Using AI to Teach AI: Lessons from an Online AI Class

Ashok K. Goel, David A. Joyner

■ *In fall 2014, we launched a foundational course in artificial intelligence (CS7637: Knowledge-Based AI) as part of Georgia Institute of Technology's Online Master of Science in Computer Science program. We incorporated principles and practices from the cognitive and learning sciences into the development of the online AI course. We also integrated AI techniques into the instruction of the course, including embedding 100 highly focused intelligent tutoring agents in the video lessons. By now, more than 2000 students have taken the course. Evaluations have indicated that OMSCS students enjoy the course compared to traditional courses, and more importantly, that online students have matched residential students' performance on the same assessments. In this article, we present the design, delivery, and evaluation of the course, focusing on the use of AI for teaching AI. We also discuss lessons we learned for scaling the teaching and learning of AI.*

AI is in vogue again. As a result the demand for AI courses and programs is growing in universities and colleges across the world. This presents an opportunity for spreading knowledge of AI globally. Some of the increase in demand comes from industry where many IT professionals want to renew their knowledge of AI or learn about it for the first time. This affords an opportunity to influence the practice of AI in the real world. But these opportunities also pose major challenges. How can we satisfy the rapidly growing desire for learning about AI? How can we scale learning of AI so that it is repeatable and testable? How can we ensure that the quality of learning AI at scale is comparable to that in small residential classes?

Recent trends in computing technology provide new affordances for both education in AI and AI in education. On one hand, the ubiquity of the Internet and the rise of cloud computing have enabled scaling for teaching almost any topic to large segments of the world's population. This has led to the development of numerous massive open online courses (MOOCs). The successes of Peter Norvig and Sebastian Thrun's MOOC, Introduction to Artificial Intelligence, and

  ISSN 0738-4602

Andrew Ng's MOOC on machine learning, both launched at Stanford University in 2011, are well known (for example, Leckart [2012], Raith [2011]). On the other hand, cognitive systems research on AI in education over the last few decades has developed human-centered AI techniques to personalize student learning and improve learning outcomes. These techniques are often embodied in intelligent tutoring systems and intelligent learning environments (for example, Azevedo and Aleven [2013]; Koedinger and Corbett 2006; Jonassen, Peck, and Wilson 1999; Sleeman and Brown 1982). Thus, at least in principle, we now have computing technology for scaling teaching as well as cognitive technology for supporting and assessing personalized learning.

In January 2014, Georgia Institute of Technology inaugurated its fully accredited online Master of Science in Computer Science (OMSCS) program. In August 2014, we launched the first foundational course in AI, CS7637: Knowledge-Based AI (KBAI), as part of the program. As a foundational course, the material presupposed no prior experience with artificial intelligence; the only prerequisites are those for admission into the program, including literacy in English and training in computer programming. From the beginning, we adopted the methodology of design-based research in developing the course, incorporated lessons from cognitive and learning sciences in the design of the course, and integrated AI techniques and tools for teaching AI (Goel and Joyner 2016a). In this article, we present the design, delivery, and evaluation of the course, focusing on the use of AI for teaching AI. We also discuss lessons we learned for scaling the teaching and learning of AI.

The Georgia Tech OMSCS Program

Georgia Tech launched its online OMSCS program[1] in January 2014. The video lessons for the OMSCS courses are delivered by the online education startup Udacity.[2] The OMSCS program currently has about 4000 students, an order of magnitude more than the number of students in the Georgia Tech residential MS in CS program, and now is the largest MS in CS program in the United States (Goodman, Melkers, and Pallais 2016). However, while the residential degree costs several tens of thousands of dollars, the OMSCS program charges only $170 per credit hour and thus costs only several thousand dollars, an order of magnitude less than the residential program.

The goal of the OMSCS program is to offer the same courses online that we offer to residential MS students, and with the same depth, substance, and rigor. Students take the same classes and complete the same assessments as residential students, receive grades from the same graders, and must meet the same requirements for graduation. The online students interact with the professor and the teaching assistants during virtual office hours and on web-based discussion forums. The video lessons are created specifically for the online program; while many online programs operate by recording professors live in residential classrooms, all OMSCS material is custom-produced for the program. The video lessons and the class forums together form the virtual classroom (Joyner, Goel, and Isbell 2016).

Knowledge-Based AI (CS7637)

It was within the OMSCS program that in January 2014 we began work on an online version of CS7637: Knowledge-Based AI. Ashok Goel, the first author of this article, was the instructor for the course after creating and teaching it on campus for some 15 years. David Joyner, the second author, was the course developer for the course after previously taking the residential knowledge-based artificial intelligence (KBAI) class one year, and working as a teaching assistant (TA) for it in another year. He also completed his Ph.D. in human-centered computing with Goel. This extant working relationship between the two proved highly valuable in developing the online KBAI course in 2014.

The KBAI class focuses on the "cognitive systems school of AI" (Langley 2012) that we characterize as creating human-level, humanlike, and human-centered AI (Goel and Davies 2011). The KBAI class adopts a design stance toward learning about AI (Goel 1994), and thus much of the learning is organized around intensive design and programming projects that build on one another. The design for the online KBAI class follows a four-tiered learning hierarchy consisting of learning goals, outcomes, assessments, and strategies. Learning goals represent what we expect students to know by the end of the course; outcomes describe what we expect them to be able to do in terms we can measure; assessments provide mechanisms for evaluating their achievement of the outcomes; and strategies prescribe methods of ensuring they accomplish the goals and outcomes, thus succeeding on the assessments.

At a high level, the goals of the class were to understand the tasks that KBAI addresses; the methods it employs to address those tasks; the systems that comprise those methods and tasks; and the relationship between creating those systems and understanding human cognition. To demonstrate mastery of these learning goals, students build systems that address complex problems, and reflect on the relationship between those systems and human cognition. A full articulation of the class's goals, outcomes, assessments, and strategies can be found in our paper An Experiment in Teaching Cognitive Systems Online (Goel and Joyner 2016a).

Design of the Online Course

The online KBAI course comprises 26 lessons on the following topics: (1) introduction to the course, (2) introduction to KBAI, (3) semantic networks, (4) generate and test, (5) means-ends analysis and problem reduction, (6) production systems, (7) frames, (8) learning by storing cases, (9) case-based reasoning, (10) incremental concept learning, (11) classification, (12) logic, (13) planning, (14) understanding, (15) commonsense reasoning, (16) scripts, (17) explanation-based learning, (18) analogical reasoning, (19) generalization and version spaces, (20) constraint propagation, (21) configuration, (22) diagnosis, (23) learning by correcting mistakes, (24) metareasoning, (25) advanced topics, and (26) course wrap-up. The lessons vary in length based on the topic (one of the advantages of preparing the class in this medium), but average approximately one hour per lesson when including the time students spend completing the interactive exercises in each lesson. The videos of all 26 lessons are now available freely through Udacity.2

Ou, Goel, Joyner, and Haynes (Ou et al. 2016) provide an analysis of the student perceptions of the video lessons. During the first offering of CS7637 in the fall 2014 term, only online students had access to these materials; however, the visuals and exercises produced for the online course were reused as the materials for the residential section, with the same structure for the online and residential classes. In the next two offerings of the residential class in the fall 2015 and fall 2016 terms, residential students were also provided access to the online lecture materials as part of experiments in flipped classrooms and blended learning.

The recommended readings came from several textbooks, including Winston (1993), Stefik (1995), Nilsson (1998), and Russell and Norvig (2009). In addition, we included several optional readings on selected topics in cognitive systems such as Lehman, Laird, and Rosenbloom (2006) on the SOAR cognitive architecture. While the course does not teach AI programming, it provides access to AI programming resources such as the reimplementation of several classic AI systems in Python (Connelly and Goel 2013) described in Norvig (1992).

Development and Delivery

Development of the online KBAI course began in February of 2014 with an intense two-day boot camp at Udacity and ran through the launch of the course in August 2014. We estimate that during this six-month period Joyner spent approximately 750 to 800 hours of his time and Goel spent about 200 to 250 hours on the course development. This investment of time was needed because we developed all the videos from scratch and specifically for the online course. The paper by Goel and Joyner (2016a) provide more details of the process of development.

We have offered the online KBAI course each fall, spring, and summer term since the fall 2014 term. To date we have offered the course eight times so far. Enrollment in the class has varied from 200 to 400 students per term; thus, at this writing more than 2000 students have taken the course. The teaching staff consists of the instructor of record, a head teaching assistant (TA), and an additional TA for every 50 students that enroll in the course. Each of the TAs work for about 20 hours per week; this results in each student receiving roughly 7 hours of dedicated TA time per semester. In the KBAI course, the TAs are primarily responsible for grading assignments, while the instructor and head TA take care of interacting with students on the forum and organizing the remaining elements of class administration.

One of the major lessons we have learned is that delivery of the online KBAI class is as important to student learning as developing the video lectures. A common misconception about online learning appears to be that the video lessons are the online equivalent of the traditional classroom for residential students. However, we quickly realized that the video lessons were more like a textbook for the online class, and that the true online classroom is in the discussion forum. It is the forum that replicates most activities that happen in a physical classroom, such as class announcements, discussions, student collaboration, and instructional support through question answering. The discussion forum's asynchronous, persistent, and self-documentation nature, however, fundamentally change how the discussions unfold and the instructional support they require (Joyner, Goel, and Isbell 2016), an observation confirmed and repeated by instructors of other classes in the program (Carey 2016). Generally, we and other instructors have observed that the online experience can be more richly interactive than the residential experience, but this requires properly understanding the ideal roles for the video material and the online forum.

Evaluation

We concentrate on two variables in evaluating the online KBAI course: class assessment outcomes and student experience. During semesters in which the residential section of the KBAI class is offered simultaneously, we approach evaluating learning outcomes using a quasi-experimental approach. Residential and online students are given the same assessments on the same schedule, and they are evaluated by the same graders. Graders evaluate the assignments blind as to whether a given student is enrolled online or residentially. Thus, we compare online and residential students' grades on the assessments to ensure that the learning outcomes online are at least as good as those on campus. As table 1 illustrates, in fall 2014 we found that the online students outperformed residential students on all 14 assessments, with statistical significance on 7 of those assessments. There may be multiple explanations of this phenomenon. On one hand, it may be

Item	Max	OMSCS (Mean)	Residential (Mean)
Assignment 1	4	3.90	3.52
Assignment 2	4	3.94	3.70
Assignment 3	4	3.95	3.52
Assignment 4	4	3.92	3.83
Assignment 5	4	3.89	3.75
Assignment 6	4	3.86	3.62
Assignment 7	4	3.91	3.77
Assignment 8	4	3.97	3.90
Project 1	100	94.47	92.61
Project 2	100	92.74	89.64
Project 3	100	93.10	92.17
Project 4	100	92.0	88.5
Midterm	100	70.2	70.0
Final Exam	75	93.76	93.48
Final Grade	100	92.32	91.31

Table 1. Average Grades Given on Each Assignment for the Residential and Online Sections of the KBAI Course during the Fall 2014 Term.

possible that the instruction online is comparable to, or perhaps even superior to, the residential instruction; a similar dynamic has been echoed by other instructors in the program (Carey 2016). On the other hand, online students tend to be older and more experienced, and so this superior performance may be solely due to their superior professional background and maturity in managing the coursework and learning the course materials. Although online students' superior performance is interesting, the greater point is that online students' performance is at least as good as residential students' performance, providing some support to the claim that the online degree is equivalent to the residential degree.

With regard to the student experience, we ask students to compare the KBAI class to other OMSCS courses, to online courses and general, and to college courses as a whole. Each and every semester, we have found the vast majority of students rate the online KBAI course as superior to courses in all three other categories. Although student evaluations and survey data are always not always completely reliable, we attribute more credibility to these results given their consistency and the experienced midcareer status of the vast majority of students in the program. Perhaps most interestingly, students reliably rate the KBAI course more favorably compared to other college courses than compared to other OMSCS courses, suggesting an underlying belief that many courses in the OMSCS program are likely to be better on the whole than traditional residential courses.

Using AI to Teach AI

Besides the detailed approach taken to the course's initial creation, the KBAI course is unique in its usage

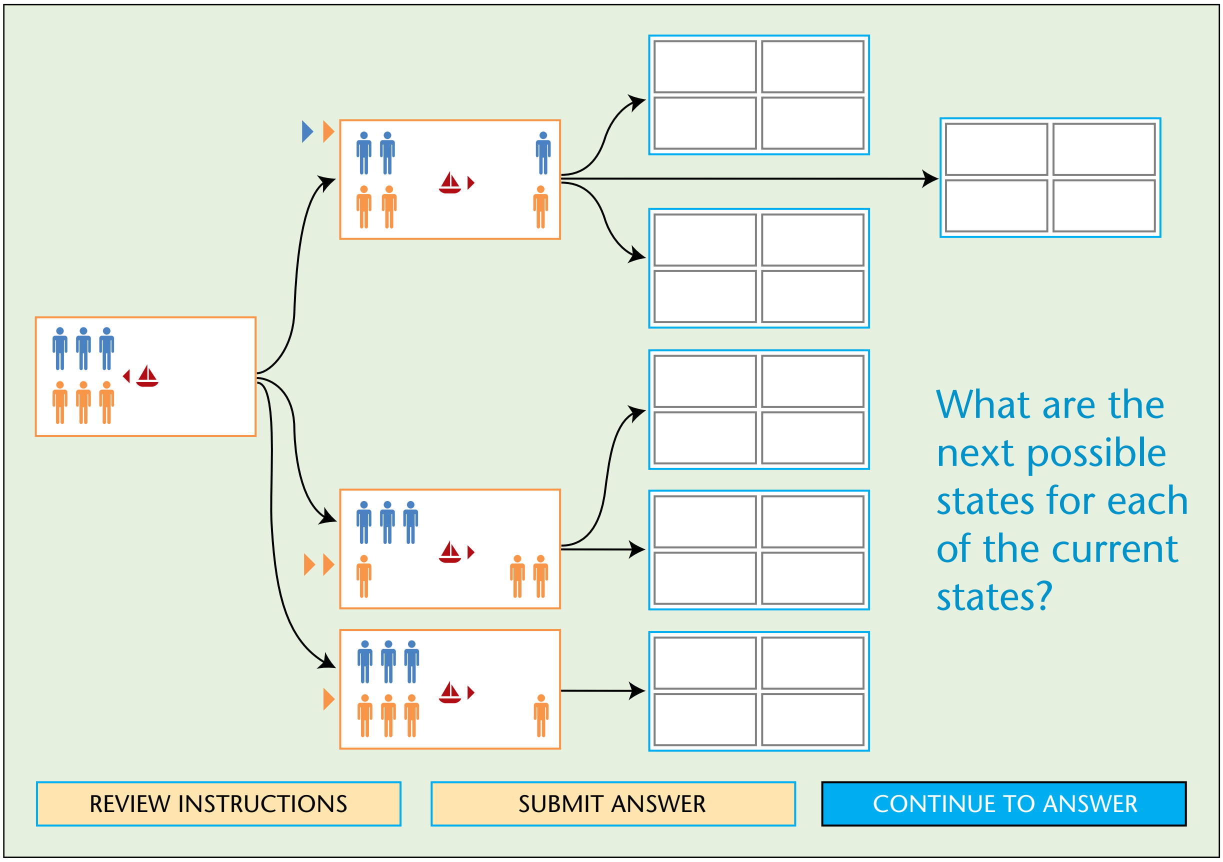

Figure 1. Example Exercise.

This is an example exercise from the fourth lesson of CS7637: Knowledge-Based AI. Here, students are asked to fill in 24 boxes to represent the possible next states of a problem in means-ends analysis in accordance with rules provided.

of AI not only as the subject matter of the course, but also as a tool to teach the course. In this section, we describe two ways in which we chose to use AI to teach AI, which we would advocate other advanced courses on artificial intelligence adopt.

Intelligent Tutoring of AI Concepts

While traditionally intelligent tutoring systems create computer-aided learning activities, the KBAI course already is online. The Udacity infrastructure for video lessons provides a facility for creating flexible interactive exercises involving multiple input types that can be evaluated by custom Python code. Using that framework, we equipped the lecture material for the course with about 150 interactive exercises. Figure 1 illustrates an example of an exercise; this exercise can be completed in the video lesson itself.

In addition, building on our prior work on intelligent tutoring systems (Joyner and Goel 2015a, 2015b), we created about 100 "nanotutors" to support the exercises and embedded them in the video lessons. The nanotutors are highly focused intelligent tutoring agents guiding students' understanding of one narrowly defined skill such as completing a semantic network for a particular problem or simulating an agent's planning in the blocks world.

Figure 2 shows some of the behaviors of the nanotutor for the exercise in figure 1. The nanotutor operates by first assessing the readability of the student's input; for example, in the exercise shown in figure 1, if a student entered a noninteger as input, the nanotutor would alert the student that the input did not match the rules of the problem, and would reiterate the exercise's acceptable input. In this way, the nanotutor first operates by taking open-ended student text input and guiding it toward the narrow-

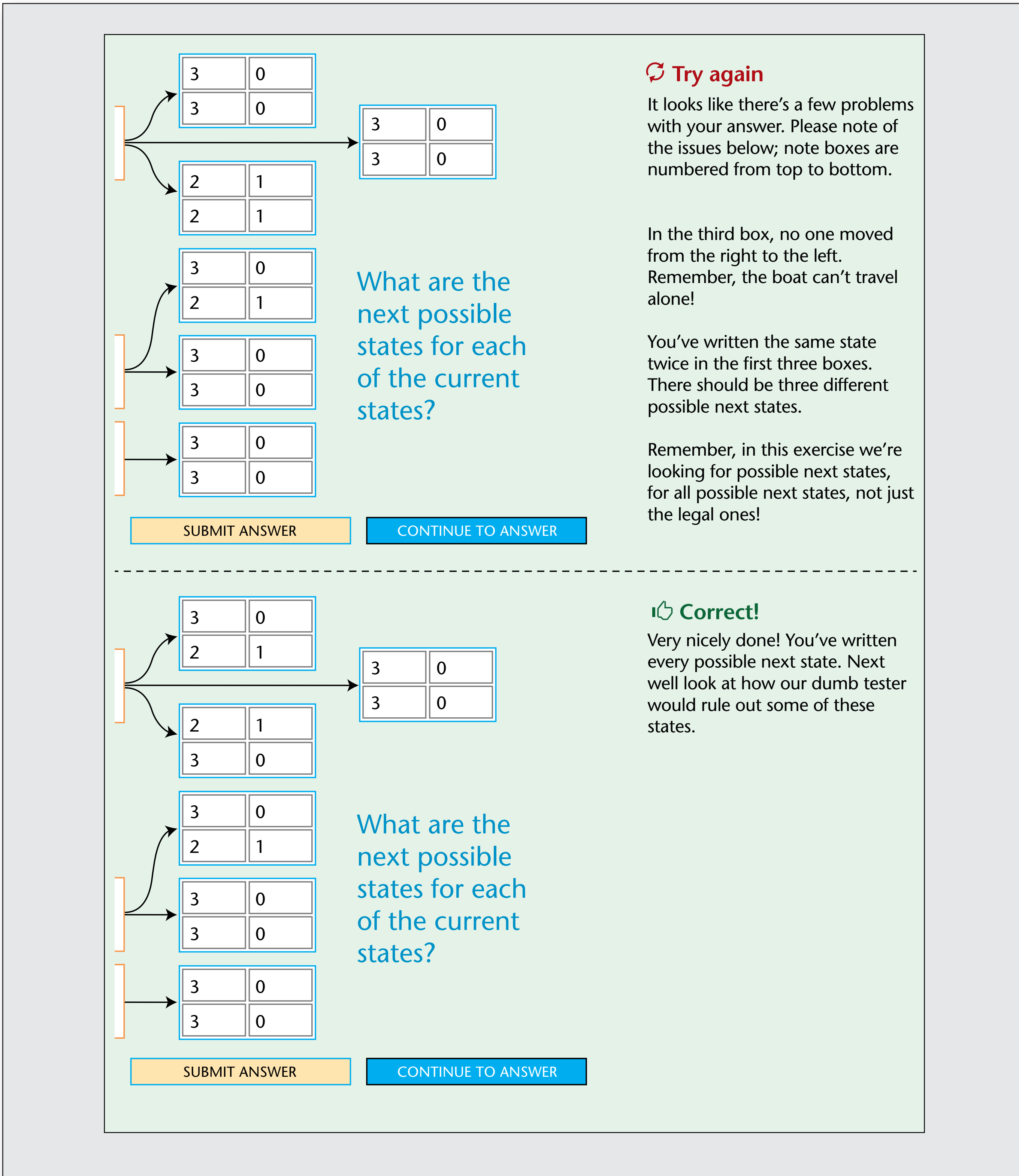

Figure 2. Feedback from the Nanotutor.

Examples of two pieces of feedback the student may receive from the nanotutor based on her input. On the top (a), there are errors with the student's reasoning in two of the states, and the nanotutor provides the guidance to correct these errors. On the bottom (b), the nanotutor confirms that the student has successfully completed the assigned exercise.

er set of inputs the agent can understand. The agent would then test whether the now-readable input obeyed the rules of the problem. In figure 2a, for example, the student disobeyed a rule of the problem. The nanotutor explains the rule to the student. Then, if the student input is readable and all rules are obeyed, the nanotutor assesses whether the final state matches the goal state. If not, the nanotutor directs the student to the difference between their answer and the goal state. At every step of the process, the nanotutor contextualizes the feedback in terms of the concept demonstrated.

Altogether, the 150 exercises in the course are equipped with approximately 100 nanotutors; some exercises share nanotutors, and others have no individualized feedback. The construction of these nanotutors addresses the problem of labor in constructing intelligent tutoring systems; each nanotutor required on average of less than an hour to build, ranging from a few minutes to several hours depending on the extent to which generalizable frameworks could be leveraged for the individual tutor's reasoning.

Evaluation of the nanotutors is embedded in the two forms of course evaluation described previously. First, the nanotutors act in support of the course's video lessons, which is assessed through the written assignments and examinations. Only online students received access to these exercises in nanotutors in the fall 2014 term, and therefore it is possible that this access is responsible for the online students' superior performance on the assessments. Second, in surveys of student satisfaction, we explicitly ask about students' perceptions of the interactive exercises and accompanying nanotutors. In general we found that about 80 percent of students agree that the interactive exercises improve their understanding of the material, and about 75 percent of students agree that nanotutors also help enhance their understanding of the material.

Authentic Engagement in AI Research

Research in cognitive and learning sciences informs us that student learning is enhanced through engagement with authentic scientific practices (for example, Edelson [1998]). Thus, for several years, our residential KBAI classes have using design and programming projects that derive from real AI research (Goel et al. 2013). In particular, our research laboratory is investigating problem solving on the Raven's Progressive Matrices Test of human intelligence (RPM; Raven [1941]) and has developed several techniques for AI agents to address RPM problems with human-level performance (Kunda, McGreggor, and Goel 2013; McGreggor, Kunda, and Goel 2014). Thus, for the last few years residential KBAI classes have been using design projects derived from our research on RPM problem solving: the students re-create the AI agents we have developed in our laboratory but are also encouraged to design their own techniques (Goel et al. 2013). The online KBAI class too has used the same kind of design projects since its inception in 2014.

To allow students to participate in these projects authentically, we supply them a set of RPM-style problems that we developed, and then test their agents against the real Raven's Progressive Matrices (which is never provided to the students directly for copyright reasons). Two examples of these RPM-style problems are shown in figure 3. This authenticity has several pedagogical benefits. First, it contextualizes the challenging elements of the project as inherent to the problem rather than artificially for the sake of difficulty. Second, it encourages students to think not just about the established principles and methods of the community, but also the dynamic and emerging theories. Third, it provides to students a fundamental view on the types of questions and methods the community asks and uses. The quality of some student projects is high enough that it already has led to one publication (Joyner et al. 2015), and more are forthcoming as former students in the class have begun follow-on projects building on their classwork.

Lessons Learned

Delivering the online version of CS7637 has been an incredible learning experience for us over the past two years. We have both been struck by the ownership of online students over their class experience. Every semester we have strived to improve the class and leverage lessons we learned during the previous semester. The following subsections are five of the lessons we have learned along the way that we would recommend transferring both to future online classes in AI and to other online learning programs in general. Interestingly, these lessons generally demand considerable expertise and commitment to developing strong online experiences; however, they do not necessarily demand enormous resource investment. Although producing the class required an enormous number of person-hours from Goel, Joyner, and the video production team at Udacity, many of the elements that contribute to the success of the class are not reliant on this kind of resource investment.

Integrating Interactivity from the Beginning

While most educators know the value of active learning in delivering superior learning outcomes, the lack of experience with online education and in the intensity of the production process can lure many first-time online course creators into a rote instructional approach. Thus, many online courses simply record traditional lectures with no interactivity whatsoever, while others inject token rather than deep interactivity. For example, simple multiple-choice or unevaluated essay prompts are common in MOOCs,

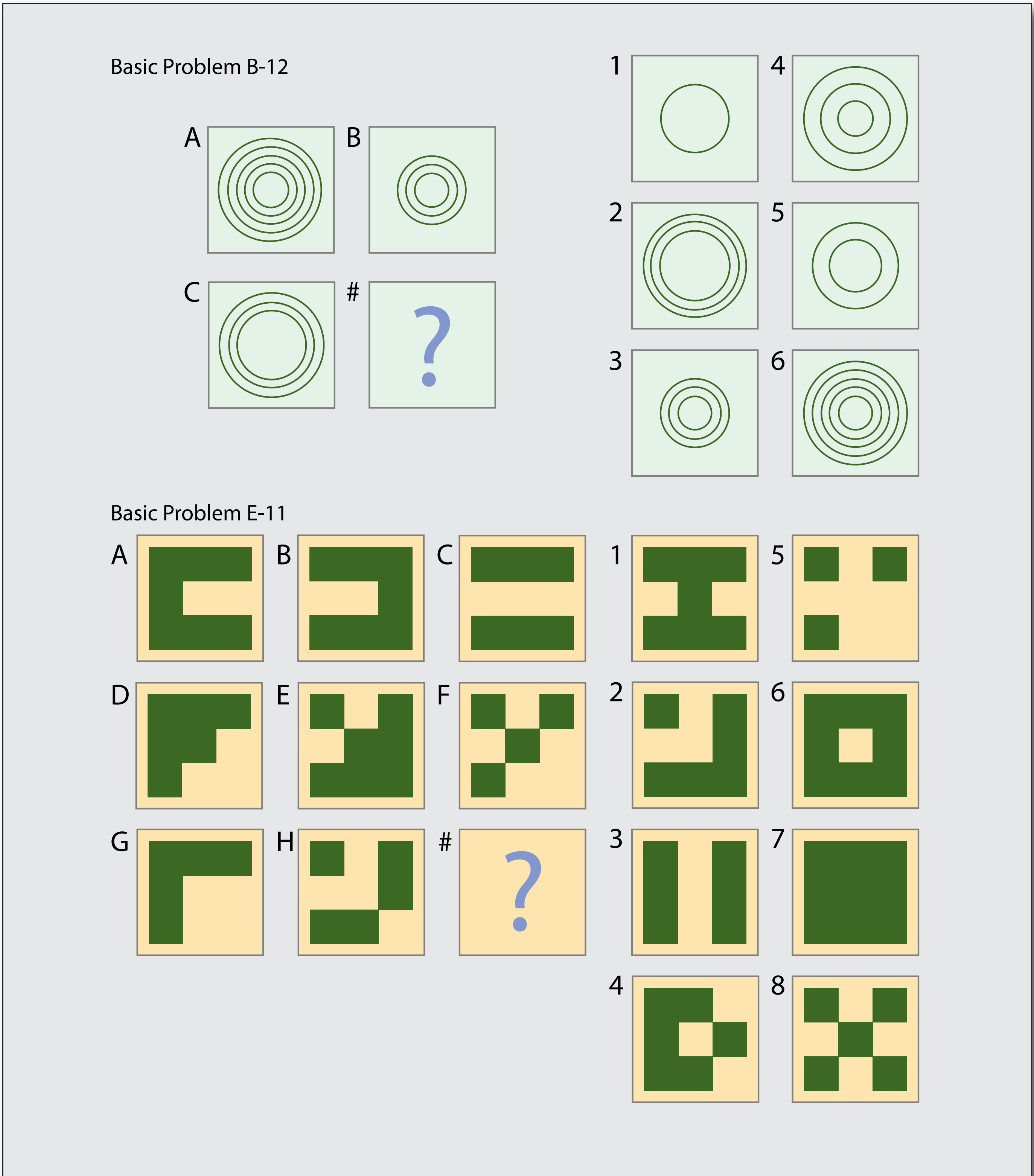

Figure 3. RPM-Style Problems.

The figure provides examples of 2x2 (top) and 3x3 (bottom) RPM-style problems used during the projects in CS7637: Knowledge-Based AI. Students begin by working on simpler 2x2 problems, and over time start to approach more complex 3x3 problems.

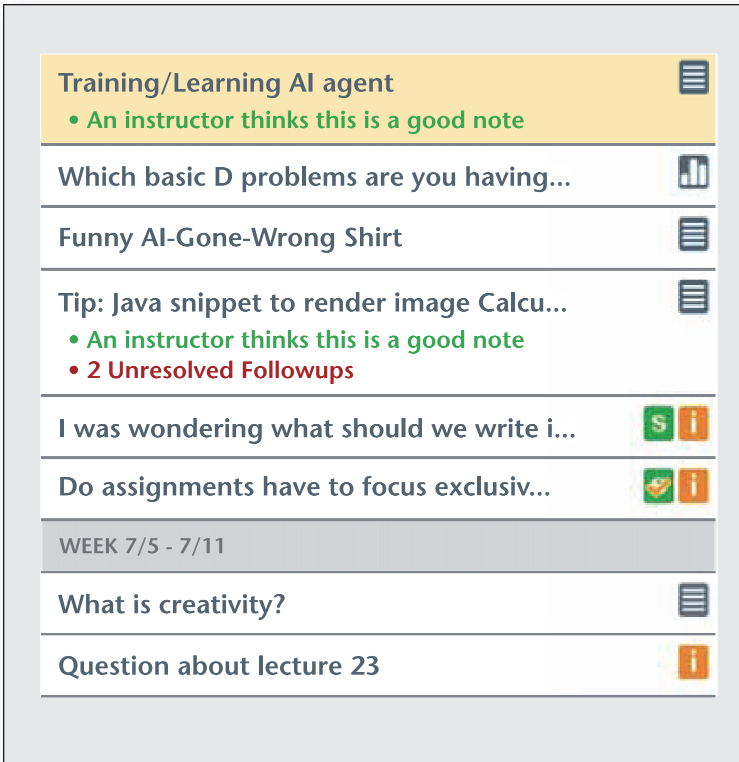

Figure 4. Discussion Snapshot.

A snapshot of the discussions created by students in a 36-hour period during the summer 2015 offering of the KBAI course.

but these approaches do not take full advantage of the interactivity possible in this medium. The video lessons in the online KBAI class, by contrast, was constructed with interactivity as its foundation. Every lesson is built around an example of some type of reasoning an AI agent could perform, and students are frequently asked to simulate or predict the results of this reasoning themselves. Rather than adding simple questions after the fact, this interactivity was the foundation of the initial course scripting process. As noted above, we observed that online students outperformed residential students in fall 2014 on the course assessments; if this result is due to the superior instructional material, it suggests this design decision improves students' learning outcomes and satisfaction with the course.

Empowering the Student Community

As noted above, arguably our greatest lesson from teaching this course has been the role of the community of learning in the online KBAI class. First, the student community in the program is remarkably well-qualified: nearly a fifth of students already have graduate-level degrees of some kind, and many have worked professionally in software development, data science, or related fields for years prior to entering the program. The community knowledge surpasses the material we could ever deliver intentionally through preprepared lecture material. Nothing we do can replicate the power of having actual AI researchers as students (and later, teaching assistants) in the class. Not only are the students fantastically qualified, but they also take significant ownership over the class experiences. Figure 4 shows a snapshot of the level of student activity in the class: students created four discussions, three questions and a poll in 36 hours during the Summer 2015 semester, drawing over three dozen responses and two student answers to classmates' questions. Three of these posts involved students sharing to help their classmates, while a fourth posed a philosophical discussion question and a fifth was purely social. No incentive was given for participation; this student ownership is purely organic.

In response to this discovery, we have learned to take active steps to empower the student community in the online KBAI class. Thus, we have created a more accommodating collaboration policy to maximize the extent to which students may learn from their well-qualified classmates. We also stress usage of a peer review system that pairs each student with several classmates on each assignment, allowing them to benefit from the professional experience of other students.

Leveraging Research for Authentic Projects

As noted previously, one of our approaches to using AI to teach AI is to engage students in authentic research projects that can immediately translate to publications or participation in active groups. This requires two unique efforts. First, the projects that students work on within the class must be designed in such a way that there is the potential they may translate to real-world publications and research. In KBAI, students re-create and contribute to an ongoing body of research pursued by the community.

However, creating projects that have the potential to carry over into real-world research does not guarantee they actually will. Steps must also be taken to support students interested in continuing to pursue those projects. We have accomplished this in a number of ways: by specifically offering students the opportunity to collaborate on a publication based on their project work; by opening master's projects and theses for students to continue developing their projects for class credit; and by setting up a research lab targeted at online students.

Recreating Features of the Residential Class

One of the common criticisms of online education is the perception that to offer the class online, certain material, relationships, or procedures must be removed, thus weakening the class. We have observed that many of these, such as student-student and student-instructor interaction, are no weaker online than in person (and in fact, may be stronger).

However, there are other elements of the residential class experience that are taken for granted and must be re-created manually. For example, we initially underestimated the extent to which having a regularly scheduled meeting time sets up what we call a classroom cadence, a rhythm to the class's interaction. We replicated this in part through weekly routine announcements to create an online equivalent of the in-person routine.

AI may play a key role in this lesson as well. The dynamics that create a classroom cadence are routine, predictable, and foreseeable; thus, it should be possible to equip an AI agent with the ability to interact in a way that establishes that rhythm. AI agents like our nanotutors may also play a role in re-creating natural features of the residential class; the online environment does not have a natural equivalent of a class exercise in which an instructor can intervene live to give feedback, but our interactive exercises play exactly this role.

Using Automated Evaluation for Frequent Formative Feedback

As education scales up to classes with hundreds or thousands of students, one of the pushes is for an increased emphasis on automatic evaluation. This can range from simple multiple-choice quizzes to more complex simulation-graded assignments. What is often lost in this emphasis is the incredible influence these forms of automated evaluation can have on students' individual feedback cycles in working on assignments. Many classes only run these automated evaluators after students have submitted their work. The emphasis here is on generating grades, not generating feedback or supporting learning experiences. If the evaluation is generated automatically, though, it presents a wonderful opportunity to equip students with the tools necessary to rapidly iterate in their understanding.

As noted previously, students in the KBAI class design agents that can answer a set of problems the student can see, and we then evaluate them against a set of problems the student cannot see. Prior to the summer 2016 semester, this latter step was only conducted after the submission deadline. In summer 2016 term, however, we launched a new automated grader that would allow students to see their agents' results on those unseen problems without having access to the problems themselves. This presents a lesson to any course developing automated evaluation solutions: while it is natural to focus on such solutions for generating the grades necessary to scale, make sure to extend the benefits of those automated evaluators to the students as well through frequent formative feedback.

Conclusions

Creating and delivering the online KBAI class has been one of the most satisfying educational experiences of our careers, and we wholeheartedly encourage anyone with the opportunity to participate in this new environment to try it out for themselves. The level of student motivation, engagement and ownership are worth the massive time needed to create and deliver these courses. That said, there are many open issues left to address. With regard to using AI to teach AI, we are still exploring the range of topics that can be addressed by nanotutors, the level of authenticity that can be provided through class projects, as well as the development of virtual teaching assistants can answer automatically some classes of questions on the discussion forums. With regard to online education as a whole, we must continue to explore metrics for ensuring that learning outcomes rival or exceed residential equivalents and represent real value to students.

Acknowledgements

We are grateful to several people for their support for the development of the OMS CS7637 KBAI class: the staff of the Georgia Tech OMSCS program, especially Zvi Galil, Charles Isbell, and David White; the Georgia Tech team within Udacity, especially Sebastian Thrun, Jason Barros, and Jennie Kim; and the staff of Georgia Tech Professional Education, especially Nelson Baker. We are especially grateful to our video producer, Aaron Gross, for producing the MOOC material for the course, and to Joe Gonzales for creating the peer review system we use in the class. Lastly, we are grateful to the numerous who have worked as CS7637 Teaching Assistants since Fall 2014. This article is based in part on Goel and Joyner (2016a, 2016b).

Notes

1. www.omscs.gatech.edu.
2. classroom.udacity.com/courses/ud409/.
3. www.youtube.com/watch?v=WbCguICyfTA.

References

Azevedo, R., and Aleven, V. 2013. *International Handbook of Metacognition and Learning Technologies*. Berlin: Springer. doi.org/10.1007/978-1-4419-5546-3

Carey, K. 2016. An Online Education Breakthrough? A Master's Degree for a Mere $7,000. *The New York Times,* September 28. (www.nytimes.com/2016/09/29/upshot/an-online-education-breakthrough-a-masters-degree-for-a-mere-7000.html)

Connelly, D., and Goel, A. 2013. Paradigms of AI Programming in Python. In *Proceedings of the Fourth Symposium on Educational Advances in AI.* Palo Alto, CA: AAAI Press.

Edelson, D. 1998. Realising Authentic Science Learning Through the Adaptation of Science Practice. In *International Handbook of Science Education,* ed. B. Fraser and K. Tobin, 317–331. Dordrecht, The Netherlands: Kluwer. doi.org/10.1007/978-94-011-4940-2_19

Goel, A., and Joyner, D. A. 2016a. An Experiment in Teaching Cognitive Systems Online. *International Journal for Scholarship of Technology — Enhanced Learning* 1(1).

Goel, A. 1994. Teaching Introductory Artificial Intelligence: A Design Stance. In Improving Instruction of Introductory Artificial Intelligence: Papers from the AAAI Fall Sympposium. Technical Report FS-94-05. Menlo Park, CA: AAAI Press.

Goel, A., and Davies, J. 2011. Artificial Intelligence. In *Handbook of Intelligence,* 3rd edition, ed. R. Sternberg and S. Kauffman, 468-484. Cambridge, UK: Cambridge University Press.

Goel, A. and Joyner, D. 2016a. An Experiment in Teaching Cognitive Systems Online. *International Journal for Scholarship of Technology-Enhanced Learning 1*(1).

Goel, A., and Joyner, D. 2016b. Design of an Online Course on Knowledge-Based AI. In *Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence,* 4089–4094. Palo Alto, CA: AAAI Press.

Goel, A.; Kunda, M.; Joyner, D.; and Vattam, S. 2013. Learning about Representational Modality: Design and Programming Projects for Knowledge-Based AI. In *Proceedings of the Fourth Symposium on Educational Advances in Artificial Intelligence*. Palo Alto, CA: AAAI Press.

Goodman, J.; Melkers, J.; and Pallais, A. 2016. Does Online Delivery Increase Access to Education? Harvard University Kennedy School Faculty Research Working Paper Series RWP16-035. Cambridge, MA: Harvard University.

Jonassen, D.; Peck, K.; and Wilson, B. 1999. *Learning with Technology in the Classroom: A Constructivist Perspective*. New York: Merrill/Prentice-Hall.

Joyner, D., and Goel, A. 2015a. Improving Inquiry-Driven Modeling in Science Education Through Interaction with Intelligent Tutoring Agents. In *Proceedings of the 20th ACM Conference on Intelligent User Interfaces,* 5-16. New York: Association for Computing Machinery. doi.org/10.1145/2678025.2701398

Joyner, D., and Goel, A. 2015b. Improving Scientific Modeling through Metacognitive Tutoring Based on Functional Roles of Teachers. In *Proceedings of the 37th Annual Meeting of the Cognitive Science Society.* Wheat Ridge, CO: Cognitive Science Society.

Joyner, D.; Bedwell, D.; Graham, C.; Lemmon, W.; Martinez, O.; and Goel, A. 2015. Using Human Computation to Acquire Novel Methods for Addressing Visual Analogy Problems on Intelligence Tests. Paper presented at the Sixth International Conference on Computational Creativity, Park City, UT, 29 June–2 July.

Joyner, D. A.; Goel, A.; and Isbell, C. 2016. The Unexpected Pedagogical Benefits of Making Higher Education Accessible. In *Proceedings of the Third Annual ACM Conference on Learning at Scale*. New York: Association for Computing Machinery. doi.org/10.1145/2876034.2893383

Koedinger, K., and Corbett, A. 2006. Cognitive Tutors: Technology Bringing Learning Science to the Classroom. In *The Cambridge Handbook of the Learning Sciences,* ed. K. Sawyer, 61–78. Cambridge, UK: Cambridge University Press.

Kunda, M.; McGreggor, K.; and Goel, A. K. (2013). A Computational Model for Solving Problems from the Raven's Progressive Matrices intelligence Test Using Iconic Visual Representations. *Cognitive Systems Research* 22-23(June): 47–66. doi.org/10.1016/j.cogsys.2012.08.001

Langley, P. 2012. The Cognitive Systems Paradigm. *Advances in Cognitive Systems* 1: 3-13.

Leckart, S. 2012. The Stanford Education Experiment Could Change Higher Learning Forever. *Wired,* March 20.

Lehman, J.; Laird, J.; and Rosenbloom, P. 2006. A Gentle Introduction to Soar, An Architecture for Human Cognition: 2006 Update. Unpublished Paper, The University of Michigan, Ann Arbor, MI.

McGreggor, K.; Kunda, M.; and Goel, A. 2014. Fractals and Ravens. *Artificial Intelligence* 215 (October): 1–23. doi.org/10.1016/j.artint.2014.05.005

Nilsson, N. 1998. *Principles of Artificial Intelligence: A New Synthesis*. San Francisco: Morgan Kauffman Publishers.

Norvig, P. 1992. *Paradigms of AI Programming*. San Francisco: Morgan Kauffman Publishers.

Ou, C.; Goel, A.; Joyner, D.; and Haynes, D. 201). Designing Videos with Pedagogical Strategies: Online Students' Perceptions of Their Effectiveness. In *Proceedings of the Third (2016) ACM Conference on Learning @Scale 2016,* 141–144. New York: Association for Computing Machinery. doi.org/10.1145/2876034.2893391

Raith, A. 2011. Stanford for Everyone: More Than 120,000 Enroll in Free Classes. *MindShift*. San Francisco: KQED Inc.

(ww2.kqed.org/mindshift/2011/08/23/stanford-for-everyone-more-than-120000-enroll-in-free-classes)

Raven, J. C. 1941. Standardization of Progressive Matrices, 1938. *British Journal of Medical Psychology* 19(1): 137–150. doi.org/10.1111/j.2044-8341.1941.tb00316.x

Russell, S., and Norvig, P. 2009. *Artificial Intelligence: A Modern Approach,* 3rd ed. Engelwood Cliffs, NJ: Prentice Hall.

Sleeman, D. H., and Brown, J. S. 1982. *Intelligent Tutoring Systems*. London: Academic Press.

Stefik, M. 1995. *Knowledge Systems*. San Francisco: Morgan Kauffman Publishers.

Winston, P. 1993. Artificial Intelligence, 3rd Ed. Cambridge, MA: The MIT Press.

Ashok Goel is a professor of computer science in the School of Interactive Computing at Georgia Institute of Technology in Atlanta, USA. He is also the director of Georgia Tech's Design and Intelligence Laboratory, and the Ph.D. program in human-centered computing. For more than 30 years, Ashok has conducted research into artificial intelligence, cognitive science, and human-centered computing, with a focus on computational design, modeling, and creativity. He is the editor-in-chief of *AI Magazine*. As part of the OMSCS KBAI class described here, he developed Jill Watson, a virtual teaching assistant for answering questions in online discussion forums.[3]

David Joyner is the product lead in charge of the Georgia Tech Online Master of Science in Computer Science (OMSCS) at Udacity, as well as a lecturer in the Georgia Tech College of Computing, teaching three online classes — CS6460: Educational Technology; CS6750: Human-Computer Interaction; and CS1301: Introduction to Computing. He is the founder and director of LucyLabs, a research lab dedicated to research by and about online students, as well as the 2016 recipient of the College of Computing's Lockheed Excellence in Teaching award, and the 2017 recipient of the College of Computing's Outstanding Instructor award.