A New AI

Charles Petrie, *Stanford University, Stanford, CA, 94305, USA*

Artificial intelligence (AI) is a recent subfield of the already young field of computer science, which has clear mathematical foundations in complexity and queuing theories as well as solid engineering in compilers, databases, and computer architecture. AI, in contrast, has been notably ill-defined and has been mostly an applied practice of two kinds: 1) symbolic reasoning and 2) neural networks. Both have an extensive scientific basis but the claim of both to be "AI" is disputed. Certainly, this is unnecessary; but the dispute arises because the actual application of AI seems unambitious given its name. We examine this through a personal retrospective and suggest some future research.

It is hopeless to cover the history of artificial intelligence (AI), so I will only hit highlights that were important to me personally. This will be my personal retrospective together with the thoughts I have had in retirement that may be, or not, of value to those who continue the research. As an officially old person, I get to go way back.

I started computing when my university got its first mainframe and offered its first undergraduate programming course (FORTRAN IV). I read the textbook during the Mardi Gras holidays (LSU), figured out how to steal an account, found the subroutines for running the Calcomp plotter, and started ignoring the class. When I had exhausted the account and the patience of the machine operators, I showed my instructor what I had done. He pulled me from the class and put me to work writing graphics subroutines for the engineering department. Until then, they had no way of rotating objects and viewing them from different perspectives on the computer. That is where we were in computing when I started.

At the same time, I was introduced to mathematical logic, which would be the basis for symbolic computing. I was only 20 and impressionable.

SYMBOLIC VERSUS LEARNING AI

With the success of deep learning, now that is what is often meant by "AI." But, of course, it is just one technique with pros and cons, and new research and developments. The division is not as deep as some imply.[1] Symbolic reasoning is still taught in computer science departments and good work is going on in combining the advantages of both.[2] Deep learning is critically important for entire nations racing for advantage in the new technology. It might be worth noting that previously Japan and the U.S. had a competition in logic programming.

My concern here is that neither is about finding a model of intelligence but rather mostly on producing more useful computer applications. In fact, I have often objected to using the word "intelligent" to describe technologies as a meaningless adjective.[3] But before I get to this, let me dig into the kinds of AI a bit more rather than giving them only the aforementioned dismissive paragraph.

SYMBOLIC AI

Symbolic AI is of particular importance to me. I was introduced to mathematical logic by Kleene's introductory textbook[4] as an undergraduate and at the same time I learned FORTRAN. So all through my first year of mathematics graduate school, I worked on what I thought was an original idea that proofs could be computer-generated in spite of Gödel's incompleteness theorem, much to the dismay of my mathematics professors who thought computers were not important. From that ignominious start, I somehow ended up studying and working under Prof. Woody Bledsoe in the 1980s, just after having discovered Jon Doyle's then recent paper on "Truth Maintenance" (TMS),[5] which has inspired much of my research. I was captured by the idea that some form of automatic theorem proving (ATM) and TMS with dependence-directed backtracking (DDB) could lead to help with

1089-7801 © 2021 IEEE
Digital Object Identifier 10.1109/MIC.2021.3063025

distributed problem solving. Our group built some pretty sophisticated (for the time) reasoning mechanisms that had some initial success.[6]

The point about this is that almost as soon as I had discovered AI, I realized that it was not about "intelligence" as much as it was harnessing new symbolic reasoning mechanisms based on computational logic to do useful things not previously feasible, starting with natural language processing under Bob Simmons.[7] This was fun, challenging, and satisfying. I got a pretty good career out of it. And I do not think that kind of AI is finished.

I pursued symbolic reasoning at Stanford with agents, planning, and semantic networks. A lot has been done by this community and it continues today. I have been a steadfast critic of overblown expectations about specific technologies, but I have never doubted their potential even if it has yet to be realized. My main criticism of the community has, in fact, been is failure to tackle hard problems. For years, I headed the Semantic Web Challenge,[8] which showed just how hard even simple problems could be, especially if one is trying to create composable technologies. In fact, that was the vision: Pieces of problem solving technology could be automatically assembled to solve ever large problems. This vision has yet to be realized, mostly because there is no academic credit to be had in solving the remaining hard engineering issues and the industrial engineering community is not interested in the exotic academic technologies.

I also espoused another vision, about which I have often written in this magazine.[9] The idea was that rather than solving problems, computers could help coordinate humans to work together in large distributed projects. The system would not just allow humans to share information but would proactively detect both overall design conflicts and opportunities and offer suggestions about how the people involved might resolve these collectively. I called this research "Coordination Science"[10] and pointed out that my own work was only an existence proof and that many issues in the topic offered a rich field of research as well as a big potential for improving large project management. In fact, it was Doyle's original TMS that inspired this vision so many years ago.

Why have none of these symbolic efforts met expectations? This is, in part, because our expectations were too high for the state of the technology. Expert systems were originally derided for their lack of success; in fact, that technology was widely used though it was embedded in other systems.[11] The general failure of symbolic reasoning to meet claims is not only that the claims were too visionary, but also that the technologies were not easily incorporated into more traditional systems and that they were often solving problems that no one was asking nor would ask. One of our initial technologies back in the 1980s was an early success for a while until the company using it determined after a few years that they simply could not train enough industrial engineers to support the new product, as there was no overlap with standard technologies.[6]

As another personal case, in looking back over my promotion of coordination science, I now see that I failed to appreciate many social factors: especially that of community. We had actually built a prototype of a coordination system that solved an important problem for a major defense contractor. But while they appreciated much of it, they took our distributed system and mashed it into a centralized one, because the engineers preferred to work together in one room on one computer rather than on distributed specialized ones. Later, I published a completely informal theory called emergent collectives also about distributed work but about ones that expanded because the individuals had the feeling they were contributing to a community.[12] This throw-away informal thesis was surprisingly popular.

I never put those two things together until now, many years later. And I suspect that much of the agent and semantic network community are still struggling with a similar lack of appreciation for how the "neurotypical" works. I strongly suspect that many of us in these fields are the kind of people who are puzzled by large groups of people who turn out to cheer sports teams. I could go on about this, but it is not the main point and I have probably said too much already. I will just note that a common joke in Silicon Valley is that it is where people with Asperger's come to breed.

DEEP LEARNING

As this is not my area of expertise, I will give it even shorter shrift than I have symbolic reasoning.

Early Days

Neural networks were widely derided by we, the symbolic reasoning researchers, in the early days. One reason was on principle. What did it mean to learn something? There are, in symbolic reasoning, only three kinds of reasoning. Inductive reasoning is common but unreliable. Abductive reasoning is less common and may lead one down wrong path, but is used in DDB; it will eventually lead to an answer consistent with deductive reasoning. Deductive reasoning is reliable but the knowledge deduced was in some sense already known as it was implicit in the logical closure of the axioms and deductive system. The only sense of "learning" in such thinking is discovery of empirical fact. So how could neural networks be learning anything?

Neural networks were also derided by people such as me because when we experimented with the systems back in the 1980s, we found that reported successes depended heavily upon the researchers stopping the cycling of the systems when they recognized a correct answer.

Finally, such systems were derided by us because it was clear that, unlike symbolic systems, the results were "black boxes" that imparted no knowledge about the relationship of intrinsic conceptual relationships. These were systems that may somehow be trained to behave usefully.

Deep Learning Wins

Flash forward to now and learning systems completely dominate "AI." The systems can be trained not only to be useful but to win cognitive games to which symbolic systems only aspired for decades. When in 2016 AlphaMove made the 37th move in the second game against Lee Sedol, it was also game over for symbolic reasoning.[13]

What happened is best explained elsewhere, but faster machines enabled more layers of neural networks (thus "deep") and together with other techniques such as reinforcement learning, combining feature extraction with classification, and various advanced computation techniques, as well as access to big data, now the successes of deep learning are widely appreciated and even feared.

And what did they learn? What they learned is at least analogous to the theories developed by scientists to explain empirical facts. These theories are successful because they usefully predict results. Anyone of these theories could be wrong in which new findings would invalidate them. So theories are not the kind of knowledge symbolic reasoners understand, or we understand them to be merely inductive. And worse, it is very difficult to impossible to actually extract such theories from deep learning systems. But if we ourselves are learning anything, it is such theories that can predict results, which turn out to be incredibly useful.

That said, their applications are still limited and can still be extended by symbolic reasoning. Their reasoning explanation systems are still virtually nonexistent, though work is advancing in this area as well. But finally, notice that though "behaving usefully" is quite an understatement of deep learning systems, they are still only doing that. While there is real science behind these systems, they tell us little about the reasoning employed for particular results.

Common Critique of Both

Symbolic AI also concentrated on results. Of course results are paramount. But symbolic reasoning researchers mostly did what I did: We realized that it was too early to usefully conjecture about the nature of intelligence, and instead, we concentrated on increasingly sophisticated reasoning systems, based upon constructed knowledge networks, that could be useful. When deep learning systems produced systems that were more useful, then that technology won that game.

Of course there is much more that could be done. And, as with learning systems, no one knows what might grow out of such research in the decades to come. Not only is the combination of symbolic and learning systems promising[2] but symbolic systems still have the potential to produce, for example, plans that are provably correct but revisable as well. Learning systems might produce better coordination systems than the symbolic ones that were they applied to that application. Semantic networks might be linked by metasemantics that augment linked systems.

Still, all this seems to miss the mark of AI even more than did the well-known simple script-based program Eliza.[14] I and other researchers have not been ambitious enough, reasonably because of the state of computer technology. But it is the 21st century and time perhaps to be more ambitious, especially since the first steps may not involve more than ambitious thinking.

INTELLIGENCE

If we are going to discuss "AI," it might be worthwhile to point out that we do not really have a good consensus definition of the word "intelligence." For instance, if you ask someone on the spectrum if they know what time it is, and they reply "yes," is that answer intelligent? Is that person?

Intelligent Software Agents

As noted previously, I do not like using the word "intelligence" to describe technologies[3] because it is usually used as a marketing term rather than to distinguish technologies, which is what names should do.[15] But to reprise a talk I have given on this topic,[†] it is useful to

[†]http://www-cdr.stanford.edu/~petrie/Intelligence/index.html

consider what this means to at least one technology class: "Intelligent software agents," because, as I have said and written many times during my professional career, technical distinctions without a difference are worse than useless.

I once gave a talk on the mushy use of this term for this technology. It was in Europe, so no one stood up in public to argue with me, but a young researcher came to me afterward with what she seemed to think was a solid counterexample: "But I'm building an intelligent agent." My reply was "How would you know?" She went off to think about it.[‡]

Intelligent software agents as a technology have been variously defined as software that is reactive, deliberative, perceptive, sensing, and/or autonomous. The alert reader will note that these are hopelessly subjective criteria. I have previously proposed an operational definition that would distinguish this technology.[16] Part of the definition depended upon the messaging protocol. Another part is more relevant for this discussion: The agents in such a system must accomplish a task by exchanging messages and must use a peer-to-peer protocol in order to achieve optimum task performance. This definition admittedly depends upon some definition of accomplishing an application task optimally. But it is better than "autonomous."

An individual software module is not an agent at all if it can communicate with the other candidate agents with only a client/server protocol without degradation of the collective task performance. In the work by Petrie,[3] I argue that this definition captured and excluded software systems that fit our intuitions of the name and so was a viable thesis.[§] I am not going to go further down that path here as I have previously written about this and it is out of scope for this retrospective and the critique of AI I want to make here.

Intelligence Is Social

The key idea that I want to abstract from that definition thesis is that of volunteering useful information in order to accomplish better a joint task. More precisely, but not much more, I want to propose that intelligence can only be demonstrated socially. An agent (of some level of personhood) is intelligent when it can understand, perhaps without being told, the goal of at least one other agent and act (perhaps only with a speech act) to either help or hinder that goal, in furtherance of its own goal. I am not going to go into a deep discussion about how these goals are to be known here, though that would be a fun discussion.

Consider the following examples, please. If you talk to someone and you can predict all of their replies, you will not think they are intelligent. If you ask them what time it is and the answer is always "noon," the conclusion is the same. If you asked in 2001 "Do you know the conversion from Euros into Deutsche Marks?" and the person replied "Yes, it is 2.1, but you should know there will be no more DM after next January," you would consider that an intelligent answer. If you asked a travel agent "Can you please provide me with a travel package to Croatia?," a clever travel agent might say "Here is everything you need based on your personal preferences." But an even more clever agent in 1994 might say "You might want to go to Greece," based upon current war conditions. "You might consider Crete this year" says the really smart travel agency in an unsolicited e-mail nine months after the first Greek trip, anticipating a return trip. You should see that these examples fit this social idea of intelligence.

One might reasonably object that various kinds intelligence tests do not involve this idea of intelligence. Again, without a long discussion, I would counter that either just some talent or skill is being tested, or the testee is figuring out what the tester intended. I have done the latter personally in order to test well. I am always modeling the mind of the tester in order to give the "right" answers. I have found this to be of immense use even in university physics. One time I scored 94 in a physics test when the next highest grade in the class was in the 1970s. I know from talks with my classmates that I did not understand the physics we were being taught any better than some of them, but I asked questions of the professor constantly and so understood the answers he wanted better than anyone else who took the test.

You should also see how this definition fits game playing: especially AlphaMove's 37th move in which it purportedly made the move thinking the human player would not expect it. My only other comment here is that we would recognize AlphaMove as really intelligent if it could do this in general circumstances rather than in just the narrow domain of the game of Go.

Machines That Outperform Humans

This should be contrasted with "Eliza," which fooled people who did not know its code. They ascribed

[‡]This is similar to the third question of the three questions I ask from Ph.D. thesis proposers: What is the problem you propose to solve, why should I care about it, and how should I determine whether you have made any progress toward a solution?

[§]The idea of such a "thesis" is that some more formal definition captures the notions of informal usage of a word, such as "computability."

agency to it, in part because humans seem predisposed to ascribe agency to everything: Vengeful winds in pantheism or cognitive atoms in process philosophy are examples. I also ascribe the power of Eliza to the fact that most people at the time were unfamiliar with computers and were surprised at the apparent insights generated by this simple algorithm, an algorithm long before discovered by humans. Eliza *seemingly* generated surprisingly and useful statements based upon knowing something about the user. It seemed intelligent. This mistake points to the social nature of our perception of intelligence.

Again, I am not denigrating autonomous vehicles or powerful automatic theorem provers (ATPs). But I have never seen an ATP program I thought was intelligent. Even in the old days when I was young and could actually prove theorems they gave us in class, even as a human, I was not doing it intelligently. I, like some others, had some degree of innate talent. Give me a theorem, and I will give you a proof, for sufficiently simple theorems. But I had no idea of the use of these theorems. In fact, I would forget ones I had proved as soon as I proved the next theorem. I was just a human ATP: I was displaying a talent, not intelligence.

At the risk of creating a strawman, this example does offer a possible critique of my own idea of intelligence though. Surely ATP is an example of "mere" useful problem-solving, at least when it is applied. And "useful" implies goals. If ATP is used to help someone, does that not fit with my proposed definition of intelligence? Not really. It is we researchers who are displaying intelligence by developing problem-solvers that will help other people. If we have correctly modeled those people and our software is seen as having the potential to solve problems about which those people care, then we may get funded. But this does not mean that the useful problem-solving software itself is intelligent.

There are many possible elaborations of this notion of intelligence. Certainly, speed of developing surprisingly useful information is important as well as the degree of surprising usefulness. Mathematics and physics provide communication mediation, as well as the hierarchies of people that understand and can appreciate these results to various degrees, and apply them to improving problem solutions.

My thesis here, in summary, is that intelligence is best understood objectively as a social phenomenon where one agent acts or volunteers information surprisingly in such a way as to help or hinder the goals of a second agent in accordance with the goals of the first. But that is just my simplistic notion of intelligence, and as such, may be wrong or not even wrong. It is certainly not well developed.

My real point is that were AI to have pursued that, or some other, specific idea of intelligence initially, we might have been further along by now, instead of developing merely (very) useful engineering applications. That is my critique of AI in this section: It has never had a well-defined goal. This brings to mind Lewis Carroll's saying that if you don't know where you're going, any road will get you there. It may be time to refocus.

But I have another and very different critique of AI, and suggestion for ambitious thought.

ARCHITECTURES OF THINKING

Another, and not inconsistent, way for AI to have gone was to try to understand "mind" in computational terms. Early work seemed to point in this direction but not much has been done. A few software systems have gone so far as to define "concepts" but I have never found these to be persuasive (here I skip references as they would be pejorative).

My father was an appliance mechanic for Sears (Google it). He drove to people's homes, usually out in the country since we lived in a rural part of Louisiana, in response to reported malfunctions. Dad would return home in the evening with gifts of vegetables and humorous stories. One story was that a woman complained that her washing machine simply did not run at all. My father pointed out to her that the wash settings dial was set in between two settings. She was used to radio dials, which were more or less continuous. This was her first exposure to digital thinking. When a researcher can show me how her concept of "dial" can evolve from continuous to discrete settings, I will allow as to how they might be on to something. When they can computationally explain why the rest of us would see the problem and laugh without an explanation that she only knew of radio dials, I will be very impressed.

This is my segue into a careful discussion of an aspect of consciousness, which I carefully avoided in the section "Machines that Outperform Humans." I am not proposing solving that problem. And I know that many people have tried to "computerize" cognition. I just think they have missed something basic.

Previously I have proposed that intelligence involves modeling someone else's goals. In order to get on with something more substantial, I will here briefly propose the not-controversial notion that consciousness involves modeling one's self. This is more than just being aware of one's surroundings and not only reacting, but planning actions, in order to maintain one's homeostasis. Consciousness allows one to reflect upon one's thinking about this process, and in fact, there is probably a metric of consciousness dependent on how many layers of regression of such thinking of which one is capable. Maybe tests can be designed to measure this. I do not know. I am interested in theories that have explicative power. That is, more specific theories that predict results consistent with our experiences. Such theories would be similar to Turing's thesis that his formal notion of computable functions fit mathematicians' intuitive use of the term.

There is a lot of work on the neural basis of consciousness. I will only cite here one book,[17] which explores the possible neural basis of consciousness in depth, and a popular article by the same author.[18] The latter notes that our consciousness arises in the cerebral cortex, which is an expanded version of the wulst, which, in turn, arose from the more primitive tectum. I am here not interested in consciousness *per se*.

I am interested in that these theories involve self-modeling in order to control *attention*. The theory espoused here is that these physical structures evolved to support increasingly powerful modeling functionality, including self-modeling, and then the modeling of others. It argues that other animals are also self-conscious to varying degrees that proved useful in their evolution. This is not much of a "theory" by my standards, but I want to quote one line in the popular article[18]: "Even if you've turned your back on an object, your cortex can still focus its processing resources on it."

What did that sentence mean? For me, a crucial question about thinking is "what is attention"? If it means switching the focus of processing resources, what are those resources? I do not think so far that the neural basis of thinking that has been done with MRIs and such is very helpful, at least to computer scientists. We should rather be asking about the computational mechanism at work.

Attention

We all know what "attention" is: We use it and depend upon drawing other people's attention to things. But really, how does that work?

We know how it works in computers. There is a central processing unit (CPU) that is fed instructions from various programs to execute. There is an operating system (OS) that determines how many of the cycles of the CPU each program gets at any one time. In computers controlling real-world processes, sensors may feed into the some kind of "interrupt pad" that will override normal programs and give precedence to a program that should respond to the sensor input. That is a kind of attention we understand at the level of a mechanical operating process, that is to say, computationally. The interrupt pad refocuses the processing power of the CPU on an important program. What is the computational mechanism of our attention? *What are the computational resources being refocused?* That is, here, the primary issue to which I want to draw your attention.

The neural studies I cite earlier, and many others, make it clear that the brain is composed of a dazzling array of specialized structures interacting with each other in ways we have yet to understand fully, if we ever do. Minsky proposed this long ago.[19] What is important for me is that all of these structures are specialized for some purpose. And everyone of those neuronic connection structures is weighted for a special purpose. That is how neural networks work. There seems to be no general purpose "CPU."

Were there such? It would be some set of neurons and connections that are so flexible as to experience and control other specialized sets, and indeed train the specialized structures. When I say "mimic," I actually do not know what I am saying exactly. When your attention is drawn to one set of thoughts over another, it is clear that the thoughts are different and specialized. But somehow the "you" that is now thinking differently is the same.

If attention is performed by a separate structure itself, and there is evidence it is and where it is located,[¶] its job is to create some kind of model that can "experience" the processes of other structures. It devotes some kind of general resource to these processes. But this general resource is some kind of specialized structure itself. Whatever it is, this is the computational resource being refocused.

We can be pretty sure there is not some set of neurons somewhere in our brain that can instantly adopt the weighings and connections of other neuronic structures and carry out their processes and thus create our experience of those processes. Rather there is more likely a structure that can model, at some level of abstraction, what processes any of those structures are performing, with inputs from those structures to stimulate the model/simulation. Attention is switching those inputs, and of course there are control mechanisms for that switching. *But it is what resources are being switched that is the computational puzzle.*

This is far from some simple CPU that can process different instructions. We actually do not know what

---

¶The book[17] precisely locates it in the dorsal temporoparietal junction.

the attention brain structure is doing, but we should. Computer science, in general, should be thinking about what kind of computational mechanism could do this.

Subjective Time

Whatever this mechanism is should explain our experience of subjective time. We all know that when we focus on an external event, we slow it down. Somehow, the more we concentrate, the slower the event occurs. A watched pot never boils. The accident unfolds in slow motion. What exactly is being concentrated? What is this resource?

On the other hand, if we are pleasantly distracted, time flies. One simple hypothesis is that one of our specialized processes is "counting" changes and keeping time that way. When we are pleasantly distracted, resources are diverted somehow from that counting process and it forgets to count. When we concentrate our resources on looking at events, we count too many of them and time slows.

Time also flies when there are few external events to count and we are not trying to do so. If you spend time alone in a cave, you can easily lose weeks of time.

So there is this computational resource that can be concentrated on counting eternal events but when this resource is diverted to other processes, the counting goes awry, as it does if there is an unusual dearth of events to count. *What is this computational resource?* How could this be modeled mathematically? What kind of computer architecture would behave like this? Why are not we thinking more about this?

There are many aspects of attention, including our ability to think about something over a long period of time and how things like smartphones train our attention mechanisms to be shorter, whatever that really means. But it seems to me that focusing on the question of what resource is being focused is the key question, and that if someone develops a computational architecture of attention, then it should make predictions about subjective time that correspond to observations.

Relevant Research

My personal research strategy is that problems that have withstood the efforts of the smartest people, well, forever are best avoided, though eventually someone may solve them. It is just not likely to be me. I did foolishly try to solve the four-color map problem in the ninth grade and I learned from that experience.

I know psychologists such as William James have addressed attention as a subject, but they have not addressed it computationally.

There is a lot of good work done by smart people in the area of trying to understand cognition computationally.

There is an area of research called *Computational Cognitive Modeling*[20] of behavior. This is actually taught as a class.[*] All this work is being done by psychologists, and does not answer the questions I am posing.

Much closer is *The Computational Theory of Mind*[21] by philosophers. This surveys the various formal approaches to modeling organic thinking. But this is largely done by philosophers and though it takes into account formal semantics and logic theory, it is not quite what I am addressing here.

I like very much "Brain Computation: A Computer Science Perspective"[22] that represents various works, and problems, in mapping our organic cognition onto computer architectures.

But none of these seems to address the obvious, glaring, problem of attention.

CONCLUSIONS

So there you have it: my personal reflection on and critique of AI. There has been a failure to define intelligence as a goal of the field and, moreover, a failure to explore the nature of attention from a computational architecture perspective. Or, put more constructively, a retrospective of our progress to date suggests that we might hope the new generation of computer science researchers might pursue the ambitious issues of modeling intelligence and attention computationally.

As I often say in lectures, I may have missed something important. I may be entirely wrong. Please feel free to correct me, or tell me why the problems I have posed are the wrong ones, or correct my vague formulation of them. I would be delighted were someone to build on these meandering thoughts.

The author would like to thank you for your attention.

REFERENCES

1. M. Wooldridge, "Artificial intelligence is a house divided," *The Chronicle of Higher Education*, The Review Jan. 20, 2021. [Online]. Available: https://www.chronicle.com/article/artificial-intelligence-is-a-house-divided

---

[*] https://brendenlake.github.io/CCM-site/

2. M. Ebrahimi, A. Eberhart, F. Bianchi, and P. Hitzler, “Towards bridging the neuro-symbolic gap: Deep deductive reasoners,” *Appl. Intell.*, 2021, doi: 10.1007/s10489-020-02165-6.
3. C. Petrie, “Agent-based engineering, the web, and intelligence,” *IEEE Intell. Syst.*, vol. 11, no. 6, pp. 24–29, Dec. 1996, doi: 10.1109/64.546579.
4. S. C. Kleene, *Mathematical Logic (Dover Books on Mathematics)*. Chelmsford, MA, USA: Courier Corp., 2013, p. 416.
5. J. Doyle, “A truth maintenance system,” *Artif. Intell.*, vol. 12, no. 3, pp. 231–272, 1979, doi: 10.1016/0004-3702(79)90008-0.
6. D. Gibson and E. Rogers, “Technology transfer and MCC,” in *R&D Collaboration on Trial*. Boston, MA, USA: Harvard Bus. School Press, 1994, p. 607.
7. R. Simmons and D. Chester, *Computations From the English: A Procedural Logic Approach for Representing and Understanding English Texts*. Englewood Cliffs, NJ, USA: Prentice-Hall, 1984, p. 297.
8. C. Petrie, T. Margaria, H. Lausen, and M. Zaremba, Eds., *Semantic Web Services Challenge: Results From the First Year* (Semantic Web and Beyond, 8, XXII). Berlin, Germany: Springer, 2009, p. 290.
9. C. Petrie, “Plenty of room outside the firm [Peering],” *IEEE Internet Comput.*, vol. 14, no. 1, pp. 92–95, Jan./Feb. 2010.[Online]. Available: http://www-cdr.stanford.edu/petrie/online/peer2peer/vision2010.pdf
10. C. Petrie, “Enterprise coordination on the Internet,” *Future Internet*, vol. 3, no. 1, pp. 49–66 2011, doi: 10.3390/fi3010049.
11. A. Hameurlain *et al.*, “Database and expert systems applications,” *in Proc. 22nd Conf.*, 2011, p. 252.
12. C. Petrie, “Emergent collectives for work and play,” *Société De Stratége AGIR Revue Generale De Stratégié 20-21, La Société De L’information*, pp. 146–152, Jan. 2005. [Online]. Available: http://www-cdr.stanford.edu/petrie/revue/
13. C. Metz, “In two moves, AlphaGo and Lee Sedol redefined the future,” *Wired Business*, 2016. [Online]. Available: https://tinyurl.com/yxnguae7
14. J. Weizenbaum, “ELIZA—A computer program for the study of natural language communication between man and machine,” *Commun. ACM*, vol. 9, no. 1, pp. 36–45, Jan. 1966, doi: 10.1145/365153.365168.
15. C. Petrie and C. Bussler, “Industrial light and semantics [Peering],” *IEEE Internet Comput.*, vol. 10, no. 1, pp. 94–96, Jan./Feb. 2006. [Online]. Available: http://www-cdr.stanford.edu/petrie/online/peer2peer/w406.pdf
16. C. Petrie, *Agent-Oriented Software Engineering (Lecture Notes AI, 1957)*, P. Ciancarini and M. Wooldridge, Eds. Berlin, Germany: Springer-Verlag, 2001, pp. 58–76. [Online]. Available: http://www-cdr.stanford.edu/petrie/agents/abse/abse.html
17. M. Graziano, *Rethinking Consciousness*. New York, NY, USA: W. W. Norton, 2019, p. 203.
18. M. Graziano, “A new theory explains how consciousness evolved,” *The Atlantic*, Jun. 6, 2016. [Online]. Available: https://www.theatlantic.com/science/archive/2016/06/how-consciousness-evolved/485558/
19. M. Minsky, *The Society of Mind*. New York, NY, USA: Simon & Schuster, 1986, p. 339.
20. R. C. Wilson and A. G. C. Collins, “Ten simple rules for the computational modeling of behavioral data,” eLife, vol. 8, 2019, Art. no. e49547, doi: 10.7554/eLife.49547.
21. M. Rescorla, “The computational theory of mind,” *Stanford Encyclopedia of Philosophy*, E. N. Zalta, Ed. Stanford, CA, USA, Stanford Univ., 2020. [Online]. Available: https://plato.stanford.edu/archives/fall2020/entries/computational-mind/
22. W. Maass, C. H. Papadimitriou, S. Vempala, and R. Legenstein, “Brain computation: A computer science perspective,” in *Computing and Software Science* (Lecture Notes in Computer Science, 10000), B. Steffen, G. Woeginger, Eds. Berlin, Germany: Springer, 2019, doi: 10.1007/978-3-319-91908-9.

CHARLES PETRIE has retired from Stanford University, Stanford, CA, USA, where he was a Senior Research Scientist with the Computer Science Department. Petrie received the Ph.D. degree in computer science from The University of Texas at Austin, Austin, TX, USA. He is the founding EIC of *IEEE Internet Computing*. His professional biography, including CV and contact information, is at http://www-cdr.stanford.edu/~petrie/bio.html. Contact him at cjpetriejr@gmail.com.