Applied Artificial Intelligence an International Journal

ISSN: 0883-9514 (Print) 1087-6545 (Online) Journal homepage: www.tandfonline.com/journals/uaai20

THE FUTURE OF ARTIFICIAL INTELLIGENCE: LEARNING FROM EXPERIENCE

ROGER C. SCHANK & STEPHEN B. SLADE

To cite this article: ROGER C. SCHANK & STEPHEN B. SLADE (1991) THE FUTURE OF ARTIFICIAL INTELLIGENCE: LEARNING FROM EXPERIENCE, Applied Artificial Intelligence an International Journal, 5:1, 97-107, DOI: 10.1080/08839519108927919

To link to this article: https://doi.org/10.1080/08839519108927919

Published online: 22 Oct 2007.

Submit your article to this journal

Article views: 169

View related articles

Full Terms & Conditions of access and use can be found at
https://www.tandfonline.com/action/journalInformation?journalCode=uaai20

THE FUTURE OF ARTIFICIAL INTELLIGENCE: LEARNING FROM EXPERIENCE

ROGER C. SCHANK
Institute for the Learning Sciences,
Northwestern University, Evanston, Illinois

STEPHEN B. SLADE
Yale Artificial Intelligence Project,
Yale Department of Computer Science,
New Haven, Connecticut

We view the future of artificial intelligence from three perspectives: scientific, technological, and educational. The common theme among these views is the central importance of learning, particularly in problem domains where case-based reasoning seems appropriate. From a scientific perspective, an adequate model of the mind must account for the phenomena of learning. Learning must also be considered a technological requirement for computer systems both to facilitate the initial knowledge acquisition and ultimately to adapt to new situations. Finally, a scientific model of learning, together with AI technology, can form the basis for a new mode of education. This technology can be applied to instruction for a wide range of subjects.

LEARNING FROM EXPERIENCE

In predicting the future, we look to the past. When faced with a new problem to solve, we are often reminded of previous similar episodes, the solutions to which may be adapted to the current problem. When we find ourselves in an unfamiliar situation, we recognize it as such precisely because we have no appropriate remindings. A novel situation has no exemplars. Such a situation primes the mind for learning: we want to remember this new situation as being a special circumstance, so that we can recall this episode in the future. Often we will not recognize a situation as being different from some past episode until an expectation generated by the earlier case fails in the new case. This failure triggers the learning for the new case. In short, this process comprises *learning from experience*, or *case-based learning*, which we will discuss below in the section on scientific goals.

Case-based learning in AI has been explored in recent years (Hammond, 1986; Lebowitz, 1980; Schank, 1982; and Schank, 1986) along with the

This work was supported in part by the Advanced Research Projects Agency of the Department of Defense and monitored by the Office of Naval Research under contract N00014-85-K-0108, by the Air Force Office of Scientific Research under contract AFOSR-85-0343, and by the National Library of Medicine under grant 5-RO1-LM04251.

Copyright © 1991 by Hemisphere Publishing Corporation

underlying paradigm of case-based reasoning (Bain, 1986; Kolodner et al., 1985; Simpson, 1985). The case-based reasoning model can be contrasted to the rule-based, production system approach in which knowledge is encoded as if-then rules. We discuss some of these differences below in the section on technology.

A psychological model of learning is useful not only in building computer programs that learn, but also for designing computer programs that *teach*. We discuss the application of the theory of case-based learning to programs for intelligent computer assisted instruction (ICAI) in the section on educational goals.

However, before we delve into specific aspects of case-based reasoning, we can briefly illustrate its use in examining the future of AI and its impact on society. This, after all, is our only recourse: it is unlikely that we have many rules that can tell us what the future of AI will be, or how AI will affect society. However, we do have relevant experience indicating how AI has progressed in the past, and what the societal impact of that progress has been, as well as similar developments in analogous and allied areas.

Here are some case-based analyses of the future of AI:

- Progress in hardware has been more steady and reliable than progress in theories or software. Lisp machines, inexpensive workstations, and mammoth connection machines are technological realities. We can forecast continued improvement in hardware performance. Given the regularity of improvements in hardware, some observers have formulated actual rules for predicting machine performance. For example, Gordon Bell has suggested that memory chip capacity in year X will be 2^{X-1962} bits.
- Major theoretical breakthroughs are unlikely, though not impossible. Results in AI in the past decades have not been as breathtaking as in other areas of computer science. The apparent lesson is: the problems are a lot harder than most people originally imagined.
- While failing to "solve the intelligence problem," AI will continue to address smaller *niche* problems. Thus, successful programs for expert systems, natural language, speech recognition, robotics, and vision will achieve that success by limiting themselves to narrow domains. Faster hardware alone will not provide wider coverage.
- Inexpensive hardware for AI will have a dramatic effect on the market for AI, and vice versa. One relevant reminder is the advent of spreadsheets and personal computers. The success of the Apple II computer was largely due to the VisiCalc spreadsheet program. The public is even more ready today to accept an AI program that addresses a real need. However, progress as measured by market acceptance can be illusory. The definition of what constitutes AI changes. In the 1950s, FORTRAN was considered automatic programming—

an AI domain. Today, practically every software product to appear makes some claim to contain AI.

- AI's impact on employment will no doubt be analogous to that of other productivity aids. The obvious comparison is to automation. One can argue that the introduction of computers has brought vast expansion to numerous industries (particularly the computer industry itself), and thus stimulated employment for certain economic sectors. However, it is not hard to find cases to support the other side of the argument. For example, in agriculture, the introduction of modern fertilizers, pesticides, hybridization, and mechanization led to dramatic increases in farm productivity, with concomitant reductions in farm employment. There may be areas of the economy for which AI may have a similar effect. Today's so-called industrial nations are in fact postindustrial. They are service economies, with services accounting for two-thirds of their employment and two-thirds of their GNP. It is therefore likely that AI will make its major impact in the service sectors of the economy, including finance, medicine, and education. A particular contribution of AI to employment should be in improved training, making it easier for people to learn a trade.

It is impossible to axiomatize the world. You cannot predict the future of AI from first principles. What can you do? You can rely on experience. However, experience can be hazy or ambiguous or even contradictory. Reasoning from cases does not guarantee a correct answer, since it is often possible to find compelling precedents that reach the opposite conclusions.

Consider the practice of law. Two opposing attorneys may cite different precedents to support their conflicting claims. The judge must decide which argument is better—which precedent is more relevant. The judge must reason by analogy from the precedents.

Consider the practice of medicine. A physician may have a patient displaying symptoms with two possible diagnoses that require conflicting treatments. The doctor may compare the patient with similar cases that he or she has seen or read about or discussed with colleagues. The doctor reasons by analogy from previous cases.

Finally, consider the actions of a politician—actions that have a direct impact on society. Most political decisions entail choices between conflicting alternatives. There are rarely rules that may be applied with guaranteed success. Usually, it is possible to reason from past cases to support either side of the question. The ultimate outcome of the decision may in fact be less important than the degree to which the politician can justify his or her action by arguing from appropriate precedents.

We suggest that the mode of reasoning by analogy is ubiquitous and that furthermore, it is not without errors. People will make mistakes when they reason from cases. However, we shall see that there is a silver lining: Case-based reasoning may lead to errors, but those errors should result in learning.

SCIENTIFIC GOALS

We view AI as a specific enterprise focused on the human mind. The mind is largely a black box. There is a vast gap between overt, observable cognitive behavior and the microscopic world of neurons. AI provides us with both a tool and a metaphor for examining the mind. The tool is the computer. The metaphor is made explicit in the physical symbol system hypothesis (Newell and Simon, 1981).

As scientists, we seek to understand the nature of understanding; to learn how people learn; to create theories of creativity. We bring to this task a computational bias. We believe that our theories must be computationally plausible—not because our computers must be able to embody the theories, but because people must. To us, psychological validity implies computability. Problems of understanding, learning, and creativity seem to converge in a central issue: how is human memory organized so as to be computationally feasible?

Our theories of learning and understanding have emerged from over 15 years of experience in writing AI programs that model human cognitive processing (Schank and Abelson, 1977; Schank, 1972; Schank, 1982; and Schank, 1986). Our research has demonstrated that learning starts from the failure to understand and that explanation of those failures can lead to important generalizations. Thus, when you encounter a problem in executing a plan that normally works, you notice the plan failure and try to account for it in a way that will let you anticipate that problem in the future, and possibly avoid it. People learn from their mistakes. Understanding a new situation requires incorporating it into your memory. If there is anything to be learned from the new situation, it will require that you recognize the differences in the new case and can account for them.

Much of our current work focuses on the development of a computational model of *explanation* that would allow a computer program not only to make observations about the world, but also to analyze the data and produce a causal hypothesis to account for the data.

The general reasoning process underlying this approach to learning and of understanding is as follows:

1. Detect a failure.
2. Generate questions to examine the failure.
3. Generate explanations to account for the failure.
4. Generalize the questions beyond the current case.
5. Verify the generalization through reminding of past cases.
6. Incorporate the new generalization into memory.

In the absence of failures, new episodes are simply incorporated into memory with their existing exemplars. Learning is minimal in that case. As the number of exemplars increases, the cases become distilled into a rule, as in the earlier example of the observed steady improvement in hardware performance.

In proposing this framework, we cannot of course claim to have solved the problem of learning. We are proposing a hypothesis. As the details of the model become more specific, aspects will be realized in a variety of computer programs. We hope that psychologists will formulate experiments to test particular aspects of the model.

TECHNOLOGICAL GOALS

In many peoples' minds, AI is more technology than science. AI and expert systems are often thought to be one and the same. The goal is to build smart machines. It does not matter if they simulate human cognitive behavior as long as they get the job done. What does the future hold for expert systems and their alternatives?

We must first ask: How is an expert system different from a regular computer program? One simple contrast is disturbing: Expert systems will make mistakes. Now, we all know that other computer programs make mistakes too, but the point is that most computer programs are the embodiment of an *algorithm*—a precise method for performing a procedure. An expert system is not an algorithm. It is a collection of rules, or heuristics, that individually specify the performance of some fraction of a procedure. However, there is nothing to guarantee the correctness of their aggregate behavior. Generally, we prefer algorithms to heuristics. We do not have expert systems for arithmetic or sorting or searching. We have algorithms. We turn to expert systems (and AI in general) when we do not know what the algorithms are for a given problem. Thus, for example, medical diagnosis and investment planning do not lend themselves to algorithmic analysis, so expert systems are the tools of choice. In a sense, it does not matter if an expert system makes mistakes, as long as it performs the task comparably to a human. No doubt the human expert made mistakes as well.

The most common approach in AI for endowing a computer program with expertise is to provide the program with some large set of rules—conditional *test-action* pairs—for solving some particular class of problems. These rule-based expert systems have achieved moderate success in narrow domains, but they fail to capture the most significant aspect of human intelligence: the ability to learn. In particular, a program should learn from experience. A program should adapt to new problems. Here are three specific instances in which rule-based expert systems fail because of their inability to use experiential knowledge.

- *Knowledge acquisition.* To build an expert system, you need an expert and a knowledge engineer. The knowledge engineer tries to discover the rules that the expert uses to solve problems in the given domain. A typical expert system may comprise several hundred such rules. The problem is that it is extremely difficult for experts to express their knowledge as rules. However, a human expert finds it easy to recount episodes about specific cases. This suggests that humans do not in fact encode knowledge as rules, but as cases. Furthermore, it should be easier to extract knowledge from an informant as cases, rather than as rules. Thus, the knowledge-acquisition bottleneck for expert systems could be allayed if the programs could assimilate cases, rather than requiring explicit rules.
- *Reuseability.* If a rule-based expert system is presented with a problem, it may fire dozens or hundreds or thousands of rules and finally come up with an answer. If exactly the same problem is then presented to the program again, the program will again fire the same set of rules and come up with the same answer. This outcome should not be surprising, given what we know about computers. However, given what we know about *people,* we should take pause. A person in that situation would no doubt *remember* having solved the problem before and not have to recalculate the result. A case-based computer system would perform this way by design: each of its own experiences would become part of its knowledge base. Solving an old problem would be easier than solving a new problem. The program would *remember* solving the problem.
- *Robustness.* If a rule-based system is given a problem that ultimately does not match any of its rules, the system must give up. It has no further alternatives. However, a human expert must be able to handle novel situations. In such cases the person would reason by analogy from one or more cases to try to piece together a solution for the unique problem. Similarly, a case-based computer program should be able to reason analogically to formulate an answer.

The technology of expert systems is critically limited by the nature of the rule-based knowledge representation. The central feature of *expertise* is *experience.* An expert is someone who has vast, specialized experience, who has witnessed numerous cases in the domain, and who has generalized this experience to apply it to new situations. When confronted with a problem, the expert is *reminded* of previous, similar problems and their respective resolutions. It may be that the expert has so many exemplars for a given problem that the experiences have been distilled into a general rule to be applied.

In the production system paradigm, the rule is hard-wired into the system. If a rule fails, the system generally requires human intervention to revise the rule. This makes the system more fragile and less robust.

An expert system that can extract information from its experience will be able to grow and acquire knowledge on its own. This is a crucial step for the long-range success of the expert system concept in AI. There are so many tasks to which automated reasoning power might be applied that it is absolutely necessary to develop a mechanism that can assimilate new knowledge directly from experience.

The development of technology for case-based systems has begun. Our previous work has included a variety of programs that reason from specific cases. CYRUS (Kolodner, 1980) used cases to answer questions about former Secretary of State Cyrus Vance. IPP (Lebowitz, 1980) read news stories about terrorist acts and developed its own set of generalizations of terrorist behavior based on these specific cases. JUDGE (Bain, 1986) simulated a judge's sentencing behavior based on prior cases, and CHEF (Hammond, 1986) created new plans (recipes) based on similarities of prior plans to new requirements.

IPP and CHEF both learn from experience. IPP read news stories about terrorism—bombings, kidnappings, shootings. The program started with generic knowledge about terrorist acts and after reading hundreds of stories developed its own set of generalizations about terrorism that it could apply to new stories. For example, when the program reads two stories about IRA terrorism in Northern Ireland, it notices that the victims are establishment, authority figures (policemen and soldiers), and that the terrorists are members of IRA. IPP then reads a third story about a shooting in Northern Ireland, and the program infers that the unidentified gunman is a member of the IRA. The types of generalizations that IPP formed were based on the similarities among stories, not a coherent, causal explanation. IPP would form erroneous generalizations that were based on coincidences. For example, after reading two stories about bombings in a certain country in which two people were killed, IPP would conclude that any bombing in that country would result in the death of exactly two people. We realized that a person would most likely disregard the similarity as a coincidence, rather than a predictable feature.

Unlike IPP, CHEF learned about events in the world based on a causal model. We have recently argued against the adequacy of the purely inductive model of learning, such as is found in IPP, in favor of explanation-based learning, such as is found in CHEF (Schank et al., 1986). CHEF tried to develop explanations for unexpected events, and use its explanations to correct for errors. The CHEF program developed new plans based on its own (simulated) experience in the domain of cooking. When faced with the task of preparing a dish for which it had no appropriate plan (recipe), CHEF would modify an existing plan to fit the new situation and then try to detect and correct any errors that resulted. CHEF would learn from its own mistakes.

Recent work includes the IVY system (Hunter, 1989), which simulates a pathologist diagnosing lung tumor cases. The knowledge base for IVY comprises thousands of cases collected over many decades by Dr. Raymond Yesner, an eminent Yale pathologist specializing in lung cancer. The cases are represented with both patient data and microscopic slide images that are displayed from an interactive videodisk indexed by computer. The IVY system is designed to make diagnoses by reasoning from its experiential knowledge base.

We view these programs as experiments. As we see our technology improve, our experiments can become more sophisticated and exacting. We see some clear trends in this technology.

- The knowledge bases must grow to comprise hundreds, if not thousands, of cases before we can begin to approximate the organizational complexity of human memory even for limited domains. The problems of indexing and search do not exist in a memory with a dozen cases.
- To grow to a large number of cases, the systems must be able to assimilate cases on their own. For example, IPP was able to read news stories about terrorism.
- Eventually, the systems must be able to expand into other domains. People have the valuable ability to generalize their knowledge across domains. That is, they can apply general principles acquired in one setting to a situation involving quite different specific knowledge. This transfer of knowledge across domains remains a significant technological goal.

EDUCATIONAL GOALS

Most educational methods are passive. Reading a book is passive. Listening to a lecture is passive. The student is not directly involved: there is very little interaction. By contrast, in learning motor skills, we clearly need interactive feedback. It is hard, if not impossible, to learn to play the piano or drive a car or play tennis simply by reading a book or listening to someone else explain how they do it. Similarly, cognitive skills benefit from the same interactive practice and feedback.

Experiential or case-based learning is ubiquitous in everyday life, but is rarely seen in the schools: its power remains largely untapped in formal education. There are few applications outside certain professional schools, and there are no appropriate tools available for teaching from cases. Traditional classroom instruction often involves teaching rules and formulas that have little applicability elsewhere and are quickly forgotten because they are not grounded in personal experience. In contrast, experiential learning allows students to formulate their own rules in response to some specific case. Students learn the limits of these rules by applying them in new circumstances, making the rules more appropriate and more memorable.

At Yale, we developed a prototype ICAI system for teaching decision making (Farrell, 1988). The system was designed to foster learning from experience by letting students explore historical cases, confronting them with decisions that require use of past experience, and challenging them to make broad generalizations based on their successes and failures. The computer program contained a number of related episodes. The program controlled an interactive videodisk containing numerous images from newsreels, magazines, and newspapers to illustrate and amplify the cases. Our research suggests that the more vivid and rich the experience, the more readily will the student learn. Thus, we want the learning episode itself to be memorable, providing a wide variety of converging experiences. The student's interaction with the system then becomes a significant case that provides a ready index to the historical cases.

Our approach resembles the case-based method used widely in law and business schools. The methods we propose should be applicable to teaching management science, economics, public policy, and a host of other domains where experience plays a major role in performing a task. This approach has recently been advocated specifically for problems in national decision making by Neustadt and May (1986).

Moreover, this same approach should apply to traditional scientific disciplines as well. In sciences, such as physics or biology, the basic task is to formulate hypotheses to explain observed phenomena. New hypotheses are tested against experience—they must be consistent with our previous observations. Competing hypotheses may cause us to reexamine past experiences to look for additional features. Or a new hypothesis may lead us to conduct a new experiment or make a new observation to test our theory. We argue that this process of hypothesis formation and testing that forms the foundation of the scientific method is, in fact, an instance of case-based reasoning.

A failure is seen to be an anomaly—something that requires explanation. Hence, a teaching system should force students to fail by putting them in challenging situations. Our program will place students at points in history where important decisions must be made. It will let the student try a plan and see the outcome. Because the historical cases used involve difficult decisions, the students will often fail. We believe that putting students in situations where they will fail is critical to the learning process. Part of the task of building our system is having the program pick cases that will challenge students just enough to make them fail, but not so much that they don't understand why they failed after seeing the outcome.

If a student is to learn from experience, he or she should be encouraged to follow up on his or her failures by explaining what went wrong. In our program, students will be asked to evaluate the outcomes of their decisions and then decide what they would do differently in the future. This will force them to explain what went wrong with an eye toward formulating a decision-making rule for the future. Learning occurs when these explanations, constructed from particular cases, are generalized to be more widely applicable. This generalization is mediated by considering past cases to which these generalizations might apply. Our program will confront students with past cases during this generalization phase, forcing them to consider similarities and differences from the current case and pushing them to formulate a more appropriate decision-making rule.

We view this program as a prototype for a new instructional paradigm with the following principles:

- Teach questions, not answers. We want the students to learn how to think critically about new problems, not simply to memorize answers. In most real-world situations, there are no right answers.
- Teach from examples. Take advantage of the fact that people are well-suited for learning from experience. It is a natural form of cognition.
- Remove the stigma from failure. To learn from experience, the student must first learn not to be afraid to fail. In this paradigm, failure is good in so far as the student can learn from his or her mistakes.

We believe that case-based AI technology is well-suited for building such systems and that they can be applied to a broad range of subjects.

CONCLUSION

We are engaged in a study of human thought, in particular, how people understand and learn about the world. We want to know how people think for several reasons. First, as scientists, we find the question of intrinsic interest. Second, we hope to use our models of learning to raise the standards and improve the capabilities of the next generation of AI applications. We want programs that can learn and adapt. Finally, we want to create a new paradigm for instruction that is grounded in a natural method of learning. We envision an educational system in which a student's natural ability to learn from experience becomes an advantage, not an impediment.

In short, we believe that the study of learning within the framework of artificial intelligence will have a significant impact on science, technology, and education.

Acknowledgments

We would like to thank Larry Birnbaum for his helpful suggestions with this paper.

REFERENCES

Bain, W. M. 1986. Case-based Reasoning: A Computer Model of Subjective Assessment. Ph.D. dissertation, Yale University, Technical Report 470.

Farrell, R. 1988. Facilitating Self-education by Questioning Assumptive Reasoning. In *Proc. of the Seventh*

Nat. Conf. on Artificial Intelligence, pp. 2–6, St. Paul, Minn., The American Association for Artificial Intelligence.

Hammond, K. J. 1986. Case-based Planning: An Integrated Theory of Planning, Learning and Memory. Ph.D. dissertation, Yale University, Technical Report 488.

Hunter, L. 1989. Knowledge Acquisition Planning: Gaining Expertise through Experience. Ph.D. dissertation, Yale University.

Kolodner, J. L. 1980. Retrieval and Organizational Strategies in Conceptual Memory: A Computer Model. Ph.D. dissertation, Yale University, Technical Report 187.

Kolodner, J. L., Simpson, R. L., and Sycra-Cyranski, K. 1985. A Process Model of Case-Based Reasoning in Problem-Solving. In *Proc. of the Ninth Int. Joint Conf. on Artificial Intelligence*, Los Angeles, Calif., IJCAI.

Lebowitz, M. 1980. Generalization and Memory in an Integrated Understanding System. Ph.D. dissertation, Yale University, Technical Report 186.

Neustadt, R. E., and May, E. R. 1986. *Thinking in Time: The Uses of History for Decision Makers*. New York: The Free Press.

Newell, A., and Simon, H. A. 1981. Computer Science as Empirical Inquiry: Symbols and Search. In *Mind Design*, ed. J. Haugeland, pp. 35–66. Montgomery, Vt.: Bradford Books.

Schank, R. C. 1972. Conceptual Dependency: A Theory of Natural Language Understanding. *Cognitive Psychology* 3(4):552–631.

Schank, R. C. 1982. *Dynamic Memory: A Theory of Learning in Computers and People*. Cambridge: Cambridge University Press.

Schank, R. C. 1986. *Explanation Patterns: Understanding Mechanically and Creatively*. Hillsdale, N.J.: Erlbaum.

Schank, R. C., and Abelson, R. 1977. *Scripts, Plans, Goals and Understanding*. Hillsdale, N.J.: Erlbaum.

Schank, R. C., Collins, G., and Hunter, L. 1986. Transcending Inductive Category Formation in Learning. *Behavioral and Brain Sciences* 9(4).

Simpson, R. L. 1985. A Computer Model of Case-based Reasoning in Problem-solving: An Investigation in the Domain of Dispute Mediation. Ph.D. dissertation, School of Information and Computer Science, Georgia Institute of Technology.