History of Artificial Intelligence Before Computers

Bruce MacLennan
University of Tennessee, USA

INTRODUCTION

The history of artificial intelligence (AI) is commonly supposed to begin with Turing's (1950) discussions of machine intelligence, and to have been defined as a field at the 1956 Dartmouth Summer Research Project on Artificial Intelligence. However, the ideas on which AI is based, and in particular those on which symbolic AI (see below) is based, have a very long history in the Western intellectual tradition, dating back to ancient Greece (see also McCorduck, 2004). It is important for modern researchers to understand this history for it reflects problematic assumptions about the nature of knowledge and cognition: assumptions that can impede the progress of AI if accepted uncritically.

BACKGROUND

Symbolic AI is the approach to artificial intelligence that has dominated the field throughout most of its history and remains important. It is based on the physical symbol system hypothesis, enunciated by Newell and Simon (1976), which asserts, "A physical symbol system has the necessary and sufficient means for general intelligent action." In effect, it implies that knowledge is represented in the brain by language-like structures, and that thinking is a computational process that rearranges these structures according to formal rules. This view has also dominated cognitive science, which applies computational concepts to understanding human cognition (Gardner, 1985).

Many symbolic AI systems are based on formal logic, which represents propositions by symbolic structures, in which all meaning is conveyed in the structure's form, and which implements inference by the mechanical manipulation of those structures. Therefore, we will discuss the origins of formal logic and of the idea that knowledge and inference can be represented in this way. We will also consider the combinatorial methods used before the invention of computers as well as in modern AI for generating possible solutions to a problem, which leads to combinatorial explosion, a fundamental limitation of symbolic AI. Then we describe early modern attempts to design comprehensive knowledge representation languages (predecessors of those used in symbolic AI) and mechanical inference machines. We conclude with a mention of alternative views of knowledge and cognition.

THE HISTORICAL ROOTS OF SYMBOLIC AI

Formal Logic

It is surprising, perhaps, that the original inspiration for symbolic knowledge representation can be found in ancient Greece, in particular in Pythagorean number theory (Burkert, 1972, Riedweg, 2005). In ancient Greece, as in many cultures, ancient and modern, pebbles were used for calculation by being moved in grooves in a similar way to the beads on an abacus. Indeed, the Latin word for *pebble* is *calculus*, and our word *calculate* comes from this manipulation of *calculi* (pebbles). In logic and mathematics, we use the word *calculus* for any system of notation in which we can accomplish some purpose by the manipulation of tokens according to formal, game-like, mechanical rules. (For example we have differential and integral calculi in mathematics and propositional and predicate calculi in logic.) To the extent that the rules are purely mechanical, they can, in principle, be carried out by a machine, which is why calculi are important in AI; if a process can be reduced to a calculus, it can be calculated by a machine.

The ancient Pythagoreans (Pythagoras, 572–497 B.C.E.) investigated number theory by means of arrangements of pebbles (Burkert, 1972; Riedweg, 2005). For example, they observed that certain numbers could be arranged into a square shape, and we still call these numbers *squares*. However, they also investigated triangular numbers as well as rectangles, pentagons, cubes, pyramids, and so forth. Although they did not prove theorems in the modern sense, they were able to demonstrate the truth of theorems in number theory by means of these arrangements. Thus, they discovered calculi could be used for reasoning as well as computation.

According to tradition, Pythagoras was the first to explain consonant musical intervals in terms of numerical ratios (Burkert, 1972). For example, a string one-half the length of another string sounds an octave higher; the shorter of two strings of lengths with the ratio 2:3 sounds a fifth higher, and so forth. Thus, a subtle perceptual distinction (the relative consonance of pitches) could be rendered logical and rational by reducing it to numerical ratios (Greek *logoi* and Latin *rationes*, terms that also refer to the articulation of thought in words or symbols; Maziarz & Greenwood, 1968). It is an example of the representation of expertise in terms of formal structures; judgments of consonance can be replaced by calculation.

The Pythagoreans believed that everything could be reduced to numbers and thus made intelligible, rational, and logical (Burkert, 1972; Burnet, 1930). Therefore, they were committed to the idea that all knowledge could be represented in terms of arrangements of otherwise meaningless tokens, that is, in formal structures (and hence, we may conclude, in computer data structures).

Aristotle (384-322 B.C.E.) is known as the originator of the science of logic, but two of his contributions in this area are especially relevant to AI. First, he began the development of formal logic by showing that valid inference could be distinguished from invalid inference on the basis of its form rather than on the meaning of its particular terms (words). In other words, Aristotle showed that valid inference is a matter of syntax (the grammatical form of an argument) rather than semantics (its meaning). This is important because it shows how inference can be carried out by the manipulation of symbols independently of their meaning, which means that, in principle, inference is a kind of computation. Stated differently, there is a calculus of valid inference.

Aristotle also began the study of modal logic, that is, logic in which propositions are not simply true or false, but in which the propositions may be possible, impossible, necessary, or contingent (Bocheński, 1970; Kneale & Kneale, 1962). Modal logic and its derivatives (such as tense logic, which deals with propositions whose truth values may change in time) are important in AI (Sowa, 1984).

Another contribution of Aristotle was the organization of knowledge into formal deductive structures, in which all the facts of a science were either stated as axioms or formally derivable from the axioms. The best-known example is Euclidean geometry, which was the exemplar of a systematic body of knowledge for over two millennia (Maziarz & Greenwood, 1968). Similar formal axiomatic structures are used in AI for representing a knowledge domain.

The investigation of logic continued over the following centuries. For example, the medieval scholastics (roughly 6th to15th centuries) refined logic into a very precise instrument, although it was still based on a natural language (Latin) in contrast to modern symbolic logic. As a consequence, they became conscious of the limitations of natural language for exact knowledge representation and strove to compensate for its deficiencies. For example, they knew that the word *dogs* is used differently in the propositions "dogs are mammals" and "*dogs* is a plural noun." AI knowledge representation languages have to deal with similar issues (Sowa, 1984). In the end, dissatisfaction with natural languages led to an interest in developing artificial languages that were intended to be more rational (logical and precise). Behind this was the assumption that there is a universal grammar underlying all natural languages, and that it corresponds to the "language of thought"; therefore an artificial language, as an ideal vehicle for thought, ought to reflect this deep structure. Similar motivations underlie the development of AI knowledge representation languages (see below).

Combinatorial Methods

The Middle Ages also saw the development of combinatorial approaches to solving problems (Bocheński, 1970). For example, the medieval scholastics used a combinatorial procedure to generate the 192 possible Aristotelian syllogisms, and then they crossed out the invalid ones. This is an example of a generate-and-test procedure, an approach still widely used in AI. The problem with generate-and-test procedures is combinatorial explosion: The number of combinations to be tested increases exponentially with their size.

These combinatorial procedures acquired an increased significance, which contributed to the eventual development of AI, from the kabbalah, a Jewish mystical tradition with Pythagorean affinities, which became popular in the Middle Ages (Eco, 1997; Scholem, 1960). According to this tradition, the text of the Torah reflects the logos (rational structure) of the universe. Therefore, since the Torah is written in the letters of the Hebrew alphabet, these letters correspond to the elementary categories and archetypal forms underlying the universe. As a consequence, the letters of the Hebrew words for things reveal their logical structure to one who knows how to interpret them. Combinatory processes figure prominently in kabbalah, and significant words, especially the names of God, were permuted in order to reveal hidden wisdom and discover new truths. For this purpose the kabbalists used rotating wheels and other devices to ensure that they did not omit any combinations of letters, an example of a mechanized generate-and-test procedure.

Similar in spirit to the kabbalah, and perhaps in part inspired by it, was the Great Art (Ars Magna) of Raymond Lull (also spelled Llull, 1232-1315; Bonner, 1985; Johnston, 1987; Yates, 1966). He intended it to be a "universal science of all sciences," a systematic method by which knowledge could be discovered and proved. There were several versions of his system, but the most common one made use of nine "divine dignities," or attributes of God, which took different forms in each domain of knowledge but provided the fundamental categories in each domain. These abstract qualities (Goodness, Magnitude, Duration, etc.) correspond closely to certain kabbalistic names of God. In Lull's Art, as in kabbalah, we see an attempt to isolate the most basic categories that constitute all knowledge and to discover, therefore, an alphabet of thought. This remains an important goal in contemporary symbolic AI.

A distinctive characteristic of Lull's Art was the extensive use of rotating wheels to generate combinations of these elementary categories in order to discover and to demonstrate philosophical truths. Thus, the Great Art combines an alphabet of elementary concepts with mechanical procedures for generating their combinations in order to produce an automated method of knowledge discovery and proof. Such, at least, was its goal. In fact, it did not work, and for the most part it could be used only for proving the theological propositions that the operator already believed. Nevertheless, it inspired many later thinkers to attempt to correct its deficiencies and to construct machines for knowledge discovery and inference, but first it had to be recast into a more logical form.

Knowledge Representation and Mechanized Inference

As knowledge and inference became more systematized, the idea developed that reasoning, when carefully and methodically executed, was a kind of calculation. One clear exponent of this view was Thomas Hobbes (1588-1679), who said, "By *ratiocination* I mean *computation*" (*Elem. Phil.*, 1.1.1.2). He explained, however, that the addition and subtraction of concepts was not the same as the addition and subtraction of numbers.

Hobbes also distinguished reasoning from causes to their effects (forward chaining in modern AI terminology) and reasoning from effects to their causes (backward chaining). In both cases, thought is a kind of mental discourse, which corresponds to a defining assumption of symbolic AI: that there is a language of thought (sometimes called "Mentalese"). Words, whether external or in the mind, are tokens, manipulated according to mechanical rules, and correct reasoning is analogized to balancing account books. That is, thought is calculation. Furthermore, since Hobbes was a complete materialist, he understood thought as a kind of matter in motion, a strictly mechanical process.

Over the centuries there have been many attempts to design ideal languages, that is, artificial languages without the perceived deficiencies of natural languages (Eco, 1997; Large, 1985). As modern science emerged in the 17th century, the goal was often to develop a philosophical language, that is, a language suitable for philosophical analysis and scientific discourse. One of the most famous of these projects was the Real Character of John Wilkins (1614-1672), the first president of the Royal Society (Large; Lewis, 2007; Rossi, 2000; Vickers, 1987). He began by isolating a universal grammar that he believed to underlie the particular grammars of all natural languages, and so it is in effect the grammar of Mentalese and hence reflects the laws of thought. Inspired by Chinese writing, Wilkins also concluded that the forms of words should reflect their logical analysis (based on a class hierarchy), and he designed a vocabulary and symbolic writing system based on a comprehensive conceptual taxonomy. His language had little direct impact beyond inspiring the conceptual taxonomy used by *Roget's Thesaurus*, but symbolic logic and AI knowledge representation languages have similar goals and approaches (and, arguably, similar failings).

Gottfried Wilhelm von Leibniz (1646-1716) investigated knowledge representation and mechanized reasoning, in which he was influenced by kabbalah, Lull, Wilkins' language, Hobbes, and Chinese writing and philosophy (Buchanan, 2005; Coudert, 1995; Kneale & Kneale, 1962; Perkins, 2004; Styazhkin, 1969). For example, although he had already invented the binary number system, he later found it in the Chinese *I Ching* (*Book of Changes*) and saw how it reduced all change in the universe to two opposites (yin and yang). This accorded with the kabbalistic and Lullian idea that the world was organized in terms of an alphabet of fundamental ideas and with his own rationalistic philosophy, which sought the true essences of concepts in a small number of atomic (indivisible) categories.

Leibniz was very impressed by Lull's Great Art and by Wilkins' Real Character, but concluded that they would not work, and so he constructed a number of knowledge representation schemes on a more logical plan. For example, any positive integer can be decomposed into a unique product of prime numbers, which is analogous to the rationalist idea that any concept can be reduced to a unique conjunction of atomic concepts. Therefore, if a prime number were assigned to each atomic concept, then every possible concept would have a unique numerical value. Conversely, if we looked up in a philosophical dictionary the number corresponding to any concept, we could discover its essence, or true definition, by reducing the number into its prime factors.

There are two ways that classes are treated in mathematics and logic: extensionally and intensionally. The extensional approach is to define a class in terms of its members, its extension, whereas the intensional approach defines it in terms of its intension, or essential attributes. Although modern logic and mathematics tend to treat classes extensionally, AI treats them intensionally (i.e., a concept is represented by a property list) for the simple reason that most concepts have small intensions but infinite extensions, so it is easier to compute with intensions. For the same reasons, Leibniz settled on an intensional representation.

Leibniz agreed with Hobbes' assertion that thought is computation, and worked on a calculus for logical inference. For example, he discovered that propositions of the form "all *S* are *P*" can be decided computationally if we know the numbers corresponding to *S* and *P*. For if all *S* are *P*, then the essential attributes of *P* are among the essential attributes of *S*; numerically, the prime factors of *P* are among the prime factors of *S*. Therefore, to decide if a proposition "all *S* are *P*" is true, all we need to do is to look up the numbers for *P* and *S* and see if the number for *P* evenly divides the number for *S*.

In summary, we can see that Leibniz had all the components of a system of knowledge representation and mechanical inference. In principle, all concepts could be analyzed into a relatively small number of elementary atomic concepts, and each concept could be assigned a unique number on the basis of this analysis. All philosophical questions, then, could be answered rationally and logically by calculation, literally by ratios (*rationes*, *logoi*). Indeed, Leibniz constructed one of the earliest digital calculating machines (1671), the first capable of multiplication and division, and so he had in principle (but not capacity) the means for actual mechanized reasoning.

George Boole (1815-1864) is well known to computer scientists and information technologists as the inventor of Boolean algebra, which is applied to digital circuit design and in many other ways to computer technology. However, his goals were much more ambitious, for in his *Investigation of the Laws of Thought*, he says he intends "to investigate the fundamental laws of those operations of the mind by which reasoning is performed" and to express them in a calculus (Boole, 1854, p. 1). In common with contemporary logicians, he expressed logical operations in an algebraic notation as opposed to a natural language, thus contributing to the development of symbolic logic. He developed an extensional class logic, in which operations on classes correspond to operations on their extensions, that is, on the sets of their members, and so he invented the algebra of sets. However, he also showed how the same algebraic operations could be interpreted as a propositional logic, which laid the foundation for Boolean circuit design, later developed by Claude Shannon (1938), the inventor of information theory. Boole stressed the formality of his logic, that is, that its rules of inference depended only on the algebraic properties of the operators (commutativity, associativity, etc.) and not on any interpretation of the terms. Therefore, these operation were not restricted to human thought, but could be implemented by machines, which was accomplished about a decade later by W. S. Jevons.

William Stanley Jevons (1835-1882) was a prolific mathematician, scientist, and philosopher who contributed to statistics, economics, meteorology, and the philosophy of science (Mays & Henry, 1953). However, he was also the first to construct fully functional logic machines capable of automated reasoning (Jevons, 1870, 1894, 1958), and thus predecessors of AI technology. His system is based, first, on the idea of a logical alphabet that lists all the possible conjunctions of a given set of terms and their complements. Second, he uses an indirect method of deduction, which is simply to eliminate from the logical alphabet those combinations that are inconsistent with the premises. Obviously, this is a generate-and-test procedure: List all the possible combinations and remove the impossible conclusions; the result is the broadest conclusion compatible with the premises.

Jevons' indirect method, like most generate-and-test procedures, is too tedious and error prone to perform manually, so he invented a succession of devices that increasingly automated the process. His most sophisticated was a completely mechanical device, called the logical piano. It had a keyboard marked with the terms (*A*, *B*, *C*, *D*, and their complements) and with various logical symbols, which was used for entering a series of logical equations representing the premises of a deduction. Above the keyboard was a (mechanical) display, a kind of spreadsheet, which represented all of the logical combinations consistent with the premises that had been entered so far. Thus, the operator could watch the developing mathematical analysis, and even try out hypothetical premises to see how they might affect the conclusion. In 1869, Jevons constructed and demonstrated a four-term machine and planned the development of a 10-term reasoning engine, which would have required an entire wall to display the 1,024 combinations of its logical alphabet. Although the machine performs relatively simple operations on bit strings, Jevons (1958, pp. 110-111) enthused that "after the Finis key has been used the machine represents a mind endowed with powers of thought," and that as each proposition is entered, "the machine analyses and digests the meaning of it and becomes charged with the knowledge embodied in that proposition." Thus, Jevons invented AI hype!

FUTURE TRENDS

In the light of this history, symbolic AI, which has dominated AI research, can be seen as the continuation of a centuries-old tradition concerning the nature of knowledge and inference. This, of course, does not imply that it is the best approach to AI, or conversely that it is not. Although some prominent researchers have declared that the symbolic approach to cognition is "the only game in town," there are alternatives, most notably connectionism (or parallel distributed processing), which is based on simplified models of neural networks in the brain (Garson, 2007; Rumelhart, McClelland, & PDP Research Group, 1986). This new approach promises to compensate for many of the limitations of the symbolic approach, and also to shed light on cognitive processes in the brains of humans and other animals. Connectionism, however, is beyond the scope of this article.

This article has focused on a few of the principal thinkers who contributed to AI before the era of the computer, but there were many others. Therefore, much work remains to be done in exploring and explaining the intellectual background of artificial intelligence. Through a deeper understanding of the assumptions we bring to knowledge and cognition, we may see new approaches to AI technology.

CONCLUSION

We may draw several conclusions from this historical survey. First, symbolic AI is built upon a foundation of philosophical and psychological premises that have been part of Western intellectual history since ancient Greece. Since these assumptions are so deeply embedded in our intellectual background, they easily may be taken for granted and escape adequate scrutiny. Nevertheless, alternatives, such as connectionism, are being explored. Second, although earlier philosophers discussed the idea that thought is a kind of computation, it was only with the advent of modern computers that there was sufficient computing power to test these theories empirically. As a consequence, experimental AI research in the late 20th century revealed both the capabilities and limitations of symbolic AI and motivated the search for alternatives. Finally, perhaps the most important conclusion is that AI is not an isolated technological discipline, nor simply the applied side of cognitive science, but it is intimately related to intellectual issues about the mind and thought that have occupied civilization for millennia.

REFERENCES

Barnes, J. (1982). *Aristotle*. Oxford, United Kingdom: Oxford University Press.

Bocheński, I. M. (1970). *A history of formal logic.* New York: Chelsea Publishing.

Bonner, A. (Ed. & Trans.). (1985). *Selected works of Ramon Llull (1232–1316)* (Vol. 1). Princeton, NJ: Princeton University Press.

Boole, G. (1854). *An investigation of the laws of thought.* New York: Macmillan.

Buchanan, B. G. (2005). A (very) brief history of artificial intelligence. *AI Magazine*, pp. 53-60.

Burkert, W. (1972). *Lore and science in ancient Pythagoreanism.* Cambridge, MA: Harvard University Press.

Burnet, J. (1930). *Greek philosophy part I: Thales to Plato.* London: Macmillan.

Coudert, A. P. (1995). *Leibniz and the kabbalah*. Dordrecht, The Netherlands: Kluwer.

Eco, U. (1997). *The search for the perfect language* (J. Fentress, Trans.). Cambridge, MA: Blackwell.

Gardner, H. (1985). *The mind's new science: A history of the cognitive revolution.* New York: Basic Books.

Garson, J. (2007). Connectionism. In E. N. Zalta (Ed.), *The Stanford encyclopedia of philosophy.* Retrieved from http://plato.stanford.edu/archives/spr2007/entries/connectionism/

Jevons, W. S. (1870). On the mechanical performance of logical inference. *Philosophical Transactions, 160*, 497.

Jevons, W. S. (1894). *Elementary lessons in logic: Deductive and inductive.* New York: Macmillan.

Jevons, W. S. (1958). *The principles of science: A treatise on logic and scientific method* (2nd ed.). New York: Dover.

Johnston, M. D. (1987). *The spiritual logic of Ramon Llull.* Oxford, United Kingdom: Clarendon Press.

Kneale, W., & Kneale, M. (1962). *The development of logic.* Oxford, United Kingdom: Oxford University Press.

Large, A. (1985). *The artificial language movement.* New York: Basil Blackwell.

Lewis, R. (2007). *Language, mind and nature: Artificial languages in England from Bacon to Locke.* New York: Cambridge University Press.

Mays, M., & Henry, D. P. (1953). Jevons and logic. *Mind, 62*, 484.

Maziarz, E. A., & Greenwood, T. (1968). *Greek mathematical philosophy.* New York: Frederick Ungar.

McCorduck, P. (2004). *Machines who think: A personal inquiry into the history and prospects of artificial intelligence.* New York: AK Peters.

Newell, A., & Simon, H. A. (1976). Computer science as empirical inquiry: Symbols and search. *Communications of the Association for Computing Machinery, 19*, 113-126.

Perkins, F. (2004). *Leibniz and China: A commerce of light.* New York: Cambridge University Press.

Riedweg, C. (2005). *Pythagoras: His life, teaching, and influence.* New York: Cornell University Press.

Rossi, P. (2000). *Logic and the art of memory: The quest for a universal language* (S. Clucas, Trans.). Chicago: University of Chicago Press.

Rumelhart, D. E., McClelland, J. L., & PDP Research Group. (1986). *Parallel distributed processing: Explorations in the microstructure of cognition.* Cambridge, MA: MIT Press.

Scholem, G. (1960). *On the kabbalah and its symbolism.* New York: Schocken.

Shannon, C. E. (1938). A symbolic analysis of relay and switching circuits. *Transactions of the AIEE, 57*, 713-723.

Sowa, J. F. (1984). *Conceptual structures: Information processing in mind and machine.* Reading, MA: Addison-Wesley.

Styazhkin, N. I. (1969). *History of mathematical logic from Leibniz to Peano.* Cambridge, MA: MIT Press.

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

Vickers, B. (1987). *English science: Bacon to Newton.* Cambridge, United Kingdom: Cambridge University Press.

Yates, F. A. (1966). *The art of memory.* Chicago: University of Chicago Press.

KEY TERMS

Calculus: A calculus is a system of physical symbols and mechanical rules for their manipulation intended to accomplish some purpose, such as calculation, differentiation, integration, or formal inference. In principle, any process that can be accomplished by a calculus can be programmed on a digital computer.

Generate-and-Test Procedure: Is a common method of search, used in AI and other applications, in which possible solutions are generated systematically and evaluated until a suitable solution is found. For example, a game-playing program might generate possible moves, which are evaluated in terms of their likelihood of leading to a win. The greatest weakness of generate-and-test procedures is combinatorial explosion, which refers to the exponential increase of the number of possible solutions of increasing complexity (e.g., the number of moves that a game-playing program looks forward).

Knowledge Representation Language: Is a formal language, implementable in the data structures of a digital computer, intended to be capable of representing all knowledge or at least all knowledge in some AI application domain. It is intended as a medium for storing knowledge and for mechanized inference in its domain. A knowledge representation language is the analogue in AI of the language of thought in cognitive science.

Language of Thought ("Mentalese"): Is a hypothesized language-like system in whose terms all human cognition is supposed to take place. Advocates of this hypothesis acknowledge that not all of our thinking is discursive (by means of an inner dialogue), but they argue that the systematic structure of ideas and thinking implies that there must be a language of thought, albeit below the level of conscious access. The language-of-thought hypothesis partly justifies symbolic AI as a sufficient basis for AI.

Semantics: Refers to the meanings of expressions in a natural or artificial language and to the study of these meanings and their relation to the expressions. It is often contrasted with syntax. Since formal systems, calculi, and symbolic AI systems deal only with the forms of expressions, they can be sensitive to semantics only to the extent that the semantics is encoded in the system's syntax.

Symbolic AI: Is an approach to AI based on the manipulation of knowledge represented in language-like (symbolic) structures in which all relevant semantics (meaning) is explicit in the syntax (formal structure). The language-of-thought hypothesis provides part of the justification of the sufficiency of the symbolic approach to AI.

Syntax: Refers primarily to the grammar rules of a language (natural or artificial), that is, to the allowable forms of expressions without reference to their meaning (semantics). In the context of AI, syntax refers to the rules of knowledge representation in terms of data structures and to the computational processes that operate on these structures.