Toward Recommendation across Learning Systems

J.T. Folsom-Kovarik, Ruben Ramirez-Padron, Cameron Copland, Caitlin Tenison	Ian Davidson, Aubrey Gress, Hongjing Zhang	LCDR Peter Walker
Soar Technology, Inc.	University of California –	Office of Naval Research
Ann Arbor, MI	Davis, CA	Arlington, VA
Jeremiah@soartech.com	davidson@cs.ucdavis.edu	peter.walker@navy.mil

ABSTRACT

In modern learning environments such as Ready Relevant Learning, often multiple learning systems work together (e.g. computer-based training, intelligent tutors, and training simulations). Effective recommendation has been well studied within each of these learning systems. However, when several systems are available that train in different ways, a new challenge emerges to understand the data that systems share, in light of their varied instructional designs and understanding of science of learning. Emerging data specifications and machine learning promise to help recommend which learning systems best fit individual learners' needs and desired learning outcome.

An exemplar recommendation component was created to drive training progression in several different learning systems. Recommendations were produced by combining and deconflicting learner information from multiple systems. Experiments with historical data and simulated students showed that the recommendation component could prioritize available learning systems and content adaptively. The recommender successfully inferred needed science of learning information, such as relating learning activities to skills and estimating the varying difficulty of skills.

The initial research reported here focuses on objective learner performance metrics. Our results show that the recommender accurately matched ground truth in estimating learner mastery and skill difficulty. The recommender also incorporated simulated input from human instructors, which reduced its error rate to near zero. Finally, a simple exemplar algorithm deconflicted learner mastery estimates from different learning systems and used them to give learners qualitatively different recommendations.

The feasibility demonstration reported here enables a 2018 human-participants study applying the same components to additional factors. Subjective learner states (boredom, confusion) and science of learning facts about training (well-defined vs. ill-defined, introductory vs. worked example, static vs. interactive) can drive the same recommendation tools (shared data specifications and machine learning). Our simulation studies suggest that recommendation across learning systems will make real training more effective than the sum of its parts.

ABOUT THE AUTHORS

J.T. Folsom-Kovarik, Ph.D. is the lead scientist at Soar Technology, Inc. for adaptation and assessment within intelligent training. His research combines modern data science and machine learning with SoarTech's trusted expert models of human cognition. Together, the approaches yield intelligent decisions in real-world training settings when available data is small, concepts evolve over time, or nontechnical users need to control the training.

Ruben Ramirez-Padron, Ph.D. is a research scientist within Intelligent Training at Soar Technology, Inc. He has over twenty years of relevant experience that includes academia, R&D, and professional software design and development. He contributes in the fields of machine learning, data science, cognitive modeling, knowledge-based systems and data-intensive Web-scale applications.

Cameron Copland is an Artificial Intelligence Engineer at Soar Technology, Inc. He works in intelligent training and decision-making domains to design and implement AI systems that make inferences about learner behavior.

Caitlin Tenison, Ph.D. is a research scientist at Soar Technology, Inc. Through her research she investigates how people learn, explores methods for supporting learning, and builds technologies that improve training through evidence-based approaches. Her work is characterized by the use of multi-modal data to provide convergent evidence about the cognitive state and skill of an individual.

Ian Davidson, Ph.D. has been a 2017/2018 visiting fellow at the Collegium de Lyon which is part of RFIEA (French Network of Institutes of Advanced Studies). In August 2018 he will return as a faculty member in the department of computer science at University of California – Davis where he is a Professor and has been for over a decade. His research interests include data mining, AI and machine learning and in particular adding human guidance to learning and data mining.

Aubrey Gress, Ph.D. received his undergraduate and master's degrees from Stanford University and a Ph.D. in computer science from U.C. Davis in 2018. He is currently a machine learning engineer at LinkedIn. His research interests include novel forms of machine learning, in particular small data problems.

Hongjing Zhang received the B.S. degree in computer science from Nanjing University, Nanjing, China, in 2015. He received the M.S. degree in computer science from Washington University in St. Louis, MO, USA, in 2017. He is currently pursuing the Ph.D. degree in computer science with University of California – Davis. His current research interests include machine learning, data mining, and their interdisciplinary applications.

LCDR Peter B. Walker, USN, Ph.D. is an Aerospace Experimental Psychologist (AEP) and Program Manager in the Office of Naval Research (ONR). He is a proponent of knowledge discovery and network discovery techniques using supervised and semi-supervised machine learning. His research spans medical and adaptive training domains.

Toward Recommendation across Learning Systems

J.T. Folsom-Kovarik, Ruben Ramirez-Padron, Cameron Copland, Caitlin Tenison
Soar Technology, Inc.
Ann Arbor, MI
Jeremiah@soartech.com

Ian Davidson, Aubrey Gress, Hongjing Zhang
University of California – Davis, CA
davidson@cs.ucdavis.edu

LCDR Peter Walker
Office of Naval Research
Arlington, VA
peter.walker@navy.mil

INTRODUCTION

Powerful change is underway within each branch of the U.S. military as leaders at the highest levels work to realize the benefits of enterprise-class training modernization (Raybourn, Schatz, Vogel-Walcutt, & Vierling, 2017). With modern technology changing the very foundations of the training experience and goals, it now becomes possible to deliver a long-term program of training that is available at the point of need, forms a measurable and evidence-based progression in a training continuum, and adapts content and presentation to individual learners.

Within the Navy, *Ready Relevant Learning* (RRL) is one of three pillars supporting Sailor 2025, the Navy's strategy for developing and retaining personnel at all stages of the career. Chief of Naval Personnel Burke has described RRL as a "top-to-bottom transformation" (Burke, 2018). RRL will make Naval training responsive to evolving threats and requirements, offer the right training at the right time, and make sure Sailors are ready for their current jobs and future progression with minimal down time (Davidson, 2017). In order to carry out this vision, RRL seeks to break free from stove-piped learning systems and fixed, linear content to create modern learning systems that work together and can deliver a myriad of learning pathways to training mastery. RRL draws connections between new and legacy learning systems such as learning management systems, intelligent tutoring systems, and instructor feedback on simulations or exercises. When all these work together, Sailors receive more timely, frequent, and tailored training opportunities.

This paper discusses one technology underpinning RRL: training across the boundaries of learning systems. First, we describe challenges and possible approaches to interoperation of new and legacy systems. Second, to explore these challenges, we assembled cutting edge, open source components and evolving technical standards to form an ecosystem comprising several learning systems. The components together demonstrate a key task: real-time, individualized recommendation regarding what to train next and how to train it. Third, in preparation for an upcoming human-participants study using the same components, we processed historical and simulated learner data to produce results that support the feasibility of combining learning systems into a coherent whole. The simulation studies demonstrated not just sharing data across systems, but making inferences and selecting actions across systems. Our results suggest it will be possible to recommend training that meets individual learners' needs.

Acting Intelligently across Learning Systems

No single database, intelligent tutor, or other learning system contains all the information and capabilities to train every Sailor in the Navy. Indeed, the Navy maintains information about individual Sailors' past training, career progression, job qualifications, performance reviews, and more spread across dozens of data warehouses, including some information that is maintained on paper. The data warehouses in turn are fed by thousands of online and offline courses, bespoke training systems, training expert personnel, simulators, and live exercises. A key problem is that each of these connections form boundaries between learning systems. While the research community has produced excellent examples over the years of intelligent behavior within one learning system, such as learning management systems that automate scoring and assessment tasks or intelligent tutors that provide tailored feedback within a course, these learning systems work best simply because they are specialized to one setting.

Currently, the field is experiencing new interest in sharing information between learning systems. This year several groups have been formed to address multiple aspects of communication between learning systems (e.g. the IEEE Industry Connections / Industry Consortium on Learning Engineering, ieeeicicle.org, and the IEEE Adaptive Instructional Systems Working Group, http://sites.ieee.org/sagroups-2247-1/). However, the instructional design and the implicit understanding of learning science that underlies each separate system remain specialized to that system and are typically not made explicit or shared. This can limit others' ability to correctly interpret and act on shared information (see Challenges, below).

Choosing to share information between systems, even with imperfect knowledge of underlying instructional assumptions, can improve learning. As an example, an experimental learning ecosystem was created by assembling 22 software systems contributed by 11 providers (Gallagher, Folsom-Kovarik, Schatz, Barr, & Turkaly, 2017). The learner-facing systems and supporting components shared learner data as learners switched between systems by means of the developing Total Learning Architecture (TLA; Folsom-Kovarik & Raybourn, 2016; Regan, Raybourn, & Durlach, 2013). A non-experimental study of N=74 participants found that training was able to produce significant learning as measured by gains from pre-test to post-test ($p<.001$, Cohen's $d=1.14$). TLA defined data models and programming interfaces (APIs) for communication across the boundaries of individual systems. At that time, TLA did include science of learning information such as content-competency alignment. However, it did not capture other information such as challenge level, presentation, or stage of learning that would help interpret the shared data.

Since that study, the TLA has improved in several ways. Most relevant to this paper, the TLA now extends existing community specifications in order to specify instructional models of learning systems and science of learning metadata. This provides the missing link needed for learning systems such as our experimental recommender to act intelligently on shared information. The added specifications can describe different instructional models or pedagogical strategies and relate the models to available content. Not all learning activities must fit within the same instructional model or understand other instructional models to function. As a result, a TLA-enabled component for the first time can usefully interpret and act on shared data when systems use different instructional designs and internal assumptions about science of learning. Further description of TLA is available at TLAcommunity.com.

The future impact on RRL promises to be important. As new content is quickly added to keep Naval training current, we will be able to describe how each one contributes to learning and validate whether they are remaining useful over time. Training delivered in different formats or on different devices when in the schoolhouse, at the waterfront, or embarked will become possible to compare and integrate into one unified understanding of Sailor knowledge, skills, and abilities. And when linear training is broken into variable, tailored modules it will be possible for a future recommender component to automate choosing between modules to meet Sailors' needs at any point in their careers.

CHALLENGES AND MULTIPLE SOLUTIONS

We focus on three challenges to recommendation across learning systems that affect RRL and similar learning modernization efforts. First, there is a challenge to understand what each learning system can contribute. This can be addressed by specifying ways to describe instructional strategy and science of learning models. Second, there is a challenge to interpret what learning systems say about learners and build a single unified understanding of a learner. This can be addressed by adding information to shared data that enables interpretation with the instructional models. Third, interpretation can be challenged by large scale and changing data which make it difficult for any researcher or practitioner to correctly record all needed instructional descriptions. Here, machine learning can help automate tasks.

With a collection of learning systems, some of them built long ago, the first unsolved challenge to combining the systems is understanding each system and interpreting its shared data. Does a score of ten in one system mean the same as a score of ten in another system? Have the systems set different mastery thresholds? What instructional design rationale explains the differing thresholds? Making these systems work together requires defining the science of learning aspects that we care about. For example, we need to know that the two systems teach or assess according to different instructional designs, make different assumptions about learning, and fit in different stages of learning. The topics, presentation, and experience within each system makes that system fit differently within a learning timeline.

One solution to the first challenge has been to standardize on one learner model. A classical learner model tells what a learner knows or can do and, sometimes, includes learner states and traits that affect learning. An example with an established, public code base is the Generalized Intelligent Framework for Tutoring (GIFT) (Sottilare, Brawner, Goldberg, & Holden, 2012). GIFT allows any learning system to integrate via APIs and send messages about learner mastery. GIFT can collect learner mastery information and recommend training content in a generalized way, both within and across learning systems. However, there is a clear division of labor. The learning systems require knowledge of GIFT instructional design logic to determine what they communicate. They need to both understand and accept the instructional design held inside GIFT, outside their control, to get useful recommendations. At the same time, learning systems within GIFT make decisions about mastery that require system-specific knowledge. As a result, components outside that one learning system cannot interpret learner experience inside the system. It is only possible to read the final score, but not possible to reason about the different learner experiences that led to the same score.

The second challenge illustrated by the above example is not only sharing data across systems but adding interpretation that makes sense in different systems. A recent effort is under way to standardize communication of fine-grained learner experiences via Experience API (xAPI) (ADL Initiative, 2017). The value of sharing minute-to-minute information about each learner experience is that modern data analytics can use the information to build a detailed understanding of learner needs and context to enable fine-grained and responsive changes in recommendations. It becomes possible to understand the reasons behind a low score or possible caveats on an apparently high score.

In order to reach the full value of sharing learner experience, the evolving TLA specification is adding information channels outside of xAPI to help interpret the data that is shared. For example, does an xAPI message with the same verb always mean the same thing about a learner? Or does the context of one system make that verb more common than in another system? Is the same message acceptable in the context of one learning path while it is simultaneously a negative indicator in another? TLA metadata that links learning systems with science of learning models can help tell the difference. The metadata allows multiple descriptions of training systems, activities, content, and experiences. There is no requirement to use a single instructional design. When more than one description is used, it is possible to identify related and exact synonyms by manually crosswalking or creating links between science of learning models.

The third challenge is that manual process limits the value of standards to address some aspects of real-world training. When there are dozens of systems to connect and thousands of learning activities to describe, manual effort to define metadata in a general way and keep it up to date may require more work than the old method of manually designing a single course of instruction. Furthermore, system descriptions are frequently in flux because of software updates, evolving tactics, techniques, and procedures (TTPs), and constant changes in the learner's context and experience.

The challenge of vast and changing learning metadata brings with it an opportunity for machine learning to combine with human expertise and enable creating detailed, fine-grained, constantly evolving understanding of the science of learning that lets each learning system work together. Future machine learning research could help to understand how training systems impact learners, how they connect to different competency frameworks and models of learning, and how fine-grained differences in learner experience can facilitate understanding individual assessments to detect reliability, bias, or change over time. To demonstrate this, a recommender prototype shows a technical path forward toward keeping all the RRL training systems and content validated and interoperating for impactful adaptive training.

A RECOMMENDER PROTOTYPE

We created an exemplar component (Figure 1) that addresses the challenges in understanding shared data and providing recommendations across learning systems. The component uses machine learning to simultaneously update learner models and science of learning information, such as difficulty to master each skill. It works with humans in the loop to help automate tasks of content-competency alignment and to collect instructor input about learners. Finally, it uses a simple deconfliction algorithm to combine information from multiple systems and make one recommendation.

Learner Modeling

The recommender incorporates two learner models. First, *transfer learning* (Davidson, Gress, & Folsom-Kovarik, 2017) requires only a few direct assessments because it uses information from similar learners to accurately estimate skill mastery. Second, an *additive factors model with slip* is publicly available through LearnSphere (MacLellan, Liu, & Koedinger, 2015). Estimating skill mastery tells when to recommend moving from one learning objective to another.

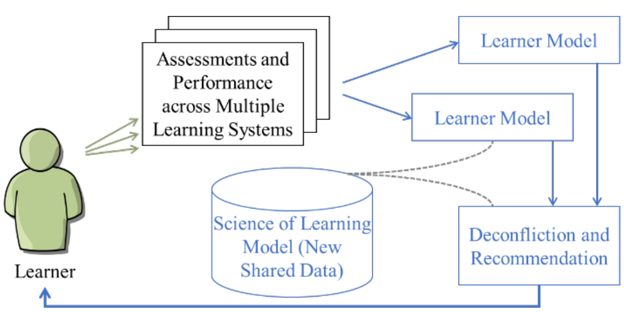

Figure 1: Exemplar recommendation across learning systems.

The recommender is not expected to have training data available before it is used with real learners. Therefore, we further modified the additive factors learner model with online machine learning using a stochastic gradient descent algorithm (Bottou, 2010). The newly added capability is *online* in the sense that it allows the learner model to update the challenge of each skill every time new data arrives, instead of using fixed parameters that are trained before the experiment. This is an example of machine learning updating the shared science of learning model. We used simulated students to show that our change produced good learner estimates after collecting small amounts of data.

Science of Learning and Modeling Training Activities

Like different challenge levels impact the interpretation of learner mastery, science of learning writ large also gives numerous insights into cognitive factors that affect how people learn. Examples include levels of explanation (what versus how), grain size (seconds versus hours), concept specification (ill-defined versus well-defined), or relevance to familiar topics. Any of these can be represented across systems to form the basis of a future recommender. We briefly describe a demonstration of modeling topic relevance in training activities.

Several dozen learning objectives were defined to categorize about two hundred training activities in preparation for study. The manual process involved assigning hundreds of links to tell what each training activity was about, which we realized is not a sustainable process. To augment the manual process word2vec, a freely available deep learning algorithm (Mikolov, Chen, Corrado, & Dean, 2013), was applied to the text of each training activity. Figure 2 shows some visualizations of the results. Individual training activities can be quickly linked (left) with learning objectives in different frameworks, serving as a starting point for expert manual refinement. In addition, proximity between activities (right) can suggest content similar to past experiences, enabling relevance as a factor for recommendation.

Work is ongoing to measure the usability of this machine learning approach. We also hope that applying similar methods with other science of learning dimensions as input will yield useful results. An important path for future work will be understanding what machine learning can tell about science of learning in existing training content.

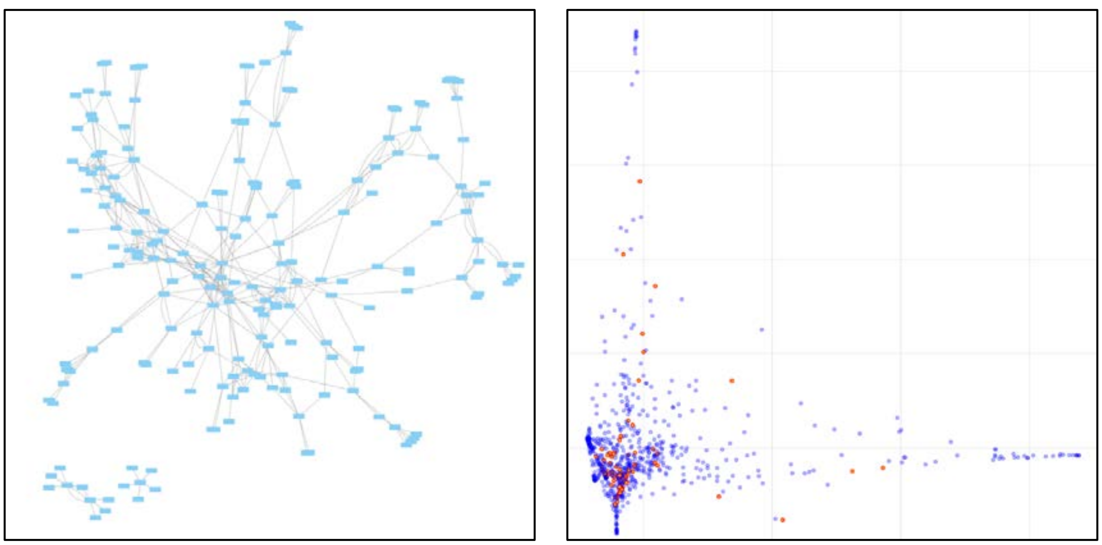

Figure 2: Machine learning can help relate content to learning objectives (links) or to past experiences (distance, right).

Instructor Guidance

Machine learning can produce a predictive model of learners and learning activities from observed assessments. Regression is a popular method of predicting valuable information such as individual learners' ability to succeed in various training activities. However, existing methods of regression work particularly well when there are many labeled data points for training an accurate model. This requires many and precise measurements, which can raise practical challenges in real-world learning systems that must handle noisy or imprecise inputs from humans.

Recently we have explored adding imprecise, even inaccurate human style guidance to inform machine learning regression via a mechanism we called relative guidance (Gress & Davidson, 2016, 2018). Rather than requiring each data point to be labeled with an absolute precise value, we allow machine learning to make better estimates using different types of inputs that are more likely to come from instructors.

We explore four types of instructor guidance:

1. *Similar*. This guidance is of the form: two students are similar, though we do not specify their exact mastery. This may be due to an instructor's observations of performance.
2. *Bound*. Here we allow the specification of an upper and lower bound on one student's performance. This can be determined for example by formative or partial assessments before training is complete.
3. *Relative*. This is instructor guidance that student *i* should have a better performance than student *j*.
4. *Neighborhood*. Here we can make the complex statement that student *i*'s performance should be closer to student *j*'s performance than student *k*'s performance. That is, *i* is more like *j* and less like *k*.

Instructor guidance is assumed to be correct more often than not, but it can still be wrong. Our method is robust to noisy guidance. It is also important to note that imprecise guidance can be used in various combinations with each other and in combination with typical inputs to estimation (learner assessment outcomes). In our simulation study, we measured how each form of instructor guidance might decrease error in our recommender.

Recommendation

Using the learner mastery estimates and information about training activities, the recommender carries out two tasks. First, it combines information from multiple learning systems. Second, it recommends what to train next.

Different learning systems may agree or conflict on whether a learner has mastered a skill. Our exemplar recommender resolves conflicts using a moving widow with a fixed number of data points (here, five) combined with a logarithmic decay function. Learner mastery estimates are assumed to change over time and therefore more recent assessments should be given more weight. Note that it is possible in future work for deconfliction to use nonlinear combinations such as voting, reinforcement learning, or otherwise varying trust in different learning systems so that they influence the recommendation differently.

Within our full recommender example, two instructional designs are available. Well-defined learning objectives are trained with more top-down directed instruction, while ill-defined learning objectives use learner-directed recommendation that allows for some exploration and choice of topics. Because learner-directed recommendations are based on learner factors such as affect (boredom, confusion) and context (familiarity, novelty) we tested only the directed instruction recommendations with our simulated students. Recommendations using the learner-directed design will be tested in our human-participants study. The sequence of learning objectives is defined by instructional design metadata including prerequisites and hierarchical learning objectives. Therefore, recommendation of the next learning objective is derived from the deconflicted learner mastery estimates and can be tested in simulations.

METHODOLOGY

The learner modeling, instructor guidance, and deconflicted recommendation components were tested with a combination of historical and simulated student data. Three studies focusing on the three components helped to identify conditions that could cause problems in real-world use, predict how real learners will progress through training in multiple systems, and highlight capabilities of the exemplar recommender component that make it well suited to the future learning environment of recommendation across learning systems.

Historical data was drawn from publicly available records of learner interactions with a single intelligent tutoring system, Andes (Gertner & VanLehn, 2000). The data were generously made public by the Andes team via Learnsphere's online repository DataShop (Koedinger, C Stamper, Leber, & Skogsholm, 2013). Dataset number 263 was used, captured during a physics course with 69 students, 2,890 possible performance observations (called steps), 175,000 recorded observations, and 608 total hours of interactions. Skills in the dataset (or knowledge components) had a hierarchical structure. In order to use this dataset for our experiments, we used data that recorded individual performance observation outcomes. Given a variable amount of initial observations to form a prediction, we predicted whether the student would succeed or fail on the next opportunity for observation. The predictions could then be compared against ground truth of actual success or failure to determine accuracy of our predictions. We measured accuracy with receiver operating characteristic (ROC) curves and area under the ROC curves (AUC).

Simulated student data was also used when historical data was not available. Since the act of recommendation changes history, it was not possible to use recorded data alone for testing the full recommender. Instead, we created simple simulated students for this purpose. The simulated students were modeled with latent learner abilities that produced observable performance. Abilities were chosen from a bimodal distribution reflecting fast learners and slow learners. Fast learners had abilities chosen from a normal distribution with mean 0.8 and standard deviation 0.8, trimmed within the range [0,1]. Slow learners had abilities chosen from a normal distribution with mean 0.3 and standard deviation 0.8. Abilities in turn produced performance observations with independent and identically distributed successes and failures intended to combine learning, guess, and slip. For each population, we measured recommendation qualitatively with descriptions of the spread or difference between recommendations offered to each learner. When the recommendations are tested on real learners, it will be possible to measure quantitative measures such as which recommendations the learner accepts, learner-reported usefulness of recommendations, and inferred usefulness measured by whether learner performance improved after each recommendation.

There were some limitations in our simulation. The historical and simulated data contained only assessments. Training activities that do not produce assessments can give additional information to a recommender but are not included in the study for simplicity. Our simulation assumed that a training activity was always available when recommended. As a result, training activities may be offered more than once, but we do not model the change in impact after activities are repeated. We model activities that output multiple assessments as separate activities, so the real recommender is expected to improve its understanding faster (after fewer activities) than the simulation.

RESULTS AND DISCUSSION

We used the simulation method to test three aspects of the prototype recommender: learner modeling from unlabeled learning activities, instructor input to update knowledge about learners and activities, and recommendation that responds to changing learner and activity information to recommend what training to deliver next.

Study 1: Learner Modeling

We first evaluated the ability of our prototype recommender to model learners' skill mastery, which will drive the training recommendations for remediation or readiness for progression. Our first simulation study showed that a new learner model with online machine learning was able to simultaneously infer learner information (probability a learner can correctly perform a skill) and science of learning information (difficulty of correct performance for each skill). As a result, the learner model was able to predict ground truth performance of historical learners.

Figure 3 depicts overlapping ROC curves that describe the learner model component of the recommender. ROC curves show the tradeoff between true positives and false positives, indicating how well a prediction matched reality as established in study 1 from historical learner data. Multiple curves shown in Figure 3 reflect improving performance of the learner model over time. Each learner assessment opportunity added more information for online machine learning to improve its estimates about all learners and all skills.

Recorded historical data was divided into ten test runs with each one preceded by adding one tenth of the available training data. The accuracy of predictions in each test run was measured by the area under the curve (AUC). AUC for the first test run was 0.72, which indicates moderate accuracy. However, the second test run increased the AUC to 0.78 and the final test run achieved the best accuracy, with AUC = 0.84. The mean accuracy over all ten runs was 0.81. As a result of this study, we see that the learner model was able to simultaneously estimate skill difficulty and learner mastery of skills. Adding online machine learning made the learner model able to quickly improve before the end of a single course dataset. This supports the potential of similar learner model approaches to work well, even without a prior constructed model or calibration data, in the course of our upcoming human-participants study.

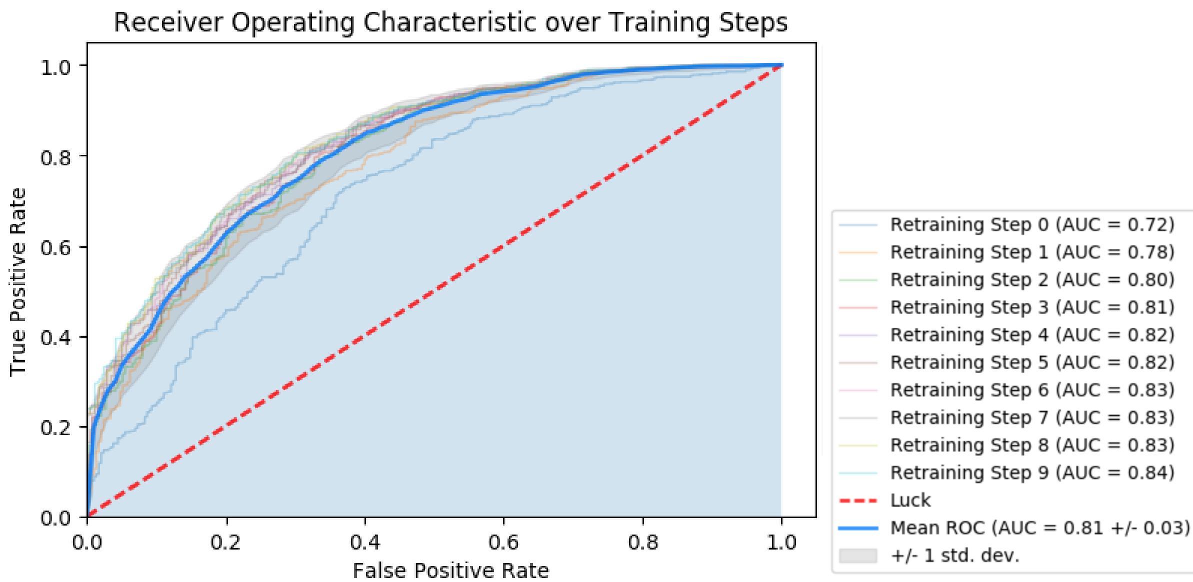

Figure 3: The machine learning component improves all learner mastery estimates each time any learner trains.

The learner model accuracy improved quickly in the first two test runs. Figure 4 depicts the accuracy of individual predictions across the recorded student population. The left side of Figure 4 shows output after one tenth of the training data was input, and the right side shows accuracy after two tenths of the training data. Accuracy without any training data was exactly 0.50, or a coin toss. However, after one test run we can see that the left-hand graph already displays a majority of predictions about individual learners are clustered around 0.2 in distance from the true value (as reflected by the AUC of 0.72 in the previous figure). After two runs, the majority of predictions have exactly zero difference from the true value. That is, the learner model not only becomes more accurate as depicted by the area under the ROC curve, but its accuracy is also reflected in more confident predictions about individual students.

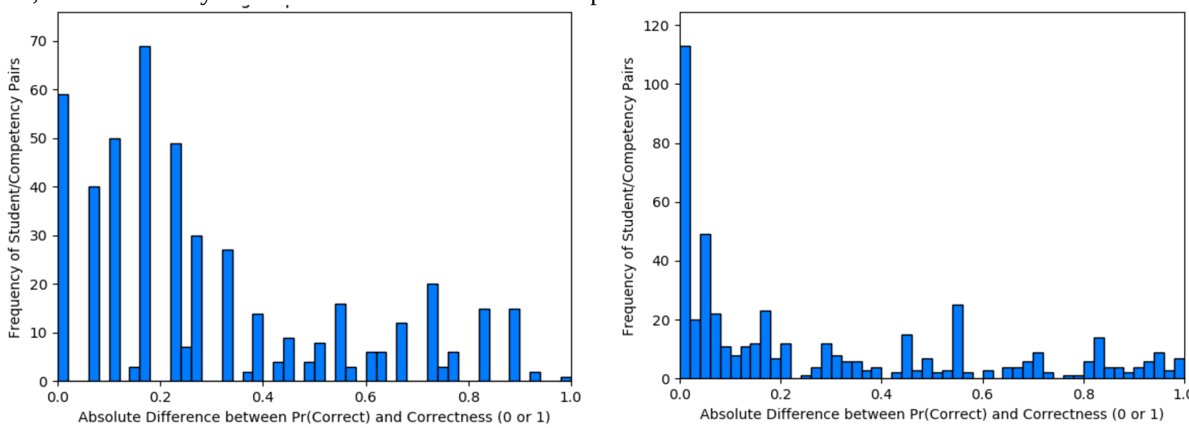

Figure 4: Increasing accuracy was matched by increasing confidence after 1 and 2 assessments (left, right respectively).

Regarding the accuracy of the learner model on predicting historical student data, we emphasize that the innovation here does not lie in extremely high accuracy. Indeed, higher accuracy may be gained by specialized algorithms that reflect knowledge about an individual training system and its contents. Instead, the value lies in the ability to find usefully accurate information simultaneously about the learners and the skills they are learning. This capability enables recommendation across multiple learning systems that need not share their underlying skill models. Similarly, we hypothesize extensions of this approach will help to quickly add newly authored content into training, maximize ability to leverage existing training, and detect obsolete content or unreliable assessments.

Study 2: Instructor Input

We next evaluated each of the four types of instructor guidance (see previous section) using the same historical dataset. We studied which types of guidance might best help quickly reduce error in a learner model. The findings in this section showed extremely promising reduction in error that quickly dwarfed machine learning alone. As a result, we find support for a human-machine teaming approach that can keep recommendation effective, across large numbers of training systems, even as individual learners come and go or learning content evolves over time.

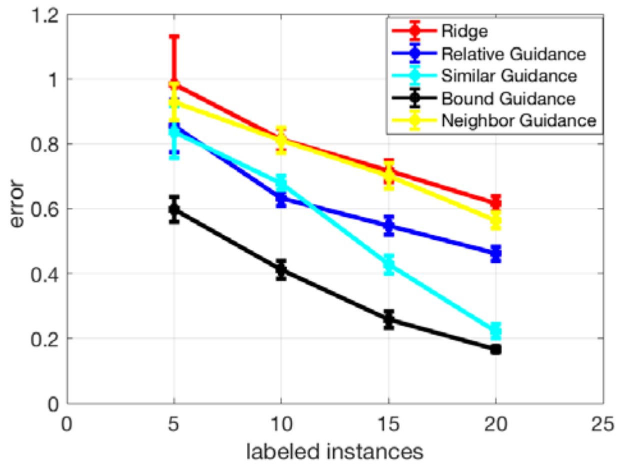

Figure 5: Guidance from instructors can reduce the error in mastery estimates compared to unguided estimates (Ridge).

Figure 5 shows the idealized performance of each different guidance type applied in turn. Here we start with a training set of 5 labeled instances where a learner's mastery of one skill or another is precisely given. To this training data set we add in more information (assessments and performance observations) as numbered on the x-axis in increments of 5. The baseline method (red line) is a ridge regression learner model, which reduces its overall error across skills as it receives more assessments. To produce the other lines, we add instructor guidance to the baseline learner model. We preface the assessment data with a fixed 25 examples of each type of instructor guidance. Figure 5 shows how much each form of guidance improves the learner model by the distance from the red line. We see that Neighborhood guidance offers the least amount of improvement whilst Bound guidance adds the most. Importantly, we see that the Similar guidance is diverging from the red line showing a compounding effect. That is, adding this type of guidance not only reduced estimate error, but also increased the speed of error reduction.

Figure 6 shows how different types of instructor guidance perform in combination with each other. Since the two plots are directly comparable, we see that all combinations of guidance in Figure 6 work better than the baseline method in Figure 5. The results are complex and combining types of guidance can produce a non-linear interaction between the forms of guidance. For example, though Relative guidance was a middle of the pack performer in Figure 5, combined with Neighbor guidance it becomes the worst, but with Similar and Bound guidance becomes near to the best performer. Perhaps most importantly, this experiment shows that using all types of guidance together produces the best result but combining two types – Similar guidance and Bound guidance – produces almost as little error as all four types combined. While the extremely low error produced by adding these types of instructor guidance should be verified with a human-participant study rather than relying on simulation alone, these results show great promise for the instructor guidance approach and also help to prioritize certain types of instructor guidance for researchers to study first.

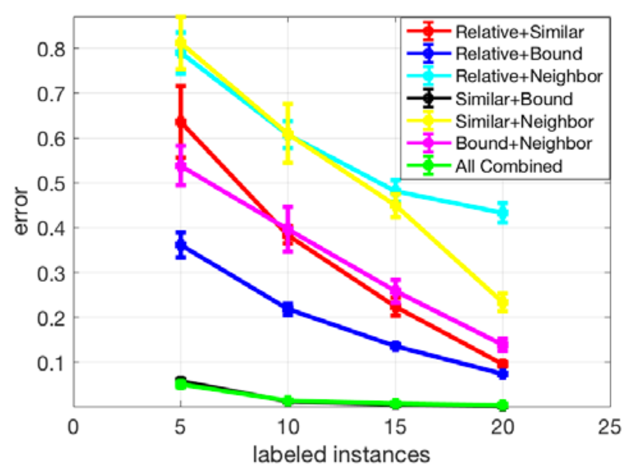

Figure 6: Different combinations of instructor guidance yield large error reduction in idealized circumstances.

Study 3: Recommendation

In our final simulation study, we examined the behavior of a recommender that combines inputs across learner models and available training activities. We did not perform a measurement of accuracy, but instead examined qualitative aspects of its recommendations to determine whether it could offer students individualized training activities. Future work will measure recommendation correctness with metrics that are not available in simulated students, such as: goal coverage, take rate, effective catalog size, and improved impact on learning.

Figure 7 compares a fixed sequence of training (top) against the exemplar recommendations (bottom). The columns indicate the passage of time as simulated by number of assessment opportunities. The qualitative difference between learner experiences in the two recommenders can be recognized in the diversity of colors per column. Each column in the top graph is mostly one color, indicating lockstep progression through a training sequence as defined in the baseline algorithm. In contrast, the bottom graph in Figure 7 shows that every column after the first one contains a mix of colors. This indicates each simulated student is able to move on quickly or receive extra help according to their estimated mastery over time. While the simulated students could not determine whether the recommender made good recommendations, evidence shows it did make varied recommendations that could differentiate learners in response to deconflicted inputs from multiple sources.

This study demonstrates a proof of the concept that a simple recommendation algorithm can join together or deconflict information from multiple learning systems. It also demonstrates that the recommender is capable of giving a wide range of outputs that are driven by and responsive to the simulated learner differences. Letting trainees progress through training quickly when they have mastered content can increase training efficiency, while giving trainees extra support to bring them up to mastery level can increase training effectiveness.

A finding with implications for the prototype recommender is that under the assumptions our simulation made, there is a long tail of trainees who need many learning activities to master a topic even though trainees on average master the topic earlier. In fact, the baseline algorithm completes training the last learner in half the time of the adaptive recommender. However, we hypothesize that the recommender is actually giving more chances to learn for those learners who need help, while the baseline is moving learners through the system before they have actually mastered the training. Spending large amounts of effort on the bottom few learners is a typical criticism of competency-based training. If our simulation assumptions reflect reality, then our upcoming experiment may need to prepare extra support in the form of more early activities. In the future a small subset of learners who need a large amount of help can also be identified, similar to an early warning system, and be remediated out of band or otherwise recommended onto a wholly different training path.

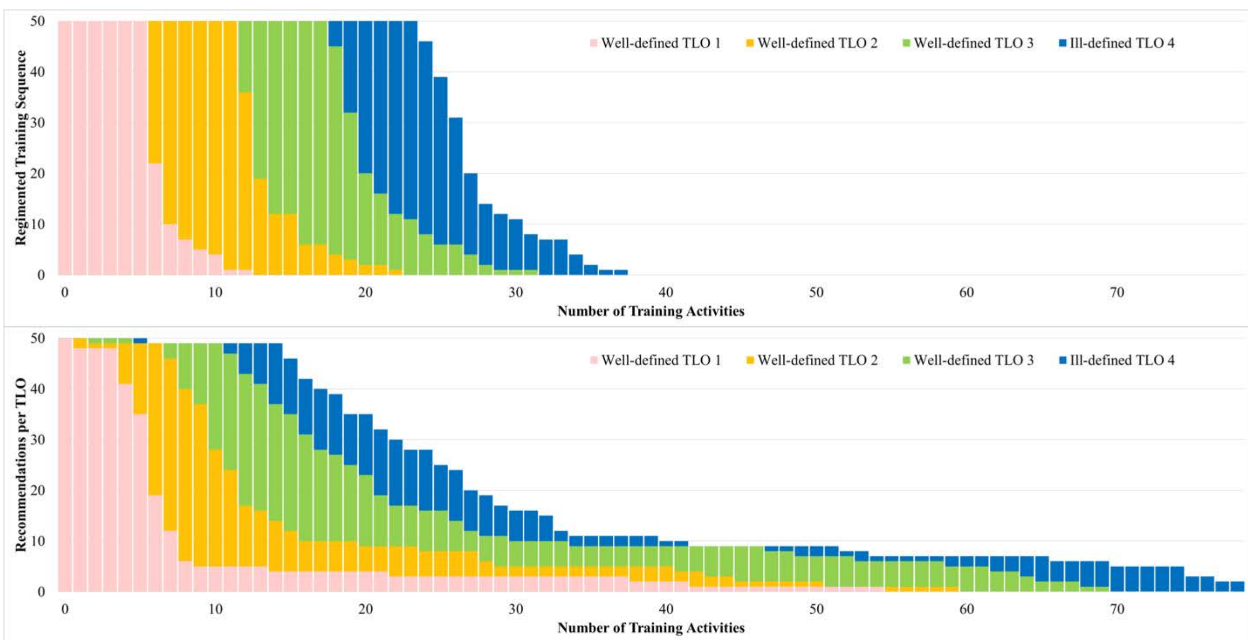

Figure 7: Recommender (bottom) lets learners progress early or receive extra help, compared to fixed sequencing (top).

CONCLUSION

Our studies with historical and simulated student data give confidence that recommendation across diverse learning systems is possible and show one approach to apply it. This approach should facilitate the successful development and deployment of an architecture that relies on wide-ranging metadata and machine learning to integrate learning systems without enforcing a specific instructional design. The components proposed here – learner modeling, instructor guidance, and deconflicted recommendation – will allow participants in such an architecture to understand how training systems work within different competency frameworks and models of learning, as well as their impact on learners' performance and the underlying skills. Based on our initial simulated studies, the exemplar components are operating well for evaluation in a human-participants study this year. In addition, we hope the description of an exemplar components captures the imagination of the research community, leading them to propose extensions and improvements on our initial prototype and address the challenges that will arise as modern training systems enter use.

ACKNOWLEDGEMENTS

This work is supported in part by the Office of Naval Research via contract N68335-17-C-0042. 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

ADL Initiative. (2017, November 15). xAPI-Spec Retrieved June 9, 2018, from https://github.com/adlnet/xAPI-Spec

Bottou, L. (2010). *Large-scale machine learning with stochastic gradient descent.* Paper presented at the 19th International Conference on Computational Statistics, Paris, France.

Burke, R. (2018, March 9). Growing to Win: Sailor 2025 – Navy's Strategy for People in our Future Fleet Retrieved June 1, 2018, from http://www.navy.mil/navydata/people/cnp/Burke/Resource/20180309-Sailor 2025 Navy Venues Article LONG Version clean (9MAR 0930).pdf

Davidson, I., Gress, A., & Folsom-Kovarik, J.T. (2017). *Transfer Learning in Intelligent Tutoring Systems: Results, Challenges and New Directions.* Paper presented at the Florida Artificial Intelligence Research Society (FLAIRS) meeting, Marco Island, FL.

Davidson, P.S. (2017). *Vision and Guidance for Ready Relevant Learning: Improving Sailor Performance and Enhancing Mission Readiness.*

Folsom-Kovarik, J.T., & Raybourn, E.M. (2016). *Total Learning Architecture (TLA) Enables Next-generation Learning via Meta-adaptation*. Paper presented at the Interservice/Industry Training, Simulation & Education Conference (I/ITSEC), Orlando, FL.

Gallagher, P.S., Folsom-Kovarik, J.T., Schatz, S., Barr, A., & Turkaly, S. (2017). *Total Learning Architecture development: A design-based research approach.* Paper presented at the Interservice/Industry Training, Simulation & Education Conference (I/ITSEC), Orlando, FL.

Gertner, A.S., & VanLehn, K. (2000). *Andes: A coached problem solving environment for physics.* Paper presented at the International conference on intelligent tutoring systems.

Gress, A., & Davidson, I. (2016). *Probabilistic Formulations of Regression with Mixed Guidance.* Paper presented at the 16th IEEE International Conference on Data Mining (ICDM).

Gress, A., & Davidson, I. (2018). *Human Guided Linear Regression with Feature-Level Constraints*. Paper presented at the 32nd AAAI Conference on Artificial Intelligence, New Orleans, LA.

Koedinger, K., C Stamper, J., Leber, B., & Skogsholm, A. (2013). *LearnLab's DataShop: A Data Repository and Analytics Tool Set for Cognitive Science* (Vol. 5).

MacLellan, C.J., Liu, R., & Koedinger, K.R. (2015). *Accounting for Slipping and Other False Negatives in Logistic Models of Student Learning*. Paper presented at the 8th International Conference on Educational Data Mining.

Mikolov, T., Chen, K., Corrado, G., & Dean, J. (2013). Efficient estimation of word representations in vector space.

Raybourn, E.M., Schatz, S., Vogel-Walcutt, J., & Vierling, K. (2017). *At the Tipping Point: Learning Science and Technology as Key Strategic Enablers for the Future of Defense and Security*. Paper presented at the Interservice/Industry Training Simulation and Education Conference (I/ITSEC), Orlando, FL.

Regan, D., Raybourn, E.M., & Durlach, P.J. (2013). Learner Modeling Considerations for a Personalized Assistant for Learning (PAL). In R. A. Sottilare, A. Graesser, X. Hu & H. Holden (Eds.), *Design Recommendations for Intelligent Tutoring Systems: Learner Modeling* (Vol. 1, pp. 217): U.S. Army Research Laboratory.

Sottilare, R.A., Brawner, K.W., Goldberg, B.S., & Holden, H.K. (2012). The generalized intelligent framework for tutoring (GIFT). Orlando, FL: US Army Research Laboratory Human Research & Engineering Directorate.