Building Intelligent Tutors: Theory, Pedagogy and Technology

Beverly Park Woolf
Department of Computer Science
University of Massachusetts, Amherst, MA. 01003, U.S.A. bev@cs.umass.edu

Brief Table of Contents

PART I: INTRODUCTION TO ARTIFICIAL INTELLIGENCE AND EDUCATION
1. Introduction to Intelligent Tutors
2. Issues and Features of Intelligent Tutors

PART II: REPRESENTING AND REASONING ABOUT KNOWLEDGE
3. Student Knowledge
4. Tutoring Knowledge
5. Communication Knowledge
7. Machine Learning and Reasoning about Uncertainty
8. Authoring Tools and Expert Systems
9. Evaluation of Tutoring Systems

PART III: LEARNING ENVIRONMENTS
10. Microworlds, Simulations and Virtual Reality
11. Collaborative and Inquiry Environments
12. Web-based Tutors

Expanded Table of Contents

Overview of the Book

This book describes the field of artificial intelligence in education (AIED), which gives rise to intelligent tutoring systems (ITS), computer systems that tailor instruction for individual students and provide adaptive and individualized responses. The field is a synthesis of artificial intelligence, cognitive science, the Internet and other information and communication technologies. It provides more effective and efficient instruction than either traditional classrooms or existing computer instruction and provide teachers with dynamic assessment of student knowledge, thus classifying manageable clusters of students at similar learning levels. Such tutors are used with either individuals or groups of students and have been built into simulations, games, open-ended learning environments, hypermedia and Web-based intelligent learning.

The book is designed for future and current researchers, trainers, teachers and students; those who will develop, build and use intelligent tutors. It addresses the strategies, tools, methodologies and pedagogy and describes how to produce the best possible intelligent learning environment, whether for classroom instruction or non-traditional ‘lifelong’ learning.

PART I: INTRODUCTION TO ARTIFICIAL INTELLIGENCE AND EDUCATION

Chapter 1. Introduction to Intelligent Tutors

This chapter focuses specifically on how to add intelligence about students and teaching to a variety of instructional software media. It describes a revolution in education resulting from three main drivers: Artificial intelligence (AI) which leads to a deeper understanding of how to represent and reason about knowledge, especially “how to” knowledge, such as procedures; cognitive science, which provides a deeper understanding of how people think, solve problems and learn; and the Internet, which supplies an unlimited source of information, available any time, any where. This chapter provides an overview of recent research in cognitive science and instructional design, traditional teaching strategies, effective teaching activities and example tutors demonstrating the powerful instructional features.

Outline Chapter 1

1.1 Effective Learning and Teaching
1.2 Traditional Educational Technology and Beyond
1.3 Example Intelligent Tutors
1.4 Overview of the Book
1.4.1 Issues and Features of Intelligent Tutors
1.4.2 Representing Knowledge in an Intelligent Tutor
Chapter 2: Issues and Features of Intelligent Tutors

Seven features distinguish intelligent tutors from traditional computer-based instruction (CAI). These include generativity, student modeling, expert modeling, mixed initiative, interactive learning, instructional modeling, and self-improving. Generativity is a key and basic feature; intelligent tutors generate customized problems, hints or help—as opposed to the presentation of already prepared “canned” instruction. Generativity relies on models of the subject matter, the student and tutoring, which enable the tutor to generate customized instructional as needed in a form students can understand. Exactly which features are most important is still being debated. In general, the more features an instructional system has, the more effective it is instructionally. The chapter provides a lense through which the reader can understand and explore the field, including the vast amount of knowledge needed to encode in a tutor and available tools and methods. The term “intelligent tutor” is not just a marketing slogan for conventional computer-assisted instruction (CAI) in spite of much current practice. The term, in fact, designates technology-based instruction that is qualitatively different and an improvement over CAI.

Outline Chapter 2
2.1 Example Intelligent Tutors
   2.1.1 AnimalWatch, an Arithmetic Tutor
   2.1.2 The Cardiac Tutor
2.2 Features of Intelligent Tutors
2.3 Environments for Active Learning
   2.3.1 Methods Derived from Learning Theories
   2.3.2 Constructivist Teaching Methods
2.4 Theoretic Approach to Learning Environments
2.5 Computer Science, Cognitive Science and Education
2.6 A Short History of Computers in Education
2.7 How are Intelligent Tutors built?
2.8 Summary

PART II: REPRESENTING AND REASONING ABOUT KNOWLEDGE

Master human teachers use vast amounts of knowledge to teach; They know about the domain to be taught, about different student learning needs and about a variety of teaching strategies. AI in Education focuses on how to provide this same knowledge to a computer, dealing with representing and reasoning about domain knowledge, a student’s learning needs and which style of teaching is most effective for which student and when. This second part of the book explores a variety of knowledge representations for students and domain knowledge and teaching strategies. It also describes methods to reason about this knowledge, assess learning, synthesize the results, and extend tutors to new domains. A variety of reasoning and control mechanisms are described along with machine learning techniques, agent technologies, authoring tools, expert systems and evaluation techniques to enable tutors to reason efficiently and effectively.

Chapter 3. Student knowledge

This chapter describes how to represent and reason about a variety of student knowledge including knowledge about skills (declarative and procedural) and affective characteristics (motivation and engagement). This knowledge is used to identify appropriate teaching strategies, problems and hints for a student’s learning needs. A master human teacher gains experience over the years by working with multiple students and uses a learner’s responses, e.g., facial expressions, body language or voice changes, to augment his/her own understanding of each student. This chapter explains why student knowledge is needed and the techniques used, including cognitive methods (overlay, bug libraries, or model tracing) and AI-based technologies (logic, expert systems, Bayesian belief networks, etc.).

Outline Chapter 3
3.1 What is student knowledge?
3.2 Why build a student model?
3.3 Classification of Student Model techniques
3.4 Example student models
   3.4.1 Modeling Skills: AnimalWatch
3.4.2 Modeling Procedures: Cardiac Tutor
3.4.3 Modeling Disposition (Affective Characteristics): Wayang Tutor

3.5 How is student knowledge represented?
3.5.1 Features of Knowledge in a Student model
3.5.2 Student modeling considerations

3.6 Techniques to Represent and Update Student Models.
3.6.1 Cognitive science techniques
3.6.1.1 Overlays models and bug libraries
3.6.1.2 Model tracing tutors.
3.6.1.3 Progressive student models.
3.6.2 Artificial intelligence techniques
3.6.2.1 Bayesian student model.
3.6.2.2 Formal logic / fuzzy logic student model
3.6.2.3 Expert system-based student models
3.6.2.4 Planning and plan recognition student models
3.6.2.5 Constraint-based student models

3.7 Open research issues in building Student Models

Summary

Chapter 4. Tutoring knowledge

Tutoring knowledge is used to decide what topic to teach and how to teach it. A variety of teaching objects (examples, simulations, hints) and actions (describe, define, interrupt) might be chosen dynamically for teaching object and each specific student. This chapter describes a variety of tutoring strategies, some culled from careful observation of human teachers (drill and practice, apprenticeship teaching), others from human learning theories (Vygotsky’s Zone of Proximal Development) and still others from the unusual power of technology, having no relation to classroom teaching (virtual reality, 3D simulations and multimedia). These teaching objects are similar to those used by human teachers to support students with differing learning needs. Methods and processes for encoding teaching objects and sophisticated reasoning about each are discussed, including traditional tutoring strategies, such as problem solving. Computer tutors reproduce a wealth of powerful teaching strategies that are known to be effective in the classroom, yet difficult to reproduce, e.g., inquiry learning and collaboration. Such strategies require 1 teacher for around 3 students and thus are rarely used due to time and resource demands.

Outline Chapter 4

4.1 What is teaching knowledge?
4.2 How is teaching knowledge represented?
4.3 Brief Examples of Teaching Knowledge
4.3.1 Apprentice teaching: Sherlock
4.3.1.0 What was Sherlock?
4.3.1.1 How was Sherlock built
4.3.1.2 Why was Sherlock built this way?
4.3.2 Model-tracing: PACT Tutor
4.3.2.1 What was the PACT tutor?
4.3.2.2 How was the PACT Tutor built?
4.3.2.3 Why was the PACT Tutor built this way?

4.4 Why model teaching knowledge? Theoretical basis of modeling teaching
4.4.1 Derive teaching strategies from models of human teaching.
4.4.1.1 Apprenticeship Learning
4.4.1.2 Problem-Solving and Handling Errors
4.4.1.3 Model-tracing Tutors
4.4.1.4 Buggy-based Tutors
4.4.1.5 Tutorial Dialogue
4.4.1.6 Case-Based Reasoning
4.4.1.7 Select New Problems/Help
4.4.1.8 Procedural Training
4.4.1.9 Collaborative Learning
4.4.2 Adapt teaching strategies from learning research/theory.
4.4.2.1 ACT* Learning Theory
4.4.2.2 Constructivist Theory (Genetic Epistemology)
4.4.2.3 Situated Learning / Experiential Learning
4.4.2.5 The Zone of Proximal Development
4.4.2.6 Reciprocal Teaching
4.4.2.7 Self-explanation
4.4.2.8 Socratic Learning
4.4.3 Facilitate teaching with technology.
4.4.3.1 Animated Pedagogical Agents
4.4.3.2 Virtual Reality
4.4.3.3 Interactive Simulations and Illustrations

4.5 Modeling several teaching strategies.

4.6 Conclusion

Chapter 5. Communication Knowledge

Effective communication is not only a responsibility of the teacher, it is a critical component of teaching. Similarly, communication knowledge is critical for an intelligent tutor. Given the best student model, teaching model and expert module, the tutor will be of limited value without effective communication. Few things about a computer are more disagreeable to a user than a confusing or difficult to use interface, or a blatantly unattractive response. Thus, a large amount of work should go into developing the communication module. This chapter describes several approaches to achieve improved communication with students, the goal being to provide advice about errors, timely reminders and intelligent feedback. This is achieved through a variety of technologies, including i) animated pedagogical agents, ii) virtual environments and iii) didactic systems and supported by plan recognition, reasoning about student and domain knowledge. Natural language includes spoken and written language as input and output, thus language recognition understanding, mixed-initiative planning and generated responses.

Outline Chapter 5

5.1 Graphic Communication
5.1.1 Pedagogical Agents
5.1.2 Virtual Reality Environment
5.1.3 Computer Graphic Techniques

5.2 Meta-Communication

5.3 Integrated Interfaces

5.4 Natural Language Communication
5.4.1 What are Natural Language-based Intelligent Tutors?
5.4.1.0 Mixed Initiative Tutoring Systems
5.4.1.1 Directed Dialogue
5.4.2 Why build natural language interfaces?
5.4.3 How are Natural Language Communication systems built?
5.4.3.1 Knowledge-based natural language methods
5.4.3.2 Statistical natural language methods
5.4.3.3 Hybrid natural language methods

5.5 Linguistic issues behind natural language processing.

5.5.1 Sound understanding
5.5.2 Syntactic parsing
5.5.3 Semantic and pragmatic processing
5.5.4 Discourse processing
5.5.5 Speech Processing

5.6 Methods of natural language processing
5.6.1 Understanding syntax, knowledge-based methods
5.6.2 Understanding Semantics and Pragmatics; Knowledge-based methods
5.6.3 Understanding Dialogue, Knowledge-based methods
5.6.3.1 Stack-based Algorithm
5.6.3.2 Dialogue Grammars and Relations
5.6.3.3 Belief Models
5.6.3.4 Response Generation
5.6.3.5 Remaining Discourse Research Issues

5.7 Summary

Chapter 7. Machine learning and reasoning under uncertainty

Every classroom teacher learns about how to teach from working with hundreds of students; s/he learns about new teaching strategies and about the variety of individual student learning needs. It is also critical for computer tutors to learn and improve over time, based on observations of thousands of student interactions. Given that each student’s behavior is variable, and that detailed records of actions, timing and behavior is recorded, computer tutors have an ideal opportunity to improve their heuristics based on experience. Using machine learning (ML) computer tutors reason about which strategies were effective and then optimize teaching based on knowledge about an individual
student. Without ML, tutors use only hand-coded heuristics to reason about their knowledge and are set free to teach and do their best given the way the tutor was programmed. Thus their knowledge remains fossilized until extended with human help. This is a limited and short-sighted approach.

Machine learning, in part, improves a tutor’s teaching ability over time and thus ultimately reduce the high cost per student taught, decrease development time and broadening the range of users for a given tutor. This chapter clarifies the concept of machine learning in intelligent tutors, describes how to integrate learning into tutors and describes the limitations and barriers of several approaches. It examines ML techniques that enable tutors to extend their reach and accommodate new students and new domains, including tutors that learn about student knowledge, misconceptions or domain knowledge. Existing machine learning tutors can predict student learning style, future student actions and which teaching action will produce optimal learning, e.g., reduced time and increased mastery. This chapter presents detailed examples of machine learning tutors and discusses reasoning under uncertainty.

Outline Chapter 7

7.1 What is a Machine Learning Tutor?
7.2 How Is Machine Learning Integrated Into An Intelligent Tutor?
7.3 Using ML Methods To Improve Tutor Functionality
  7.3.1 Using ML to Learn about student and domain knowledge.
  7.3.2 Using ML to Predict Student’s Learning Style and Future Actions
  7.3.3 Using ML To Learn About Effective Teaching Actions
7.4 Two Example Machine Learning Tutors.
  7.4.1 ANDES: Using a Bayesian Belief Network to Reason about Student Knowledge
  7.4.2 AnimalWatch: Using Reinforcement Learning to Predict Student Action
    7.4.2.1 Interacting With The User
    7.4.2.2 Details of the Learning Mechanism
    7.4.2.3 Gathering Training Data For The Machine Learner
    7.4.2.4 State information recorded
    7.4.2.5 Current problem information
    7.4.2.6 Feedback information
    7.4.2.7 Information about the context
    7.4.2.8 Induction technique used by the learning mechanism
    7.4.2.9 Component Evaluation
    7.4.2.10 Evaluation of the Learning Model in AnimalWatch
    7.4.2.11 Classroom Evaluations of the Learning Mechanism
7.5 Reasoning Under Uncertainty
  7.5.1 Handling Uncertainty
  7.5.2 Basic probability notation
  7.5.3 Reasoning about a student’s related beliefs
7.6 Bayesian Belief Networks to Model Student Knowledge
  7.6.1 Why use Bayesian Belief Networks
  7.6.2 How are Bayesian Belief Networks built?
  7.6.3 Applying Bayes’ Rule
  7.6.4 Classifying Bayesian Networks for student models
  7.6.5 Bayesian Belief Networks From Other Domains
7.7 Other probabilistic methods
  7.7.1 Hidden Markov Graphs
  7.7.2 Fuzzy logic
  7.7.3 Decision Theory
7.8 Using Machine Learning to Optimize Tutor Actions.
  7.8.5 Optimizing Pedagogical Decision Making
  7.8.5.1 Memory and cognitive abilities

Chapter 8. Authoring Tools

This chapter discusses authoring tools that enable teachers and designers to construct new tutors, just as expert system shells have for decades enabled people to build expert systems. Dozens of pedagogical authoring tools have been built yet only a small selection of authoring tool are available to buy or use. Demonstrated success has been seen in several cases. It is still too early to know which of the many approaches will prove most useful or marketable. This chapter examines the methodology and features of authoring tools and examines expert systems and the authoring tools that have facilitated rapid development of practical expert system.

Outline Chapter 8

8.1 Authoring Tools for Tutors
  8.1.1 Authoring a Tutor/Building the Components
Chapter 9. Evaluation

This chapter describes how to evaluate an intelligent tutor and the results of many evaluation studies. Evaluation studies should demonstrate that the tutor is mostly bug-free, is polished enough for a production setting and that it is statistically defensible and shows pedagogical effectiveness. Additionally, evaluation methods should determine whether a tutor contributes to basic research, computational or educational questions, classroom learning or generalizations beyond the system and sample. This chapter reviews evaluation studies of intelligent tutors from a methodological perspective and discusses both short and long-term approaches, including how to choose multiple sites, counter balance the design and statistically control for multiple sites.

Outline Chapter 9

9.1 What is a Tutor Evaluation?
9.2 Evaluation Methodology
  9.2.1 Type of Evaluation
  9.2.2 Form of Comparison
  9.2.3 Evaluation Design
  9.2.4 Potential bias and common problems
9.3 Descriptions of Evaluation Studies
  9.3.1 Sherlock, a tutor for complex procedural skills.
  9.3.2 Model tracing tutors: Algebra and LISP tutors.
  9.3.3 Stat Lady, a statistics tutor (Slute, 1995)
  9.3.4 SQL Tutor: a constraint-based tutor for learning A database language
  9.3.5 ANDES: A Physics Tutor
  9.3.6 AnimalWatch, an Arithmetic Tutor
9.4 Meta-studies of instructional systems
9.5 Long-term Evaluation Issues

GLOSSARY

PART III: LEARNING ENVIRONMENTS

Earlier parts of this book explained the core principles and methodology for developing prototypical intelligent tutors designed for one-to-one tutoring. This part of the book extends the narrow definition of intelligent tutors to include a wide variety of instructional environments that are augmented with intelligent reasoning, including micro-worlds, 3d simulations, games, virtual reality, web-based systems and collaborative and inquiry systems. Intelligent tutors can incorporate multiple modality (simulations, OLE, games, tutorials, hypermedia, or multimedia) and can focus on either individuals or groups in collaborative learning. They can present instruction or enable people to learn for themselves (representing the ongoing debate between constructivist and instructivist activities).

Chapter 11. Inquiry and Collaborative Systems

This chapter discusses support for inquiry and collaborative learning. In these learning modes, students articulate and reflect upon their own knowledge; facilitating learning and transfer. Case-based or problem-based learning encourages student-initiated learning and investigations as an outgrowth of case analysis. It is one of the ways to bring inquiry communication to learning. Collaboration tutors embed shared workspaces, chat boxes, servers and modifiable artifacts (e.g., charts, graphs). Multi-user environments help move classroom activities from teacher-centered didactic instruction to student-centered collaborative inquiry. Grouped problem solving allows students to tackle more complex problems. The chapter focuses on technology to support each of these activities.
Chapter 11. Inquiry and Collaborative Environments

11.1 What are inquiry and collaborative environments?
11.2 Why build inquiry and collaborative environments?
11.3 How are inquiry environments built?
11.4 Examples of Active Inquiry Environments
   11.4.1 Belvedere
   11.4.2 Rashi
   11.4.3 Inquiry Systems at Northwestern
   11.4.4 Inquiry Support at Michigan
11.5 How are collaborative environments built?
   11.5.1 Joint construction of knowledge
   11.5.2 Joint negotiation of alternatives
11.6 Examples of collaboration environments
   11.6.1 LeCS: Case Study Environment
   11.6.2 Community of Advisors for Inquiry
   11.6.3 Suthers collaboration
11.7 Summary

Chapter 12. Web-based Intelligent Instruction

This chapter explores the impact of the World Wide Web on development of intelligent tutors. It describes the theory, development, deployment and assessment of intelligent Web-based instructional systems. Pedagogical and technical issues are explored, including the integration of intelligence and adaptivity within Web tutors. This chapter discusses these issues, provides examples of Web-based educational systems and examines tutors inspired by Web technology, such as adaptive navigation or content.

Outline Chapter 12
12.1 Theoretical Background
12.2 Collaboration and critical thinking on the Web
   12.2.1 Rashi, an Inquiry Web Tutor
   12.2.2 Why support critical thinking?
   12.2.3 How to build inquiry tutors
12.3 Adaptive Web-tutors
   12.3.1 iManic, an adaptive Web tutor
   12.3.2 How are adaptive Web tutors built?
   12.3.3 Details of building an adaptive web-based tutors
      12.3.3.0 Adaptive navigation: Customize travel to new pages
      12.3.3.1 Adaptive Presentation: Customize page content
12.4 Nuts and bolts (networks)
12.5 FUTURE TRENDS