

Reflecting on Cognitive Process Models to Enhance Adaptive Training

Mandira Hegde, Dan Allison, Todd W. Griffith, Ph.D.

Discovery Machine, Inc.

Williamsport, PA

mhegde@discoverymachine.com, dallison@discoverymachine.com, tgriffith@discoverymachine.com

ABSTRACT

This paper presents work sponsored by Phase I Small Business Innovation Research (SBIR) funding from the Office of Naval Research to develop a system that combines valuable aspects of intelligent tutoring systems and human-to-human adaptive training to aid instructors in providing individualized training. Our system, the Virtual Observer Controller for Adaptive Training (VOCAT), is an adaptive, intelligent virtual observer controller that has been built and demonstrated to run end-to-end in a synthetic training event example. Human instructors recognize that observed student actions may be the result of many possible cognitive reasoning paths and are able to provide student-specific guidance accordingly. VOCAT aims to aid instructors and enhance human-led adaptive training by reflecting on student-specific cognitive process models in order to: 1) identify how and when to provide student-specific guidance during training and 2) identify students' cognitive reasoning behind key decision points for use in an after action review.

ABOUT THE AUTHORS

Mandira Hegde is a Knowledge Engineer at Discovery Machine, Inc. She graduated from Case Western Reserve University in 2011 with a Bachelor of Science degree in Systems Biology and a minor in Cognitive Science. She has worked on various behavior modeling projects at Discovery Machine, focusing on building deployable knowledge models of simulated entities within constructive military training environments. Her past experience includes neuroimaging research with Case Western, the Cleveland Clinic, and New York University.

Dan Allison is the VP of Professional Services at Discovery Machine, Inc. He has 25 years of experience in system engineering and modeling in a variety of businesses including DoD SONAR and RADAR systems, medical device design and military training systems. His most recent work focuses on knowledge engineering using the Discovery Machine Knowledge Capture Methodology and building deployable knowledge models using Discovery Machine's Modeler software. Mr. Allison earned a BSEE from Penn State and MSEE from Syracuse University and is six sigma certified. He has received two patents on the topic of deconvolution algorithms and has been an active member of both IEEE and INCOSE.

Dr. Todd W. Griffith is the founder and CTO of Discovery Machine, Inc. He has been working in the area of intelligent systems research for over twenty years and has published papers in the areas of cognitive science, human-computer interaction, and intelligent systems. He has presented regularly at conferences and meetings since 1996 and has received four patents in the area of knowledge systems. Prior to founding Discovery Machine, Dr. Griffith taught computer science and artificial intelligence at Georgia Institute of Technology and Bucknell University. Dr. Griffith received his Ph.D. in computer science and artificial intelligence in 1999 from Georgia Tech with a focus on modeling the problem solving strategies of Subject Matter Experts (SMEs).

Reflecting on Cognitive Process Models to Enhance Adaptive Training

Mandira Hegde, Dan Allison, Todd W. Griffith, Ph.D.

Discovery Machine, Inc.

Williamsport, PA

mhegde@discoverymachine.com, dallison@discoverymachine.com, tgriffith@discoverymachine.com

INTRODUCTION

Individualized, human-to-human adaptive training has been shown to be 98% more effective than traditional, group-based instruction (Bloom, 1984). While highly effective, this approach is neither time nor cost-efficient for maritime training purposes; its efficacy relies heavily on the instructor's ability to determine how and when to intervene to address the cognitive problems of the student (Porayska-Pomsta et al., 2008). In contrast, traditional intelligent tutoring systems are more efficient but only 36% more effective than group-based methods, lacking the effectiveness of one-on-one, human-led training (Dede, 2008).

Our solution, the Virtual Observer Controller for Adaptive Training (VOCAT), is an adaptive, intelligent virtual observer controller that combines valuable aspects of intelligent tutoring systems and human-to-human adaptive training to aid instructors in providing efficient and effective individualized training. The VOCAT system is used by an instructor to monitor student progress, adapt the training scenario in real time, and collect quantitative data for use in the after action review (AAR). VOCAT extends the science of real-time adaptive training for the Navy through its unique approach of utilizing student-specific cognitive process models to aid during training. This approach, supported by Phase I Small Business Innovation Research (SBIR) funding from the Office of Naval Research, serves as the focus of this paper.

In this paper, we begin with background information on the knowledge capture and behavior modeling approach used in developing the VOCAT system. Next, we describe the VOCAT system's authoring console and use of training task blocks as well as an overview of the VOCAT model architecture. We then describe a framework for the creation of student-specific cognitive process models and the implementation of a critic model within VOCAT that infers cognitive reasoning patterns from the models for training use. We end with an example of the VOCAT system running end-to-end in a synthetic training scenario, showing its technical merit, as well as a discussion of planned future work.

BACKGROUND

Knowledge Capture Methodology

Throughout the research effort described in this paper, we have used a knowledge capture methodology and toolset designed to help subject matter experts (SMEs) recall and articulate their tacit knowledge. The Discovery Machine® Methodology is a patented approach to SME knowledge acquisition resulting in knowledge which can be easily modeled. Iterative phases within this methodology -- expert domain scoping, conceptualize, formalize, operationalize, and test/deploy -- proceed from abstract to more concrete, resulting in executable models that perform tasks similar to the way in which a SME would (Sharp & Potts, 2011).

We used this knowledge capture methodology specifically on Anti-Submarine Warfare Evaluator (ASWE) instructors to extract knowledge about how they operate as observer/controllers during training events. This knowledge was used to construct the VOCAT training task blocks and student cognitive process models described later in this paper.

Behavior Modeling Approach

We have employed a modeling approach which represents knowledge captured from Subject Matter Experts (SMEs) in a language called Task-Method Knowledge (TMK). Our entire VOCAT system, including the VOCAT model, training task blocks, and student cognitive process models, have been constructed in TMK. TMK models provide a mechanism by which a set of process models is created and stored from the outset. Process models are represented as task decomposition hierarchies made up of three types of elements shown in Fig. 1: tasks, methods, and procedures.

Tasks represent particular actions to be completed and are depicted in TMK models as yellow rectangles. These tasks decompose into method child elements, depicted as blue ovals, and/or procedure child elements, depicted as yellow rectangles with black stripes. These two child elements represent alternative options for accomplishing the parent task (Potts et al., 2010).

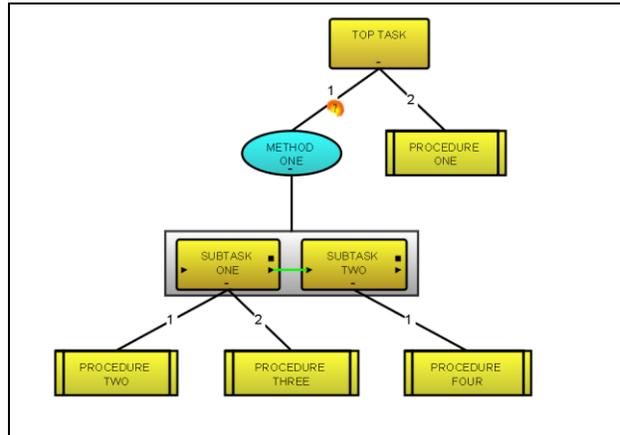


Figure 1. Example TMK Model

Authoring Console and NTA Blocks

On previous projects, we have developed domain-specific authoring consoles to allow instructors and SMEs the ability to author custom behavior models specific to their training. For this research effort, we leveraged this previous work to construct an authoring console for VOCAT models that gives instructors the ability to build models specific to the set of training objectives at hand, as shown in Fig. 2.

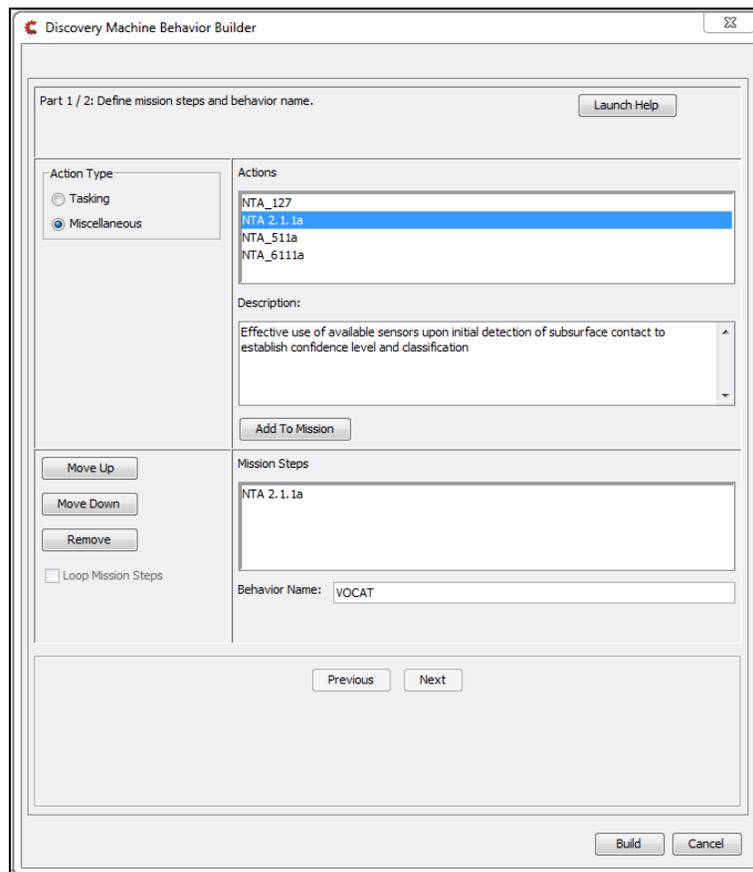


Figure 2. VOCAT Authoring Console

We used Navy Training Task (NTA) blocks to represent these training objectives. Table 1 shows training objective descriptions for a sample of NTAs. Each NTA block encapsulates a particular training objective. For example, NTA block 2.1.1 monitors the student's ability to effectively use his available sensors upon initial detection of a subsurface contact in order to establish confidence level and classification.

Table 1. Example NTAs with Training Objective Descriptions

NTA Number	NTA Name	Training Objective Description
NTA 1.2.1	Water Space Management	Tests how quickly students establish a No Attack Zone
NTA 2.1.1	Collect Target Information	Tests students' use of available sensors upon initial detection of a subsurface contact to establish confidence level and classification
NTA 5.1.1	Display Tactical Picture	Tests students' use of displayed tactical picture to maintain situation awareness
NTA 6.1.1	Protection of Naval Forces	Tests students' utilization of appropriate defensive formations, screens, and maneuvers

During authoring, the instructor selects any combination of NTA blocks to be added to the VOCAT model. The authoring console uses the model template described in the next section to construct a fully executable VOCAT model. The selected NTA blocks execute concurrently while the VOCAT model is running during training. The VOCAT model assists the instructor by monitoring student performance and intervening in training as necessary, ensuring that the training session covers all of the training objectives encapsulated within the selected NTA blocks.

VOCAT MODEL ARCHITECTURE

High-Level View

All authored VOCAT models consist of two simultaneously executing branches at the top level, shown in Fig. 3.

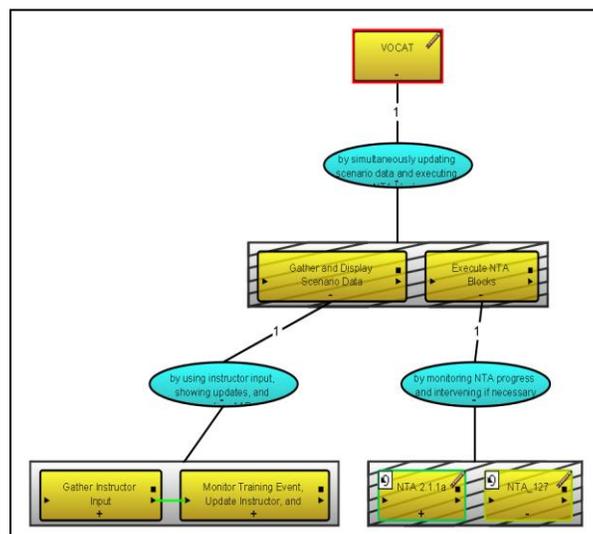


Figure 3. High-Level View of VOCAT Model

The first branch, 'Gather and Display Scenario Data', receives and comprehends input from the instructor and the training environment, performing situation awareness processing. The second branch, 'Execute NTA Blocks', contains the NTA building blocks and filters information from the first branch with regard to the training objectives.

Situation Awareness Processing

The 'Gather and Display Scenario Data' branch provides the VOCAT model with situation awareness (SA) level one perception data as described by Endsley (1995). The SA of the VOCAT model is similar to the SA of an instructor monitoring a training event. The branch begins to build its SA by gathering key information about the student and the scenario from the interfaces shown in Fig. 4 and Fig. 5.

The screenshot shows a software window titled "Scenario Information and Updates" with a tabbed interface. The "Student Information" tab is active, displaying four sections for data entry:

- Sonar Knowledge:** Includes instructions for selecting basic, intermediate, or advanced levels based on the student's understanding of sonar. The dropdown menu is currently set to "Basic".
- Weapons Knowledge:** Includes instructions for selecting none, indirect, or direct experience with weapons. The dropdown menu is currently set to "Direct".
- Assertiveness of Student:** Includes instructions for selecting high or low assertiveness based on decision-making speed. The dropdown menu is currently set to "High".
- Knowledge of Adversary:** Includes instructions for selecting basic or advanced understanding of the adversary. The dropdown menu is currently open, showing "Basic" and "Advanced" options.

A "Start Training" button is located at the bottom right of the window.

Figure 4. Student Information Input Screen

Figure 5. Scenario Information Input Screen

This information represents the a priori knowledge of the instructor when running a training event. While the training scenario is running, the branch continues to update its situation awareness by monitoring all entities in the simulation and gathering data necessary for an AAR. Such data may include entity positions and their relative distances, elapsed time, and student sensor use.

NTA Block Execution

The second branch, the NTA branch ('Monitor Training Objectives') monitors student actions with regard to the NTA blocks selected for the training exercise. When particular student actions are observed, the VOCAT model will reflect on the appropriate student model to infer why the student performed the action and then determine the best course of action for instruction.

REFLECTION ON STUDENT-SPECIFIC COGNITIVE PROCESS MODELS

Generating Student-Specific Cognitive Process Models

The first input screen presents the instructor with the opportunity to enter information specific to the student performing the training. While some tutoring systems classify students into general categories ranging from novice to expert, SME knowledge indicates that instructors adapt training based on expertise across different categories. For ASWE training in particular, instructors consider students' knowledge of SONAR, weapons usage, and adversary tactics. VOCAT gathers input from the instructor for these categories in order to select the appropriate student cognitive process model. The selected cognitive process model is a representation of the instructor's knowledge of the student's decision making process for the training objectives encapsulated within the chosen NTA blocks.

We used the knowledge capture methodology previously described on ASWE SMEs to derive the information necessary to construct these student cognitive process models. We constructed student cognitive process models at varying proficiency levels for each NTA block. Fig. 6 shows two possible cognitive process models that may be selected based on instructor input for NTA block 2.1.1, "Collect Target Information."

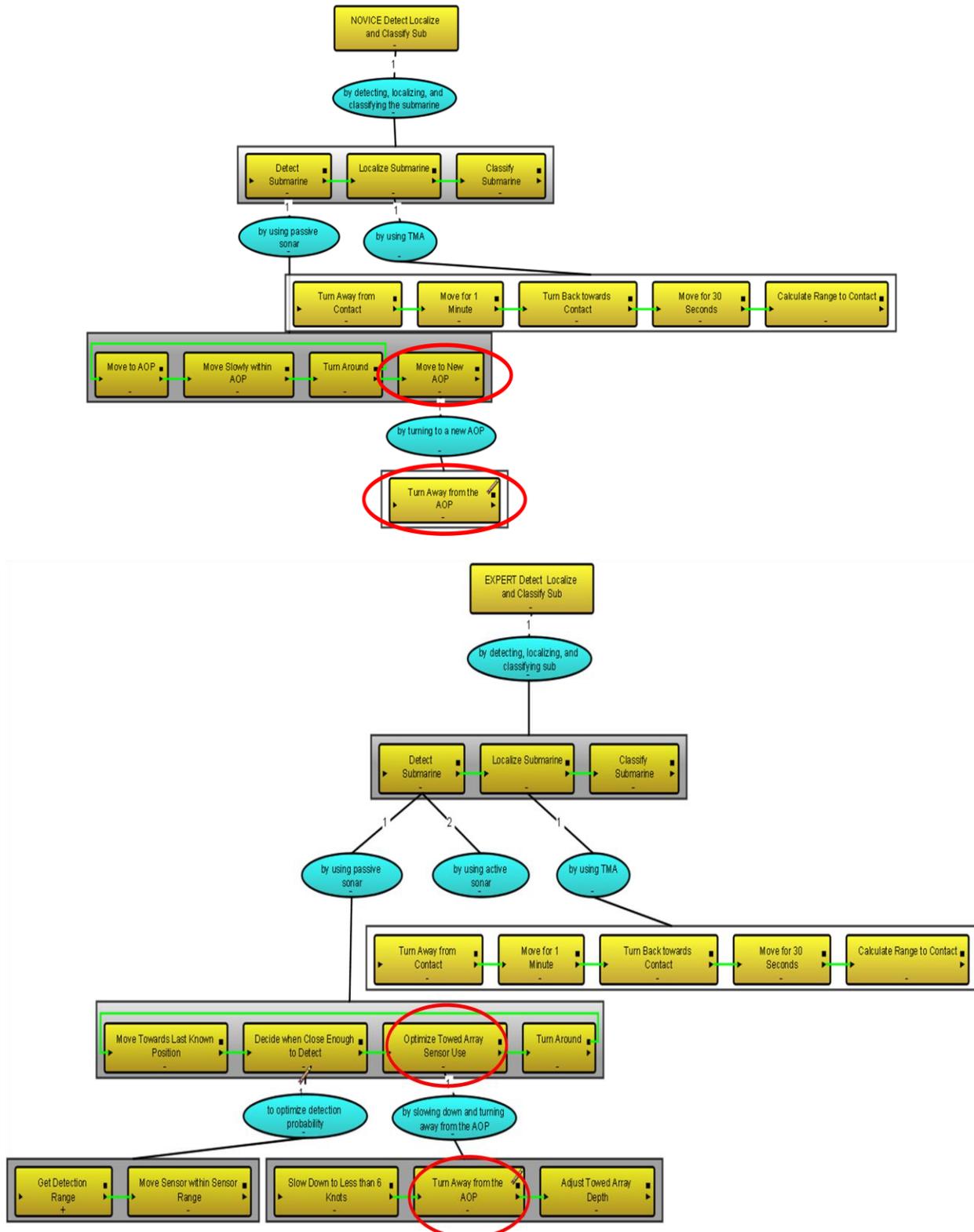


Figure 6. Novice and Expert Student Cognitive Process Models

Note that we use TMK to represent this cognitive process of the student. The novice model is selected when the instructor selects the lowest level of expertise for each category. Conversely, the expert model is selected when the

instructor selects the highest level of expertise for each category. Models representing intermediate levels of expertise are selected for students with varying levels of expertise across the different categories. As shown in Fig. 6, the expert model is more ablated than the novice model, representing the more complex decision making process that experts use compared to novices.

Reflection on Cognitive Process Models during Training

The ability to infer the student's reasoning is achieved through use of a critic model built into the VOCAT model. A critic model is a TMK hierarchy that uses DMI's reflection technology to understand or reflect on a representation of the student's decision making process, what we call the student cognitive process model. Through our knowledge capture process, we have identified student actions that would be of notable concern to instructors during various training objectives ("trigger actions"). These trigger actions prompt the critic model to reflect on the selected student model and search for the task which matches the observed student action. Once this task has been found in the cognitive process model, the critic model references its parent task to infer why the action was initiated. This works because, as in a typical task decomposition, the lower-level child elements in our TMK models are the set of actions performed to achieve the higher-level parent tasks. Conversely, the higher-level parent tasks describe why subtasks are being performed at a lower level.

Fig. 6 shows the student cognitive process models with the task containing the trigger action of "Turn Away from the AOP (Area of Probability)" and parent task circled in both models. The reason VOCAT would infer for this trigger action would depend on which cognitive process model was selected for the student. If the novice cognitive process model were selected, VOCAT would infer the reason the student turned his ship would be to move to a new AOP. On the other hand, if the expert cognitive process model were selected, VOCAT would infer the student was turning away from the AOP to optimize use of a towed array.

VOCAT Intervention during Training

VOCAT displays messages to alert the instructor when the student performs trigger actions. An example can be seen in Fig. 7. Additionally, if reflection on the student cognitive process model deems that the student needs help completing an NTA objective, VOCAT adapts the training scenario in real time in order to help the student, just as an instructor would.

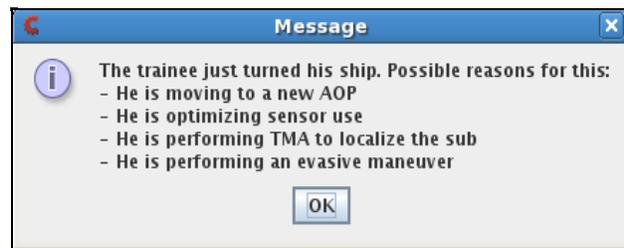


Figure 7. Alert Message to Instructor

We have identified effective ways for the VOCAT system to intervene during various situations from knowledge capture sessions with SMEs. In the previous example of the student turning away from the AOP, VOCAT would intervene if the reason the student turned away was to move to a new AOP but not if the reason was to optimize the use of the towed array.

In addition to intervening when a student performs a trigger action, VOCAT alerts the instructor and intervenes during training based on general scenario information, as seen in Fig. 8. For example, if the student is in the process of completing NTA 2.1.1, "Collect Target Information", half the training scenario time has elapsed and the student has yet to make contact on the sub, VOCAT tells the sub to ping in order to help the student find it. If an even smaller percentage of time is remaining, VOCAT tells the sub to fire a torpedo to further help the student find the sub.

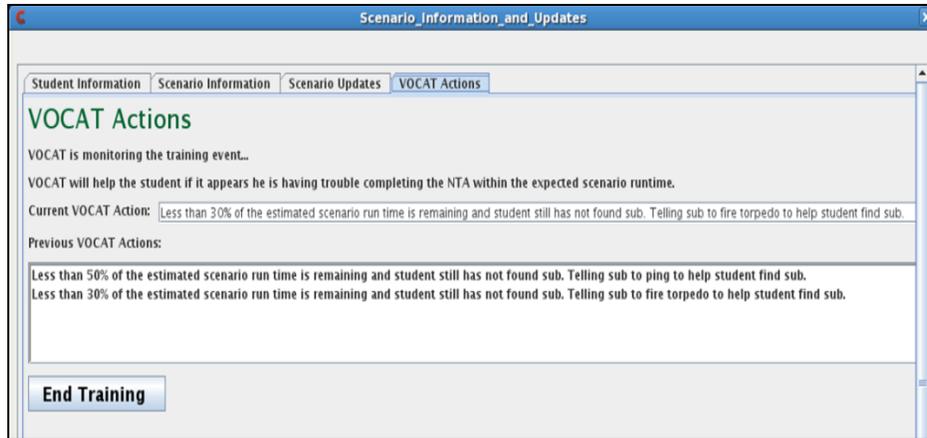


Figure 8. VOCAT Actions Screen

VOCAT intervenes differently depending on the objective the student is trying to complete. For other NTAs, such as NTA 1.2.1, "Water Space Management", VOCAT could direct the sub to create situations where the student must maneuver the force in order to avoid conflicts in the area. For NTAs regarding tactical oceanographic analysis, VOCAT uses the knowledge of active and passive ranges of the day to move the sub in or out of range.

DISCUSSION OF TECHNICAL FEASIBILITY

Anti-Submarine Warfare (ASW) Synthetic Training Event Test

We constructed a synthetic scenario, similar to training event scenarios we have observed at the Fleet ASW Training Center, San Diego, with the purpose of testing the technical merit of the VOCAT system. We began by using the authoring console to construct a VOCAT model containing NTA block 2.1.1, "Collect Target Information." Our synthetic scenario, constructed in Joint Semi-Automated Forces (JSAF), contains a student ship, enemy submarine, and high value unit (HVU), as shown in Fig. 9. The submarine was assigned the JSAF behavior of moving on a heading away from the student. We manually controlled the student ship and had the HVU moving with the student. We played the role of the instructor, entering input into the student and scenario information screens. We selected novice inputs for each of the student input categories and ran the scenario for ten minutes. We ran the scenario for ten minutes for demonstration purposes although a real training event would be on the order of 60 minutes.

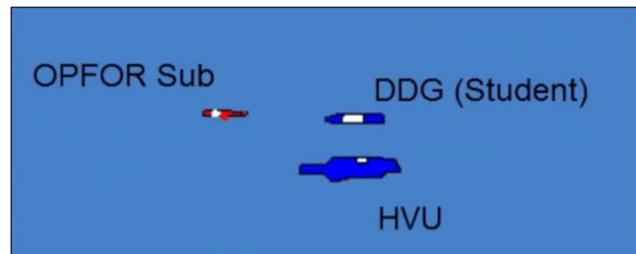


Figure 9. JSAF Training Scenario

After 50% of the scenario run time had elapsed, we had yet to localize the sub. We saw that the instructor had been updated with the action VOCAT was performing in order to help the student, which was having the sub ping to help the student find the sub. After 80% of the scenario run time had elapsed, we saw that the instructor had been alerted that the student had not yet found the sub and that VOCAT had told the sub to fire a torpedo to help the student find it. In the remaining few minutes of the scenario, we turned the student ship away from the sub. VOCAT successfully alerted us to this action and turned the sub in the direction of the student to further help the student find the sub.

After 100% of the scenario had elapsed, VOCAT ended the training scenario and created a file containing key information from the training scenario, shown in Fig. 10. This report contained accurate distance measurements from the student ship to the sub throughout the training scenario, which is information that SMEs have identified as key to ASWE instructors. VOCAT also documented the closest distance the student got to the sub along with the time at which it occurred. In addition, it documented the trigger action performed by the student, the reason for the action, and the time at which it occurred.

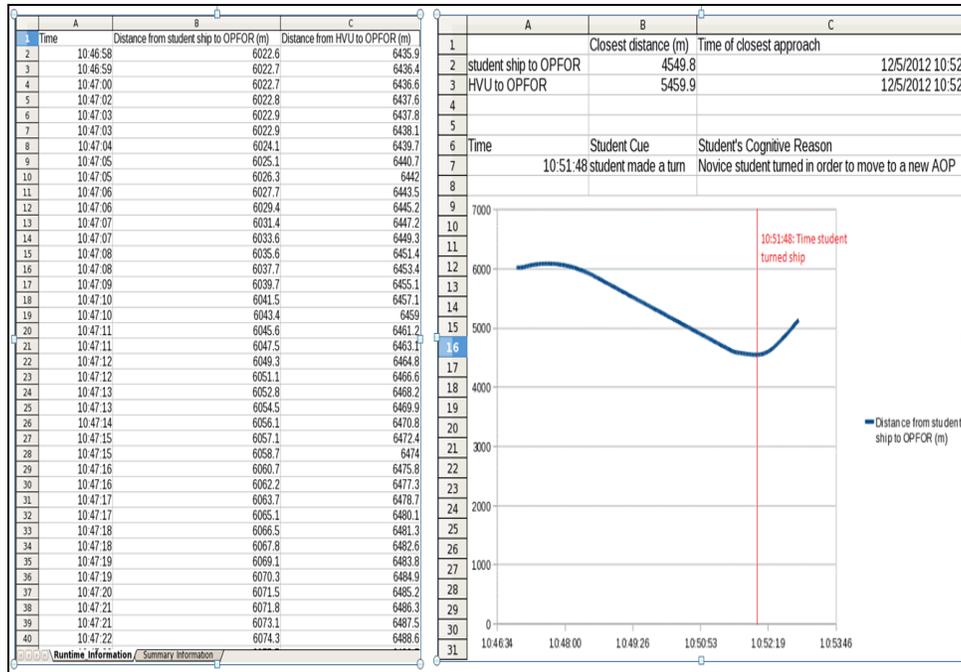


Figure 10. AAR Data

Our ASW synthetic training event test has demonstrated that the VOCAT system is able to run end-to-end and is thus feasible from a technical standpoint. In addition, our test identified three ways in which the VOCAT system has demonstrated the potential to successfully aid instructors and enhance adaptive training. In the synthetic training event test, VOCAT first aided the instructor by focusing the instructor's attention on key information, including entity positions, student sensor usage, and elapsed time, in real time. Second, VOCAT selected the correct student cognitive process model based on instructor input and reflected on it in order to determine how and when to intervene during training, freeing the instructor from having to do so. Third, it saved key information from the training session, including relevant entity distances, trigger actions, and the inferred reasons behind the actions to aid the instructor during AAR.

Addressing Current Intelligent Tutoring System Criticisms

In addition to building a system that enhances adaptive training and aids instructors during training, we aimed to bridge the gap between traditional intelligent tutoring systems and one-on-one instructor-led training by addressing two criticisms intelligent tutoring systems commonly face.

The first criticism is the failure of many intelligent tutoring systems to ask students why they perform certain actions during training. Without knowledge of the reasons behind these actions, it is difficult for tutoring systems to determine how and when to offer instruction (Jonassen, 1995). We addressed this limitation by creating student cognitive process models for VOCAT to reflect on during training, as previously described. Knowing why students performed trigger actions helps VOCAT to intervene effectively. It is important to note that the student cognitive process models represent instructors' knowledge of the decision-making reasoning of their students. Therefore, it is possible that there are discrepancies between the reasoning represented in the cognitive process models and the actual reasoning behind the students' actions (i.e. VOCAT does not surpass the effectiveness of one-on-one instructor-led training). However, as previously described, all trigger actions and decision points from training sessions are documented in the AAR created after the training session ends. This provides instructors the ability to review key decision points with students and ask the right questions.

The second criticism is the immediate feedback offered by many intelligent tutoring systems before students have the time to work through challenges on their own. Often times, when feedback is immediately offered during training, students immediately turn to the help before trying to solve problems or complete tasks. Thus, this style of instruction is criticized for its failure to develop deep learning in students (Koedinger & Alevan, 2007). Unlike these

systems, VOCAT does not immediately intervene during training. Rather, as previously described, it intervenes only after it determines that students need help or after certain percentages of time have elapsed, as suggested by SMEs. This ensures that students have had enough time to attempt to complete tasks independent of outside instruction. In addition, VOCAT does not attempt to complete tasks for students. Instead, it aims to provide just the right amount of help to allow students to finish tasks on their own.

CONCLUSION

In this paper, we have presented VOCAT, an adaptive, intelligent virtual observer controller built to aid instructors to help naval students meet training objectives. It does so by reflecting on student cognitive process models during training. During a training event, the VOCAT system is used by an instructor to monitor student progress, adapt the training scenario in real time, and collect data for use in an AAR. After training, the instructor can use the data collected by VOCAT in an AAR with the student. Our technical feasibility test has demonstrated that VOCAT is able to run end-to-end and has the potential to be of aid to instructors.

We have visited Navy training facilities, witnessed several live training events, and discussed with ASWE instructors the potential of VOCAT to aid in ASWE training. Thus, the next phase of our work is to expand VOCAT's capability to run in real training scenarios. We plan to collect data that will provide us with a quantitative measure of the VOCAT system's effectiveness in aiding instructors during training. We also plan to test whether the use of VOCAT enhances adaptive training by comparing student performance during training with and without the use of VOCAT. Additionally, we plan to collect feedback from instructors and students on the usefulness of the data and student cognitive reasoning VOCAT provides in the AAR.

Our work on VOCAT had also sets grounds for future work in the areas of adaptive, intelligent training and AAR. In particular, we are interested to see whether VOCAT's use as a training aid will reduce instructor-to-instructor variation in training, and consequently, student learning. In addition, we are interested in allowing instructors to author their own NTA blocks and use their own expertise to determine how VOCAT should intervene during training. Finally, while our work so far has focused on ASWE training, we are eager to expand our efforts to enhance training across the Defense sector. We look forward to working with others in the training and simulation community to further advance these areas.

ACKNOWLEDGEMENTS

We would like to thank the Office of Naval Research for their support of the work described in this paper.

REFERENCES

- Bloom, B.S. (1984). The Search for Methods of Group Instruction as Effective as One-to-One Tutoring. *Educational Researcher*, 13(6), 4-16.
- Dede, C. (2008). Theoretical Perspectives Influencing the Use of Information Technology in Teaching and Learning. *Springer International Handbook of Information Technology in Primary and Secondary Education*, 20, 43-62.
- Endsley, M.R. (1995). Toward a Theory of Situation Awareness in Dynamic Systems. *Human Factors*, 37(1), 32-64.
- Jonassen, D.H. (1995). Computers as Cognitive Tools: Learning *with* Technology, Not *from* Technology. *Journal of Computing in Higher Education*, 6(2), 40-73.
- Koedinger, K.R., & Aleven, V. (2007). Exploring the Assistance Dilemma in Experiments with Cognitive Tutors. *Educational Psychology Review*, 19(3), 239-264.
- Porayska-Pomsta, K., Mavrikis, M., & Pain, H. (2008). Diagnosing and acting on student affect: the tutor's perspective. *User Modeling and User-Adapted Interaction*, 18(1-2), 125-173.
- Potts, J.R., Griffith, T., Sharp, J.J., & Allison, D. (2010). Subject Matter Expert-driven Behavior Modeling Within Simulation. *Proceedings of the Interservice/Industry Training, Simulation, and Education Conference 2010*.
- Sharp, J.J., & J.R. Potts. (2011). Improving Trainee Engagement Levels through Adaptive Entity Behaviors. *Proceedings of the Interservice/Industry Training, Simulation, and Education Conference 2011*.