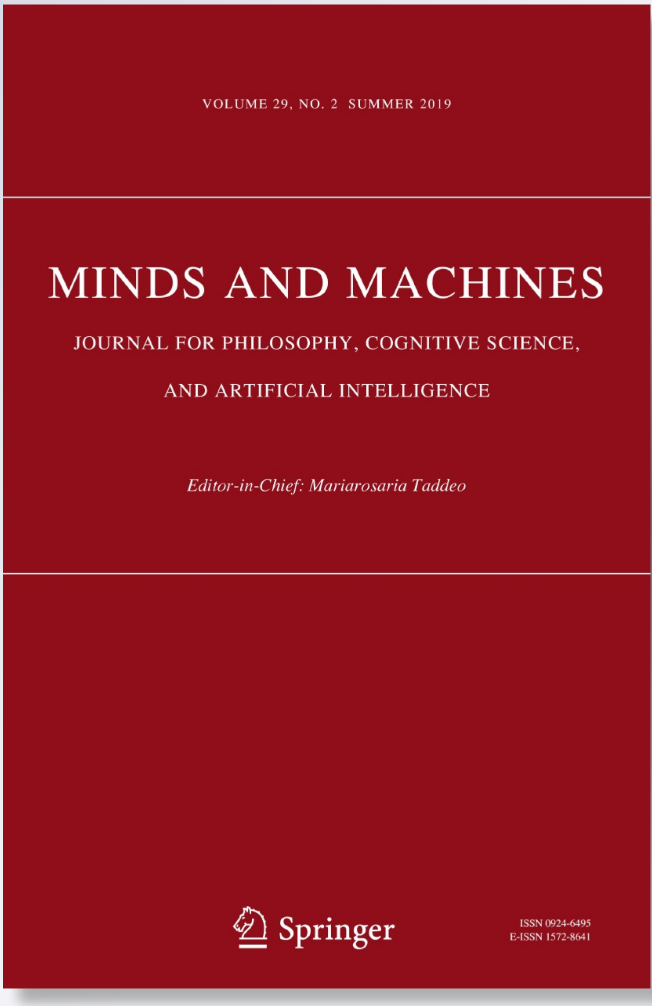
Computers Are Syntax All the Way Down: Reply to Bozşahin

William J. Rapaport

Minds and Machines
Journal for Artificial Intelligence, Philosophy and Cognitive Science

ISSN 0924-6495
Volume 29
Number 2

Minds & Machines (2019) 29:227-237
DOI 10.1007/s11023-018-9485-2



Springer

Minds and Machines (2019) 29:227–237
https://doi.org/10.1007/s11023-018-9485-2

Computers *Are* Syntax All the Way Down: Reply to Bozşahin

William J. Rapaport[1]

Received: 29 October 2018 / Accepted: 12 December 2018 / Published online: 17 December 2018

Abstract

A response to a recent critique by Cem Bozşahin of the theory of syntactic semantics as it applies to Helen Keller, and some applications of the theory to the philosophy of computer science.

Keywords Chinese Room Argument · Helen Keller · Philosophy of computer science · Semantics · Syntactic semantics · Syntax

1 Introduction

In "How Helen Keller Used Syntactic Semantics to Escape from a Chinese Room", I argued that "A computer can come to understand natural language the same way Helen Keller did: by using 'syntactic semantics'—a theory of how syntax can suffice for semantics, i.e., how semantics for natural language can be provided by means of computational symbol manipulation" (Rapaport 2006, p. 381; see also Rapaport 2011). By contrast, in "Computers Aren't Syntax All the Way Down or Content All the Way Up", Cem Bozşahin (2018, p. 544) argues "that computers cannot be syntactic machines all the way down. They have to have non-syntactic primitives ... to be able to carry out their syntactic processing." The present essay is a response to Bozşahin's critique of my earlier essay, with some applications to the philosophy of computer science.

William J. Rapaport
rapaport@buffalo.edu
http://www.cse.buffalo.edu/~rapaport/

[1] Departments of Computer Science and Engineering, Philosophy and Linguistics, Center for Cognitive Science, University at Buffalo, The State University of New York, Buffalo, NY 14260-2500, USA

2 Syntactic Semantics

Briefly, the theory of syntactic semantics consists of three principles (Rapaport 2018, §3):[1]

Internalism:	Cognitive agents have direct access only to internal representatives of external objects.
Syntacticism:	It follows that words, their meanings, and semantic relations between them are all syntactic.
Recursive Understanding:	Understanding is recursive: We understand one kind of thing semantically in terms of another that is already understood; the base case is to understand something in terms of itself, which is syntactic understanding.

Why does Syntacticism follow from Internalism? Let SYN be a non-empty set whose members have various (internal) properties and stand in various relations with each other. These properties and relations constitute the *syntax* of SYN. (SYN might be a formal system; in that case, its proof theory—its axioms and rules of inference—are part of its syntax.) To provide a *semantic* interpretation of SYN, we need *another* set, SEM, each of whose members, typically, will be a "meaning" of a corresponding member of SYN (and the syntax of SEM is its "ontology"). To provide the semantics of SYN in terms of SEM, we need relations *between* the members of SYN and the members of SEM. These semantic-interpretation relations, however, are neither part of SYN nor part of SEM; they "are external to both" (Rapaport 2017b, p. 7). Hence, in order to talk about these relations, we must consider a new set, $U = SYN \cup SEM$. The syntax of U includes: (1) all of the properties of, and relations among, the members of SYN, (2) all of the properties of, and relations among, the members of SEM, *and* (3) the semantic relations *between* the members of SYN and the members of SEM. Thus, U's *syntax* enables us to talk about SYN's *semantics* (its semantics in terms of SEM). That is why semantics is syntactic.

3 Helen Keller

In Rapaport (2006), I focused on Keller's "well-house episode", in which her teacher, Anne Sullivan, finger-spelled 'w-a-t-e-r' in one of Keller's hands while water poured over her other hand. As Keller (1905, p. 36) tells it, "the mystery of language was revealed to me. I knew then that 'w-a-t-e-r' meant the wonderful cool something that was flowing over my hand."

In commenting on Rapaport (2006), Bozşahin (2018, p. 563) writes:

[1] See also Rapaport (1986), Rapaport (1988), Rapaport (1995), Rapaport (2000), Rapaport (2012) and Rapaport (2017b).

> How did Keller know that what Sullivan does to one hand when finger-spelling what happened in the other one is semantics, but not syntax?

Suppose that Sullivan poured water over Keller's left hand while finger-spelling 'w-a-t-e-r' in her right hand (Keller 1905; Rapaport 2006, p. 396). Bozşahin asks how Keller knew that Sullivan's water in her left hand was semantics, not syntax—presumably, a semantic interpretation of the syntax located in her right hand.

His question has a false presupposition: The water was *not* semantics; it *was* syntax! Actually, it would be better to put it this way: It was *both* semantics *and* syntax; indeed, it was syntactic semantics. Why? It was syntactic in the same way that the finger-spelled word 'w-a-t-e-r' was syntactic: Both were sensed by Keller in precisely the same way; both were represented in her brain by neuron firings that form the syntax of her "language of thought". But it was semantic in the sense that it played the role of semantic interpretation of what was being finger-spelled in her right hand.

Immediately following the preceding passage, Bozşahin writes:

> If she really were in a Chinese room, she could not have known that. This is because the conditions of the Chinese room experiment ... [are] such that only formal symbols are allowed to enter the room. (p. 563)

That is, if Keller had really been in a Chinese room, she could not have known that what was happening in her left hand was semantics and that what was happening in her right hand was syntax. Absolutely correct! For her, they are *both* syntax: As Bozşahin says, "only formal symbols are allowed to enter the room". The difference is the role that each plays in her syntactic system. No water—no "wonderful cool" stuff—entered her "Chinese room"; only neuron firings *representing* water entered the "room", as did neuron firings *representing* the finger-spelled 'w-a-t-e-r'. The neuron firings are the formal symbols in this case.

Here is another way to put it: The word *spelled* into her right hand was a member of SYN. The water *spilled* onto her left hand was a member of SEM. More precisely, the neuron firings representing the *word* were a member of SYN; the neuron firings representing the *water* were a member of SEM. But Keller's neural representations of both were members of U = SYN ∪ SEM.

4 Causal Links

Bozşahin continues:

> If Rapaport's argument is that she was once in the Chinese Room, then Keller could only take formal symbols inside, even if we follow his assumption that *being in a physical world is a causal link.* (p. 563, my italics)

Here, Bozşahin is referring to an earlier comment he makes about my presentation of syntactic semantics in Rapaport (1988):

> Rapaport (1988): 84 does discuss a causal link that is necessary in addition to syntax, to give rise to semantics. This link is assumed to be a type of non-syntactic semantics that links the agent to the external world by being in it, but according to him it is "not the kind of semantics that is of computational interest." (p. 562)

Here is what I said in Rapaport (1988):

> [M]y thesis is that *syntax suffices*. I shall qualify this somewhat by allowing that there will also be a certain causal link between the computer and the external world, which contributes to a *certain kind* of nonsyntactic semantics, but not the kind of semantics that is of computational interest. What kind of causal link is this? Well, obviously, if someone built the computer, there's a causal link between it and the external world. But the particular causal link that is semantically relevant is one between the external world and what I shall call the computer's "mind"—more precisely, the "mind" of the process produced by the running of the natural-language-understanding program on the computer. ... So, my thesis is that (suitable) purely syntactic symbol-manipulation of the system's knowledge base (its "mind") suffices for it to understand natural language. Although there is also a causal link between its "mind" and the external world, I do not think that this link is necessary *for understanding natural language*. ... [M]y reasons for taking this position are roughly methodologically solipsistic: the system has no access to these links, and a second system conversing with the first only has access to its own internal representations of the first system's links. Nevertheless, given that there are in fact such links, what might they be like? ... [T]hey are perceptual links .... (Rapaport 1988, pp. 84–86)

Adapted to our present example, what I was referring to there were the facts that it was real water that Sullivan poured over Keller's left hand, and that *the neuron firings representing that water were physically caused by it*. Thus, if one wanted to give a semantic interpretation *of those neuron firings* in terms of something in the *external* world, it would be the water. *But that water has nothing of present interest to do with the relation between the two sets of neuron firings*. This is the sense in which "it is 'not the kind of semantics that is of computational interest'." It *is* interesting, of course, but it has no bearing on Keller's syntactic-semantic interpretation of 'w-a-t-e-r'. Again, what "is of no computational interest"—*for our present purposes*—is *where* the actual water came from or *how* it got transduced into certain neuron firings. The only thing that *is* of computational interest is the neuron firings (this is the point of Fodor's 1980 methodological solipsism).

For Keller, the actual water in her left hand and the word 'w-a-t-e-r' in her right hand are *both* formal symbols for her. But they are of different types in the sense that they play different roles for her. One of the fundamental ideas of syntactic semantics is that (1) the set of entities that play the semantic role is (in general) distinct from the set of entities that play the syntactic role; (2) semantic interpretation is a relation between these two sets of entities; (3) but, when the two sets are unioned (as they are in Keller's case—there is only one set of neuron firings), those semantic interpretation relations become part of the *syntax* of the unioned set.

5 Content as Form

According to Bozşahin,

> If we follow Rapaport's escape explanation, then Keller would have to invent generating meanings internally. She can do that if she already knows where to start. (p. 563)

She starts with antecedently understood *things*—water, in this case. But how does she understand such things? According to the principle of recursive understanding, there are two ways to understand something: Something can be understood semantically by understanding it in terms of something *else* that is antecedently understood. But how is that other thing understood? Either (a) in terms of yet another antecedently understood thing (a recursive chain or "correspondence continuum" (Smith 1987) of semantic understandings), or—to avoid an infinite regress or a circular understanding—(b) it must be understood syntactically, that is, understood in terms of itself (the base case) (Rapaport 1986, 1995).[2] More precisely, a *system* is syntactically understood in terms of the system *itself*; a *member* (e.g., a word) of that system is syntactically understood in terms of its syntactic relations to the *rest* of the system—i.e., holistically[3] (Rapaport 2002).

So, Keller understood water because of her experiences with it. She later came to understand 'w-a-t-e-r' in terms of water. More accurately, she came to understand 'w-a-t-e-r' *as experienced by her via certain neuron firings* in terms of water *as experienced by her via other neuron firings*. Because both are experienced as neuron firings in a single brain, they are both part of the same "language of thought"—they are both syntactic.

Bozşhin says that the

> ... Chinese Room's conditions are very unlikely to be part of the truth about relating content with form. (p. 563)

Part of my theory of syntactic semantics is that content is *not* different from form. They are both represented internally via neuron firings, some of which play the role of form ('w-a-t-e-r') and some of which play the role of content (water).

Bozşahin concludes:

> What avoids infinite regress here is the intensional consideration that it can not be turtles all the way down; so, those meanings will be interpreted by the

[2] I have a certain fondness for circular understanding, i.e., holistic understanding, which is itself a form of syntactic understanding (Rapaport 2002). Perhaps to be understood in terms of itself is circular understanding with zero radius, so to speak.

[3] Circularly, if you prefer.

> virtue of syntax taking them to ground cases where no further syntactic translation takes place. (pp. 563–564)

But it *is* "turtles"—that is, syntax—"all the way down". The "ground cases" are simply those parts of the syntax that can only be understood holistically in terms of the rest of the syntactic system. Bozşahin (and many others, of course, most notably Harnad 1990) considers (syntactic) form to be grounded ultimately in (semantic) content. But this is not necessarily the case! For Keller, water was content (certain neuron firings representing water played the role of content) and 'w-a-t-e-r' was form (certain other neuron firings representing 'w-a-t-e-r' played the role of form). But the relation can be reversed; "content" can be grounded in "form": For me, growing up in New York City and never experiencing a real rabbit till much later in life, the (syntactic) *word* 'rabbit' and (syntactic) *pictures* of rabbits were my reality—my "content". (Perhas I was like Jackson's (1986) Mary with respect to rabbits!)

6 Contextual Vocabulary Acquisition

Syntactic semantics also underlies our ability to figure out a meaning for an unknown word from context, where "context" includes, not only the word's relations to other words in its surrounding *text* (SYN), but also its relations to other words (or concepts) in the reader's prior knowledge (SEM). For example, from the following text containing the unknown (to the reader) word 'brachet' (slightly paraphrased from Malory 1470, pp. 66, 72; my boldface):

> There came a white hart running into the hall with a white **brachet** next to him, and thirty couples of black hounds came running after them. As the hart went by the sideboard, the white **brachet** hit him in the buttock. The knight arose, took up the **brachet** and rode away with the **brachet**. A lady came in and cried aloud to King Arthur, "Sire, the **brachet** is mine." There was the white **brachet** which bayed at him fast.

and the following (reasonable) prior knowledge:

> Only physical objects have color.
> Only animals bite.
> Only small things can be picked up and carried.
> Only valuable things are wanted.
> Hounds are hunting dogs.
> Only hounds bay.

a contextual-vocabulary-acquisition computer system programmed using the SNePS knowledge-representation and reasoning system (Shapiro and Rapaport 1987, 1992, 1995) came to understand 'brachet' as meaning

> "a small, white, valuable dog (a hound) that can bite, bay, and hunt".

(That is an English paraphrase of the actual output; see Ehrlich 1995; Rapaport 2003, 2005; Rapaport and Kibby 2002, 2007, 2010, 2014 for details. The *Oxford English Dictionary* definition, by comparison, is: "A kind of hound which hunts by scent".)

7 Are Computer Programs Intentional?

The idea that computer "understanding"—and computer processing, more generally—is syntactic "all the way down" is related to another issue in the philosophy of computer science:

- Is the proper form of a computer program a completely syntactic (or "narrow") algorithm of the form

  Do *A*

  where *A* is either a primitive computation, or else a set of computations recursively structured by sequence, selection, and repetition?
- Or (as, e.g., Hill 2016 would have it) is the proper form of a computer program a semantic ("wide") algorithm of the form

  In order to accomplish goal *G*, do *A*

  where *G* is an intentional or teleological description of what *A* is intended to accomplish?

Goal *G* and algorithm *A* are separable: There are examples of computer programs in which an algorithm *A* can "succeed" although the program "fails" to accomplish its goal *G*. And there are examples of computer programs in which a single algorithm *A* can accomplish distinct goals $G_1 \neq G_2$. For example, Cleland (1993) discusses a hollandaise-sauce recipe that, when executed "correctly" on, say, the Moon, fails there to produce the emulsion necessary for the mixture to be hollandaise sauce. And Fodor (1978) discusses two computers, one of which uses a certain algorithm to play chess and the other of which uses the *same* algorithm to simulate the Six Day War. (I survey these and others in Rapaport 2017a and Rapaport 2019, Ch. 17.) Bozşahin's "content" can best be understood as focusing on *G*. But computers are purely syntactic devices, concerned only with *A*.

This distinction also suggests a way to respond to Searle's (1990) wall that allegedly executes the Wordstar program. If the program has the form "To execute (or use) Wordstar, do *A*", where *A* is the underlying, purely syntactic algorithm, then it may be possible for the molecular motion in Searle's wall to be "doing *A*" even though the wall is not executing—or usable as—Wordstar.

Indeed, the wall is *not* usable as Wordstar, because there is no facility for user input. The wall might be a Turing machine executing *A*, but it is not interactive, and so cannot accomplish the goal of running Wordstar. At best, the wall is like an implementation of Wordstar that has been "opened" on a computer like a Mac Mini that has not been connected to either a keyboard or a monitor, and so just sits there waiting, Godot-like, for user input.

8 Turing's Strange Inversion

There is one more consideration in favor of computers being syntax all the way down—what Daniel Dennett has called "Turing's strange inversion of reasoning":

> The Pre-Turing world was one in which computers were people, who had to understand mathematics in order to do their jobs. Turing realised that this was just not necessary: you could take the tasks they performed and squeeze out the last tiny smidgens of understanding, leaving nothing but brute, mechanical actions. IN ORDER TO BE A PERFECT AND BEAUTIFUL COMPUTING MACHINE IT IS NOT REQUISITE TO KNOW WHAT ARITHMETIC IS. —Dennett (2013, p. 570, capitalization in original)[4]

This can be illustrated by an experience I once had:

> My wife recently opened a restaurant and asked me to handle the paperwork and banking that needs to be done in the morning before opening (based on the previous day's activities). She wrote out a detailed set of instructions, and one morning I went in with her to see if I could follow them, with her looking over my shoulder. As might be expected, there were gaps in her instructions, so even though they were detailed, they needed even more detail. Part of the reason for this was that she knew what had to be done, how to do it, and why it had to be done, but I didn't. This actually disturbed me, because I tend to think that algorithms should really be just "Do A," not 'To G, do A." Yet I felt that I needed to understand G in order to figure out how to do A. But I think the reason for that was simply that she hadn't given me an algorithm, but a sketch of one, and, in order for me to fill in the gaps, knowing why I was doing A would help me fill in those gaps. But I firmly believe that if it made practical sense to fill in all those gaps (as it would if we were writing a computer program), then I wouldn't have to ask why I was doing it. No "intelligence" should be needed for this task if the instructions were a full-fledged algorithm. If a procedure (a sequence of instructions, including vague ones like recipes) is not an algorithm (a procedure that is fully specified down to the last detail), then it can require "intelligence" to carry it out (to be able to fill in the gaps, based, perhaps on knowing why things are being done). If intelligence is not available (i.e., if the executor lacks relevant knowledge about the goal of the procedure), then the procedure had better be a full-fledged algorithm. There is a difference between a human trying to follow instructions and a machine that is designed to execute an algorithm. The machine cannot ask why, so its algorithm has to be completely detailed. But a computer (or a robot, because one of the tasks is going

[4] See also the more easily accessible Dennett (2009), p. 10061.

> to the bank and talking to a teller!) that could really do the job would almost certainly be considered to be "intelligent."
> —Rapaport, quoted in Hill and Rapaport (2018, p. 35)

There are various methods in which *G* can be "internalized" into *A* to produce a purely syntactic computer program that we might think of as a "union" of the syntactic *A* and the semantic *G*. (These are discussed in Rapaport 2017a, 2019.) But far from making the computer program "intentional" or "semantic" in a wide or "external" sense, such internalization is precisely how it gets a "narrow", internal[5] syntactic semantics. And this is why it becomes purely syntactic—all the way down.

References

Bozşahin, C. (2018). Computers aren't syntax all the waydown or content all the way up. *Minds and Machines*, *28*(3), 543–567.

Cleland, C. E. (1993). Is the Church-Turing thesis true? *Minds and Machines*, *3*(3), 283–312.

Dennett, D. C. (2009). Darwin's 'strange inversion of reasoning'. *Proceedings of the National Academy of Science, 106*(suppl. 1), 10061–10065. http://www.pnas.org/cgi/doi/10.1073/pnas.0904433106 (see also Dennett 2013).

Dennett, D. C. (2013). Turing's 'strange inversion of reasoning'. In S. B. Cooper & J. van Leeuwen (Eds.), *Alan Turing: His work and impact* (pp. 569–573). Amsterdam: Elsevier (see also Dennett 2009).

Ehrlich, K. (1995). *Automatic vocabulary expansion through narrative context*. Technical report 95-09, SUNY Buffalo Dept. of Computer Science, Buffalo, NY.

Fodor, J. A. (1978). Tom Swift and his procedural grandmother. *Cognition 6*, 229–247. http://www.nyu.edu/gsas/dept/philo/courses/mindsandmachines/Papers/tomswift.pdf.

Fodor, J. A. (1980). Methodological solipsism considered as a research strategy in cognitive psychology. *Behavioral and Brain Sciences*, *3*(1), 63–109.

Harnad, S. (1990). The symbol grounding problem. *Physica D*, *42*, 335–346.

Hill, R. K. (2016). What an algorithm is. *Philosophy and Technology*, *29*, 35–59.

Hill, R. K., & Rapaport, W. J. (2018). Exploring the territory: The logicist way and other paths into the philosophy of computer science. *American Philosophical Association Newsletter on Philosophy and Computers, 18*(1), 34–37. https://cdn.ymaws.com/www.apaonline.org/resource/collection/EADE8D52-8D02-4136-9A2A-729368501E43/ComputersV18n1.pdf.

Jackson, F. (1986). What Mary didn't know. *Journal of Philosophy*, *83*, 291–295.

Keller, H. (1905). *The story of my life* (p. 1954). Garden City, NY: Doubleday.

Malory, S. T. (1470). *Le Morte Darthur*. New York: Collier Books, 1982. Ed. and trans. by R. M. Lumiansky.

Moor, J. H. (Ed.). (2003). *The Turing test: The elusive standard of artificial intelligence*. Dordrecht: Kluwer Academic.

Rapaport, W. J. (1986). Searle's experiments with thought. *Philosophy of Science, 53*, 271–279. http://www.cse.buffalo.edu/~rapaport/Papers/philsci.pdf.

Rapaport, W. J. (1988). Syntactic semantics: Foundations of computational natural-language understanding. In J. H. Fetzer (Ed.), *Aspects of artificial intelligence* (pp. 81–131). Dordrecht, The Netherlands: Kluwer Academic Publishers. http://www.cse.buffalo.edu/~rapaport/Papers/synsem.pdf; reprinted with numerous errors in Eric Dietrich (ed.) (1994), *Thinking machines and virtual persons: Essays on the intentionality of machines* (San Diego: Academic Press) (pp. 225–273).

Rapaport, W. J. (1995). Understanding understanding: Syntactic semantics and computational cognition. In J. E. Tomberlin (Ed.), *AI, connectionism, and philosophical psychology* (Philosophical perspectives, Vol. 9, pp. 49–88). Atascadero, CA: Ridgeview. http://www.cse.buffalo.edu/~rapaport/Papers/

[5] Or "indigenous" (Rescorla 2012, 2014).

rapaport95-uu.pdf. Reprinted in Toribio, Josefa, & Clark, Andy (eds.) (1998), *Language and meaning in cognitive science: Cognitive issues and semantic theory (Artificial intelligence and cognitive science: Conceptual issues*, Vol. 4) (New York: Garland).

Rapaport, W. J. (2000). How to pass a Turing test: Syntactic semantics, natural-language understanding, and first-person cognition. *Journal of Logic, Language, and Information, 9*(4), 467–490. http://www.cse.buffalo.edu/~rapaport/Papers/TURING.pdf. *Reprinted in Moor*, *2003*, 161–184.

Rapaport, W. J. (2002). Holism, conceptual-role semantics, and syntactic semantics. *Minds and Machines, 12*(1), 3–59. http://www.cse.buffalo.edu/~rapaport/Papers/crs.pdf.

Rapaport, W. J. (2003). What is the 'context' for contextual vocabulary acquisition? In P. P. Slezak (Ed.), *Proceedings of the 4th international conference on cognitive science/7th Australasian Society for cognitive science conference (ICCS/ASCS-2003; Sydney, Australia)* (Vol. 2, pp. 547–552). Sydney: University of New South Wales. http://www.cse.buffalo.edu/~rapaport/Papers/context.auconf.pdf.

Rapaport, W. J. (2005). In defense of contextual vocabulary acquisition: How to do things with words in context. In A. Dey, B. Kokinov, D. Leake, & R. Turner (Eds.), *Modeling and using context: 5th international and interdisciplinary conference, CONTEXT 05, Paris, France, July 2005, Proceedings* (pp. 396–409). Berlin: Springer-Verlag Lecture Notes in Artificial Intelligence 3554. https://cse.buffalo.edu/~rapaport/Papers/35540396.pdf.

Rapaport, W. J. (2006). How Helen Keller used syntactic semantics to escape from a Chinese room. *Minds and Machines, 16*, 381–436. http://www.cse.buffalo.edu/~rapaport/Papers/helenkeller.pdf (see reply to comments, in Rapaport 2011).

Rapaport, W. J. (2011). Yes, she was! Reply to Ford's 'Helen Keller was never in a Chinese room'. *Minds and Machines, 21*(1), 3–17. http://www.cse.buffalo.edu/~rapaport/Papers/Papers.by.Others/rapaport11-YesSheWas-MM.pdf.

Rapaport, W. J. (2012). Semiotic systems, computers, and the mind: How cognition could be computing. *International Journal of Signs and Semiotic Systems, 2*(1), 32–71. http://www.cse.buffalo.edu/~rapaport/Papers/Semiotic_Systems,_Computers,_and_the_Mind.pdf. Revised version published as Rapaport 2018.

Rapaport, W. J. (2017a). On the relation of computing to the world. In T. M. Powers (Ed.), *Philosophy and computing: Essays in epistemology, philosophy of mind, logic, and ethics* (pp. 29–64). Cham, Switzerland: Springer. Paper based on 2015 IACAP Covey Award talk; preprint at http://www.cse.buffalo.edu/~rapaport/Papers/rapaport4IACAP.pdf.

Rapaport, W. J. (2017b). Semantics as syntax. *American Philosophical Association Newsletter on Philosophy and Computers, 17*(1), 2–11. http://c.ymcdn.com/sites/www.apaonline.org/resource/collection/EADE8D52-8D02-4136-9A2A-729368501E43/ComputersV17n1.pdf.

Rapaport, W. J. (2018). Syntactic semantics and the proper treatment of computationalism. In M. Danesi (Ed.), *Empirical research on semiotics and visual rhetoric* (pp. 128–176). Hershey, PA: IGI Global. References on pp. 273–307; http://www.cse.buffalo.edu/~rapaport/Papers/SynSemProperTrtmtCompnlism.pdf. Revised version of Rapaport 2012.

Rapaport, W. J. (2019). *Philosophy of computer science*. Current draft in progress at http://www.cse.buffalo.edu/~rapaport/Papers/phics.pdf.

Rapaport, W. J., & Kibby, M. W. (2002). Contextual vocabulary acquisition: A computational theory and educational curriculum. In N. Callaos, A. Breda, & M. Y. Fernandez J. (Eds.), *Proceedings of the 6th world multiconference on systemics, cybernetics and informatics (SCI 2002; Orlando, FL), Vol. II: Concepts and applications of systemics, cybernetics, and informatics I* (pp. 261–266). Orlando: International Institute of Informatics and Systemics. http://www.cse.buffalo.edu/~rapaport/Papers/sci2002.tr.pdf.

Rapaport, W. J., & Kibby, M. W. (2007). Contextual vocabulary acquisition as computational philosophy and as philosophical computation. *Journal of Experimental and Theoretical Artificial Intelligence, 19*(1), 1–17. http://www.cse.buffalo.edu/~rapaport/Papers/cva-jetai.pdf.

Rapaport, W. J., & Kibby, M. W. (2010). *Contextual vocabulary acquisition: From algorithm to curriculum*. http://www.cse.buffalo.edu/~rapaport/CVA/reading4CgSJnl.pdf.

Rapaport, W. J., & Kibby, M. W. (2014). Contextual vocabulary acquisition: From algorithm to curriculum. In A. Palma (Ed.), *Castañeda and His Guises: Essays on the work of Hector-Neri Castañeda* (pp. 107–150). Berlin: Walter de Gruyter. http://www.cse.buffalo.edu/~rapaport/Papers/reading4HNC.pdf.

Rescorla, M. (2012). Are computational transitions sensitive to semantics? *Australian Journal of Philosophy, 90*(4), 703–721. http://www.philosophy.ucsb.edu/docs/faculty/papers/formal.pdf.

Rescorla, M. (2014). The causal relevance of content to computation. *Philosophy and Phenomenological Research, 88*(1), 173–208. http://www.philosophy.ucsb.edu/people/profiles/faculty/cvs/papers/causalfinal.pdf.

Searle, J. R. (1990). Is the brain a digital computer? *Proceedings and Addresses of the American Philosophical Association, 64*(3), 21–37. Reprinted in slightly revised form as Searle 1992, Ch. 9.

Searle, J. R. (1992). *The rediscovery of the mind*. Cambridge, MA: MIT Press.

Shapiro, S.C., & Rapaport, W. J. (1987). SNePS considered as a fully intensional propositional semantic network. In N. Cercone and G. McCalla (Eds.), *The knowledge frontier: Essays in the representation of knowledge* (pp. 262–315). New York: Springer. https://www.cse.buffalo.edu/~rapaport/676/F01/shapiro.rapaport.87.pdf.

Shapiro, S.C., & Rapaport, W. J. (1992). The SNePS family. *Computers and Mathematics with Applications, 23*, 243–275. Reprinted in F. Lehmann (ed.), *Semantic networks in artificial intelligence* (Oxford: Pergamon Press, 1992): 243–275; http://www.sciencedirect.com/science/article/pii/0898122192901436.

Shapiro, S.C., & Rapaport, W. J. (1995). An introduction to a computational reader of narratives. In J. F. Duchan, G. A. Bruder, and L. E. Hewitt (Eds.), *Deixis in narrative: A cognitive science perspective* (pp. 79–105). Hillsdale, NJ: Lawrence Erlbaum Associates. http://www.cse.buffalo.edu/~rapaport/Papers/shapiro.rapaport.95.pdf.

Smith, B. C. (1987). *The correspondence continuum*. Technical report CSLI-87-71, Center for the Study of Language & Information, Stanford, CA.

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.