Symbolic versus Subsymbolic Computation and Representation

(Eliasmith, C. and W. Bechtel (2003). In Lynn Nadel (Ed.), *Encyclopedia of Cognitve Science*. MacMillian. Penultimate draft)

1. Historical Introduction

Explanations in cognitive science frequently take the form of computational models. Since the inception of cognitive science in the 1950s, there has been a competition between two approaches to computational modeling: the symbolic approach and the subsymbolic approach. These two approaches differ both in their conception of computation and in their conception of the representations that figure in computations. Not surprisingly, then, symbolic and subsymbolic explanations of cognitive behavior tend to be quite different.

Both approaches have a long pedigree. Indeed, both were advocated in talks on September 11, 1956 at the Symposium on Information Theory, which George Miller (1979) has identified as the birth event of cognitive science. Exemplifying the symbolic approach, Newell and Simon presented a paper describing the "Logic Theory Machine," while Rochester, Holland, Haibt, and Duda presented a subsymbolic implementation of Hebb's neuropsychological theory of cell assemblies.

In their prototypical instantiations, symbolic and subsymbolic computational models are clearly distinct. Symbolic models typically employ syntactically structured representations and logic-like rules that are sensitive to such structure. Thus, Logic Theorist adopted the language of symbolic logic, and relied on algorithms that were sensitive to the logical operators in those representations. Subsymbolic models, on the other hand, typically employ neuron-like units that excite or inhibit each other according to their current activation values. However, specifying in detail what is critical to a computational model counting as symbolic or subsymbolic turns out to be challenging. Before undertaking that task, it is helpful to show how the two approaches have developed historically.

1.1 The Historical Development of the Symbolic Tradition

The early history of the symbolic approach is one of repeated success. Not only did Newell and Simon's Logic Theorist serve as an early existence proof that computers could be employed to model processes thought to require intelligence in humans, but shortly thereafter they developed an even more sophisticated artificial intelligence program called General Problem Solver (GPS) that did more than just prove theorems. GPS was applied to a number of reasoning problems including the Tower of Hanoi problem and problems of cryptoarithmetic. A central goal of this research was to model human reasoning, and GPS was able to provide a detailed 'trace' of state transitions that matched very well to the verbal reports of a human working on the same problem (Newell and Simon 1976).

GPS and other early symbolic programs (several exemplars are presented in Minsky 1966) were applied to linguistically posed problems, but symbolicist success was also had in other domains, such as interacting with physical objects. Around 1970, Terry Winograd developed a program called SHRDLU that functioned in a simulated world of blocks. The program responds to natural language queries about what blocks are present and how they are arranged. As well, it can carry out simulated manipulations on the virtual blocks. Advocates claimed that the program seems to *understand* its limited domain: it carries out appropriate actions based on commands, and responds correctly to questions about its environment (for a challenge that it did not really exhibit understanding, see Dreyfus, 1972).

These successes (albeit in limited domains such as logic and block worlds) led a number of researchers to begin developing general frameworks for symbolic cognitive modeling. Two early examples that were developed over the course of several decades were John Anderson's ACT and ACT* (1983, 1990), and Allen Newell's SOAR (1990). Both have, at their heart, a *production system*. That is, a system of if-then (condition-production) rules that operate on syntactically structured symbolic representations that are stored in a working memory.

1.2 The physical symbol system hypothesis

The early successes and seemingly unlimited promise of the symbolic approach lead two of its early proponents, Newell and Simon (1976, p. 116), to propose what they called the Physical Symbol System Hypothesis:

A physical symbol system has the necessary and sufficient means for general intelligent action.

According to Newell and Simon, a physical symbol system is subject to the laws of physics and traffics in physical patterns. These patterns, or symbols, can be physically related so as to comprise expressions (i.e., symbol structures). The system itself is composed of such structures and a collection of processes that operate on these structures to produce other structures. With unlimited memory and unlimited time, physical symbol systems are capable of universal computation (Newell 1990, p. 76), just like the more famous Turing machine (Turing 1950).

The Physical Symbol System Hypothesis advances a very strong claim that (1) any intelligent system is a physical symbol system and 2) any physical symbol system of "sufficient size" can exhibit intelligence (Newell and Simon 1976, p. 116). Newell (1990 pp. 51, 76) also introduces the notion of a *knowledge system*. The symbols in such a system encode knowledge about that system's goals, actions, environment, and the relations between these items. Knowledge systems are governed by a single law: take actions to attain goals using all of the knowledge available to the system (ibid., p. 50). The relation between knowledge systems and symbol systems is clear: symbol systems are supposed to "realize knowledge systems by implementing representation laws so that the symbol structures encode the knowledge about the external world" (Newell, 1990, pp. 78-79). The conclusion for cognitive science is obvious: "humans are symbol systems that are at least modest approximations of knowledge systems" (Newell, 1990, p. 113). Notably, then, for the purposes of explaining cognition, a description in terms of a physical symbol system is a complete description and, also for these purposes, the fact that such a system might be implemented in a brain (i.e. a bunch of interconnected neurons) is irrelevant.

This view was widely accepted by many early cognitive scientists for a number of reasons. First, in computer science, universal computers had been proven to be able to compute all computable functions. If physical symbol systems are universal computers, and people don't compute non-computable functions, then clearly symbol systems have the computational resources necessary for exhibiting human intelligence. Second, in linguistics, Chomsky argued that using the grammars of natural languages required precisely that kind of computational power. Third, in psychology and philosophy, the Language of Thought hypothesis – the hypothesis that all thought is a language-like 'mentalese' and subject to a mental grammar – suggested that symbol processing was the basis of human cognition (Fodor 1975). Lastly, such a picture of human function is nicely consistent with our everyday folk psychology, which characterizes people's mental states as attitudes (e.g., believes, desires) toward propositions. If these very propositions are stored and operated on in symbolic models, then those models offer an explanation of why accounts of people's behavior in terms of beliefs and desires succeed much of the time.

1.3 The Historical Development of the Subsymbolic Approach

Even during its heyday, the symbolic approach was not the only game in town. One of the major early developers of the alternative, subsymbolic approach to understanding cognition was Frank Rosenblatt, the developer of a computational device called PERCEPTRON (Rosenblatt 1958). This device consists of a grid of photocells, randomly connected to associators that collect and process the electrical impulses produced when objects are placed in front of the photocells. The associator units are, in turn, connected to response units by weighted connections. The weights determine how much a particular associator unit influences a response unit. These weights are modifiable by a learning procedure that increases or decreases the weights from active associator units to a response unit so as to reduce any error on the response unit. In 1962, Rosenblatt proved the Perceptron Convergence Theorem, which states that his learning procedure will succeed in finding weights that permit the network to produce correct responses *if such weights exist*. In fact, the device was quite successful at recognizing various letters of the alphabet and provided a clear alternative to symbolic models. However, excitement over this less logic-like approach soon diminished, mostly as a result of Minsky and Papert's (1968) demonstration that there are classes of computations that models of this kind can't perform. Rosenblatt was well aware of these limitations. However, he also knew that this problem could be overcome by inserting additional layers of trainable weights. But, he failed to develop a generalized learning rule for finding these weights.

It wasn't until the mid 1980s that a learning rule—backpropogation—was developed that applied to multiple layers of connections. Around the same time, deep difficulties with the symbolic approach began to become apparent. These two developments resulted in the subsymbolic approach again attracting much attention in cognitive science. Now known by a variety of names, including *connectionism*, *artificial neural networks* (ANNs), and *parallel distributed processing* (PDP), the subsymbolic approach has enjoyed its own barrage of successes

(Rumelhart and McClelland 1986; Bechtel and Abrahamsen 1991). New and better models of language processing, object recognition, speech recognition, and motor control were proposed by connectionists. The stage was set for a conflict between computational paradigms.

2 The Clash Between the Symbolic and Subsymbolic Approaches

2.1 Clarifying the Differences Between the Symbolic and Subsymbolic Approaches

The differences between these two approaches can be grouped under two headings: representational differences and computational differences.

2.1.1 Representational differences

In symbolic models, the basic units of representation (i.e., symbols) are semantically interpretable, and are frequently modeled as words in natural languages. In contrast, in the subsymbolic approach, the basic representational units are below the level of semantic interpretation (hence, they are *sub*-symbols) (Smolensky 1988). Only patterns of activation *distributed* over large populations of network units are semantically interpretable (van Gelder, 199x). Moreover, any individual unit can be a constituent of many patterns, each of which has a different interpretation.

On occasion, the difference between symbolic and subsymbolic approaches is presented as a difference between continuous and discrete, or digital and analog representation. Paul Churchland (1995) has construed the difference in this way (p. 243):

So-called "digital" or *discrete-state* computing machines are limited by their nature to representing and computing mathematical functions that range over the *rational* numbers. . . .

> This is a potentially severe limitation on the abilities of a digital machine. . . . Therefore, functions over real numbers cannot strictly be computed or even represented within a digital machine. They can only be approximated.

In fact, subsymbolic models almost always employ digital representations. There is really nothing inherent in the 'digitalness' of these models that makes them only approximations of *actual* subsymbolic models. Furthermore, there is evidence that biological networks do not employ continuous representations (Eliasmith in press). Thus, it is misleading and inappropriate to draw the symbolic/subsymbolic distinction using the distinction between digital and analog representation.

2.1.2 Computational differences

Computational differences between symbolic and subsymbolic modeling can be understood as turning on the preferred metaphor for cognition. Symbolic theorists typically see cognizers as analogous to serial digital computers; there is a central processing unit (i.e., higher cognition), input systems (i.e., peripheral sensory systems), and output systems (i.e., motor systems)(see e.g., Newell 1990). Subsymbolic theorists, in contrast, see cognizers as inherently brain-like; massively parallel processors that rely on uniform functional units. Thus, subsymbolicists claim that they work with a "neurally inspired model of cognitive processes" (Rumelhart and McClelland 1986, p. 130) or with "brain-style computation" (Rumelhart 1989).

The serial/parallel distinction is symptomatic of a deeper distinction between the symbolic and subsymbolic approaches. It is symptomatic since we *could* translate symbolic models to run on parallel computers. Such a translation, however, is far from trivial since it requires decomposing a problem into independently solvable pieces and maintaining communication between parallel processors. However, there is a more basic difference: the kind of computation in a parallel instantiation of a symbolic program is not at all like that in a subsymbolic model. Computation in a symbolic system is governed by many different, explicitly encoded rules (as in a production system). In subsymbolic models, the computation is governed by one or a few mathematical rules that rely on numeric parameters to determine how activation of one unit affects the activation of other units. Although in simple models these parameters might be fixed by hand, as models become more complex researchers rely on learning rules to fix them (???not really, charlie and I always compute our weights directly???). In addition, the set of possible computations a subsymbolic model can realize is governed by the overall design of the system (e.g., what *sets* of nodes are connected to what other sets, and how densely). Generally, this system 'architecture' has been determined by the designer, but some subsymbolic models now rely on tools such as genetic algorithms to develop the pattern of connectivity in the model (???G.A.s have also been used to generate symbolic models???).

2.2 Subsymbolicist critiques of symbolic models

As noted, by the 1980s some cognitive scientists began to be frustrated by perceived limitations with the symbolic approach, limitations that the subsymbolic alternative seemed ideally poised to overcome. The first limitation stemmed from the very fact that symbolic representations are generally sentential. Initially, this seemed to be a strength since the problems that people solve are frequently and easily represented linguistically.

However, many basic cognitive tasks, such as the classification of sensory stimuli (e.g. taste, touch, smell, sound, and sight) are not presented linguistically. These tasks involve responding to statistical regularities in the environment, something symbolic representations are generally not sensitive to (Smolensky 1995). Subsymbolic representations, in contrast, are equally well suited to modeling representations in any modality and have been successfully applied to visual (Qian and Sejnowski 1988; Raeburn 1993), olfactory (Skarda and Freeman 1987), auditory

(Lazzaro and Mead 1989) and tactile problems. Moreover, Kosslyn (1980; 1994) and others have convincingly argued that a large class of reasoning problems seems to entail operations on images, which are a paradigmatically non-linguistic representation. Subsymbolic models are well suited to modeling this kind of perceptual processing.

In addition, symbolic systems tend to be brittle. Any degradation of a symbolic representation (e.g., a piece of computer code) can radically alter or destroy the whole system's functionality. Such brittleness is psychologically unrealistic. People often exhibit behavior that indicates partial retrieval of representations; "tip-of-the-brain" recall, prosopagnosia (loss of face recognition), and blindsight are all instances in which mental representations are incomplete. Although such incomplete representations reduce performance, they generally don't completely destroy it or radically alter it. Human performance tends to degrade gracefully as information is lost. Subsymbolic models, too, are not very brittle. Whereas minor damage to a symbolic conceptual network causes loss of entire concepts, damage to a subsymbolic conceptual network causes a loss in accuracy, just as occurs in human subjects (Churchland and Sejnowski 1992).

There are also difficulties in modeling learning in symbolic models. In general, learning in symbolic systems depends entirely on current symbols in the system. The most common example of this is 'chunking', in which many related symbols are grouped together and represented by another single symbol. This kind of learning can effectively capture some kinds of human learning, but is deficient for explaining how new symbols themselves are learned. Learning in subsymbolic models, in contrast, involves strengthening or weakening connections, which is well-suited to modeling low-level perceptual learning; learning that may explain development of new symbols.

Furthermore, symbolic systems suffer from their natural affinity to serial processing. Serial processing is quite compatible with the discrete, non-statistical character of symbolic representations. If the relations between elements are irrelevant to their effect on cognition, then processing them one at a time makes perfect sense. However, if the relations between elements are important for determining how they should/can be processed (i.e., if holistic considerations are relevant) then parallel processing is in order. Many cognitive problems involve satisfying soft constraints (constraints that ideally would all be satisfied but some may have to be violated to arrive at the best solution). For example, generating analogies depends on the ability to match a current situation to any other possible situation we have encountered before, where no previous situation is a perfect match.

In general, the representational and computational commitments of the subsymbolic approach have been conducive to avoiding many of the difficulties with the symbolic approach. Holistic processing, soft-constraint satisfaction, statistical sensitivity and graceful degradation are all more successfully incorporated into the subsymbolic approach.

2.3 Symbolicist critiques of subsymbolic processing

Many symbolicists counter that, whatever the shortcomings of the symbolic approach, the cost of the subsymbolic remedy is too great. Fodor and Pylyshyn (1988) constitute the *locus classicus* of the critique, arguing that the failure to employ structured representations and structure-sensitive operations leaves subsymbolic models unable to explain truly *cognitive* phenomena.

Fodor and Pylyshyn focus on two features of cognition subsymbolic models cannot explain: productivity and systematicity. Productivity is the capacity to produce arbitrarily many expressions. This is achieved in symbol systems by means of recursion (e.g. 'John told Sue who told Mary who told Bob…'). Clearly a physical symbol system can construct just these sorts of structures since the rules used to compose representations employ (possibly recursive) processes defined over atomic elements (symbols). Fodor and Pylyshyn contend that by eschewing compositional rules and not recognizing the importance of syntactic structure, subsymbolic approaches fail to be productive.

Systematicity concerns the relations that exist between representations. Fodor and Pylyshyn note that anyone who has mastered English must admit *all* of the following as sentences: "John loves the girl", "the girl loves the girl", "John loves John", and "the girl loves John" (ibid., p. 38). They claim that in order to explain the systematicity found both in language and thought, symbolicists can appeal to syntactic structure. Structured schemas such as 'noun-phrase transitive-verb noun-phrase' can be generalized to *any* noun-phrase and transitive-verb. Symbolic models routinely employ such structured representations and so can readily explain systematicity. However, since subsymbolic models do not employ this kind of representation, Fodor and Pylyshyn argue that these models lack the resources needed to explain systematicity.

In characterizing the representational structures that can explain the systematicity and productivity of cognition, Fodor and Pylyshyn rely on the notion of a *compositional* syntax and semantics. They thereby emphasize that the particular *syntactic* rules that are employed are appropriate for ensuring *semantic* coherence. A compositional representation is one in which meaning is an additive function of the meaning of its components. Thus, in a sentence like "The ball is red," the meaning of the structure can be derived from the meaning of each of the component lexical items (words). Furthermore, each lexical item makes the same semantic contribution to each expression in which it occurs. This, claims the symbolicist, is typical of both language and thought. This 'compositionality' is a problem for subsymbolic models because population activation vectors, the semantic units for subsymbolic models, have no particular structure. It isn't clear how an activation vector that represents a sentence would be related to an activation vector that represents each of the words in the sentence. It isn't clear because the sentence vector isn't built up out of a *linear* combination of word vectors.

Subsymbolic models, then, will have a hard time accounting for some basic aspects of human cognition. All of the failings of the subsymbolic approach derive from not recognizing the importance of structured representations.

2.4 State of the art subsymbolic models – A response to symbolicists

Symbolic theorists have identified potential serious shortcomings of the subsymbolic approach. Subsymbolic theorists cannot solve these problems by merely implementing a physical symbol system in a connectionist architecture. Such implementation is in principle possible since both kinds of computational systems are universal computational systems (i.e. both are equivalent to a universal Turing machine). However, for a subsymbolic model *merely* to implement a symbolic one is for the subsymbolicist to admit that *sub*-symbols are themselves inadequate to model cognition. Indeed, such implementation amounts to admitting that nothing is gained by turning to sub-symbols for a new and different understanding of cognition.

While not opting for mere implementation, some theorists do think that adequately answering the symbolicist's challenge does require some appeal to symbol systems. They thus advocate hybrid systems in which symbolic structures are implemented in subsymbolic structures in such a manner that the subsymbolic implementation plays a cognitive role. The goal in such hybrid systems is for the subsymbolic implementation to provide statistically sensitive processing, while allowing the implemented symbol system to provide structural sensitive processing. Some researchers have developed interesting hybrid models (Sun, other refs. ???Barnden, Hendler), but face problems in developing a principled means of determining and justifying a desirable combination of structure-sensitive and statistically-sensitive processes.

Another, more purely subsymbolic approach, is to attempt to develop a significant measure of structure sensitivity directly in subsymbolic models. Two approaches have exhibited some successes.

Pollack (1990) developed Recurrent Auto-Associative Memory (RAAM) networks for recursively creating compressed vectors of input vectors. If linguistic strings (structured representations) are employed as inputs, these networks can created representations that are not explicitly compositional, but from which the compositional structure of the input can be recovered. Moreover, other structure sensitive operations (e.g., using a network to transform compressed representations of active sentences into compressed representations of passive sentences) can be performed directly on the compressed strings without first extracting the compositional structure. Since the compressed representations are not themselves syntactically structured but seem to perform tasks that require information about such structure, they are construed as functionally compositional (van Gelder, 199x).

Another approach for accounting for productivity and systematicity in subsymbolic models employs distributed representation schemes based on vector products (Smolensky 1990; Plate 1994). Essentially, these schemes include operations for combining and extracting vectors. Two vectors, *A* and *B*, can be combined to produce a third vector *C*. From C either *A* or *B* can subsequently be extracted, given the other. This provides a way of building up structures in vector representations. Because of the generality of the defined operations, problems of systematicity no longer arise for this representational scheme. As well, productivity is solved because these operations can be recursively applied.

In fact, these kinds of schemes have been successfully used to model high-level cognitive function. For example, Eliasmith and Thagard (in press) have proposed a model of analogy based on Plate's representational scheme. Analogical mapping is notoriously structurally sensitive and a typically *cognitive* task (Gentner and Toupin 1986; Holyoak and Thagard 1995). This kind of model is thus an existence proof of the ability of the subsymbolic approach to handle structure sensitivity.

However, these subsymbolic attempts to account for productivity and systematicity do not provide for truly *unlimited* realizations of either of these features of cognition. For example, as longer strings are recursively supplied to a RAAM, errors result. Thus, these subsymbolic approaches only provide a degree of productivity. The reason for this is clear in the vector product approaches that restrict the size of the vectors. The product vector, C, isn't merely an additive combination of vectors A and B. Rather, A and B are encoded into C and, in Plate's scheme for example, A and B are only partly encoded into C. So, the meaning of C isn't a straightforward function of the meaning of its component parts.

Does this mean that subsymbolicists fail to meet the symbolicist's challenge? That depends on the precise nature of compositional human mental representations, and how productive and systematic human thought is. The suggestion that mental representations might be less compositional than symbolicists contend comes from examples in natural languages. Such languages are not nearly as compositional as symbolicists would like to believe. The meaning of most colloquial expressions (e.g., "it's raining cats and dogs", "what a couch potato", or "break a leg") clearly isn't an additive function of the meaning of their components. Meanings are so flexible that, for example, the words 'unravel' and 'ravel' mean the same thing despite opposite 'compositional' meanings. Given these sorts of examples, it is not so clear that lack of compositionality is devastating to the subsymbolic approach. Since compositionality is intended to explain productivity and systematicity, and since subsymbolic models only provide a measure of these properties, subsymbolic theorists thus try to refocus the debate onto such issues as to how much productivity and systematicity is needed to model human cognitive abilities.

3 Summary

Both the language-like symbolic approach, and the neurally inspired subsymbolic approach to cognitive science have had a significant success at explaining certain aspects of human cognition. Neither approach, however, obviously explains more aspects than the other. Although recent advances in the subsymbolic tradition have made headway into symbolicist domains, it is not yet clear if these advances overcome, or merely limit, the problems of systematicity and productivity.

4 References

Anderson, J. R. (1983). The architecture of cognition. Cambridge, MA, Harvard University Press.

Bechtel, W. and Abrahamsen, A. (1991). Connectionism and the mind: an introduction to parallel processing in networks. Cambridge, MA, Basil Blackwell.

Bechtel, W. (1994). "Natural deduction in connectionist systems." Synthese 101: 433-463.

Churchland, P. (1995). The engine of reason, the seat of the soul: a philosophical journey into the brain. Cambridge, MA, MIT Press.

Dreyfus, H. L. (1972). What computers can't do: A critique of artificial reason. New York, Harper and Row.

Eliasmith, C. (in press). “Is the brain analog or digital?: The solution and its consequences for cognitive science.” Cognitive Science Quarterly.

Eliasmith, C. and P. Thagard (in press). “Integrating structure and meaning: A distributed model of analogical mapping.” Cognitive Science.

Fodor, J. (1975). The language of thought. New York, Crowell.

Fodor, J. and Z. Pylyshyn (1988). “Connectionism and cognitive architecture: A critical analysis.” Cognition 28: 3-71.

Gardner, H. (1985). The mind's new science. New York, Basic Books.

Gentner, D. and C. Toupin (1986). “Systematicity and surface similarity in the development of analogy.” Cognitive Science 10: 277-300.

Holyoak, K. and P. Thagard (1995). Mental leaps: Analogy in creative thought. Cambridge, MA, MIT Press.

Kosslyn, S. (1980). Image and mind. Cambridge, Harvard University Press.

Kosslyn, S. (1994). Image and brain: The resolution of the imagery debate. Cambridge, MA, The MIT Press.

Le Cun, Y. and J. S. Denker (1993). “Natural versus universal probability, complexity, and entropy.” The Physics of Computation.

McClelland, J. L. and D. E. Rumelhart, Eds. (1986). Parallel distributed processing: Explorations in the microstructure of cognition. Cambridge MA, MIT Press/Bradford Books.

Minsky, M. and S. Papert (1968). Perceptrons. Cambridge, MA, MIT Press.

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

Newell, A. and H. Simon (1976). “Computer science as empirical enquiry: symbols and search.” Communications of the Association for Computing Machinery 19: 113-126.

Newell, A. and H. A. Simon (1976). GPS, a program that simulates human thought. Computers and thought. E. Feigenbaum and J. Feldman. New York, NY, McGraw-Hill.

Plate, T. A. (1994). Distributed representations and nested compositional structure. Computer Science. Toronto, University of Toronto.

Pollack, J. (1990). “Recursive distributed representations.” Artificial Intelligence 46: 77-105.

Pylyshyn, Z. (1973). “What the mind's eye tells the mind's brain: A critique of mental imagery.” Psychological Bulletin 80: 1-24.

Rosenblatt, F. (1958). “The perceptron: A probabilistic model for information storage and organization in the brain.” Psychological Review 65: 386-408.

Rumelhart, D. E. (1989). The architecture of mind: A connectionist approach. Foundations of cognitive science. M. I. Poser. Cambridge, MA, MIT Press.

Rumelhart, D. E. and J. L. McClelland, Eds. (1986). Parallel distributed processing: Explorations in the microstructure of cognition. Cambridge MA, MIT Press/Bradford Books.

Smolensky, P. (1988). “On the proper treatment of connectionism.” Behavioral and Brain Sciences 11(1): 1-23.

Smolensky, P. (1990). “Tensor product variable binding and the representation of symbolic structures in connectionist systems.” Artificial Intelligence 46: 159-217.

Smolensky, P. (1995). Computational models of mind. A companion to the philosophy of mind. S. Guttenplan. Cambridge, MA, Blackwell.

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