AIED Industry & Innovation Track

Supplementary Proceedings of the 16th International Conference on Artificial Intelligence in Education

AIED
2013
MEMPHIS

Memphis, Tennessee, USA.
July 2013

Industry and Innovation Reports Organization

Track Co-Chairs

W. Lewis Johnson (Alelo Inc.)
Ari Bader-Natal (The Minerva Project)

Program Committee

Jared Bernstein (Pearson)
Chris Brannigan (Caspian Learning)
John Carney (Team Carney)
Jared Freeman (Aptima)
Neil Heffernan (Worcester Polytechnic Institute)
David Kuntz (Knewton)
Garrett Pelton (Carnegie Speech)
Sowmya Ramachandran (Stottler Henke Associates)
Steve Ritter (Carnegie Learning)
Alicia Sagae (Alelo)
Mike Van Lent (Soar Technologies)

Table of Contents

The AIED Industry and Innovation Track

W. Lewis Johnson[1] and Ari Bader-Natal[2]

[1]Alelo Inc.
12910 Culver Bl., Suite J, Los Angeles, CA 90066 USA

[2]The Minerva Project
1145 Market St., San Francisco, CA 94103 USA

Abstract. The new Industry and Innovation Track of the AIED 2013 conference includes submissions from commercial and entrepreneurial organizations that are putting AIED technologies into practice. As digital tutors enter the main stream, and demand increases for advanced capabilities such as automated assessment and personalized learning, there is increasing interest in learning products that incorporate artificial intelligence technologies. The Industry and Innovation Track is intended to attract innovators, practitioners, and technology adopters to the AIED conference to share lessons learned and best practices, and draw on emerging technologies and methods. It includes regular papers and posters, as well as late-breaking reports from fast-moving efforts.

Keywords: Innovation, technology transition, adoption-based research

1 Introduction

Education is in the midst of a period of rapid technological change. New types of online learning resources such as Khan Academy videos (Khan Academy, 2013) and massive open online courses (MOOCs) offer the potential for "flipping" conventional classroom instruction, enabling new paradigms of blended learning, or eliminating brick-and-mortar instruction altogether. As more learning moves on line there is a growing need for tools to track learner progress, personalize curricula, and provide feedback. These are all topics that the AIED community has researched over a number of years, often in research laboratory environments. There is now an unprecedented opportunity to put AIED-based methods into practice on a large scale. This can lead to improved learning solutions. It can also inform AIED research through access to real data and experience with real learning problems.

The Industry and Innovation Track of AIED aims to bring together researchers, practitioners, and innovators in the education space to share experiences related to putting AIED technologies into practice. We recruited a program committee of industry leaders and individuals experienced with applying learning technologies, who could bring an industry perspective to the evaluation process. Because commercial

adfa, p. 1, 2011.

efforts tend to move rapidly and aim for quick results, we included a late-breaking reports category with a reduced time between submission and publication.

Like many learning innovations, the AIED Innovation and Industry Track is an iterative work in progress. The number of contributions this year is relatively small, but includes several interesting contributions from a cross-section of industrial research laboratories, government agencies and commercial enterprises engaged in educational innovations. We will draw lessons from this pilot effort and use them to grow the industry-and-innovation component of the AIED conference in future years.

2 Contributions

Two contributions to the Industry and Innovation Track are included in this proceedings volume. Melinda Gervasio and Karen Myers of SRI International report on an automated capability for assessing procedural skills, developed to support training for a software system in widespread use across the US Army. Jeremiah Folsom-Kovarik and Robert Wray report on their work on adaptive assessment algorithms, which will enable adaptive assessment in real-world training settings where calibration data is sparse. A third paper by Brian Vogt of the US Army was also accepted, on the topic of a methodology for assessing scenarios in the UrbanSim strategy game. Unfortunately Mr. Vogt is unable to attend AIED and present the paper.

There are also three late-breaking reports, which will be published in a separate volume at the conference. Brian Duffy and team at Team Carney report on a case study of gamification of traditional courseware. Lewis Johnson gives an interim report on Alelo's Tactical Interaction Simulator, and current efforts to integrate it into instruction at the Defence Forces Language School in Australia. Finally Jennifer Sabourin and team at the SAS Institute report on their SAS® Read Aloud app for early reading, and discuss opportunities for incorporating intelligent technologies to further improve and understand early literacy reading.

References

1. Khan Academy (2013). A free world-class education for anyone anywhere. Retrieved from http://www.khanacademy.org/about

Drill Evaluation for Training Procedural Skills

Melinda Gervasio and Karen Myers

SRI International, 333 Ravenswood Ave., Menlo Park, CA

Abstract. The acquisition of procedural skills requires *learning by doing*—students learn by trying to solve problems, getting feedback on mistakes, and requesting assistance in the face of impasses. This paper describes an automated capability for assessing procedural skills that was developed to support training for a complex software system in widespread use throughout the U.S. Army. The automated assessment uses a soft graph matching capability to align a trace of student actions to a predefined gold standard of allowed solutions, providing a basis to assess student performance, identify problems, give hints for improving performance, and indicate pointers to relevant tutorial documentation.

Keywords: procedural skills, automated assessment, relaxed graph matching

1 Introduction

Today's workers require a broad and growing set of *procedural skills,* which involve learning multistep procedures to accomplish a task. Procedural skills apply both to physical environments (e.g., how to repair a device) and online environments (e.g., how to create a pivot table in Excel).

This paper reports on a system called Drill Evaluation for Training (DEFT) that was developed to facilitate the learning of procedural skills related to the use of a complex piece of software. More specifically, DEFT provides an automated assessment capability to evaluate student performance as they learn how to use the Command Post of the Future (CPOF)—a collaborative geospatial visualization environment system used extensively by the U.S. Army to develop situational awareness and to plan military operations. Although a powerful tool, CPOF is difficult to learn; furthermore, CPOF skills decay rapidly when not in regular use. Because soldiers have limited availability for formal training sessions, the result is that many users struggle when using CPOF in the field.

DEFT addresses the training problem for CPOF by providing significant automated support to assess learned skills. Having the ability to automate assessment of student performance would reduce the burden on instructors in classroom settings, thus enabling them to provide more personalized attention to individual students. It would also enable students to pursue independent supplemental training beyond a formal classroom setting.

XXX, p. 1, 2013.

We begin the paper with some additional background on CPOF, followed by a technical overview of DEFT. We then present results of a user study that assessed the usability and utility of DEFT for CPOF training. We close with a discussion of related work, a summary of contributions, and directions for future work.

2 Command Post of the Future (CPOF)

CPOF is a state-of-the-art command and control (C2) visualization and collaboration system. The CPOF software is part of the U.S. Army's Battle Command System, and as such is standard equipment for virtually every Army unit. Since its inception in 2004, thousands of CPOF systems have been deployed. Its usage spans organizational echelons from Corps to Battalion in functional areas that include intelligence, operations planning, civil affairs, and engineering. CPOF is used extensively to support C2 operations for tasks covering information collection and vetting, situation understanding, daily briefings, mission planning, and retrospective analysis [4].

CPOF uses geospatial, temporal, tabular, and quantitative visualizations specifically tailored to information in the C2 domain. Users can collaborate synchronously in CPOF by interacting with shared products. The ability to dynamically incorporate new information is critical to the success of any C2 operation; CPOF's "live" visualizations continually update in response to changes sourced from user interactions or underlying data feeds, thus ensuring that data updates flow rapidly to users.

The U.S. Army offers the Battle Staff Operations Course (BSOC) to provide instruction to students on basic CPOF interaction skills. Much of what is taught in the BSOC is procedural, i.e., determining what steps to perform and in what order to achieve a particular result. The following provides a portion of an exercise from the BSOC course materials: *Create a 2D map. Create a notional unit; name it A10 #X 1v2. Edit the size, type, and affiliation. Place the unit on the 2D map.*

An analysis of an examination used to test student mastery of BSOC material showed that 69% of the questions required demonstration of procedural skills; another 6% involved true/false or multiple-choice questions; the remaining 25% required short-answer responses. Similar exercises are used within the course itself to enable students to apply the classroom knowledge in a hands-on fashion. This predominance of procedural skills within the BSOC curriculum motivated the development of DEFT, as having an ability to automatically assess student performance could dramatically alter the manner in which CPOF training is conducted.

3 DEFT Technical Components

DEFT performs real-time monitoring of students as they attempt to complete exercises (see Fig. 1). While a student works on an exercise, DEFT logs a trace of his actions. That trace is compared to a representation of allowed solutions to the exercise (the *gold standard*) to create assessment information that identifies conceptual errors or mistakes, provides guidance in the form of hints to help the student complete a task, and suggests links to contextually relevant training materials.

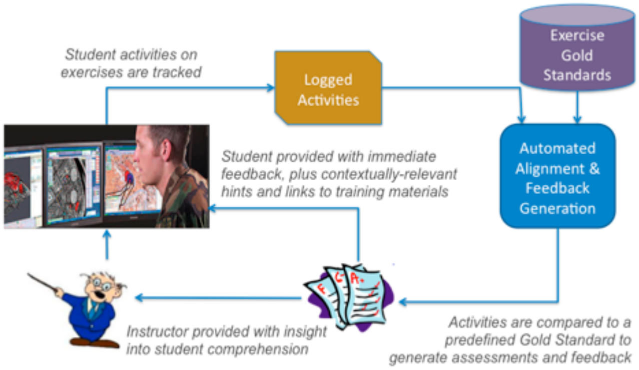

Student activities on exercises are tracked
Logged Activities
Exercise Gold Standards
Student provided with immediate feedback, plus contextually-relevant hints and links to training materials
Automated Alignment & Feedback Generation
Instructor provided with insight into student comprehension
Activities are compared to a predefined Gold Standard to generate assessments and feedback

Fig. 1. Automated assessment in DEFT

3.1 Gold Standard Representation

The *gold standard* defines the space of acceptable solutions to an exercise. For BSOC exercises, solutions cannot be readily specified by enumeration as there can be numerous approaches to completing a task that may involve different actions and orderings between them, and significant variability in the specific objects that are created or manipulated by the actions.

We represent the gold standards as one or more traces obtained through demonstrations of correct solutions to an exercise, augmented with additional annotations that define allowable variations from the trace. A gold standard defines a partial ordering on the steps of a trace, where a step can be a (parameterized) CPOF action, a class of actions, or set of options, each of which is itself a partially ordered set of steps. The annotations take the form of constraints over steps or parameters. Currently, DEFT supports action ordering constraints, parameter equality constraints, parameter value constraints (between parameters and constant values), and a limited set of query constraints. Query constraints are intended to capture requirements on the application state or on object properties that cannot be determined from the arguments of the actions themselves. The abstractions provided by this scheme can result in significantly more compact representation of potentially very large solution spaces.

We anticipate that instructors will play a critical role in gold standard development by providing solution traces and annotating them. However, we can also leverage automated reasoning and machine learning techniques to facilitate the process. For example, we can apply heuristics to determine default annotations and generalize over parameters and actions from multiple examples.

3.2 Alignment

The automated assessment capability in DEFT centers on determining a mapping from the student's submitted response for an exercise to the predefined gold standard for that exercise. We have framed this alignment problem as a form of inexact semantic graph matching in which a similarity metric based on graph edit distance is used to rate the quality of the mappings. Graph edit distance measures the number—more generally, the cumulative cost—of graph editing operations needed to transform the student response into an instance consistent with the gold standard. Intuitively, finding the lowest-cost alignment corresponds to DEFT finding the specific solution the student is most likely to have been attempting.

To use this graph matching approach in DEFT, we represent the gold standard as one or more pattern graphs, with each graph representing a family of possible solutions to the exercise. Actions and their parameters are nodes; parameter roles within actions are links; and required conditions within the solution (precedence between actions, values of textual or numerical parameters, etc.) are constraints. The student response is represented similarly as a candidate graph.

Alignment involves finding the mapping between the candidate and a pattern with the lowest edit distance cost. We associate costs that impose a penalty in the score for the response for missing the respective action, parameter, constraint, etc. Alignment to the closest solution allows DEFT to generate an assessment that identifies differences between the response and the gold standard, which translate both to specific errors the student has made (e.g., out-of-order actions, incorrect action parameter values, missing or extra actions) and to the corrections needed.

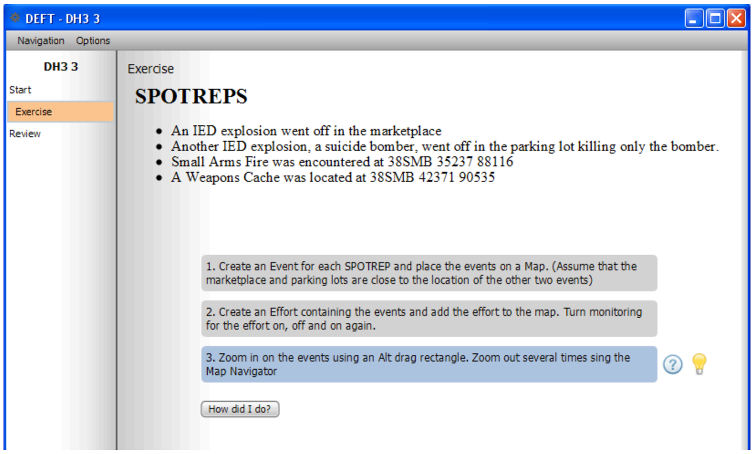
The alignment capability in DEFT builds on a pattern matching algorithm that was developed originally for link analysis applications [10]. While this algorithm provided a reasonably good fit for solving the alignment problem, we developed a set of performance optimizations linked to the structure of our specific matching problem that significantly prune the overall search space.

3.3 Student Interface

DEFT's student interface serves two functions. First, it provides a framework for exercise administration: presenting exercises for selection, supporting navigation through the exercises, and making available contextually relevant hints and documentation links. Second, it presents students with visual feedback on their solutions that shows problems detected by the automated assessment capability.

When a user selects an exercise, he is presented with background information on the exercise from the BSOC training materials, including a statement of the learning objectives for the exercise and links to relevant study materials. The user can begin the exercise by clicking on a *Start* button on the bottom of the screen. The exercise is presented to the student incrementally as a sequence of numbered tasks. For example, Fig. 2 shows the three tasks that comprise an exercise related to Spot Reports. The user performs actions in CPOF to complete each task in turn, with instrumentation logging those actions. Upon completing a task, the user clicks on a button at the bottom of the screen to proceed to the next task.

Users are presented with context-sensitive hints (accessed via the light bulb icon) and documentation links (accessed via the question mark icon) to facilitate their completion of tasks. DEFT uses hint sequences, with initial hints providing high-level guidance and subsequent hints progressively disclosing more complete directions for the task. Clicking on a documentation link displays the relevant section of the online CPOF documentation in a Web browser. After completing all tasks, the user can click on the *'How did I do?'* button to view the DEFT assessment of his performance.



Exercise

SPOTREPS

- An IED explosion went off in the marketplace
- Another IED explosion, a suicide bomber, went off in the parking lot killing only the bomber.
- Small Arms Fire was encountered at 38SMB 35237 88116
- A Weapons Cache was located at 38SMB 42371 90535

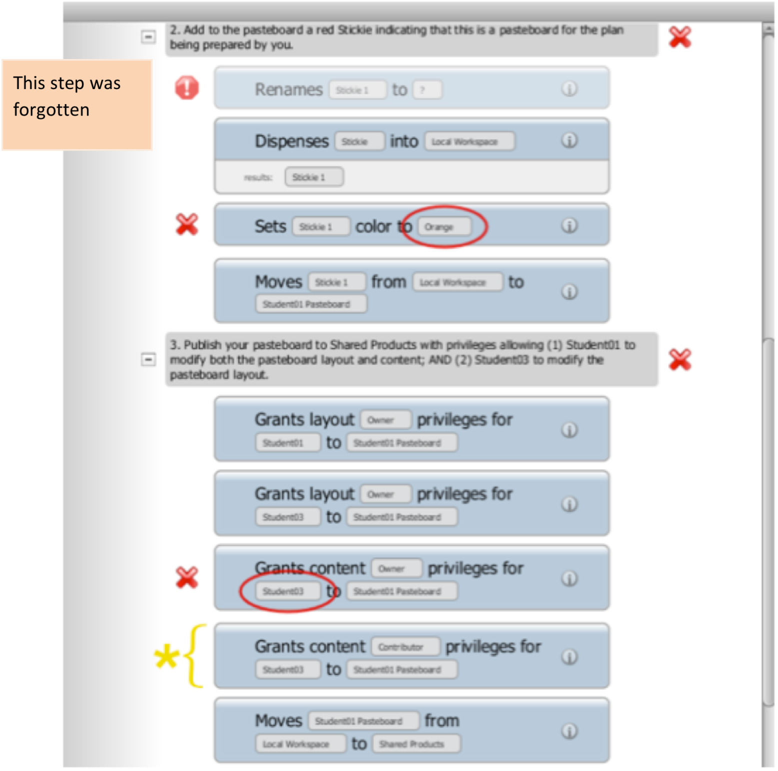
1. Create an Event for each SPOTREP and place the events on a Map. (Assume that the marketplace and parking lots are close to the location of the other two events)

2. Create an Effort containing the events and add the effort to the map. Turn monitoring for the effort on, off and on again.

3. Zoom in on the events using an Alt drag rectangle. Zoom out several times sing the Map Navigator

Fig. 2. Student interface: task structure for an exercise

Fig. 3. Sample feedback from a BSOC exercise



This step was forgotten

2. Add to the pasteboard a red Stickie indicating that this is a pasteboard for the plan being prepared by you.

Renames Stickie 1 to ?

Dispenses Stickie into Local Workspace

results: Stickie 1

Sets Stickie 1 color to Orange

Moves Stickie 1 from Local Workspace to Student01 Pasteboard

3. Publish your pasteboard to Shared Products with privileges allowing (1) Student01 to modify both the pasteboard layout and content; AND (2) Student03 to modify the pasteboard layout.

Grants layout Owner privileges for Student01 to Student01 Pasteboard

Grants layout Owner privileges for Student03 to Student01 Pasteboard

Grants content Owner privileges for Student03 to Student01 Pasteboard

Grants content Contributor privileges for Student03 to Student01 Pasteboard

Moves Student01 Pasteboard from Local Workspace to Shared Products

Fig. 3 shows sample feedback generated by DEFT. An icon to the right of each subtask indicates whether the subtask was completed successfully (green checkmark), contained mistakes (red x), or triggered warnings (yellow checkmark). An icon to the left of a step denotes a specific type of problem with that step. Hovering on the icon presents a textual description of the problem (e.g., the orange box in the figure). Possible problem types include incorrect step values (red X and red circle on incorrect value), a missing step (red exclamation mark beside a grayed-out step), an unnecessary step (yellow asterisk), and incorrect ordering of steps (not shown here).

4 User Study

We conducted a user study to evaluate DEFT's ability to provide students with correct and comprehensible feedback regarding their performance on exercises derived from the BSOC training material. We originally intended to conduct the study with active duty soldiers, but because of their limited availability, we instead recruited ten participants from SRI, none with military backgrounds, spanning a variety of job roles including administrative assistants, technical editors, and project administrators. None had previous exposure to CPOF so they were given a two-hour hands-on CPOF training session the week before the study.

4.1 Methodology

The user study comprised ten individual participant sessions, each lasting two hours. Each session involved the participant, a facilitator, and a note-taker; and was conducted in three parts. First was a 15-minute introduction to the use of DEFT to perform exercises in CPOF. The participant was guided by the facilitator in performing an exercise and introduced to the hints and online help mechanisms. Second was a 75-minute think-aloud session during which participants were asked to think aloud as they performed exercises on their own and viewed DEFT's assessments of their solutions. They were also presented with assessments of erroneous solutions handcrafted to include various types of errors. Finally was a 30-minute debrief where the participant was asked to complete two brief questionnaires and then engaged in an open discussion. The first questionnaire was a standard questionnaire for calculating System Usability Scale (SUS) scores [3]; the second was a compilation of questions regarding computer usage. The open discussion was structured around "product response cards" [2], a set of 55 adjectives (positive and negative) from which the participants were asked to select five that best described what they thought of DEFT and then to elaborate on their selections.

4.2 Results

Demographics. All ten participants self-reported being "comfortable" or "very comfortable" with the use of computers. On the questions regarding computer and software use, on a scale of 0 to 4 (where 0 = never and 4 = very often), they averaged

3.22 on online activities, 2.73 on office applications (e.g., word processing, spreadsheets), 1.67 on games, and 0.56 on advanced computer use (e.g., programming, sound/video editing). Six reported having taken a programming class at some point, but none were active programmers. All reported having taken a computer-based training or online course.

Automated Assessment. Each participant completed two to three exercises and viewed two to three additional assessments within the time allotted. Performance on the exercises varied greatly, with some completing exercises with few errors or none at all, while others struggled on all exercises. The instructions in the exercises were intentionally designed to elicit some errors and all the participants committed at least a few errors. DEFT's automated assessment module was able to correctly identify all the errors except in two situations where the system crashed due to unanticipated CPOF instrumentation issues. All the participants were able to correctly interpret the error feedback on their solutions and, in the cases where they were asked to repeat an exercise, to correct their mistakes. Everyone was also able to interpret assessments of the handcrafted erroneous solutions but required more effort to do so because of the additional need to interpret someone else's solution.

However, based on the results of the think-aloud sessions and the discussions afterwards, it was apparent that most participants found the assessment visualizations too busy or too long. Several stated that they would prefer a simple textual rendering, with a few suggesting just a summary of the results. One participant found DEFT's focus on error feedback (i.e., only errors were pointed out) to be particularly harsh and suggested providing positive feedback as well. Many also wanted not just to be told what they had done wrong but also to be directed on how to fix it.

The perceived deficiencies of the assessment visualization were somewhat surprising, given that we had designed them in close collaboration with CPOF instructors. However, we realized that instructors and students have distinct needs. For an instructor, who needs to see the performance of an entire classroom, seeing individual user responses and high-level assessments in the form of markups (checkmarks, Xs, and circled elements) is especially valuable. In contrast, students already know what they did and are more interested in the assessment itself. A recap of what they did and an overall view of how they did is much less useful than a report on how they did and, if they made errors, what they need to do to fix them.

Exercise Administration. The study also provided the opportunity to evaluate DEFT's exercise administration functionality. Participants found the DEFT workflow of loading an exercise, performing a sequence of tasks, and getting an assessment to be straightforward. However, a few expressed a desire for more immediate feedback to help guide them through an exercise. There were a number of situations where a participant started floundering and was then unable to make progress without intervention from the facilitator.

DEFT's task-specific hints and links to online help were perceived by all participants to be valuable and everyone relied on them at some point. Although a few tasks involved CPOF concepts that the participants had not been or were only briefly exposed to during their CPOF training, most were able to use the hints and help to accomplish the tasks anyway. Most participants preferred the brevity and directness of hints, often finding the online CPOF documentation to be overwhelming.

Usability and Usefulness. The SUS scores ranged from 35 to 90, with a mean of 61.25 and a median of 62.5 (scores that can be interpreted to mean roughly "average"). There are too few participants to draw statistically significant conclusions; however, together with our observations during the think-aloud sessions and the open discussions with the participants, these results indicate that although the participants found DEFT easy to use, there remain gaps in its exercise administration and automated assessment capabilities.

In the product response cards exercise, participants were asked to choose the five words best describing what they thought of DEFT. The results (Fig. 4) reveal that participants had a predominantly positive response to DEFT, with several describing it as "useful", "straightforward", "relevant", and "valuable". A few participants found DEFT "frustrating"; further probing revealed that their reaction was at least partly due to their lack of familiarity with CPOF and with military terminology in the exercises.

Across the board participants expressed their belief that DEFT was a valuable training tool. They appreciated its tight integration with the training application (CPOF, in this case). All the participants readily suggested examples where they thought a tool like DEFT could be useful for training. These included various procedures they had encountered in their work, such as accounting processes, website navigation, webpage creation, and timecard management; as well as more unusual suggestions such as learning a new language or how to play an instrument.

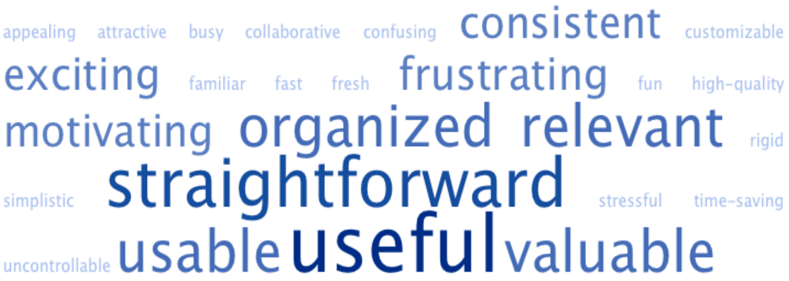


Fig. 4. Tag cloud depicting subjective participant response to DEFT, with word size reflecting the number of times in appeared in participants' Top 5 lists.

4.3 Discussion

The user study provided valuable feedback and encouraging results regarding DEFT as a training tool for procedural tasks. It is notable that although the participants in the study were complete novices in both the application (CPOF) and the domain (military operations), they were able to use DEFT to complete real training exercises in CPOF. And in spite of the difficulty in performing a task (encountered by most of the participants at some point during the study), the participant response to DEFT was predominantly positive. However, as a prototype system whose primary focus has been on automated assessment, DEFT has room for improvement. In particular, to be an effective tool for self-directed learning, it needs to provide more student-focused interactions, including a tighter integration between performance, assessment, and correction; and more comprehensive, focused, explanatory feedback.

5 Related Work

Example-tracing tutors [1] assess procedural skills by comparing student actions against a *behavior graph* that represents all acceptable ways of achieving a task, much like DEFT compares student solutions against a *gold standard*. Both behavior graphs and gold standards capture a range of solutions by allowing alternative actions, ranges of values used in actions, and alternative action orderings. However, because an example-tracing tutor's primary task is to *teach* a procedural skill, its assessment is focused on recognizing what the student is trying to do and ensuring that the student remains on track to successful accomplishing a task. In contrast, DEFT is designed primarily to *assess* how well a student has performed a skill and is thus focused on identifying key mistakes in the student solution.

This distinction also applies when comparing DEFT to model-tracing [6,9] and constraint-based tutors [7]. In addition, model-tracing tutors are designed for domains such as math and physics where automated problem-solvers can be developed; they are less applicable to open-ended domains like CPOF. Meanwhile, constraint-based tutors are designed for tasks where the challenge is not s in the selection of actions and parameter values but in the selection of values that satisfy potentially complex constraints. Although CPOF requires capturing such constraints as well, the variety of actions available to accomplish a task requires evaluating the procedures themselves.

In *programming*, assessment can be performed entirely on the end product (the program): whether it produces the correct results, meets complexity and style criteria, is efficient, etc. [5] To some extent, such assessment can be performed on the final information products in CPOF but the real-world need for efficient operation and adherence to best practices demands assessment of how products are created as well.

6 Conclusion and Future Work

Several CPOF instructors have enthusiastically endorsed our automated assessment concept, noting benefits of the technology on several levels. In a classroom setting, it would enable high achievers to move more rapidly through a curriculum, potentially exploring challenge concepts beyond the baseline skills required for the entire cohort; for students for whom the curriculum poses a greater challenge, the technology would provide real-time, personalized feedback. The instructors were also excited by the prospect of being able to track individual and aggregate student performance to help them identify concepts that are problematic for students and to adjust their instruction accordingly. Importantly, the technology opens the door to supporting student-directed acquisition of skills outside of the classroom.

The automated assessment capability in DEFT is currently a research prototype. Given the encouraging results from the user study and the strong desires expressed by

CPOF trainers for a capability of this type, we believe that it would be valuable to continue this line of work with the objective of generating a fully operational assessment capability that could be deployed to enable self-directed CPOF training.

To date, gold standards for the BSOC exercises have been hand-coded by members of our research team. Ideally, curriculum developers would be able to construct gold standards on their own. For this, we envision a tool that would enable an instructor to demonstrate the procedural structure of an exercise solution, augmented with an annotation mechanism for specifying the companion constraints that define allowed variations from the demonstration. We believe it would be feasible to develop such an authoring tool, leveraging learning by demonstration technology we have previously deployed within CPOF to enable automation of routine tasks [8].

7 References

1. Aleven, V., McLaren, B., Sewall, J., and Koedinger, K. A new paradigm for intelligent tutoring systems: Example-tracing tutors. *Intl. J. of AI in Education,* 19(2), 105-154 (2009)
2. Benedek, J. and Miner, T. Measuring desirability: New methods for evaluating desirability in a usability lab setting. *Proc. of the Usability Professionals Assoc. Conf.* (2002)
3. Brooke, J. SUS—a quick and dirty usability scale. Brooke, J. In P. W. Jordan, B. Thomas, B. A. Weerdmeester, and A. L. McClelland (Eds.), *Usability Evaluation in Industry*, London: Taylor and Francis (1996)
4. Croser, C. Commanding the Future: Command and Control in a Networked Environment, *Defense & Security Analysis,* 22(2) (2006)
5. Douce, C., Livingstone, D., and Orwell, J. Automatic test-based assessment of programming: a review. *ACM Journal of Educational Resources in Computing*, 5(3) (2005)
6. Koedinger, K. R., Anderson, J. R., Hadley, W. H., & Mark, M. A. Intelligent tutoring goes to school in the big city. *Intl. Journal of AI in Education,* 8, 30-43 (1997)
7. Mitrovic, A. NORMIT: a web-enabled tutor for database normalization. *Proc. of the Intl. Conf. on Computers in Education*, 1276-1280 (2002)
8. Myers, K., Kolojejchick, J., Angiolillo, C., Cummings, T., Garvey, T., Gaston, M., Gervasio, M., Haines, W., Jones, C., Keifer, K., Knittel, J., Morley, D., Ommert, W. and Potter, S. Learning by Demonstration for Collaborative Planning. *AI Magazine*, 33(2), 15-27 (2012)
9. VanLehn, K., Lynch, C., Schulze, K., Shapiro, J. A., Shelby, R., Taylor, L., Treay, D., Weinstein, A., and Wintersgill, M. The Andes physics tutoring system: lessons learned. *Intl. J. of AI in Education*, 15:3 (2005)
10. Wolverton, M., Berry, P., Harrison, I., Lowrance, J., Morley, D., Rodriguez, A., Ruspini, E. and Thomere, J. LAW: A workbench for approximate pattern matching in relational data**.** *Proc. of the 15th Conf. on Innovative Applications of AI* (2003)

Adaptive Assessment in an Instructor-Mediated System

Jeremiah T. Folsom-Kovarik, Robert E. Wray, Laura Hamel

Soar Technology, Inc.

{jeremiah.folsom-kovarik,wray,lhamel}@soartech.com

Abstract. *Instructor-mediated* training systems give end users direct control over instructional content, increasing acceptance but introducing new technical challenges. Decreased opportunities for parameter estimation (or manual setting) limit the utility of item-response or Bayesian approaches to adaptive assessment. We present four adaptive assessment algorithms that require little data about characteristics of test items. Two algorithms present about half as many test items as random selection before producing accurate skill estimates. These algorithms will enable adaptive assessment in real-world training settings where calibration data is sparse.

Keywords: assessment, adaptive training, instructor-mediated design

1 Introduction

Instructor-mediated design is a pattern the authors use to help ensure training systems fit practitioner needs. The goal is to increase instructors' control over the content and operation of a training system and is in contrast to systems where instructional or content changes require technical expertise or formal specification [1]. Giving instructors direct control over how their training systems work can improve system acceptance and effectiveness in real-world use. It can reduce costs, turnaround time for changes, and the errors introduced during communication between end users and developers. However, when adaptive elements are complex, instructor-mediated design can place technical burdens on instructors or, more likely, result in incompleteness and incorrectness in the authored system. To enable instructor-mediated design for adaptive training, adaptation should be simple and transparent [2].

Formal adaptive assessment includes several related Bayesian and Item Response Theory (IRT) approaches [3-5]. They are well studied both theoretically and in real-world applications, for example in the widely administered Graduate Record Examination [6]. However, these approaches have drawbacks for some adaptive training uses. They require specification of multiple important values that are nuisance parameters from an instructor point of view, such as prior beliefs about learner ability and item discrimination or difficulty. Principled machine learning and calibration of these parameters necessitate large amounts of empirical data, on the order of 1800 people or more answering each test item [7]; there is a possibility of incorrect outcomes before the model saturates; and the learned parameters are sensitive to small changes in item content or context [8]. Approximate methods such as [9] can reduce but not eliminate these requirements. Whether they are learned from data or set by authors, the number and precision of model parameters in these approaches are barriers to transparency, instructor acceptance, and quick changes to content.

To combine adaptive assessment with instructor-mediated design, we investigated transparent selection algorithms that would not require large amounts of calibration data. If these algorithms could adapt to individual students, our system would let instructors understand their students more quickly and in better detail. We developed simulated students to empirically evaluate selection algorithms under a range of circumstances and found a simple algorithm can choose effectively between skills to test, an important task for real-world adaptive training and assessment.

2 Adaptive Assessment

We are exploring the challenge of readiness assessment in the context of a US Navy course that trains tactical decision-making skills for mid-career officers. Officers entering the course may have very different prior knowledge and experience. For example, an officer previously stationed on a mine hunting vessel may have had little exposure to anti-submarine concepts. The instructors must identify specific knowledge and skill gaps for individual incoming students, as efficiently as possible. Specific requirements make a traditional test design approach less effective:

1. Instructors are primarily interested in skill evaluation, rather than conceptual knowledge. We developed a testing approach inspired by Kalyuga and Sweller's [10] rapid skill testing method. Students see a partially "solved" tactical situation in a simulation and must provide a summary of their next steps within a few seconds [11]. These simulations and rapid skill tests can be authored by instructors and do not require technical intervention or study before use. Therefore, there is no opportunity to carefully characterize individual test items before presenting them to students. Further, the test item pool is small.
2. Instructors create and modify relationships test items and skills, using their perspectives on the structure of the underlying learning domain. We seek to help expert instructors express their domain understanding rather than fit their experience into a skill taxonomy we define. Because instructors establish skill relationships, it is not practical to rely on precise weights in a skill network or complex inference methods that otherwise could help interpret test items.
3. Because test items quickly become outdated in a changing tactical environment, there is limited reuse of test items over time. We estimate that an individual test item might be presented to between 50 and 500 students before being retired. Compared to approaches that evaluate test items with many learners over time, the limited item reuse provides little opportunity to estimate item characteristics such as discrimination or reliability.

In order to address these needs, we studied four adaptive assessment algorithms (**Fig. 1**). The candidates were chosen for their minimal instructor input requirements, potential to work with small amounts of data, and their transparency to instructors.

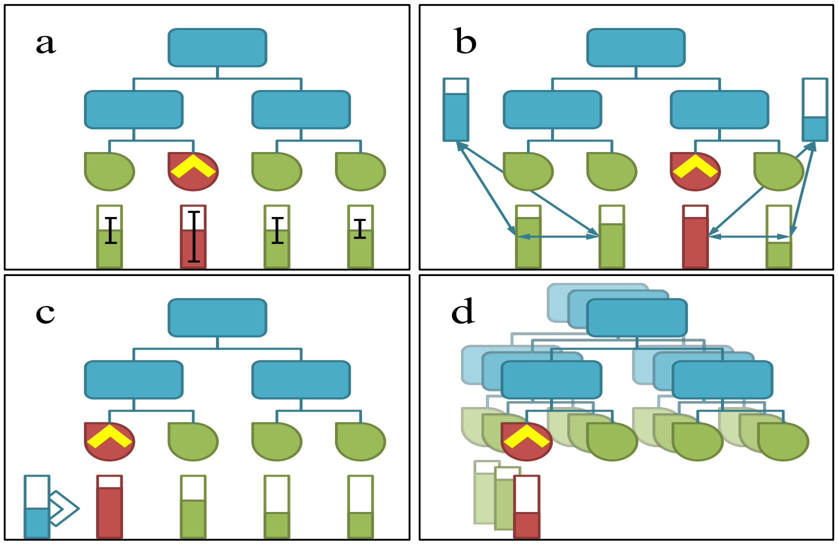


Fig. 1. Four adaptive item selection candidates: (a) Select (yellow arrow) the skill with the least confident estimate. (b) Select an item to present by comparing with neighbors. (c) Model a single proficiency value. (d) Compare skill proficiency to the rest of the student population.

Least confidence: This algorithm asks more questions about skills for which the system has *least certainty*, whether because of mixed student performance or possible problems with test items. Sample variance is an easily calculated proxy for estimate certainty and does not require configuring multiple nuisance parameters as do Bayesian certainty measures like posterior confidence intervals. It is mathematically related to information-theory certainty measures like entropy and mutual information, but its meaning is more accessible to instructors. Sample variance is calculated for each skill proficiency estimate for each student. The skill with the least certainty in the student's proficiency estimate (the highest sample variance) was selected for testing.

Neighbor divergence: This adaptive algorithm uses the skill tree topology to compare proficiency estimates of each skill with those of skills near it. Each skill was compared with its parents, children, and siblings in order to find the average absolute difference between the node and its neighbors. The node with the greatest difference from its neighbors was selected. The motivation is that *local outliers* in proficiency estimates may indicate errors introduced by, for example, a lucky guess. To the extent that the skill tree topology reflects real relationships between individual skills, proficiency in one skill should help to predict proficiency in closely related skills.

Overall divergence: This algorithm is a variation of the second algorithm. It still concentrates on outliers in estimate means, but compares node estimates to the overall (domain) skill estimate of the learner. Student performance on all skills contributed equally to updating the proficiency average, and the average weighted each test item equally (meaning that skills that were tested with more items contributed greater weight). To select a new test item to present, each skill was ranked by its difference from the overall average. The most distant skill was then selected.

Population divergence: This algorithm uses a simple model of the *population distribution* of ability across all simulated students and chooses test items that test a skill that differs most from the population mean for that skill. Ability in real students often follows a unimodal distribution such as a normal or Poisson distribution. Thus, it may be advantageous to estimate the population mean and identify which student estimates are far from that point. All else being equal, exceptional estimates are more likely to represent sampling error than estimates that are closer to the center of the distribution. In testing with simulated students, the distribution of ability in the population would indeed (approximately) follow a normal distribution. However, this assumption is also widely made in training and educational settings, where it is often the basis of grading curves and tests for significance. In addition, the simulated distribution of ability actually was not wholly unimodal. The population had a somewhat bimodal distribution because a small but significant proportion of simulated students were completely unable to perform certain skills, as if they had never learned a particular topic.

Baselines: Finally, we compared our adaptive algorithms to two baselines. These two, termed *random selection* and *perfect knowledge* bracket the results [12]. Random ordering bounds the low end of performance and, because item selection is not systematic in the current system, may be an apt estimate of current assessment efficiency. To find the upper bound on performance, we also compared candidate algorithms against item selection with perfect knowledge of the true underlying proficiency values for each skill, enabling the system to choose the most apt question at all times.

3 Method

We now present a series of experiments designed to study the performance of the candidate algorithms across a range of realistic instructional and trainee characteristics. First, we evaluated the algorithms in a simulation based on the readiness assessment use case. Second, we evaluated performance when the material being tested has a range of different underlying structures.

Simulated Instructional Material. We created abstract simulations of instructional material that could generalize to a wide variety of instructional use cases. We defined test items as reflecting student skill according to a modified three-parameter logistic model widely used in IRT research and practice [13]. We modified the model by fixing two of the parameters indicating item discrimination and probability of guessing. The effect is that all questions targeting a particular skill were interchangeable; item selection represents a decision of which skill to test. As above, in actual use there is not sufficient test data to characterize effectively instructor-authored questions.

Skills were arranged into hierarchies that describe how different skills relate to each other. Parent nodes represent higher-order skills that integrate the skills of child nodes. We tested four arrangements of skills, labeled A through D (**Fig. 2**). In topology A, skills are arranged at varying depths from the root of the tree, a typical configuration that reflects observed usage. In topology B, all skills are children of one parent node. A hierarchy like this might arise legitimately when many component skills are required to carry out one task or by accident when authors define skills without carefully considering their relationships. Topology C tests how algorithms function with a forest of unrelated skill trees. Because there are several root nodes, the skills can be viewed as unrelated, such as the case when a system must evaluate all three of perceptual skill, math aptitude, and reading ability. Finally, topology D tests algorithms on a highly interrelated skill graph. This skill graph might easily arise when a single skill pertains to several parents. For example, skilled marksmanship may be expected to improve performance on a firing range and also in combat exercises.

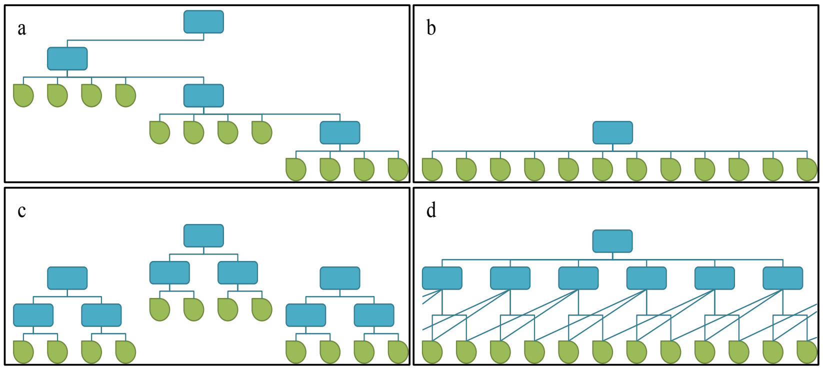


Fig. 2. Relationships between tested skills in the simulation environment. In topology (a), skills were related at varying depths in a hierarchical arrangement. Topology (b) represents a flat skill tree while topology (c) represents a forest of unrelated skill trees. In topology (d), skills are interrelated and have multiple parents.

In experiments each item tested only one skill, a principle of good test design [14] that our readiness assessment follows. Skills could have multiple test items. Test items existed for every skill, and more items were available to test skills at the leaves of a tree than skills at internal nodes. In our experience, instructors often find it easier to create test items that target a low-level skill, such as a specific procedure, and more difficult to construct items that test an integrative or higher-order skill (also see, e.g., [15]). To challenge the adaptive selection algorithms, skills were assigned varying difficulties that affected how well an average student could perform on each one. All test items were generated with the difficulty of their target skill.

Simulated Students. Simulated students demonstrated realistic ability distributions. Following the conventions of IRT analysis, we assigned each student a single underlying, hidden proficiency in each skill. Simulated students were generated with normally distributed proficiencies, and proficiencies in skills not at the root of a skill tree were normally distributed around the values of the node's parents. The student model contained additional noise caused by some skills that students were completely unable to perform (except by guessing). These cases, reflecting completely unlearned material, were generated with increasing probability for any skill where a student had proficiency one standard deviation or more below the mean.

Underlying student proficiencies affected the probability of answering any given question correctly. A logistic function mapped underlying proficiency to probability of answering a question correctly. According to this model, even highly proficient students still had a small chance of slipping or answering a question incorrectly. Additional sources of variance in the model included chance of guessing the correct answer, skill difficulty, and accuracy of prior proficiency estimates. Chance of guessing was calculated as if test items had a four-alternative multiple-choice design. Skill difficulty was normally distributed, and served to shift the population mean proficiency higher or lower relative to a particular skill. Prior proficiency scores were randomly initialized with a wide variance for each skill to emulate information from other sources such as tests other than our own or instructor inputs. Adaptive assessments could use the estimates as a starting point, but had to quickly identify incorrect values and improve those results to get a good measure of students' real proficiency.

All the adaptive selection algorithms had access to the same item pool, and were compared on their ability to score the same randomly generated students. The answer a particular student gave in response to each test item was held the same for every experimental run, ensuring maximum comparability. Only the order of item selection could be controlled by the algorithms being studied.

Simulation Process. One hundred simulated students were generated for each experimental run. A run consisted of a cycle of selecting a test item for a student, evaluating the student's response, and updating proficiency estimates until all test items were presented. Response evaluation and proficiency updates were the same for all runs, and were simple in order to reflect a basic level of practice. Each item response could be either correct or incorrect. Skill proficiency was estimated simply by the percentage of correct responses to items targeting that skill.

A property of the simple item scoring method used is that varying the order of scoring inputs does not impact the final score. Therefore, the experiments provide a closer approximation of the real-world testing use case that entails a single test event, as opposed to monitoring skills over time. An interesting direction for future research might be to explore possible interactions between item selection and a more sophisticated method for item scoring and proficiency estimation.

The fundamental metric we used to evaluate adaptive selection was time to reduce mean absolute error (MAE) of all skill estimates as compared to the true underlying value drawn from the simulation. Error change over time by itself is abstract and not easily comparable to other experiments with different students or test questions. For this reason, we present a normalized and concrete metric: the fraction of the entire item pool required to remove half of the error that is possible to remove. For example, in one experiment, MAE before asking any questions was 0.241 and after asking every question in the pool MAE was reduced to 0.137. Therefore, to evaluate the algorithms for choosing among these questions we examined the number of test items needed to reduce population error below the average of these values, 0.189.

4 Results

The readiness assessment use case. We were most interested in algorithm performance on topology A because it represented a typical skill hierarchy, representative of many training domains including our specific domain. The results of testing all candidate algorithms on topology A are shown in **Fig. 3**. For comparison, two additional approaches to leveraging skill tree topology rather than facts about individual learners are included in the graph. Two candidate algorithms outperformed the rest: prioritizing the skills whose proficiency estimates were least similar to the nodes around them, and prioritizing the skills that were least similar to an overall ability estimate.

We next tested a modified metric in order to determine whether the differences in candidate performance reached statistical significance. Rather than considering overall performance on the entire class of students, we measured performance on each individual student. This produced a parallel experiment with a population of 100 and a large variance in outcomes because of the wide variation between individual students' proficiencies. A series of unpaired, two-tailed Welch's t-tests showed that only the two identified algorithms performed significantly better than random ($p < 0.01$ for each). Comparing the mean differences in these pairwise comparisons against the perfect-knowledge baseline showed that modeling local outliers eliminated 53% of wasted item presentations, while modeling a single proficiency value eliminated 49%. Therefore, we next worked to characterize these two algorithms in more detail.

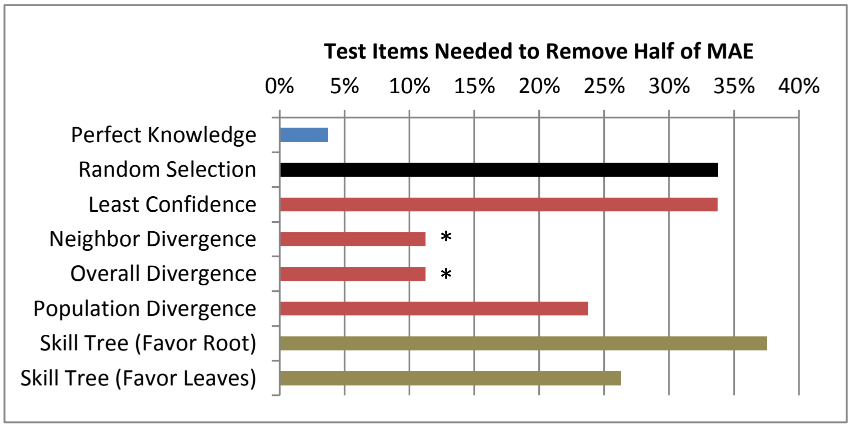


Fig. 3. Comparison of all candidate algorithms on topology A, a typical arrangement of skills into a multi-level hierarchy. Two candidate algorithms (*) yielded statistically significant improvements over random selection and were explored further in additional experiments.

Sensitivity to Skill Tree Topology. We next examined the performance of the neighbor divergence and overall divergence algorithms in tests about different general structures of instructional material that real-world instructors might employ. The results of this evaluation are shown in **Fig. 4**.

When instructors arrange skills in a flat, undifferentiated topology like topology B, the two adaptive algorithms tested perform equally well as would be expected when all nodes are neighbors to each other. We wanted to learn whether the two adaptive algorithms would perform much better (or worse) than random. We found performance was comparable to the outcome for topology A.

Topology C evaluated algorithm outcomes when the material being tested represents multiple skill clusters that are not closely related to each other. Not surprisingly, this topology challenged the single proficiency value algorithm. The single value was not sufficient for representing a student when the student could be very good at one group of skills and simultaneously poor at another. However, in this case the neighbor divergence approach still worked well. It was able to differentiate between a student who had scored poorly on a whole cluster of skills and one who had scored poorly on a single skill and might deserve a second chance on that result.

Finally, we tested the algorithms on a highly interconnected topology D. This relationship graph reflected the case in which each skill being tested was related to several others. We wanted to explore whether the increased interrelationship would cause conflicts in the adaptive algorithms, for example in identifying outlier nodes. Also, simulated student proficiencies under this topology included fewer outliers and more subtle distinction between students due to multiple parent nodes pulling toward the mean during proficiency generation. However, we did not observe a performance degradation, and the adaptive algorithms still performed better than random.

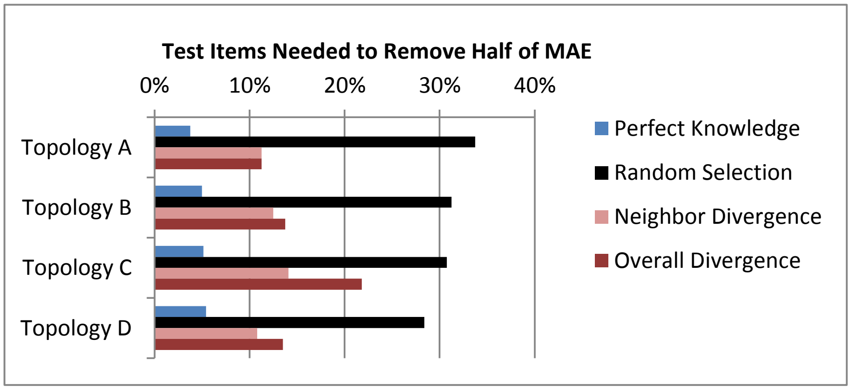


Fig. 4. Comparison of two adaptive algorithms on different arrangements of test skills. Both perform well, and the more complex algorithm reduces overall error more quickly when several unrelated skills must be assessed at once.

5 Discussion and Conclusions

The results of our experiments suggest that two adaptive algorithms, neighbor divergence and overall divergence, are capable of controlling adaptive assessment in training settings that do not offer large amounts of empirical data, calibration time, or formal expertise to fully characterize skill relationships and individual test items. At the same time, the algorithms are simple to explain and are likely to garner acceptance and adoption from practitioners in need of quick and efficient assessment.

The empirical results we present are stable over a range of simulation parameters. In addition to the different skill hierarchies studied, we evaluated the assessment algorithms under conditions of increased and decreased degree of imputation in scoring and noise in prior proficiency estimates. We found that adding imputation, that is, the ability to draw conclusions about more skills from a single test item presentation, benefitted all algorithms proportionally. Removing noise in the prior estimates, on the other hand, decreased the performance of the random baseline but increased the performance of the adaptive algorithms. When prior estimates were accurate, random selection could not find the few estimates that needed updating, but the adaptive algorithms could. We used these explorations to ensure the settings for both parameters were moderate in our evaluation.

The experiments we present here will support adaptive assessment in a real-world training environment. We will next implement the best-performing neighbor divergence algorithm in the readiness assessment system described above, where it will drive adaptive assessment for real students. In that setting, it will be possible to evaluate with real instructors and students how well the algorithms perform in efficiently identifying skill gaps. In addition, we intend to use the same algorithms in other, related training settings. For example, the same algorithms could add information to a learner model and indirectly or directly drive selection of training scenario content. Therefore, many adaptive training systems that require extended time and effort to execute a scenario or otherwise have sparse data available could benefit.

Acknowledgement

This work is supported in part by the Office of Naval Research via contracts N00014-10-C-0526 and N00014-11-M-0504. The views and conclusions contained in this document are those of the authors and should not be interpreted as representing the official policies, either expressed or implied, of the Department of Defense or Office of Naval Research. The U.S. Government is authorized to reproduce and distribute reprints for Government purposes notwithstanding any copyright notation hereon.

References

1. Rochlin, G.I.: Expert Operators and Critical Tasks. In: Trapped in the Net: The Unanticipated Consequences of Computerization. 1997. Princeton University Press. pp. 108-129.
2. Core, M.G., Lane, H.C., Van Lent, M., Gomboc, D., Solomon, S., and Rosenberg, M.: Building Explainable Artificial Intelligence Systems. In: Proceedings of the National Conference on Artificial Intelligence. 2006. AAAI Press.

3. van der Linden, W.J. and Pashley, P.J.: Item Selection and Ability Estimation in Adaptive Testing. In: Elements of Adaptive Testing, W.J. van der Linden and C.A.W. Glas, Editors. 2010. pp. 3-30.
4. Baker, F.B.: The Basics of Item Response Theory. 2001: ERIC Clearinghouse on Assessment and Evaluation.
5. Pardos, Z.A., Heffernan, N.T., Anderson, B., and Heffernan, C.L.: Using Fine-Grained Skill Models to Fit Student Performance with Bayesian Networks. In: Handbook of Educational Data Mining, R. Christobal, et al., Editors. 2010, CRC Press. pp. 417-426.
6. Educational Testing Service: GRE. 2013 [cited 2013 January 24]; Available from: www.ets.org/gre.
7. Cook, L.L. and Eignor, D.R.: IRT Equating Methods. In: Educational Measurement: Issues and Practice, 2005. **10**(3): pp. 37-45.
8. Davey, T. and Lee, Y.H.: Potential Impact of Context Effects on the Scoring and Equating of the Multistage GRE® Revised General Test. 2011, ETS GRE Board: Princeton, NJ.
9. Folsom-Kovarik, J.T., Sukthankar, G., Schatz, S., and Nicholson, D.: Scalable POMDPs for Diagnosis and Planning in Intelligent Tutoring Systems. In: Proactive Assistive Agents: Papers from the AAAI Fall Symposium. 2010, Association for the Advancement of Artificial Intelligence: Arlington, VA.
10. Kalyuga, S. and Sweller, J.: Rapid Dynamic Assessment of Expertise to Improve the Efficiency of Adaptive e-Learning. In: Educational Technology, Research and Development, 2005. **53**: pp. 83-93.
11. Wray, R. and Munro, A.: Simulation2Instruction: Using Simulation in All Phases of Instruction. In: the Interservice/Industry Training, Simulation & Education Conference (I/ITSEC). 2012. NTSA.
12. Kieras, D.E. and Meyer, D.E.: The Role of Cognitive Task Analysis in the Application of Predictive Models of Human Performance. In: Cognitive Task Analysis, J.M. Schraagen, S.F. Chipman, and V.L. Shalin, Editors. 2000, Erlbaum: Mahwah, NJ. pp. 237-260.
13. Birnbaum, A.: Some Latent Trait Models and their Use in Inferring an Examinee's Ability. In: Statistical Theories of Mental Test Scores, F.M. Lord and M.R. Novick, Editors. 1968, Addison-Wesley: Reading, MA.
14. Haladyna, T.M., Downing, S.M., and Rodriguez, M.C.: A Review of Multiple-Choice Item-Writing Guidelines for Classroom Assessment. In: Applied Measurement in Education, 2002. **15**(3): pp. 309-333.
15. Stiggins, R.J., Griswold, M.M., and Wikelund, K.R.: Measuring Thinking Skills through Classroom Assessment. In: Journal of Educational Measurement, 2005. **26**(3): pp. 233-246.

The Tactical Interaction Simulator: Finding the Motivational Sweet Spot in Game-Based Language Learning

W. Lewis Johnson

Alelo, Inc., 12910 Culver Bl., Suite J, Los Angeles, CA 90066 USA

Abstract. This is a late-breaking report on development and deployment of the TI Simulator, a game-based tool for learning communication skills in a foreign language. Learners practice their communication skills in spoken dialogs with animated characters. The TI Simulator is designed so that learners will keep practicing so that they develop good communication skills, develop confidence in their ability to communicate, and maintain their skills over time. To achieve this it employs game-based techniques to optimize learner motivation. It is designed to find the motivational "sweet spot" in game-based learning, where intrinsic motivation combines with extrinsic motivation to promote learning to mastery and persistent effort. This article reports on findings from initial evaluations, and efforts to promote adoption by teachers as well as learners.

Keywords: Simulation-based learning, motivation and affect, educational games, pedagogical agents, natural language processing, language learning, deployment of AIED systems

1 Introduction

The TI Simulator (Tactical Interaction Simulator) is Alelo's best example to date of serious gaming applied to language and culture training (Emonts et al., 2012). Developed in collaboration with the Australian Defence Force School of Languages (DFSL), the approach has broad relevance for learning communication skills.

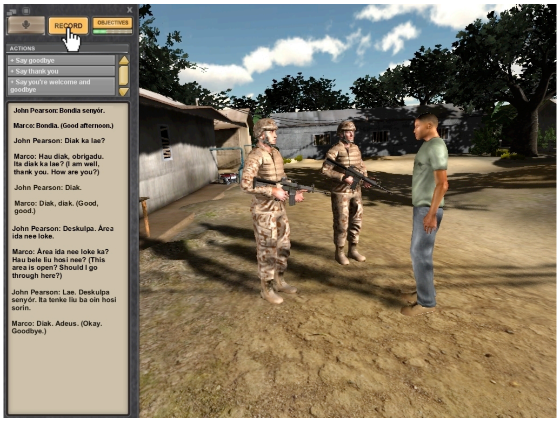
As in Alelo's earlier simulation-based language training courses (Johnson, 2010), the TI Simulator gives learners opportunities to practice their communication skills in immersive simulations, in which learners engage in spoken dialogs with animated characters. However unlike in other Alelo courses, the focus of the TI Simulator is entirely on the simulations. Our objective has been to make the simulations as effective as possible as a learning and sustainment tool. Learners enrolled in a classroom-based language course can use the TI Simulator to develop and practice their conversational skills, and develop confidence in their ability to employ those skills in a range of situations. Then after learners complete their initial course they can use the TI Simulator to continue to maintain their skills.

For the TI Simulator to succeed, it is critical that it optimize learner motivation. Many adult language learners experience performance anxiety, lack of self-confidence, and other negative affect when speaking in a foreign language, and this contributes to student attrition and lack of persistence (Bailey et al., 2003). The simulation-based approach gives learners opportunities to practice in an unthreatening environment, to build their self-confidence. But in addition to overcoming negative motivational factors, the TI Simulator should serve as a positive motivational influence. Learners should be motivated to use the TI Simulator, and this motivation should result in better learning.

To promote learner motivation, we employ techniques from game design, in what has come to be known as gamification (Kumar & Herger, 2013). Game mechanics such as scoring and achievements can provide extrinsic motivation to engage in a learning activity. However the TI Simulator aims to beyond that—it seeks to find the motivational "sweet spot" where intrinsic motivation combines with extrinsic motivation to optimize learning. Trainees learn as they play the game, they keep playing to achieve higher levels of attainment, they keep playing to achieve higher levels of mastery, and they learn outside the game in order to perform better within the game. Finding that sweet spot is not easy—it requires a proper integration of instructional design and game design, and it requires learner testing and iteration.

To succeed the TI Simulator must be adopted by teachers as well as learners. To this end we have just completed an initial round of evaluations with students and teachers and students at the DFSL, and are using the findings from that evaluation to guide further development.

2 How The TI Simulator Works



John Pearson: Bondia senyór.

Marco: Bondia. (Good afternoon.)

John Pearson: Diak ka lae?

Marco: Hau diak, obrigadu. Ita diak ka lae? (I am well, thank you. How are you?)

John Pearson: Diak.

Marco: Diak, diak. (Good, good.)

John Pearson: Deskulpa. Área ida nee loke.

Marco: Área ida nee loke ka? Hau bele liu hosi nee? (This area is open? Should I go through here?)

John Pearson: Lae. Deskulpa senyór. Ita tenke liu ba oin hosi sorin.

Marco: Diak. Adeus. (Okay. Goodbye.)

Figure 1. A beginner-level tactical interaction

The TI Simulator includes a collection of simulations of common interactions learners might have with local people. The learner plays the role of one of the soldiers, while the other characters are controlled by intelligent agents. The Simulator utilizes a combination of automated speech recognition, natural language processing, and intelligent agent technology to implement the interaction (Johnson et al., 2012). When the learner speaks into the microphone the system recognizes the learner's utterance, using a recognition grammar derived from the words and phrases in the curriculum. The system then interprets the intended meaning of the utterance, as well as the manner in which the meaning is conveyed. The non-player characters will react negatively if the learner speaks in an impolite or inappropriate way. The character behaviors are implemented as animations in the Unity3D game engine.

Each simulation can be played at different levels of difficulty. Color codes are used to indicate the difficulty level. Red (i.e., hostile) encounters have the lowest level of linguistic complexity – the trainees mainly give orders to bring the situation under control. Amber encounters have the highest level of complexity – the attitude of the locals is uncertain, and might become friendly or hostile depending upon what the learner says and does. Providing multiple levels of difficulty helps to ensure that learners are playing at a level that is neither too easy nor too difficult for them, so they are motivated to master the current level and then progress to higher levels.

As the learners practice the system performs a detailed assessment of learner performance along multiple dimensions. Learners receive points for accomplishing each communicative objective in the scenario. They fail to gain points if fail to communicate effectively, e.g., by failing to properly answer a local's question, by resorting to an interpreter, or by provoking a hostile response. Learners are scored for using linguistic forms that are appropriate for the situation, at the appropriate level of politeness or directness. They gain points when they employ a wide variety of vocabulary. Learners get immediate feedback through the responses of the non-player characters, as well feedback at the end of the scenario. The detailed assessment helps learners understand clearly where they need to improve, so that they are motivated to improve.

Learners can also select how much scaffolding they receive. At the beginner level, as shown in Figure 1 (left), learners are prompted with a choice of actions. They can receive hints as to what to say. They see a transcript of the conversation, both in the target language and in English. At higher levels these hints and prompts are progressively removed, until the scaffolding pane is eliminated entirely.

We use achievements to encourage learners to continue develop the ability to master the simulations without scaffolding. Once they fully master a simulation at the beginner level they receive one gold star. To gain additional stars they must progress to the higher levels and demonstrate accurate performance without the scaffolding.

3 Initial Evaluations

The DFSL has just completed an initial evaluation of an alpha version of the TI Simulator for Tetum (spoken in East Timor), as well as a companion self-study course called the Operational Tetum Skillbuilder. Instructors and students participated in the evaluation. We chose to conduct the evaluation before the courses were complete so that we could use the findings from the evaluation to refine and improve the product.

The first question to answer in the evaluation was whether the game would function as expected for the learners and teachers, using the outdated computers and microphones available to the learners at the school. Given the complex combination of speech recognition, artificial intelligence, and 3D game technology integrated into the product, this was not a given. As it turned out the system performed quite well, at least for male users. Female users experienced some occasional difficulties with the speech understanding software, which we are working to correct.

Initial indications are that the product is on its way to hitting the motivational sweet spot. The students and teachers agree that it is a fun and motivating way to learn, and a useful way to maintain language skills. The product will require further polishing before we know for sure that learners maintain a good flow of gameplay.

There can be numerous practical barriers to adoption of a learning system such as this, and the experience with the TI Simulator is no exception. One issue is that the spelling standards for Tetum used by the DFSL have changed, and so the content in the TI Simulator has revised to use the new spelling, before it can be integrated into the DFSL curriculum.

4 What's Next

We are now performing further enhancements to the TI Simulator, based on the evaluation feedback that we have received so far. We have received requests from the instructors for further improvements and new features, which will inform future development. The final release is planned for August 2013, after which we plan another training session with the DFSL instructors. We look forward to receiving additional feedback from the teachers regarding the product, as well as learning how they envision putting the product to use in their course.

References

1. Bailey, P., Onwuegbuzie, A.J., & Daley, C.E. (2003). Foreign language anxiety and student attrition. *Academic Exchange Quarterly*, Summer 2003.
2. Emonts, M., Row, R., Johnson. W.L., Thomson, E., Joyce, H. de S., Gorman, G., & Carpenter, R. (2012). Integration of social simulations into a task-based blended training curriculum. In Proceedings of the 2012 Land Warfare Conference. Canberra, AUS: DSTO.
3. Johnson, W.L. (2010). Serious use of a serious game for language learning. *Int. J. of Artificial Intelligence in Ed*. 20(2).
4. Johnson, W.L., Friedland, L., Watson, A.M., & Surface, E.A. (2012). In P.J. Durlach & A.M. Lesgold (Eds.), The art and science of developing intercultural competence. 261-285. New York: Cambridge University Press.
5. Kumar, J. & Herger, M. (2013). Gamification at Work: Designing Engaging Business Software. Aarhus, DK: Interaction Design Foundation.

SAS® Read Aloud: A Mobile App for Early Reading

Jennifer L. Sabourin, Lucy R. Shores, Scott W. McQuiggan

SAS Institute, 100 SAS Campus Dr. Cary, North Carolina 27513

{Jennifer.Sabourin, Lucy.Shores,
Scott.McQuiggan}@sas.com

Abstract. Shared reading is an important instructional technique for developing literacy in young readers. It helps to develop print and phonological awareness and fosters motivation and enjoyment of reading. Mobile reading technologies have capitalized on some of the benefits of shared reading, but there has been limited systematic investigation into how they can be most effectively used to support early literacy. This work presents SAS® Read Aloud, a mobile iPad app for early reading with a design that is grounded by empirical research. We also identify opportunities for incorporating intelligent technologies to further improve and understand early literacy learning.

Keywords: Mobile learning, reading development, shared reading

1 Background

Reading skills are multifaceted and are acquired overtime beginning at a very young age. Before children being to demonstrate the skill of reading autonomously, several foundational skills develop during a period known as emergent reading [1]. Given the cumulative nature of the development of reading skills, increasing instructional quality at the emergent reader level has been identified as a powerful technique for preventing reading difficulties later in development [1]. However, according to the 2011 National Assessment of Educational Progress [2], on average, only 34% of US 4th graders demonstrated proficient reading status leaving the majority of students reading at or below the basic level [2]. Furthermore, longitudinal investigations have identified 3rd grade reading performance as a critical variable for predicting children's lifelong academic success [3]. Therefore, there is great motivation for enhancing children's experiences during the emergent and early reading stages.

Shared reading has been touted as one of the most influential instructional techniques for both phonological awareness and written language awareness development [1, 4, 5]. In fact, according to the *Commission on Reading* [6], "The single most important activity for building knowledge required for eventual success in reading is reading aloud to children" (p. 23). Also known as joint book reading and storytime, shared reading occurs when a parent or more advanced peer reads aloud to a developing reader [4]. While reading aloud might seem casual and basic on the surface, several longitudinal studies have shown shared reading experiences to be a better predic-

adfa, p. 1, 2011.

tor of later reading performance than common educational predictors such as socioeconomic status or parent education [4, 7].

Among the benefits of shared reading, these experiences allow emergent readers to practice and develop print and phonological awareness—skills necessary for further reading development. For example, by reading along with others, children are given opportunities to see written language in various forms (print awareness) as well as draw connections between text features such as written words and letters and spoken language (phonological awareness). Furthermore, shared reading creates a space where children can observe fluent readers modeling more advanced skills such as comprehension strategies and fluency [8]. Additionally, through reading aloud with friends and family, children can develop positive associations with reading, which have been shown to encourage and influence later motivation for reading [9].

Therefore, parents are encouraged to expose their emergent readers to literacy experiences, such as shared reading, as often as possible, especially during the preschool years [4]. Moreover, while all shared reading experiences are valuable, the utility of the session as an instructional tool is dependent on quality [5, 8]. Research investigating eye gaze has shown that, without guidance, emergent readers generally focus on illustrations as opposed to text during shared reading sessions [10] and that children appear to benefit more from active sessions in which the reader engages the child using various attention-focusing methods [5]. As ubiquitous reading applications are becoming available on popular mobile devices, it is important to investigate how application developers can leverage this technology to provide shared reading experiences designed with best practices for reading development.

2 SAS® Read Aloud

SAS® Read Aloud is a mobile iPad application for early literacy centered around a digital library of freely available books. At the time of writing, this library consists of over 24 books designed to support different levels of literacy learners including early emergent, emergent, and early fluent readers. Each book may be read in one of three modes, designed to guide learners through different stages of reading:

- **Read to Me:** In this mode, readers see words highlighted as the book is read aloud by a narrator. Readers experience the intonation, rhythm, and stress provided by each speaker. This mode is designed to engage readers in stories and offer an introduction to text that may be beyond the learner's current abilities.
- **Help Me Read:** In this mode, readers are guided through the book and control the speaker's pace as each word is read aloud independently. Readers focus on developing both print and phonological awareness. This mode is intended to guide students towards independent reading by drawing attention to individual words and how these combine to make sentences and stories.
- **Read by Myself:** In this mode, readers are encouraged to read through the book independently with the ability to select the words they would like to be read aloud. This mode is intended to allow readers to build confidence in their reading abilities while supporting them when there are words they are unfamiliar with.

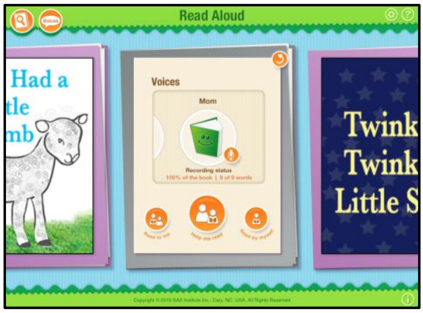
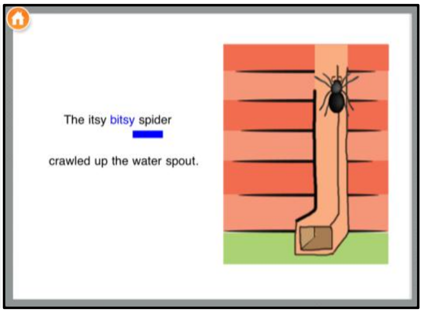


Fig. 1. SAS Read Aloud


The itsy bitsy spider

crawled up the water spout.

Fig 2. *Help Me Read* mode

Each book includes a default narration designed to engage the reader and support development of correct pronunciation and intonation during reading. Additionally, SAS® Read Aloud encourages users such as parents, teachers or young readers to record themselves reading each story. During recording, the narrator follows the text that is being read aloud with their finger to indicate which word should be highlighted along with the recording. In this way, users can still enjoy all the features of the *Read to Me* mode including intonation and rhythm with the literacy support features of word-by-word highlighting. Narrators are also encouraged to record individual words with a focus on clarity and pronunciation to support the word-level learning encouraged by the *Help Me Read* and *Read by Myself* modes. Through these recordings, students will be able to listen to books recorded by people they know and love such as parents, grandparents, and teachers. It is hoped that this will develop a motivational and emotional connection to reading books with SAS® Read Aloud similar to that of one-on-one storytime. Together, these features seek to promote a love of reading and provide the tools for young readers to guide their own literacy learning.

3 Proposed Directions

Since its release, conversations with users have identified several important areas for future directions. Many users are interacting with SAS® Read Aloud in unexpected ways to further support early literacy. Most prominently, teachers are encouraging students to record themselves reading a story and listen to it to identify their own reading disfluencies. Some students are asked to listen to their own recording and compare it to an expert. Others are asked to complete multiple recordings to see how they have improved with practice. These patterns of interaction have prompted the development of additional features titled *Practice and Progress*.

In this mode students are encouraged to record multiple practice readings of a book. Each recording is stored along with the date and time so that students may play back any earlier readings. During playback students, teachers or parents are able to identify areas of difficulty and indicate errors such as stumbles or mispronounced words. This information can be used in reports or feedback designed to demonstrate a student's progress and improvement over time and encourage targeted reading practice. Furthermore, it is hoped that this mode can be used in place of traditional "practice your reading" assignments often given by teachers. Now, instead of relying on student reports of the time they spent reading, teachers can verify students' practice and easily assess any issues the student may be having.

The inclusion of this mode opens several opportunities for intelligent modeling and adaptation. First, with user consent, practice recordings and annotations will be uploaded to a secure server for analysis. This is expected to result in a large corpus of early reader speech that can be used for building targeted speech recognition models. This, along with the annotation of errors, will aid in the development of automated identification of reader mistakes and fluency. In this way, mistakes can be highlighted automatically for users. Additionally, detailed fluency metrics can be tracked across time without requiring user annotation. The second major opportunity is to apply machine learning techniques to the corpus of annotated errors to identify common mistakes (e.g. compound vowels, or multisyllabic words). Learned models can then be incorporated into the application to guide users towards exercises or books that may help them practice in areas where they have difficulties. This will help the user receive the targeted support and practice they need to improve their reading skills. Finally, the corpus may be analyzed to identify patterns of how early readers learn to read. Data mining approaches can highlight common patterns of development and suggest new areas for investigation. Overall, intelligent modeling and adaptation provides significant opportunities for advancing the efficacy and impact of early reading applications with the goal of addressing the national need for more proficient readers.

References

1. Piasta, S. B., Justice, L. M., McGinty, A. S., Kaderavek, J. N.: Increasing young children's contact with print during shared reading: longitudinal effects on literacy achievement. Child Development 83, pp. 810–820 (2012).
2. National Center for Educational Statistics: The nation's report card: Reading 2009: National Assessment of Educational Progress at grades 4 and 8. Washington D.C., (2009).
3. Annie E. Casey Foundation: Early Warning! Why Reading by the End of Third Grade Matters. 1–62 (Baltimore, MD, 2010).
4. Bus, A. G., Van IJzendoom, M. J., Pellegrini, A. D.: Joint Book Reading Makes for Success in Learning to Read: A Meta-Analysis on Intergenerational Tramission of Literacy. Review of Educational Research 65, 1–21 (1995).
5. Lane, H. B., Wright, T. L.: Maximizing the Effectiveness of Reading Aloud. The Reading Teacher 60, 668–675 (2007).
6. Anderson, R. C., Heibert, E. F., Scott, J. A., Wilkinson, I. A.: Becoming a nation of readers: The report of the Commission on Reading. Washington D.C., (1985).
7. Senechal, M., LeFevre, J.: Parental Involvement in the Developent of Children's Reading Skill: A Five-Year Longitudinal Study. Child Development 73, 445–460 (2002).
8. Fisher, D., Flood, J., Lapp, D., Frey, N.: Interactive Read-Alouds: Is There a Common Set of Implementation Practices? The Reading Teacher 58, 8–17 (2004).
9. Baker, L., Scher, D. Beginning Readers' Motivation for Reading in Relation to Parental Beliefs and Home Reading Experiences.: Reading Psychology 23, 239–269 (2002).
10. Evans, M. A., Saint-Aubin, J.: What Children Are Looking at During Shared Storybook Reading. Psychological Science 16, 913–920 (2005).