Intelligent Agents as Cognitive Tools for Education

Baylor, A. L. (1999). Intelligent agents as cognitive tools for education. *Educational Technology, 39*(2), 36-40.

Abstract

This paper considers the role of intelligent agents as cognitive tools for human learning. With significant theoretical support, the potential of intelligent agents as cognitive tools is discussed together with issues regarding learning *by* the intelligent agents. In terms of educational potential for intelligent agents as cognitive tools, the following three functions for development are discussed: 1) managing information overload; 2) serving as a pedagogical expert; and 3) creating programming environments for the learner.

What are intelligent agents?

As described by Roesler and Hawkins (1994), intelligent agents are independent computer programs operating within software environments such as operating systems, databases, or computer networks. One typical use of intelligent agents is to help users with routine computer tasks, while still accommodating individual habits. For example, an intelligent agent could search on the Internet to find a car matching a list of criteria, like tracking down the best price for purchasing a purple Honda Civic 1994 DX. In this case, the intelligent agent supports users in obtaining additional information so that they do not have to obtain it for themselves. Another simple intelligent agent is in Microsoft Word for Office 97, where a personified paper clip "Clippit" makes suggestions on better use of the word processor based on user actions. For example, when a user types the word "Dear," it assumes that s/he is about to write a letter and prompts with information regarding relevant templates.

This technology combines artificial intelligence (reasoning, planning, natural language processing, etc.) and system development techniques (object-oriented programming, scripting languages, human-machine interface, distributed processing, etc.). Bradshaw (1997) cites the distinguishing characteristics of intelligent agents to be in managing cooperation among distributed programs and/or other agents, providing intelligent assistance to users when traditional interfaces are insufficient, and enabling more humanlike interaction. From an educational vantage point, a better description might be that intelligent agents are computer programs that simulate a human relationship by doing something that another person could otherwise do for you (Seiker, 1994, p.92). Intelligent agents have been predicted to be the most important computing paradigm in the next 10 years. Janca (1995) claims that by the year 2000 every significant application will have some form of agent functionality. Consequently, the use of intelligent agents for educational purposes is of interest.

Intelligent agents as cognitive tools

Negroponte (1970) described intelligent agents as electronic "butlers," performing such tasks as filtering email, scheduling appointments, informing regarding investments, and making travel arrangements. Negroponte's (1997) more modern metaphor for the intelligent agent is as a "digital sister-in-law." As he explains, when he wants to go the movies, rather than read reviews, he asks his sister-in-law, as she is both an expert on movies and an expert on him! In this paper, the interest is in extending the intelligent agent's reach beyond that of a "butler" or "sister-in-law" so as to serve as a coach or *cognitive tool* in educational contexts. Using technology as a cognitive tool is stated by Jonassen (1995) as follows:

> Computer technologies as cognitive tools represent a significant departure from traditional conceptions of technologies. In cognitive tools, information and intelligence is not encoded in the educational communications, which are designed to efficiently transmit that knowledge to the learners. Rather than using technologies by educational communications specialists to constrain the learners' learning processes through prescribed communications and interactions, the technologies are taken away from the specialists and given to the learner to use as media for representing and expressing what they know.

In considering the use of intelligent systems for education, there are generally two camps (Lajoie & Derry, 1993): 1) those who promote using the system as a cognitive tool to stimulate the student to monitor and diagnose performance; and, 2) those who promote using the model building approach to use the system as an intelligent tutor. In this paper it will be argued that it is more appropriate for the agent to function as an intelligent cognitive tool than as an intelligent tutor. A critical issue in terms of educational value is in moderating between the agent taking over thinking for the student with the agent training the

student to think more effectively. Salomon (1993a) refers to this as the difference between the effects "of" and "with" technology, with effects "with" technology being more desirable. In this sense, the computer technology in and of itself is of little interest whereas what activities it affords *are* of interest.

Cognitive tools (e.g. Lajoie & Derry, 1993) are mental and computational devices that support, guide, and extend the thinking processes of their students (Derry, 1990). In terms of educational psychological theory, the concept of distributed cognitions (Salomon, 1993b) readily applies to intelligent agents since they could be used to serve as extensions of a person's intellectual capacity. Similarly, Vygotsky (1962) defines a person's zone of proximal development to be the limit for his/her ability to imitate processes demonstrated by others. By extending the cognitive capabilities of the person, intelligent agents as cognitive tools could serve to thus augment a person's zone of proximal development.

For an intelligent agent to best facilitate the learning process, it is necessary for the student to actively use the agent as a cognitive tool rather than passively letting the agent retrieve information. In this way the intelligent agent provides an environment where the learner must think harder and more deeply about the content, using the agent as a natural cognitive extension. In this manner, the intelligent agent could serve as a technological "reciprocal teacher" (e.g. Palinscar & Brown, 1984), prompting the individual to engage in analysis of his/her own cognitive processes. This use would serve to encourage the individual to assess what cognitive strategies are being used, similar to Salomon's pedagogic computer program, the Writing Partner (Salomon, 1993a), which asks the learner intelligent questions through the writing process.

Intelligent agents can learn

Aside from serving as a cognitive tool, an additional feature is that intelligent agents can be developed to adapt behavior based on the student's history of performances. In this way, the agents could analyze the student's approach to a task, build a database of past activities, and provide suggestions of better strategies. In effect, the agents learn the way that the student learns and could infer expected behavior and advise the student accordingly. In terms of helping the student learn, the agent can coach, tutor, and/or provide help (Laurel, 1997). For example, the agent could note that the learner consistently performs a library search using very specific word descriptors. As a result, the learner “misses” the other potential matches that do not contain those exact words. In this case, the agent could suggest that the learner enter a more general term the next time that the learner starts a search.

Maes (1997) takes a machine learning approach to intelligent agent development inspired by the metaphor of a personal assistant. For Maes, the key issues for the intelligent personal assistant are trust and competence. In her approach the agent gradually develops its abilities so that the user is also given time to gradually build up a model of how the agent makes decisions, thereby improving trust. The agent acquires competence from four sources: 1) agent monitors user, noticing his/her behavior; 2) agent provides direct and indirect user feedback; 3) user can train the agent from examples given explicitly by the user; and, 4) agent can ask for advice from agents that assist others with same task. She claims another benefit of this learning approach is that it requires less work from end-user and application developer.

According to the design principle of semiformal systems (Malone, Lai & Grant, 1997), the reasoning of the agent should be visible to the user. Rather than creating intelligent agents whose operations are “black boxes” designers should try to create “glass boxes” where the essential elements of the agents’ reasoning can be seen and modified by users. (p. 118) Along this line, a serious risk for “learning agents” is that agents will infer incorrect rules (or fail to infer correct ones) when users could have easily described the rules they actually wanted to use. As Erikson (1997) points out, the user needs understanding of what happened and why. As he explains:

> Consider an intelligent tutoring system that is teaching introductory physics to a teenager. Suppose the system notices that the student learns best when information is presented as diagrams and adapts its presentation appropriately. But even as the system is watching for events, interpreting them, and adjusting its actions, so is the student watching the system, and trying to interpret what the system is doing. Suppose that after a while the student notices that the presentation consists of diagrams rather than equations: it is likely that the student will wonder why: ?Does the system think I’m stupid? If I start to do better, will it present me with equations again?’ There is no guarantee that

> the students' interpretations will correspond with the system's. How can such potentially negative misunderstandings on the user's part be minimized? (p. 83)

Even if the reasoning of the intelligent agent is visible to the user, there are three other critical issues regarding learning by the agent. First, from a theoretical perspective, intelligent agents essentially require metaknowledge, or knowledge about the knowledge they are working with, which is a very difficult requirement. Second, the agent needs "common-sense" reasoning (Kearsley, 1993) where it can make reasonable inferences about what the learner wants and how to accomplish it. Third, with both of these considerations, even in the simplest tasks, it is often important that requests are not interpreted literally but in terms of their intent. Thus, an intelligent agent must be able to understand the consequences of actions and compare them to the desired outcomes. Understanding intentionality and being able to infer consequences are sophisticated intellectual skills? making the design of intelligent agents a very difficult challenge. To add to the complexity in terms of the *amount* of artificial intelligence that should be used by an agent, it is important to consider that more intelligence is not necessarily better from a pedagogical perspective (e.g., Salomon, Perkins & Globerson, 1991).

Educational Applications

Considering intelligent agents as cognitive tools suggests educational applications in the three following areas, each of which will be described in the next section: 1) managing large amounts of information; 2) serving as a pedagogical expert; and, 3) creating programming environments for the learner.

Managing information overload

An important way that an intelligent agent could support higher order thinking skills is by reducing information overload. By filtering and selecting information, or handling administrative activities such as scheduling meetings, the agent can free a person's working memory to allow for higher order processing. As autonomous agents, intelligent agents can serve as conduits of information on the Internet. One significant potential use in education is in terms of information management: to filter the great quantities of information available to a learner, customizing it to the learner's specifications. In this manner, the learner would prescribe what s/he wanted the agent to find, and then the agent would work independently and report its findings back to the learner. Kahn and Vinton (1988) called such agents "knowbots," which navigate the information source. Agents will be critical to deal with the wealth of information available on the Internet and electronic databases.

In this sense one goal of the intelligent agent is to find and present information according to the particular interests or needs of the learner. As stated by Kearsley (1993), agents are able to produce multiple representations of the same information. For example an agent could compose an explanation of chemical bonding in terms of the underlying mathematics or by showing diagrams of molecules. Laurel (1997) more strongly asserts that for an agent to be deemed competent and responsive by the user, the agent must possess or be able to generate multiple representations of information. Overall, this is a quality shown by any good teacher: the ability to present a subject matter in a manner that is relevant to the student's interests or ability level. In this way the intelligent agent can customize information for the individual's learning situation.

In terms of designing the agent specifically for coordinating and sharing information, Malone, Lai & Grant (1997) cite two key properties: semiformal systems and radical tailorability. First, according to the design principle of semiformal systems, the reasoning of the agent should be visible to and be modifiable by the user. This provision eliminates some of the risk that agents will infer incorrect rules (or fail to infer correct ones) when users could have easily described the rules they actually wanted to use. A semiformal approach to designing intelligent agents would also suggest that any attempts to have agents automatically "learn" from observing users' behavior should occur only *after* the system already provides a way for users to directly specify what they want (p. 119). Second, according to the design principle of radical tailorability, the user should be able to change the system without leaving the application domain and work in a separate "programming" domain. This allows the user to assert more control with obtaining the information desired. The term "radical" implies that very large changes can be made (i.e., changes more significant than the preferences in a word processing program). In this way, radically tailorable systems can help reduce the "cognitive distance" between using an application and designing it

(Hutchins, Hollan, and Norman, 1986). These two principles of semiformal systems and radical tailorability enhance user control over managing and coordinating information.

Serving as a pedagogical expert

A different role of the intelligent agent is to serve as a pedagogical expert where it can monitor and evaluate the timing and implementation of teaching interventions (e.g. help, feedback). From a cognitive tool perspective, this relationship of the intelligent agent to the student can be described as a cognitive apprenticeship, where the student improves his/her performance while working with the more expert performer: the intelligent agent. In support of this feature, Collins & Brown (1987) suggest that students may learn best in environments including modeling and coaching of formative skills. A favorable situation with intelligent agents might be that as the student gains expertise, the agent would fade and allow for more student initiative. Additionally, by drawing from human research on expert tutoring, the intelligent agent could mentor flexibly, as a human mentor would. However, there are significant difficulties in developing pedagogical expertise in an intelligent agent. As McArthur, Lewis and Bishay (1993) state, the pedagogical component of intelligent systems receives relatively little mention with current systems demonstrating little pedagogical expertise. As they suggest, most intelligent tutoring systems are constrained to a single method of teaching and learning, while truly expert human tutors can adopt different methods.

Even so, intelligent agents can successfully serve as coaches or advisors, according to Kearsley (1993). COACH is an intelligent agent system by Seiker (1994) that records learner experience to create personalized learner help for LISP with an adaptive interactive help system. As Seiker describes, (p. 92) "Just as a football coach will stand on the sidelines and encourage, cajole or reprimand, so COACH is an advisory system that does not interfere with the learner's actions but comments opportunistically to help the learner along." Like COACH, an intelligent agent could serve as an advisory-style agent that builds a learner relationship with the explicit goal of educating the individual (Seiker, p. 93). Another example is a proposed agent (called the Personal Intelligent Mentor (PIM)) that has special potential for facilitating learners' metacognitive processing (Baylor & Kozbe, 1998a), which is a difficult area for traditional instruction. Specifically, the PIM serves as a metacognitive "coach," encouraging students to think about their cognitive processing and implicitly learn to model the agent's learning strategies.

An important consideration in terms of feedback is that the intelligent agent should not provide too many insights and thereby annoy the student. As Negroponte (1997) suggests, the human act of winking can connote a lot of information to others simply in the *lack* of information. This sort of familiarly is needed for the pedagogical agent to avoid relentless explicitness. To address this issue, part of the pedagogical task should include the monitoring of the timing and implementation of the advisements. With the principle of minimal help as the default, there could also be the possibility for the student to select a feedback option depending on the amount of structure, interaction, and feedback s/he desires when problem-solving.

Another critical area, described in more detail in Baylor (1998b) is that of the social relationship between the agent and learner, including the agent's persona. This instructional relationship of the learner and agent requires the intelligent agent to be perceived by the learner as trustworthy and competent (Maes, 1997), empathetic (Lepper & Chabay, 1987), responsive (Laurel, 1997), demonstrating emotion (Bates, 1994), honest, and cooperative while providing feedback.

Creating environments for programming by the learner

Another educational use of intelligent agents is for the learner to create an intelligent agent as a learning experience. Kearsley (1993) proposes the possibility of building agents into a programming environment so that students can create them directly. In this manner, the programming of intelligent agents could follow what happened with the LOGO programming language in terms of incorporating elements of computer languages into a thinking tool for students. In this approach, students would need to learn the concept of instructing an agent to carry out tasks, which is a complex communication task in itself. In the Vivarium project, promoted by Alan Kay, children develop agents in this manner in an object-oriented programming environment. Smith, Cypher & Spohrer (1997) applied graphical user interface (GUI) principles to the process of programming through the creation of KidSim. KidSim is a tool kit for kids and nonprogramming adults to construct and modify simulations by programming their behavior. KidSim

agents are objects, similar to those in Logo Microworlds, but are created through a very different manner using two GUI techniques of graphical rewrite and programming by demonstration.

Beyond intelligent tutoring systems

Kearsley (1993) attests that "intelligent agents embrace many of the elements of intelligent tutoring systems but go considerably beyond."(p. 298) For example, the capability to create a model of the student is an important attribute of most tutoring systems and agents likewise need to have a very robust model of the learner if they are to carry out actions that depend upon a knowledge of personal preferences or interests. However, in a tutoring system, as Kearsley explains, the student model is a representation of what has been learned, whereas in an agent, the model covers the entire personality of the learner. Such broader coverage would remedy the narrow focus of tutoring systems that excludes important elements in learning such as motivation, intentions, and cognitive styles. Similarly, he expresses the need for agents to create knowledge networks at a macroscopic level to deal with the whole student and learning in general, unlike intelligent tutoring systems that modeled such roles at very microscopic level for particular instructional tasks.

Conclusion: Intelligent agents' role in education

As Kearsley (1993) suggests, intelligent agents introduce a new paradigm for instruction that is based on the concept of shared abilities and cooperative learning between humans and computers. Aside from the difficulties in actually constructing intelligent agents for education, exploring the development of artificially intelligent agents is a worthy task in that it helps further our theoretical understanding of instruction. While the focus of this paper has been on intelligent agents as cognitive tools for learning, intelligent agents can also apply similar cognitive strategies for training (Baylor, in press) in industry. Consequently, results of theoretical and technological progress in this area also provides valuable information for the development of individualized instructional systems and electronic performance support systems (EPSS).

References

Bates, J. (1994). The role of emotion in believable agents. *Communications of the ACM, 37*(7), 122-125.

Baylor, A. L. & Kozbe, B. (1998). A Personal Intelligent Mentor for Promoting Metacognition in Solving Logic Word Puzzles. Paper presented at workshop "Current Trends and Applications of Artificial Intelligence in Education," The Fourth World Congress on Expert Systems, Mexico City, Mexico, 16-20 March 1998.

Baylor, A.L. (1998). Intelligent Agents as Mentors. Unpublished manuscript.

Baylor, A. L. (1998). Cognitive strategies for training with technology. Accepted for publication in *TechTrends.*

Bradshaw, J.M. (1997). An introduction to software agents. In *Software Agents,* edited by J.M. Bradshaw (pp. 3-46). Menlo Park, CA: MIT Press.

Collins, A. & Brown, J.S. (1987). The computer as a tool for learning through reflection. In H. Mandl and A. Lesgold (Eds.), Learning issues for intelligent tutoring systems. New York: Springer-Verlag.

Derry, S. (1990). Flexible cognitive tools for problem solving instruction. Paper presented at the annual meeting of the American Educational Research Association, Boston, MA, April 16-20.

Erickson, T. (1997). Designing agents as if people mattered. In *Software Agents,* edited by J.M. Bradshaw (pp. 79-96). Menlo Park, CA: MIT Press.

Hutchins, E.L., Hollan, J. D., and Norman, D. A. (1986). Direct manipulation interfaces. In *User-centered system design,* Eds. D. Norman and S. Draper (pp. 87-124). Hillsdale, NJ: Lawrence Erlbaum.

Janca, P. (1995). Pragmatic application of information agents: theory and practice. *BIS Strategic Decisions.*

Jonassen, D. H. (1995). Computers as Cognitive Tools: Learning with Technology, Not from Technology. *Journal of Computing in Higher Education, 6*(2), 40-73.

Kahn, R. & Vinton (1988). *An open architecture for a digital library system and a plan for its development.* Corporation for National Research Initiatives Technical Report, March 1988.

Kearsley, G. (1993). Intelligent agents and instructional systems: Implications of a new paradigm. *Journal of Artificial Intelligence and Education, 4* (4), 295-304.

Lajoie, S. & Derry S. (1993) *Computers as cognitive tools.* Hillsdale: Lawrence Earlbaum Associates.

Laurel, B. (1997). Interface agents: Metaphors with character. In *Software Agents,* edited by J.M. Bradshaw (pp. 67-78). Menlo Park, CA: MIT Press.

Lepper, M.R. & Chabay, R.W. (1987). Socializing the intelligent tutor: bringing empathy to computer tutors. In H. Mandl and A. Lesgold (Eds.), *Learning issues for intelligent tutoring systems*. (pp. 242-257) New York: Springer-Verlag.

Maes, P. (1997). Agents that reduce work and information overload. In *Software Agents,* edited by J.M. Bradshaw (pp. 145-164). Menlo Park, CA: MIT Press.

McArthur, D., Lewis, M., & Bishay, M. (1993, November). The roles of artificial intelligence in education: Current progress and future prospects. RAND, Santa Monica, CA, DRU-472-NFS.

Malone, T.W., Lai, K-Y, & Grant, K.R. (1997). Agents for information sharing and coordination: A history and some reflections. In *Software Agents,* edited by J.M. Bradshaw (pp. 109-144). Menlo Park, CA: MIT Press.

Negroponte, N. (1970). *The Architecture Machine: Towards a More Human Environment.* MIT Press.

Negroponte, N. (1997). Agents: From direct manipulation to delegation. In *Software Agents,* edited by J.M. Bradshaw (pp. 57-66). Menlo Park, CA: MIT Press.

Palinscar, A., & Brown, A. (1984). Reciprocal teaching of comprehension-fostering and comprehension-monitoring activities. *Cognition and Instruction, 1(2),* 117-175.

Roesler, M. & Hawkins. D. (1994). Intelligent Agents: Software servants for an electronic information world (and more!). *Online*, 19-32.

Salomon, G. (1993). On the nature of pedagogic computer tools: The case of the writing partner. *Computers as cognitive tools.* (pp. 179-196). Hillsdale: Lawrence Earlbaum Associates.

Salomon, G. (1993). *Distributed cognitions: Psychological and educational considerations.* Cambridge: Cambridge University Press.

Salomon, G., Perkins, D.N., & Globerson, T. (1991). Partners in cognition: extending human intelligence with intelligent technlogies. *Educational Research, 20* (3), 2-9.

Seiker, T. (1994). Coach: A teaching agent that learns. *Communications of the ACM*, 37(7), 92-99.

Smith, D.C., Cypher, A., & Spohrer, J. (1997). KidSim: Programming agents without a programming language. In *Software Agents,* edited by J.M. Bradshaw (pp. 165-190). Menlo Park, CA: MIT Press.

Vygotsky, L. (1962). *Thought and language.* Cambridge: Massachusetts Institute of Technology. (Original work published 1934)