Formal Models in AGI Research

Pei Wang

Temple University, Philadelphia PA 19122, USA
http://www.cis.temple.edu/~pwang/

Abstract. Formal models are necessary for AGI systems, though it does not mean that any formal model is suitable. This position paper argues that the dominating formal models in the field, namely logical models and computational models, can be misleading. What AGI really needs are formal models that are based on realistic assumptions on the capacity of the system and the nature of its working environment.

1 The Power and Limit of Formal Models

The need for formal models for AGI research is not a novel topic. For example, AGI-09 had a workshop titled "Toward a Serious Computational Science of Intelligence" [1]. In [2], I proposed the opinion that a complete A(G)I work should consist of (1) a *theory* of intelligence, expressed in a natural language, (2) a formal *model* of the theory, expressed in a symbolic language, and (3) a computer *implementation* of the model, expressed in a programming language. Though the necessity of (1) and (3) are obvious, there is a large number of AGI projects without a clearly specified formal model. Such projects are often described and carried out according to the common practice of software engineering.

If an AGI system is eventually built as a computer system with software and hardware, why bother to have a formal model as an intermediate step between the conceptual design and the physical implementation? As I argued in [3], formalization improves a theoretical model by disambiguating (though not completely) its notions and statements. In particular for AGI, a formal model tends to be domain independent, with its notions applicable to various domains by giving the symbols different interpretations. Though it is possible to skip formalization, such a practice often mixes the conceptual issues and the implementational issues, thus increasing the complexity of a system's design and development.

However, to overemphasize the importance of formalization for AGI may lead to the other extreme, that is, to evaluate a formal model for its own sake, without considering its empirical justification as a model of *intelligence*, or its feasibility of being implemented in a *computer system*. Though the rigor and elegance of a model are highly desired, they are still secondary when compared with the correctness and applicability of the fundamental assumptions of the model. A mathematical theory may have many nice properties and may solve many practical problems in various fields, but this does not necessarily mean that it will be equally useful for AGI. Actually it is my conjecture that a major

reason for the lack of rapid progress in this field is the dominance of the *wrong* formal models, in particular, those based on mathematical logic, the theory of computation, and probability theory. In this paper, I summarize my arguments against certain logical models and computational models in AGI.

2 Logical Models and AGI

As I argued in [4, 2], mathematical logic was established to provide a logical foundation for mathematics, by formalizing the valid inference patterns in theorem proving. However, "theorem proving" is very different from commonsense reasoning, and this conclusion has been reached by many logicians and AI researchers. Consequently, various non-classical logics and reasoning models have been proposed, by revising or extending traditional mathematical logic [5, 6]. Even so, the following fundamental assumptions in classical logic are still often taken for granted:

Correspondence theory of truth: The truth-value of a statement indicates the extent to which the statement corresponds to an objective fact.

Validity as truth-preserving: An inference rule is valid if and only if it derives true conclusions from true premises.

My own AGI project NARS is a reasoning system that rejects both of the above assumptions. Instead, they are replaced by two new assumptions:

Empirical theory of truth: The truth-value of a statement indicates the extent to which the statement agrees with the system's experience.

Validity as evidence-preserving: An inference rule is valid if and only if its conclusion is supported by the evidence provided by its premises.

Based on the above assumptions, as well as the assumption that an intelligent system should be adaptive and can work with insufficient knowledge and resources, NARS is designed, which implements a formal logic [7, 2, 8]. NARS fundamentally differs from mathematical logic, since it is designed to work in realistic situations, while the latter is for idealized situations.

NARS consistently handles many issues addressed in non-classical logics:

Uncertainty: NARS represents several types of uncertainty, including randomness, fuzziness, ignorance, inconsistency, etc., altogether as the effects of various forms of negative or future evidence.

Ampliativity: Beside deduction, NARS also carries out various types of non-deductive inference, such as induction, abduction, analogy, and other types of inference that produce "ampliative" conclusions.

Openness: NARS is always open to new evidence, which may challenge the previous beliefs of the system, and therefore lead to belief revisions and conceptual changes.

Relevance: The inference rules not only demand truth-value relationships between the premises and the conclusions, but also semantic relationships, that is, their contents must be related.

Both in logic and in AI, the above issues are usually addressed separately, and a new logic is typically built by extending or revising a single aspect of classical logic, while leaving the other aspects unchanged [6, 9, 10]. NARS takes a different approach, by treating the issues as coming from a common root, that is, the assumption on the insufficiency of knowledge and resources [4, 2].

3   Computational Models and AGI

Since an AGI will eventually be implemented in a computer system, it is often taken for granted that all processes in the system should be designed and analyzed according to the theory of computation. Concretely, it means the *problem* the system needs to solve will be defined as a *computation*, and its *solution* as an *algorithm* that can be implemented in a computer [11].

I have argued previously that such a conceptual framework is not suitable for AI at the problem-solving level [7, 2]. Like mathematical logic, the theory of computation also came from the study of problem solving in mathematics, where the "problems" are abstracted from their empirical originals, and the "solutions" are expected to be *conclusively correct* (i.e., cannot be refuted or revised later), *context independent* (i.e., having nothing to do with where and when the problem appears), and *expense irrelevant* (i.e., having nothing to do with how much time has been spent on producing it).

Therefore, problem solving in mathematics can be considered as "time-free" and repeatable. When dealing with abstract problems, such an attitude is justifiable and even preferred — mathematical solutions should be universally applicable to different places in different times.

However, the problem-solving processes in intelligence and cognition are different, where neither the problems nor the solutions are time-free or accurately repeatable. In practical situations, most problems are directly or indirected related to predictions of future events, and therefore have time requirements attached. In other words, "solving time" is part of the problem, and a "solution" coming too late will not qualify as a solution at all. On the other hand, the solutions for a problem usually depend on the system's history and the current context. This dependency comes from the adaptive nature of the system and the real-time requirement. By definition, in an adaptive system the occurrences of the same problem get different solutions, which are supposed to be better and better in quality. Therefore, each occurrence of a problem is unique, if the system's state is taken into consideration as a factor. For an adaptive system, usually in its lifetime its internal states never repeat, and nor does its external environment. Consequently, its problem-solving processes cannot be accurately repeatable.

Though it is still possible to focus on the relative stable aspects of an intelligent system, so as to specify its stimulus–response relationship as a *function*, in the sense that the same (immediate) input always lead to the same (immediate) output. However, such a treatment excludes some of the prominent features of intelligence, such as its adaptivity, originality, and flexibility.

For instance, from the very beginning of AI, "learning" has been recognized by many researchers as a central aspect of intelligence. However, the mainstream "machine learning" research has been carried out in the framework of computation:

The objective of learning is to get a function. At the object level, though during the learning process the same problem instance gets multiple solutions with improving quality, it is usually expected that the problem-solution relation will eventually converge to a function,

The learning process follows an algorithm. At the meta-level, each type of learning is usually defined as a computation, and follows an algorithm, with the training data as input, and the learned function as output.

The human learning process does not fit into this framework, because it is usually open-ended, and does not necessarily converge into a stable function that maps problems into solutions. Even after intensive training in a certain domain, an expert can still keep the flexibility and adaptivity when solving problems. Also, human learning processes do not follow fixed procedures, because such processes are usually not accurately predictable or repeatable.

The above conclusions do not mean that intelligence has nothing to do with computation. At a certain level of description, the activities of an intelligent system can be analyzed as consisting of many "basic steps", each of them is repeatable and can be specified as a computation following a fixed algorithm. It is just that a problem-solving process typically consists of many such steps, and its composition depends on many factors that are usually not repeated during the system's life cycle.

Such a formal model is provided in NARS [7, 2, 8]. As a reasoning system, NARS solves each problem by a sequence of inference steps, where each step is a simple computation, but since the steps are linked together at run time according to many ever-changing factors, the problem-solving process cannot be considered as a computation, since it is not repeatable.

Here I want to argue that "intelligence" should be formalized differently from "computation". As far as *time* is concerned, the differences are:

The time-dependency of problem. In *computation*, a problem-solving process can take an arbitrarily long time, as long as it is finite. In *intelligence.* a problem-solving process is always under a time pressure, though the time requirement is not necessarily represented as a hard deadline. In general, the utility value of a solution decreases over time, and a solution may lose most of its utility if it is found too late. When time-pressure changes, the problem is also more or less changed.

The time-dependency of solution. In *computation*, the correctness of a solution has nothing to do with when the problem appears, but in *intelligence*, it does. Whether a solution is reasonable should be judged not only according to the problem, but also the available knowledge and resources at the moment. A reasonable solution obtained by a student in an emergency may not be reasonable when provided by an expert after a long deliberation.

To implement such a model, NARS is designed in the following way:

– Each inference task has a priority-value attached to reflect the (relatively defined) *urgency* for it to be processed. Similarly, each belief and concept in the system has a priority-value attached to reflect its *importance* at the moment. All these values are adjusted by the system according to its experience.
– The selection of inference rules is data-driven, decided by the task and belief winning the system's attention at the moment. Since the selection of task and belief is context-sensitive, so is the overall inference process.

In this way, it is possible to implement a non-computable process using computable steps [2].

4   Summary

AGI research needs formal models of intelligence and cognition. Since all the existing models were designed for other purposes, they should not be directly applied without fundamental revision. Here the key issue is not in the specific features of a model, but in the basic assumptions behind it. More effort should be put into the developing of new formal models that satisfy the requirements of AGI, though the task is difficult and the result will not be perfect soon.

References

1. Bringsjord, S., Sundar G, N.: Toward a serious computational science of intelligence (2009) Call for Papers for an AGI 2010 Workshop.
2. Wang, P.: Rigid Flexibility: The Logic of Intelligence. Springer, Dordrecht (2006)
3. Wang, P.: Theories of artificial intelligence – meta-theoretical considerations. In Wang, P., Goertzel, B., eds.: Theoretical Foundations of Artificial General Intelligence. Atlantis Press, Paris (2012) 305–323
4. Wang, P.: Cognitive logic versus mathematical logic. In: Lecture notes of the Third International Seminar on Logic and Cognition, Guangzhou (2004) Full text available online.
5. McCarthy, J.: Artificial intelligence, logic and formalizing common sense. In Thomason, R.H., ed.: Philosophical Logic and Artificial Intelligence. Kluwer, Dordrecht (1989) 161–190
6. Haack, S.: Deviant Logic, Fuzzy Logic: Beyond the Formalism. University of Chicago Press, Chicago Press (1996)
7. Wang, P.: Non-Axiomatic Reasoning System: Exploring the Essence of Intelligence. PhD thesis, Indiana University (1995)
8. Wang, P.: Non-Axiomatic Logic: A Model of Intelligent Reasoning. World Scientific, Singapore (2013).
9. Gabbay, D.M.: Logic for Artificial Intelligence and Information Technology. College Publications, London (2007)
10. Russell, S., Norvig, P.: Artificial Intelligence: A Modern Approach. 3rd edn. Prentice Hall, Upper Saddle River, New Jersey (2010)
11. Marr, D.: Vision: A Computational Investigation into the Human Representation and Processing of Visual Information. W. H. Freeman & Co., San Francisco (1982)