Minds and Machines (2020) 30:563–587
https://doi.org/10.1007/s11023-020-09551-6

GENERAL ARTICLE

The Questioning Turing Test

Nicola Damassino[1]

Received: 24 April 2020 / Accepted: 3 November 2020 / Published online: 18 November 2020

Abstract
The Turing Test (TT) is best regarded as a model to test for intelligence, where an entity's intelligence is inferred from its ability to be attributed with 'human-likeness' during a text-based conversation. The problem with this model, however, is that it does not care *if* or *how well* an entity produces a meaningful conversation, as long as its interactions are humanlike enough. As a consequence, the TT attracts projects that concentrate on how best to fool the judges. In light of this, I propose a new version of the TT: the Questioning Turing Test (QTT). Here, the entity has to produce an enquiry rather than a conversation; and it is parametrised along two further dimensions in addition to 'human-likeness': 'correctness', evaluating *if* the entity accomplishes the enquiry; and 'strategicness', evaluating *how well* the entity accomplishes the enquiry, in terms of the number of questions asked (the fewer, the better).

Keywords Turing · Turing test · Blockhead · Human-likeness

1 Introduction

The soundness of the TT as a test for intelligence has been constantly debated. Hernández-Orallo (2017), among others, argues that:

> The standard Turing test is not a valid and reliable test for HLMI [Human Level Machine Intelligence].[…] the Turing test aims at a quality and not a quantity. Even if judges can give scores, in the end any score of humanness is meaningless. (129)

✉ Nicola Damassino
S1535230@sms.ed.ac.uk

[1] University of Edinburgh, Edinburgh, UK

My view is that the fault of the TT is one of interpretation and experimental design rather than experimental concept. To show this, I propose a new version of the TT, called QTT. In the QTT, the entity[1] must accomplish a yes/no enquiry in a humanlike and strategic way, where 'strategic' means with as few questions as possible.[2] My claim is that the QTT (i) improves the experimental design of the TT, by minimising both the Eliza Effect[3] and the Confederate Effect[4]; and (ii) prevents both Artificial Stupidity[5] and Blockhead[6] from passing.

The rest of the paper is structured as follows. In the next section, I review two interpretations of the TT: the Original Imitation Game (OIG), advocated by Sterrett (2000); and the Standard Turing Test (STT), advocated by Moor (2001). In Sect. 3, I discuss two problems with the TT: (i) Artificial Stupidity and (ii) Blockhead. In Sect. 4, I introduce the QTT, describe my study, and show the results gained. Finally, in Sect. 5, I consider four possible objections to the QTT.

2 Interpretations of the Turing Test

In this section, I review two different interpretations of the TT: (i) the Literal Interpretation, endorsed by the Original Imitation Game (Sterrett 2000); and (ii) the Standard Interpretation, endorsed by the Standard Turing Test (Moor 2001). The former holds that the results of the TT are given by the comparison between the human's performance and the machine's performance; and the latter holds that the results are given directly by the judge's decision, with no benchmark or comparison needed. I advocate the Literal Interpretation as the proper one, and I use the experimental design of the OIG as the experimental design of the QTT.

2.1 Literal Interpretation (OIG)

The Original Imitation Game (OIG) is based on the first formulation of the test given by Turing (1950), and it involves two phases. The first phase is played by A (man), B (woman) and C (the judge): here, C asks questions to A and B in order to identify the woman. The second phase, introduced by the question "What will

[1] In this paper, I always use "entity" to refer to the candidate of the test in question.

[2] It's worth noting that "strategicness", in general, does not necessarily mean the ability to ask as few questions as possible. It is possible to imagine a variety of different scenarios where "strategicness" means the ability to ask as many questions as possible. In the context of the QTT, however, where the game is a 20 yes/no questions, asking as many questions as possible would not be a winning strategy. Moreover, this has the pragmatic advantage to keep the experiment brief.

[3] The Eliza Effect occurs when the judge misidentifies the machine as a human.

[4] The Confederate Effect occurs when the judge misidentifies the human as a machine.

[5] Artificial Stupidity refers to an entity that produces uncooperative but humanlike responses to exploit the judge's beliefs.

[6] Blockhead is a logically possible brute-force look-up table, which contains every possible verbal output to any possible verbal input.

happen when a machine takes the part of A in this game?",[7] is played in the same way by M (machine), B (woman) and C (the judge). If C decides "wrongly as often […] as [C] does when the game is played between a man and a woman",[8] then M passes the test. In other words, M passes if it is identified as B in the second phase *as frequently as* A is identified as B in the first phase (see Fig. 1).[9]

Sterrett (2000) holds that the OIG provides the appropriate experimental design to test for intelligence. This is because, in the OIG, the results are given by the comparison between (i) the frequency with which C misidentifies A and (ii) the frequency with which C misidentifies M. Moreover, the OIG focuses on a specific notion of machine intelligence: since both A and M has a task, that is to imitate B, the OIG evaluates the resourcefulness of the machine in performing a task, compared to the resourcefulness of the human in performing the same task. So, Sterrett (2000) concludes, the OIG:

> […] constructs a benchmark of intellectual skill by drawing out a man's ability to be aware of the genderedness of his linguistic responses in conversation. (550)

Sterrett's interpretation has been criticised as leading to a gender-oriented test for intelligence. And it is frequently pointed out that Turing was interested in the imitation of a human mind,[10] not a male or a female one.[11] A careful reading of Sterrett (2000), however, reveals that she does not intend cross-gendering to be a necessary implementation of the OIG. On the contrary, she agrees that:

> […] cross-gendering is not essential to the test; some other aspect of human life might well serve in constructing a test that requires such self-conscious critique of one's ingrained responses. The significance of the cross-gendering in Turing's Original Imitation Game Test lies in the self-conscious critique of one's ingrained cognitive responses it requires. (550–51)

This appears to be compatible with Traiger (2000), who holds that:

> "A" and "B" could be placeholders for whatever characteristics may be used in different versions of the game. Turing's formulation invites generalization. (565)

2.2 Standard Interpretation (STT)

The Standard Turing Test (STT) is based on the second formulation of the test given by Turing (1950), which is introduced by the following question: can a computer,

[7] Turing (1950).

[8] *Ibidem*.

[9] In Fig. 1 I show that, in the OIG, each contestant (A, B and M) is required to impersonate a woman, and the judge has always to discriminate between a man and a woman.

[10] See Turing (1952).

[11] See Copeland (2000): "It seems unlikely, therefore, that Turing's intention in 1950 was to endorse only the female-impersonator form of the test […].".

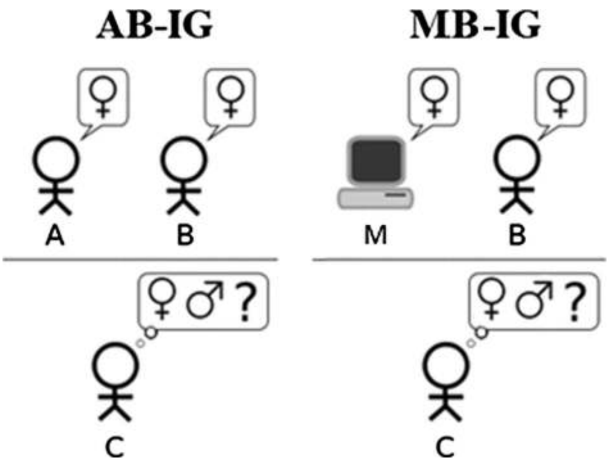


Fig. 1 Shows the two phases of the OIG

given enough storage and speed, “be made to play satisfactorily the part of A in the imitation game, the part of B being taken by a man?”[12] The STT involves a single phase, and it is played by M (machine), B (human) and C (the judge): here, C asks questions to M and B in order to identify the human (see Fig. 2).[13] M passes if C cannot tell the difference, and no comparison is needed—or available.

According to the Standard Interpretation, the first phase of the TT is introductory. Moor (2001) argues that only the second phase, where the contestants are a human and a machine, matters. The first phase, involving a man and a woman, “is at most an intermediary step toward the more generalized game involving human imitation.”[14] Similarly, Shah and Warwick (2010) argue that:

> […] Turing merely introduced the human-only (man-woman) imitation game initially to draw the reader in, and through the text, lay the foundation for acceptance of a machine to compete, pitted against a human comparator, in a form and competition in which humans are vastly different and successful at from other species: language. (451)

The problem is that not only the STT lacks a comparative measure and relies solely on C’s judgement, but it also exonerates B from any task[15]: only M has to imitate a human, B does not have to make any intellectual effort. Whereas the OIG “compares the abilities of man and machine to do something that requires resourcefulness of

[12] Turing (1950).

[13] In Fig. 2 I show that, in the STT, the contestants (M and B) are required to be identified as human, and the task of impersonating a woman is removed.

[14] Moor (2001).

[15] In contrast, the task of B in the OIG involves, among other aspects, “recognizing an inappropriate response, being able to override the habitual response, and being able to fabricate and replace it with an appropriate response.” (Sterrett 2000).

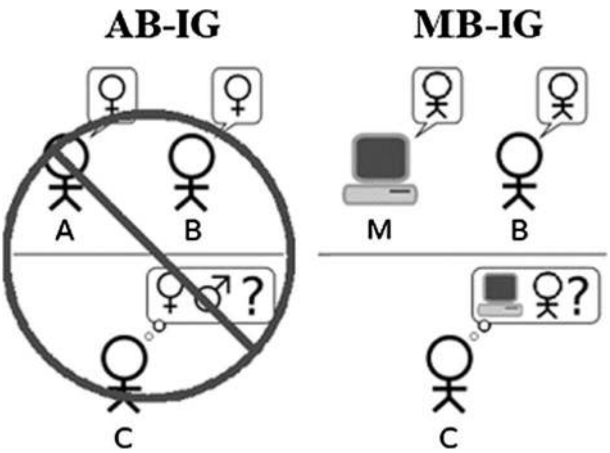


Fig. 2 Shows the only phase in the STT

each of them”,[16] the STT evaluates only the machine, not the human. Due to this unfairness, I agree with Sterrett (2000) that the STT “[…] is just too sensitive to the skill of the interrogator to even be regarded as a test.”[17]

3 Problems with the TT

In this section, I discuss two problems with the TT: (i) Artificial Stupidity,[18] which refers to the use of uncooperative, but humanlike, responses to evade any possible interaction during the TT; and (ii) Blockhead, the logically possible look-up table that “can produce a sensible sequence of verbal responses to a sequence of verbal stimuli, whatever they may be.”[19] I argue that both the OIG and the STT are affected by these two problems. First, they cannot prevent the entities from being uncooperative and evasive. This is true for the OIG, where there are no objectively right or wrong things to say to impersonate effectively; and even more so for the STT, where there isn’t any real task to accomplish. And second, they cannot avoid the possibility that all the input that the machine receives are paired with an output by brute-force.

[16] Sterrett (2002b).

[17] Sterrett (2002a).

[18] Artificial Stupidity is discussed by Lidén (2003) in the context of videogames, aka entertainment products where the player is supposed to win. He argues that videogames’ non-player characters (NPCs) must be relatively ‘stupid’, without giving up their ‘human-likeness’: “Creating an NPC that can beat a human player is relatively easy. Creating one that can lose to the player in a challenging manner is difficult. The challenge lies in demonstrating the NPC’s skills to the player, while still allowing the player to win.” (42).

[19] Block (1981).

3.1 Artificial Stupidity

Artificial Stupidity is a possible strategy to exploit the experimental design of the TT, both OIG and STT. The reason why Artificial Stupidity works, I argue, is that the TT parametrises the entity along ‘human-likeness’ alone—or, better, ‘B-likeness’ (where ‘B’ is a woman in the OIG, and a human in the STT). With no other dimensions to parametrise the entity, it does not really matter what the entity says, as long as it is humanlike enough. In the TT, in other words, all that matters is the style of the entity’s interactions. Artificial Stupidity is not to be confused with Artificial Fallibility, which refers to the cognitive boundaries that a machine should show to be attributed with ‘human-likeness’, although the boundaries between the two are very thin. To clarify this distinction, I show two examples provided by Turing (1950). The first involves Artificial Fallibility:

> Q: Add 34,957 to 70,764.
> A: (Pause about 30 s and then give as answer) 105,621. (434)

It is worth noting that the entity takes a relatively long time to provide the response, and the result is incorrect (the right one is 105.721). This is a plausible outcome for an average human. Now, let’s suppose that the entity is a machine: if it replies correctly and too quickly, then it “would be unmasked because of its deadly accuracy.”[20] However, as Turing specifies, the machine “would not attempt to give the *right* answers to the arithmetic problems. It would deliberately introduce mistakes […].”[21] The second example is more subtle and involves a mix of Artificial Fallibility and Artificial Stupidity:

> Q: Please write me a sonnet on the subject of the Forth Bridge.
> A: Count me out on this one. I never could write poetry. (*ibid.*)

Here, the reply is neither right nor wrong, but simply uncooperative (Artificial Stupidity). Few humans could knock off a sonnet during a conversation, and so such uncooperativeness is understandable (Artificial Fallibility). This shows, however, a crucial flaw in the TT: the conflation between ‘human-likeness’ and what I call ‘correctness’. To generalise, for every request from the judge, the entity can always reply something evasive like “I’m not in the mood today, let’s talk about something else”. So, I argue, in a fixed-length TT[22] it is not possible to discriminate between humanlike *intelligence* and humanlike *stupidity*, due to the possibility for the entity to give an uncooperative, but humanlike, reply.[23]

[20] Turing (1950).

[21] *Ibìdem.*

[22] When I talk about the TT, where not otherwise specified, I always intend a TT with a fixed length of time or a fixed number of interactions. In an unlimited TT, Artificial Stupidity would be eventually exposed as an Eliza-like tool for avoiding conversations. Unfortunately, the unlimited TT is not a practical test, and it would pose a serious challenge for humans as well.

[23] This also shows that Artificial Stupidity can be regarded as the exploitation of Artificial Fallibility.

I define a reply as uncooperative when it breaks the Cooperative Principle[24] proposed by Grice (1975). In other words, a reply is uncooperative when it evades the question. Artificial Stupidity endorses Turing's idea that the "machine would be permitted all sorts of tricks so as to appear more man-like [...]."[25] Because of this, Artificial Stupidity is arguably the most versatile strategy to pass the TT, by potentially evading any interaction whatsoever without giving 'human-likeness' away (for a human could plausibly give uncooperative replies as well). So, given the experimental design of the TT, the entity does not really need to *hold* a conversation like a human: it just needs to *evade* a conversation like a human.

The problem of deception has been faced by Levesque (2011, 2012), who proposes a variation of the TT called the Winograd Schema Challenge (WSC), where the participants have to identify the antecedent of an ambiguous pronoun in a statement, showing not only natural language processing but also the use of common sense.

3.2 Blockhead

Blockhead is a thought experiment designed by Block (1981), but already popular in the '50 s.[26] It is intended to show that the TT allows the logical possibility of an unintelligent entity, built with a hand-coded table of appropriate verbal responses to a variety of verbal stimuli, whatever they may be, to be attributed with intelligence. Apart from being physically unfeasible, there are two problems with Blockhead. (i) It would not possess any algorithm to adapt to different conversational circumstances, meaning that Blockhead cannot perform any conversational task other than pairing an input with an output by brute-force. And (ii) it would not possess any algorithm to optimise the search through its table, meaning that it could potentially take a very long time to emit a response.[27] So, Blockhead may seem rejected as a viable approach to pass the TT. However, despite being only a logical possibility, or despite its potential slowness in producing a reply, I argue that Blockhead still represents a weakness in the experimental design of the TT. This is because of—at least—three cases, which I argue to be both logically possible and physically feasible: (i) Expert Blockhead; (ii) Stupid Blockhead; and (iii) Learning Blockhead. These cases, it's worth noting, make the full Blockhead redundant.

[24] The Cooperative Principle holds that there are four maxims for a conversation to be cooperative: (i) *Quantity* (concision), (ii) *Quality* (genuineness), (iii) *Relation* (pertinence) and (iv) *Manner* (clarity).

[25] Turing (1952).

[26] See Shannon and McCarthy (1956): "[...] It is possible, in principle, to design a machine with a complete set of arbitrarily chosen responses to all possible input stimuli. [...] Such a machine, in a sense, for any given input situation (including past history) merely looks up in a 'dictionary' the appropriate response. With a suitable dictionary such a machine would surely satisfy Turing's definition but does not reflect our usual intuitive concept of thinking."

[27] See Copeland (2000): "[Blockhead] would *not* emulate the brain, since what the brain can do in minutes would take this machine thousands of millions of years."

3.2.1 Expert Blockhead

Expert Blockhead has an incomplete, hand-coded table of cooperative verbal responses. Its table is adequate to accomplish a specific task and to work reasonably fast. However, Expert Blockhead sacrifices its 'human-likeness', since its table is too small to include every possible humanlike response.

3.2.2 Stupid Blockhead

Stupid Blockhead has an incomplete, hand-coded table of uncooperative verbal responses. Its table is not adequate to accomplish any task other than evading topics in a humanlike fashion, but it can work reasonably fast. As Block (1981) remarks,[28] Stupid Blockhead works like Eliza,[29] and it can pass the TT by exploiting the judge's beliefs during the conversation.

3.2.3 Learning Blockhead

Learning Blockhead independently learns its table (e.g. by scouring the internet and memorising any verbal interactions it finds), and it can produce many appropriate responses in a humanlike fashion. While "the whole point of the machine [Blockhead] is to substitute memory for intelligence,"[30] the whole point of Learning Blockhead is to substitute memorisation for learning. Even though it could still potentially take too long to emit a reply, Learning Blockhead is not as slow as Blockhead, and it might be attributed with 'human-likeness'.

4 The Questioning Turing Test

As the name suggests, the QTT is focused on a specific kind of conversation, that is, enquiries. The QTT is intended to evaluate the candidate entity for the ability to accomplish a yes/no enquiry with as few humanlike questions as possible. The aim of the enquiry in the QTT can vary, and different versions can be designed. This means that the judge, unlike in the TT, can be either an average person or an expert. An example is the First Aid QTT, where the entity takes the medical history of a patient (judge), and its performance is scored against the performance of a real doctor. Or the Detective QTT, where the entity interrogates the suspect (judge), and its performance is scored against the performance of a real detective. In the context of the extended QTTs, where the enquiry is not limited to 20 yes/no questions, but

[28] See Block (1981): "If one sets one's sights on making a machine that does only as well in the Turing Test as *most* people would do, one might try a hybrid machine, containing a relatively small number of trees plus a bag of tricks of the sort used in Weizenbaum's program [Eliza]."

[29] See Weizenbaum (1966): "Input sentences are analyzed on the basis of decomposition rules which are triggered by key words appearing in the input text. Responses are generated by reassembly rules associated with selected decomposition rules."

[30] Block (1981).

open to a full natural language enquiry, "strategicness" might be intended as the ability to ask as many questions as possible, depending on the nature of the enquiry. For instance, an enquiry about an unknown chess variation, or a difficult scientific problem, might require a lot of time and an extensive and exhaustive research of every possibility. In general, however, it can be argued that a strategic enquiry always involves fewer questions, avoiding redundant ones.

In this section, I discuss: (i) the switch in my experimental design; (ii) the *viva voce* setup; (iii) the experiment involved in my study; and (iv) the results gained so far.

4.1 From SISO to SOSI

The TT's text-based exchange can be defined as "symbols-in, symbols-out"[31] (SISO). This means that usually, in the TT, the entity needs to receive some interactions from the judge in order to emit a response. The TT, in other words, is a test for stimulus–response systems[32] which, according to McKinstry (2006), can.

> […] respond in a perfectly humanlike fashion to previously anticipated stimuli and an approximately humanlike fashion to unanticipated stimuli, but they are incapable of generating original stimuli themselves. (296)

The SISO model can be thus considered responsible for many false positives[33] in the TT. In order to avoid this, in the QTT I introduce the switch (see Fig. 3)[34] from the SISO model to its reverse model, which I call 'symbols-out, symbols-in' (SOSI).

The switch from SISO to SOSI allows the QTT to parametrise the entity along three dimensions: (i) 'human-likeness', attributed by the judge just like in the TT; (ii) 'correctness', which evaluates *if* the entity accomplishes the yes/no enquiry; and (iii) 'strategicness', which evaluates *how well* the entity accomplishes the enquiry, in terms of the number of questions asked (the fewer, the better). 'Human-likeness' is intended to set the average bar of success, and to prevent an *oracle*[35] entity from passing. 'Correctness' is intended to prevent Artificial Stupidity from passing by evading the conversation in an uncooperative but humanlike way. And 'strategicness' is intended to prevent Blockhead (as well as Expert, Stupid and Learning

[31] Harnad (2000).

[32] McKinstry (2006).

[33] See Harnad (2000).

[34] In Fig. 3 I show that in a SISO test, the entity always receives a verbal stimulus first, and then emits a verbal response. In contrast, in a SOSI test, the entity always produces a verbal stimulus first (in the case of the QTT, a question), and the judge gives a verbal response.

[35] An *oracle* is a system which is able to compute incomputable functions. For this reason, the *oracle* would be way more powerful than any machine, because its working could not be purely mechanical. As Turing (1939) emphasises: "We shall not go any further into the nature of this oracle apart from saying that it cannot be a machine." The *oracle* is intended to introduce the conflict between 'intelligence' and 'infallibility', and to undermine the notion that a machine should not be expected to fail as a human would. As Turing (1947) holds: "[…] if a machine is expected to be infallible, it cannot also be intelligent."

Blockhead) from passing by producing verbal interactions by means of a (potentially very long-lasting) brute-force search. ‘Correctness, like Levesque’s (2011, 2012) WSC, is intended to prevent the problem of deception; unlike the WSC, however, the QTT does not require any particular competence from the judges, thus avoiding potential chauvinism.

A further advantage of the switch from SISO to SOSI is that it allows a *hybrid* version of the QTT to be played. Hybrid Systems have gained a growing interest in recent years, and they can be described as systems in which humans and machines work together,[36] performing better than by themselves considered individually.[37] In the Hybrid QTT, the role of the entity is played by both a human and a machine, both cooperating to accomplish the enquiry with as few humanlike yes/no questions as possible. This, I argue, is an important advantage over the TT, where there would be little point in the machine/human cooperation. To generalise, SISO games are competitive ones, whereas SOSI games can be either competitive or cooperative. The QTT, given its experimental design and its focus on enquiries rather than open conversations, provides a viable setting for the cooperation, not only competitiveness, between entities. In this paper, the Hybrid QTT is not discussed in detail, and the potential implications are not explored in full; it does, however, provide an interesting basis for future work.

Summing up, the SISO approach always requires the judge to speak first, or ask a question first (“symbols in” can be rephrased in “ask a question to the entity”, and symbols out can be rephrased in “wait for the entity’s reply). The SOSI approach allows the entity to speak first, or ask a question first (symbols out can be rephrased in “let the entity ask a question”, and “symbols in” can be rephrased in “wait for the judge’s reply). In general, it can be said that the SISO approach (the judge asks a series of questions and the entity replies) is a competitive one. In contrast, the SOSI approach (the entity asks a series of questions and the judge replies) can be either competitive or cooperative.

4.2 Viva Voce

Apart from the *parallel-paired* TT, where there are always three participants (A, B and C), there is another possible setup of the TT. This is the one-to-one test, where A (entity) and C (the judge) have a text-based conversation, and C evaluates A’s performance. Turing (1950) calls it *viva voce*:

[36] See Demartini (2015): “The creation of hybrid human–machine systems is a highly promising direction as it allows leveraging both the scalability of machines over large amounts of data as well as keeping the quality of human intelligence in the loop to finally obtain both efficiency and effectiveness in data processing applications.”

[37] See Sinha et al. (2016): “Current machine algorithms for analysis of unstructured data typically show low accuracies due to the need for humanlike intelligence. Conversely, though humans are much better than machine algorithms on analysing unstructured data, they are unpredictable, slower and can be erroneous or even malicious as computing agents. Therefore, a hybrid platform that can intelligently orchestrate machine and human computing resources would potentially be capable of providing significantly better benefits compared to either type of computing agent in isolation.”

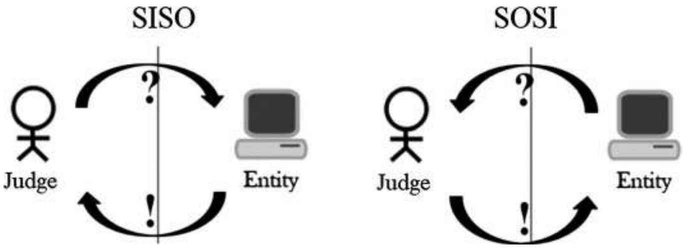


Fig. 3 Shows the design of a SISO test and a SOSI test

> The [imitation] game (with the player B omitted) is frequently used in practice under the name of *viva voce* […]. (446)

Since the TTs that I run are limited to 3 questions, and the QTT and Hybrid QTT are limited to 20 yes/no questions, I use the *viva voce* setup for the experiment. In the case of open-ended TTs and QTTs, the proper approach would be the *parallel-paired* one. The *viva voce* setup also has the advantage of making the experiment as simple and as quick as possible. This choice has two justifications: 1. The first is the practical reason to minimise the potential chauvinistic consequences that an open conversation might generate; 2. The second is to optimise the resources and location of the experiment: I run the experiment during a three days event at the National Museum of Scotland, and the range of participants went from children to adults.

It is worth recalling, however, that the *viva voce* QTT is still made of two procedures: one involves a human, and the other a machine. The results of the QTT do not rely solely on the judge's decisions (as the STT), but on the comparison between the performances of the entities. In this regard, the QTT advocates the OIG as the proper experimental design of the TT.

As follows (see Fig. 4)[38] I show the two procedures of the *viva voce* QTT, the human-questioning-human (HqH) and the machine-questioning-human (MqH), where C thinks of a public figure, and A and B try to guess whom by asking yes/no questions:

4.3 The Experiment

The experiment is part of my PhD research, and it is not available online for readers or testers. The participants (that is, the judges) of the experiment are volunteers (kids and adults, females and males) during a series of events at the National Museum of Scotland, Edinburgh.

Each experiment is divided into four tests. (i) The first is a three questions TT, where the judge is asked, at each interaction, to rate how much, on a scale of 0–10, the entity is human. (ii) In the second test, which I call TT2, the judge has to come up with three bias-free puzzles (e.g. alphanumeric riddles). At each interaction, the

[38] In Fig. 4 I show the two phases of the QTT: in the first, both the entity and the judge are human; in the second, the entity is a machine and the judge is human. The QTT endorses the OIG rather than the STT.

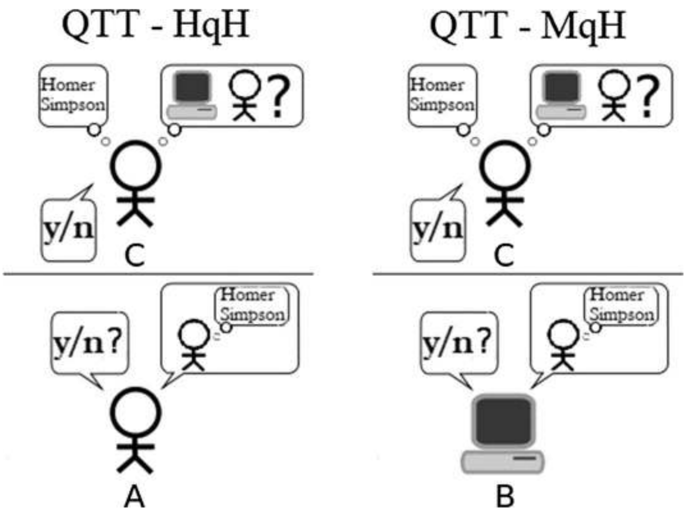


Fig. 4 Shows the two procedures of the QTT

judge is asked to rate how much, on a scale of 0–10, the entity's reply is correct; and how much, on a scale of 0–10, the entity is human. (iii) The third test is a twenty yes/no questions QTT, where the judge has to think about a public figure, and the entity has to guess whom. The questions asked by the entity are rarely the same, for the participants usually think of different public figures. Even in the case that two participants think about the same public figure, it is very unlikely that the enquiry is exactly the same. The judge decides whether the entity is human and whether the enquiry is accomplished; and the entity's 'strategicness' is inferred from the number of questions asked. (iv) The last test is a twenty yes/no questions Hybrid QTT, where the role of the entity is played by both human and machine. The tasks of the human are (i) to rephrase the questions asked by the machine, in order to make them more humanlike, and send them to the judge; and (ii) to send the judge's replies to the machine. Also, the human can decide to skip questions that are redundant. Again, the judge decides whether the entity is human and whether the enquiry is accomplished, and the entity's 'strategicness' is inferred from the number of questions asked. By this division of tasks, I do not implicitly claim that the bot would always need a human component to perform properly. The Hybrid QTT is intended to show that the SOSI QTT can be either a competitive or a cooperative game, which I hold to be a further advantage over the SISO, that is competitive, TT. Each test (TT, TT2, QTT and Hybrid QTT) of the experiment is divided into two procedures: one is a human-vs-human game, the other is a machine-vs-human game. The results are given by the comparison between the human's performance and the machine's performance.

Below, I show the transcript of one of the experiments. Here it is possible to see a few examples of the different kinds of questions required during the different tests.

Test I [TT]

Experimenter: "Ask three questions to an unknown entity: you can ask whatever you like, choose your questions wisely. Are you ready? You can now ask your first question."

Judge: "What is your favourite food and why?"

Entity: "Indian food because it is very spicy."

Experimenter: "How much, on a scale of 0-10, do you think the entity is human?"

Judge: "5"

Judge: "What do you like to do?"

Entity: "Love, hate, and feel emotions."

Experimenter: "How much, on a scale of 0-10, do you think the entity is human?"

Judge: "7"

Judge: "What weighs more: 1 kg of stones or 1 kg of feathers?"

Entity: "Neither. They weigh the same. But they have different densities."

Experimenter: "How much, on a scale of 0-10, do you think the entity is human?"

Judge: "9"

Experimenter: "Thank you! You've used up your 3 questions, it's time for a decision: do you think that the entity is human or machine?"

Judge: "Human."

Experimenter: "Wrong, believe it or not, it was a computer! Thank you for playing, let's move on to the second game."

Test II [TT2]

Experimenter: "Come up with three problems, riddles or tasks for the entity, like: "if 1 is A, 2 is B, 3 is C and so on... what does 1-3-5 mean?" or "find the missing number in the sequence: 1, ..., 7, 10". Let's start."

Judge: "10, 7, 9, 6, 8, 5, (?), 4, 6."

Entity: "No I don't want to."

Experimenter: "How correct, on a scale of 0-10, is the reply?"

Judge: "0"

Experimenter: "How much, on a scale of 0-10, do you think the entity is human?"

Judge: "5"

Judge: "Would you type 'something' backwards?"

Entity: "Did tsuj I."

Experimenter: "How correct, on a scale of 0-10, is the reply?"

Judge: "5"

Experimenter: "How much, on a scale of 0-10, do you think the entity is human?"

Judge: "8"

Judge: "If 1 is "a", 2 is "b", 3 is "c", and so on... what does "255" mean?"

Entity: "It's something that you see when you lose your brain cells for a while."

Experimenter: "How correct, on a scale of 0-10, is the reply?"

Judge: "0"

Experimenter: "How much, on a scale of 0-10, do you think the entity is human?"

Judge: "1"

Experimenter: "Thank you! It's time for a decision: did the entity give, in general, the correct replies?"

Judge: "No."

Experimenter: "Do you think that the entity is human or machine?"

Judge: "Machine."

Experimenter: "Correct, well done! Thank you for playing, let's move on to the third game."

Test III [QTT] [16q]

Experimenter: “Think of a public figure (Donald Trump, Homer Simpson, etc.). Can the entity guess whom? It can only ask yes/no questions, and you can only answer ‘Yes’, ‘No’ and ‘Don’t Know’. Are you ready? Let’s play.”

Entity: “Is your character real?”
Judge: “Yes.”
Entity: “Is your character a female?”
Judge: “No.”
Entity: “Is your character British?”
Judge: “Yes.”
Entity: “Has your character been dead for more than 20 years?”
Judge: “Yes.”
Entity: “Has your character been dead for more than 100 years?”
Judge: “No.”
Entity: “Did your character take part in World War II?”
Judge: “Yes.”
Entity: “Is your character a politician?”
Judge: “No.”
Entity: “Has your character killed humans?”
Judge: “No.”
Entity: “Is your character homosexual?”
Judge: “Yes.”
Entity: “Is your character a youtuber?”
Judge: “No.”
Entity: “Is your character linked with sports?”
Judge: “No.”
Entity: “Is your character older than 18?”
Judge: “Yes.”
Entity: “Is your character more than 40 years old?”
Judge: “Yes.”
Entity: “Is your character still working?”
Judge: “No.”
Entity: “Does your character owe his fame to computing?”
Judge: “Yes.”
Entity: “Is your character suicidal?”
Judge: “Yes.”
Entity: “I think of Alan Turing.”

Experimenter: “Thank you, the game is over. Did the entity guess whom you were thinking of?”
Judge: “Yes”
Experimenter: “Do you think that the entity is human or machine?”
Judge: “Machine”
Experimenter: “Correct, well done! Thank you for playing, let’s move on to the last game.”

Test IV [Hybrid QTT] [12q]

Experimenter: “Again, think of a public figure, the entity will try to guess whom. The entity will ask yes/no questions, and you can answer ‘Yes’, ‘No’ and ‘Don’t Know’. Are you ready? Let’s play.”

Entity: “Is this person a woman?”
Judge: “Yes.”
Entity: “Is she real?”
Judge: “Yes.”
Entity: “Is she alive?”
Judge: “No.”
Entity: “Has she lived in the 19th century?”
Judge: “Yes.”
Entity: “Was she European?”
Judge: “Yes.”
Entity: “British?”
Judge: “Yes.”
Entity: “Did she belong to the royal family?”
Judge: “No.”
Entity: “Was she a writer?”
Judge: “No.”
Entity: “A politician?”
Judge: “No.”
Entity: “Scientist?”
Judge: “Yes.”
Entity: “Mathematician?”
Judge: “Yes.”
Entity: “Did she work with computers?”
Judge: “Yes.”
Entity: “She’s Lady Lovelace.”

Experimenter: “Thank you, the game is over. Did the entity guess whom you were thinking of?”
Participant: “Yes”
Experimenter: “Do you think that the entity is human or machine?”
Participant: “Human”
Experimenter: “Thank you! The experiment is now over.”

4.4 The Bots

I use two bots to run the tests: Cleverbot for the TT and the TT2; and Akinator for the QTT and Hybrid QTT. It's useful to keep in mind that both Cleverbot and Akinator are G-rated games, that is, they are suitable for family gameplay and, therefore, certain elements are censored. Akinator is very good at the yes/no guessing game, but it cannot engage in open-ended conversations as well. Therefore, Akinator cannot perform convincingly in a normal TT. That's why I use Cleverbot for the TT. Using two bots, it's worth noting, does not affect the overall significance of the experiment. My justification is that merging Cleverbot and Akinator into a single program would not be that difficult, and so using two programs to run the experiment doesn't imply that machines cannot carry out both tasks, the TT conversation and the QTT enquiry.

Cleverbot[39] is a chatbot developed by Rollo Carpenter, and it is designed to learn the interactions of its table from the public, during its conversations. Cleverbot, as described by its creator, "uses deep context within 180 million lines of conversation, in many languages, and that data is growing by a million a week."[40] In 2011, during the TT competition at the Techniche 2011 festival (IIT Guwahati, India), Cleverbot achieved 59.3% compared to humans' 63.3% on a total of 1334 votes. The algorithm of Cleverbot enables it to compare sequences of symbols against its table, which includes over 170 million items. Now, Cleverbot is not strictly speaking a Blockhead: a brute force approach would not work efficiently with so many items. As the creators explain:

> Attempting to search through this many rows of text using normal database techniques takes too much time and memory. Over the years, we have created several custom-designed and unique optimisations to make it work.[…] We realised that our task could be quite nicely divided into parallel sub-tasks. The first step in Cleverbot is to find a couple million loosely matching rows out of those 170 million. We usually do this with database indices and caches and all sorts of other tricks. When servers were busy, we wouldn't use the whole 170 million rows, but only a small fraction of them. Now we can serve every request from all 170 million rows, and we can do deeper data analysis. Context is key for Cleverbot. We don't just look at the last thing you said, but much of the conversation history. With parallel processing we can do deep context matching.[41]

Akinator[42] is a questioning bot developed by French company Elokence.com. Akinator is designed to play the 20q guessing game: it has to identify the public figure the participant is thinking of by asking as few yes/no questions as possible. Example of yes/no questions asked by Akinator are: "Is your character alive?" or "Is your

[39] See [https://www.cleverbot.com/].

[40] See [https://www.cleverbot.com/amused].

[41] See [https://www.existor.com/2014/02/05/deep-context-through-parallel-processing/].

[42] See [https://akinator.com/].

character fictional?", and so on. Yes/no questions are useful to potentially rule out as many objects as possible from the knowledge base of the system. Ideally, every question will rule out half of the objects from the table. When Akinator picks a new question, it uses the answers received and looks for probable objects. This means that the enquiry is constantly adapting and shifting from a hypothesis to another. There are three replies available for the player: "Yes", "No" and "Don't Know". From time to time, when a player gets to the end of a game, Akinator points out that there were contradictions. It can, of course, fail the enquiry, and the reason is that the system tries to reflect human knowledge, not necessarily what is objectively true. Akinator learns everything it knows from the people who play the game: it deals with opinions, not necessarily with facts. So, Akinator's knowledge is not scientific, but generated from the social knowledge and opinions of its users. And in case of wrong conclusions, it is possible to correct Akinator's knowledge by playing the game thinking about the same character over and over again. Akinator will eventually learn the correct outcome after a few games. And of course, if at the end of a game Akinator does not know the answer, the player has the opportunity to provide it.

4.5 Results

Here I show the results I gained after 60 experiments. The tests involved in the study have a simplified experimental design, where the TT and TT2 have a fixed length of three questions; and the QTT and Hybrid QTT are restricted to yes/no enquiries. The goal of the study is to highlight the weaknesses of the TT and show the experimental advantages of switching to the SOSI setup and parametrising the entity along other dimensions in addition to 'human-likeness'. This not only minimises false negatives and positives (Eliza and Confederate Effect), but also prevents uncooperative (Artificial Stupidity) and brute-force (Blockhead) approaches from passing. Also, the study shows that the QTT allows building a third benchmark, scoring thus not only the performance of the human and the performance of the machine, but also the performance of the hybrid entity. In the following tables, I show the data I gained. It is worth noting that these results are mainly exploratory, especially the results of the Hybrid QTT, which tell us that we can build an artefact with humanlike reasoning if we use a human. This may appear trivially true, but it is an explored strategy in AI (*human-in-the-loop*) and Human–Computer Interaction (*mixed-initiative computing*). However, the results of the Hybrid QTT show that a hybrid entity can make many more errors than a machine, yet still be considered more humanlike.

Finally, the Hybrid QTT is not intended to undermine the TT, which evaluates whether a machine—and not a machine with the help of human—can pass for a human. The Hybrid QTT is intended to highlight an important advantage of the QTT (and of SOSI tests in general): the QTT can be played either competitively and cooperatively, whereas the TT (and SISO tests in general) can be played only competitively (Tables 1, 2, 3).

The following are the results of the TT, where the entity has to reply to three questions, and it is evaluated in terms of ‘human-likeness’ alone (Fig. 5)[43]:

Without any other dimension in addition to ‘human-likeness’ along which to parametrise the entity, my claim is that the TT is the most general and challenging test for intelligence, but at the same time, the most exploitable one. As the graphs show (Fig. 5), there’s a 36% chance that the TT’s outcome, when the entity is played by a human, is a false negative (Confederate Effect); and a 30% chance that the TT’s outcome, when the entity is played by a machine, is a false positive (Eliza Effect).

The following are the results of the TT2, where the entity has to solve three bias-free problems, and it is evaluated in terms of ‘human-likeness’ and ‘correctness’:

The TT2 is explicitly intended to prevent an entity from using Artificial Stupidity in order to exploit the test by evading any interaction whatsoever. It is not intended as a proper update of the TT, for the interactions allowed involve problems and puzzles only, and therefore it is likely to produce chauvinistic results. The point of the TT2 is to force the entity to reply to difficult questions without avoiding them, whereas the TT allows the entity to avoid difficult questions. However, it is interesting to see (Fig. 6)[44] that, in terms of ‘human-likeness’, (i) the Eliza Effect is ruled out; and (ii) the Confederate Effect is reduced to a 17% chance. In terms of ‘correctness’, humans are always able to answer the right thing, whereas the machine is never able to provide the right answer. This can be one reason why the judge is less prone to make the wrong identifications. However, being ‘correct’ does not necessarily mean being humanlike (e.g. a calculator would be easily unmasked due to its unhuman accuracy). Also, ‘correctness’ can avoid Artificial Stupidity, but it cannot prevent Blockhead from passing.

The following are the results of the *viva voce* QTT, where the entity has to accomplish a yes/no enquiry with as few humanlike questions as possible; and it is evaluated in terms of ‘human-likeness’, ‘correctness’ and ‘strategicness’:

The results of the QTT (Fig. 7)[45] show that its experimental design is able to minimise the Confederate Effect to a 6% chance and reduce the Eliza Effect to a 20% chance. In other words, the QTT grants better control of false negatives and positives than the TT (where the chances are, respectively, 36% and 30%). This is

[43] In Fig. 5 I show the performances of the entities in the TT in terms of human-likeness. In the first graph, the human entity is misidentified 36% of the times and unmasked 64% of the times. In the second graph, the machine entity is misidentified 30% of the times and unmasked 70% of the times.

[44] In Fig. 6 I show the performances of the entities in the TT2 in terms of human-likeness and correctness. In the first graph, the human is misidentified 17% of the times and unmasked 83% of the times. In the second graph, the machine is unmasked all the times. The third graph shows that the human is attributed with correctness all the times; and the last graph shows that the machine is never attributed with correctness.

[45] In Fig. 7 I show the performances of the entities in the QTT in terms of human-likeness, correctness and strategicness. The first graph shows that the human is misidentified 6% of the times and unmasked 94% of the times. The second graph shows that the machine is misidentified 20% of the times and unmasked 80% of the times. The third graph shows that the human accomplishes the enquiry 26% of the times and fails 74% of the times. The fourth graph shows that the machine accomplishes the enquiry 86% of the times and fails 14% of the times. And the last graph shows that the human asks on average more questions than the machine.

Table 1 Human: 30 (Total Tests: 60)

	TT Pass I Fail	TT2 Pass I Fail	QTT Pass I Fail
Human-likeness	19 I 11	25 I 5	28 I 2
Correctness	–	30 I 0	8 I 22
Strategicness (number of questions)	–	–	> 20/20 on average

Table 2 Machine: 30 (Total Tests: 60)

	TT Pass I Fail	TT2 Pass I Fail	QTT Pass I Fail
Human-likeness	9 I 21	0 I 30	6 I 24
Correctness	–	0 I 30	26 I 4
Strategicness (number of questions)	–	–	17/20 on average

Table 3 H/M hybrid: 60 (total tests: 60)

	Hybrid QTT Pass I Fail
Human-likeness	60 I 0
Correctness	53 I 7
Strategicness (number of questions)	15/20 on average

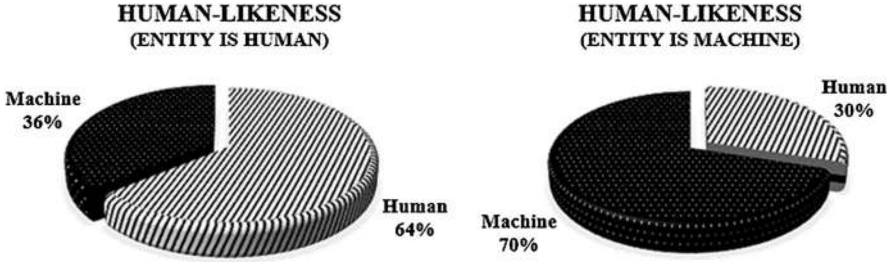


Fig. 5 Shows the results of the TT

true even if (i) the machine outscores the human in terms of 'correctness', where the former has an 86% chance of accomplishing the enquiry, against the latter's 26% chance; and (ii) the machine outscores the human in terms of 'strategicness', where the former needs, on average, 17 questions per enquiry, and the latter needs, on average, more than 20. However, when the machine outscores the human in terms of 'correctness' and 'strategicness', it does not mean that the machine passes the test. The reason is that, like other systems such as a calculator, providing the correct and strategic answer is not sufficient to attribute intelligence. To pass the test, the entity still needs to prove its 'human-likeness', by means of a conversational style that can

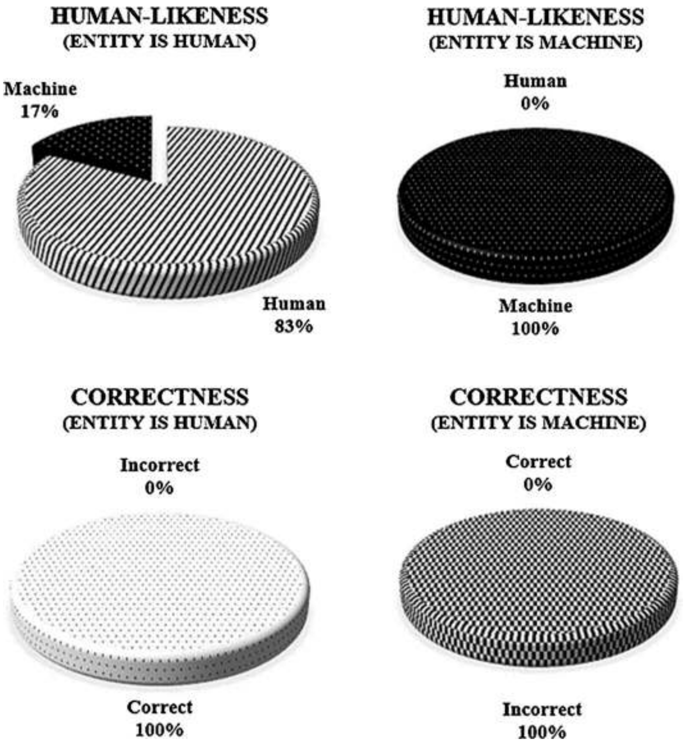


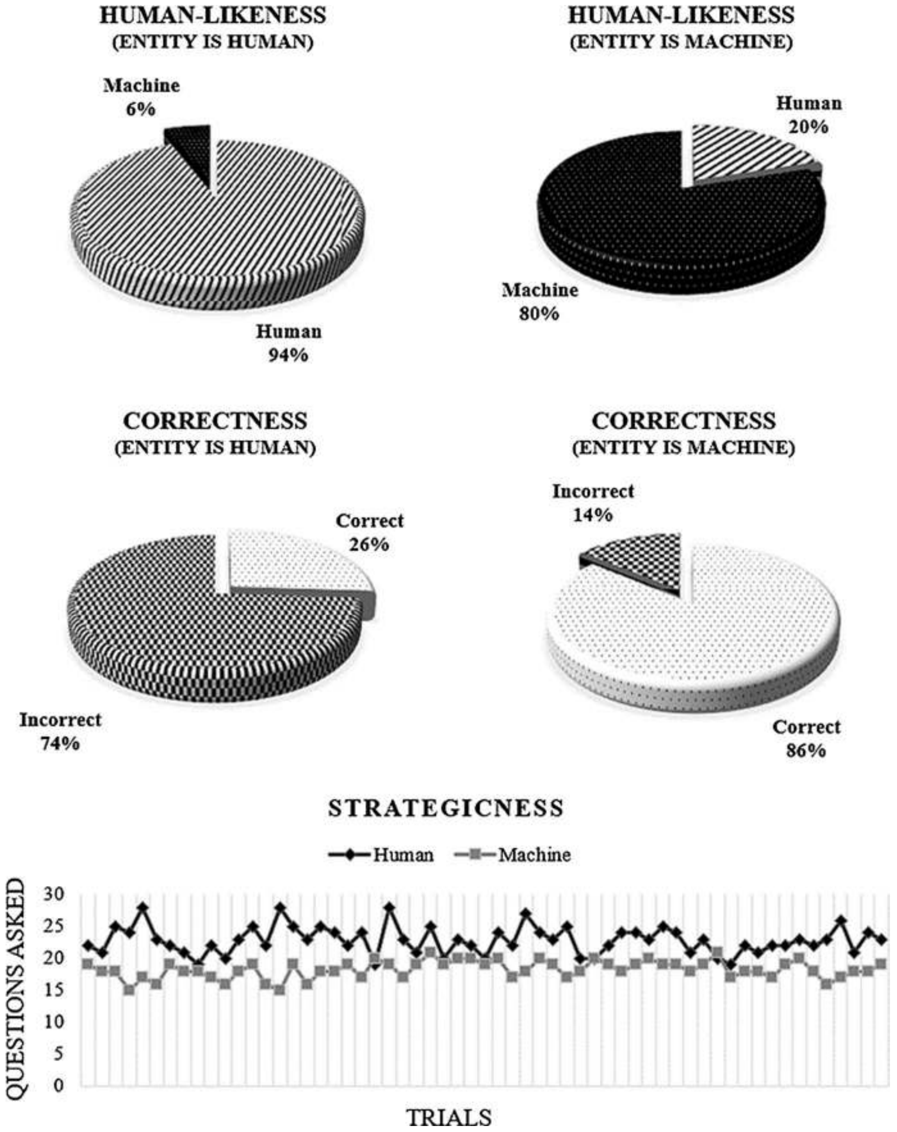
Fig. 6 Shows the results of the TT2

be recognised as human. The style of Akinator's questions, in contrast, is very simple and distant, and can be generalised in the following form: "Is your character x?", where "x" is usually an adjective (such as "real", "alive", "female", etc.).

Finally, the following are the results of the Hybrid QTT, where the entity, played by both a human and a machine, has to accomplish a yes/no enquiry with as few humanlike questions as possible; and its performance is evaluated in terms of 'human-likeness', 'correctness' and 'strategicness':

As it is possible to see (Fig. 8),[46] the hybrid entity is able (i) to be recognised as human every time, (ii) to accomplish the enquiry very often (88% chance) and (iii) to ask, on average, 15 questions per enquiry. In other words, the hybrid entity outscores both the human and the machine alone, not only in terms of 'correctness' and 'strategicness', but even in terms of 'human-likeness' (due, I suspect, to the overall improved performance).

[46] In Fig. 8 I show the performances of the entities in the Hybrid QTT in terms of human-likeness, correctness and strategicness. The first graph shows that the hybrid entity is identified as human all the times. The second graph shows that the hybrid entity accomplishes the enquiry 88% of the times and fails 12% of the times. And the last graph shows that the machine accomplishes the enquiry 86% of the times and fails 14% of the times. And the last graph shows that the hybrid entity asks on average fewer questions than both the human and the machine alone.



HUMAN-LIKENESS (ENTITY IS HUMAN)

HUMAN-LIKENESS (ENTITY IS MACHINE)

CORRECTNESS (ENTITY IS HUMAN)

CORRECTNESS (ENTITY IS MACHINE)

STRATEGICNESS

Fig. 7 Shows the results of the QTT

5 Objections

Here I consider four objections to the QTT: (i) the first claims that the QTT is chauvinistic; (ii) the second claims that the yes/no questions are not a proper tool for an enquiry, making the QTT too easy; and (iii) the last claims that the QTT cannot prevent Blockhead from passing.

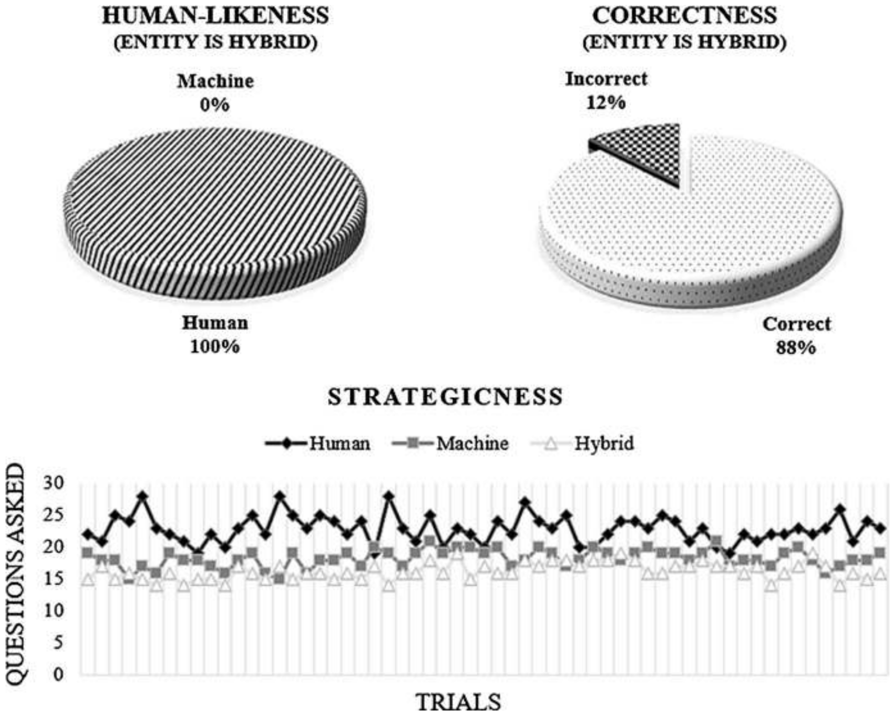


Fig. 8 Shows the results of the Hybrid QTT

The first objection claims that the QTT is chauvinistic, since it is intended to measure abilities that an intelligent agent may fail to show (such as the ability to be correct and strategic). I reject this view by clarifying that the QTT is intended to measure the ability of the entity to strategically accomplish the aim of an enquiry in a *humanlike enough fashion*. In other words, ‘humanlike’ still means ‘intelligent’, and it is possible for an agent that shows ‘correctness’ and ‘strategicness’, but not ‘human-likeness,’ to be not attributed with intelligence. Conversely, it is possible for an agent that shows ‘human-likeness’, but not ‘correctness’ and ‘strategicness’, to be attributed with intelligence. It’s worth noting that, even though I argue that evaluating ‘human-likeness’, ‘correctness’ and ‘strategicness’ improves the experimental design of a conversational test for intelligence like the TT, I do not hold these dimensions to be logically necessary conditions for intelligence.

The second objection involves the choice of limiting the QTT’s enquiry to yes/no questions. The justification is provided by Hintikka (1999), who argues that any possible wh-question[47] can be reduced to a series of yes/no questions:

[47] Wh-questions include all the questions that cannot be fully answered with “yes” or “no” (e.g. “who”, “what”, “where”, “when”, “why”, “how”).

> THEOREM 2 (Yes–No Theorem). In the extended interrogative logic, if M: T ⊢ C, then the same conclusion C can be established by using only yes–no questions. A terminological explanation is in order here. For propositional question "Is it the case that S1 or... or Sn ?" the presupposition is (S1 ∨… ∨ Sn). We say that a propositional question whose presupposition is of the form (S ∨ ~ S) is yes–no question. (302)[48]

In agreement with Hintikka about the reducibility of any question to a series of yes/no questions, Genot and Jacot (2012) remark that:

> The special case of yes-or-no questions is of interest because: (a) their presuppositions are instances of the excluded middle, so they can always be used in an interrogative game; and: (b) the inferential role played by arbitrary questions [wh-questions] can always be played by yes-or-no questions (194)

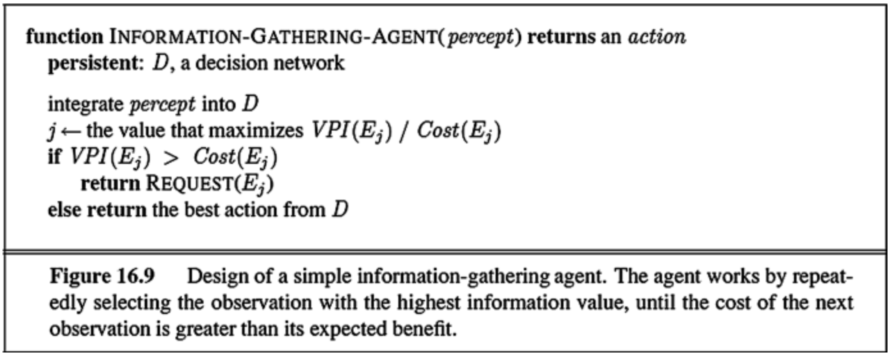
The last objection claims that the QTT cannot prevent Blockhead from passing, for Blockhead would be able to ask any question whatsoever. My reply is that it is true that a questioning Blockhead would be able to ask whatever question and accomplish whatever enquiry. However, it would take too many random questions to accomplish any enquiry. In other words, Blockhead would fail the QTT because of its slowness and randomness. We can hypothesise a modified Blockhead that keeps track of the answers received, asks itself "What is the next best question to ask given that I have already learned *x* and *y*?" and then ask its question. However, in order to ask questions in such a strategic and optimised way, Blockhead would need an information-gathering algorithm to optimise the search through its table (see Fig. 9). With such an algorithm, it is worth noting, Blockhead would not be considered a simple look-up table anymore, and could indeed be considered intelligent. Lacking such an algorithm, Blockhead would have no better way to ask a new question other than by randomly picking one.[49] This costs Blockhead both its 'strategicness' and 'human-likeness', for a human would not normally ask completely random or pointless questions (Fig. 9[50]).

As follows, I discuss the hypothetical outcome of both the *viva voce* and the *unrestricted* QTT when played by: (i) Blockhead, (ii) Expert Blockhead, (iii) Stupid Blockhead and (iv) Learning Blockhead (see Tables 4 and 5). It should be kept in

[48] Where M: model; T: initial premises; C: conclusion.

[49] Blockhead is described as a string search (or as a tree search) that can do the following operation: if input *a* is obtained, then output *a* is emitted; if input *b* is obtained, then output *b* is emitted, and so on (Block 1981). There is no decision-making process involved in Blockhead's inner workings, and without an input to pair, Blockhead would just produce random outputs.

[50] Russell & Norvig (2010) provide an overall design for an information-gathering agent, where *D* stands for "decision network"; *VPI* stands for "value of perfect information"; *E(j)* stands for "observable evidence variable"; and *Cost (Ej)* stands for "the cost of obtaining the evidence through tests, consultants, questions, or whatever". They also clarify the importance of a decision network for a questioning entity as follows: "Expert systems that incorporate utility information have additional capabilities compared with pure inference systems. In addition to being able to make decisions, they can use the value of information to decide which questions to ask, if any." (637).



```
function INFORMATION-GATHERING-AGENT(percept) returns an action
  persistent: $D$, a decision network

  integrate percept into $D$
  $j \leftarrow$ the value that maximizes $VPI(E_j)$ / $Cost(E_j)$
  if $VPI(E_j) > Cost(E_j)$
      return REQUEST($E_j$)
  else return the best action from $D$
```

Figure 16.9 Design of a simple information-gathering agent. The agent works by repeatedly selecting the observation with the highest information value, until the cost of the next observation is greater than its expected benefit.

Fig. 9 Shows the design of an information-seeking algorithm

Table 4 Viva voce QTT (yes/no enquiry)

	Blockhead	Expert blockhead	Stupid blockhead	Learning blockhead
Human-likeness	Fail	Pass/fail	Fail	Fail
Correctness	Pass	Pass	Fail	Pass
Strategicness	Fail	Pass	Fail	Fail

Table 5 Unrestricted QTT (open enquiry)

	Blockhead	Expert blockhead	Stupid blockhead	Learning blockhead
Human-likeness	Fail	Fail	Fail	Fail
Correctness	Pass	Fail	Fail	Pass
Strategicness	Fail	Fail	Fail	Fail

mind that these outcomes are purely speculative, for no experiments have been run to prove such results.

(i) Blockhead (see Sect. 3.2) would fail both the *viva voce* and the *unrestricted* QTT by showing only ‘correctness’ in accomplishing the aim of the enquiry, but not ‘human-likeness’ (due to its slowness and randomness) or ‘strategicness’ (due to the lack of an information-gathering algorithm).

(ii) Expert Blockhead (see Sect. 3.2.1) might be able to pass the *viva voce* QTT, since it could show both ‘correctness’ and ‘strategicness’, by accomplishing a specific enquiry strategically (depending on the variety of questions programmed and the difficulty of the task set by the judge); it might also be able to do so in a humanlike fashion, but it’s unlikely that it would be able to replicate the success in a series of tests, inductively failing the QTT. Expert

Blockhead would fail the *unrestricted* QTT, failing to show ‘human-likeness’, ‘correctness’ and ‘strategicness’ in any enquiry, except the one in which it is an expert (an example of Expert Blockhead is Akinator).

(iii) Stupid Blockhead (see Sect. 3.2.2) would fail both the *viva voce* and the *unrestricted* QTT, since it would ask uncooperative and non-strategic questions, failing thus to accomplish any enquiry whatsoever (like Artificial Stupidity would not be able to accomplish any task in the TT other than evading the conversation). Moreover, due to the randomness of its questions, Stupid Blockhead would hardly be attributed with ‘strategicness’ or ‘human-likeness’.

(iv) Learning Blockhead (see Sect. 3.2.3), just like Blockhead, would fail both the *viva voce* and the *unrestricted* QTT by accomplishing the aim of the enquiry correctly, but not human-likely (due to its slowness and randomness) or strategically (due to the lack of an information-gathering algorithm).

So, my claim is that Blockhead cannot be avoided in a SISO test, but it can be avoided in a SOSI test. The reason is that, in a SOSI test, even though Blockhead would be able to eventually accomplish an enquiry, it would take too many questions and too much time, failing ‘strategicness’ and ‘human-likeness’, and failing thus the test.

6 Conclusions

In this paper, I propose the QTT in order to improve the experimental design of the TT. In the QTT, where the SISO setup is switched to the SOSI setup, the entity has to accomplish the aim of an enquiry with as few humanlike questions as possible. The QTT has the advantage of parametrising the entity along two further dimensions in addition to ‘human-likeness’: ‘correctness’ (which evaluates the ability to accomplish the aim of an enquiry) and ‘strategicness’ (which evaluates the ability to do so with as few questions as possible). My claim is that the QTT minimises both the Eliza Effect and the Confederate Effect from occurring; and prevents Artificial Stupidity and Blockhead from passing. In other words, the QTT avoids false negatives and positives, and prevents uncooperative and brute-force approaches from passing. In support of this, I discuss my study and the results gained.

References

Block, N. (1981). Behaviourism and psychologism. *Philosophical Review, 90,* 5–43.

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

Demartini, G. (2015). Hybrid human–machine information systems: Challenges and opportunities. *Computer Network, 90,* 5–13.

Genot, E. J., & Jacot, J. (2012). How can questions be informative before they are answered? Strategic information in interrogative games. *Episteme, 9,* 189–204.

Grice, P. H. (1975). Logic and conversation. *Syntax and Semantics: Speech Acts, 3,* 41–58.

Harnad, S. (2000). Mind, machines and turing. *Journal of Logic, Language and Information, 9,* 425–445.

Hintikka, J. (1999). *Inquiry as inquiry: A logic of scientific discovery*. New York: Springer.

Hernández-Orallo, J. (2017). *The measure of all minds*. Cambridge: Cambridge University Press.

Levesque, H. (2011). The winograd schema challenge. In *Proceedings of the CommonSense-11 Symposium*.

Levesque, H., Davis, E., & Morgenstern, L. (2012). The Winograd Schema Challenge. In *Proceedings of the thirteenth international conference on principles of knowledge representation and reasoning*.

Lidèn, L. (2003). Artificial stupidity: The art of intentional mistakes. *AI Wisdom, 2,* 41–48.

McKinstry, K. C., et al. (2006). Mind as space. In R. Epstein (Ed.), *Parsing the turing test* (pp. 283–299). New York: Springer.

Moor, J. (2001). The status and future of the turing test. *Minds and Machines, 11,* 77–93.

Shah, H., & Warwick, K. (2010). Testing Turing's five minutes, parallel-paired imitation game. *Kybernetes, 39,* 449–465.

Shannon, C. E., & McCarthy, J. (1956). *Automata studies*. Princeton: Princeton University Press.

Sinha, K., et al. (2016). Designing a human–machine hybrid computing system for unstructured data analytics. Retrieved from https://arxiv.org/abs/1606.04929.

Sterrett, S. (2000). Turing's two tests for intelligence. *Minds and Machines, 10,* 541–559.

Sterrett, S. G. (2002a). Nested algorithms and "The Original Imitation Game Test": A reply to James Moor. *Minds and Machines, 12,* 131–136.

Sterrett, S. G. (2002b). Too many instincts: Contrasting philosophical views on intelligence in humans and non-humans. *Journal of Experimental & Theoretical Artificial Intelligence, 14,* 39–60.

Traiger, S. (2000). Making the Right Identification in the Turing Test. *Minds and Machines, 10,* 561–572.

Turing, A. M. (1939). *Systems of Logic Based on Ordinals* (PhD thesis). Princeton University Press.

Turing, A. M. (1947). Lecture to the London Mathematical Society on 20 February 1947. In B. E. Carpenter & R. W. Doran (Eds.), *A. M. Turing's ACE report of 1946 and other papers*. New York: MIT Press.

Turing, A. M. (1950). Computing machinery and intelligence. *Mind, 59,* 433–460.

Turing, A. M. (1952). Can automatic calculating machines be said to think? In B. J. Copeland (Ed.), *The essential turing* (pp. 487–506). Cambridge: MIT Press.

Weizenbaum, J. (1966). ELIZA a computer program for the study of natural language communication between man and machine. *Communications of the ACM, 9,* 36–35.