TOWARDS EVALUATING HUMAN-INSTRUCTABLE SOFTWARE AGENTS

Robert D. Grant, David DeAngelis, Dan Luu, and Dewayne E. Perry
Empirical Software Engineering Laboratory
Center for Advanced Research in Software Engineering
The University of Texas at Austin, Austin, TX 78712
{bgrant,deangelis,luu,perry}@mail.utexas.edu

Kathy Ryall
Advanced Information Technologies
BAE Systems, Burlington, MA 01803
kathy.ryall@baesystems.com

ABSTRACT

The Bootstrapped Learning (BL) project is an attempt to create software agents (e-students) that are instructable by human teachers through natural instruction methods [Oblinger, 2006]. In this paper, we present an introduction to BL and three years of case studies investigating the use of human subjects in evaluating e-students. In our studies we investigate human teachers' expectations of e-students and important differences between human and software learners, including the greater semantic understanding of humans, the eidetic memory of e-students, and the importance of various study parameters including timing issues and lesson complexity to human performance.

KEYWORDS

User Study, Evaluation, Instructable Computing, Electronic Student, Machine Learning

1. INTRODUCTION

Bootstrapped Learning (BL) is a DARPA program aiming to create software agents that human instructors teach rather than program [Oblinger, 2006]. Creating a domain-independent learning agent (i.e., e-student) can be viewed as providing a more intelligent, natural user interface for underlying machine learning algorithms. Computational agents with this interface could be trained by domain experts who are not necessarily skilled programmers; this is especially valuable for systems that benefit from being "field-trainable", or specializable to a particular need by end users at a faster rate than is usually supported by a traditional software development lifecycle.

From an HCI perspective, we address two important issues: determining which instruction methods are most important for supporting human instruction of e-students, and developing human benchmarks for evaluating an e-student's success at learning. Our group has investigated these issues through a series of exploratory case studies.

In this paper, we begin by providing an overview of the BL research program as context for our evaluation work. We then present our findings from an initial study investigating how human teachers attempt to instruct e- students. Finally, we present two case studies where we explore using human students to generate benchmarks for experimental evaluation of e-students. We omit details of the domain of the final two case studies, as the testing domain must be kept hidden from those creating the e-students until after e-student evaluation.

2. RELATED WORK

Bootstrapped learning provides the context for this HCI research. Specifically, we investigate how humans teach and learn and then apply those findings to understand how humans can teach machines in a natural way.

2.1 Bootstrapped Learning Overview

The goal of the Bootstrapped Learning program is to build an e-student that can be taught by a human instructor in the same ways that humans instruct one another. As BL provides natural ways for a human to impart knowledge to a software learner, it does not require programming expertise; human instruction of e-students will make it possible to delegate tasks to computers that cannot be easily delegated today and will enable users to rapidly modify deployed systems.

BL differs from other kinds of machine learning (ML) in several ways. Current ML is primarily a modeling tool; it is used to build models when we know something, but not everything, about some target problem. Current ML is a process of discovery, and requires induction over large datasets over which to induce its models. There is no guarantee that target knowledge will be discovered.

BL allows users to impart the target knowledge in a more direct fashion; however, because it involves "natural" ways to impart knowledge, it does not require programming expertise. BL supports conceptual bootstrapping; it leads to meaningful intermediary levels of learned concepts. E-students learn laddered-curricula in which lessons build on previous lessons, whereas in current ML, learning is generally from unstructured data. Like its human counterpart, an e-student assumes all necessary knowledge is possessed by the instructor, and its goal is to learn using the "same" instruction methods used between humans.

Two teams have been working in parallel to explore this new learning paradigm. Our team, the Curriculum Team, is developing BLADE (Bootstrapped Learning Analysis and Curriculum Development Environment). This research includes developing a framework to support BL, a set of laddered curricula across a variety of domains as testing vehicles for the e-student, and an evaluation of the e-student on both hidden and known domains. A separate Learning Team is developing an e-student incorporating several learning strategies [C. Morrison and D. Bryce and I. Fasel and A. Rebguns, 2009].

BLADE includes three agents, whose interactions and relationships are shown in Figure 1. A teacher agent serves as a proxy for an eventual human teacher, instructing and testing the e-student. The student agent is the embodiment of the e-student, typically employing a number of learning algorithms. The world agent serves as a proxy for a domain simulator. Over the first three phases of the BL program the Curriculum Team has developed a set of laddered curricula in a variety of complex domains including Blocksworld [Berland and Perry, 2009], unmanned aerial vehicles (UAV), diagnosis tasks for an international space station (ISS), armored task force maneuvers (ATF), planning robotic arm movements, and a hidden domain.

BLADE uses IL (InterLingua) and ITL (InteracTion Language) [Curtis, 2009], developed specifically for BL, to pass messages between agents in the BLADE framework. For evaluation purposes an automated teacher agent is used to ease scaling and reproducibility. Part of our research is to explore how to best incorporate a human teacher. In a parallel effort we are developing a tool to support human-/e-student instructional interactions, in part informed by the evaluations described in this paper.

We expect the eventual outcome of our research to have impact in two useful ways. First, we will demonstrate that instructable computing is a valid and successful means of providing learning to an e-student. Second, the system we develop will provide the tools for other groups to pursue research in Bootstrapped Learning. These groups will be able to access our framework and supporting materials, including our BL curricula, our associated research papers and documents, and our software for supporting human benchmarking. Access to these resources will allow other researchers to experiment with the development of their own e-students, and explore UI designs and techniques for human instructors to interact with e-students. This living repository will help catalyze future work in BL just as the Irvine Repository [Frank and Asuncion, 2010] drove machine learning in the 1980s, and as DARPA's MUC competitions [Grishman and Sundheim, 1996] inspired research in corpus-based approaches to natural language processing.

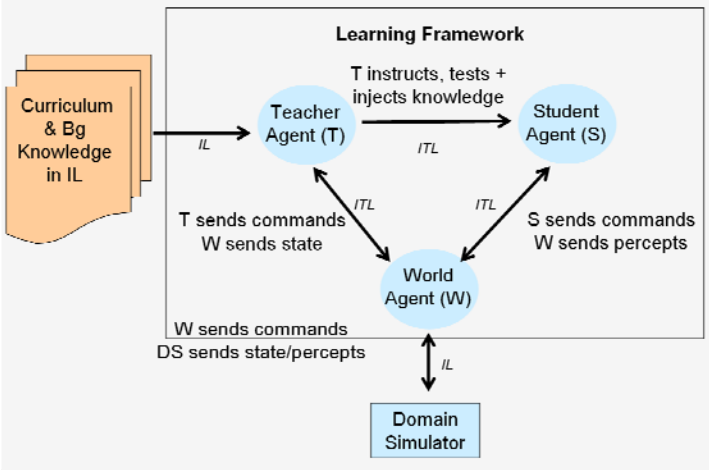


Figure 1: The BLADE Framework

2.2 Human Learning and Teaching

Like its human counterpart, an e-student assumes its instructor possesses all relevant capabilities, and its goal is to learn using the same instruction methods used between humans. As part of the Evaluation Team, we are not allowed access to e-student implementation details to ensure our benchmarks are unbiased. We know that the learners are designed to be domain-independent, and that they are specialized to particular Natural Instruction Methods (NIMs) rather than particular problem domains.

The area of computer tutoring can be seen as an inverse problem to what we are investigating. In particular, the area of teachable agents bears some surface similarity to BL. In this field, human students teach learning agents in order to improve their own understanding of concepts ("Learning by Teaching"). One example is the "Betty's Brain" system [Davis et al., 2003]. However, in these systems the importance is placed on how well the human instructor learns, not on the capabilities of the learning agent.

2.3 Human Case Studies

Our human studies use well known techniques from behavioral research, as covered in standard texts such as [Rosenthal and Rosnow, 1991] and [Yin, 2008]. In particular, since little was known about what factors would be critical, our empirical approach is that of exploratory case studies. An exploratory case study is the best approach when little is understood about the subject under study [Yin, 2008]. The intent is to build a deeper understanding of the phenomena in question and to formulate the beginnings of a corresponding theory that can be tested, revised, and expanded with further empirical studies.

3. PHASE I STUDY: BLOCKSWORLD

The Curriculum Team performed an initial case study to explore what approaches a human teacher (HT) would take in teaching a particular curriculum to an e-student, what assumptions a teacher would make about a student, and how these assumptions would work in the context of an e-student [Berland and Perry, 2009]. The HTs were asked to consider teaching to the level of a bright two year old. The domain for this study was the Blocksworld simulated environment created by Cycorp, Inc. This environment consisted of a "claw", or crane-like device which the e-student could control to manipulate "square blocks" and "long blocks".

In this case study, each of five human teachers first attempted to teach an e-student to construct a stack of three blocks and then to build a simple "doorway" out of blocks in this environment. The target was a simple structure of two stacks of blocks topped with a lintel. We used a Wizard of Oz (WOz) [Dahlback et al., 1993] style methodology, where the teachers' natural-language instructions were translated into precise terms (IL) for the e-student by a human interpreter.

We compared the methods of the human teachers in terms of type of curriculum taught, teaching time, andhow well the e-student performed. We also gathered information from the teachers about their difficulties in the experience and their models of the e-student before and after their teaching session.
All five teachers successfully instructed the student to make a doorway, and more importantly, we gained important insight into how humans attempt to instruct e-students. Some of our observations are listed in Table 1.

Table 1: Phase I Observations{TC "1 Phase I Observations" \f t}

Description	Impact
All HTs ended up using a bottom-up approach to teaching (possibly due to capabilities of the e-student). Some HTs initially used a top-down approach but became frustrated and reverted to a bottom up approach	All BL curricula to date have been authored using a bottom-up structure. In the Phase II and Phase III evaluations with human learners (rather than e-students), we have found that human subjects often prefer a top-down instructional method. It is an open question as to how an e-student might learn with a different lesson structure.
All the HTs overestimated what the e-student knew and could do, assuming knowledge of primitives such as "choose a block" or "look for a clear space."	We believe the domain-independent e-student needs a minimal level of injected knowledge to support "basic human competencies". This can be achieved through the use of background knowledge which can be injected or built into an e-student.
Many of the HTs employed repetition and mnemonics when teaching.	Instructional interfaces should support natural human methods for teaching, including support for informalities. The Curriculum Team is currently exploring this issue.
The HTs differed in their assumptions regarding the linguistic capabilities of the e-student.	Instructional interfaces should mask the linguistic limitations of an e-student and/or a teacher should have a way to query an e-student's capabilities and understanding.

A human teacher teaches differently depending on the target audience [Dahlback et al., 1993]. Phase I was designed to gain insight into how a human teacher would instruct an e-student. Later phases use a standardized curriculum and are more focused on how well the students (human and electronic) learn the curriculum. Incidental benefits from Phase I included a test of the e-student instruction language and a greater understanding of the learning performance of an early version of the e-students.

4. PHASES II AND III: HUMAN COMPARISON STUDIES

In these case studies, the goals were to define and refine requirements, problem solving strategies, and evaluation methodologies for e-students by evaluating a version of the e-student curriculum with human students. We aimed to produce lessons and tests on which human students who scored less than 20% in pre-test could score at least 75-80% in post-test, indicating that learning occurred.

We performed two studies, one in Summer 2009 and one in Summer 2010, which we respectively refer to as "Phase II" and "Phase III". Our overarching goal was to mimic the e-student context as closely as possible in the human studies, as the eventual goal is to directly compare e-student learning with human-student learning on "identical" curricula in controlled experiments.

In all studies, the basic procedure was as follows:

- Introduction and background material presentation
- Pre-test
- Curriculum lessons with web-based quizzes
- Post-test

4.1 Natural Instruction Methods

We presented the curriculum to students via three different "Natural Instruction Methods" (NIMs): teaching by telling, teaching by example, and teaching by feedback. Respectively, these consisted of utterances emitted by a teacher, examples performed by a teacher in a domain simulator, and instructions for a student to apply techniques in a simulator with teacher feedback. We abbreviate these lesson types as T, E, and F. In Phase II, we gave each human student all three lesson types, with the option to skip. In Phase III, we tested some students with all three NIMs, and other groups with only one or two NIMs. We also allowed a small group of students to ask questions about the curriculum. We refer to the students who received all three instruction types as the "baseline".

4.2 Evolution of Testing Procedure

As these studies were exploratory, over their course we evolved the details of our testing procedure significantly. At the beginning of Phase II, we began with a direct analog of the relationship between an automated teacher and an e-student. Since we were concerned with evaluating the curriculum and not the subjects, a major concern was preventing a human teacher from unconsciously providing extra-curricular information to the student through facial expressions, gestures, tone, etc.

In this design there was one teacher, one student, and one observer per session. The teacher fed each line of curriculum to the student one-at-a-time through an electronic messaging interface. For the E lessons, a view of the teacher's domain simulator was duplicated on a monitor visible to the student. For the F lessons, a view of the student's domain simulator was duplicated on a monitor visible to the teacher, and the teacher provided feedback through the messaging interface. This teaching method was time-consuming, error-prone, and there was no protocol allowing the teacher to report errors or redo instructions. We quickly transitioned to a "self-paced" version of the curriculum.

In this version, instead of a human teacher feeding every line of curriculum to the student and demonstrating simulator usage manually, we formatted the curriculum as PowerPoint slides with instructions and accompanying figures. For the feedback/practice lessons, we provided the student with instructions on procedures to try in the simulator and what the outcome should look like if the procedure was performed correctly; we called this the "choose-your-own-adventure" style. This eliminated complexities in our lab setup, reduced concerns about students learning from extra-curricular cues, and allowed us to run several students in parallel. We began running two subjects at once in Phase II, and with some additional automation we were able to run six subjects at once in Phase III. For more details on our testing procedure see [Grant et al., 2011].

4.2.1 Quantitative Results

In these results we focus on a few aspects of our study that are interesting from an HCI perspective. In all results we exclude students that passed the pretest. When we discuss a "post-test score" in the following analysis, we mean the fraction correct out of five randomly-chosen post-test scenarios. When we discuss a realtime post-test, we mean that human students were tested in a domain simulator that automatically advanced through states in realtime. In non-realtime post-tests, the students advanced states manually (though there was still an overall time limit).

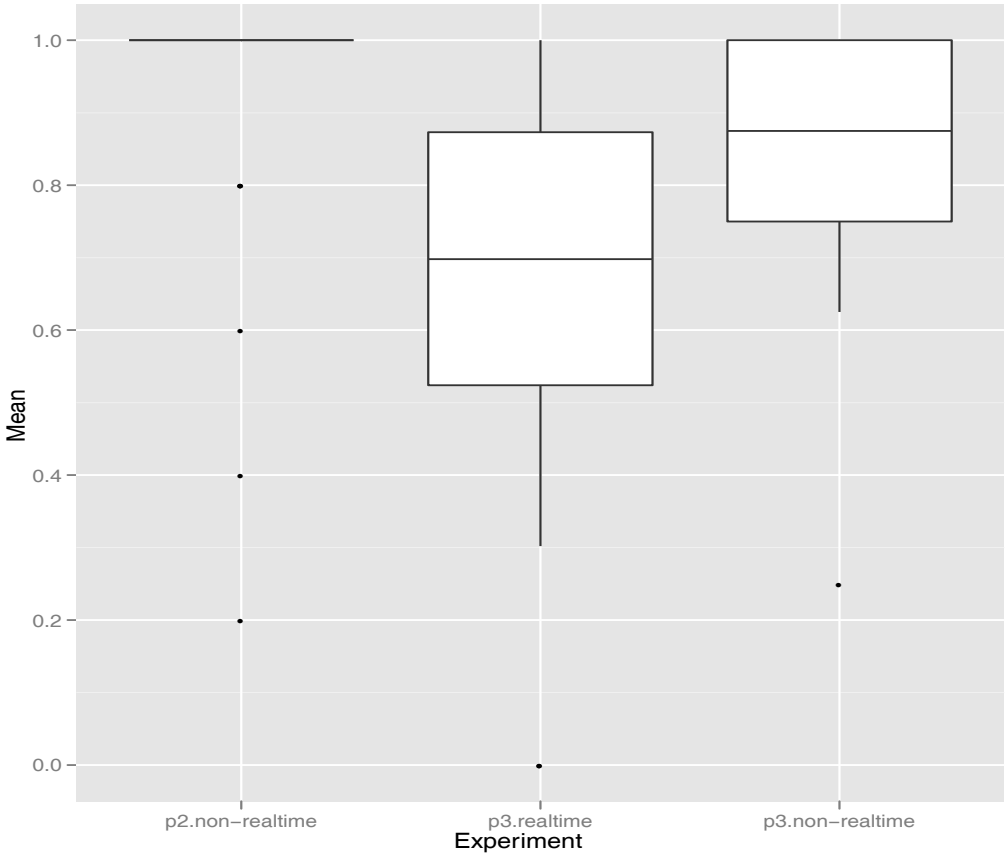
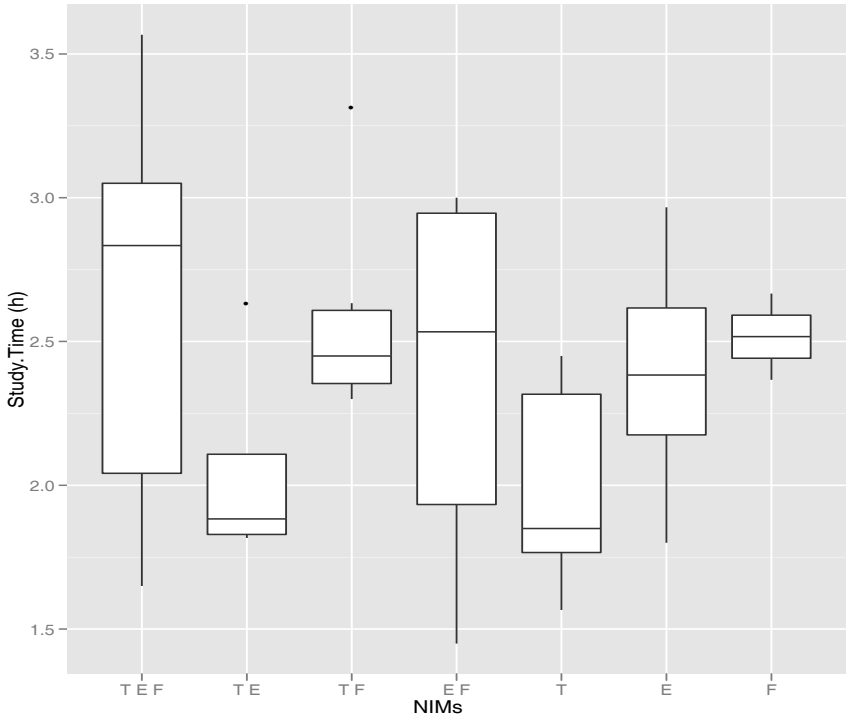


Figure 2: Baseline Plot Test Means

We can compare three groups of baseline subjects: p2.non-realtime, p3.realtime, and p3.non-realtime, denoting students from Phases II and III with and without realtime post-tests, as indicated. These groups contained 28, 12, and 19 subjects respectively. All baseline subjects completed the study in under four hours. Because we increased the difficulty and complexity of the curriculum in Phase III, subjects generally took longer and scored lower than the Phase II subjects (Figure 2). A major revelation in Phase III was the impact of real-time testing on subject post-test scores. We discuss this more in Section 4.2.2.

In Figure 3(a), we show the length of study by NIM-set given. We can see that subjects tended to take less time when given the T NIM, and more time when given the F NIM. In general, the T lessons were shorter than the F lessons, and the F lessons also required subjects to interact with the simulator and encouraged subjects to repeat if necessary.



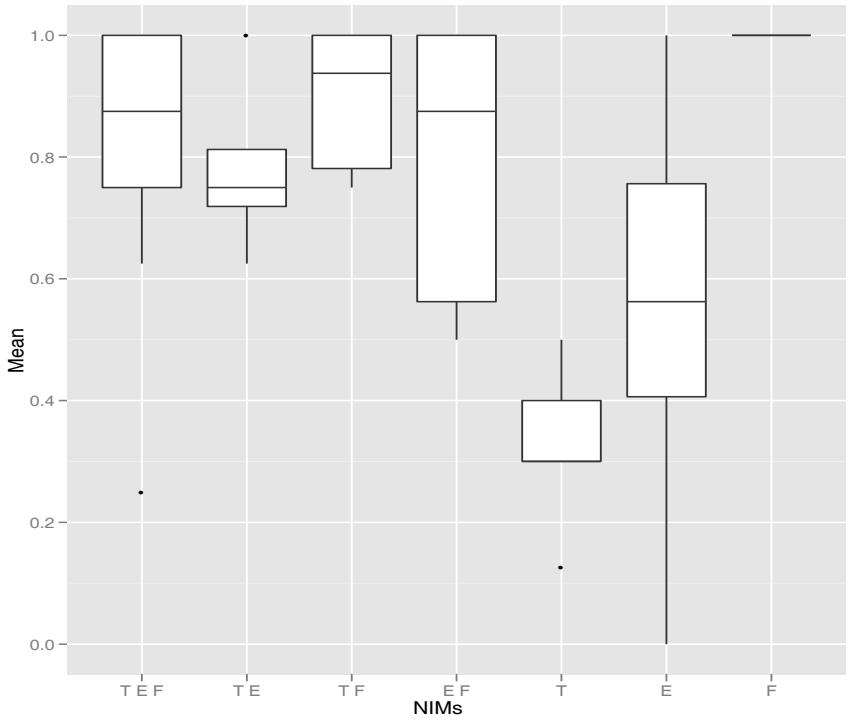


Figure 3(a) Study Time by NIM Set **Figure 3(b) Post Test Mean by NIM Set**

In Figure 3(b), we show mean post-test score by NIM-set. The double-NIM subjects seemed to do almost as well as those given the full set of NIMs. The subjects given the F NIM seemed to do the best of the restricted sets; in fact, all subjects in the F-only group had a perfect post-test score. There are several reasons why this might be the case. This NIM is somewhat a combination of "by feedback" and "by telling"; in the self-paced "by feedback" lessons, if a subject doesn't follow the correct procedure, the subject is given a "by

telling" description of what should have been done. This is also the only NIM where subjects get any practice with the simulator before the post-test.

We also tried allowing subjects of the lowest performing NIM sets (T and E) to ask questions about the curriculum through a restricted interface. Counter to our expectations, it was very difficult to get subjects to ask anything. Finally, by choosing subjects who knew us and who were majoring in relatively non-technical fields, we got a few questions. Even then, the questions were relatively basic questions about definitions, such as a misunderstanding about the meaning of "absolute value". Since these subjects did not end up differing very much from the standard groups, we did not separate them in our results.

4.2.2Qualitative Results

Creating the context and protocols needed to understand and rigorously evaluate the goals of the Bootstrapped Learning project using human teachers and students has been challenging. We offer the following lessons we have learned in our exploration of this design and empirical space.

Human and electronic students differ in fundamental ways that make it difficult to create analogous contexts without providing one side with undue advantages over the other. First, e-students have perfect memory of all lesson material they have seen. We compensate for this in human testing by allowing subjects to take notes and review lessons if desired. Human students also have a harder time interpreting formal language or concepts expressed in other "unnatural" ways. Because of this, we were forced to produce a more natural version of the e-student curriculum for the human students, introducing possible confounding factors into our comparison.

On the other hand, human students have a greater understanding of the semantics of words and have the ability to gain domain knowledge outside the formal channel of the curriculum, such as through voice intonation or gestures inadvertently expressed by a human teacher. We addressed the issue of "leaky" semantics by being careful that our choice of terms didn't leak unintentional knowledge and by going through several preliminary iterations of the curriculum. Interestingly, increased semantic understanding was also occasionally detrimental to human subjects, when the knowledge leaked by terms was misleading. The innovation of the self-paced curriculum was critical for addressing the problem of extra-curricular knowledge transfer.

Increased automation of lesson structures was essential for the greater curriculum complexity and greater number of subjects and groups needed for Phase III. For example, we automated the generation of curriculum and test configurations and the subsequent storing of results and grading.

The issue of decreased scores with realtime testing in Phase III was unexpected. We hypothesize it was because of 1) subject boredom and 2) the issue of training vs. education. When testing subjects with realtime simulators, subjects learned that there were states where nothing occurred and would lose focus; we had to be vigilant for the appearance of web-capable smartphones during this time. Additionally, since stricter time-limits were placed on individual tasks (though the time for all tasks together was actually greater), we believe this may be issue of training (skill gained through repetitive practice) rather than education (knowledge gained through learning). Results on models of human task-performance such as the Human Model Processor [Card et al., 1986] and GOMS [John and Kieras, 1996] may be relevant here. We do not know whether this will be an issue for e-students.

ACKNOWLEDGEMENT

This research was sponsored by AFRL under contract FA8650-07-C-7722. Special thanks to our colleagues at BAE Systems, Cycorp, Inc., Sarnoff Corporation, Stottler Henke Associates, Inc., and Teknowledge Corporation for their collaboration in developing the various BL curricula and testing materials, to Matthew Berland for early work on this project, and to study participants at The University of Texas at Austin.

REFERENCES

Matthew Berland and Dewayne E. Perry. Novice Human Teachers of a Virtual Toddler: A Case Study. Technical report, The University of Texas at Austin, 2009.URL http://users.ece.utexas.edu/~perry/work/papers/090123-MB-blexp1.pdf.

C. Morrison and D. Bryce and I. Fasel and A. Rebguns. Augmenting Instructable Computing with Planning Technology. In *ICAPS'09 Workshop on the International Competition for Knowledge Engineering in Planning and Scheduling*, 2009.

S. Card, T. Moran, and A. Newell. The model human processor- An engineering model of human performance. *Handbook of Perception and Human Performance*, 2:45–1, 1986.

Jon Curtis. BAE Tech Report TR-2231: Bootstrapped Learning Interaction Language. Technical report, BAE, April 2009.

N. Dahlback, A. Jonsson, and L. Ahrenberg. Wizard of Oz studies – Why and How. *Knowledge-Based Systems*, 6(4):258 – 266, 1993. ISSN 0950-7051. doi: DOI:10.1016/0950-7051(93)90017-N. Special Issue: Intelligent User Interfaces.

J. Davis, K. Leelawong, K. Belynne, B. Bodenheimer, G. Biswas, N. Vye, and J. Bransford. Intelligent user interface design for teachable agent systems. In *Proceedings of the 8th international conference on Intelligent user interfaces*, pages 26–33. ACM, 2003. ISBN 1581135866.

A. Frank and A. Asuncion. UCI Machine Learning Repository, University of California, Irvine, School of Information and Computer Sciences, 2010. URL http://archive.ics.uci.edu/ml.

Robert D. Grant, David DeAngelis, Dan Luu, Dewayne E. Perry, and Kathy Ryall. Designing Human Bench- mark Experiments for Testing Software Agents. In *Proceedings of the 14th International Conference on Evaluation and Assessment in Software Engineering* (to appear).BCS eWIC, 2011.

Ralph Grishman and Beth Sundheim. Message Understanding Conference-6: A Brief History. In *Proceedings of the 16th conference on Computational linguistics*, pages 466–471, Morristown, NJ, USA, 1996. Association for Computational Linguistics.doi: http://dx.doi.org.ezproxy.lib.utexas.edu/10.3115/992628.992709.

Bonnie E. John and David E. Kieras. The GOMS Family of User Interface Analysis Techniques: Comparison and Contrast. *ACM Trans. Comput.-Hum. Interact.*, 3(4):320–351, 1996. ISSN 1073-0516. doi: http://doi.acm.org/10.1145/235833.236054.

Dan Oblinger. Bootstrapped Learning: Creating the Electronic Student that Learns from Natural Instruction. AAAI Briefing, 2006. URL http://www.darpa.mil/ipto/programs/bl/docs/AAAI_Briefing.pdf.

R. Rosenthal and R. Rosnow. *Essentials of Behavioral Research: Methods and Data Analysis*. McGraw-Hill Series in Psychology.McGrawHill, New York, second edition, 1991.

R.K. Yin. *Case Study Research: Design and Methods*. Sage Publications, Inc, 2008.