Gu Test: A Progressive Measurement of Generic Intelligence

Author: Scott Lifan Gu **Email: gulifan@hotmail.com**

Abstraction

Do computers already have human level intelligence? Could they understand and process the semantics of irrational numbers without knowing the exact values? Human can. How about uncountable sets? These are necessary to build sciences and real world modeling. Does human intelligence exceed the power of Turing Machine? This paper explains that behavior-based Turing Test cannot measure some intrinsic human intelligence, due to the bottleneck in expression, the bottleneck in capacity, and black box issue, etc. And it does not provide a progressive measurement up to human level intelligence. Similar issues exist in other current testing methods, due to the limitations of behavior-based, knowledge-based or task-based, etc. Measurements based on intrinsic mechanisms could provide better testing. This paper identifies several design goals, to further improve the measurement. Gu Test, a progressive generic intelligence measurement with levels and potential structures, is proposed based on these goals, to measure the intrinsic mechanism for semantics, potential and other intelligence. The semantics of irrational numbers and uncountable sets are identified as two test levels. More work need be done to expand the test feature sets and structures, and provide some suggestions for the direction of future Artificial Intelligence (AI) researches.

1. The Measurement of Generic Intelligence

Machines like clocks can do something better than humans long, long time ago. However, this does not mean these machines have generic intelligence, or human level intelligence. So some measurement of intelligence is needed.

Before discussing the measurement of generic intelligence, there is a question: whether generic intelligence is needed? If throwing in more computing power and design better algorithms based on Turing Machine model can solve all problems, there is no need for generic intelligence.

Unfortunately, computers still lack of something which are in human intelligence. Humans have no idea how to add these into computers so far. Computers cannot write software from beginning. They only run software written by humans, or generate code specified by humans. More generically, humans are highly adaptive, innovative, and can learn many types of knowledge and skills, and can switch from one task to another quickly, etc. Developing intelligence for scientific researches is even more challenging.

Due to such adaptive, innovative, and evolutionary nature, it is extremely difficult to define generic human intelligence accurately, if not impossible. But it is obvious there are big differences between current computers and human intelligence. Testing methods could be used to measure such differences. Clocks can measure time without an accurate definition of time.

Turing Test [1] is the first of such testing methods proposed. Several others were suggested in later years. They could be classified into indistinguishability (or imitation) tests, knowledge aggregation tests, or task aggregation tests, etc.

Testing methods can only test a small portion of intelligence due to time limit and availability. So it is very critical what to test and how to test. However, the existing testing methods cannot test some intrinsic human intelligence capabilities, such as how to understand and use the semantics of irrational numbers and uncountable sets, etc., which are fundamental to sciences and real world modeling. And they cannot test the potentials of humans to develop better capabilities.

Current computers can only approximate the values of irrational numbers with very limited semantics. Due to the sensitivity to intial conditions and exponential divergence in nonlinear chaotic phenomena, there are problems in such approximations. In reality, nonlinearity is the norm rather than exception.

Actually nonlinearity and butterfly effect are the main frustrations to von Neumann's meteorology ambitions. It is highly questionable whether algorithms based on Turing Machine model could accomplish generic intelligence.

Since the existing test methods cannot really measure generic intelligence, they cannot provide good guides to AI researches. Actually very little progress in generic intelligence was made during past decades. It is time to change this. Measurements based on intrinsic mechanisms could provide better testing.

The following sections will discuss the bottlenecks and issues in Turing Test and other existing test methods first. Several design goals are identified to address these issues and provide better measurement. Gu Test, is proposed to accomplish most of these design goals. Some directions for future researches are discussed.

2. Turing Test and Chinese Room Concern

Alan Turing described an imitation game in his paper Computing Machinery and Intelligence, i.e. Turing Test, which tests whether a human could distinguish a computer from another human only via communication without seeing each other.

Turing Test provides two results: pass or fail. It cannot measure partial generic intelligence, i.e. how close a computer system is to generic human intelligence. The testing results depend on the subjective judgement of testers without objective criterions. Objective criterions are needed in scientific experiments, especially for the phenomena in macro physical worlds.

John Seale also raised a Chinese Room issue [2], i.e., computers could pass this test by symbolic processing without really understanding the meanings of these symbols. Due to the limited number of phrases in real usage, it is possible to build a computer system with enough associations between phrases such that humans cannot distinguish the system from humans within limited testing time. However, this does not mean the computers already have human level intelligence.

Chinese Room argument also raise the semantics issue: could computers really understand the

semantics of natural languages?

More important, there are the bottleneck in expression, the bottleneck in capacity, and issues of black box test, etc., as described below, which make Turing Test unable to really test generic intelligence.

Turing Test uses interrogation to test, so it only can test those human characteristics which already be understood well by humans and can be expressed in communication. Some people could manage to understand each other by body languages, rich tones, analogy, metaphor, implication and suggestion, etc., in certain environments, which cannot be expressed in pure symbolic processing. So Turing Test behind veils is not a right way to test these intrinsic intelligence abilities. Even if without veils, current test methods still cannot test those abilities or potentials not understood well by humans yet, which obviously cannot be expressed well yet. This is the bottleneck in expression.

There is also a bottleneck in the capacity of communication or storage: even if those rich subtle varieties of information could be digitized, the size of the information could far exceed the capacity of communication or storage. The current von Neumann architectures only have finite memory units. Turing Machine has infinite but countable memory units. Could Turing Machine be enhanced with uncountable memory units?

There is also a black box issue. Say, a system can produce a huge number of digits of an irrational number. It is impractical to wait for these digits one by one within limited testing time. However, it is straightforward to examine the code to see whether it implements such a feature correctly.

Chinese Room issue, the bottlenecks in expression or capacity, and the black box issue, stem from the testing methods themselves. Due to these problems, certain intrinsic intelligence and potentials cannot be tested with such methods.

However, with white box methods, people could measure mechanisms, instead of behaviors. The designers of the systems could explain what and how they implement in their software and hardware. Testers could analyze whether these claims are true or false based on reasoning, and examine the systems to see whether they are implemented as expected. This is the procedure of Gu Test.

3. Other Test Methods

There are several other methods aiming at testing generic intelligence. Although some of they could provide some test levels, they cannot measure higher level intelligence close to humans, because they do not measure the mechanisms behind generic intelligence. So they lack the understanding and processing of real semantics, and cannot test the potentials of human development.

One is Feigenbaum test. According to Edward Feigenbaum,"Human intelligence is very multidimensional", "computational linguists have developed superb models for the processing of human language grammars. Where they have lagged is in the 'understand' part", "For an artifact, a computational intelligence, to be able to behave with high levels of performance on complex intellectual tasks, perhaps surpassing human level, it must have extensive knowledge of the domain." [3].

Feigenbaum test is actually a good method to test the knowledge in expert systems. The test tries to produce generic intelligence by aggregating many expert systems. That is why it needs to test extensive

knowledge.

However, since these types of knowledge are still expressed and stored in symbolic data, the bottlenecks of expression or capacity still exist. It is still a black box test. Although it tries to solve the "understand" part, there are no solutions so far to test real semantics of knowledge from these symbolic data.

Another issue of Feigenbaum test is: individual humans may not have very extensive knowledge in many domains, but they have certain potentials. So testing extensive knowledge may not be necessary, if not impossible. What to be figured out is how to test these potentials.

Minimal Intelligent Signal Test (MIST) [4] is similar to Feigenbaum test. But it only uses binary answer "yes" or "no" as test results so it can leverage statistical inference to analyze the test results. The bottlenecks in expression and capacity still exist. It is still a black box testing. By using binary answers, it oversimplifies the knowledge with even less understanding of semantics than Feigenbaum test.

Another method is Shane Legg and Marcus Hutter's solution [5], which is actually agent-based, a good test for the performance of specific tasks. In their framework, an agent sends its actions to the environment and receives observations and rewards from it. If their framework is used to test generic intelligence, then it assumes that all the interactions between humans and their environment could be modeled by actions, observations, rewards, etc. This assumption has not been tested yet. The bottlenecks in expression or in capacity still exist in the definitions of actions, observations, rewards, etc.

Furthermore, Humans have very diversified specialties. It is impractical to aggregate performance for a very large number of tasks. Humans have the potentials to learn new tasks and be innovative. They could gain deeper observations, take better actions, and gain other rewards than what in the specified task definitions. Such potentials cannot be tested in the black box performance testing for specified tasks. So this method does not really test the generic intelligence, too.

If Turing Test is enhanced with vision and manipulation ability, it could become similar to Shane Legg and Marcus Hutter's solution. Interrogation could become task performing. Even if test not behind veils, these bottlenecks and issues still exist.

In a summary, the existing testing methods do not measure generic intelligence well as expected. As a result, the studies of generic intelligence are still clueless. To design a better measurement of generic intelligence, the existing bottlenecks and issues should be resolved. Some design goals should be identified to provide good directions and better solutions.

4. The Design Goals for Better Measurement of Generic Intelligence

Based on the analysis done in previous sections, some design goals are suggested here:

1) Resolve Chinese Room issue, i.e., to test the real understanding of semantics, not just behavior imitating or symbolic processing.

2) Resolve the bottleneck in expression, by not purely relying on interrogation. Find some ways to test those intrinsic intelligence abilities which have not been understood and expressed well.

3) Resolve the bottleneck in capacity, by leverage of some properties of concepts and semantics.

4) Use white box test to examine the implemented mechanisms directly.

5) Involve as less domain knowledge as possible, since regular humans may not have much knowledge in specific domains. But find some ways to test the potentials to develop intelligence.

6) Develop leveled test schemes up to generic human intelligence, to measure continuous progress in intelligence.

7) Develop a framework to test structured and associated intelligence, adaptive and innovative abilities, and diversified specialties, etc.

5. Gu Test

Based on these design goals, Gu Test is proposed. It comprises a testing procedure with some selected testing features and structures, so it can measure the intrinsic mechanisms for intelligence. Initially it includes two test levels: the understanding and processing of the semantics of irrational numbers and uncountable sets.

More levels and structures could be added in future. However, to make it possible to test the full range of intelligence, it should include only critical features as less as possible.

Humans can derive new usages of irrational numbers without knowing the exact values of these numbers. Obviously they understand these semantics. The situation is similar with uncountable sets, but at a more difficult level, whereas regular people with average education have the potential to understand irrational numbers. These intelligences are critical to sciences and real world modeling.

Gu Test is to test whether computers or machines have such intelligence. Humans own such abilities or potentials, but they do not understand why and how these work, and cannot express these semantics, potentials, and intelligence as pure symbolic data yet.

It is a white box test. The test procedure is as below:
1) It is up to the designers of the systems to explain what semantics, potentials, or other intelligence, etc., they want to implement and how. In this way, Gu Test does not restrict what and how the designers want to implement, and allows full exploration..
2) Testers analyze whether these claims are true or false based on reasoning. The interpretation and representation of intelligence features only can be judged based on reasoning.
3) Testers examine the software and hardware of the systems, to see whether these mechanisms (including whatever representation of intelligence passed in step 2) are really implemented as expected.

This procedure could be applied to the selected features including irrational numbers and uncountable sets, etc., or to other intelligence features included in Gu Test in future, or those claimed by customers. The procedure also could be applied to low level claims, such as whether some mechanisms at physical level, biological level, or psychological level, etc., are accomplished.

The test does not rely on black box interrogation. So it opens the door for designers' and testers' imaginations to test whatever intelligence or mechanisms humans have, without the external bottlenecks in expression or capacity stemming from testing methods.

Irrational number is a primitive concept developed in Pythagoras' age. The concept is necessary to so many domains, but involves very little domain-specific knowledge. Uncountable set is an advanced concept used in modern sciences and mathematics. Physical semantics could be in complete different dimensions. It would be very different challenges to add intelligence in different domains, although concepts such as time, distance and energy could be good potential candidates.

The current efforts are to achieve the design goals 1) to 6). The work to meet goal 7), i.e., to test structured and associated intelligence, adaptive and innovative abilities, and diversified specialties, etc., will be left to future researches.

6. The Comparison With other Test Methods

As said, Gu Test is very different from indistinguishability (or imitation) tests, knowledge aggregation tests, or task aggregation tests, etc. As said, it only selects critical testing features as less as possible. It is a white box test. It requires humans designers to explain what intelligence their systems implement and how, and human testers to analyze whether these claims are true or false and examine the systems to see whether they implement these mechanisms as expected. So it can test intrinsic mechanisms instead of behaviors, knowledge, or tasks, etc.

It does not have the bottlenecks in expression or in capacity stemming from testing methods, and could test higher level intelligence such as semantics understanding up to or even beyond human intelligence.

Gu Test represents a complete paradigm shift from previous test methods. It provides some guides or insights related to generic human intelligence, without restricting how to implement these.

7. Future Research

Much more work need be done to add more test levels to Gu Test and meet the design goals 7).

The analysis on the bottlenecks and issues of Turing Test, would naturally lead to the questions of the power and limitations of Turing Machine and von Neumann architecture. This paper does not make any conclusion on what platforms or architectures are better for generic intelligence, as long as they could truly pass the test. Rather, it opens the door to allow people to make full exploration.

To really understand the essentials of intelligence, people have to study the history of knowledge development, including philosophy, mathematics, and sciences, etc. It is a reasonable option to develop intelligence models based on a multi-level structure of physics, life sciences, and psychology.

References

[1] Turing, A. M., 1950, "Computing machinery and intelligence". Mind 59, 433–460.
[2] Searle, John. R., 1980, "Minds, brains, and programs". Behavioral and Brain Sciences 3 (3): 417-457.
[3] Feigenbaum, Edward A., 2003, "Some challenges and grand challenges for computational intelligence". Journal of the ACM 50 (1): 32–40.
[4] McKinstry, Chris, 1997, "Minimum Intelligent Signal Test: An Alternative Turing Test", Canadian Artificial Intelligence (41)
[5] Legg, S. & Hutter, M., 2006, "A Formal Measure of Machine Intelligence", Proc. 15th Annual Machine Learning Conference of Belgium and The Netherlands, pp.73-80.