Modeling Proficiency in a Tailored, Situated Training Environment

Jeremiah T. Folsom-Kovarik, Charles Newton, Joshua Haley, Robert E. Wray
Soar Technology, Inc.
3600 Green Court, Suite 600
Ann Arbor, MI 48105
jeremiah, charles.newton, joshua.haley, wray@soartech.com
Keywords:
Skill Training, Learner Proficiency, User Modeling, Individualization, Dynamic Tailoring, Scenario-Based Training

ABSTRACT: *Situated training via simulation has many beneficial properties including the potential to adapt or to tailor training experience to the evolving needs and competencies of individual learners. Such tailoring often requires estimates of learner proficiency. These estimates inform the tailoring that is undertaken by the system. In this paper, we propose that proficiency models must be analytic, informal, and transparent to support situated training systems. We then assess the extent to which some previous approaches to proficiency modeling meet these requirements and describe several enhancements to an established proficiency-modeling paradigm. We hypothesize these enhancements will enable practical use of proficiency modeling in situated training systems. We illustrate with examples from application in a perceptual training system currently under development.*

1. Tailored, Situated Training

Situated training refers to demonstration, practice, and assessment of skills in a representative environment, often a simulated one, with scenarios or background stories to link the skills. Situated training gives learners context for the skills being trained, supporting their ability to recognize cues and construct mental models for increased acquisition speed, retention, and transfer (Oser, Cannon-Bowers, Salas, & Dwyer, 1999).

Tailored training adds individualization of scenario experience, so that scenarios become more closely aligned with the needs of individual learners in terms of progression, support, and instructional strategy selection (Wray & Woods, 2013). In a situated training environment, tailoring can change scenario events and story details to better meet training needs. These *simulation-intrinsic* interventions, as opposed to extrinsic interventions such as text boxes or explicit tutor dialogs, introduce an ability to assess and remediate target skills within the context of a situation without distracting the learner from the simulated experience or decreasing learner engagement.

To deliver such an individualized experience, a model of the student's proficiency is typically needed The information stored in our proficiency model includes a list of domain skills, estimates of individual proficiency for each skill, and descriptions of how the estimates vary in reaction to specific observations about the learner and changes elsewhere in the model (via skill relationships). The proficiency estimates are then queried to decide (along with other factors not covered here) what tailoring is appropriate.

In addition to modeling proficiency itself, situated training and practical constraints impose additional requirements on the design of the proficiency model component. For example, the model we describe is not a strongly principled cognitive model in comparison to, e.g., Anderson and colleagues many years of work carefully modeling skill acquisition (e.g., Anderson, 1982). Instead, we are targeting an operational mechanism that enables different modes of adaptation. The model may not predict skill level as accurately or precisely as other methods. However, we see one of the advantages of the "fuzzy" categorization we use within the model is that the estimate of the learner's skill is not tied to a precise and fixed assessment.

This paper describes these requirements and then outlines a *fuzzy vector* proficiency model (Katz & Lesgold, 1991; Lesgold, Lajoie, Bunzo, & Eggan, 1988), enhanced with new capabilities targeting improved handling of situated data. Although the proficiency modeling approach is designed for use across multiple domains, we focus here on how it facilitates the delivery of tailored, situated training for a specific advanced military skills training simulator.

1.1 Requirements for proficiency modeling

The *Virtual Observation Platform* (VOP) is an immersive simulation that trains perceptual and cultural skills in the setting of a military observation post and its environs (Schatz, Wray, Folsom-Kovarik, & Nicholson, 2012). Skill practice is situated in the context of an ongoing mission and framing story, and the details of the scenario can be tailored in real time through the VOP's *Dynamic Tailoring System* (DTS). In building this complex system (and others) for real-world military training, we are identifying requirements for a trainee proficiency model.

The DTS carries out tailoring that is designed to manipulate the pedagogical experience through changes intrinsic to the scenario. Tailoring is used to scaffold (Pea 2004) and fade scaffolding based on estimated learner ability, as well as to increase challenge for the purpose of motivating learning (Vygotsky, 1978). For example, in the VOP, if a novice trainee is believed to have low proficiency in the perceptual skill of noticing human kinesic (gestural) cues then the DTS might exaggerate the magnitude and visual salience of a simulated character's movement to support the trainee seeing that gesture. Conversely, as the trainee becomes more expert in recognizing gestures the DTS might introduce visual distractors to challenge performance.

Tailoring in the VOP can differentially affect a range of separate skills (Fautua et al., 2010) that combine to affect observable trainee performance simultaneously. For example, a scenario in the VOP might require correct visual scan technique for the trainee to see an event taking place, sociocultural sensemaking skills (Klein, Moon, & Hoffman, 2006) to correctly interpret the importance of the observed event, and finally correct selection and execution of doctrinal responses to the event such as reporting an important observation to teammates or commanders. Therefore, proficiency estimates for multiple interrelated skills are needed to effectively tailor situated training of different skills.

Because training is situated, the interpretation of performance is more complex. Interpretation must take place in the context of previous proficiency estimates, variable task complexity and difficulty, current level of support or challenge that the tailoring system has implemented, and likely slips or guesses that we expect might make a particular skill performance unreliable in indicating actual underlying ability. Further, instructor judgment is a valuable source of proficiency input. While technically challenging to collect in a user-friendly and useful manner, instructor input must be incorporated in the DTS both to improve interpretation and to gain user acceptance. An important contribution of this work is the ability for a proficiency model to account for all these sources of context.

Mitigating the complexity of the requirements outlined thus far, there is a useful commonality shared by the kinds of intrinsic pedagogical tailoring that the DTS executes. While intrinsic tailoring does create a meaningful scenario difference between what experiences are appropriate at each proficiency level, it also provides some forgiveness if the estimated level differs from a trainee's actual proficiency. Unlike simulation-extrinsic pedagogical interventions (e.g., initiating a dialog to correct an error), intrinsic interventions are unlikely to interfere with learning or even distract the trainee if they are occasionally displayed incorrectly. If the DTS estimates a trainee has not observed a target behavior when he actually has, then increasing the behavior's salience has less negative impact on training efficacy than displaying an error message to the trainee. The low commitment of intrinsic tailoring to a particular skill estimate is valuable to the design we present.

Based on the requirements associated with situated training, we next briefly explore differences between several candidate models of trainee proficiency.

1.2 Comparing proficiency models for tailored, situated training

Three desiderata describe a trainee proficiency model that can meet the above requirements and support tailored, situated training. The model must be analytic, informal, and transparent. We briefly consider some proficiency models in terms of these three dimensions.

First, a proficiency model should be *analytic*, that is, it should support analysis of its contents that aims to understand latent proficiency as an underlying reason for observed learner performance. Here we include many kinds of analysis, from Bayesian inference to simple comparison of decimal estimates. These are in contrast to non-analytic models where interpretation does not add knowledge because of too-simple values or structure. An analytic proficiency model is needed to make a training system wholly adaptive to individual learners, rather than simply reacting to learner errors piecemeal by taking each one at face value (Schatz, Oakes, Folsom-Kovarik, & Dolletski-Lazar, 2012).

Some ways an analytic proficiency model plays an important role in interpreting learner performance and other inputs are by smoothing input noise, such as discounting slips and guesses so that they do not impact the system as much as other performance. Another important kind of analysis is the application of hierarchical relationships in understanding performance observations. Making one skill part-of or prerequisite to another skill lets a training system reason about learner performance and about intervention planning. Models with skill relationships allow the system to infer proficiency with fewer or no direct measurements for some skills.

An example of a highly analytic proficiency model might be Bayesian approaches to implementing "knowledge tracing" in ACT-R systems (Corbett & Anderson, 1995), if we consider the mastery of a knowledge component as a skill whose proficiency is being estimated. Examples of training systems where the proficiency model might be said to be less than analytic include the pioneering ACT intelligent tutoring systems (Anderson, 1993) before they included such components. Today, the analytic property is most lacking in training systems that do without a distinct proficiency model, where the model of a learner's proficiency is instead implicit in the components that constrain performance and gate progression (e.g., Wray, Woods, & Priest, 2012). The original fuzzy vector model was only moderately analytic. It provided proficiency estimates more granular than "mastered" or not and it captured learner differences and change over time, but did not make use of relationships between skills (Katz & Lesgold, 1991).

Second, a proficiency model should be *informal*. This property reflects the extent to which the model tolerates imprecise inputs and authoring. An informal proficiency model is one in which estimates are represented in a manner that allows wide ranges rather than precise values. Furthermore, end users of an informal model may enter values that control it, such as proficiency estimates or relationships between skills, without technical knowledge of exact probabilities or second- and third-order effects on model internals.

The property of informality is useful in situated training because intrinsic adaptation, as discussed, reduces the demand on the proficiency model to be precise or even correct in all of its estimates. Further, when interacting with a human instructor, the proficiency model should support categorizing a learner or forming a subjective opinion about a learner rather than delivering a precise grade. A proficiency model with the property of informality is explicitly not designed to support end users who need high-stakes pass-fail decisions. Furthermore, such a proficiency model will not be appropriate to interpret in terms of any output as precisely defined as a letter grade or a percentile. Rather, its outputs should be presented to humans in imprecise terms such as (perhaps) quartiles, even if internally it contains more precise estimates.

An informal proficiency model supports *instructor-mediated design* (Folsom-Kovarik, Wray, & Hamel, 2013). This is a design pattern the authors use to better meet practitioner needs and increase instructor acceptance. The goal is to remove technical barriers to instructors' control over the content and operation of a training system – thereby reducing costs, turnaround time for changes, and errors introduced during communication between end users and developers. Examples of the changes instructors want to be able to make in training scenarios might include editing scenario events or changing how trainee actions are assessed when the state of the art or doctrine evolve.

Instructors and end users who want to add training content or control a system behavior are not typically interested in carefully balancing weights or analyzing probabilities. By selecting a proficiency model that is informal, we increase the likelihood that end users will be able and willing to make changes in the model content between training sessions to reflect ongoing evolution in training, as well as to manipulate model values and estimates during simulations to reflect their beliefs about individual trainees' abilities.

An example of an informal proficiency model is the original fuzzy vector model in the Sherlock series of ITSs (Katz & Lesgold, 1991; Lesgold et al., 1988). In Sherlock, each skill was represented by a vector of five real values representing "no knowledge, limited knowledge, unautomated knowledge, partially automated knowledge, and fully developed knowledge." Fuzzy logic interpreted the vector similarly to a probability distribution across the five possible values. This model is functionally informal despite the fact that it includes probabilities and percentages. One reason is that the categories used are qualitative, broadly defined, and "fuzzily" labeled so that they do not tend to translate into scores or letter grades in the minds of end users. A second reason is that the rules for updating proficiency estimates are not based on mathematical principles and thus do not display the greater sensitivity to error that attends greater precision in estimation. There is room in the fuzziness to allow some play or variation in non-technical users' application of the rules. This is in contrast to the least informal proficiency models, which take advantage of precise Bayesian inference but which therefore require very carefully defined relationships between skills and proficiency estimates, or may exhibit unpredictable errors (Henrion, 1987).

Third and finally, a proficiency model should be *transparent* to end users, explainable, and authorable. While we use informality to refer to the way end users can easily enter in and read out proficiency estimates, we use the transparency property to describe the way the model changes those values over time. The changes must be easy for end users to understand and control. This property ensures a system will remain useful after deployment, as instructional designers and instructors can understand what the system is doing and, when necessary, manipulate the proficiency model to reflect the learner population or changes in training material.

The need for model transparency means that instructors should have oversight or manipulation abilities over proficiency model content and structure. In terms of content, it should be possible for example to add a new skill to the model and to change the interpretation of an existing skill. Examples of control over model structure include introducing and removing relationships between skills, such as recording the fact that a new skill has an existing skill as a prerequisite. However, as opposed to structure, the parameters defining a skill relationship, for example the strength of the new dependency, might not be considered feasible for nontechnical authors to control. The reason is that numeric parameters like weights are expected to have exact and possibly subtle effects on model performance, contravening the requirement for informal rather than precise functionality. An example of parameter control that a nontechnical person might be expected to exert would be changing the number of consecutive correct responses required to infer mastery of a skill. An example that would not be feasible might be setting the probability that a learner will remember a skill after a particular system intervention.

An example of a transparent proficiency model is the simple overlay, which is still used today in systems where proficiency modeling is not the focus of study. Overlay models (Carbonell, 1970) function similarly to a checklist of skills that an expert displays. A system can compare a learner to an overlay without needing knowledge of how the skills are learned and so forth. However, transparency in an overlay is only achieved through great simplicity such as estimates that only reflect pass or fail, and lack of skill relationships.

In various examples of Bayesian proficiency models, the outputs and value updates may be explained to non-technical users with some level of effort (Almond, Shute, Underwood, & Zapata-Rivera, 2009). However, model authoring is not typically allowed. Fuzzy models represent a happy medium for situated tutors in terms of transparency and explainability. They implement a small number of update rules. Rules are defined with a unified structure that enables non-technical authoring, comparison, and understanding of rule outcomes.

Based on these requirements, we selected fuzzy vectors for for modeling proficiency in the DTS. Fuzzy vector outputs are transparent and easily explainable, while their simple specification with qualitative categories and without precise underlying semantics makes them informal enough for end users to control and update. Their analytic capacity to provide inferences about latent values from observations is limited but could be expanded. The next section describes how we enhanced the original fuzzy vector architecture in order to make it more analytic, informal, and transparent and to meet the needs of tailored, situated training.

2. Enhanced Fuzzy Vector Models

We describe the historical fuzzy vector model and three enhancements to it we have found useful. Linking rules to training context enables greater accuracy, generalization of model update rules to a matrix representation increases expressiveness, and new skill relationships increase the analytic power of the model by enabling greater inference from each input.

2.1 Related work

The basis of the fuzzy model we use is the one developed for the Sherlock II ITS, itself a situated training environment providing practice in electronics troubleshooting in the context of a simulated radio (Katz & Lesgold, 1991). For each skill measured, a vector contains five elements representing the probability that the learner is in one of five categories. The five categories are designed to capture progress through the acquisition of declarative knowledge and then procedural skill, for example as according to Anderson (1982). The lowest category indicates lack of any knowledge. The next two indicate partial or complete declarative knowledge, culminating in the middle category with a quality of knowledge equivalent to classroom exposition before any practice or to Anderson's Phase I of skill acquisition characterized by conscious, effortful performance. The fourth category corresponds with Anderson's Phase II of knowledge compilation or skill integration. The fifth and final category is Anderson's Phase III or automatized skill performance. In the VOP, we use these five model categories for all perceptual, sociocultural, and other skills (Newell & Rosenbloom, 1981). However, different training environments may assign alternative interpretations to the skill vector via their selection of interventions corresponding to each vector element.

The fuzzy logic that combines the five real-valued vector elements is simpler than a full probability distribution. At any given time the system will maintain a probability distribution over all the five categories in the fuzzy vector, but the proficiency model will report only one of the categories for each skill as its concrete, 100% estimate of current proficiency. The probability that a category is selected as the current estimate corresponds to its value in the underlying fuzzy vector. To reduce fluctuation in estimates, the estimate for a skill is resampled only when the underlying vector for that skill is updated. Fuzzy logic is more effective than conventional methods in system control applications where end users are not technical or natural language categories may be imprecisely defined and related (Lee, 1990). For training, the fuzzy logic approach allows a system to collapse internal probability distributions into varied, individualized categorical estimates of proficiency. These estimates then inform tailored interventions and are meaningful and actionable to instructors.

In Sherlock, all skill proficiencies were evaluated independently. Summative skill estimates represented a weighted average of certain directly estimated proficiencies, yielding the beginnings of a skill hierarchy with a mostly flat implementation. Previous tailored training research built on the Sherlock fuzzy model by adding a skill hierarchy of arbitrary depth (Wray et al., 2009).

Update rules in Sherlock were defined in a way that supported changing estimates differentially based on characteristics of the learner. After any performance observation, vector elements could be decreased by a certain percentage of their previous value, with the difference moving into the next higher or next lower category depending on whether performance was correct or incorrect. Different rules could also be created that specified whether to move a large or a small percentage from the higher or lower categories. The salient fact about this rule representation is that each new proficiency estimate was most strongly determined by the previous estimate, a characteristic of the learner rather than of the learning context.

2.2 Capturing training context in update rules

The first enhancement we introduce is the addition of a context dimension to the model update rules. Although rules still describe how to change the proficiency estimate for a skill based on information about the current estimate, the intelligent system that uses the proficiency model must now choose which rule to apply based on information that is not contained in the skill estimates.

In any system, there must be at least two update rules: a rule to increase skill estimates and a rule to decrease them. There are many other context considerations that might inspire additional update rules. For example, the system might differentiate between correct performance with low support (scaffolding) and high support. The update rule for low support would increase the probability of automaticity more than the rule for high support (Figure 1).

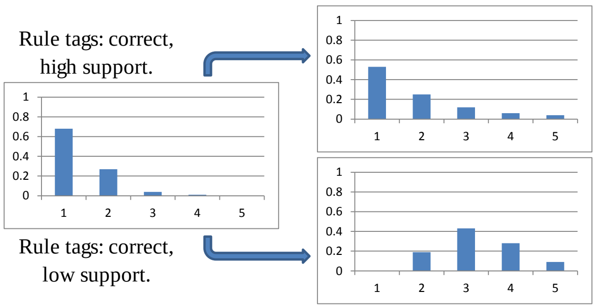

Figure 1: Differential effect of training context.

Context considerations in the current system include performance correctness, task difficulty, speed or facility of response, current amount of support or challenge, and confidence in credit assignment (which skill gap caused the learner to perform poorly).

Creating an update rule for every possible combination of context factors potentially requires an exponential number of rules, although some rules might be reused or deemed impossible to occur. Instead, the model supports rule selection with tags that correspond to different contextual dimensions. For example, one performance observation might be tagged as correct and, separately, as low support – allowing lookup of the corresponding rule. We currently define rules manually, but a sufficiently complex training system might want to synthesize the rule definitions analytically or empirically.

Currently, the conditions which relate rules to particular tags are manually specified by users, either instructors or developers. These mappings require more detailed scenario authoring than we expect many instructors will accept. In parallel with this effort, we are also developing scenario representations based on plan-based narrative representations (Magerko, Laird, Assanie, Kerfoot, & Stokes, 2004; Riedl, Stern, Dini, & Alderman, 2008) that describe both the alternative paths thru a scenario and the rationales for individual paths. For example, one choice point in the scenario may be to take a path that presents fewer and more predictable objects and events to a learner than another.

We hypothesize that metadata associated with these different paths can be used to automatically generate situation and context tags that contextual update rules can match against. For example, the path that presents fewer objects and more predictable outcomes can be automatically labeled as a "high support" path. When student actions are assessed and this path is active, "high support" proficiency update rules can be applied. This approach would then obviate the need for an instructor or developer to explicitly author a mapping between update rules and student actions.

2.3 Generalized rule representation

To accommodate the addition of context to rules, we generalize the fuzzy vector model update rules to a new, consistently organized format which is more expressive than the earlier fuzzy model. Encoding more of the update rule logic in the rule itself rather than in the rule application engine enables more automatic selection and application of context-specific rules in order to robustly handle context with least commitment to a specific algorithm.

In the enhanced model, we now define model update rules as a square matrix indicating the change from every possible vector element to every possible element. As long as the rows of the update matrix sum to one, an update can be carried out efficiently by multiplying the old vector with the update rule matrix. The significance of the update rule as a matrix, again, is that the representation widens the focus of the fuzzy vector model away from solely characteristics of the learner, changing what is easy and hard to express.

The matrix update rules can be authored to capture as much or as little as desired of principled, Bayesian explanation or other models that might be relevant to training, such as item-response theory. The matrices can encode a probabilistic state transition function as induced from empirical data, or simply the intuitions of nontechnical experts in the field. The matrices can also be initialized to approximate values and then refined through machine learning if so desired.

Whether authors create the update rule values with item response in mind or not, we note that the matrix representation suggests an easily implemented heuristic to carry out adaptive assessment selection. The rule matrix entries on the diagonal of the matrix describe the degree to which the proficiency estimate of a particular skill will remain the same after a particular observation. Given all the possible update rules that an assessment could trigger based on the different learner performances in response to it, we might therefore select the test item that minimizes the average of the diagonal entries weighted by the current proficiency estimate. This heuristic works based on the actual functioning of the rules as defined, even if the definitions are imprecise, and so offers the possibility of an intelligent system selecting among scenario content or tailoring strategies without first requiring the technical tasks of carefully characterizing and balancing item properties.

2.4 Relationships between skills

In the context of a different simulation environment (Folsom-Kovarik et al., 2013), we showed that a well-chosen assessment can provide an intelligent system with useful information about multiple proficiency estimates by leveraging simple, qualitative relationships between skills. In that system, skills are arranged in a graph representing part-of relationships only. For example, skill at single-digit addition is a subset of skill at addition in general. Relationships do not have a weight property or any numerical parameters. We believe that such a structure both reflects typical usage in our target instructional community and provides a useful balance between utility and ease of specification for non-technical users.

The granularity of skill definitions in the hierarchies we use is determined by what supports useful tailoring. Rather than modeling atomic knowledge components, we have had success defining fewer, coarser skills that can each be relevant to trainee performance at multiple points throughout training. Since skills typically interact in a naturalistic setting, the permutations of these skills give sufficient variety for practical tailoring without onerous burden for authoring scenario content or initializing reasonable estimate values.

In our current training system, whenever a rule updates a proficiency estimate, the changes to the vector propagate to other estimates that are part of the changed estimate either directly or indirectly. Propagation is currently simple but still useful for updating multiple estimates. For each skill that is a parent of a changed skill, update each vector element by a fraction of the change in the corresponding element from the changed child. The fraction is determined by a tuning constant divided by the number of children the parent has. Propagation is recursive to allow changes throughout the skill graph. Figure 3 presents a pseudocode summary of the enhanced fuzzy vector update method overall.

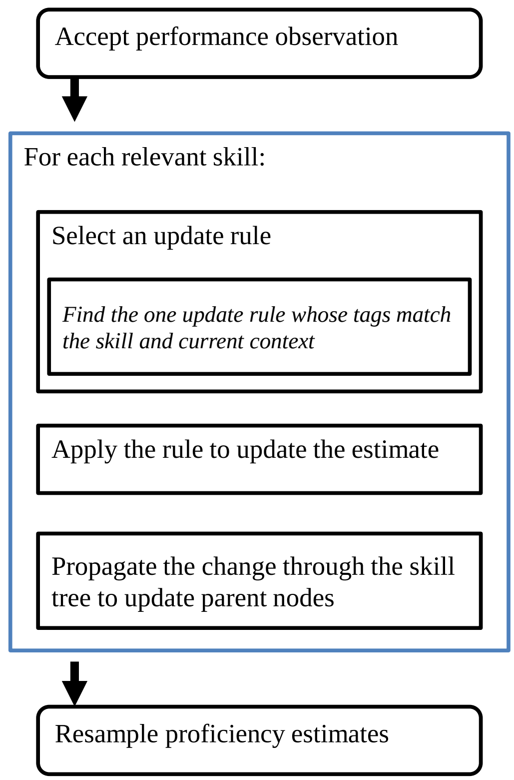

Figure 2: Enhanced fuzzy vector updates.

We plan to evaluate this simple propagation method for accuracy in experiments with simulated students. More advanced propagation, if needed, might for example propagate negative changes to parents and positive changes to children. These algorithms still would not require any parameters to define skill relationships.

3. Discussion and Future Work

The enhanced fuzzy vector model we describe has been implemented in the Dynamic Tailoring System and is in use for two training systems, the VOP and a combat flight simulator. It will next be evaluated in terms of usability, acceptance, and training effectiveness through a series of human-participant studies.

As we evaluate the model and its properties, some open questions remain to be explored. For example, we believe that any additional regularities we can identify will tend to increase the authorability of update rules. For example, can we impose symmetries such as moving up by two categories is twice as hard as moving up by one, or moving down one category is equally as hard? Is there anything we can say about altering matrix entries to reflect various probabilities of guessing and slipping? Is it important to preserve the distribution shape during updates? Should rules be designed to narrow the distribution with repeated applications, reflecting increased confidence, or should they be designed to resist the natural narrowing of the distribution which permanently discards some detail? And in terms of the skill hierarchy, what is the benefit of encoding additional information known to impact performance, such as skill prerequisite relations? These benefits need to be balanced against any additional authoring load.

Another open question is how models of this type should be used across multiple scenarios. We initially envisioned the model as tracking longitudinal progression across a corpus of scenarios. In other words, "proficiency" referred to a learner's proficiency in the domain rather than within a specific training scenario. However, one challenge with this usage is that the rate of proficiency change within a single scenario is typically small, resulting in few dynamic changes to the overall tailoring strategies. Using the model with rules that make larger changes to estimate proficiency within each scenario enables more scenario-specific tailoring and more sensitivity to learning gains, but the estimates at the end of the training session may have marginal utility for subsequent scenarios.

We are evaluating and exploring several different ways to address the difference between domain proficiency and proficiency as demonstrated in a particular training session. These include the use of skill testing approaches to "set" proficiency for each scenario (Folsom-Kovarik et al., 2013), incorporating scenario-specific proficiency estimates into longitudinal models, and a hybrid approach. The hybrid approach is to elaborate each longitudinal proficiency node with a "child" node for proficiency in each scenario that requires that skill. We expect the scenario-specific estimates will change quickly while contributing to an overall proficiency estimate for the learning objective that changes more slowly. We are evaluating whether the hierarchical propagation algorithm described earlier is sufficient for this usage, or whether this approach requires additional contextual update rules that are sensitive (via tags) to node type (i.e., longitudinal or scenario-specific). The expected advantage of this hybrid approach is that the analytic, informal, and transparency properties of the model can be maintained uniformly across the span of a training course while also providing fine-grained proficiency estimates within each scenario for dynamic tailoring.

In conclusion, we have discussed proficiency models from the viewpoint of giving instructors an analytic, informal, and transparent understanding of trainee proficiency. We described a fuzzy vector model enhanced with training context information and skill relationships. This model enables tailoring a training scenario to give training individualization, unobtrusive assessment and intervention, and increased learning impact without imposing undue burden on end users.

5. References

Almond, R.G., Shute, V.J., Underwood, J.S., & Zapata-Rivera, J.-D. (2009). Bayesian networks: A teacher's view. *International journal of approximate reasoning, 50*(3), 450–460.

Anderson, J.R. (1982). Acquisition of cognitive skill. *Psychological Review, 89*(4), 369.

Anderson, J.R. (1993). *Rules of the Mind*. Hillsdale, NJ: Lawrence Erlbaum Associates.

Carbonell, J. (1970). AI in CAI: An artificial-intelligence approach to computer-assisted instruction. *IEEE Transactions on Man-Machine Systems, 11*(4), 190–202.

Corbett, A.T., & Anderson, J.R. (1995). Knowledge tracing: Modeling the acquisition of procedural knowledge. *User modeling and user-adapted interaction, 4*, 253-278.

Fautua, D., Schatz, S., Kobus, D., Spiker, V.A., Ross, W.A., Johnston, J.H., et al. (2010). *Border Hunter Research Technical Report* (ISTTR1001). Orlando, FL: Institute for Simulation and Training.

Folsom-Kovarik, J.T., Wray, R.E., & Hamel, L. (2013, July 9-13). *Adaptive assessment in an instructor-mediated system.* Paper presented at the 16th International Conference on Artificial Intelligence in Education, Memphis, TN.

Henrion, M. (1987). Some practical issues in constructing belief networks. In L. Kanal, T. Levitt & J. Lemmer (Eds.), *Uncertainty in Artificial Intelligence* (pp. 161-174).

Katz, S., & Lesgold, A. (1991, August 30). *Modeling the Student in Sherlock II.* Paper presented at the International Joint Conference on Artificial Intelligence Workshop on Agent Modeling for Intelligent Interaction.

Klein, G., Moon, B., & Hoffman, R.F. (2006). Making sense of sensemaking I: Alternative perspectives. *IEEE Intelligent Systems, 21*(4), 70-73.

Lee, C.-C. (1990). Fuzzy logic in control systems: fuzzy logic controller---Part I. *IEEE Transactions on Systems, Man and Cybernetics, 20*(2), 404-418.

Lesgold, A., Lajoie, S., Bunzo, M., & Eggan, G. (1988). SHERLOCK: A coached practice environment for an electronics troubleshooting job. In J. Larkin, R. Chabay & C. Scheftic (Eds.), *Computer assisted instruction and intelligent tutoring systems: Establishing communication and collaboration* (pp. 49–80). Hillsdale, NJ: Lawrence Erlbaum Associates.

Magerko, B., Laird, J.E., Assanie, M., Kerfoot, A., & Stokes, D. (2004). AI characters and directors for interactive computer games. *Ann Arbor, 1001*(48), 109-2110.

Newell, A., & Rosenbloom, P.S. (1981). Mechanisms of skill acquisition and the law of practice. *Cognitive skills and their acquisition*, 1-55.

Oser, R.L., Cannon-Bowers, J.A., Salas, E., & Dwyer, D.J. (1999). Enhancing Human Performance in Technology-Rich Environments: Guidelines for Scenario-Based Training. *Human/Technology Interaction in Complex Systems, 9*, 175-202.

Riedl, M.O., Stern, A., Dini, D.M., & Alderman, J.M. (2008). Dynamic experience management in virtual worlds for entertainment, education, and training. *International Transactions on Systems Science and Applications, Special Issue on Agent Based Systems for Human Learning, 4*(2), 23-42.

Schatz, S., Oakes, C., Folsom-Kovarik, J.T., & Dolletski-Lazar, R. (2012). ITS + SBT: A Review of Operational Situated Tutors. *Military Psychology (special issue on current trends in adaptive training for military applications), 24*(2), 166-193.

Schatz, S., Wray, R.E., Folsom-Kovarik, J.T., & Nicholson, D. (2012, October 22-26). *Adaptive Perceptual Training in a Virtual Environment.* Paper presented at the 56th Human Factors and Ergonomics Society Annual Meeting (HFES), Boston, MA.

Wray, R.E., Lane, H.C., Stensrud, B., Core, M., Hamel, L., & Forbell, E. (2009). *Pedagogical experience manipulation for cultural learning.* Paper presented at the Conference on Artificial Intelligence in Education, Brighton, England.

Wray, R.E., & Woods, A. (2013). *A Cognitive Systems Approach to Tailoring Learner Practice.* Paper presented at the 2nd Advances in Cognitive Systems Conference, Baltimore, MD.

Wray, R.E., Woods, A., & Priest, H. (2012). *Applying Gaming Principles to Support Evidence-based Instructional Design.* Paper presented at the Interservice/Industry Training, Simulation & Education Conference (I/ITSEC).

Acknowledgement

This work is supported in part by Office of Naval Research (ONR) contract N00014-11-C-0193. 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 ONR or the US Government. The US Government is authorized to reproduce and distribute reprints for Government purposes notwithstanding any copyright notation hereon.

We would like to thank Mark Core, Laura Hamel, H Chad Lane, Brian Magerko, Brian Stensrud, and Angela Woods, who helped shape the design and implementation of the current model in earlier explorations and iterations of proficiency modeling for situated, tailored training.

Author Biographies

JEREMIAH T. FOLSOM-KOVARIK is a Research Scientist in the intelligent training area of Soar Technology, Inc. His research addresses innovative ways adaptive systems can learn about, model, and influence human behavior, cognition, and learning.

CHARLES NEWTON is an Artificial Intelligence Engineer at Soar Technology. His research and development experience covers human behavior modeling, natural language-based systems, and design patterns in intelligent systems.

JOSHUA HALEY is a Software Engineer at Soar Technology and a graduate student at the University of Central Florida, where he studies intelligent systems and machine learning approaches.

ROBERT E. WRAY is a Senior Scientist at Soar Technology. His research encompasses many areas of artificial intelligence including knowledge representation and ontology, agent-based systems and agent architectures, artificial intelligence in education, and cognitive science.