decision making in gaming. Cybern Syst 22(2):223–236

Gaschnig J, Klahr P, Pople H, Shortliffe E, Terry A (1983) Evaluation of expert systems: issues and case studies. Build Exp Syst 1:241–278
Geissman JR, Schultz RD (1988) Verification & validation. AI Exp 3(2):26–33
Genesereth M, Love N, Pell B (2005) General game playing: overview of the AAAI competition. AI Mag 26(2):62
Gerónimo D, López AM (2014) Datasets and benchmarking. In: Vision-based pedestrian protection systems for intelligent vehicles. Springer, pp 87–93
Goertzel B, Pennachin C (eds) (2007) Artificial general intelligence. Springer, New York
Goertzel B, Arel I, Scheutz M (2009) Toward a roadmap for human-level artificial general intelligence: embedding HLAI systems in broad, approachable, physical or virtual contexts. Artif Gen Intell Roadmap Initiat
Goldreich O, Vadhan S (2007) Special issue on worst-case versus average-case complexity editors' foreword. Comput complex 16(4):325–330
Gordon BB (2007) Report on panel discussion on (re-)establishing or increasing collaborative links between artificial intelligence and intelligent systems. In: Messina ER, Madhavan R (eds) Proceedings of the 2007 workshop on performance metrics for intelligent systems, pp 302–303
Gulwani S, Hernández-Orallo J, Kitzelmann E, Muggleton SH, Schmid U, Zorn B (2015) Inductive programming meets the real world. Commun ACM 58(11):90–99
Hand DJ (2004) Measurement theory and practice. A Hodder Arnold Publication, London
Hernández-Orallo J (2000a) Beyond the Turing test. J Logic Lang Inf 9(4):447–466
Hernández-Orallo J (2000b) On the computational measurement of intelligence factors. In: Meystel A (ed) Performance metrics for intelligent systems workshop. National Institute of Standards and Technology, Gaithersburg, pp 1–8
Hernández-Orallo J (2000c) Thesis: computational measures of information gain and reinforcement in inference processes. AI Commun 13(1):49–50
Hernández-Orallo J (2010) A (hopefully) non-biased universal environment class for measuring intelligence of biological and artificial systems. In: Artificial general intelligence, 3rd International Conference. Atlantis Press, Extended report at http://users.dsic.upv.es/proy/anynt/unbiased.pdf, pp 182–183
Hernández-Orallo J (2014) On environment difficulty and discriminating power. Auton Agents Multi-Agent Syst. 29(3):402–454. doi:10.1007/s10458-014-9257-1
Hernández-Orallo J, Dowe DL (2010) Measuring universal intelligence: towards an anytime intelligence test. Artif Intell 174(18):1508–1539
Hernández-Orallo J, Dowe DL (2013) On potential cognitive abilities in the machine kingdom. Minds Mach 23:179–210
Hernández-Orallo J, Minaya-Collado N (1998) A formal definition of intelligence based on an intensional variant of Kolmogorov complexity. In: Proceedings of international symposium of engineering of intelligent systems (EIS'98), ICSC Press, pp 146–163
Hernández-Orallo J, Dowe DL, España-Cubillo S, Hernández-Lloreda MV, Insa-Cabrera J (2011) On more realistic environment distributions for defining, evaluating and developing intelligence. In: Schmidhuber J, Thórisson K, Looks M (eds) Artificial general intelligence, LNAI, vol 6830. Springer, New York, pp 82–91
Hernández-Orallo J, Flach P, Ferri C (2012a) A unified view of performance metrics: translating threshold choice into expected classification loss. J Mach Learn Res 13(1):2813–2869
Hernández-Orallo J, Insa-Cabrera J, Dowe DL, Hibbard B (2012b) Turing Tests with Turing machines. In: Voronkov A (ed) Turing-100, EPiC Series, vol 10, pp 140–156
Hernández-Orallo J, Dowe DL, Hernández-Lloreda MV (2014) Universal psychometrics: measuring cognitive abilities in the machine kingdom. Cogn Syst Res 27:50–74
Hernández-Orallo J, Martínez-Plumed F, Schmid U, Siebers M, Dowe DL (2016) Computer models solving intelligence test problems: progress and implications. Artif Intell 230:74–107
Herrmann E, Call J, Hernández-Lloreda MV, Hare B, Tomasello M (2007) Humans have evolved specialized skills of social cognition: the cultural intelligence hypothesis. Science 317(5843):1360–1366
Hibbard B (2009) Bias and no free lunch in formal measures of intelligence. J Artif Gen Intell 1(1):54–61
Hingston P (2010) A new design for a Turing Test for bots. In: 2010 IEEE symposium on computational intelligence and games (CIG), IEEE, pp 345–350
Hingston P (2012) Believable bots: can computers play like people?. Springer, New York
Ho TK, Basu M (2002) Complexity measures of supervised classification problems. IEEE Trans Pattern Anal Mach Intell 24(3):289–300
Hutter M (2007) Universal algorithmic intelligence: a mathematical top→down approach. In: Goertzel B, Pennachin C (eds) Artificial general intelligence, cognitive technologies. Springer, Berlin, pp 227–290

Igel C, Toussaint M (2005) A no-free-lunch theorem for non-uniform distributions of target functions. J Math Model Algorithms 3(4):313–322
Insa-Cabrera J (2016) Towards a universal test of social intelligence. Ph.D. thesis, Departament de Sistemes Informátics i Computació, UPV
Insa-Cabrera J, Dowe DL, España-Cubillo S, Hernández-Lloreda MV, Hernández-Orallo J (2011a) Comparing humans and ai agents. In: Schmidhuber J, Thórisson K, Looks M (eds) Artificial general intelligence, LNAI, vol 6830. Springer, New York, pp 122–132
Insa-Cabrera J, Dowe DL, Hernández-Orallo J (2011) Evaluating a reinforcement learning algorithm with a general intelligence test. In: Lozano JA, Gamez JM (eds) Current topics in artificial intelligence. CAEPIA 2011, LNAI series 7023. Springer, New York
Insa-Cabrera J, Benacloch-Ayuso JL, Hernández-Orallo J (2012) On measuring social intelligence: experiments on competition and cooperation. In: Bach J, Goertzel B, Iklé M (eds) AGI, lecture notes in computer science, vol 7716. Springer, New York, pp 126–135
Jacoff A, Messina E, Weiss BA, Tadokoro S, Nakagawa Y (2003) Test arenas and performance metrics for urban search and rescue robots. In: Proceedings of 2003 IEEE/RSJ international conference on intelligent robots and systems, 2003 (IROS 2003), IEEE, vol 4, pp 3396–3403
Japkowicz N, Shah M (2011) Evaluating learning algorithms. Cambridge University Press, Cambridge
Jiang J (2008) A literature survey on domain adaptation of statistical classifiers. http://sifaka.cs.uiuc.edu/jiang4/domain_adaptation/survey
Johnson M, Hofmann K, Hutton T, Bignell D (2016) The Malmo platform for artificial intelligence experimentation. In: International joint conference on artificial intelligence (IJCAI)
Keith TZ, Reynolds MR (2010) Cattell–Horn–Carroll abilities and cognitive tests: what we've learned from 20 years of research. Psychol Schools 47(7):635–650
Ketter W, Symeonidis A (2012) Competitive benchmarking: lessons learned from the trading agent competition. AI Mag 33(2):103
Khreich W, Granger E, Miri A, Sabourin R (2012) A survey of techniques for incremental learning of HMM parameters. Inf Sci 197:105–130
Kim JH (2004) Soccer robotics, vol 11. Springer, New York
Kitano H, Asada M, Kuniyoshi Y, Noda I, Osawa E (1997) Robocup: the robot world cup initiative. In: Proceedings of the first international conference on autonomous agents, ACM, pp 340–347
Kleiner K (2011) Who are you calling bird-brained? An attempt is being made to devise a universal intelligence test. Economist 398(8723, 5 March 2011):82
Knuth DE (1973) Sorting and searching, volume 3 of the art of computer programming. Addison-Wesley, Reading
Koza JR (2010) Human-competitive results produced by genetic programming. Genet Program Evolvable Mach 11(3–4):251–284
Krueger J, Osherson D (1980) On the psychology of structural simplicity. In: Jusczyk PW, Klein RM (eds) The nature of thought: essays in honor of D. O. Hebb. Psychology Press, London, pp 187–205
Langford J (2005) Clever methods of overfitting. Machine Learning (Theory). http://hunch.net
Langley P (1987) Research papers in machine learning. Mach Learn 2(3):195–198
Langley P (2011) The changing science of machine learning. Mach Learn 82(3):275–279
Langley P (2012) The cognitive systems paradigm. Adv Cogn Syst 1:3–13
Lattimore T, Hutter M (2013) No free lunch versus Occam's razor in supervised learning. Algorithmic Probability and Friends. Springer, Bayesian Prediction and Artificial Intelligence, pp 223–235
Leeuwenberg ELJ, Van Der Helm PA (2012) Structural information theory: the simplicity of visual form. Cambridge University Press, Cambridge
Legg S, Hutter M (2007a) Tests of machine intelligence. In: Lungarella M, Iida F, Bongard J, Pfeifer R (eds) 50 Years of Artificial Intelligence, Lecture Notes in Computer Science, vol 4850, Springer Berlin Heidelberg, pp 232–242. doi:10.1007/978-3-540-77296-5_22
Legg S, Hutter M (2007b) Universal intelligence: a definition of machine intelligence. Minds Mach 17(4):391–444
Legg S, Veness J (2013) An approximation of the universal intelligence measure. Algorithmic Probability and Friends. Springer, Bayesian Prediction and Artificial Intelligence, pp 236–249
Levesque HJ (2014) On our best behaviour. Artif Intell 212:27–35
Levesque HJ, Davis E, Morgenstern L (2012) The winograd schema challenge. In: Proceedings of the thirteenth international conference on the principles of knowledge representation and reasoning, pp 552–561
Levin LA (1973) Universal sequential search problems. Prob Inf Transm 9(3):265–266
Levin LA (1986) Average case complete problems. SIAM J Comput 15:285–286

Levin LA (2013) Universal heuristics: how do humans solve unsolvable problems? In: Dowe DL (ed) Algorithmic probability and friends. Bayesian prediction and artificial intelligence, lecture notes in computer science, vol 7070. Springer, New York, pp 53–54
Li M, Vitányi P (2008) An introduction to Kolmogorov complexity and its applications, 3rd edn. Springer, New York
Livingstone D (2006) Turing's test and believable AI in games. Comput Entertain CIE 4(1):6
Llargues-Asensio JM, Peralta J, Arrabales R, González-Bedía M, Cortez P, López-Peña AL (2014) Artificial intelligence approaches for the generation and assessment of believable human-like behaviour in virtual characters. Expert Systems with Applications
Long D, Fox M (2003) The 3rd international planning competition: results and analysis. J Artif Intell Res JAIR 20:1–59
Lord FM (1980) Applications of item response theory to practical testing problems. Erlbaum, Mahwah
Macià N, Bernadó-Mansilla E (2014) Towards UCI+: a mindful repository design. Inf Sci 261:237–262
Madhavan R, Tunstel E, Messina E (2009) Performance evaluation and benchmarking of intelligent systems. Springer, New York
Mahoney MV (1999) Text compression as a test for artificial intelligence. In: Proceedings of the national conference on artificial intelligence, AAAI, p 970
Marché C, Zantema H (2007) The termination competition. In: Term rewriting and applications, Springer, pp 303–313
Marcus G, Rossi F, Veloso M (2016) Beyond the Turing test (special issue). AI Mag 37(1):3–101
Masum H, Christensen S (2003) The turing ratio: a framework for open-ended task metrics. J Evol Technol
Masum H, Christensen S, Oppacher F (2002) The turing ratio: metrics for open-ended tasks. In: GECCO, Citeseer, pp 973–980
McCarthy J (2007) What is artificial intelligence. Technical report, Stanford University. http://www-formal.stanford.edu/jmc/whatisai.html
McCorduck P (2004) Machines who think. A K Peters/CRC Press, Boca Raton
McDermott J, White DR, Luke S, Manzoni L, Castelli M, Vanneschi L, Jaśkowski W, Krawiec K, Harper R, Jong KD, O'Reilly UM (2012) Genetic programming needs better benchmarks. In: Proceedings of the 14th international conference on Genetic and evolutionary computation conference. ACM, Philadelphia, pp 791–798
McGuigan M (2006) Graphics Turing Test. arXiv preprint arXiv:cs/0603132
Melkikh AV (2014) The no free lunch theorem and hypothesis of instinctive animal behavior. Artif Intell Res 3(4):p43
Mellenbergh GJ (1994) Generalized linear item response theory. Psychol Bull 115(2):300
Mesnil G, Dauphin Y, Glorot X, Rifai S, Bengio Y, Goodfellow IJ, Lavoie E, Muller X, Desjardins G, Warde-Farley D, et al (2012) Unsupervised and transfer learning challenge: a deep learning approach. JMLR: Workshop and Conference Proceedings, 2012 ICML Workshop on Unsupervised and Transfer Learning vol 27, pp 97–110
Messina E, Meystel A, Reeker L (2001) PerMIS 2001, white paper. In: Meystel AM, Messina ER (eds) Measuring the performance and intelligence of systems: proceedings of the 2001 PerMIS Workshop, September 4, 2001, National Institute of Standards and Technology (NIST) Special Publication 982. Gaithersburg, pp 3–15
Meystel A (2000) Permis 2000 white paper: measuring performance and intelligence of systems with autonomy. In: Meystel AM, Messina ER (eds) Measuring the performance and intelligence of systems: proceedings of the 2000 PerMIS Workshop, August 14–16, 2000, National Institute of Standards and Technology (NIST) Special Publication 970. Gaithersburg, pp 1–34
Meystel A, Albus J, Messina E, Leedom D (2003a) Performance measures for intelligent systems: measures of technology readiness. Technical report, DTIC Document
Meystel A, Albus J, Messina E, Leedom D (2003) Permis 2003 white paper: performance measures for intelligent systems—measures of technology readiness. In: Meystel AM, Messina ER (eds) Measuring the performance and intelligence of systems: proceedings of the 2003 PerMIS Workshop, National Institute of Standards and Technology (NIST) Special Publication 1014. Gaithersburg
Minsky ML (ed) (1968) Semantic information processing. MIT Press, Cambridge
Mnih V, Kavukcuoglu K, Silver D, Rusu AA, Veness J, Bellemare MG, Graves A, Riedmiller M, Fidjeland AK, Ostrovski G et al (2015) Human-level control through deep reinforcement learning. Nature 518(7540):529–533
Morgenstern L, Davis E, Ortiz-Jr CL (2016) Planning, executing, and evaluating the Winograd schema challenge. AI Mag 37(1):50–54

Mueller S, Jones M, Minnery B, Hiland JM (2007) The bica cognitive decathlon: a test suite for biologically-inspired cognitive agents. In: Proceedings of behavior representation in modeling and simulation conference, Norfolk

Mueller ST (2010) A partial implementation of the BICA cognitive decathlon using the psychology experiment building language (PEBL). Int J Mach Conscious 2(02):273–288

Mueller ST, Minnery BS (2008) Adapting the Turing Test for embodied neurocognitive evaluation of biologically-inspired cognitive agents. In: Proceedings of 2008 AAAI fall symposium on biologically inspired cognitive architectures

Newell A (1973) You can't play 20 questions with nature and win: projective comments on the papers of this symposium. In: Chase W (ed) Vis Inf Process. Academic Press, New York, pp 283–308

Newell A (1980) Physical symbol systems. Cogn Sci 4(2):135–183

Newell A (1990) Unified theories of cognition. Harvard University, Cambridge

Newell A, Simon HA (1976) Computer science as empirical inquiry: symbols and search. Commun ACM 19(3):113–126

Nizamani AR (2015) Reasoning with bounded cognitive resources. Ph.D. thesis, Department of Applied Information Technology, Chalmers University of Technology & University of Gothenburg, Sweden

Oppy G, Dowe DL (2011) The Turing Test. In: Zalta EN (ed) Stanford Encyclopedia of Philosophy, Stanford University. http://plato.stanford.edu/entries/turing-test/

Pan SJ, Yang Q (2010) A survey on transfer learning. IEEE Trans Knowl Data Eng 22(10):1345–1359

Perez D, Samothrakis S, Togelius J, Schaul T, Lucas S, Couëtoux A, Lee J, Lim CU, Thompson T (2015) The 2014 general video game playing competition. IEEE Transactions on Computational Intelligence and AI in Games

Potthast M, Hagen M, Gollub T, Tippmann M, Kiesel J, Rosso P, Stamatatos E, Stein B (2013) Overview of the 5th international competition on plagiarism detection. CLEF (2013) Evaluation labs and workshop working notes papers, pp 23–26 September. Valencia, Spain

Proudfoot D (2011) Anthropomorphism and AI: Turing's much misunderstood imitation game. Artif Intell 175(5):950–957

Quinn AJ, Bederson BB (2011) Human computation: a survey and taxonomy of a growing field. In: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, ACM, pp 1403–1412

Rajani S (2011) Artificial intelligence—man or machine. Int J Inf Technol 4(1):173–176

Rao RB, Fung G, Rosales R (2008) On the dangers of cross-validation. an experimental evaluation. In: SDM, SIAM, pp 588–596

Rohrer B (2010) Accelerating progress in artificial general intelligence: choosing a benchmark for natural world interaction. J Artif Gen Intell 2(1):1–28

Rothenberg J, Paul J, Kameny I, Kipps JR, Swenson M (1987) Evaluating expert system tools: a framework and methodology-workshops. Technical report, DTIC Document

Russell S, Norvig P (2009) Artificial intelligence: a modern approach. Prentice Hall, Upper Saddle River

Sanghi P, Dowe DL (2003) A computer program capable of passing IQ tests. In: 4th international conference on cognitive science (ICCS'03), Sydney, pp 570–575

Schaeffer J, Burch N, Bjornsson Y, Kishimoto A, Muller M, Lake R, Lu P, Sutphen S (2007) Checkers is solved. Science 317(5844):1518

Schaie KW (2010) Primary mental abilities. Corsini Encyclopedia of Psychology

Schaul T (2014) An extensible description language for video games. IEEE Trans Comput Intell AI Games PP(99):1–1. doi:10.1109/TCIAIG.2014.2352795

Schenck C (2013) Intelligence tests for robots: Solving perceptual reasoning tasks with a humanoid robot. Master's thesis, Iowa State University

Schlenoff C, Scott H, Balakirsky S (2011) Performance evaluation of intelligent systems at the National Institute of Standards and Technology (NIST). Technical report, DTIC Document

Schmid U, Ragni M (2015) Comparing computer models solving number series problems. In: Artificial general intelligence. Springer, pp 352–361

Schweizer P (1998) The truly total Turing test. Minds Mach 8(2):263–272

Searle JR (1980) Minds, brains, and programs. Behav Brain Sci 3:417–457

Seber GAF, Salehi MM (2013) Adaptive cluster sampling. In: Adaptive sampling designs. Springer, pp 11–26

Settles B (2012) Active learning. Synth Lect Artif Intell Mach Learn 6(1):1–114

Shettleworth SJ (2010) Cognition, evolution, and behavior. Oxford University Press, Oxford

Shettleworth SJ, Bloom P, Nadel L (2013) Fundamentals of comparative cognition. Oxford University Press, Oxford

Shieber SM (2016) Principles for designing an AI competition, or why the Turing test fails as an inducement prize. AI Mag 37(1):91–96

Silver D, Huang A, Maddison CJ, Guez A, Sifre L, Van Den Driessche G, Schrittwieser J, Antonoglou I, Panneershelvam V, Lanctot M et al (2016) Mastering the game of go with deep neural networks and tree search. Nature 529(7587):484–489
Simmons R (2000) Survivability and competence as measures of intelligent systems. In: Meystel AM, Messina ER (eds) Measuring the performance and intelligence of systems: proceedings of the 2000 PerMIS Workshop, August 14–16, 2000, National Institute of Standards and Technology (NIST) Special Publication 970. Gaithersburg, pp 162–163
Simon HA (1995) Artificial intelligence: an empirical science. Artif Intell 77(1):95–127
Sloman A, Scheutz M (2002) A framework for comparing agent architectures. Proceedings of UKCI 2
Smith WD (2002) Rating systems for gameplayers, and learning. NEC, Princeton, NJ, Technical report, pp 93–104
Smith WD (2006) Mathematical definition of "intelligence" (and consequences). Unpublished report
Soares C (2009) UCI++: improved support for algorithm selection using datasetoids. In: Advances in knowledge discovery and data mining. Springer, pp 499–506
Solomonoff R (1996) Does algorithmic probability solve the problem of induction. Inf Stat Induction Sci 7–8
Solomonoff RJ (1964) A formal theory of inductive inference. Part I. Inf Control 7(1):1–22
Solomonoff RJ (1984) Optimum sequential search. Oxbridge Research, Cambridge. http://world.std.com/~rjs/optseq.pdf
Srinivasan R (2002) Importance sampling: applications in communications and detection. Springer, New York
Starkie B, van Zaanen M, Estival D (2006) The Tenjinno machine translation competition. In: Grammatical inference: algorithms and applications. Springer, pp 214–226
Sternberg RJ (ed) (2000) Handbook of intelligence. Cambridge University Press, Cambridge
Strannegård C, Amirghasemi M, Ulfsbücker S (2013a) An anthropomorphic method for number sequence problems. Cogn Syst Res 22–23:27–34
Strannegård C, Nizamani A, Sjöberg A, Engström F (2013b) Bounded Kolmogorov complexity based on cognitive models. In: Kühnberger KU, Rudolph S, Wang P (eds) Artificial general intelligence. Lecture notes in computer science, vol 7999. Springer, Berlin Heidelberg, pp 130–139
Strickler RE (1973) Change in selected characteristics of students between ninth and twelfth grade as related to high school curriculum
Sturtevant N (2012) Benchmarks for grid-based pathfinding. Trans Comput Intell AI Games 4(2):144–148. http://web.cs.du.edu/~sturtevant/papers/benchmarks.pdf
Sutcliffe G (2009) The TPTP problem library and associated infrastructure: the FOF and CNF Parts, v3.5.0. J Autom Reason 43(4):337–362
Sutcliffe G, Suttner C (2006) The state of CASC. AI Commun 19(1):35–48
Thrun S (1996) Is learning the n-th thing any easier than learning the first? In: Advances in neural information processing systems, pp 640–646
Thrun S, Pratt L (2012) Learning to learn. Springer, New York
Thurstone LL (1938a) Primary mental abilities. Psychometric monographs
Thurstone LL (1938b) Primary mental abilities. Psychometric monographs
Togelius J, Yannakakis GN, Karakovskiy S, Shaker N (2012) Assessing believability. In: Believable bots, Springer, pp 215–230
Torrey L, Shavlik J (2009) Transfer learning. Handb Res Mach Learn Appl 3:17–35
Turing AM (1950) Computing machinery and intelligence. Mind 59:433–460
Valiant LG (1984) A theory of the learnable. Commun ACM 27(11):1134–1142
Vallati M, Chrpa L, Grzes M, McCluskey TL, Roberts M, Sanner S (2015) The 2014 international planning competition: progress and trends. AI Mag 36(3):90–98
van Rijn JN, Bischl B, Torgo L, Gao B, Umaashankar V, Fischer S, Winter P, Wiswedel B, Berthold MR, Vanschoren J (2013) Openml: a collaborative science platform. In: Machine learning and knowledge discovery in databases. Springer, pp 645–649
Vanschoren J, Blockeel H, Pfahringer B, Holmes G (2012) Experiment databases. Mach Learn 87(2):127–158
Vanschoren J, van Rijn JN, Bischl B, Torgo L (2014) Openml: networked science in machine learning. ACM SIGKDD Explor Newsl 15(2):49–60
Vázquez D, López AM, Marín J, Ponsa D, Gerónimo D (2014) Virtual and real world adaptation for pedestrian detection. IEEE Trans Pattern Anal Mach Intell 36(4):797–809. doi:10.1109/TPAMI.2013.163
Vere SA (1992) A cognitive process shell. Behav Brain Sci 15(03):460–461
von Ahn L (2009) Human computation. In: Design automation conference, 2009. DAC'09. 46th ACM/IEEE, IEEE, pp 418–419
von Ahn L, Blum M, Langford J (2004) Telling humans and computers apart automatically. Commun 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
Wallace CS, Boulton DM (1968) An information measure for classification. Comput J 11(2):185–194
Wallace CS, Dowe DL (1999) Minimum message length and Kolmogorov complexity. Comput J 42(4):270–283 (special issue on Kolmogorov complexity)
Wang G, Mohanlal M, Wilson C, Wang X, Metzger M, Zheng H, Zhao BY (2012) Social Turing tests: crowdsourcing sybil detection. arXiv preprint arXiv:1205.3856
Wang P (2010) The evaluation of agi systems. In: Proceedings of the third conference on artificial general intelligence, Citeseer, pp 164–169
Warwick K (2014) Turing Test success marks milestone in computing history. University or Reading Press Release,
Wasserman EA, Zentall TR (2006) Comparative cognition: Experimental explorations of animal intelligence. Oxford University Press, Oxford
Watkins CJCH, Dayan P (1992) Q-learning. Mach Learn 8(3):279–292
Weiss DJ (2011) Better data from better measurements using computerized adaptive testing. J Methods Meas Soc Sci 2(1):1–27
Weizenbaum J (1966) ELIZA—a computer program for the study of natural language communication between man and machine. Commun ACM 9(1):36–45
Wellman M, Reeves D, Lochner K, Vorobeychik Y (2004) Price prediction in a trading agent competition. J Artif Intell Res JAIR 21:19–36
White DR, McDermott J, Castelli M, Manzoni L, Goldman BW, Kronberger G, Jaśkowski W, O'Reilly UM, Luke S (2013) Better GP benchmarks: community survey results and proposals. Genet Program Evolvable Mach 14:3–29. doi:10.1007/s10710-012-9177-2
Whiteson S, Tanner B, White A (2010) The reinforcement learning competitions. AI Mag 31(2):81–94
Whiteson S, Tanner B, Taylor ME, Stone P (2011) Protecting against evaluation overfitting in empirical reinforcement learning. In: 2011 IEEE symposium on adaptive dynamic programming and reinforcement learning (ADPRL), IEEE, pp 120–127
Williams PL, Beer RD (2010) Information dynamics of evolved agents. In: From animals to animats 11, Springer, pp 38–49
Winikoff M, Cranefield S (2014) On the testability of bdi agent systems. J Artif Intell Res JAIR 51:71–131
Wolpert DH (1996) The lack of a priori distinctions between learning algorithms. Neural Comput 8(7):1341–1390
Wolpert DH (2012) What the no free lunch theorems really mean; how to improve search algorithms. Technical report, Santa fe Institute Working Paper
Wolpert DH, Macready WG (1995) No free lunch theorems for search. Technical report SFI-TR-95-02-010 (Santa Fe Institute)
Wolpert DH, Macready WG (2005) Coevolutionary free lunches. IEEE Trans Evol Comput 9(6):721–735
Yampolskiy RV (2015) Artificial superintelligence: a futuristic approach. CRC Press, Boca Raton
Yonck R (2012) Toward a standard metric of machine intelligence. World Future Rev 4(2):61–70
You J (2015) Beyond the turing test. Science 347(6218):116–116
Zatuchna Z, Bagnall A (2009) Learning mazes with aliasing states: an LCS algorithm with associative perception. Adapt Behav 17(1):28–57
Zhou ZH (2012) Ensemble methods: foundations and algorithms. CRC Press, Boca Raton